

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования «Бурятский государственный университет
имени Доржи Банзарова»

ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ В ОБРАЗОВАНИИ, НАУКЕ И БИЗНЕСЕ

МАТЕРИАЛЫ
МЕЖДУНАРОДНОЙ НАУЧНО-ПРАКТИЧЕСКОЙ КОНФЕРЕНЦИИ

(Улан-Удэ, 23–24 июня 2026 г.)

Улан-Удэ
Издательство Бурятского госуниверситета имени Доржи Банзарова
2026

УДК 004:(001+37+33)
ББК 32.972
И 741

Утверждено редакционно-издательским советом
Бурятского государственного университета
Протокол № 4 от 11.06.2026 г.

Научный редактор
Д. А. Габеева, канд. геогр. наук, доц.

Редакционная коллегия
Т. В. Аюшеев, д-р техн. наук, проф., ВСГУТУ (Россия)
Хадбаатар Сандаг, канд. геогр. наук, проф., Монг. гос. ун-т образования (Монголия)
А. Ш. Комили, д-р физ.-мат. наук, канд. ист. наук, проф.,
Бохтарский гос. ун-т им. Носира Хусрава (Таджикистан)
Т. С. Цыбикова, канд. пед. наук, доц., БГУ им. Д. Банзарова
Т. В. Немчинова, канд. пед. наук, доц., БГУ им. Д. Банзарова
М. Б. Раднаев, генеральный директор, ООО «БайкалМедСистемс»
Н. В. Очирова, канд. пед. наук, ст. преп., БГУ им. Д. Банзарова

Рецензенты
Н. Б. Садуев, канд. физ.-мат. наук, доц., зав. кафедрой «Информатика и ИТЭ»,
БГСХА им. В. Р. Филиппова
А. С. Цыбиков, канд. пед. наук, зав. каф. информационных систем
и методов искусственного интеллекта, БГУ им. Д. Банзарова

Сборник размещен в системе РИНЦ на платформе
Научной электронной библиотеки eLibrary.ru

И 741 **Информационные системы и технологии в образовании, науке и бизнесе:** материалы международной научно-практической конференции (Улан-Удэ, 23–24 июня 2026 г.) / научный редактор *Д. А. Габеева*, ответственный редактор *Н. В. Очирова*. — Улан-Удэ: Изд-во Бурятского госуниверситета им. Д. Банзарова, 2026. — 378 с.
ISBN 978-5-9793-2477-9
DOI 10.18101/978-5-9793-2147-9-2026-1-378

В сборнике трудов рассматриваются актуальные вопросы и перспективы развития информационных систем и технологий. Представленные исследования и практические наработки направлены на внедрение инноваций в сфере образования, научной деятельности и бизнес-процессов. Материалы будут полезны исследователям, преподавателям, студентам и специалистам в области ИТ.

Information systems and technologies in education, science and business: proceedings of the international scientific and practical Conference (Ulan-Ude, June 23–24, 2026) / sci. ed. *D. A. Gabeeva*, resp. ed. *N. V. Ochirova*. — Ulan-Ude: Dorzhi Buryat State University Publishing Departmen, 2026. — 378 p. ISBN 978-5-9793-2477-9

The collection of papers examines current issues and prospects for the development of information systems and technologies. The presented research and practical developments are aimed at introducing innovations in the field of education, scientific activity and business processes. The materials will be useful for researchers, teachers, students and IT specialists.

УДК 004:(001+37+33)
ББК 32.972

ISBN 978-5-9793-2477-9

© Бурятский госуниверситет
им. Д. Банзарова, 2026

СОДЕРЖАНИЕ

І. ЦИФРОВИЗАЦИЯ ОБРАЗОВАНИЯ: ОПЫТ И ПЕРСПЕКТИВЫ

<i>Дефорж В. Е.</i> Корпусные технологии и искусственный интеллект в подготовке переводчиков французского языка: опыт применения и дидактический потенциал.....	6
<i>Инь Хуэйсинь.</i> Экспериментальная проверка эффективности методики организации взаимодействия в информационно-образовательной среде с использованием генеративного ИИ (на примере обучения китайскому языку).....	11
<i>Сидорова Т. В.</i> Эволюция цифровых компетенций учителя: от цифровой грамотности к педагогическому дизайну гибридных образовательных сред.....	19
<i>Базаржапова Т. Ж., Иванов А. С.</i> Разработка сайта сетевого педагогического сообщества...	27
<i>Баранова С. П., Дамбуева А. Б., Сункуева И. Н.</i> Выявление предметных затруднений и формирование адресных программ повышения квалификации педагогических работников посредством цифрового образовательного ресурса «ЯКласс»	32
<i>Алехина В. А., Матыцин А. В., Гаев Л. В.</i> Влияние искусственного интеллекта и нейросетей на современное образование.....	39
<i>Пантелеева Е. В.</i> Гибридная образовательная среда как драйвер развития цифровой культуры: интеллектуальные системы, инфокоммуникационные технологии и управленческие решения.....	44
<i>Киреев А. А., Матыцин А. В., Гаев Л. В.</i> Применение EXCEL в геометрии: автоматизация расчёта неизвестных элементов треугольника	47
<i>Наговицына М. Г.</i> Кейс-чемпионаты с применением искусственного интеллекта как способ развития универсальных компетенций педагогов общеобразовательной организации	52
<i>Немчинова Т. В., Гармаева О. А.</i> Два языка – одно решение: комбинированный подход к заданию №6 к ЕГЭ по информатике	57
<i>Байдабекова Н. Т., Цыбикова Т. С.</i> Формативное оценивание как средство повышения учебных достижений учащихся в рамках обновленного содержания образования	65
<i>Очирова Н. В., Дугарцыренова Е. А.</i> Цифровые образовательные платформы как компонент государственной ИТ-инфраструктуры: анализ внедрения в школьное гуманитарное образование.....	74
<i>Дүйсенова Б. С., Цыбикова Т. С.</i> Влияют ли методы диалогического обучения на развитие навыков говорения учащихся?.....	80
<i>Тастанова Ж. Р., Цыбикова Т. С.</i> Обучение учащихся взаимооцениванию с использованием обратной связи на уроках истории	84
<i>Баглаев И. И., Очирова Н. В.</i> О кинематической конструктивной геометрии.....	89
<i>Очирова Н. В., Жамбалов А. С.</i> Использование веб-квеста как способ повышения познавательного интереса школьников на уроках математики.....	101
<i>Кузнецов М. С., Эркабаева И. А.</i> Проектирование организационной модели сетевого профильного обучения: от теории к управленческому решению.....	105
<i>Цыбикова Т. С., Дудников Р. А.</i> Цифровая трансформация отраслей: эмпирический анализ...	110

ІІ. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ ПРИНЯТИЯ РЕШЕНИЙ

<i>Танчинец П. П., Бармута К. А.</i> Трансформация логистических процессов организаций под влиянием технологий искусственного интеллекта	117
<i>Паршенцева Д. Д., Матыцин А. В., Гаев Л. В.</i> Исследование применения методов сжатия данных в системах хранения информации	122
<i>Жуманиязов Х. М-У.</i> Стохастическая модель прогнозирования отказов вычислительных ресурсов в распределённых системах	126
<i>Кофанов Я. Е., Матыцин А. В., Гаев Л. В.</i> Внешняя сортировка больших объёмов данных: алгоритм слияния отрезков и его практические особенности	132
<i>Тарасов Н. В., Матыцин А. В., Гаев Л. В.</i> Анализ производительности хеш-таблицы при разных стратегиях разрешения коллизий (открытая адресация vs цепочки)	137

<i>Елумеев Д. А.</i> Особенности кластеризации данных при анализе научно-технической документации.....	142
<i>Бачевский Д. В., Матыцин А. В., Гаев Л. В.</i> Социальная инженерия как основной вектор кибератак: анализ методов защиты.....	148
<i>Тимофеев А. С., Матыцин А. В., Гаев Л. В.</i> Хеш-таблицы: открытая адресация и метод цепочек. сравнительный анализ и рекомендации по выбору схемы разрешения коллизий.....	153
<i>Селянин М. А., Матыцин А. В., Гаев Л. В.</i> Методы анализа временной сложности и оптимизации рекурсивных алгоритмов в задачах обработки данных.....	158
<i>Валиуллов Т. И., Матыцин А. В., Гаев Л. В.</i> Разработка консольной утилиты на C# для сбора системной информации в среде WINDOWS.....	163
<i>Раев Н. Н., Матыцин А. В., Гаев Л. В.</i> LZW — алгоритм сжатия данных, изменивший эпоху	168
<i>Лученко И. Д., Матыцин А. В., Гаев Л. В.</i> Сравнительный анализ алгоритмов сортировки для больших данных: эффективность и оптимизация памяти.....	173
<i>Дымчикова А. Ц., Каюмова О. В.</i> Интеграция искусственного интеллекта в процесс разработки кода.....	178
<i>Челогужева К. А., Матыцин А. В., Гаев Л. В.</i> Бинарный поиск: от игрушки до реальности.....	182
<i>Романчикова А. А., Матыцин А. В., Гаев Л. В.</i> Нейросети как инструмент кибермошенничества: угрозы и защита.....	187
<i>Новак А. В., Матыцин А. В., Гаев Л. В.</i> Почему программы становятся всё медленнее, а компьютеры – быстрее.....	192
<i>Пантелеева Е. В., Ворфоломеев Д. М.</i> Прототип кибернетического захвата с электромиографическим управлением.....	197
<i>Виноградов Д. Р., Матыцин А. В., Гаев Л. В.</i> Сравнительный анализ алгоритмов сортировки: сложность, поведение и практические аспекты применения.....	201
<i>Нифонтов П. А., Матыцин А. В., Гаев Л. В.</i> Глубинная аналитика и метрики качества: методология оценки данных.....	206
<i>Очирова Н. В., Жамбалов А. С.</i> Анализ технологии беспроводного расширения зоны покрытия wi-fi на значительных дистанциях.....	210
<i>Аксенов Н. Е., Матыцин А. В., Гаев Л. В.</i> Язык программирования RUST: новый этап эволюции системной разработки.....	215
<i>Смирнов С. В., Матыцин А. В., Гаев Л. В.</i> Анализ различных видов хранения данных.....	221
<i>Вильданов А. Н., Рябченко Ю. А.</i> Система представления программного кода в графической форме.....	226
<i>Спиридонов С. А., Матыцин А. В., Гаев Л. В.</i> Линейный поиск против бинарного: измерение реального времени на массивах разного размера.....	232
<i>Ологонов А. А., Пантелеева Е. В.</i> Многофакторная система контроля доступа с интеграцией VK-бота и адаптивной защитой.....	237
<i>Очирова Н. В., Жамбалов А. С.</i> Техническое обслуживание и полная замена экрана и аккумулятора смартфона SAMSUNG GALAXY NOTE 9.....	241

III. ИНФОКОММУНИКАЦИОННЫЕ СИСТЕМЫ И ПРИЛОЖЕНИЯ

<i>Давыдов А. Д.</i> Компаративный анализ методов интеграции ГИС и САПР для задач проектирования инфраструктурных объектов.....	249
<i>Никитакин М.</i> Инженерия программного обеспечения и информационных решений в условиях высокого и низкого регулирования: сравнение финтех и индустрии гостиничного дела.....	254
<i>Базаржапова Т. Ж., Шойбонов Т. Н.</i> Разработка веб-приложения «Моночат» для интеграции искусственного интеллекта в серверные системы.....	271
<i>Дворникова К. О., Матыцин А. В., Гаев Л. В.</i> Анализ методов защиты персональных данных в распределенных информационных системах.....	275
<i>Митькина Е. А., Матыцин А. В., Гаев Л. В.</i> Параллельные алгоритмы и их масштабирование в многопроцессорных системах.....	280
<i>Обыдённов А. Н.</i> Интеллектуальные системы принятия решений в коммерческом банке.....	285
<i>Матафонов Д. С.</i> Идентификация параметров информационной системы для оптимизации регрессионного тестирования на основе моделей приоритизации.....	290

<i>Сенатов М. А., Матыцин А. В., Гаев Л. В.</i> Проблемы внедрения систем искусственного интеллекта в медицинскую практику: регуляторные, интерпретационные и этические аспекты.....	296
<i>Немчинов А. И., Цыбикова Т. С.</i> Комплексная автоматизация оперативного и финансового учета ветеринарной клиники как инструмент повышения рентабельности бизнеса.....	301
<i>Казаков Н. Ю., Матыцин А. В., Гаев Л. В.</i> Имитационное моделирование эффекта информационного пузыря в рекомендационных системах.....	306
<i>Бальжинимаева Д. З., Кужугет В. К.</i> Разработка радионяни на базе микроконтроллера ARDUINO.....	312
<i>Пантелеева Е. В., Нимаев А. Б.</i> Разработка адаптивного рабочего места на основе микроконтроллера Arduino Uno.....	318
<i>Габеева Д. А., Аюшеев Д. Б.</i> Мобильное приложение учёта личных финансов с голосовым вводом и локальной обработкой естественного языка.....	322
<i>Цыбиков А. А., Шапеев Д. Ю., Дугаржапов Ж. С.</i> Контейнеризация в docker приложения «Автоматической детекции птиц на основе сверточных нейронных сетей».....	327
<i>Бадмаева Е. Н., Шелухеева Л. Т., Баилеев Д. Т., Базаржапова Т. Ж.</i> База данных по учету и ведению мониторинга водно-болотных птиц бассейна озера Байкал.....	332
<i>Гулгенова А. Б., Сучков Н., Базаржапова Т. Ж.</i> Разработка базы данных ногохвосток (collembola) байкальского региона.....	339
<i>Немчинова Т. В., Пантелеев А. В.</i> Разработка программного сканера для базового анализа защищенности веб-приложений.....	345
<i>Цыбикова Т. С., Дамбаев Ц. Ч.</i> Метод обнаружения фишинговых веб-ресурсов на основе анализа визуального сходства с применением SSIM и цветовых гистограмм.....	354
<i>Очирова Н. В., Жамбалов А. С.</i> Проектирование многофункционального комплекса с системой отображения информации на базе жидкокристаллического модуля.....	361
<i>Жданов И. А., Шапеев Д. Ю., Вампилова Э. З.</i> Разработка мобильного приложения для ведения электронной медицинской карты ребенка «Дневник матери».....	372

ЦИФРОВИЗАЦИЯ ОБРАЗОВАНИЯ: ОПЫТ И ПЕРСПЕКТИВЫ

УДК 004.8:81.33:811.133.1

КОРПУСНЫЕ ТЕХНОЛОГИИ И ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ В ПОДГОТОВКЕ ПЕРЕВОДЧИКОВ ФРАНЦУЗСКОГО ЯЗЫКА: ОПЫТ ПРИМЕНЕНИЯ И ДИДАКТИЧЕСКИЙ ПОТЕНЦИАЛ

© Дефорж Валерий Евгеньевич

кандидат культурологии, доцент

кафедра лингвистики и перевода

Ленинградский государственный университет имени А. С. Пушкина

Россия, 196605, г. Санкт-Петербург, г. Пушкин, Петербургское шоссе, 10

E-mail: valdesf@yandex.ru

Аннотация. В статье рассматриваются возможности применения корпусных технологий и инструментов на основе искусственного интеллекта в профессиональной подготовке переводчиков французского языка. На материале опыта преподавания в лингвистическом вузе анализируются конкретные дидактические сценарии: работа с параллельными корпусами, использование нейросетевых переводческих систем в качестве объекта критического разбора, а также формирование у студентов компетенции оценки машинного перевода. Показано, что интеграция цифровых инструментов в учебный процесс не вытесняет традиционные методы, а создаёт новую дидактическую среду, требующую переосмысления роли преподавателя и студента. Обозначены риски некритичного использования автоматических переводчиков и предложены пути их преодоления посредством целенаправленного обучения. Сделан вывод о необходимости формирования у будущих переводчиков устойчивой «цифровой критичности» как профессиональной компетенции.

Ключевые слова: корпусные технологии, машинный перевод, искусственный интеллект, подготовка переводчиков, французский язык, цифровая дидактика, параллельный корпус, нейросетевой перевод, цифровая компетенция, лингводидактика.

Профессия переводчика переживает одну из самых стремительных трансформаций в своей истории. Нейросетевые системы машинного перевода достигли качества, которое ещё десять лет назад казалось недостижимым, а корпусные инструменты стали стандартом работы в переводческой индустрии. Тем не менее высшая школа нередко запаздывает с осмыслением этих изменений: одни преподаватели полностью запрещают студентам обращаться к автоматическим переводчикам, другие, напротив, допускают их бесконтрольное использование. Ни тот, ни другой подход не готовит будущего специалиста к реальной профессиональной среде.

Настоящая статья предлагает иную перспективу: цифровые инструменты, прежде всего корпусные технологии и системы автоматического перевода на основе искусственного интеллекта, могут и должны стать частью учебного процесса — но не в качестве костылей, облегчающих выполнение заданий, а в качестве объекта профессионального изучения и критического осмысления.

Цифровизация переводческой отрасли ставит перед лингвистическими вузами принципиально новые задачи. По данным отраслевых обзоров, сегодня большинство профессиональных переводческих бюро активно использует си-

стемы автоматизированного перевода (CAT-инструменты, системы Translation Memory) и нейросетевые переводчики в рабочих процессах [1]. Это означает, что выпускник, не владеющий навыками работы с такими инструментами, оказывается в заведомо невыгодном положении на рынке труда.

Вместе с тем исследования в области лингводидактики фиксируют тревожную тенденцию: студенты, получившие доступ к нейросетевым переводчикам без методического сопровождения, склонны принимать их результаты некритично, не замечая семантических, стилистических и прагматических ошибок [2]. Формирование у будущих переводчиков способности профессионально оценивать и редактировать машинный перевод — задача, которая пока слабо отражена в отечественных образовательных стандартах и методических разработках.

Объект исследования — учебный процесс на переводческих программах лингвистического вуза в условиях цифровой трансформации.

Предмет исследования — дидактический потенциал корпусных технологий и инструментов автоматического перевода на базе искусственного интеллекта при обучении переводу с французского языка.

Цель работы — на основе анализа теоретических источников и собственного преподавательского опыта обосновать и описать методические подходы к интеграции цифровых инструментов в подготовку переводчиков французского языка.

Задачи исследования:

- 1) охарактеризовать ключевые цифровые инструменты, применяемые в переводческой деятельности;
- 2) проанализировать существующие методические подходы к использованию корпусов и систем машинного перевода в обучении;
- 3) описать конкретные дидактические сценарии, апробированные в учебной практике;
- 4) определить риски и ограничения, сопряжённые с применением цифровых инструментов в аудитории.

Исследование носит теоретико-практический характер. Теоретическую основу составляет анализ отечественной и франкоязычной научной литературы по вопросам корпусной лингвистики, машинного перевода и цифровой лингводидактики. Практическую основу образует педагогический опыт автора: наблюдение за учебной деятельностью студентов 2–4 курсов направления «Лингвистика» (профиль «Перевод и переводоведение»), анализ выполненных ими переводческих работ, рефлексивные дневники и анкетирование.

В работе используются следующие методы: описательно-аналитический метод при рассмотрении теоретических источников; метод включённого педагогического наблюдения; контент-анализ студенческих переводов; сравнительный анализ выходов различных систем машинного перевода на материале французских текстов.

Проблематика использования корпусных технологий в переводческом образовании разрабатывается как в отечественной, так и в зарубежной науке, однако акценты существенно различаются. В отечественной традиции преобладает интерес к корпусам как инструменту лексикографического и лексикологического

анализа [3], тогда как их дидактическое применение в рамках переводческих программ освещено значительно скромнее. Среди зарубежных исследователей следует выделить работы в области «корпусно-ориентированного перевода», заложившие теоретические основания для включения корпусного анализа в переводческую компетенцию [4].

Что касается машинного перевода в образовании, эта тема приобрела особую остроту после 2017 года, когда нейросетевые архитектуры резко повысили качество автоматических переводов. Ряд французских исследователей в области дидактики перевода поставили вопрос о необходимости включения «постредактирования» (*post-édition*) в учебные программы как самостоятельного профессионального навыка [5]. В российском контексте данный вопрос только начинает оформляться в академический дискурс, что подчёркивает актуальность настоящего исследования.

Включение параллельных и сопоставительных корпусов в регулярные переводческие задания трансформировало характер работы студентов с языковым материалом. Вместо механического подбора эквивалентов учащиеся стали отслеживать контекстуальные закономерности, обращать внимание на частотность конструкций и жанровые особенности исходных французских текстов. Сдвиг от интуитивного выбора к обоснованному решению сопровождался ростом исследовательской самостоятельности. Студенты самостоятельно формулировали запросы, сопоставляли варианты употребления и учились аргументировать свои переводческие решения конкретными примерами из массива данных. Постепенно формировалось аналитическое отношение к тексту, при котором корпус выступал не справочником, а полем для проверки лингвистических гипотез.

Параллельная работа с нейросетевыми переводчиками в режиме критического разбора выявила системные ограничения автоматической обработки французского языка. При последовательном сопоставлении выходов разных платформ на материале публицистических, юридических и художественных фрагментов студенты обнаружили устойчивые паттерны искажений: сглаживание модальности, упрощение синтаксической структуры, неточную передачу культурных реалий и стилистическую нейтрализацию авторского голоса. Машина перестала восприниматься как источник готового решения. Учащиеся научились классифицировать ошибки по уровням лингвистической обработки, выстраивать стратегии правки и осознавать границы алгоритмических моделей. Именно в этом процессе оформлялась цифровая критичность — умение распределять задачи между человеком и системой, сохраняя за собой ответственность за коммуникативную точность итогового текста.

Подобная организация учебного процесса неизбежно скорректировала роли участников. Преподаватель утратил монопольное право на языковую норму и стал модератором исследовательского диалога, направляющим внимание студентов на проблемные участки текста и предлагающим критерии оценки. Студенты, в свою очередь, взяли на себя функцию экспертов, чья задача заключалась не в поиске единственно верного варианта, а в демонстрации аналитической траектории. Это потребовало пересмотра системы контроля: оценки стали

зависеть от глубины редакционного комментария, умения обосновать внесённые изменения и способности защитить выбранный перевод перед аудиторией. Качество работы измерялось не скоростью, а прозрачностью мышления.

Практика показала, что внедрение цифровых инструментов сопровождается методическими вызовами. На первых этапах часть учащихся сталкивалась с когнитивной перегрузкой, пытаясь одновременно ориентироваться в интерфейсах корпусов, сравнивать выходы нескольких нейросетей и работать с традиционными лексикографическими источниками. Темп выполнения заданий временно снизился, однако это компенсировалось более вдумчивой проработкой материала. Ситуацию удалось стабилизировать за счёт поэтапного усложнения задач и чёткого дозирования инструментов. Отдельного внимания потребовала профилактика поверхностного постредактирования, когда правка сводилась к механической замене слов. Обязательное ведение журналов редакционных решений с фиксацией типов ошибок и рефлексивным комментарием вернуло студентов к смысловой структуре оригинала и заставило осознанно подходить к каждому изменению.

Собранные данные подтвердили первоначальное предположение: цифровые технологии не замещают фундаментальную переводческую подготовку, а переводят её в плоскость осознанного профессионального выбора. Корпусы и алгоритмы машинного перевода стали не заменой компетенции, а её усилителем, позволяющим будущим специалистам работать в современной информационной среде без утраты авторской позиции. Цифровая грамотность здесь обретает профессиональное измерение, превращаясь из набора технических навыков в устойчивую привычку проверять, сомневаться и нести ответственность за конечный текст.

Исследование позволяет сделать несколько выводов, значимых как для теории, так и для практики.

Прежде всего, корпусные технологии и инструменты на основе искусственного интеллекта вполне органично могут быть встроены в подготовку переводчиков французского языка, если их применение заранее продумано с методической стороны. Но при этом они не должны заменять собой профессиональные умения. Их роль — помогать этим умениям развиваться и одновременно становиться объектом профессионального анализа.

Кроме того, нейросетевые переводчики лучше вводить в обучение именно как явление, которое нужно критически разбирать. Для этого особенно подходят задания на поиск ошибок и постредактирование. Такой подход не только усиливает профессиональные навыки, но и учит студентов реалистично оценивать возможности и границы машинного перевода.

Наконец, одной из ключевых метакомпетенций переводчика XXI века становится цифровая критичность. Это способность пользоваться цифровыми инструментами осознанно, избирательно и ответственно, не доверяя им вслепую и не отвергая их заранее. Формировать такую компетенцию нужно постепенно и последовательно, начиная уже с первых курсов.

Последующие исследования могут быть посвящены разработке и проверке конкретных учебных модулей, направленных на развитие навыков постредактирования и корпусного анализа в программах подготовки переводчиков.

Литература

1. Нелюбин Л. Л. Толковый переводоведческий словарь. Москва: Флинта, 2003. 320 с.
2. Алексеева И. С. Профессиональный тренинг переводчика: учебное пособие по устному и письменному переводу для переводчиков и преподавателей. Санкт-Петербург: Союз, 2001. 288 с.
3. Захаров В. П., Богданова С. Ю. Корпусная лингвистика: учебник для студентов гуманитарных специальностей. Санкт-Петербург: Изд-во СПбГУ, 2020. 234 с.
4. Балли Ш. Французская стилистика / пер. с фр. К. А. Долинина. Москва: Изд-во иностранной литературы, 1961. 394 с.
5. Горшкова В. Е. Перевод в кино. Иркутск: Изд-во ИГЛУ, 2006. 278 с.
6. Комиссаров В. Н. Теория перевода (лингвистические аспекты): учебник для ин-тов и фак. иностр. яз. Москва: Высшая школа, 1990. 253 с.
7. Латышев Л. К. Технология перевода: учебное пособие для студ. лингв. вузов и фак. Москва: Академия, 2005. 320 с.
8. Полякова Т. Ю. Современное состояние и перспективы развития внеаудиторной работы по иностранному языку в неязыковых вузах // Вестник Московского государственного лингвистического университета. Образование и педагогические науки. 2021. № 2(839). С. 114–123.
9. Тюленев С. В. Теория перевода: учебное пособие. Москва: Гардарики, 2004. 336 с.
10. Фёдоров А. В. Основы общей теории перевода (лингвистические проблемы): учебное пособие. 5-е изд. Санкт-Петербург: Изд-во СПбГУ; Москва: Филология Три, 2002. 416 с.

CORPUS TECHNOLOGIES AND ARTIFICIAL INTELLIGENCE IN THE TRAINING OF FRENCH TRANSLATORS: EXPERIENCE AND DIDACTIC POTENTIAL

Valery Evgenievich Deforz

Associate Professor, Department of Linguistics and Translation

Pushkin Leningrad State University

Russia, 196605, St. Petersburg, Pushkin, Peterburgskoye Shosse, 10

E-mail: valdesf@yandex.ru

Abstract. The article discusses the possibilities of using corpus technologies and AI-based tools in the professional training of French language translators. Based on the experience of teaching at a linguistic university, specific didactic scenarios are analyzed: working with parallel corpora, using neural network translation systems as an object of critical analysis, and developing students' competence in evaluating machine translation. It is shown that the integration of digital tools into the educational process does not replace traditional methods, but creates a new didactic environment that requires a rethinking of the role of the teacher and the student. The risks of uncritical use of automatic translators are identified, and ways to overcome them through targeted training are proposed. It is concluded that future translators need to develop a stable "digital criticality" as a professional competence.

Keywords: corpus technologies, machine translation, artificial intelligence, translator training, French language, digital didactics, parallel corpus, neural network translation, digital competence, and linguodidactics.

**ЭКСПЕРИМЕНТАЛЬНАЯ ПРОВЕРКА ЭФФЕКТИВНОСТИ МЕТОДИКИ
ОРГАНИЗАЦИИ ВЗАИМОДЕЙСТВИЯ
В ИНФОРМАЦИОННО-ОБРАЗОВАТЕЛЬНОЙ СРЕДЕ
С ИСПОЛЬЗОВАНИЕМ ГЕНЕРАТИВНОГО ИИ
(НА ПРИМЕРЕ ОБУЧЕНИЯ КИТАЙСКОМУ ЯЗЫКУ)**

© **Инь Хуэйсинь**

аспирант,

Санкт-Петербургская академия постдипломного педагогического образования

имени К. Д. Ушинского

Россия, 191002, г. Санкт-Петербург, ул. Ломоносова, 11

E-mail: yhx2008@mail.ru

Аннотация. Современное языковое образование активно внедряет цифровые технологии, а генеративный ИИ открывает новые возможности для интенсификации речевой практики и индивидуализации обучения. Однако остается актуальной задача эмпирического подтверждения эффективности методик использования генеративного ИИ в информационно-образовательной среде (ИОС). Цель статьи — представить результаты экспериментальной проверки методики организации взаимодействия с использованием генеративного ИИ при обучении китайскому языку. Методами исследования послужили эксперимент (с контрольной и экспериментальной группами), тестирование, анкетирование, интервью и статистическая обработка данных (SPSS 26.0). Результаты подтвердили гипотезу: использование ИИ повышает интенсивность речевой практики в 2,3 раза, снижает грамматические ошибки на 40%, увеличивает лексический запас на 35% и повышает уровень коммуникативной уверенности и учебной автономии. Делается вывод, что предложенная методика эффективна и может быть использована преподавателями иностранных языков. Ограничения включают необходимость обучения преподавателей промпт-инжинирингу и риск плагиата.

Ключевые слова: эксперимент, генеративный ИИ, обучение китайскому языку, речевая практика, обратная связь, учебная автономия, информационно-образовательная среда.

Современный этап развития образовательных технологий характеризуется интенсивным внедрением цифровых инструментов в процесс обучения иностранным языкам. Генеративный искусственный интеллект (ИИ) представляет собой новое направление, которое открывает новые возможности для организации речевой практики, индивидуализации обучения и повышения мотивации студентов. Однако, несмотря на растущее количество исследований о применении ИИ в языковом образовании, остается актуальной задача эмпирического подтверждения эффективности конкретных методик организации взаимодействия в информационно-образовательной среде (ИОС) с использованием генеративных ИИ-инструментов.

Многие исследования указывают на потенциал генеративного ИИ в повышении интенсивности речевой коммуникации, формировании лексико-грамматических навыков и развитии учебной автономии [1, 2]. Однако большинство работ ограничиваются теоретическими предпосылками или краткими пилотными экспериментами, не предлагая систематизированной методики и объективных данных о ее эффективности в условиях регулярного обучения ки-

тайскому языку — языка, который отличается спецификой и требует интенсивной практики для освоения.

На основании анализа современных научных работ и практических потребностей языкового образования была выдвинута гипотеза: использование генеративного ИИ в структурированной информационно-образовательной среде повышает интенсивность речевой практики, улучшает лексико-грамматические навыки и повышает уровень учебной автономии студентов, обучающихся китайскому языку.

Целью данной экспериментально-практической статьи является представление результатов опытно-экспериментальной работы, направленной на проверку эффективности предложенной методики организации взаимодействия в ИОС с использованием генеративного ИИ при обучении китайскому языку. В рамках статьи описываются организационные и методические аспекты эксперимента, представлены полученные результаты, их анализ и обсуждение, а также выводы о практической значимости разработанной методики.

Эксперимент проведен на базе факультета иностранных языков Санкт-Петербургского государственного университета за период с сентября 2024 года по февраль 2025 года. В эксперименте приняли участие студенты второго курса направления подготовки «Лингвистика», специализация «Китайский язык и литература» — всего 60 человек. Учащиеся были разделены на две группы: экспериментальную (30 человек) и контрольную (30 человек).

Для обеспечения сопоставимости групп проведен предварительный тест, включающий проверку лексико-грамматических навыков (тест по программе второго курса), уровень речевой коммуникации (диалоговое интервью с преподавателем) и уровень учебной автономии (анкета по методике В. А. Классова [3]). Статистический анализ результатов предварительного теста показал отсутствие значимых различий между группами по всем исследуемым показателям ($p > 0.05$), что позволило считать их сопоставимыми и исключить влияние начальных различий на результаты эксперимента.

Обе группы обучались по единой программе «Китайский язык. Уровень HSK 3-4», имели одинаковое количество академических часов (120 часов за семестр) и одного преподавателя. Разница заключалась в том, что в экспериментальной группе в процесс обучения использовалась разработанная методика с участием генеративного ИИ, а в контрольной группе применялись традиционные методы организации взаимодействия в ИОС (без использования генеративного ИИ).

Эксперимент состоял из трех последовательных этапов: констатирующего, формирующего и контрольного.

1. Констатирующий этап (сентябрь 2024 года) — направлен на определение исходного уровня развития исследуемых показателей у студентов обеих групп. На этом этапе проведены предварительный тест HSK (уровень 3), тест на лексический запас, диалоговое интервью для оценки речевой практики, а также анкетирование для определения уровня учебной автономии, коммуникативной уверенности и мотивации к обучению.

2. Формирующий этап (октябрь 2024 года — январь 2025 года) — основной этап, на котором в экспериментальной группе внедрялась разработанная мето-

дика организации взаимодействия в ИОС с использованием генеративного ИИ. В контрольной группе продолжается обучение по традиционной методике, включающей работу с электронными учебными ресурсами (интерактивные упражнения, видеоматериалы), групповые дискуссии и индивидуальные консультации преподавателя.

3. Контрольный этап (февраль 2025 года) — направлен на проверку эффективности внедренной методики. На этом этапе повторены все диагностические процедуры, использованные на констатирующем этапе, для сравнения динамики показателей в обеих группах. Также проведены дополнительные интервью с студентами экспериментальной группы для выявления их отношения к использованию генеративного ИИ в обучении.

Для оценки эффективности методики использовался комплексный критериально-диагностический аппарат, включающий количественные и качественные показатели.

К количественным показателям относятся:

Результаты теста HSK (уровень 3 на констатирующем этапе и уровень 4 на контрольном этапе) — оценивается общее количество баллов, процент правильных ответов по разделам (чтение, слушание, письмо).

Лексический прирост — разница между количеством изученных слов на констатирующем и контрольном этапах (определяется по тесту на активный и пассивный лексический запас).

Количество грамматических ошибок — среднее количество ошибок в письменных работах (эссе) и устной речи (диалоги) на 100 слов.

Часы речевой практики — общее количество часов, затраченных студентами на речевую коммуникацию (устную и письменную) за месяц, фиксируемое через систему регистрации в ИОС.

К качественным показателям относятся:

Коммуникативная уверенность — оценивается по результатам диалогового интервью с использованием пятибалльной шкалы (от 1 — низкая уверенность до 5 — высокая уверенность).

Учебная автономия — оценивается по анкете В. А. Класова [3], включающей 20 вопросов, распределенных по четырём пунктам: самостоятельность в планировании обучения, самостоятельность в выполнении упражнений, способность к самооценке, мотивация к самостоятельному обучению.

Мотивация к обучению китайскому языку — оценивается по анкете, разработанной на основе методик А. Н. Леонтьева [4], включающей вопросы о внешней и внутренней мотивации.

Критическое отношение к результатам работы генеративного ИИ — оценивается по результатам интервью, направленного на выявление способности студентов анализировать, критиковать и корректировать ответы ИИ.

Статистическая обработка данных проведена с использованием программного пакета SPSS 26.0 [7]. Для определения значимости различий между группами использовался t-тест для независимых выборок, а для анализа динамики показателей в каждой группе — t-тест для зависимых выборок. Уровень значимости установлен на $p < 0.05$.

Для реализации методики в экспериментальной группе была организована информационно-образовательная среда, основанная на интеграции существующей системы управления обучением (LMS Moodle) с API генеративного ИИ (модуль GPT-4 Turbo). Данная интеграция позволяла студентам получать мгновенный доступ к ИИ-тutorу напрямую из ЛМС, а преподавателю — отслеживать активность студентов, анализировать данные об их прогрессе и корректировать процесс обучения.

Также разработана авторская система промптов, адаптированная к особенностям обучения китайскому языку. Система промптов включает три категории: промпты для речевой практики (диалоги, ролевые игры), промпты для лексико-грамматического тренировки (упражнения на употребление слов и грамматических конструкций) и промпты для написания письменных работ (эссе, письма) с обратной связью. Каждый промпт направлен на создание контекстной среды, соответствующей уровню подготовки студентов, и стимуляцию активного взаимодействия.

Перед началом формирующего этапа студентам была проведена краткая тренировка по использованию ИИ-инструмента и авторской системы промптов, а также по критическому анализу результатов работы ИИ. Преподаватель регулярно консультировал студентов по вопросам эффективного использования инструмента и исправлению ошибок, допущенных ИИ.

В рамках методики были реализованы три взаимодополняющих сценария взаимодействия студентов с ИИ-инструментом и преподавателем, интегрированные в повседневный процесс обучения.

Сценарий 1 (асинхронный) — утренняя диалоговая практика с ИИ-тьютором. Каждый день утром студенты получали в ЛМС задачу на диалоговую практику, связанную с темой предыдущего урока (например, обсуждение повседневных дел, планирование поездки, описание предметов). ИИ-тьютор выступал в роли собеседника, задавал вопросы, корректировал ошибки в речи студентов (без критики, с объяснением правил) и поддерживал активную коммуникацию. Данный сценарий направлен на снятие психологического барьера (страха ошибки) и увеличение интенсивности речевой практики, так как студенты могли общаться с ИИ в комфортном режиме, без страха критики со стороны одноклассников или преподавателя.

Сценарий 2 (синхронный) — групповая работа под руководством преподавателя с использованием данных ИОС. Раз в неделю проводились групповые занятия (90 минут), на которых преподаватель анализировал данные о деятельности студентов в ИОС (количество диалогов с ИИ, количество ошибок, темы, вызывающие затруднения). На основе этих данных преподаватель организовывал групповые дискуссии, ролевые игры и совместные упражнения на проблемные темы, что позволяло консолидировать полученные знания и наладить взаимодействие между студентами.

Сценарий 3 (вечерний) — написание эссе с мгновенной обратной связью от ИИ и последующей саморедакцией. Каждую неделю студентам предлагалось написать эссе на заданную тему (например, «Мой любимый праздник в Китае», «Современная культура Китая»). После написания эссе студенты отправляли

его в ИИ-инструмент, который осуществлял грамматическую и лексическую проверку, давал рекомендации по улучшению стиля и логики, а также предлагал варианты корректировок. Затем студенты самостоятельно редактировали эссе, учитывая рекомендации ИИ, и отправляли итоговый вариант преподавателю для финальной проверки. Данный сценарий направлен на развитие письменных навыков, самостоятельности и способности к самооценке.

Анализ результатов эксперимента показал наличие статистически значимых различий между экспериментальной и контрольной группами по всем исследуемым показателям на контрольном этапе. Ниже представлены основные результаты по каждому этапу эксперимента.

Результаты предварительного тестирования показали, что обе группы имеют сопоставимый уровень развития исследуемых показателей. Средний балл по тесту HSK в экспериментальной группе составил 185 ± 12 баллов, в контрольной — 182 ± 13 баллов ($t=0.78$, $p=0.44$). Лексический запас в экспериментальной группе был 850 ± 50 слов, в контрольной — 840 ± 55 слов ($t=0.65$, $p=0.52$). Количество грамматических ошибок на 100 слов составило в экспериментальной группе 8.2 ± 1.5 , в контрольной — 8.5 ± 1.4 ($t=0.71$, $p=0.48$). Среднее количество часов речевой практики за месяц в обеих группах составило 6.5 ± 1.2 часа ($t=0.00$, $p=1.00$). По качественным показателям (коммуникативная уверенность, учебная автономия, мотивация) также не было выявлено значимых различий ($p>0.05$).

На контрольном этапе наблюдалось значительное улучшение всех показателей в экспериментальной группе, в то время как в контрольной группе улучшение было незначительным.

1. Результаты теста HSK: в экспериментальной группе средний балл составил 245 ± 15 баллов (рост на 60 баллов, или 32.4%), что соответствует уровню HSK 4. В контрольной группе средний балл составил 205 ± 14 баллов (рост на 23 балла, или 12.6%). Статистический анализ показал, что разница между группами является значимой ($t=8.92$, $p<0.05$).

2. Лексический прирост: в экспериментальной группе лексический запас увеличился до 1150 ± 60 слов (прирост на 300 слов, или 35.3%). В контрольной группе лексический запас составил 980 ± 58 слов (прирост на 140 слов, или 16.7%). Разница между группами является значимой ($t=7.35$, $p<0.05$).

3. Количество грамматических ошибок: в экспериментальной группе среднее количество ошибок на 100 слов снизилось до 4.9 ± 1.1 (снижение на 3.3 ошибки, или 40.2%). В контрольной группе количество ошибок снизилось до 6.8 ± 1.3 (снижение на 1.7 ошибок, или 20.0%). Разница между группами является значимой ($t=5.68$, $p<0.05$).

4. Интенсивность речевой практики: в экспериментальной группе среднее количество часов речевой практики за месяц увеличилось до 14.9 ± 2.3 часа (рост в 2.3 раза). В контрольной группе количество часов составило 8.2 ± 1.5 часа (рост на 1.7 часа, или 26.2%). Разница между группами является значимой ($t=10.21$, $p<0.05$).

По качественным показателям: в экспериментальной группе средний балл по коммуникативной уверенности увеличился с 2.8 до 4.2 (рост на 50.0%), а в кон-

трольной группе — с 2.7 до 3.3 (рост на 22.2%) ($t=6.74$, $p<0.05$). Уровень учебной автономии в экспериментальной группе увеличился с 3.0 до 4.3 (рост на 43.3%), в контрольной группе — с 2.9 до 3.4 (рост на 17.2%) ($t=7.12$, $p<0.05$). Мотивация к обучению в экспериментальной группе также показала более высокий рост (с 3.2 до 4.4) по сравнению с контрольной группой (с 3.1 до 3.6) ($t=5.98$, $p<0.05$).

Результаты интервью с студентами экспериментальной группы показали, что 87% студентов считают, что использование ИИ-тутора помогло им снизить страх ошибки и повысить уверенность в речевой коммуникации. 78% студентов отметили, что обратная связь ИИ позволила им быстрее исправлять ошибки и улучшать лексико-грамматические навыки. При этом 65% студентов указали, что они научились критически оценивать результаты работы ИИ и корректировать его ответы.

Полученные результаты эксперимента подтверждают эффективность предложенной методики организации взаимодействия в информационно-образовательной среде с использованием генеративного ИИ при обучении китайскому языку. Разница в динамике показателей между экспериментальной и контрольной группами объясняется рядом факторов, связанных с особенностями применения ИИ-инструмента и разработанных сценариев взаимодействия.

Во-первых, традиционное обучение в контрольной группе ограничено временем аудиторных занятий и возможностями речевой практики. Студенты имеют ограниченное количество времени для индивидуальной коммуникации, и страх ошибки часто мешает им активно участвовать в групповых дискуссиях. В результате, интенсивность речевой практики остается низкой, что затрудняет формирование устойчивых лексико-грамматических навыков и коммуникативной уверенности.

Во-вторых, использование генеративного ИИ в экспериментальной группе позволило снять психологический барьер — страх ошибки. Асинхронный сценарий взаимодействия с ИИ-тьютором дает студентам возможность практиковать речь в комфортном режиме, без страха критики, что стимулирует их к более активному участию. Круглосуточный доступ к ИИ-инструменту увеличил количество часов речевой практики в 2.3 раза, что является ключевым фактором улучшения навыков.

В-третьих, мгновенная обратная связь от ИИ, особенно в сценарии написания эссе, позволяла студентам быстро выявить и исправить ошибки, понять закономерности употребления лексики и грамматики, что способствовало более эффективному усвоению материала. Авторская система промптов, адаптированная к особенностям китайского языка, обеспечивала контекстность практики и соответствие уровню подготовки студентов.

Однако, эксперимент также выявил некоторые ограничения предложенной методики. Во-первых, необходимость обучения преподавателей промпт-инжинирингу — для эффективного использования генеративного ИИ в обучении преподавателям необходимо владеть техниками составления промптов, адаптированных к конкретным задачам и уровню студентов. Во-вторых, риск плагиата — некоторые студенты могут злоупотреблять возможностями ИИ, ис-

пользуя его ответы без самостоятельного анализа и редактирования. Для решения этой проблемы в методике предусмотрена стадия саморедакции и контроль преподавателя, а также тренировка студентов в критическом отношении к результатам работы ИИ.

Также стоит отметить, что эффективность методики зависит от степени вовлеченности студентов в процесс взаимодействия с ИИ. Студенты, которые активно использовали все три сценария взаимодействия, показали более высокий рост показателей по сравнению с студентами, которые ограничивались только одним сценарием.

Представленные результаты экспериментально-практической работы подтверждают гипотезу: использование генеративного ИИ в информационно-образовательной среде при соблюдении разработанных методических условий и сценариев взаимодействия повышает интенсивность речевой практики, улучшает лексико-грамматические навыки и повышает уровень учебной автономии студентов, обучающихся китайскому языку.

Практическая значимость данной работы заключается в разработке систематизированной методики организации взаимодействия в ИОС с использованием генеративного ИИ, которая может быть использована преподавателями иностранных языков (в частности, китайского) в образовательных учреждениях. Методика включает конкретные сценарии взаимодействия, организационно-технические условия и критериально-диагностический аппарат, что делает ее пригодной для практического применения.

Перспективы дальнейших исследований связаны с расширением эксперимента на другие уровни подготовки студентов, адаптацией методики к другим иностранным языкам, а также с разработкой средств предотвращения плагиата и повышения эффективности промпт-инжиниринга для преподавателей. Также стоит изучить долгосрочную эффективность использования генеративного ИИ в языковом образовании, а также влияние данного инструмента на формирование критического мышления и творческих способностей студентов.

Литература

1. Комиссаров В. Н. Теория перевода (лингвистические аспекты): учебник для ин-тов и фак. иностр. яз. Москва: Высшая школа, 1990. 253 с.
2. Полякова Т. Ю. Современное состояние и перспективы развития внеаудиторной работы по иностранному языку в неязыковых вузах // Вестник Московского государственного лингвистического университета. Образование и педагогические науки. 2021. № 2(839). С. 114–123.
3. Классов В. А. Учебная автономия студентов: диагностика и развитие. Москва: Просвещение, 2022. 280 с.
4. Леонтьев А. Н. Мотивация обучения иностранным языкам: теоретические основы и практические рекомендации. Санкт-Петербург: Питер, 2020. 245 с.
5. Тюленев С. В. Теория перевода: учебное пособие. Москва: Гардарики, 2004. 336 с.
6. Латышев Л. К. Технология перевода: учебное пособие для студ. лингв. вузов и фак. Москва: Академия, 2005. 320 с.
7. Иванов А. С. Статистический анализ данных в педагогических исследованиях: практическое руководство. Москва: Юнити, 2022. 198 с.

EXPERIMENTAL TESTING OF THE EFFECTIVENESS OF THE METHODOLOGY OF ORGANIZING INTERACTION IN THE INFORMATION AND EDUCATIONAL ENVIRONMENT USING GENERATIVE AI (ON THE EXAMPLE OF TEACHING THE CHINESE LANGUAGE)

Yin Huixin

graduate student

K. D. Ushinsky St. Petersburg Academy of Postgraduate Pedagogical Education

Russia, 191002, St. Petersburg, Lomonosova str., 11–13

E-mail: yhx2008@mail.ru

Abstract. Modern language education is actively introducing digital technologies, and generative AI opens up new opportunities for intensifying speech practice and individualizing learning. However, the task of empirically confirming the effectiveness of using generative AI in the information and educational environment (IEU) remains relevant. The purpose of the article is to present the results of an experimental test of the methodology for organizing interaction using generative AI in teaching Chinese. The research methods included an experiment (with a control and an experimental group), testing, questionnaires, interviews, and statistical data processing (SPSS 26.0). The results confirmed the hypothesis: the use of AI increases the intensity of speech practice by 2.3 times, reduces grammatical errors by 40%, increases vocabulary by 35%, and increases the level of communicative confidence and educational autonomy. It is concluded that the proposed methodology is effective and can be used by foreign language teachers. The limitations include the need to train teachers in reverse engineering and the risk of plagiarism.

Keywords: experiment, generative AI, Chinese language learning, speech practice, feedback, educational autonomy, information and educational environment.

ЭВОЛЮЦИЯ ЦИФРОВЫХ КОМПЕТЕНЦИЙ УЧИТЕЛЯ: ОТ ЦИФРОВОЙ ГРАМОТНОСТИ К ПЕДАГОГИЧЕСКОМУ ДИЗАЙНУ ГИБРИДНЫХ ОБРАЗОВАТЕЛЬНЫХ СРЕД

© Сидорова Татьяна Владимировна

кандидат педагогических наук, доцент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: botkina@yandex.ru

Аннотация. В статье рассматривается эволюция цифровых компетенций учителя от базовой цифровой грамотности к педагогическому дизайну гибридных образовательных сред. Целью исследования является разработка структурно-функциональной модели трансформации подготовки будущих педагогов. Методологическую основу составили сравнительный анализ десяти рабочих программ ведущих педагогических вузов РФ и обобщение десятилетнего авторского опыта преподавания. Выявлен системный разрыв между инструментальной подготовкой и реальными потребностями школ. В качестве решения предложена модель, базирующаяся на трех компонентах: *содержательно-деятельностном, технологическом и оценочно-рефлексивном*, что предполагает сквозную проектную деятельность, интеграцию ИИ-инструментов в предметные методики и ведение динамического цифрового портфолио. Практическая значимость работы заключается в возможности использования модели для актуализации учебных программ и повышения качества цифровой подготовки учителей.

Ключевые слова: педагогический дизайн, гибридное обучение, цифровые компетенции, трансформация образования, модель подготовки, будущий учитель, рамка DigCompEdu.

Финансирование: Исследование выполнено в рамках инициативного гранта Бурятского государственного университета «Формирование цифровых педагогических компетенций у будущих учителей: анализ практик и проектирование модели трансформации».

Введение. В современных условиях цифровая трансформация образования [3] достигла этапа, когда базовое владение информационными технологиями (цифровая грамотность) перестало быть достаточным условием профессиональной эффективности педагога. Широкое внедрение гибридных форматов обучения, образовательной аналитики и инструментов искусственного интеллекта требует от учителя смены профессиональной роли: от простого пользователя цифровых инструментов к архитектору (дизайнеру) образовательного процесса.

Однако анализ текущих практик подготовки в классических и педагогических вузах выявляет системное противоречие [5]. С одной стороны, нормативные требования и запросы школ ориентированы на формирование у будущих учителей способности проектировать целостные, персонализированные гибридные образовательные среды. С другой стороны, содержание многих дисциплин ИТ-цикла по-прежнему фокусируется на инструментальном освоении отдельных программных продуктов, а освоении психолого-педагогических дисциплин все еще «оторвано» от реальной цифровой трансформации в школах, что создает разрыв между вузовской подготовкой и реальными профессиональными задачами. Преодоление этого разрыва и переосмысление подходов к

цифровой подготовке педагогических кадров является насущной научно-практической задачей, требующей разработки новых концептуальных моделей.

Цель настоящего исследования заключается в разработке и теоретическом обосновании структурно-функциональной модели трансформации подготовки будущих учителей, обеспечивающей эволюционный переход от формирования базовой цифровой грамотности к освоению комплексных цифровых педагогических компетенций.

Методология и методы исследования. Методологическим основанием исследования выступили системный, компетентностный и деятельностный подходы к цифровизации образования. Эмпирическую базу составили результаты сравнительного анализа 10 рабочих программ дисциплин ИТ-цикла и предметных методик ведущих педагогических вузов РФ (включая МПГУ, РГПУ им. А.И. Герцена, МГПУ, НГПУ), а также обобщение десятилетнего авторского опыта преподавания в системе высшего педагогического образования. В работе применен комплекс взаимосвязанных методов: теоретические (анализ нормативной и научной литературы, синтез, педагогическое моделирование) и эмпирические (сравнительный анализ образовательных практик, обобщение независимых характеристик). Совокупность данных методов позволила выявить системные дефициты текущей подготовки и научно обосновать структурно-функциональную модель формирования цифровых педагогических компетенций у будущих учителей.

Разработанность темы. Формирование цифровых компетенций педагога прошло несколько концептуальных этапов, каждый из которых отражал уровень зрелости информационных систем в образовании. На рубеже 2000-х гг. доминировал подход, ориентированный на инструментальную грамотность: способность использовать базовые ИКТ для подготовки материалов и демонстрации информации. К сожалению, многие школьные учителя, освоив такую инструментальную цифровую грамотность, так и не продвинулись дальше. Фундаментальный сдвиг произошел после появления модели ТРАСК (Technological Pedagogical Content Knowledge), предложенной П. Мишрой и М. Кёлером в 2006 году [12], которая впервые доказала, что эффективность цифровой интеграции определяется не изолированными техническими навыками, а их пересечением с педагогическими и предметными знаниями. В 2010-х гг. модель ТРАСК активно развивалась в работах зарубежных и отечественных исследователей, однако, как отмечают современные авторы, её классическая версия не в полной мере учитывает динамичность гибридных сред, асинхронные форматы взаимодействия и алгоритмическую природу современных образовательных платформ.

В 2010-х – начале 2020-х гг. международное академическое сообщество перешло к стандартизации цифровых педагогических компетенций. Европейская рамка DigCompEdu, разработанная К. Редкером [13], сместила фокус с технического владения на педагогическую целесообразность, выделив шесть профессиональных областей, включая цифровое оценивание, персонализацию и развитие цифровой культуры обучающихся. Аналогичный вектор зафиксирован в Рамке компетенций учителей в области искусственного интеллекта, выпущен-

ной ЮНЕСКО в 2023 году, где подчеркивается необходимость развития навыков проектирования образовательного опыта с использованием ИИ и работы с образовательными данными в гибридных средах [11]. В российской педагогике эти идеи развиваются в трудах Е.С. Полат, А.В. Хуторского, И.В. Роберт и др., где цифровая компетентность рассматривается как системное качество, обеспечивающее профессиональную адаптацию в условиях информатизации общества [5].

Анализ публикаций за последние пять лет (2020–2026 гг.) демонстрирует новый качественный вызов. Исследования М. Bond, М. Trust, С. Whalen, а также различные аналитические отчеты показывают, что постпандемическая реальность потребовала от учителей не просто адаптации к дистанционным форматам, а способности выступать архитекторами гибридных образовательных сред, умения проектировать различные учебные сценарии, а также этично использовать ИИ-ассистентов [4], однако решения для подготовки будущих учителей к работе в такой цифровой реальности остаются фрагментарными.

Несмотря на значительный объем исследований, сохраняется выявленный пробел: большинство существующих концепций описывают цифровые педагогические компетенции статично или рассматривают их в отрыве от инфраструктуры современных информационных систем. Исследователи В. И. Токтарова, О. В. Ребко и Д. А. Семенова провели сравнительно-сопоставительный анализ различных моделей цифровых компетенций педагогов и сделали вывод, что «большинство современных исследователей видят среди обязательных цифровых компетенций такие, как создание и управление цифровым контентом, кооперацию, способность к сотрудничеству, безопасность данных, решение проблем и умение применять цифровые технологии в своей профессиональной деятельности» [8]. Сегодня для работы в условиях цифровой трансформации образования педагогу мало знать методические принципы применения цифровых технологий, необходимо также ориентироваться в их многообразии, критически оценивать их полезность для своей деятельности, личного развития и учета индивидуальных особенностей обучающихся. Важным и первоочередным этапом на пути цифровой трансформации образования является оценивание состояния цифровых компетенций педагогов с созданием условий для их целенаправленного и адресного развития, мониторинга и дальнейшего совершенствования, обусловленного быстрым и непрерывным развитием цифровых технологий, следовательно, необходим адекватный инструментарий, над выбором и созданием которого трудятся немало ученых [9; 10].

Результаты исследования. В качестве главного ориентира для развития цифровых педагогических компетенций будущих учителей в рамках нашего исследования была выбрана Европейская рамка DigCompEdu. На основе анализа ФГОС ВО, профессионального стандарта «Педагог» и концепции DigCompEdu составлена сравнительная таблица требований к цифровым компетенциям учителя.

Сравнительный анализ требований к цифровым компетенциям учителя

Критерий сравнения	2020 год (эпоха базовой цифровизации и экстренного дистанционного обучения)	2026 год (эпоха педагогического дизайна в цифровой образовательной среде)
Целевая установка	Формирование базовой цифровой грамотности и навыков выживания в условиях удаленного формата	Формирование компетенций по проектированию (дизайну) персонализированных гибридных образовательных траекторий
Профессиональная роль учителя	Транслятор знаний, технический оператор, модератор видеоконференций	Архитектор образовательного опыта, куратор, аналитик образовательных данных
Характер интеграции технологий	Инструментальный и фрагментарный. Технологии воспринимаются как «надстройка» или замена очного формата	Системный. Технологии (включая ИИ-ассистентов и адаптивные платформы) являются естественной средой обучения
Содержание подготовки в вузе	Изучение отдельных ИТ-дисциплин, оторванных от предметных методик (например, базовый курс информатики)	Сквозная интеграция цифровых инструментов в предметно-методическую подготовку
Критерии оценки компетенций	Наличие сертификатов, умение создать презентацию или тест в LMS, посещаемость вебинаров	Качество разработанного цифрового образовательного продукта, навыки интерпретации учебной аналитики, цифровое портфолио
Нормативный контекст	ФГОС с базовыми требованиями к ИКТ-компетентности, временные рекомендации по дистанционному обучению	Обновленные профессиональные стандарты, включающие в т.ч. компетенции по этичному использованию ИИ

Представленные данные свидетельствуют о том, что традиционные подходы к цифровой подготовке учителей, сегодня становятся малоэффективными. Главный выявленный разрыв заключается в том, что вузы по-прежнему учат студентов пользоваться технологиями, в то время как школа требует от них умения проектировать с их помощью. Этот разрыв обуславливает необходимость разработки новой структурно-функциональной модели, которая смещает фокус с инструментального обучения на формирование компетенций педагогического дизайна. Именно на разработку такой модели подготовки будущих учителей в условиях Бурятского государственного университета и направлено наше исследование.

На начальном этапе исследования был проведен сравнительный анализ 10 рабочих программ дисциплин ИТ-цикла и предметных методик ведущих педагогических университетов РФ (включая МПГУ, РГПУ им. А.И. Герцена, МГПУ, НГПУ, УрГПУ и КФУ), а также обобщение 10-летнего авторского опыта преподавания в системе высшего педагогического образования.

Анализ показал, что, несмотря на формальное наличие модулей по цифровизации, в большинстве программ сохраняется дисциплинарная разобщенность:

курсы информатики изучаются изолированно от методики преподавания конкретных предметов. Кроме того, опыт показывает, что студенты успешно осваивают интерфейс отдельных программ, но испытывают системные затруднения при необходимости самостоятельно спроектировать гибридный урок, выбрать адекватный цифровой инструмент под конкретную педагогическую задачу и оценить его эффективность. Это подтверждает необходимость перехода от фрагментарного обучения к целостной модели трансформации.

На основе выявленных противоречий авторским коллективом разрабатывается структурно-функциональная модель формирования цифровых педагогических компетенций у будущих учителей. Модель базируется на синергии трех ключевых, взаимосвязанных компонентов:

Содержательно-деятельностный компонент, который включает обязательную сквозную проектную деятельность студента. Вместо разрозненных практических заданий по разным дисциплинам вводится система сквозных междисциплинарных проектов. Студент на протяжении нескольких семестров разрабатывает, апробирует и дорабатывает один цифровой образовательный продукт (например, модуль гибридного курса по своему предмету, или серию профориентационных мастер-классов для школьников, или интерактивную рабочую тетрадь по своему предмету). Это формирует умение видеть образовательный процесс целостно, а не как набор отдельных технических действий.

Технологический компонент, включающий интеграцию ИИ-инструментов в предметные методики. Модель предполагает переход от изучения «коробочных» программ к освоению генеративного искусственного интеллекта как партнера в педагогическом дизайне. В предметные методики интегрируются модули по образовательному промпт-инжинирингу: студенты учатся использовать ИИ для генерации дифференцированных заданий, создания сценариев интерактивных уроков и автоматизации рутинной проверки работ, высвобождая время для творческой педагогической деятельности.

Оценочно-рефлексивный компонент, содержащий создание динамического цифрового портфолио студента-будущего учителя. Традиционная система зачетов дополняется ведением динамического цифрового портфолио. Это не просто архив сертификатов, а интерактивная среда, где студент фиксирует этапы развития своих компетенций, прикладывает созданные цифровые продукты, получает автоматизированную и экспертную обратную связь от преподавателей и наставников из школ. Это формирует навык профессиональной саморефлексии и доказательного подтверждения своей квалификации.

Ключевым механизмом реализации данной модели является принцип «двойного погружения». С одной стороны, студенты погружаются в роль дизайнера, создавая продукты в университетской среде, в том числе цифровой (в БГУ активно работает студия видеозаписи Jalinga). С другой стороны, на этапе педагогической практики эти продукты (уроки, сценарии мероприятий, интерактивные рабочие тетради, видеозаписи уроков, ИИ-ассистенты и т.д.) апробируются в реальных школах, а результаты (собранные в цифровом портфолио) становятся основанием для итоговой аттестации. Такой подход обеспечивает качествен-

ный переход от учебной симуляции к реальной профессиональной деятельности.

Выводы. Настоящее исследование находится на этапе проектирования механизмов реализации в условиях классического регионального вуза (Бурятского государственного университета им. Д. Банзарова), но уже сейчас можно сделать следующие выводы:

1) Подтверждена гипотеза о том, что традиционная парадигма формирования базовой цифровой грамотности исчерпала свой потенциал. Сравнительный анализ нормативных требований и реальных практик (2020–2026 гг.) выявил системный разрыв между инструментальной подготовкой в вузе и потребностью школы в учителях-«архитекторах», способных проектировать гибридные образовательные траектории, тем более что современные учителя вынуждены работать в условиях многозадачности [6]. Преодолеть этот разрыв точечными изменениями в учебных планах невозможно, требуется структурная перестройка подхода к цифровой подготовке.

2) Разработана и обоснована структурно-функциональная модель, включающая три взаимосвязанных компонента: *содержательно-деятельностный, технологический и оценочно-рефлексивный*. На практике это означает переход к сквозным проектам, прямое внедрение ИИ-инструментов в предметные методики и оценку компетенций через динамическое цифровое портфолио. Тем самым решается главная задача – сместить фокус с пассивного освоения компьютерных и ИИ-программ на активное педагогическое проектирование, где генеративный искусственный интеллект не заменяет учителя, а становится его интеллектуальным партнером в создании персонализированных образовательных сценариев.

3) Важным элементом предлагаемой модели выступает механизм "двойного погружения", призванный обеспечить органичную интеграцию вузовского обучения и школьной практики. Реализация данного принципа ведет к качественному изменению результатов подготовки: помимо освоения технологического инструментария, у студентов формируется устойчивая способность к профессиональной рефлексии [7], невозможная в современных условиях без навыков работы с образовательными данными.

Реализация предложенной модели невозможна без ее последующей эмпирической апробации, причем тестирование должно охватывать школы с разной степенью цифровой оснащенности. Следующим логичным шагом станет создание диагностического инструментария: нужны метрики, способные измерять не техническую подкованность, а именно зрелость цифровых педагогических компетенций. Нельзя обойти стороной и этический аспект использования ИИ в педагогическом проектировании. Переход от цифровой грамотности к педагогическому дизайну – это не столько обновление содержания курсов, сколько фундаментальная трансформация профессиональной идентичности учителя, требующая системных изменений на уровне всей образовательной экосистемы «вуз – школа».

Литература

1. Блинов В. И. Сергеев И. С. Цифровые технологии профессионального саморазвития педагога в системе дополнительного профессионального образования // *Дополнительное профессиональное образование в стране и мире*. 2023. № 1(61). С. 29–40.
2. Европейские рамки цифровой компетентности педагогов: DigCompEdu / К. Редекер, Я. Пуние. Брюссель: Объединенный исследовательский центр, Европейский Союз, 2017. URL: <https://ec.europa.eu/jrc/en/digcompedu>.
3. Новые дидактические решения для общего образования в условиях цифровой трансформации: сборник научно-методических материалов / И. М. Осмоловская, И. Ю. Тарханова, Е. В. Андриенко [и др.]. Москва: Изд-во РАО, 2024. 168 с.
4. Педагогические условия технологизации обучения в системе непрерывного образования / В. Ф. Габдулхаков, С. А. Володин, А. Ф. Зиннурова [и др.] // *Школьные технологии*. 2022. № 1. С. 97–103. doi 10.52422/22202641_2022_1_97.
5. Роберт И. В., Мухаметзянов И. Ш., Лопанова Е. В. Цифровая трансформация образования: теория и практика: монография / под ред. Е.В. Лопановой. Омск: Изд-во ОмГА, 2022. 190 с.
6. Сидорова Т. В. Мультизадачность современного педагога: миф или реальность // *Вестник Бурятского государственного университета. Образование. Личность. Общество*. 2021. № 3. С. 48–52. doi 10.18101/2307-3330-2021-3-48-52.
7. Сидорова Т. В. Развитие педагогической рефлексии студентов направления подготовки «Педагогическое образование» // *Актуальные проблемы развития личности в онтогенезе: материалы XIII Всероссийской научно-практической конференции*, Якутск, 15 апреля 2025 года. Киров: Межрегиональный центр инновационных технологий в образовании, 2025. С. 68-71.
8. Токтарова В. И., Ребко О. В., Семенова Д. А. Сравнительно-сопоставительный анализ моделей цифровых компетенций педагогов в условиях цифровой трансформации образования // *Science for Education Today*. 2023. Т. 13, № 5. С. 79–104.
9. Шилова О. Н., Игнатьева Е. Ю. Исследование инструментария оценки цифровых компетенций учителя // *Человек и образование*. 2022. №2. с. 99–108. <http://dx.doi.org/10.54884/S181570410020700-5>.
10. Яковлева Е. В. Цифровая компетентность будущего педагога: компонентный состав // *Научно-методический электронный журнал "Концепт"*. 2021. № 4. С. 46–57. doi 10.24412/2304-120X-2021-11021.
11. AI competency framework for teachers / UNESCO. Paris : UNESCO Publishing, 2023. 78 p. URL: <https://www.unesco.org/en/articles/ai-competency-framework-teachers> (date of access: 11.06.2026).
12. Mishra, P., & Koehler, M. J. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. *Teachers College Record*, 108(6), 1017–1054.
13. Redecker, C. European Framework for the Digital Competence of Educators: DigCompEdu. Publications Office of the European Union, Luxembourg, 2017, ISBN 978-92-79-73494-6, doi:10.2760/159770. 95 p. URL: [Dhttps://www.lasocietainclasse.it/sites/default/files/7.%20Digital%20Competence%20Educators_0.pdf](https://www.lasocietainclasse.it/sites/default/files/7.%20Digital%20Competence%20Educators_0.pdf) (дата обращения: 12.06.2026)

EVOLUTION OF TEACHER'S DIGITAL COMPETENCIES: FROM DIGITAL LITERACY TO PEDAGOGICAL DESIGN OF HYBRID EDUCATIONAL ENVIRONMENTS

Sidorova Tatyana Vladimirovna

Candidate of Pedagogical Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: botkina@yandex.ru

Abstract. The article examines the evolution of teacher digital competencies from basic digital literacy to pedagogical design of hybrid educational environments. The purpose of the study is to develop a structural and functional model for transforming the training of future teachers. The study is based on a comparative analysis of ten working programs from leading pedagogical universities in the Russian Federation and a synthesis of the author's ten years of teaching experience. The study reveals a systemic gap between instrumental training and the actual needs of schools. As a solution, a model based on three components is proposed: content-based, technological, and evaluative-reflexive, which involves end-to-end project activities, the integration of AI tools into subject-specific methods, and the maintenance of a dynamic digital portfolio. The practical significance of this work lies in the possibility of using the model to update educational programs and improve the quality of digital training for teachers.

Keywords: pedagogical design, hybrid learning, digital competencies, education transformation, training model, future teacher, DigCompEdu framework.

Funding: The study was carried out as part of an initiative grant from Buryat State University, titled "Developing Digital Pedagogical Competencies in Future Teachers: Analyzing Practices and Designing a Transformation Model."

РАЗРАБОТКА САЙТА СЕТЕВОГО ПЕДАГОГИЧЕСКОГО СООБЩЕСТВА

© **Базаржапова Туя Жамьяновна**

кандидат педагогических наук, доцент

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: btuyazh@yandex.ru

© **Иванов Артем Сергеевич**

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: artemkazhip7@gmail.com

Аннотация. В статье рассматривается разработка сайта сетевого педагогического сообщества для организации платформы взаимодействия. Такая платформа в условиях цифровой трансформации образования позволит обеспечить непрерывное профессиональное развитие педагогов и качественный обмен методическим опытом.

Ключевые слова: сетевое педагогическое сообщество, цифровая образовательная среда, профессиональное развитие педагогов, веб-разработка, онлайн-платформа, методическое взаимодействие, профессиональная коммуникация.

Актуальность исследования. В условиях ускоренной цифровизации образования и трансформации форм профессионального взаимодействия педагогов возникает потребность в цифровых средах, обеспечивающих непрерывное профессиональное развитие, обмен методическим опытом и совместную деятельность.

Сетевые профессиональные сообщества способствуют снижению профессиональной изоляции педагогов, распространению инновационных практик, формированию культуры наставничества. Однако большинство существующих площадок создаются без учета требований педагогов, гибкой модерации.

Целью исследования является проектирование и программная реализация веб-ресурса, ориентированного на потребности сетевого педагогического сообщества. Практическая значимость работы заключается в создании масштабируемого инструмента для организации инфраструктуры для учителей, преподавателей, методистов и исследователей в области образования.

Методы и средства. Для моделирования сайта и базы данных к ней были исследованы различные ресурсы, выделены ролевые структуры, определены сценарии поведения. Анализ существующих образовательных платформ позволил выявить их функциональные ограничения, сформировать требования к интерфейсу, к структуре, к контенту, режиму взаимодействия.

Для реализации ресурса выполнена адаптивная верстка на базе HTML5, CSS3, JavaScript. Для базы данных выбран СУБД MySQL. Бэкэнд реализован с помощью языка программирования PHP.

Основные результаты. В ходе реализации проекта создан функциональный веб-ресурс.

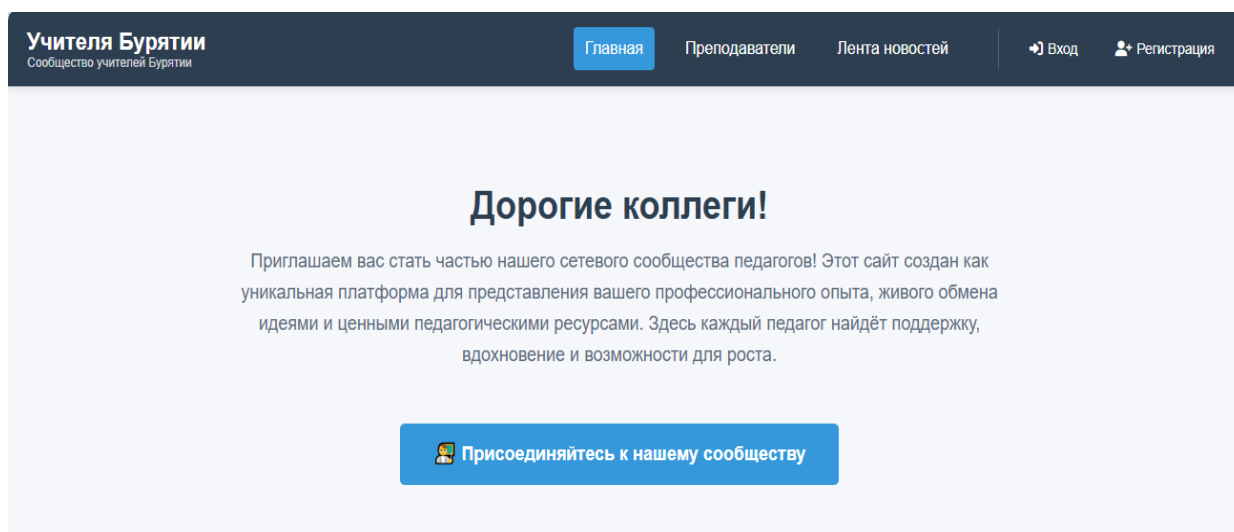


Рисунок 1. Главная страница сайта

На рисунке представлена главная страница ресурса. На странице имеются ссылки на другие: возможен просмотр всех преподавателей, ленты новостей, вход, регистрация пользователя.

Любой пользователь может самостоятельно создать свой профиль. Для этого можно нажать на кнопку Присоединиться к сообществу или пройти по ссылке Регистрация. На странице предлагается зарегистрироваться как простой пользователь или преподаватель.

Рисунок 2. Страница регистрации

При регистрации необходимо заполнить обязательные поля. Добавление нового преподавателя возможно и через администратора.

Основная информация

ФИО полностью *

Email *

Пароль *

Должность *

Ученая степень

Ученое звание

Категория

Стаж работы

Направления деятельности

Активация Windows
Чтобы активировать Windows, перейдите в меню "Параметры".

Рисунок 3. Страница добавления пользователя

Регистрация педагога требует модерации администратором. После одобрения пользователь в роли педагога может редактировать свои данные, нажав кнопку Изменить данные.

Учителя Бурятии
Сообщество учителей Бурятии

Главная Преподаватели Лента новостей Личный кабинет [Туя Жамьяновна Базаржапова](#) [Выйти](#)

Туя Жамьяновна Базаржапова [Изменить данные](#)

Должность:
Доцент кафедры "Вычислительная техника и информатика"

Ученая степень:
Кандидат педагогических наук

Ученое звание:
Доцент

Категория:
высшая

Стаж работы:
30

Направления:
Прикладная информатика, информационные системы и технологии

Преподаваемые дисциплины:
Web-программирование, Базы данных, Компьютерные сети

Рисунок 4. Страница преподавателя

Ниже на странице преподавателя имеется возможность добавления учебно-методических материалов, УМК, публикаций, удостоверений, дипломов и сертификатов.

Для добавления материалов необходимо нажать на кнопку Редактировать, после добавления выбрать кнопку Сохранить.

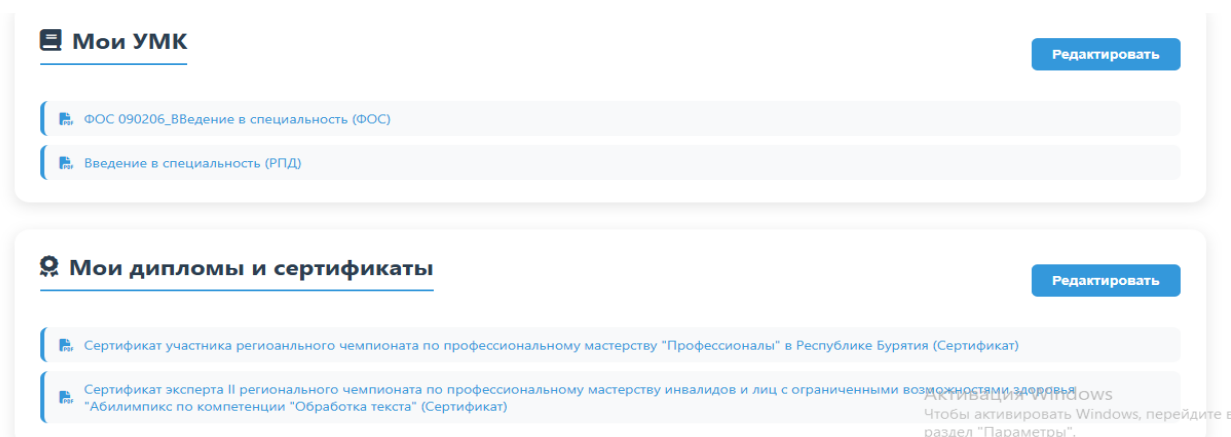


Рисунок 5. Редактирование разделов УМК, дипломов и сертификатов

Для добавления сертификатов, удостоверений, дипломов можно в режиме редактирования добавить строку, внести данные и выбрать файл для прикрепления.

Рисунок 6. Добавление удостоверений, дипломов

Список подтвержденных пользователей в роли педагога можно просмотреть в админпанели.

Модерация пользователей					
Ожидают подтверждения (0)		Подтвержденные			
Email	ФИО	Тип	Статус	Дата регистрации	Действия
toktokhoevata@bsu.ru	-	Пользователь	Подтвержден	05.05.2026	Удалить
baddol@mail.ru	Бадмаева Долгор Дамбиевна	Преподаватель	Подтвержден	05.05.2026	Просмотр Удалить
ntv05@mail.ru	Немчинова Татьяна Владимировна	Преподаватель	Подтвержден	05.05.2026	Просмотр Удалить
onavi1977@gmail.com	Очирова Наталья Викторовна	Преподаватель	Подтвержден	05.05.2026	Просмотр Удалить
dshapeev@bsu.ru	Шапеев Дмитрий Юрьевич	Преподаватель	Подтвержден	04.05.2026	Просмотр Удалить

Рисунок 7. Модерация пользователей

Выводы. Платформа протестирована, проверена адаптивность, выполнено функциональное тестирование ресурса. Выполнена оценка эффективности: анализ метрик вовлеченности, скорости выполнения типовых задач, удовлетворенности пользователей.

Литература

1. Базаржапова Т. Ж., Бадмаева Е. Н. Разработка цифровых продуктов инструментальными средствами проектирования web-приложений // Известия Тульского государственного университета. Технические науки. 2024. № 3. С. 657-660. – DOI 10.24412/2071-6168-2024-3-657-658. EDN TPKDXC.
2. Базаржапова Т. Ж. Совершенствование информационной компетентности педагогов в условиях инфокоммуникационной среды : специальность 13.00.01 "Общая педагогика, история педагогики и образования" : диссертация на соискание ученой степени кандидата педагогических наук. Улан-Удэ, 2013. 182 с. – EDN SURXWJ.

DEVELOPMENT OF A NETWORKED EDUCATIONAL COMMUNITY WEBSITE

Bazarzhapova Tuya Zhamyanovna

Candidate of Pedagogical Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: btuyazh@yandex.ru

Ivanov Artem Sergeevich

Student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina Street 24a

Abstract. The article discusses the development of a network pedagogical community website for organizing a platform for interaction. Such a platform in the context of digital transformation of education will allow for continuous professional development of teachers and high-quality exchange of methodological experience.

Keywords: online pedagogical community, digital educational environment, professional development of teachers, web development, online platform, methodological interaction, and professional communication.

ВЫЯВЛЕНИЕ ПРЕДМЕТНЫХ ЗАТРУДНЕНИЙ И ФОРМИРОВАНИЕ АДРЕСНЫХ ПРОГРАММ ПОВЫШЕНИЯ КВАЛИФИКАЦИИ ПЕДАГОГИЧЕСКИХ РАБОТНИКОВ ПОСРЕДСТВОМ ДИАГНОСТИЧЕСКИХ ИНСТРУМЕНТОВ ЦИФРОВОГО ОБРАЗОВАТЕЛЬНОГО РЕСУРСА «ЯКЛАСС»

© Баранова Светлана Павловна

главный специалист по работе с регионами ООО «ЯКласс»

119002, г. Москва, а/я 72

Контактный телефон: +7 914 775-39-46

E-mail: s.baranova@vk.team

© Дамбуева Альбина Борисовна

кандидат физико-математических наук, доцент

Бурятский государственный университет

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: mail: abain76@list.ru

© Сункуева Ирина Николаевна

старший преподаватель,

Центр непрерывного повышения профессионального мастерства ГАУ ДПО РБ «БРИОП»

Россия, 670000, г. Улан-Удэ, ул. Советская, 30

E-mail: sunkuevair@rambler.ru

Аннотация. В статье представлены итоги диагностического тестирования учителей математики в Республике Бурятия, организованного с использованием ресурсов платформы «ЯКласс». Выявленные профессиональные дефициты легли в основу проектирования целевых моделей дополнительного образования педагогов. Платформа «ЯКласс» обеспечивает автоматизированную проверку и сбор статистики, что позволяет проводить точный анализ результатов выполнения заданий. В ходе анализа зафиксированы основные проблемные зоны: слабые навыки пространственного мышления и сложности в решении задач практической направленности. Для ГАУ ДПО РБ «БРИОП» разработаны практические ориентиры по обновлению региональных программ курсов, предполагающие усиление блоков по практико-ориентированным задачам, тренинги визуально-аналитических навыков и внедрение цифровых средств самооценки для педагогов. Полученные данные доказывают, что цифровая диагностика является действенным механизмом для выстраивания обоснованной системы методического сопровождения и совершенствования профессиональной компетентности учителей в вариативных образовательных условиях.

Ключевые слова: предметная диагностика, профессиональные дефициты учителей, математика, цифровая образовательная платформа, ЯКласс, повышение квалификации, анализ затруднений, педагогическая компетентность.

«ЯКласс» — российская аккредитованная IT-образовательная онлайн-платформа, входящая в группу компаний VK и включённая в федеральный перечень электронных образовательных ресурсов (приказ Минпросвещения РФ от 23.07.2025 № 551 [1]). Платформу используют более 50 000 образовательных организаций, 650 000 педагогов и 10 млн обучающихся [2].

На платформе содержатся методические материалы для педагогов, блоки теории и другие материалы согласно ФООП, цифровые задания и тесты с авто-

матической проверкой по школьной программе для обучающихся с 1 по 11 класс и студентов СПО, онлайн-тренажёры для подготовки к ГИА. Техническим преимуществом платформы являются более 2 трлн вариантов заданий и дополнительный инструмент для самообучения — «Шаги решения», которые позволяют выработать определённый алгоритм решения задач и сформировать у обучающихся понимание порядка выполнения логических действий [4].

В целях повышения качества общего образования, выявления предметных затруднений и формирования адресных программ повышения квалификации педагогических работников (учителей) в образовательных организациях Республики Бурятия была проведена диагностика предметных затруднений педагогических работников (далее — Диагностика), преподающих математику: 2025 г. — 119 чел., 2026 г. — 68 человек.

Диагностика была реализована на некоммерческой основе по заказу Министерства образования и науки Республики Бурятия и ГАУ ДПО РБ «БРИОП». Задания и тесты, включённые в Диагностику, созданы в соответствии с актуальными требованиями и форматами федерального тестирования. По итогам Диагностики региональные органы власти получили обобщённый аналитический отчёт, а педагогам предоставлялся индивидуальный отчёт с рекомендациями. Платформа «ЯКласс» также позволяет педагогам предварительно протестировать свои знания в специальном разделе «Подготовка к федеральному тестированию педагогов» (<https://hub.yaklass.ru/yaprofi>). Педагогам предоставляются результаты с комментариями о сильных и западающих сторонах, а также рекомендации для персонального развития [3].

В диагностическую работу по предмету «Математика» были включены задания, направленные на проверку ключевых предметных умений и профессиональных компетенций педагогов. Перечень проверяемых элементов содержания:

- Решение текстовых задач.
- Задачи на проценты.
- Задачи на движение.
- Задачи с экономическим содержанием.
- Работа с диаграммами.
- Вероятность события.
- Свойства ромба.
- Окружность. Углы, вписанные в окружность.
- Свойства трапеции.
- Объём цилиндра.
- Площадь боковой поверхности цилиндра.
- Пирамида. Сечение пирамиды.
- Вычисление по формулам.
- Геометрический смысл производной.
- Нахождение максимума логарифмической функции.
- Тригонометрические функции. Основные тригонометрические формулы.
- Решение тригонометрических уравнений с применением тригонометрических формул.

- Решение показательных уравнений.
- Решение рациональных неравенств. Метод интервалов.
- Решение системы неравенств с параметром.
- Свойства чисел.

Перечислим задания с минимальным (менее 70%) процентом выполнения. На изображениях приведены скриншоты заданий на платформе «ЯКласс».

Задание 11. Упражнение типа ЕГЭ, проверяющее навыки выполнения тождественных преобразований и нахождения неизвестной переменной по формуле. Необходимые умения: умение осуществлять практические расчёты по формулам — 64% в 2025 г., 66% в 2026 г. (рис. 1).

Два батискафа движутся навстречу друг другу в жидкости по кратчайшему расстоянию. Первый батискаф издаёт звуковой сигнал, частота которого равна $f_0 = 135$ Гц. Второй батискаф улавливает и регистрирует эти сигналы, причём частота улавливаемого сигнала больше испускаемой первым батискафом частоты и вычисляется по формуле $f = f_0 \frac{c + u}{c - v}$ (Гц). Скорость второго батискафа равна $u = 10$ м/с, скорость первого равна $v = 15$ м/с (скорости даны относительно жидкости), c — скорость распространения сигнала в жидкости (в м/с). При какой максимальной скорости c распространения сигнала в жидкости частота сигнала во втором батискафе будет не менее 140 Гц? Ответ дай в метрах в секунду.

Ответ:

Рисунок 1. Задание 11

Задание 15. Тригонометрическое уравнение, сводимое к квадратному, — 39% в 2025 г. и 32% в 2026 г. (рис. 2).

Задание 16. Построение сечения пирамиды и доказательство перпендикулярности прямых — 17% в 2025 г. и 13% в 2026 г. (рис. 3).

Задание 17. Дробно-рациональное неравенство. Метод интервалов — 57% в 2025 г. и 60% в 2026 г. (рис. 4).

Задание 18. Окружность, вписанная в четырёхугольник, — 55% в 2025 г. и 56% в 2026 г. (рис. 5).

Задание 19. Выплаты кредита равными платежами — 32% в 2025 г. и 40% в 2026 г. (рис. 6).

Задание 20. Исследование системы показательных неравенств — 30% в 2025 г. и 43% в 2026 г. (рис. 7).

Задание 21. Оценка среднего арифметического чисел — 65% в 2025 г. и 81% в 2026 г. (рис. 8).

а) Реши уравнение $\sin^2 \frac{x}{2} - \cos^2 \frac{x}{2} = \cos(-7\pi + 2x)$.

☐ $\pm \frac{2}{3}\pi + 2\pi k, k \in \mathbb{Z}$
☐ $2\pi k, k \in \mathbb{Z}$
☐ $\pi k, k \in \mathbb{Z}$
☐ $\pm \frac{\pi}{4} + 2\pi k, k \in \mathbb{Z}$
☐ $-\frac{\pi}{3} + 2\pi k, k \in \mathbb{Z}$
☐ $\pm \frac{\pi}{6} + 2\pi k, k \in \mathbb{Z}$
☐ $\frac{\pi}{3} + 2\pi k, k \in \mathbb{Z}$
☐ $\pi + \pi k, k \in \mathbb{Z}$

б) Выбери корни, принадлежащие отрезку $\left[-\frac{11}{2}\pi; -4\pi\right]$:

☐ -4π
☐ $-\frac{21\pi}{4}$
☐ -5π
☐ $-\frac{14\pi}{3}$
☐ $-\frac{16\pi}{3}$
☐ $-\frac{31\pi}{6}$
☐ $-\frac{25\pi}{6}$
☐ $-\frac{17\pi}{4}$

Рисунок 2. Задание 15

В пирамиде $SABC$ $AB = BC = AC = 56$, $AS = BS = CS = 48$. $M \in SC$, $K \in AB$, $SM : MC = AK : KB = 17 : 10$. Плоскость α содержит прямую MK и параллельна прямой SA . Докажи, что сечение пирамиды $SABC$ плоскостью α — прямоугольник.

Решение
Некоторые утверждения и этапы доказательства (сделай рисунок в тетради, сохранив обозначения точек).

Варианты ответов:

$KT \parallel AS$	$BS \perp AC$	проводим TM	$PM = KT$	проводим KP	проводим KM
$AS \perp BC$	$TM \perp SC$	$PM \parallel BS$	$PM \perp AC$	$PM \parallel KT$	$PM \parallel AC$
$PM \parallel AS$	проводим KS	$AS \perp SC$	$KM \parallel AC$		

Строим :

.

Так как и , то $KTMP$ — параллелограмм.

$TM \perp MP$, так как $PM \parallel AS$ и .

Рисунок 3. Задание 16

Реши неравенство:

$$\frac{87x + 348}{x^2 + 5x + 4} + 2x + 31 \leq \frac{10}{x + 1}.$$

Выбери правильный ответ:

- ☐ $(-\infty; -12] \cup [-4, 5; -1)$
- ☐ $(-\infty; -12] \cup [-4, 5; -4) \cup (-4; -1).$
- ☐ $(-\infty; -12) \cup (-4; -1)$
- ☐ $(-\infty; -12] \cup [-4, 5; -1) \cup (-1; +\infty)$

Рисунок 4. Задание 17

Окружность с центром O_1 касается оснований BC и AD и боковой стороны AB трапеции $ABCD$. Окружность с центром O_2 касается сторон BC , CD и AD . Известно, что $AB = 11$, $BC = 13$, $CD = 17$, $AD = 34$.

Докажи, что прямая O_1O_2 параллельна основаниям трапеции $ABCD$.

Найди O_1O_2 .

Ответ: .

Рисунок 5. Задание 18

31 декабря 2011 года Василий взял в банке 4804800 рублей в кредит под 20 % годовых. Схема выплаты кредита следующая: 31 декабря каждого следующего года банк начисляет проценты на оставшуюся сумму долга, затем Василий переводит в банк платёж, составляющий целое число рублей. Весь долг Василий выплатил тремя равными платежами. На сколько рублей меньше он бы отдал банку, если бы смог выплатить долг двумя равными платежами?

Ответ: .

Рисунок 6. Задание 19

Найди все значения параметра a , при каждом из которых множеством решений системы неравенств

$$\begin{cases} 4^{x+2a} \leq 16^{x+a-11} \\ 4^{x-3a-3} \geq 16^{x+a-3} \end{cases}$$

является отрезок числовой прямой, длина которого равна 3.

Ответ: .

Рисунок 7. Задание 20

На доске написаны числа $a_1, a_2, \dots, a_n$, каждое из которых не меньше 50 и не больше 90. Каждое из чисел a_i уменьшили на $r_i\%$ так, что либо $r_i = 9$, либо число a_i уменьшилось на 9. (То есть некоторые числа уменьшились на 9, остальные уменьшились на 9%).

Может ли среднее арифметическое чисел r_i быть равным 20?

Ответ: .

Рисунок 8. Задание 21

Проведённый анализ результатов диагностической работы, выполненной учителями на платформе «ЯКласс», позволил выявить не только текущий уровень предметной компетентности педагогов, но и устойчивые профессиональные дефициты, требующие системной коррекции. Динамика результатов за 2025–2026 гг. демонстрирует неоднозначную картину: наряду с позитивными сдвигами по отдельным позициям наблюдается регресс в выполнении заданий, традиционно считающихся «сложными» в структуре ЕГЭ профильного уровня. Сравнительный анализ показал разнонаправленные тенденции.

Положительная динамика зафиксирована в заданиях, связанных с прикладными вычислениями и стандартными алгоритмами: задание 19 (увеличение на 8%), задание 20 (увеличение на 13%) и задание 21 (увеличение на 16%). Это свидетельствует о том, что при целенаправленной методической подготовке учителя успешно компенсируют пробелы в «технических» темах.

Стагнация наблюдается в заданиях, требующих развитого пространственного мышления и глубокого понимания свойств геометрических объектов. Наибольшее снижение зафиксировано в задании 16 (с 17 до 13%), а также задании 15 (с 39 до 32%), что указывает на укоренённость методических ошибок в преподавании данных разделов.

Полученные в ходе тестирования данные указывают на приоритетные зоны для методической работы с педагогами. В их числе — переход от формального воспроизведения знаний к осмысленному применению, освоение комплексных межпредметных заданий, развитие как предметных, так и методических навыков, а также усиление прикладной составляющей уроков математики.

Площадка «ЯКласс» зарекомендовала себя как надёжный инструмент для охватной проверки компетенций и выстраивания персональных маршрутов самосовершенствования учителя. Анализ диагностических работ, который проводится на протяжении трех лет на платформе «ЯКласс» дает возможность выявить наилучшие и наихудшие результаты по образовательным организациям муниципалитетов, что дает нашим службам более адресно оказывать методическую, консультативную помощь.

Литература

1. Министерство Просвещения Российской Федерации: официальный сайт. URL: edu.gov.ru (дата обращения: 16.06.2026).
2. ЯКласс : официальный сайт. URL: www.yaklass.ru (дата обращения: 16.06.2026).
3. Подготовка к федеральному тестированию педагогов: официальный сайт. URL: hub.yaklass.ru/yaprofi (дата обращения: 16.06.2026).

4. Разумова О. В., Садыкова Е. Р., Кукушкина А. В. Цифровой образовательный ресурс «ЯКЛАСС» как средство развития информационной культуры учащихся // Информация и образование: границы коммуникаций INFO. 2020. № 12. С. 63–65.

IDENTIFICATION OF SUBJECT-RELATED DIFFICULTIES AND FORMATION OF TARGETED PROFESSIONAL DEVELOPMENT PROGRAMS FOR TEACHING STAFF THROUGH DIAGNOSTIC TOOLS OF THE DIGITAL EDUCATIONAL RESOURCE "YAKLASS"

Svetlana Pavlovna Baranova

Chief Specialist for Working with Regions of LLC "YAKLASS"

119002, Moscow, P.O. Box 72

Contact phone: +7 914 775-39-46

E-mail: s.baranova@vk.team

Dambueva Albina Borisovna

Candidate of Physical and Mathematical Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina Street, 24a

E-mail: mail: abain76@list.ru

Sunkueva Irina Nikolaevna

Senior Lecturer of the Center for Continuous Professional Development of the State Autonomous Institution of Further Professional Education of the Republic of Buryatia

«Buryat Republican Institute of Educational Policy»

Russia, 670000, Ulan-Ude, Sovetskaya Street, 30

E-mail: sunkuevaira@rambler.ru

Abstract. The article presents the results of diagnostic testing of mathematics teachers in the Republic of Buryatia, organized using the resources of the "YClass" platform. The identified professional shortcomings formed the basis for designing target models of additional education for teachers. The YaClass platform provides automated verification and statistics collection, which allows for accurate analysis of the results of task completion. During the analysis, the main problem areas were identified: weak spatial thinking skills and difficulties in solving practical problems. For the State Autonomous Institution for Additional Professional Education of the Republic of Bashkortostan "BRIOPS", practical guidelines have been developed for updating regional course programs, which include strengthening blocks on practical-oriented tasks, visual and analytical skills training, and the introduction of digital self-assessment tools for teachers. The data obtained proves that digital diagnostics is an effective mechanism for building a well-founded system of methodological support and improving the professional competence of teachers in diverse educational environments.

Keywords: subject diagnostics, professional shortcomings of teachers, mathematics, digital educational platform, YaKlass, advanced training, analysis of difficulties, and pedagogical competence.

ВЛИЯНИЕ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА И НЕЙРОСЕТЕЙ НА СОВРЕМЕННОЕ ОБРАЗОВАНИЕ

© **Алехина Валерия Александровна**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: lera.zerkalo.95@mail.ru

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье рассматривается влияние технологий искусственного интеллекта и нейросетей на современное образование. Проводится анализ дидактического потенциала ИИ, включая персонализацию обучения, автоматизацию рутинных задач и развитие цифровой грамотности; выделяются ключевые риски – когнитивная предвзятость к технологиям нейросетей, цифровой разрыв, вопросы конфиденциальности и этики. Особое внимание уделяется границам применимости искусственного интеллекта и необходимости формирования культуры ответственного использования нейросетей в учебном процессе. Приводятся примеры образовательных платформ, использующих ИИ. Делается вывод о необходимости комплексного подхода, сочетающего технологические инновации с педагогической экспертизой.

Ключевые слова: искусственный интеллект, нейросети, образование, персонализация обучения, цифровая грамотность, риски, этика.

Современный этап развития общества характеризуется стремительным проникновением технологий искусственного интеллекта (ИИ) во все сферы человеческой деятельности, и образование не является исключением. Цифровизация образовательного процесса, переход к гибридным моделям обучения и необходимость обработки растущих объемов информации делают использование технологий нейросетей ключевым фактором повышения эффективности как преподавателей, так и обучающихся. На сегодняшний день степень интеграции искусственного интеллекта в сферу образования, а в частности в планирование и проведение занятий остается острым вопросом для образовательных учреждений – от школ до университетов. В связи с этим более значимыми становятся вопросы о пределах применимости нейросетей в образовательном процессе и возможных негативных последствиях таких решений [1, с.33].

Цель и задачи. Цель исследования – комплексный анализ воздействия ИИ и нейросетей на образование, а также выявление их дидактических преимуществ и потенциальных рисков. Задачами работы являются определение понятия искусственного интеллекта, оценка дидактического потенциала нейросетей, а также определение рисков, связанных с внедрением ИИ в образовательный процесс.

Методология и методы исследования. В работе использованы методы теоретического анализа научной литературы, классификации подходов к применению ИИ в образовании, а также синтеза и обобщения данных о дидактическом потенциале и рисках нейросетей.

Разработанность темы. Вопросы применения искусственного интеллекта в образовании рассматриваются в работах Бармашовой Т.И., Давыдова С.Г., Шобонова Н.А. и др. Однако комплексный анализ как дидактического потенциала, так и педагогических рисков нейросетей в российском образовательном контексте представлен недостаточно.

Объект и предмет исследования. Объектом исследования является образовательный процесс с использованием технологий искусственного интеллекта. Предметом исследования — дидактический потенциал и педагогические риски применения ИИ и нейросетей в образовании.

Понятие искусственного интеллекта и нейросетей в образовательном контексте. Прежде чем анализировать влияние ИИ на образование, важно уточнить, что из себя представляет искусственный интеллект. В академической среде «искусственный интеллект» относится не только к «умным» машинам, но и к научной дисциплине, изучающей их поведение. Как отмечает Ю.В. Сибирякова, ИИ – это одновременно и наука, и практический результат: способность машин решать задачи, ранее считавшиеся исключительно человеческими [5, с. 211].

В образовательной практике, однако, часто говорят не об ИИ как об абстрактном понятии, а о нейросетях – специфических математических моделях, которые учатся на данных и служат «рабочим инструментом» современного искусственного интеллекта. Нейросети могут делать многое: генерировать тексты и изображения, переводить с одного языка на другой, распознавать речь и почерк, создавать чат-боты, редактировать фотографии и сжимать информацию. В образовании это открывает двери для новых форматов обучения, делает содержание уроков более разнообразным и насыщенным, а также упрощает рутинную работу учителей [2, с. 687].

Дидактический потенциал искусственного интеллекта в образовании. Внедрение технологий искусственного интеллекта в школьную и университетскую практику открывает огромные возможности для улучшения образовательного процесса. Дидактический потенциал ИИ проявляется в нескольких ключевых областях.

Во-первых, нейросети позволяют автоматизировать рутинные задачи учителя, такие как проверка стандартных заданий, составление отчетов и подготовка шаблонов документов. Это высвобождает время для творческой и аналитиче-

ской работы, живого общения со студентами и более глубокой методической разработки уроков.

Во-вторых, ИИ позволяет подобрать индивидуальный подход к обучению каждого ученика в связи с его особенностями, стилем и темпом обучения. Использование генеративных нейросетей позволяет создавать интерактивные материалы, повышая интерес учащихся к процессу обучения. Преподаватели получают гибкий инструмент для разработки разноуровневых заданий и построения гибкой образовательной среды.

В-третьих, использование искусственного интеллекта способствует формированию и дальнейшему развитию цифровой грамотности как у обучающихся, так и у преподавателей. Освоение технологий создания визуального и текстового контента, умение корректно формулировать запросы, а также понимание базовых принципов работы алгоритмов становятся необходимыми элементами современного образования.

Риски внедрения ИИ в образовательный процесс. Несмотря на значительный дидактический потенциал, интеграция ИИ в образовательный процесс сопряжена с рядом серьезных рисков, требующих тщательного анализа и разработки мер по их минимизации.

Одной из главных проблем является когнитивная предвзятость – систематические ошибки обработки информации, влияющие на объективность принимаемых решений. В случае искусственного интеллекта это усугубляется тем, что алгоритмы учатся на данных, которые уже могут содержать искажения и предвзятость, а это может значительно навредить учебному процессу, усиливая стереотипы и закрепляя ошибки. Для снижения риска необходимы не только коррекция алгоритмов, но и критический анализ выдаваемых результатов со стороны педагогов.

Другим существенным риском является цифровой разрыв, проявляющийся не только в разном доступе к технологиям, но в различиях уровня цифровой грамотности. Внедрение ИИ в образовательные учреждения может углубить образовательное неравенство, если учащиеся и преподаватели не владеют достаточными навыками использования таких инструментов.

Отдельную тревогу вызывает вопрос авторства и «фейковой» информации. Контент, сгенерированный нейросетью, порождает неопределенность: кто является автором – нейросеть, ее разработчик или пользователь, написавший запрос – промт. Кроме того, существующие произведения. Защищенные авторским правом, используются для обучения ИИ, что может привести к непреднамеренному или преднамеренному заимствованию. Еще один серьезный риск – генерация ложной, «фейковой» информации, когда система «додумывает» ответ из-за недостатка или недостоверности данных [6].

Значительный круг вопросов связан с этикой, конфиденциальностью данных. В работе [4] анализируются ключевые этические аспекты применения ИИ в образовании и меры по защите приватности данных пользователей. Среди более важных проблем выделяются:

1. Прозрачность алгоритмов – пользователи должны понимать, на каких данные основаны ответы ИИ.

2. Объективность и отсутствие предубеждений – нейросети могут наследовать и усиливать социальные стереотипы.
3. Ответственность за решения – необходимо определить, кто отвечает за ошибки алгоритмов, влияющие на успеваемость учащегося или карьеру преподавателя.
4. Недопустимость манипулятивных технологий – например, распознавание эмоций может создавать стрессовую обстановку и нарушать личные границы.

Также следует отметить риск чрезмерной технологической зависимости. Использование ИИ в самых простых задачах способно ухудшать аналитические способности, ограничивать глубину самостоятельных исследований и ослаблять рефлексивное мышление [5].

Методические аспекты использования ИИ в учебном процессе. Ключевым элементом работы с нейросетями является формулирование запросов – промптов. Качество и структура запроса определяют как точность, так и педагогическую ценность получаемого результата. Эффективный запрос должен содержать четкое описание задачи, указание желаемого формата, объема, тона и стиля ответа, а также дополнительные параметры.

Выводы. Внедрение технологий искусственного интеллекта и нейросетей в современное образование представляет собой сложный и многогранный процесс. С одной стороны, ИИ открывает возможности для адаптации обучения под конкретного учащегося, автоматизации рутинных задач, создания интерактивного контента для вовлечения учащихся в образовательный процесс и повышения цифровой грамотности. С другой стороны, возникают серьезные вопросы: когнитивные искажения, цифровое неравенство, проблемы конфиденциальности данных, авторства и этики.

Успешная интеграция нейросетей требует не только технологических решений, но и формирования новой этической, правовой и педагогической культуры, обеспечивающей безопасное и обоснованное использование ИИ в образовании.

Литература

1. Бармашова Т. И. Цифровые технологии и искусственный интеллект в образовательной сфере: плюсы и минусы // Профессиональное образование в современном мире. 2025. № 15(1). С. 39–46.
2. Давыдов С. Г., Матвеева Н. Н., Адемукова Н. В., Вичканова А. А. Искусственный интеллект в российском высшем образовании: текущее состояние и перспективы развития // Университетское управление: практика и анализ. 2024. № 28(3). С. 32–44.
3. AI в обучении: на что способны технологии уже сейчас? // EduTech. 2022. № 4(49). 60 с. URL: https://lib.tsu.ru/sp/assets/users/_smirnov/EduTech_49_web.pdf (дата обращения: 10.05.2026).
4. Шобонов Н. А. Искусственный интеллект в образовании // Проблемы современного педагогического образования. 2023. № 79-4. С. 288–290.
5. Искусственный интеллект и высшее образование: возможности, практики и будущее / К. А. Баранников, М. С. Добрякова, Е. Г. Новикова [и др.] // Яндекс Образование. URL: <https://education.yandex.ru/aihighreport> (дата обращения: 10.05.2026).

6. Миндигулова А. А. Возможности и ограничения инструментов искусственного интеллекта в образовании // Современное педагогическое образование. 2022. № 3.

THE IMPACT OF ARTIFICIAL INTELLIGENCE AND NEURAL NETWORKS ON MODERN EDUCATION

Alekhina Valeria Aleksandrovna

student,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: lera.zerkalo.95@mail.ru

Matytsin Artem Vladimirovich

student,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article discusses the impact of artificial intelligence technologies and neural networks on modern education. The didactic potential of AI is analyzed, including the personalization of learning, automation of routine tasks and the development of digital literacy; key risks are identified – cognitive bias towards neural network technologies, the digital divide, privacy and ethics issues. Special attention is paid to the limits of the applicability of artificial intelligence and the need to form a culture of responsible use of neural networks in the educational process. Examples of educational platforms using AI are given. The conclusion is made about the need for an integrated approach combining technological innovations with pedagogical expertise.

Keywords: artificial intelligence, neural networks, education, personalized learning, digital literacy, risks, ethics.

ГИБРИДНАЯ ОБРАЗОВАТЕЛЬНАЯ СРЕДА КАК ДРАЙВЕР РАЗВИТИЯ ЦИФРОВОЙ КУЛЬТУРЫ: ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ, ИНФОКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ И УПРАВЛЕНЧЕСКИЕ РЕШЕНИЯ

© Пантелеева Елена Владимировна

преподаватель,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: elenapanteleyeva99715@yandex.ru

Аннотация. В статье рассматривается гибридная образовательная среда как ключевой фактор формирования цифровой культуры участников образовательного процесса. Анализируется роль интеллектуальных систем поддержки принятия решений, инфокоммуникационных технологий и организационно-управленческих моделей в обеспечении непрерывности и персонализации обучения. Показано, что интеграция очных и дистанционных форматов создает условия для развития критического мышления, информационной безопасности и адаптивности к технологическим изменениям. Предлагаются конкретные управленческие решения, направленные на снижение цифрового разрыва и повышение эффективности взаимодействия в гибридной среде. Результаты исследования могут быть использованы при модернизации образовательных программ и разработке локальных нормативных актов вузов.

Ключевые слова: гибридная образовательная среда, цифровая культура, интеллектуальные системы, инфокоммуникационные технологии, управленческие решения, персонализация обучения, информационная безопасность.

Введение. Трансформация системы высшего образования под влиянием цифровых технологий привела к возникновению гибридных образовательных сред, объединяющих традиционное аудиторное взаимодействие и онлайн-коммуникации. Однако до настоящего времени недостаточно изучен вопрос о том, как именно гибридная среда становится драйвером развития цифровой культуры преподавателей, студентов и управленческого персонала. Данное исследование восполняет этот пробел, акцентируя роль интеллектуальных систем и инфокоммуникационных решений.

Актуальность. Потребность в целенаправленном формировании цифровой культуры обусловлена ускоренным внедрением информационных систем во все сферы деятельности. Без развития соответствующей культуры эффективность цифровой трансформации образования остается низкой. Современные вызовы требуют не просто технической грамотности, но и способности критически оценивать информацию, соблюдать этические нормы в цифровой среде и принимать обоснованные решения с использованием интеллектуальных систем.

Объект исследования – образовательный процесс в высшем учебном заведении. **Предмет** – совокупность интеллектуальных систем, инфокоммуникационных технологий и управленческих решений, обеспечивающих функционирование гибридной образовательной среды и способствующих развитию цифровой культуры.

Цель работы – выявить и обосновать механизмы влияния гибридной образовательной среды на развитие цифровой культуры участников образовательных отношений. Задачи исследования включают анализ существующих интеллектуальных систем в образовании, оценку инфокоммуникационной инфраструктуры, разработку рекомендаций по управленческим решениям.

Методология и методы исследования. Теоретическую основу составили работы отечественных и зарубежных авторов в области цифровой педагогики, теории гибридного обучения и информационной безопасности. Использованы методы системного анализа, моделирования, а также анализ документов и педагогической практики. Эмпирическая часть базируется на наблюдении за реализацией гибридных форматов в Бурятском государственном университете в 2024-2025 учебном году.

Разработанность темы. В научной литературе широко представлены исследования по дистанционному обучению (А.А. Андреев, Е.С. Полат) и отдельным аспектам цифровой культуры (Г.У. Солдатова, Т.Н. Носкова). Однако комплексных работ, связывающих архитектуру гибридной среды, интеллектуальные системы поддержки решений и целенаправленное развитие цифровой культуры именно через управленческие механизмы, недостаточно. Данное исследование предлагает такой интегративный подход.

Результаты исследования. Анализ показал, что ключевыми компонентами гибридной среды, влияющими на цифровую культуру, выступают три группы факторов. Первая группа — интеллектуальные системы (адаптивные рекомендательные сервисы, интеллектуальные помощники для планирования учебной нагрузки, системы анализа академической активности). Эти системы обучают участников работе с данными, принятию решений в условиях неопределенности и самоорганизации.

Вторая группа — инфокоммуникационные технологии (платформы видеоконференцсвязи с элементами дополненной реальности, облачные хранилища с совместным доступом, чат-боты для оперативной поддержки). Их использование формирует навыки безопасной коммуникации, управления цифровым следом и соблюдения информационной гигиены.

Третья группа — управленческие решения, включающие создание локальных нормативных актов о признании онлайн-активности, введение рейтингов цифровой культуры для преподавателей, организацию постоянно действующих семинаров по цифровой этике. Установлено, что без целенаправленной управленческой поддержки технологические инновации не приводят к устойчивому росту цифровой культуры.

Выводы. Гибридная образовательная среда становится драйвером развития цифровой культуры только при условии синхронного применения интеллектуальных систем, современных инфокоммуникационных технологий и продуманных управленческих решений. Управленческие решения должны обеспечивать регулярную рефлексию цифрового опыта, мотивацию сотрудников и студентов к безопасному и продуктивному поведению в гибридной среде. Полученные результаты могут быть использованы при проектировании образовательных программ и стратегий цифровой трансформации вузов. Перспективным направле-

нием дальнейших исследований является разработка методики оценки уровня цифровой культуры в гибридной среде с применением интеллектуальных систем анализа больших данных.

Литература

1. Андреев А. А., Солдаткин В. И. Дистанционное обучение: сущность, технологии, организация. Москва: Изд-во МЭСИ, 1999. 196 с. ISBN 5-7764-0164-X.
2. Носкова Т. Н., Павлова Т. Б., Яковлева О. В. Инструменты педагогической деятельности в электронной среде // Высшее образование в России. 2017. № 8–9. С. 121–130. DOI: 10.24075/MTCPE.2021.021.
3. Шобонов Н. А. Искусственный интеллект в образовании // Проблемы современного педагогического образования. 2023. № 79-4. С. 288–290.
4. Dziuban C., Graham C. R., Moskal P. D. Blended learning: Research perspectives. New York: Routledge, 2018. 342 p.
5. Garrison D. R., Vaughan N. D. Blended learning in higher education: Framework, principles, and guidelines. San Francisco: Jossey-Bass, 2020. 245 p.
6. Ващенко В. А. Интеллектуальные системы поддержки принятия решений в организациях высшего образования: возможности и ограничения // Бизнес. Образование. Право. 2020. № 4(53). С. 224–229. DOI: 10.25683/VOLBI.2020.53.477.
7. Информационные технологии в образовании: учебник для вузов / Е. В. Баранова, М. И. Бочаров, С. С. Куликова и др. 3-е изд., стер. Санкт-Петербург: Лань, 2025. 296 с. ISBN 978-5-507-53194-3.

HYBRID EDUCATIONAL ENVIRONMENT AS A DRIVER OF DIGITAL CULTURE DEVELOPMENT: INTELLIGENT SYSTEMS, INFORMATION AND COMMUNICATION TECHNOLOGIES, AND MANAGEMENT SOLUTIONS

Panteleeva Elena Vladimirovna

Lecturer

Dorzhi Banzarov Buryat State University

24a Smolina Street, Ulan-Ude, 670000 Russia

E-mail: elenapanteleyeva99715@yandex.ru

Abstract. The article examines the hybrid educational environment as a key factor in the formation of the digital culture of educational process participants. The role of intelligent decision support systems, information and communication technologies, and organizational and managerial models in ensuring the continuity and personalization of learning is analyzed. It is shown that the integration of face-to-face and remote formats creates conditions for the development of critical thinking, information security, and adaptability to technological changes. Specific managerial solutions are proposed to reduce the digital divide and improve the efficiency of interaction in a hybrid environment. The research results can be used in the modernization of educational programs and the development of local regulations for universities.

Keywords: hybrid educational environment, digital culture, intelligent systems, information and communication technologies, management solutions, personalization of education, and information security.

ПРИМЕНЕНИЕ EXCEL В ГЕОМЕТРИИ: АВТОМАТИЗАЦИЯ РАСЧЁТА НЕИЗВЕСТНЫХ ЭЛЕМЕНТОВ ТРЕУГОЛЬНИКА

© Киреев Артем Александрович

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: mig31lipetsk@yandex.ru

© Матыцин Артем Владимирович

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© Гаев Леонид Витальевич

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. Треугольник — одна из важнейших геометрических фигур, широко используемых в науке и технике. Решение его элементов применяется на практике, например, для определения расстояния до недоступной точки. В статье описывается создание программы в Microsoft Excel, в которой по заданным значениям элементов треугольника автоматически вычисляются все остальные параметры.

Ключевые слова: элементы треугольника, формулы в Microsoft Excel, логические условия и функции Excel, условное форматирование, визуализация решения.

Введение. Были изучены 6 видов треугольников [1]:

- 3 по сторонам (разносторонний, равнобедренный, равносторонний);
- 3 по углам (прямоугольный, остроугольный, тупоугольный).

Также были изучены элементы треугольника. Выполнен чертёж треугольника ABC с обозначениями его элементов для их визуализации. Определены ячейки в Excel для ввода данных и последующему обращению к ним.

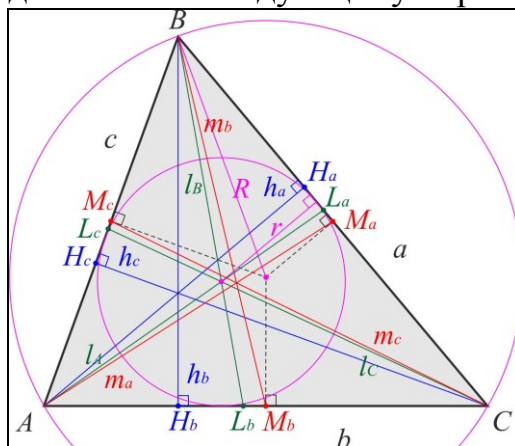


Рисунок 1. чертёж треугольника с обозначениями элементов

Составление формул и разбор ошибок. Как видно по чертежу, треугольник ABC состоит из других треугольников, образованных делением исходного треугольника различными элементами – биссектрисами, медианами, высотами.

Было найдено 75 формул [1, 2, 5] для расчёта элементов треугольника, описал формулы в Microsoft Excel и вставил в ячейки для каждого элемента. Получилась исходная таблица для решения элементов треугольника.

Написал логические условия «ЕСЛИ», при выполнении которых формулы, вставленные в исходную таблицу, начали бы считать значение элемента. У функции ЕСЛИ возможны два результата. Первый результат возвращается в случае, если сравнение ИСТИНА, второй – если сравнение ЛОЖЬ. Для сокращения этой функции, когда много условий, можно использовать функцию «И», с помощью которой можно определить, все ли проверяемые условия принимают значение ИСТИНА, при этом выполнение всех этих условий обязательно, иначе «0» – ЛОЖЬ [3, 4]. При невыполнении логического условия программа выдаёт информацию «Нет решения» или «Ошибка ввода» на главной странице ввода исходных данных.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH
1																																		
2	D2	сторона [a]	a	3,0																														
3	D3	сторона [b]	b	5,0																														
4	D4	сторона [c]	c	4,0																														
5	D5	угол A, в градусах	A	36,9																														
6	D6	угол B, в градусах	B		90,0																													
7	D7	угол C, в градусах	C			53,1																												
8	D8	высота к стороне [a]	h _a																															
9	D9	высота к стороне [b]	h _b																															
10	D10	высота к стороне [c]	h _c																															
11	D11	медиана к стороне [a]	m _a																															
12	D12	медиана к стороне [b]	m _b																															
13	D13	медиана к стороне [c]	m _c																															
14	D14	биссектриса угла A	l _A					4,2																										
15	D15	биссектриса угла B	l _B						2,4																									
16	D16	биссектриса угла C	l _C							3,4																								
17	D17	периметр треугольника	p																															
18	D18	площадь треугольника	S																															
19	D19	радиус описанной окружности	R																															
20	D20	радиус вписанной окружности	r																															
21	D21	сторона [a]	a	3,0																														
22	D22	сторона [b]	b	5,0																														
23	D23	сторона [c]	c	4,0																														
24	D24	угол A, в градусах	A	36,9																														
25	D25	угол B, в градусах	B	90,0																														
26	D26	угол C, в градусах	C	53,1																														
27	D27	высота к стороне [a]	h _a																															
28	D28	высота к стороне [b]	h _b																															
29	D29	высота к стороне [c]	h _c																															
30	D30	медиана к стороне [a]	m _a	4,3																														
31	D31	медиана к стороне [b]	m _b	2,5																														
32	D32	медиана к стороне [c]	m _c	3,8																														
33	D33	биссектриса угла A	l _A	4,2																														
34	D34	биссектриса угла B	l _B	2,4																														
35	D35	биссектриса угла C	l _C	3,4																														
36	D36	периметр треугольника	p	12,0																														
37	D37	площадь треугольника	S																															
38	D38	радиус описанной окружности	R																															
39	D39	радиус вписанной окружности	r																															

Рисунок 2. таблица, в которую введены формулы расчёта элементов треугольника (с найденными значениями по условиям)

Получение результата вычислений. После того, как я забил все 75 формул в таблицу (E2:CA20), необходимо получить результат вычислений в ячейках. Для этого я использовал функцию МАКС, которая возвращает наибольшее значение из всех результатов вычислений в строках, т.е. выбирает одно найденное значение для каждого элемента треугольника [3, 4]. В каждой ячейке столбца (CB2:CB20) записал условия МАКС(E2:CA2), МАКС(E20:CA20). Полученные значения в этом столбце стали первым решением элементов треугольника. После этого я скопировал таблицу (A2:CA20) со столбцом первого решения и вставил ниже. В новой таблице заменил значение «ДАНО» (D21:D39) на полученные решения D21=CB2, D22=CB3, D39=CB20 таблицы, стоящей выше. В столбце (CB21:CB39) получилось второе решение. Полученную вторую таблицу скопировал и вставил ниже ещё три раза. В столбце (CB78:CB96) получил решение, которое перенёс в начало таблицы в столбец (CC2:CC20). Это конеч-

ное решение элементов треугольника, которое переносится на главную страницу.

Форматирование полученной таблицы. В таблице, кроме рассчитанных значений, оказалось много значений равных нулю. Они мешали визуальной оценке информации. Для того, чтобы их убрать и выделить нужную информацию, я использовал условное форматирование [3, 4], выделив все таблицы расчетов (E2:CA96). Теперь, при решении задачи, в таблице будут отображаться только решения по данным условиям, выделенные цветом. Чтобы увидеть, какая формула описана в ячейке таблицы расчёта элементов треугольника (решение), в строку (E1:CA1), где указаны номера формул, я вставил гиперссылки на файлы JPG, в которых написаны эти формулы.

Проблемы решения. В отдельных случаях программа может выдать решение, нарушающее аксиомы треугольника (например, угол 180° или нарушение неравенства треугольника). Для этого в таблицу добавлены проверочные условия. Если введённые углы не совпадают с рассчитанными — выводится сообщение «Нет решения!».

Визуализация решений. Увидеть результаты решений элементов треугольника лучше на рисунке, который покажет элементы треугольника наглядно. Для этого используются выпадающие списки с показом изображений. Всего получилось 28 возможных вариантов треугольников:

- Равносторонний треугольник (1);
- Равнобедренные прямоугольные (3), остроугольные (3), тупоугольные (3) треугольники с разным расположением углов;
- Прямоугольный треугольник с разным расположением углов (6);
- Остроугольный треугольник с разным расположением углов (6);
- Тупоугольный треугольник с разным расположением углов (6).

Каждому рисунку полученных треугольников присвоены названия, вставил эти названия в ячейки столбца (CF2:CF29) и описал эти треугольники условиями.

Далее, используя функции ПОИСКПОЗ и СМЕЩ, составлен именованный динамический диапазон, чтобы программа могла найти рисунок треугольника из выпадающего списка. Функция ПОИСКПОЗ выполняет поиск указанного элемента в диапазоне ячеек и возвращает относительную позицию этого элемента в диапазоне, а функция СМЕЩ возвращает ссылку на диапазон, отстоящий от ячейки или диапазона ячеек на заданное число строк и столбцов. Возвращаемая ссылка может быть отдельной ячейкой или диапазоном ячеек. Можно задавать количество возвращаемых строк и столбцов. [3, 4]. Рисунки были расположены на третьем листе. Теперь, при решении элементов треугольника (его углов), изображается рисунок найденного треугольника. Причём, равносторонний и равнобедренный прямоугольный треугольники изображаются точно, а все остальные — приближённо к найденным значениям углов. Без ввода условий в «ДАНО», отображается общий вид треугольника с обозначением всех элементов.

Главный лист программы. Результаты вычислений выведены на главный лист. Программа Excel сама выдаёт ошибки в результатах. Введена функция ЕСЛИОШИБКА, которая возвращает значение если выражение ошибочно, в противном случае, возвращает само выражение [3, 4] для правильного написания значений в ДАНО.

Элементы треугольника:	Дано:	Решение:
сторона [a]	5,0	Нет решения!
сторона [b]	10,0	Нет решения!
сторона [c]	1	Нет решения!
угол A, в градусах	A	Нет решения!
угол B, в градусах	B	Нет решения!
угол C, в градусах	C	Нет решения!
высота к стороне [a]	h_a	Нет решения!
высота к стороне [b]	h_b	Нет решения!
высота к стороне [c]	h_c	Нет решения!
медиана к стороне [a]	m_a	Нет решения!
медиана к стороне [b]	m_b	Нет решения!
медиана к стороне [c]	m_c	Нет решения!
биссектриса угла A	l_a	Нет решения!
биссектриса угла B	l_b	Нет решения!
биссектриса угла C	l_c	Нет решения!
периметр треугольника	P	Нет решения!
площадь треугольника	S	Нет решения!
радиус описанной окружности	R	Нет решения!
радиус вписанной окружности	r	Нет решения!

Рисунок 3. Варианты с ошибкой

Элементы треугольника:	Дано:	Решение:
сторона [a]	3,0	3,0
сторона [b]	4,0	4,0
сторона [c]	5,0	5,0
угол A, в градусах	A	36,9
угол B, в градусах	B	53,1
угол C, в градусах	C	90,0
высота к стороне [a]	h_a	4,0
высота к стороне [b]	h_b	3,0
высота к стороне [c]	h_c	2,4
медиана к стороне [a]	m_a	4,3
медиана к стороне [b]	m_b	3,6
медиана к стороне [c]	m_c	2,5
биссектриса угла A	l_a	4,2
биссектриса угла B	l_b	3,4
биссектриса угла C	l_c	2,4
периметр треугольника	P	12,0
площадь треугольника	S	6,0
радиус описанной окружности	R	2,5
радиус вписанной окружности	r	1,0

Рисунок 2. Варианты с решением

Для наглядности введённых значений и полученных результатов, я использовал условное форматирование столбцов «Дано» и «Решение». Для запрета ввода значений в «Дано», когда решение уже есть, условным форматированием сделал запретные ячейки красными. Вводить данные в программу нужно строго сверху вниз.

Вывод. Таким образом, используя математическую программу Microsoft Excel, в единую таблицу были собраны известные и найденные формулы решений элементов треугольника без их доказательства [1, 2, 5], что позволило при правильном вводе значений в ДАНО находить необходимые элементы треугольника. При затруднении в решениях задач, можно посмотреть, как находился тот или иной элемент треугольника, вспомнить нужные формулы нажав на ссылки номеров в первой строке в таблице с формулами.

Используя программу, можно составлять свои задачи с объяснениями их решений. На главной странице, в программе также есть видео объяснения, которые помогут ознакомиться с созданием программы, если будет нужно. В апреле 2023 года программа и её описание были опубликованы на сайте <https://workproekt.ru/>.

Литература

- Атанасян Л. С. Геометрия. 7-9 классы. Москва: Просвещение, 2014. 383 с.
 - Атанасян Л. С. Математика: алгебра и начало математического анализа, геометрия. Геометрия. 10-11 классы. Москва: Просвещение, 2022. 287 с.
 - Павлов Н. Образовательный YouTube канал «PlanetaExcel». URL: <https://www.youtube.com/@planetaexcel> (дата обращения: 21.04.2026).
 - Павлов Н. Интернет-сайт «Планета Excel». URL: <https://www.planetaexcel.ru> (дата обращения: 21.04.2026)
 - Сдвижков О. А. Геометрия: практикум в Excel: учебное пособие. Москва: ИНФРА-М, 2025. 313 с. URL: <https://znanium.ru/catalog/product/1414398> (дата обращения: 21.04.2026).
- Ссылки на программу:
- Яндекс Диск. URL: <https://disk.yandex.ru/d/cuOtoXdjNdW4xg>.
 - WorkProekt.ru. URL: <https://library.workproekt.ru/work/praktiko-orientirovannyiy/426/9>.

APPLICATION OF EXCEL IN GEOMETRY: AUTOMATION OF CALCULATION OF UNKNOWN ELEMENTS OF A TRIANGLE

Kireev Artem Aleksandrovich

student,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: mig31lipeck@yandex.ru

Matytsin Artem Vladimirovich

student,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor,

Lipetsk State Technical University

Russia, Lipetsk

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. A triangle is one of the most important geometric figures widely used in science and technology. The solution of its elements is used in practice, for example, to determine the distance to an inaccessible point. The article describes how to create a program in Microsoft Excel that automatically calculates all other parameters based on the given values of the triangle's elements.

Keywords: triangle elements, Microsoft Excel formulas, Excel logical conditions and functions, conditional formatting, and solution visualization.

КЕЙС-ЧЕМПИОНАТЫ С ПРИМЕНЕНИЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА КАК СПОСОБ РАЗВИТИЯ УНИВЕРСАЛЬНЫХ КОМПЕТЕНЦИЙ ПЕДАГОГОВ ОБЩЕОБРАЗОВАТЕЛЬНОЙ ОРГАНИЗАЦИИ

© Наговицына Мария Геннадиевна

аспирант,

Омский государственный педагогический университет

Россия, 644099, г. Омск, наб. Тухачевского, 14

E-mail: mary3007@bk.ru

Аннотация. В статье предложено использование кейс-чемпионатов с применением искусственного интеллекта как эффективного способа развития универсальных компетенций педагогов. Представлен практический опыт методической службы школы-интерната для детей коренных малочисленных народов Севера. Описаны преимущества и риски метода, приведены примеры кейсов. Автор делает вывод, что рассмотренный метод в сочетании с искусственным интеллектом персонализирует методическое сопровождение и повышает профессиональное мастерство учителей.

Ключевые слова: кейс-чемпионаты, метод кейсов, универсальные компетенции, персонализация, наставничество, функциональная грамотность, нейросети.

Определение универсальных компетенций педагога активно разрабатывается в отечественной науке на протяжении двух последних десятилетий, с тех пор, как в 2001 году Минобразования России опубликовало «Стратегию модернизации содержания общего образования» [3]. В ней компетентностный подход к деятельности педагога является предпосылкой обновления образовательных стандартов России.

Наиболее точным определением универсальных компетенций в рамках нашего исследования является авторское определение к.п.н А. М. Динаева, где «универсальные компетенции рассматриваются как единство гибких компетенций (креативность, командная работа, коммуникативные навыки, эмоциональный интеллект, гибкость), цифровых компетенций (навыки цифровой безопасности, поиск, анализ, передача, хранение информации, создание цифрового контента, навыки коммуникации в Интернете) и функциональной грамотности (читательская, математическая, естественно-научная грамотность)» [2]. Здесь и далее, говоря об универсальных компетенциях, мы будем опираться на определение А. М. Динаева.

Статья поднимает проблему развития универсальных компетенций педагогов общеобразовательной организации в цифровой образовательной среде. В качестве гипотезы исследования выдвинем предположение о том, что применение искусственного интеллекта в работе школьного методиста и регулярное проведение кейс-чемпионатов для педагогов помогут развить их универсальные компетенции и заложить основу для профессионального совершенствования.

Актуальность исследования подтверждается целями и задачами «Стратегии развития образования в России до 2036 года (с перспективной до 2040)» [5]. Этот проект ещё находится на стадии доработки, но на всех этапах, с самого

начала общественных обсуждений, одним из приоритетов стратегия провозглашала развитие технологий и поддержку педагогов. О применении нейросетей в педагогике, как правило, говорят в ключе индивидуализации образования [4]. Данное же исследование обладает практической значимостью и новизной, поскольку здесь нейросети выступают как ресурс создания кейсов для конкретных запросов определённой образовательной организации с её уникальным укладом.

Объект исследования – способы развития универсальных компетенций педагогов.

Предмет – кейс-чемпионат с применением искусственного интеллекта.

Цель – представление опыта работы методической службы школы по проведению кейс-чемпионатов для педагогов. Опишем чемпионат профессиональных кейсов как ведущий механизм, с помощью которого педагоги совершенствуют гибкие, цифровые компетенции и функциональную грамотность.

Качество системы образования не может быть выше качества подготовки работающих в ней учителей [6]. Вероятно, и уровень овладения функциональной грамотностью у обучающихся не может быть выше, чем у педагогов. Значит, эти компетенции необходимо развивать в первую очередь у преподавателей. Методисты, методологи так часто говорят, пишут о функциональной грамотности у школьников, что порой не задумываются: а как с нею обстоят дела у нас, преподавателей, учителей?

Обратимся к такой форме методического сопровождения педагогов, как кейс-чемпионат. Как метод он зародился на Западе в XIX веке в вузовской среде, но быстро нашёл поклонников с сферах бизнеса, политики, управления [7]. Сочетая в себе преимущества метода кейсов и соревновательность, азартность чемпионатов, дебатов, этот способ работы служит достижению нескольких целей в образовательной организации. Продуктивное присвоение знаний и навыков может произойти только в активном системно-деятельностном подходе, чему служат школьные чемпионаты кейсов для учителей и воспитателей школьного интерната [1]. В нашей образовательной организации есть структурное подразделение-интернат для детей коренных малочисленных народов Севера (родители обучающихся ведут традиционный кочевой образ жизни, а дети девять месяцев в году проживают в интернате).

Идея кейс-чемпионатов проста: педагоги делятся на команды (деление можно провести нестандартно, развивая одно из направлений функциональной грамотности – креативное мышление). Команды получают одинаковые кейсы, имеют одинаковое количество времени для их решения, затем представляют на суд жюри (администрация школы, родители, педагоги, школьное самоуправление) свои рассуждения. Эксперты выносят вердикт, опираясь на предварительно разработанные критерии, награждают победителей.

Разрабатывать проблемные ситуации, опираясь на реальные запросы, вызовы, помогает искусственный интеллект. Грамотно заданный промпт помогает учесть особенности конкретной организации (в нашем случае это именно северная школа-интернат), особенности конкретного коллектива (например, с преобладанием молодых педагогов, не имеющих квалификационной катего-

рии), особенности контекста работы школы (например, школа в селе или с преобладающим числом высокомотивированных обучающихся).

Главное преимущество кейс-чемпионата – это возможность для педагога «прожить» универсальные компетенции на собственном опыте. Учителя оказываются в ситуации, где эти навыки необходимо применить для решения конкретной задачи.

Обсуждение кейсов развивает навыки аналитического и критического мышления, командного взаимодействия. Если пригласить коллег из других школ, чемпионат становится мощным инструментом для знакомства с практиками других организаций и даже регионов. В ситуации успеха на кейс-чемпионате может оказаться любой, даже новоиспечённый специалист, что делает такие события незаменимыми при адаптации молодых кадров. Именно с этой целью первый чемпионат в школе мы проводим сразу в начале сентября – увидеть друг друга, новых коллег, пообщаться после отпуска в деловой и в то же время творческой обстановке.

Такой формат наставничества порой дарит учителям готовые решения реально существующих проблем, позволяет отточить в теории то, с чем мы можем столкнуться завтра. Удобство кейс-чемпионатов в том, что эта практика успешно масштабируется, она гибкая по сути и может проводиться во всех типах образовательных организаций, от дошкольных до вузов.

Вот несколько примеров кейсов, предложенных нашим педагогам в этом учебном году:

1. в 7 классе на уроке алгебры учитель объясняет тему «Проценты в жизни». Группа мальчиков откровенно скучает и занимается своими делами. На замечание учителя один из них отвечает: «Зачем нам эти проценты? Мы хотим стать переводчиками, там математика не нужна». Учитель понимает, что стандартные задачи из учебника про вклады в банке подростков не вдохновляют. Нужно срочно менять подход, чтобы сохранить дисциплину и мотивацию. Придумайте серию коротких математических задач на тему процентов, но связанных с профессиональной деятельностью переводчика. Спроектируйте фрагмент урока (10 минут);

2. ученик 10 класса хвастается перед одноклассниками, что нашел «реально легкий заработок» в интернете: ему платят за перевод денег через его карту («криптообменник»). Он не понимает, почему это может быть опасно, и искренне считает себя «крутым бизнесменом». Учитель информатики случайно слышит этот разговор в коридоре. Он понимает, что подросток, скорее всего, втягивается в схему дропперства (незаконный перевод средств), но, если просто вызвать полицию или родителей, можно потерять доверие подростка и не изменить его отношение к проблеме. Как бы вы поступили?

Важно следить, чтобы готовые задания соответствовали методическим требованиям, не нарушали этические и научные принципы, не противоречили законодательству РФ.

Каковы недостатки чемпионатов по решению кейсов при развитии универсальных компетенций у педагогов?

Прежде всего следует опасаться уклона в сторону «игра ради игры»: важна рефлексия, проговаривание целей и достигнутых результатов. Также следует тщательно выбирать задания – они действительно должны ссылаться на различные аспекты универсальных компетенций (гибкие навыки, цифровая грамотность, креативность). Конечно, есть человеческий фактор: возможность конфликтов между командами, между командами и жюри, низкая вовлечённость коллектива, непрофессионализм экспертов, отсутствие постсопровождения. Риски можно минимизировать, если проводить регулярную подготовительную работу, чётко ставить цели кейс-чемпионата, быть открытыми к различным мнениям.

Результативность и практическая польза применения такого способа при развитии универсальных компетенций среди педагогов несомненны, гипотеза находит своё подтверждение.

Педагоги школы, принимавшие регулярное участие в работе кейс-чемпионатов, становились победителями и призёрами конкурсов муниципального и регионального этапов конкурсов профмастерства «Учитель года», «Педагогический дебют». Обучающиеся под руководством успешных учителей чаще добиваются успехов в научном и олимпиадном движении. Школа стала региональной стажировочной площадкой по наставничеству в образовании.

Анализ современных подходов к повышению профессионализма, в частности, в области формирования компетенций, приводит к пониманию необходимости смены доминирующей модели методического сопровождения педагогов. Традиционная вертикаль «методист – учитель» имеет существенные ограничения, а вот методология «равный – равному» (peer-to-peer), обучение в деятельности с опорой на цифровые инструменты более эффективна и целенаправленна. Искусственный интеллект персонализирует традиционный метод кейсов и погружает участников чемпионата в ситуации, приближенные к реальным в процессе преподавания и воспитания.

Литература

1. Бердникова А. Н. Формирование универсальных компетенций в педагогическом образовании: теоретические основы и подходы // Вестник науки. 2025. Т. 3, № 1(82). С. 561-568. URL: <https://www.vestnik-nauki.pf/article/20864> (дата обращения: 19.05.2026).
2. Динаев А. М. Автореферат диссертации на соискание ученой степени кандидата педагогических наук «Развитие универсальных компетенций будущего учителя в инновационной среде педагогической мастерской», г. Грозный, 2024. URL: <https://chspu.ru/wp-content/uploads/2024/10/автореферат-Динаев.pdf> (дата обращения: 19.05.2026).
3. Распоряжение Правительства РФ от 29.12.2001 N 1756-р «О Концепции модернизации российского образования на период до 2010 года». URL: <https://www.consultant.ru/> (дата обращения: 18.05.2026 г.).
4. Рого Г. Э., Рого Э. В., Использование технологий искусственного интеллекта для решения проблемы индивидуализации образования // StudArctic Forum. 2024. Т. 9, № 1. С. 87-94.
5. Стратегия развития образования Российской Федерации до 2036 года (с перспективой до 2040 года). URL: <https://образование.объясняем.рф/#goals> (дата обращения: 18.05.2026).
6. URL: https://elar.uspu.ru/bitstream/ru-uspu/12971/1/edscience_2015_3_122_002.pdf (дата обращения: 21.05.2026).
7. URL: <https://olymp.hse.ru/championship/about> (дата обращения: 19.05. 2026).

CASE CHAMPIONSHIPS USING ARTIFICIAL INTELLIGENCE AS A WAY TO DEVELOP UNIVERSAL COMPETENCIES OF TEACHERS IN EDUCATIONAL ORGANIZATIONS

Nagovitsyna Maria Gennadievna

Postgraduate Student,

Omsk State Pedagogical University

Russia, 644099, Omsk, Naberezhnaya Tukhachevskogo, 14

E-mail: mary3007@bk.ru

Abstract. The article proposes the use of case championships with the use of artificial intelligence as an effective way to develop universal competencies of teachers. The practical experience of the methodological service of a boarding school for children of indigenous small peoples of the North is presented. The advantages and risks of the method are described, and examples of cases are given. The author concludes that the method under consideration, when combined with artificial intelligence, personalizes methodological support and improves the professional skills of teachers.

Keywords: case championships, case method, universal competencies, personalization, mentoring, functional literacy, and neural networks.

ДВА ЯЗЫКА – ОДНО РЕШЕНИЕ: КОМБИНИРОВАННЫЙ ПОДХОД К ЗАДАНИЮ №6 К ЕГЭ ПО ИНФОРМАТИКЕ

© **Немчинова Татьяна Владимировна**

кандидат педагогических наук, доцент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: ntv05@mail.ru

© **Гармаева Оюна Алексеевна**

старший преподаватель,

БГСХА им. В.Р. Филиппова

Россия, г. Улан-Удэ,

garmaeva_oa@mail.ru

Аннотация. Актуальность исследования обусловлена изменениями рекомендованного перечня программного обеспечения для подготовки и проведения КЕГЭ по информатике (исключение среды «Кумир» с 2026 года), а также сохраняющейся нестабильной динамикой выполнения задания №6. Аналитическое решение геометрических задач, связанных с подсчетом целочисленных точек, площади и периметра фигур, затруднено для значительной части экзаменуемых из-за недостаточного уровня пространственного мышления и навыков формализации. В статье рассматривается комбинированный метод решения задания №6 КЕГЭ, объединяющая визуализацию траектории исполнителя «Черепаша» в среде PascalABC и автоматизированного вычисления геометрических характеристик на языке Python. Рассмотрены два типовых примера с сайта К.Ю. Полякова. Разработанная методика может быть рекомендована учителям информатики и репетиторам для подготовки учащихся к КЕГЭ, а также самим выпускникам как эффективный инструмент для самопроверки.

Ключевые слова: КЕГЭ, информатика, задание № 6, исполнитель «Черепаша», PascalABC, Python, комбинированный метод, целочисленные точки, площадь, периметр

В 2023 году в демонстрационной версии контрольно-измерительных материалов ЕГЭ по информатике появилось новое задание для исполнителя «Черепаша». Выполнение данного задания требует от выпускников навыков пространственного мышления и понимания геометрических построений. Внедрение компьютерной модели государственной итоговой аттестации по информатике (2021 г.) дало возможность участникам экзамена применять вспомогательное программное обеспечение для решения ряда задач, в частности для выполнения задания №6 КЕГЭ стало возможным использование среды программирования Кумир.

Согласно данным аналитических отчетов средний процент выполнения задания №6 составил: в 2023 году – 21,6% [2], 2024 – 41,5% [3] и 2025 г. – 39,8% [4]. Приведенные данные свидетельствуют о нестабильной динамике результатов и сохраняющихся трудностях у значительной части экзаменуемых.

К типичным ошибкам, допускаемым при решении задания №6 можно отнести следующие:

Категория ошибки	Проявление
Ошибки алгоритмического и логического мышления	Путаница объединения и пересечения фигур; неумение выделять результат из условия
Ошибки в работе с исполнителем «Черепаша» в среде КУМИР	Забывают поднять и опустить хвост; неверно определяют длины сторон и углы поворота, неверно выставляют масштаб
Ошибки в вычислениях и формулах	Неправильный подсчет периметра и площади фигуры; неверное использование формулы Пика

В 2026 году Федеральная служба оп надзору в сфере образования (Рособрна-дзор) исключила язык и среду программирования «Кумир» из числа рекомен-дуемых для подготовки и проведения КЕГЭ. Вместе с тем только аналитиче-ское (без использования средств для автоматизации) решение задания, осно-ванное лишь на геометрических построениях, не всегда является удобными и может приводить к ошибкам при определении итогового ответа. Так, в извест-ных методических разработках (например, в разборах Алексея Кабанова, Лени Шастина и др.) предложен подход, базирующийся исключительно на языке Ру-thon, в частности на использовании библиотеки turtle для визуализации траек-тории исполнителя и последующего анализа. Однако как показывает практика, для значительной части экзаменуемых реализация такого подхода сопряжена рядом затруднений. Во-первых, написания корректного кода на Python с ис-пользованием библиотеки turtle требует уверенных навыков программирования, включая понимание системы координат, управления циклом и условиями, что не всегда сформировано у школьников, особенно на базовом уровне подготов-ки. Во-вторых, даже при успешной визуализации траектории последующий аналитический подсчет целочисленных точек (особенно при сложной конфигу-рации фигуры) или определении длин сторон вызывает серьезные ошибки, свя-занные с неточностью восприятия графического вывода и неумением формали-зовать геометрические характеристики фигуры. Монолитный подход на основе одного языка Python, при всей его наглядности остается затруднительным для большинства сдающих, что подтверждается стабильно низкими результатами выполнения задания №6 в 2023-2025 гг.

На наш взгляд для более эффективного выполнения задания №6 целесооб-разно применять комбинированный подход, объединяющий достоинства раз-ных инструментов: язык программирования PascalABC - для построения фигу-ры по заданному алгоритму исполнителя «Черепаша», и язык Python - для по-следующего подсчета точек(например, количества целочисленных точек внутри фигуры или на ее границе), также вычисления площади или периметра образо-вавшихся фигур.

Для систематизации работы может быть использован опорный конспект (рис.1), содержащий перечень команд исполнителя «Черепаша» в среде Pas-

calABC, а также стандартные алгоритмы на языке Python, предназначенные для подсчета точек.

Рассмотрим алгоритм решения на конкретных примерах¹.

Общее условие для всех типов задач (типовая формулировка задания №6 КЕГЭ с исполнителем «Черепаша») [6]:

Исполнитель Черепаша действует на плоскости с декартовой системой координат. В начальный момент Черепаша находится в начале координат, её голова направлена вдоль положительного направления оси ординат, хвост опущен. При опущенном хвосте Черепаша оставляет на поле след в виде линии.

ЗАДАНИЕ №6. ЦИКЛИЧЕСКИЕ АЛГОРИТМЫ ДЛЯ ИСПОЛНИТЕЛЯ

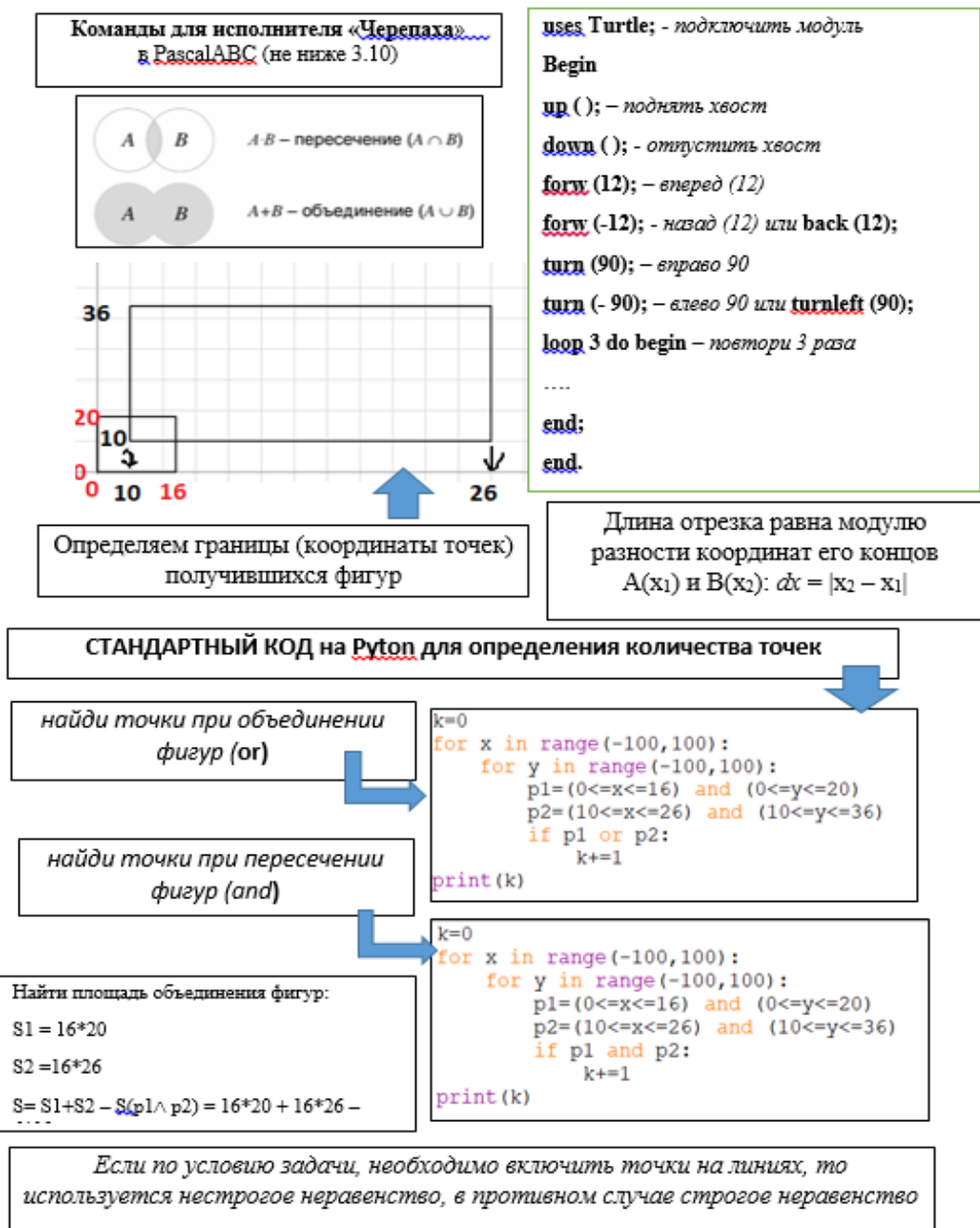


Рисунок 1. Опорный конспект

¹ Примеры взяты с сайта К.Ю. Полякова

В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существует 6 команд: Поднять хвост, означающая переход к перемещению без рисования; Опустить хвост, означающая переход в режим рисования; Вперёд n (где n – целое число), вызывающая передвижение Черепахи на n единиц в том направлении, куда указывает её голова; Назад n (где n – целое число), вызывающая передвижение в противоположном голове направлении; Направо m (где m – целое число), вызывающая изменение направления движения на m градусов по часовой стрелке, Налево m (где m – целое число), вызывающая изменение направления движения на m градусов против часовой стрелки.

Запись

Повтори k [Команда1 Команда2 ... КомандаS]

означает, что последовательность из S команд повторится k раз.

Пример №1. (Демо 2026). Черепахе был дан для исполнения следующий алгоритм [6]:

Повтори 2 [Вперёд 14 Налево 270 Назад 12 Направо 90]

Поднять хвост

Вперёд 9 Направо 90 Назад 7 Налево 90

Опустить хвост

Повтори 2 [Вперёд 13 Направо 90 Вперёд 6 Направо 90]

Определите, сколько точек с целочисленными координатами находятся внутри объединения фигур, ограниченного заданными алгоритмом линиями, включая точки на линиях.

Решение комбинированным методом:

1. Построение фигуры в PascalABC.

Пишем программу (рис.2), имитирующую движение Черепахи (используя графический модуль) и определения границ полученной фигуры.

```
uses Turtle;
Begin
  down();
  loop 2 do begin
    forw(14);
    turn(-270);
    forw(-12);
    turn(90);
  end;
  up();
  forw(9);
  turn(90);
  forw(-7);
  turn(-90);
  down();
  loop 2 do begin
    forw(13);
    turn(90);
    forw(6);
    turn(90);
  end;
end.
```

Рисунок 2. Листинг программы на языке PascalABC для Примера 1.

После выполнения программы на экране появляется графическое изображение (рис.3), позволяющее определить координаты границы каждого прямоугольника:

P1 ($-12 \leq x \leq 0$ И $0 \leq y \leq 14$)

P2 ($-7 \leq x \leq -1$ И $9 \leq y \leq 22$).

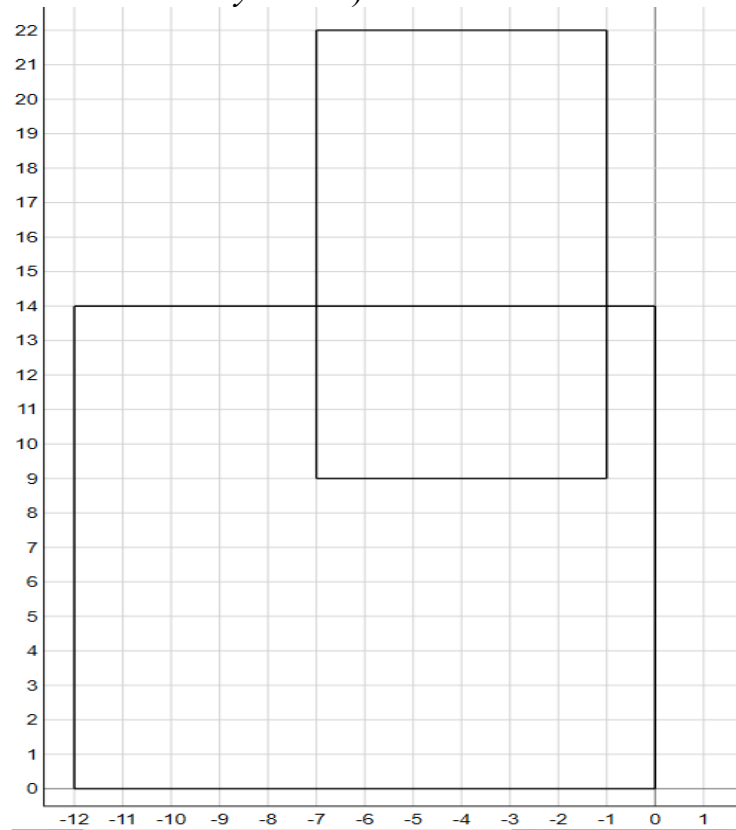


Рисунок 3. Реализация кода на языке PascalABC для Примера 1.

2. Программа на Python (рис.4) для подсчета целочисленных точек внутри объединения фигур.

251

```

6.py - C:/Users/tvnemchinova/Desktop/6.py (3.13.0)
File Edit Format Run Options Window Help

k=0
for x in range(-100,100):
    for y in range(-100,100):
        p1=(-12 <= x <=0) and (0 <= y <=14)
        p2 = (-7 <= x <=-1) and (9 <= y <=22)
        if p1 or p2:
            k+=1
print(k)

```

Рисунок 4. Скрин программы на языке Python для Примера 1.

Пример №2. Черепахе был дан для исполнения следующий алгоритм [6]:

Направо 180

Повтори 9 [Вперёд 45 Налево 90 Вперёд 91 Налево 90]

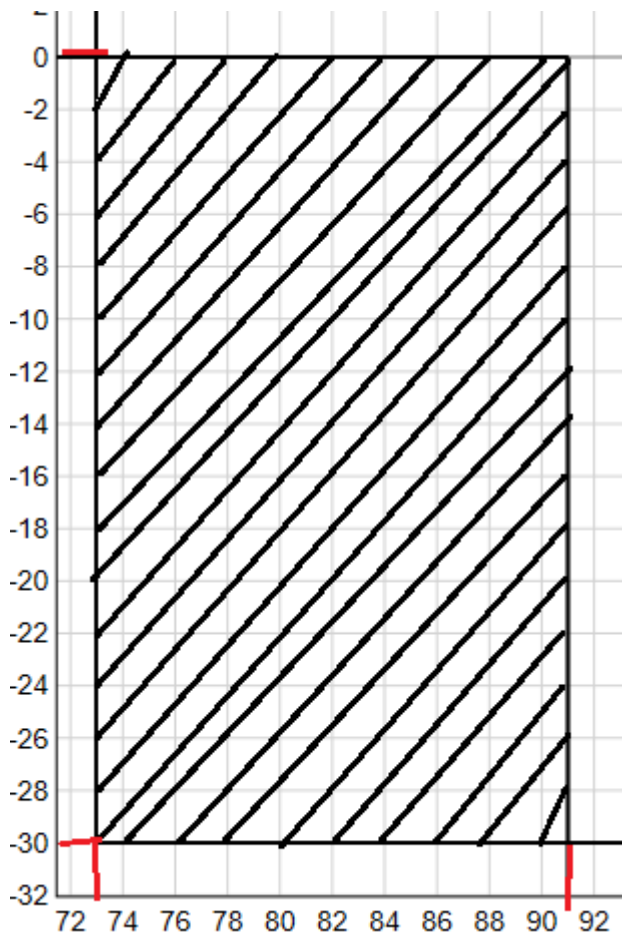
Поднять хвост

Вперёд 15 Налево 90 Вперёд 18 Направо 90

Опустить хвост

Повтори 9 [Вперёд 128 Направо 90 Вперёд 92 Направо 90]

Определите периметр области пересечения фигур, ограниченных заданными алгоритмом линиями.



```
uses Turtle;
Begin
  down();
  turn(180);
  loop 9 do begin
    forw(45);
    turn(-90);
    forw(91);
    turn(-90);
  end;
  up();
  forw(15);
  turn(-90);
  forw(18);
  turn(90);
  down();
  loop 9 do begin
    forw(128);
    turn(90);
    forw(92);
    turn(90);
  end;
end.
```

Рисунок 5. Реализация кода на языке PascalABC и листинг программы для Примера 2.

По полученному графическому изображению (рис.5) определяем границы (координаты точек) заштрихованной фигуры:

$(73 \leq x \leq 91) \text{ И } (-30 \leq y \leq 0)$.

Длина прямоугольника по абсиссе (x) равна $dx = |91 - 73| = 18$. Длина прямоугольника по ординате (y): $dy = |-30 - 0| = 30$.

Периметр прямоугольника, это сумма всех его сторон, значит $P = (18 + 30) * 2 = 96$.

Таким образом, предложенный комбинированный подход, интегрирующий визуализацию траектории исполнителя «Черепаха» в среде PascalABC (модуль Turtle) и последующий количественный анализ средствами Python позволяет

эффективно преодолеть основные затруднения, возникающие у экзаменуемых при решении задания №6 КЕГЭ. В отличие от монолитных методов, базирующихся исключительно на одном языке программирования или на аналитическом ручном счете, предлагаемая методика позволит учащимся со средним уровнем подготовки выполнить данное задание: визуализация в PascalABC не требует сложных графических алгоритмов, а скрипты на Python могут быть представлены в виде готовых шаблонов с минимальными возможностями модификации. Экспериментальная проверка на примерах, взятых с открытого банка ФИПИ подтвердила надежность и воспроизводимость результатов.

Литература

1. Крылов С. С., Чуркина Т. Е. ЕГЭ 2026. Информатика: типовые экзаменационные варианты. Москва: Национальное образование. 2026. 29 с.
2. Крылов С. С. Методические рекомендации для учителей, подготовленные на основе анализа типичных ошибок участников ЕГЭ 2023 года. URL: https://doc.fipi.ru/ege/analiticheskie-i-metodicheskie-materialy/2023/inf_mr_2023.pdf
3. Крылов С. С. Методические рекомендации для учителей, подготовленные на основе анализа типичных ошибок участников ЕГЭ 2024 года. URL: https://doc.fipi.ru/ege/analiticheskie-i-metodicheskie-materialy/2024/inf_mr_2024.pdf
4. Крылов С. С. Методические рекомендации для учителей, подготовленные на основе анализа типичных ошибок участников ЕГЭ 2025 года. URL: https://doc.fipi.ru/ege/analiticheskie-i-metodicheskie-materialy/2025/inf_mr_2025.pdf
5. Немчинова Т. В., Цыбикова Т. С. Дидактический потенциал опорных конспектов в системе подготовки, обучающихся к единому государственному экзамену по информатике. // Педагогическое образование. 2026. Т. 7, № 1. С.321-327
6. Поляков Ю. К. Сайт для подготовке к ЕГЭ по информатике. URL: <https://kpolyakov.spb.ru/school/ege/generate.htm>

TWO LANGUAGES – ONE SOLUTION: A COMBINED APPROACH TO EXAMINATION QUESTION 6 IN INFORMATICS

Nemchinova Tatyana Vladimirovna

Candidate of Pedagogical Sciences, Associate Professor,
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: ntv05@mail.ru

Garmayeva Oyuna Alekseevna

Senior Lecturer,
V.R. Filippov Baikal State Agricultural University
Russia, Ulan-Ude,
garmayeva_oa@mail.ru

Abstract. The relevance of the study is due to the changes in the recommended list of software for preparing and conducting the KGEI in computer science (the elimination of the "Kumir" environment starting from 2026), as well as the continued unstable performance of task 6. The analytical solution of geometric problems related to calculating integer points, area, and perimeter of figures is difficult for a significant portion of examinees due to their insufficient

level of spatial thinking and formalization skills. The article discusses a combined method for solving task 6 of the KGEU, which combines the visualization of the Turtle's trajectory in the PascalABC environment and the automated calculation of geometric characteristics in Python. Two typical examples from the website of K.Yu. Polyakov are considered. The developed methodology can be recommended to computer science teachers and tutors for preparing students for the KGEU, as well as to graduates themselves as an effective tool for self-checking.

Keywords: KGEU, computer science, task 6, Turtle performer, PascalABC, Python, combined method, integer points, area, perimeter.

ФОРМАТИВНОЕ ОЦЕНИВАНИЕ КАК СРЕДСТВО ПОВЫШЕНИЯ УЧЕБНЫХ ДОСТИЖЕНИЙ УЧАЩИХСЯ В РАМКАХ ОБНОВЛЕННОГО СОДЕРЖАНИЯ ОБРАЗОВАНИЯ

© **Байдабекова Нуржамал Толебековна**

заместитель директора по учебно-воспитательной работе,
Коммунальное государственное учреждение «Общеобразовательная школа № 63» отдела
образования Сарыагашского района управления образования Туркестанской области
Республика Казахстан, 160911, Туркестанская область, Сарыагашский район,
Дарбазинский сельский округ, с. Дарбаза, ул. Егемен Казакстан 11
E-mail: zhaksygul.tastanova@mail.ru

© **Цыбикова Туяна Сандаликовна**

кандидат педагогических наук, доцент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина 24а
E-mail: cts2001@mail.ru

Аннотация. В статье описана работа по определению эффективности влияния методов формативного оценивания на повышение учебных достижений учащихся. В ходе исследования авторами было организовано изучение содержания обновленного образования, оказана профессиональная поддержка учителям в процессе обсуждения эффективных методов формативного оценивания, проведено обучение планированию уроков с использованием данных методов, направленных на улучшение результатов обучения, а также реализованы коучинговые занятия для совершенствования профессиональных знаний педагогов. Ожидаемыми результатами работы стали: осознание учителями важности оценивания для обучения и его влияния на учебный процесс, систематическое применение методов формативного оценивания на уроках, что способствовало улучшению учебных достижений учащихся и развитию практики использования формативного оценивания в школьном сообществе.

Ключевые слова: формативное оценивание, учебные достижения, коучинг, обратная связь, методы оценивания, учащиеся, обновленное содержание.

Актуальность и причины исследования. Внедрение обновленного содержания образования в школьную практику ставит перед нами масштабные вопросы. Учителей волнуют такие вопросы, как: «Как организовать обучение учащихся?», «Что мне делать, если учащиеся не усваивают материал, какие действия предпринять, чтобы улучшить обучение учащихся?». Поскольку одной из целей внедрения формативного оценивания в школе является улучшение обучения, учителя должны знать, как использовать эти методы формативного оценивания на уроках. Возможно, они знают о формативном оценивании, но не имеют полного понимания того, как эффективно применять эти методы на уроке сформировалось? Наверное, это одна из глубоких идей обновленного образования? Если учитель не использует методы формативного оценивания на практике и не формирует у учащихся глубокое понимание обучения, то улучшить обучение на основе опыта невозможно.

Тогда каково качество применения формативного оценивания учителями? Каково мнение большинства учителей школы — молодых учителей, учителей со стажем, учителей с высшим образованием? Следовательно, в школьном сообществе большинству учителей необходима поддержка. Учитывая это, с целью улучшения процесса преподавания и обучения в школе, я пришла к выводу, что в ходе коучинга необходимо повышать профессиональные навыки учителей, поддерживать их в использовании методов критериального оценивания для улучшения обучения учащихся и формировать у них правильное понимание. Наряду с этим, проанализировав проблемы низких результатов успеваемости учащихся в некоторых классах по итогам четверти в школьной среде, я пришла к выводу о необходимости проведения совместной работы с учителями.

В качестве объекта исследования были взяты педагогические действия учителей-предметников школы.

Мы согласны с позицией исследователей П. Блэк и Д. Уильям, что: «Оценка, полученная учащимися, направлена на улучшение их собственного понимания, а не только на сравнение результатов; это помогает им осознать свои первые шаги» [2].

А. Алимов в своей работе относительно оценивания выразил следующее мнение: «Под результатами обучения мы понимаем демонстрацию учащимися в ходе обучения усвоенных ими компетенций на различных уровнях. Для измерения и оценки того, что они знают и умеют делать, требуются критерии, определяющие их учебные достижения. В связи с этим должна измениться и деятельность учителя: в оценивании должны присутствовать ответы на вопросы "что?", "кто?", "как?", "когда?", "для чего?"». Если учитель уверен в ценности формативного оценивания, то активность учащихся возрастает, повышается качество знаний [1].

Ознакомившись с трудами автора, я сделала выводы о методах оценивания: написание эссе, пятистрочный стих (синквейн), четыре предложения, карта INSERT, таблица «ЗХУ», стратегия «Карусель», мысленный штурм, прогнозирование, кластеры, ДЖИГСО и т.д. Эти методы помогли мне при планировании коучинга.

Для определения текущей ситуации по актуальной проблеме перед внедрением изменений, несомненно, необходима достоверная информация. Для сбора данных я использовала качественные и количественные методы исследования:

- анкетирование;
- наблюдение уроков;
- интервью со структурированными вопросами;
- данные об успеваемости учащихся.

Один из удобных способов сбора данных с целью определения реальной ситуации — это анкетирование. Преимущество анкетирования в том, что оно позволяет собрать большой объем информации за один раз с помощью правильно направленных вопросов. После завершения работы я сравнила результаты анкетирования с первоначальными данными и увидела конкретные изменения в школьной среде. Поэтому я посчитала правильным определить реальную ситуацию через анкетирование. В анкетировании приняли участие 55 учителей школы.

Результаты анкетирования

Вопросы	Всегда	Иногда	Никогда
Использую методы, предназначенные для формативного оценивания	9% (5)	22% (12)	69% (38)
Использую методы ССП (КБӨ) для улучшения обучения учащихся	8% (4)	20% (11)	72% (40)
Знаю, как эффективно применять ССП (КБӨ) в групповой работе	15% (8)	29% (16)	56% (31)
При использовании ССП (КБӨ) учитываю голос учащегося	10% (6)	26% (14)	64% (35)

(На графике): Показатели ответов «Никогда» в анкете для учителей

- 69% — Использую методы, предназначенные для оценивания
- 72% — Отношения обратной связи для улучшения обучения учащихся
- 56% — В групповой работе умею эффективно использовать
- 64% — Учитываю голос учащихся

Результаты анкетирования показали уровень использования школьными учителями методов формативного оценивания, уровень оценивания, рассмотрение различных видов оценивания с целью повышения качества знаний учащихся, возможность проведения формативного оценивания в ходе урока. На основе этого была определена необходимость проведения исследования с целью улучшения данного процесса.

Наблюдение уроков

Наблюдение за уроками также является одним из способов определения проблемы. Наблюдая за уроками, можно собрать информацию о том, как прошел урок, проанализировать его и сделать выводы. Поэтому этот метод широко используется в школьной практике.

В ходе исследования было проведено наблюдение более чем за 20 уроками:

- использование методов, предназначенных для формативного оценивания;
- совместно с учителями использовалась графическая карта наблюдения урока для оценивания.

Поскольку использование этой графической карты наблюдения урока дает учителю четкое представление о том, как прошел урок, или наглядно показывает через рисунок, какой ученик и как принимал участие в уроке, я посчитала целесообразным реализовать это на уроке.

В ходе проведения интервью с учителями были заданы следующие вопросы:

1. Повлияли ли методы формативного оценивания, использованные на вашем уроке, на учащихся?
2. Повлияли ли методы формативного оценивания на вовлеченность учащихся в урок?
3. Помогло ли вам коучинг-занятие «Как организовать использование методов формативного оценивания в оценивании?»

В ходе анализа ответов учителей было выявлено, что у них существуют препятствия при использовании методов формативного оценивания:

- у учителей отсутствует четкое понимание целей формативного оценивания;

- в процессе использования методов формативного оценивания учебные достижения учащихся организуются формально, без обратной связи между учителем и учеником.

План действий по реализации изменений и отслеживанию результатов

Для устранения препятствий, указанных выше, возникла необходимость планирования мероприятий с целью повышения учебных достижений учащихся посредством применения формативного оценивания.

Таблица 2

План мероприятий

№	Содержание мероприятия	Сроки	Вид мероприятия
1	Определение первоначального состояния школы	Сентябрь	Изучение годового плана школы
2	Определение исследовательского вопроса	Сентябрь	На заседании методсовета
3	Сбор литературы по исследовательскому	Сентябрь	Сбор данных
4	Проведение анкетирования для определения первоначальной проблемы	Октябрь	Сбор данных
5	Организация наблюдения уроков для определения первоначальной проблемы	Октябрь	Наблюдение уроков
6	Организация мероприятий по исследовательскому вопросу	Ноябрь	Составление плана коучинга
7	Важность формативного оценивания	Ноябрь	Проведение коучинга
8	Использование методов формативного оценивания	Декабрь	Проведение коучинга
9	Важность обратной связи в формативном оценивании	Январь	Проведение коучинга
10	Анализ полученных данных и разъяснение	Февраль	Анализ полученных данных
11	Наблюдение уроков	Март	Наблюдение уроков
12	Интервью с учителями	Апрель	Интервью
13	Голос ученика	Апрель	Голос ученика
14	Рефлексия. Анализ внесенных изменений	Май	Написание отчета

Первоначальные данные, полученные из этих источников, легли в основу моего исследовательского вопроса: «Как эффективное использование методов формативного оценивания учителями влияет на повышение учебных достижений учащихся?». С этой целью был скорректирован план работы школы.

Для решения проблемы в рамках темы «Важность формативного оценивания», «Применение методов формативного оценивания», «Важность обратной связи в формативном оценивании» я провела серию последовательных коучинг-занятий.

Первый коучинг: Тема «Важность формативного оценивания».

- Цель: Разъяснить учителям важность формативного оценивания.

В ходе коучинг-сессии был задан вопрос: «Для чего нужно оценивание?», проведен мозговой штурм с использованием метода «Ассоциации», заслушаны мнения учителей. По 2-му заданию был дан вопрос: «Каким должно быть оценивание?», «Для чего нужно оценивание?». Учителя обсудили эти вопросы в

группах и презентовали свои ответы. Через метод «Светофор» группы оценили друг друга, обосновывая, что именно они оценили и почему.

Учителя поняли важность оценивания. На этапе рефлексии после коучинг-сессии, отвечая на вопросы: «Что...? Как...? Для чего...? Зачем...?», они использовали фразы из слов обратной связи. На основе рефлексии учителей я заметила, что у них повысился интерес к обучению.

Второй коучинг: Тема «Использование методов формативного оценивания на уроках».

- Цель: Познакомить учителей с методами формативного оценивания, внести изменения в планы уроков, внедрить элементы оценивания в коучинг-сессию.

Был использован метод «Мысленный шторм» по теме «Учиться — это не только возможно, но и вероятно», заслушаны мнения учителей. По 2-му заданию группе 1 было дано задание дать определение понятию «Прогнозирование», группе 2 — дать определение понятию «Кластер», группе 3 — дать определение понятию «Инсерт». Текст был разделен на три части, в конце каждой части задавался вопрос по прогнозированию. Текст предоставлялся после просмотра видеоматериала. Группе 3 было дано задание составить кластер по теме видеоматериала, заполнить таблицу «ЗХУ» по содержанию текста.

Задание 3 — топик «Кластер» (групповая работа). Задание 4 — карта «INSERT» (парная работа). Каждый учитель оценивал свои задания с помощью методов «Инсерт», «ЗХУ», «ЗХУ», графического органайзера «Две звезды, одно пожелание». Каждая пара презентовала свою работу остальным.

Учителя оценили предложенные задания по критериям. Я заметила, что учителя при оценивании заданий «Инсерт», «ЗХУ», «ЗХУ» обратили внимание на особенности понимания материала друг другом. Метод «Две звезды, одно пожелание» показал, насколько эффективны данные методы в коучинге поняли, насколько эффективны методы формативного оценивания на основе рефлексии учителей после коучинга. Отзывы педагогов:

- «Метод "INSERT" очень помог в начале урока и при переходе к новой теме»;
- «С помощью метода "ЗХУ" учащиеся могут исправить свои ошибки, проанализировать и развить интерес к уроку, а метод "ЗХУ" направляет учащихся на поиск и систематизирует знания»;
- «Метод "ЗХУ" помогает точно определить, что знает ученик, что ему нужно закрепить и чему он научился».

Третий коучинг: Тема «Важность обратной связи в формативном оценивании».

- Цель: Разъяснить важность обратной связи в формативном оценивании, научить применять её на уроках. На основе предыдущих коучингов заполнить стикеры с ответами на вопросы учителей «Что я узнал? Чему я научился? Что я хочу узнать в будущем?».

Какая польза от обратной связи на эти вопросы? Ответ на этот вопрос показывает, что понимает учитель. Посредством вопросов учителя дают обратную связь о важности обратной связи, высказывают свои мнения. Я научила их разнообразить вопросы и давать обратную связь на их основе. Я заметила, что после этого коучинга учителя стали давать эффективную обратную связь.

А. учитель: «Через обратную связь учащихся я вижу такие моменты, как "замечательно, отлично, хорошо", это дает им мотивацию к обучению. Теперь я поняла, что нужно конкретно указывать ученику, что ему нужно улучшить, и давать эффективную обратную связь».

Этап анализа данных

В ходе исследования я увидела пути решения многих проблем в школьной среде. Для этого я поняла, что в школьной среде необходимо рассматривать процесс коучинга как системное явление. В ходе проведения трех коучингов я увидела, что все методы формативного оценивания эффективны. Учителя поняли важность обратной связи при оценивании, у них изменились взгляды, они научились давать обратную связь. Они сравнили отношения обратной связи, которые использовали ранее, с тем, как они дают обратную связь сейчас, однако некоторые учителя отметили, что давать обратную связь — это непросто.

В ходе второго коучинга учителя научились применять методы «Инсерт», «ЗХУ», «ЗХУ» и поняли, как показывать критерии друг другу. Я увидела эффективность этих методов, наблюдая за ними в процессе работы и объясняла, на каком этапе урока его целесообразно использовать. Исходя из этого, я рассмотрела методы внедрения навыков формативного оценивания в практику на уроках учителей и наблюдала за изменениями в их действиях. Я увидела, что учителя коучинга заинтересованы в изменениях.

Сказав: «Мое достижение в том, что...», учителя за короткий промежуток времени пересмотрели свои краткосрочные планы уроков и начали вносить изменения после коучинг-занятий. Они начали правильно подбирать методы. Я заметила их стремление к изменениям.

Я не смогла посетить уроки всех учителей, прошедших коучинг, так как результаты, ожидаемые от моего исследования, все еще внедряются и учителя только адаптируются к этому процессу.

Интересно то, что большинство учителей сразу включились в работу, не думая о том, что это сложно, и число учителей, стремящихся использовать изученные методы, увеличилось.

После коучинга я посетила 18 уроков. Я наблюдала за тремя уроками каждого учителя. В ходе посещения уроков я увидела, что методы формативного оценивания используются эффективно. В ходе уроков учителя стали демонстрировать навыки применения этих методов. Учителя начали диверсифицировать методы преподавания после проведения трех последовательных уроков «коучинг — три урока». Я подумала, что этот урок нужно исследовать в будущем. На этих уроках я увидела влияние методов на успеваемость учащихся. С этой целью я провела исследование уроков 3 учащихся. Причиной исследования этих учащихся стало то, что они полностью вовлеклись в процесс урока. Они начали активно участвовать в уроках. В ходе проведения исследования уроков (Lesson Study) я составила графическую карту наблюдения за уроком (Рисунок 1) с целью отслеживания изменений в их действиях. Графическая карта наблюдения позволила наглядно увидеть эффективность методов преподавания учителя. Коучинг помог учителям понять эффективность графической карты. Учителя

без затруднений приняли карту наблюдения, согласились с предложениями и высказали пожелания по улучшению качества уроков.

«Мой урок изменился, мне нужна помощь» — количество учителей с таким мнением увеличилось.

После завершения уроков был заслушан «голос ученика»:

- Что ты узнал на сегодняшнем уроке? Анализировать, делать выводы, решать уравнения, сравнивать с жизнью — 20%.
- Какое из заданий тебе было интересно? Работа с видеоматериалом, рисунками — 24%.
- Требовалась ли тебе помощь при выполнении заданий? Да, нужна была помощь — 30%.

Они отметили, что им понравилась работа в группах, совместное составление кластеров с одноклассниками, совместное составление вопросов. А теперь, если проанализировать вторую часть таблицы INSERT после прослушивания уроков, то на основе заполнения таблицы INSERT учащиеся определили связь предметов друг с другом, высказали мнения о том, чего они не знают, и отметили, что такие уроки им нравятся. Такие обратные связи определяют правильность направления исследования.

Выводы: В ходе исследования в действии методы формативного оценивания, применяемые учителями, вызвали у меня большой интерес. Наряду с этим, меня заинтересовало, как меняются действия учащихся. Поэтому я исследовала эти проблемы вместе с учителями. В ходе исследования я увидела, что у большинства молодых учителей повысился интерес к планированию уроков и проведению совместных уроков. Для меня было большой радостью видеть использование методов формативного оценивания на уроках. Я направила свои силы на оказание методической помощи учителям в использовании коучинга, менторинга. Я поняла, что учителя в любое время готовы помогать друг другу и оказывать поддержку.

Учителя приступили к внедрению методов формативного оценивания для улучшения обучения учащихся. Я сама задала себе вопрос: «Улучшилось ли применение формативного оценивания в моей практике?». Да, улучшилось. Доказательство тому — уровень знаний учащихся в классе повышается с каждой четвертью. Таким образом, вновь полученные результаты анкетирования показали, что у учителей сформировалось стремление к исследовательской деятельности, они осознали важность педагогической практики.

Как показала практика, количество учителей, эффективно использующих методы формативного оценивания в исследовании, увеличилось. Один из учителей отметил: «По результатам моего урока, когда я наблюдала за исследованием, количество учащихся, принимающих участие в уроке, увеличилось. Я поняла, что обратная связь важна как для меня, так и для учащихся, они стали замечать свои достижения».

Опыт показывает, что формативное оценивание является эффективным методом. Практическая значимость педагогического исследования доказала свою целесообразность. «Педагогическая практика, прежде всего, с помощью методов делает нас сильнее — учителей. Предложения учителей по улучшению ре-

зультатов обучения увеличиваются. Следовательно, это оказывает положительное влияние на учебные достижения учащихся», — эти слова Альфреда Бине подтверждают данную мысль. Этот процесс развивает наше внимание, память, мышление, логику, делает нас более мудрыми и практичными по сравнению с прошлым.

Литература

1. Алимов А. К. Использование интерактивных методов обучения в школе: учебное пособие. Астана, 2014. 188 с.
2. Блэк П., Уильям Д. Внутри Черной Рамки: Повышение Стандартов через аудиторное оценивание // Фи Дельта Капан. 1998. Октябрь. С. 139–148. URL: https://dylanwiliam.org/Dylan_Wiliams_website/Publications_files/13_L2W2D4_%D0%A7%D1%82%D0%B5%D0%BD%D0%B8%D0%B5_%D0%A0%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%20%D0%B2%D0%BD%D1%83%D1%82%D1%80%D0%B8%20%C2%AB%D0%A7%D0%B5%D1%80%D0%BD%D0%BE%D0%B3%D0%BE%20%D1%8F%D1%89%D0%B8%D0%BA%D0%B0%C2%BB_rus.pdf (дата обращения: 10.08.2023).
3. Оценивание учебных достижений учащихся: методическое руководство / сост. Р. Х. Шакиров, А. А. Буркитова, О. И. Дудкина. Бишкек: Билим, 2012.
4. Руководство по формативному и внутреннему суммативному оцениванию. Астана, 2015.
5. Формативное оценивание в начальной школе: практическое пособие для учителя / сост. О. И. Дудкина, А. А. Буркитова, Р. Х. Шакиров. Бишкек: Билим, 2012.
6. Чекетаева Р. С., Шаянбаева Ж. А., Оразбаева А. М. Оценивание для обучения и оценивание обучения: новый взгляд : методическое пособие. Усть-Каменогорск, 2013.

FORMATIVE ASSESSMENT AS A MEANS OF IMPROVING STUDENTS' EDUCATIONAL ACHIEVEMENTS IN THE FRAMEWORK OF THE UPDATED EDUCATIONAL CONTENT

Baidabekova Nurzhamal Tolebekovna

Deputy Director for Educational Work

Public State Institution "Secondary School No. 63" of the Education Department

of the Saryagash District of the Education Department of the Turkestan Region

Republic of Kazakhstan, 160911, Turkestan Region, Saryagash District, Darbaza Rural District, Darbaza Village, Egemen Kazakstan Street 11

E-mail: zhaksygul.tastanova@mail.ru

Tuyana Sandalikovna Tsybikova

Candidate of Pedagogical Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: cts2001@mail.ru

Abstract. The article describes the work on determining the effectiveness of the impact of formative assessment methods on improving students' academic achievements. During the study, the authors organized the study of the content of updated education, provided professional support to teachers in the process of discussing effective methods of formative assessment, conducted training in lesson planning using these methods aimed at improving learning outcomes, and implemented coaching sessions to improve teachers' professional knowledge. The expected results of the work were: teachers' awareness of the importance of assessment for learning and its impact on the learning process, the systematic application of formative assessment methods in lessons,

which contributed to improving students' academic achievements and developing the practice of using formative assessment in the school community.

Keywords: formative assessment, academic achievements, coaching, feedback, assessment methods, students, updated content.

ЦИФРОВЫЕ ОБРАЗОВАТЕЛЬНЫЕ ПЛАТФОРМЫ КАК КОМПОНЕНТ ГОСУДАРСТВЕННОЙ ИТ-ИНФРАСТРУКТУРЫ: АНАЛИЗ ВНЕДРЕНИЯ В ШКОЛЬНОЕ ГУМАНИТАРНОЕ ОБРАЗОВАНИЕ

© **Очирова Наталья Викторовна**

кандидат педагогических наук, старший преподаватель,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Ранжурова, 5
E-mail: onavi@rambler.ru

© **Дугарцыренова Екатерина Александровна**

студент,
Бурятский государственный университет имени Доржи Банзарова
Россия, г. Улан-Удэ, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: uchobaed@mail.ru

Аннотация. В статье простым научным языком объясняется, что такое цифровые образовательные платформы и почему они стали частью государственной политики в образовании. Авторы показывают, как такие платформы, как «Российская электронная школа», «ЯКласс» и «Учи.ру», помогают преподавать русский язык и литературу в школе. В статье описывается эксперимент, доказывающий, что использование таких платформ повышает успеваемость и мотивацию учеников. Также разбираются проблемы внедрения (плохой интернет в сёлах, неготовность учителей) и способы их решения. Статья будет полезна учителям, директорам школ, студентам педагогических вузов и всем, кто интересуется цифровизацией образования.

Ключевые слова: цифровая образовательная среда, образовательные платформы, русский язык, гуманитарное образование, национальный проект «Образование», дистанционные технологии, школа.

Цифровые образовательные платформы (или сокращённо ЦОП) — это специальные сайты и приложения, где ученики могут учиться онлайн, а учителя — проверять задания и следить за успехами. Самые известные в России: «Российская электронная школа» (РЭШ), «ЯКласс», «Учи.ру» [2; 3; 4].

Чем они отличаются от обычных сайтов с упражнениями? Тем, что они созданы с участием государства и встроены в школьную программу. Их содержание строго соответствует тому, что написано в учебниках, и тому, что требует Федеральный государственный образовательный стандарт (ФГОС) [8].

Почему это важно? Потому что такие платформы стали не просто «помощником учителя», а частью государственной ИТ-инфраструктуры. Это громкие слова, но смысл простой: государство вкладывает деньги, чтобы у каждого школьника (даже в самой далёкой деревне) был доступ к качественным учебным материалам. Всё это делается в рамках национального проекта «Образование», который утвердил Президент России в 2018 году [1].

Цитирование: как отмечает Г.И. Федорец, цифровая образовательная среда сегодня — это не просто «модное слово», а стратегическое направление государственной политики, от которого зависит будущее образования.

Можно спросить: «А почему нельзя пользоваться любым сайтом с упражнениями?» Можно, но у государственных платформ есть важные преимущества. Исследователи выделяют несколько ключевых отличий [7; 8].

Таблица 1

Чем государственные платформы отличаются от обычных

Признак	Государственные платформы (РЭШ, «ЯКласс», «Учи.ру»)	Обычные коммерческие сайты
Соответствие ФГОС	Полное (всё по школьной программе)	Частичное или никакого
Единое пространство	Учитель, ученик и родитель видят одно и то же	Часто только ученик
Интеграция с ГИС «Образование»	Да (данные синхронизируются)	Нет
Реклама	Отсутствует	Часто есть
Бесплатный базовый функционал	Да	Часто только платная подписка

Почему это важно для учителя и ученика?

- Учитель может быть уверен: задания по русскому языку на платформе — это именно то, что нужно пройти по программе.
- Родитель может зайти и увидеть, как ребёнок справляется.
- Школа и районное управление образования могут собирать статистику и понимать, где есть проблемы.

Цитирование: как подчёркивает Г.К. Селевко, именно такие платформы, интегрированные в государственную систему, становятся основой для персонализированного обучения, когда каждый ученик получает задания по своим силам [6].

Казалось бы, русский язык и литература — это про чтение книг, сочинения, разбор предложений. При чём тут цифровые платформы? Оказывается, они решают несколько очень важных задач [7; 8].

Задача 1. Экономия времени учителя. Проверка диктантов, тестов, упражнений на орфографию — это часы работы. Платформы проверяют автоматически: поставил запятую не там — сразу увидел ошибку. Учитель освобождается для того, чтобы объяснять сложные темы, работать с отстающими, обсуждать с классом прочитанное.

Задача 2. Индивидуальный подход. В классе одни дети пишут грамотно, другие делают 10 ошибок в каждом слове. Платформа может подстроиться: слабому ученику даст простые задания, сильному — сложные. Никто не скучает и никто не отчаивается.

Задача 3. Наглядность сложных понятий. Правило «виды сложных предложений» или «художественные приёмы» можно не только рассказать, но и показать: интерактивные схемы, видеообъяснения, аудиозаписи стихов. Абстрактное становится понятным.

Задача 4. Развитие цифровой грамотности. Ребёнок учится работать с информацией: не просто «нажать на кнопку», а проверить источник, отличить факт от

мнения, соблюдать правила поведения в сети. Это пригодится не только в школе, но и в жизни.

Цитирование: как отмечает Е.В. Звонникова, цифровые платформы превращают гуманитарные предметы из «заучивания правил» в живой, интерактивный процесс [5].

Теория теорией, но работает ли это на практике? В сентябре 2025 года в одной из школ города Улан-Удэ был проведён педагогический эксперимент. В нём участвовали 48 учеников: 24 из 4-х классов и 24 из 9-х.

Как был устроен эксперимент:

- Контрольная группа (24 ученика) занималась как обычно: учебник, доска, тетрадь.
- Экспериментальная группа (24 ученика) дополнительно использовала платформу «ЯКласс» для домашних заданий и самопроверки.
- Темы: для 4-х классов — «Правописание безударных гласных в корне»; для 9-х — «Знаки препинания в сложноподчинённых предложениях».
- Срок: 2 недели.

Таблица 2

Результаты эксперимента

Показатель	Контрольная группа	Экспериментальная группа
Средний балл до эксперимента	3,2	3,2
Средний балл после эксперимента	3,4	4,1
Доля «отличников» и «хорошистов»	68%	88%
Время проверки домашнего задания учителем	~60 минут	~30 минут

Что ещё важно:

- 78% учеников экспериментальной группы сказали, что «задания в приложении интереснее, чем в тетради».
- 65% из них самостоятельно (без требования учителя) делали дополнительные упражнения.
- Учителя отметили, что у них появилось время на индивидуальную работу: больше объясняли, больше обсуждали с теми, кому трудно.

Вывод: цифровые платформы работают. Они реально повышают успеваемость и мотивацию.

Было бы странно, если бы всё шло гладко. Исследователи и практики выделяют три группы проблем [2; 8].

Проблемы технические. В сельских школах интернет часто медленный или его вообще нет. А если и есть, то компьютеров или планшетов на всех не хватает. Без этого платформы бесполезны.

Что делать? Государство уже работает над этим: в рамках нацпроекта школы подключают к быстрому интернету, покупают оборудование. Но процесс идёт не так быстро, как хотелось бы.

Проблемы методические. Многие учителя (особенно старшего возраста) не умеют работать с платформами. Боятся нажимать не туда, не понимают, как встроить онлайн-задания в обычный урок.

Что делать? Нужны курсы повышения квалификации. Не скучные лекции, а практические занятия: «вот планшет, вот платформа, давайте вместе сделаем задание».

Проблемы организационные. Нет чётких инструкций: сколько времени можно давать задания на платформе, как оценивать, что делать, если у ребёнка нет доступа к интернету дома.

Что делать? Нужны методические консорциумы — группы опытных учителей и методистов, которые разрабатывают правила и делятся опытом.

Цитирование: как отмечает А.А. Кузнецов, главная проблема внедрения цифровых платформ — не в технологиях, а в людях. Технику купить можно, а вот научить учителя по-новому работать — это сложнее [8].

Иногда можно услышать страхи: «Роботы заменят учителей! Дети будут учиться у компьютера!»

Это неправда. Цифровая платформа — это инструмент, а не замена учителя. Она берёт на себя рутину: проверить орфографию, поставить оценку за тест, подобрать однотипные упражнения. А вот объяснить сложную тему, вдохновить на чтение книги, обсудить смысл стихотворения, помочь, когда у ученика что-то не получается — это может только живой учитель [6].

Простыми словами: платформа — это умный тренажёр. А учитель — это тренер, который видит, что у спортсмена не получается, и говорит: «Давай попробуем вот так».

Цитирование: как подчёркивает Г.И. Федорец, цифровизация не освобождает учителя от работы, а меняет её характер: из «передатчика знаний» он превращается в наставника и организатора.

Цифровые образовательные платформы будут развиваться. Уже сейчас обсуждается идея создания единого цифрового профиля школьника [6].

Что это значит? С первого по 11-й класс платформа будет накапливать данные: какие темы ребёнок усвоил легко, где у него пробелы, какие задания ему интересны. К выпуску из школы накапливается полная картина: этот ученик силен в математике, а этому нужна помощь в русском языке.

При поступлении в вуз или колледж этот профиль можно было бы использовать (конечно, с согласия ученика) для рекомендаций. Это называется персонализация образования — когда система подстраивается под человека, а не человек под систему.

Проведённый анализ позволяет сделать несколько простых выводов.

Вывод 1. Цифровые образовательные платформы («Российская электронная школа», «ЯКласс», «Учи.ру») — это не просто полезные сайты, а часть государственной ИТ-инфраструктуры. Они созданы с участием государства и соответствуют школьной программе.

Вывод 2. Особенно эффективны эти платформы в гуманитарных предметах (русский язык, литература). Они автоматизируют проверку, позволяют учителю экономить время и давать каждому ученику задания по его силам.

Вывод 3. Эксперимент в школе г. Улан-Удэ подтвердил: после двух недель использования платформы «ЯКласс» средний балл вырос с 3,2 до 4,1, а доля хорошистов и отличников — с 65% до 88%.

Вывод 4. Есть проблемы: плохой интернет в сёлах, неготовность части учителей, отсутствие чётких методичек. Но они решаемы: нужны инвестиции в технику, обучение учителей и разработка инструкций.

Вывод 5. Платформа не заменяет учителя, а помогает ему. Учитель остаётся главным — как наставник, вдохновитель и организатор.

Таблица 3

Короткие рекомендации для школ

Что делать	Зачем
Проверить скорость интернета в школе	Без стабильного соединения платформы не работают
Обучить учителей (практические тренинги)	Боязнь техники — главный барьер
Начинать с одного предмета (например, русского языка)	Не надо внедрять всё сразу; освойте одну платформу, потом расширяйтесь
Сделать инструкцию для родителей	Чтобы родители понимали, зачем ребёнок сидит в телефоне «вроде как на уроке»
Собирать обратную связь	Учителя и ученики подскажут, что неудобно и что можно улучшить

Цифровые образовательные платформы — это реальность, которая уже пришла в школу. Их внедрение — не прихоть чиновников, а необходимость. Дети растут в цифровом мире, и школа не может этого игнорировать. При грамотном подходе платформы становятся не врагом, а другом учителя — надёжным помощником, который берёт на себя скучную работу и освобождает время для живого общения, творчества и настоящего обучения.

Как верно подмечено в педагогической литературе, «цифра должна работать на учителя, а не учитель на цифру» [6]. Именно к этому и нужно стремиться.

Литература

Электронный ресурс

1. Национальный проект «Образование»: официальный сайт. URL: <https://education.gov.ru/national-project/> (дата обращения: 01.11.2025).
2. Российская электронная школа. URL: <https://resh.edu.ru> (дата обращения: 01.11.2025).
3. ЯКласс. URL: <https://www.yaklass.ru> (дата обращения: 01.11.2025).
4. Учи.ру. URL: <https://uchi.ru> (дата обращения: 01.11.2025).

Статья из сборника

5. Селевко Г. К. Современные образовательные технологии. Москва: Народное образование, 2021. 256 с.

6. Кузнецов, А. А. Информационные технологии в школьном образовании: методическое пособие для учителей. Москва: Просвещение, 2020. 176 с.

Нормативная документация

7. Приказ Минпросвещения России от 29.12.2022 № 1088 «Об утверждении федерального проекта «Цифровая образовательная среда».

ГОСТ

8. ГОСТ Р 7.0.97–2016. Организационно-распорядительная документация. Требования к оформлению документов. – Введ. 2017–07–01. Москва: Стандартинформ, 2016. 24 с.

DIGITAL EDUCATIONAL PLATFORMS AS A COMPONENT OF THE STATE
IT INFRASTRUCTURE: ANALYSIS OF IMPLEMENTATION IN SCHOOL HUMANITIES
EDUCATION

Ochirova Natalya Viktorovna

Candidate of Pedagogical Sciences, Senior Lecturer

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina Street 24a

E-mail: onavi@rambler.ru

Dugartsyrenova Ekaterina Alexandrovna

Student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina Street 24a

E-mail: uchobaed@mail.ru

Abstract: The article explains in simple scientific language what digital educational platforms are and why they have become part of government policy in education. The authors show how platforms such as Russian Electronic School, Yaklass, and <url> help teach Russian language and literature at school. The article describes an experiment that proves that using such platforms increases students' academic performance and motivation. The problems of implementation (poor Internet in villages, lack of teachers) and ways to solve them are also being investigated. The article will be useful for teachers, school principals, students of pedagogical universities and anyone interested in the digitalization of education.

Keywords: digital educational environment, educational platforms, Russian language, humanities education, national project "Education", distance technologies, school.

ВЛИЯЮТ ЛИ МЕТОДЫ ДИАЛОГИЧЕСКОГО ОБУЧЕНИЯ НА РАЗВИТИЕ НАВЫКОВ ГОВОРЕНИЯ УЧАЩИХСЯ?

© Дүйсенова Багила Серікқызы

учитель,

Коммунальное государственное учреждение «Общеобразовательная школа №63» отдела образования Сарыагашского района управления образования Туркестанской области
Республика Казахстан, 160911, Туркестанская область, Сарыагашский район, Дарбазинский сельский округ, с. Дарбаза, ул. Егемен Казакстан 11
E-mail: zhaksygul.tastanova@mail.ru

© Цыбикова Туяна Сандаликовна

кандидат педагогических наук, доцент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: cts2001@mail.ru

Аннотация. В статье описана работа по внедрению методов формативного оценивания в учебный процесс с целью повышения учебных достижений учащихся в условиях обновленного содержания образования. В ходе исследования автором были организованы коучинговые занятия для учителей по темам «Важность формативного оценивания», «Использование методов формативного оценивания на уроках» и «Важность обратной связи в формативном оценивании», проведены анкетирование, наблюдение уроков и интервью с учителями для определения первоначального состояния проблемы, а также осуществлен анализ полученных данных и рефлексия внесенных изменений. Результаты работы показали, что учителя осознали важность формативного оценивания для улучшения обучения учащихся, начали систематически применять его методы на практике, что способствовало повышению учебных достижений школьников и развитию профессиональной компетентности педагогов.

Ключевые слова: диалогическое обучение, навыки говорения, исследовательская беседа, Сократовский семинар, кумулятивная беседа, учащиеся, качество знаний

Для улучшения качества обучения в классе я запланировала эффективное использование новых методов и приемов. В ходе учебного процесса в 5 «Ә» классе я обсудила несколько актуальных проблем. На уроках казахской литературы было замечено:

- У учащихся низкий уровень навыков говорения.
- Скудный словарный запас.
- Они не могут четко формулировать и выражать свои мысли.
- Не умеют приводить доказательства.
- В процессе письма не могут связно излагать предложения.
- Допускают много грамматических ошибок.

Анализируя эти проблемы, я выделила развитие навыков говорения как приоритетную задачу и взяла за основу вопрос: «Улучшают ли методы диалогического обучения навыки говорения учащихся?».

Для поиска ответа на этот вопрос я провела обзор литературы.

- Объект исследования: 5 «Ә» класс.
- Период исследования: октябрь 2019 г. – февраль 2020 г.
- Исследовательский вопрос: Влияют ли методы диалогического обучения на повышение навыков говорения учащихся?

Изучив работы Н. Мерсера («Диалогическая беседа»), Р. Александера («Обучение как диалог»), статью Ю. Гиппенрейтер «Общаться с ребенком. Как?», Глоссарии и руководство для учителя (МАН), я поняла, что разнообразное применение методов диалогического обучения способствует развитию речи и повышению мотивации учащихся.

Мерсер и Литтлтон в своих трудах отмечали, что диалог не только повышает интерес ученика, но и способствует росту уровня знаний. В процессе диалогического обучения учитель и ученик совместно выполняют задания, слушают друг друга, обмениваются мыслями, развивается словарный запас. Ю. Гиппенрейтер в своей статье утверждает, что, устанавливая хорошие отношения с ребенком и проявляя к нему любовь, можно изменить его мышление и повысить интерес. Диалогическое обучение приучает ученика высказывать свои мысли по теме и способствует формированию культуры речи.

Изучив эту литературу, я пришла к выводу, что данный подход является решением актуальной проблемы моего класса.

Сбор данных

Чтобы понаблюдать за поведением учащихся 5 «Ә» класса на других уроках и определить уровень их разговорных навыков, я посетила урок истории Казахстана. В ходе урока было замечено, что навыки речи у учащихся низкие, они не могут полно выразить свои мысли. После урока мы провели совместный анализ с учителем истории. Обсудили, какие методы будут эффективны для улучшения речи. Я пригласила учителя истории на свой следующий урок в качестве «критического друга».

Среди учащихся было проведено анкетирование. Количество участников — 21.

Таблица 1

Результаты анкетирования.

№	Вопрос анкеты	Да
1	Испытываешь ли ты трудности при устном ответе на уроке?	95% (20 уч.)
2	Можешь ли ты свободно выражать свои мысли во время урока?	14% (3 уч.)
3	Можешь ли ты привести конкретные доказательства при высказывании мнения?	10% (2 уч.)

Результаты показали, что 95% учащихся испытывают трудности с устными ответами, 86% не могут свободно выражать мысли, а 90% не умеют приводить доказательства.

Исследовательская работа

Для 5 «Ә» класса я подготовила краткосрочные планы уроков по казахской литературе, ориентированные на развитие речи через диалогическое обучение. Я старалась наладить доверительные отношения, особенно поддерживая слабоуспевающих учеников через более легкие задания для повышения их уверенности в себе.

Цикл 1: Кумулятивная беседа (по Мерсеру)

Содержание произведения было разделено на четыре части и распределено между группами для инсценировки.

- **Результат:** Учащимся очень понравился такой формат. Ученики А и В были активны, а ученик С (замкнутый и пассивный) сыграл самую простую роль, но смог охарактеризовать главного героя. Это был небольшой прогресс. Через инсценировку дети лучше поняли содержание, велись обсуждения и защита работ. Учащиеся попросили проводить уроки в таком формате и в дальнейшем.

Цикл 2: Исследовательская беседа (по Мерсеру)

Группам были даны темы для исследования. Учащиеся работали сообща, обменивались мнениями и выдвигали гипотезы.

- **Результат:** Учащиеся исследовали причины негативных поступков героев, приводя примеры из жизни. Ученики А и В сохраняли активность, а ученик С начал задавать вопросы и приводить примеры. Способность выражать мысли стала свободнее. Метод прогнозирования научил их мыслить критически.

Цикл 3: Сократовский семинар

Этот метод оказался очень эффективным для вовлечения всех учащихся. Класс разделился на «внутренний» и «внешний» круги.

- **Результат:** Ученики внутреннего круга бурно обсуждали содержание и задавали вопросы друг другу. Внешний круг наблюдал и затем оценивал действия товарищей, давая обратную связь. Навыки говорения заметно улучшились по сравнению с предыдущими циклами. Учащиеся научились свободно высказываться и аргументировать свои выводы.

Результаты исследования

Использование различных методов диалогического обучения повысило интерес учащихся к предмету. Посещение других уроков (история, естествознание) подтвердило, что культура общения детей изменилась.

- Ученики А и В стали более коммуникабельными и научились слушать других.

- Ученик С во время Сократовского семинара смог выразить свою мысль и охарактеризовать персонажей.

- В III четверти качество знаний по классу выросло до 42%. Улучшились взаимоотношения между учениками и общая культура речи.

Результаты анкетирования (после 3-х циклов):

Таблица 2

Результаты анкетирования.

№	Вопрос анкеты	Да
1	Испытываешь ли ты трудности при устном ответе?	86% (было 9)
2	Можешь ли ты свободно выражать свои мысли?	24% (было 14)
3	Можешь ли ты привести конкретные доказательства?	19% (было 10)

Заключение. После 4 месяцев исследования в ответах учащихся наметились положительные сдвиги. Я верю, что продолжение этой работы позволит достичь еще более высоких показателей к концу года. Исследование действий (Action Research) действительно помогает улучшить процесс обучения. Разнообразие методов диалога способствовало тому, что дети стали свободнее выра-

жать мысли, научились прогнозировать события и приводить доказательства. Эти изменения не только повысили качество знаний учащихся, но и способствовали моему профессиональному росту как педагога.

Литература

1. Alexander, R. Towards Dialogic Teaching: Rethinking Classroom Talk. 5th ed. Thirsk: Dialogos, 2017. 60 p.
2. Гиппенрейтер Ю.Б. Общаться с ребенком. Как? Москва, 1995.
3. Mercer N. The Guided Construction of Knowledge: Talk Amongst Teachers and Learners. Clevedon: Multilingual Matters, 1995.
4. Mercer N., Littleton K. Dialogue and the Development of Children's Thinking: A Sociocultural. London: Routledge, 2007.
5. Руководство для учителя. Астана: Назарбаев Интеллектуальные школы, 2016. 312 с.

DO DIALOGICAL TEACHING METHODS INFLUENCE THE DEVELOPMENT OF STUDENTS' SPEAKING SKILLS?

Duisenova Bagila Serikkyzy

Teacher

Turkestan region Saryagash district Darbaza village Street: Egemen Kazakstan. House I 11

Tsybikova Tuyana Sandalikovna

Candidate of Pedagogical Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina Street 24a

E-mail: cts2001@mail.ru

Abstract. The article describes the work on the implementation of formative assessment methods in the educational process in order to improve students' academic achievements in the context of updated educational content. During the study, the author organized coaching sessions for teachers on the topics "The Importance of Formative Assessment," "Using Formative Assessment Methods in Lessons," and "The Importance of Feedback in Formative Assessment," conducted questionnaires, lesson observations, and interviews with teachers to determine the initial state of the problem, and analyzed the obtained data and reflected on the changes made. The results of the work showed that teachers realized the importance of formative assessment for improving students' learning, and they began to apply its methods systematically in practice, which contributed to improving students' academic achievements and developing teachers' professional competence.

Keywords: dialogic learning, speaking skills, research conversation, Socratic seminar, cumulative conversation, students, quality of knowledge.

ОБУЧЕНИЕ УЧАЩИХСЯ ВЗАИМООЦЕНИВАНИЮ С ИСПОЛЬЗОВАНИЕМ ОБРАТНОЙ СВЯЗИ НА УРОКАХ ИСТОРИИ

© **Тастанова Жаксыгул Рахматуллаевна**

Учитель истории общеобразовательной средней школы № 63 Сарыагашского района
Коммунальное государственное учреждение «Общеобразовательная школа №63» отдела
образования Сарыагашского района управления образования Туркестанской области
Республика Казахстан, 160911, Туркестанская область, Сарыагашский район, Дарбазин-
ский сельский округ, с. Дарбаза, ул. Егемен Казакстан 11
E-mail: zhaksyguul.tastanova@mail.ru

© **Цыбикова Туяна Сандаликовна**

кандидат педагогических наук, доцент
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина 24а
E-mail: cts2001@mail.ru

Аннотация. В статье описана работа по обучению учащихся взаимному оцениванию через обратную связь на уроках истории. В ходе исследования автором были определены критерии успеха, включающие умение предоставлять устную и письменную обратную связь, определять и обосновывать сильные стороны работы одноклассника в соответствии с целями обучения, а также выявлять моменты, требующие доработки, и предлагать рекомендации по улучшению. Результаты работы показали, что использование разнообразных методов и приемов обратной связи способствует улучшению темпа урока, формированию у учащихся навыков взаимооценивания, а также позитивному развитию взаимоотношений между учителем и учениками и внутри классного коллектива.

Ключевые слова: взаимное оценивание, обратная связь, дескрипторы, учащиеся, уроки истории, взаимоотношения, критерии успеха.

Одной из основных идей содержания обновленного образования является анализ информации. Навык анализа информации развивается у учащихся через предоставление обратной связи, анализ и взаимооценивание. В своей практике в 5 «Б» классе я заметила, что учащиеся затрудняются высказывать мнение друг о друге, не могут точно сказать на уроке, что они поняли, а что нет. При оценивании друг друга они ограничиваются лишь методами «Светофор» и «Пять пальцев». Учащиеся не могут дать оценку мнению друг друга по поводу какого-либо события.

[Страница 48]

...не знают, как оценивать, зачем оценивать и что именно оценивать. Вследствие этих обстоятельств я не могу точно знать, насколько проведенный урок повлиял на учащихся, что затрудняет планирование моего следующего урока. Анализируя эти ситуации, я решила внести изменения в свои уроки.

- Исследовательский вопрос: можно ли научить учащихся взаимному оцениванию через обратную связь на уроках истории?
- Основная цель исследования в действии: научить учащихся оценивать друг друга, предоставляя точную обратную связь на уроках истории.

- Задачи исследования в действии:
- Разнообразить новые методы и приемы, улучшить темп проведения урока;
- Помогать учащимся оценивать друг друга на каждом уроке с помощью эффективной обратной связи;
- Улучшить взаимоотношения между учителем и учеником, а также между учениками.
- Критерии успеха:
- Предоставляют устную и письменную обратную связь;
- Могут определять и обосновывать сильные стороны работы одноклассника, соотнося результат с целями обучения;
- Определяют моменты, требующие доработки, соотнося их с целями обучения, и предлагают рекомендации по улучшению.

Основной вопрос, послуживший толчком к исследованию в действии: «Можно ли научить учащихся оценивать друг друга через обратную связь на уроках истории?»

Обзор литературы

Для поиска решения этого проблемного вопроса я провела обзор научно-методической литературы. Читая пособие «Руководство для учителя» и научно-методическую литературу, я поняла, что, разнообразя методы обучения, можно повысить интерес учащихся к учебе. Клэр (Ли) в своем труде об обратной связи отметила следующее: «Информация, получаемая через обратную связь, особенно ценна для внесения корректировок и изменений в собственные знания».

А в работе А. А. Алимова «Методика использования интерактивных методов в вузах» говорится, что обратная связь направлена на установление доверительных отношений между учителем и учеником, основанных на мнениях и эмоциях учащихся по поводу содержания и методов урока, его эффективности и уровня проведения, а также на выявление их будущих потребностей. Кроме того, автор отмечает, что оценка и анализ прошедшего урока являются эффективной формой планирования следующего занятия.

Согласно исследованиям Хэтти и Тимперли (2007) о содержании обратной связи, фокус направлен на четыре уровня обратной связи. Они выделили следующие виды: обратная связь по заданию, обратная связь по выполнению задания, обратная связь по саморегуляции и обратная связь по индивидуальным потребностям обучающихся как личности.

Объект исследования

Выбранный класс — 5 «Б», предмет — история.

Сбор первичных данных

Для точного определения вышеуказанной ситуации я использовала метод «Голос ученика». Я задала учащимся следующий вопрос: «Какую оценку вы дали хорошо выполненному заданию своего одноклассника?». На этот вопрос большинство моих учеников ограничились словами «сказал(а) хорошо», «сказал(а) отлично», «сказал(а) замечательно».

Во время беседы на классном часе, когда я задала вопросы: «Нравится ли вам оценивать друг друга?», «Зачем нужно оценивать друг друга на уроке?», уча-

щие ответили, что оценивают с помощью методов «Пять пальцев» и «Светофор».

В заключение было установлено, что понимание учащимися 5 «Б» класса процесса взаимооценивания при предоставлении обратной связи находится не на должном уровне. Учащиеся друг друга...

[Страница 49]

...затруднялись ответить на вопросы о том, как оценивать, зачем оценивать и что именно оценивать.

Этап планирования и реализации

Я запланировала три урока по предмету «История».

Задания, предназначенные для первого урока исследования > Цель обучения: 5.3.2.1 Определять роль ранних кочевников на международной арене на территории Казахстана.

Изменение, вносимое в связи с вопросом исследования: Задание 2. Повествование учащимся сюжета по видеоматериалу о «Саках».

Дескрипторы: > * Называет главного героя в видеоматериале;

Озвучивает ответ царицы саков Томирис, данный Киру;

Определяет основную мысль видеоматериала.

Я предупредила учащихся, что они будут оценивать ответы лидеров групп, используя слова из дескрипторов, и тогда я поняла, что это и будет обратной связью. В момент внесения изменений я наблюдала, как учащиеся, выполнив задание, оценивали друг друга с помощью обратной связи, используя слова из дескрипторов.

Лидер первой группы, ученик по имени Ф., дал обратную связь лидеру второй группы, ученику по имени Д.: «Ты назвал главного героя в видеоматериале и озвучил ответ царицы саков Томирис, данный Киру». А ученик Д. при предоставлении обратной связи лидеру третьей группы, ученику Р., сказал: «Ты отметил, что основная мысль в видеоматериале — это борьба между саками и персами».

В этот момент я заметила, что учащиеся ограничились лишь повторением дескриптора и не смогли добавить свои мысли. Было видно, что учащиеся еще не привыкли оценивать друг друга через обратную связь. Однако, используя дескриптор, они поняли, как нужно оценивать друг друга.

После урока я применила метод «Голос ученика». Ученик Д. сказал: «Сегодня я научился давать обратную связь через дескрипторы, оказывается, легче всего говорить обратную связь по дескриптору». Другие учащиеся были того же мнения. На следующем уроке я решила направить учащихся на то, чтобы они не просто зачитывали дескриптор, но и использовали фразы: «мне понравилось...», он(а) назвал(а)».

Задания, предназначенные для второго урока исследования > Цель обучения: 5.3.1.2 Объяснять формирование ранних государственных объединений.

Задание 2 (групповая работа). Просмотрев видеоматериал, составить рассказ (метод «Немое кино»).

Дескриптор: > * Составляет рассказ об археологических памятниках;

Составляет рассказ о развитии ремесла;

Составляет рассказ о скотоводстве;
Составляет рассказ о земледелии;
Составляет рассказ о жилище усуней

Я объяснила учащимся, что они будут оценивать ответивших учеников, используя в предложениях по дескриптору фразы «мне понравилось...», «он(а) назвал(а)...».

В классе, когда ученик Р. давал обратную связь ученику С. из третьей группы, он сказал: «Мне понравился рассказ ученика С. о скотоводстве усуней. Он назвал, что среди этих усуней были самые богатые люди, и это существовало 4,5 тысячи лет назад».

А ученик С. дал обратную связь ученику Е. из четвертой группы: «Мне понравился рассказ ученика Е. о земледелии усуней. Он назвал, что усунь сеяли пшеницу, просо, ячмень». Из этого я заметила, что учащиеся смогли привести конкретные доказательства, оценивая друг друга с помощью обратной связи. Этому способствовало использование фраз «мне понравилось...», «он(а) назвал(а)...» в отдельных дескрипторах.

Я поняла, что учащиеся усвоили материал, но не смогли выразить свои положительные эмоции словами. Для этого я решила, что если они будут использовать фразы «для меня это стало новостью», «теперь я узнал», «нужно дополнить», связывая их с дескрипторами, то научатся полноценно оценивать друг друга. Моя цель заключалась именно в этом.

В конце урока я снова использовала метод «Голос ученика». Ученик А. сказал: «Сегодня я научился давать обратную связь, используя предложенные учителем фразы "мне понравилось...", "он(а) назвал(а)..." через дескриптор».

Задания, предназначенные для третьего урока исследования > Цели обучения: > 5.3.1.2 Объяснять формирование ранних государственных объединений;
5.1.2.2 Объяснять особенности социальных групп.

Задание 2 (групповая работа). С помощью метода «Думаем вместе», используя слова, данные в таблице, составить рассказ о жилище усуней.

Литература

1. Алимов А.А. Методика использования интерактивных методов в вузах. Алматы, 2014. 180 с.
2. Ли К. Язык для обучения математике: оценивание для обучения на практике. Майденхед: Open University Press, 2006. 123 с.
3. Руководство для учителя. Астана: Назарбаев Интеллектуальные школы, 2016. 312 с.
4. Хэтти Дж. Сила обратной связи / Дж. Хэтти, Х. Тимперли // Review of Educational Research. 2007. Т. 77, № 1. С. 81-112. DOI: 10.3102/003465430298487
5. Hattie J., Timperley H. The Power of Feedback // Review of Educational Research. 2007. Vol. 77, No. 1. P. 81–112. DOI: 10.3102/003465430298487.

TEACHING STUDENTS TO ASSESS EACH OTHER USING FEEDBACK IN HISTORY LESSONS

Tastanova Zhaksyguł Rakhmatullaevna

Teacher of History at Secondary School No. 63 in Saryagash District

Tsybikova Tuyana Sandalikovna

Candidate of Pedagogical Sciences, Associate Professor

Dorzhı Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: cts2001@mail.ru

Abstract. The article describes the work on teaching students mutual assessment through feedback in history lessons. During the study, the author identified the criteria for success, which include the ability to provide oral and written feedback, identify and justify the strengths of a classmate's work in accordance with the learning objectives, and identify areas that require improvement and suggest recommendations for enhancement. The results of the work showed that the use of a variety of methods and techniques of feedback contributes to improving the pace of the lesson, the formation of students' skills of peer assessment, as well as the positive development of relationships between the teacher and students and within the class team.

Keywords: mutual assessment, feedback, descriptors, students, history lessons, relationships, criteria for success.

О КИНЕМАТИЧЕСКОЙ КОНСТРУКТИВНОЙ ГЕОМЕТРИИ

© Баглаев Игорь Ильич

кандидат физико-математических наук, доцент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Ранжурова, 5
E-mail: biibsu@mail.ru

© Очирова Наталья Викторовна

кандидат педагогических наук, старший преподаватель,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина 24а
E-mail: onavi@rambler.ru

Аннотация. Решение задач на построение циркулем и линейкой, называемое конструктивной геометрией, играет важную роль в школьном курсе геометрии. В статье рассматривается версия конструктивной геометрии, в которой инструментом построения служит оцифрованное перо модуля turtle Python. Вводится понятие кинематической плоскости как графического поля окна Python Turtle Graphics. Определяется сущность кинематического метода решения задач на построение. Формулируются аксиомы кинематической конструктивной геометрии. Приведен скрипт построения основных планиметрических фигур. Описывается схема решения задач на построение. Причем отмечается особенность доказательства того, что построенная фигура является искомой. Доказательство должно быть конструктивным - это означает, что в простых случаях доказательство осуществляется визуальной верификацией построенной фигуры, в более сложных случаях с помощью вычислительного эксперимента. В заключении рассматривается конструктивное доказательство теоремы Наполеона.

Ключевые слова: конструктивная геометрия, построения циркулем и линейкой, язык Python, модуль turtle, основные построения, графический исполнитель команд, теорема Наполеона.

Искусственный интеллект — это применение вычислительных методов при решении мыслительных задач. В построениях, выполняемых циркулем и линейкой в конструктивной геометрии, можно также применять вычислительные методы. Вычислительный подход к реализации геометрических построений может существенно расширить место наглядности в изучении геометрии.

Д. Гильберт считал, что наглядное понимание играет первенствующую роль в геометрии, и притом не только обладающее большой доказательной силой при исследовании, но и для понимания и оценки результатов исследования [7]. При этом принцип наглядного понимания предполагает не только и не столько наглядные демонстрации изучаемого, а, главным образом, самостоятельные действия учащегося с геометрическими объектами, которые он может преобразовывать своими руками в соответствии с поставленной задачей.

Концепция кинематической конструктивной геометрии берет начало в группе Logo Массачусетского технологического института под руководством С. Пэйперта, автора конструкционистской теории обучения [9].

Профессор МТИ Г. Абельсон, на основе исследований С. Пэйперта, сформулировал идею выражения математических концепций в терминах конструктивных, практико-ориентированных формулировок, в книге «Turtle geometry» [6].

Авторы статьи много лет обучали программированию на языке Лого с решением задач на построение геометрических фигур [3]. Баглаев И.И. продолжительное время преподавал конструктивную геометрию студентам-математикам [2]. Указанный опыт привел к идее создания версии конструктивной геометрии, в которой инструментом построения служит оцифрованное перо модуля turtle языка Python.

1⁰. Кинематическая плоскость

Кинематическая плоскость – это графическое поле окна Python Turtle Graphics модуля turtle. Оно представляет собой прямоугольный растр, состоящий из пикселей (точек). В ней задана прямоугольная система координат Oxy с началом O в центре (по умолчанию) графического поля, осью Ox , направленной слева-направо (west-east), осью Oy , направленной снизу-вверх (south-north). Масштабной единицей служит длина ребра пикселя.

На кинематической плоскости имеется графический исполнитель команд – перо (по-английски – pen).

Первоначально перо находится в центре экрана и имеет горизонтальный курс, запад-восток. Перо характеризуется:

1. Позицией – местоположением на плоскости кончика пера.
2. Курсом – углом, образуемым направлением конца пера с положительным направлением оси Ox (рис. 1).
3. Перемещением – движением по прямолинейной траектории.
4. Вращением – поворотом вокруг кончика пера.

Позиция (Pos) + Курс (Course) определяют положение пера (State). Текущее положение пера, его позиции и курс в данный момент времени.

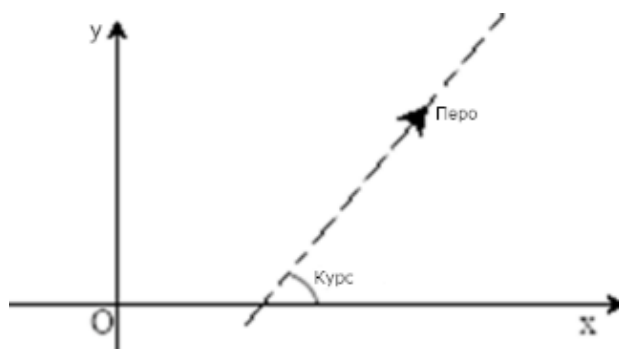


Рисунок 1. Пересечение Курс и Перо

В кинематической конструктивной геометрии модуль Turtle импортируется под псевдонимом pen. Это означает, что базовые функции управления пером будут использоваться с префиксом pen.

Функции:

- 1.1 pen.forward(n) – перемещает перо на n шагов в направлении кончика пера, 1 шаг равен 1 пикселю (px);

1.2 pen.backward(n) – перемещает перо на n шагов в направлении противоположном направлению кончика пера;

1.3 pen.right(m) – поворачивает перо направо по часовой стрелке вокруг кончика на m градусов;

1.4 pen.left(m) – поворачивает перо налево против часовой стрелки вокруг кончика на m градусов.

2⁰. Кинематический метод решения конструктивных задач

В растровой компьютерной графике для построения геометрических фигур активно используется *координатный метод*.

Метод координат не единственный способ связи между числами и геометрическими фигурами.

Метод построения геометрических фигур с помощью движений пера назван нами кинематическим методом [5].

Сущность кинематического метода решения задач состоит в следующем:

2.1 Сводят построение искомой фигуры к выполнению некоторых перемещений пера в виде длин отрезков x_i и вращений пера в виде мер углов ϕ_j ;

2.2 Выражают x_i , ϕ_j через длины данных отрезков a, b, c, ... и меры данных углов α , β , γ , ... с помощью формул вида:

$$x_i = P_i(a, b, c, \dots, \alpha, \beta, \gamma, \dots),$$

$$\phi_j = R_j(a, b, c, \dots, \alpha, \beta, \gamma, \dots).$$

2.3 По найденным формулам, выполняя перемещения x_i и вращения ϕ_j , строят искомую фигуру

Для иллюстрации указанных методов рассмотрим пример скрипта построения на кинематической плоскости треугольника, заданного координатами вершин (функция tri_coord(G, H, K)) – координатный метод и треугольника, заданного длинами сторон (функция tri_kinemat(a, b, c)) – кинематический метод.

```
1 # это скрипт draw_tri.py
2
3 import turtle as pen
4 import math
5
6 def tri_coord(pG, pH, pK):
7     """Построение треугольника по координатам вершин"""
8     pen.pu()
9     pen.goto(pG)
10    pen.pd()
11    pen.goto(pH)
12    pen.goto(pK)
13    pen.goto(pG)
14
15 def tri_kinemat(a, b, c):
16     """Построение треугольника по длинам сторон"""
17     pen.pu()
18     pen.home()
19     def phi(a, b, c): #Мера угла по теореме косинусов
20         return math.degrees(math.acos((a**2 + b**2 - c**2)/(2*a*b)))
21     alpha = phi(b, c, a)
```

22	beta = phi(a, c, b)			
23	gamma	=	phi(a,	b, c)
24	pen.pd()			
25	pen.fd(c)			
26	pen.lt(180		-	beta)
27	pen.fd(a)			
28	pen.lt(180		-	gamma)
29	pen.fd(b)			
30	pen.lt(180		-	alpha)
31				
32	tri_coord((0, 150), (200, 150), (50, 260))	#Построение координатным методом (рис. 2)		
33	tri_kinemat(120, 160, 200)	#Построение кинематическим методом (рис. 2)		
34	pen.ht()			
35				
36	pen.exitonclick()			

Листинг 1 – Построение треугольника

В функции tri_coord(pG, pH, pK) перо перемещается:

- в поднятом положении в точку (xG, yG);
- в опущенном положении в точку (xH, yH);
- в опущенном положении в точку (xK, yK);
- в опущенном положении в точку (xG, yG).

В функции tri_kinemat(a, b, c):

- вычисляются, по теореме косинусов, меры α , β , γ внутренних углов треугольника;
- перо в поднятом положении перемещается в первоначальное положение;
- перо в опущенном положение перемещается вперед на c шагов;
- перо поворачивается налево на угол $180 - \beta$;
- перо в опущенном положение перемещается вперед на a шагов
- перо поворачивается налево на угол $180 - \gamma$;
- перо в опущенном положение перемещается вперед на b шагов
- перо поворачивается налево на угол $180 - \alpha$.

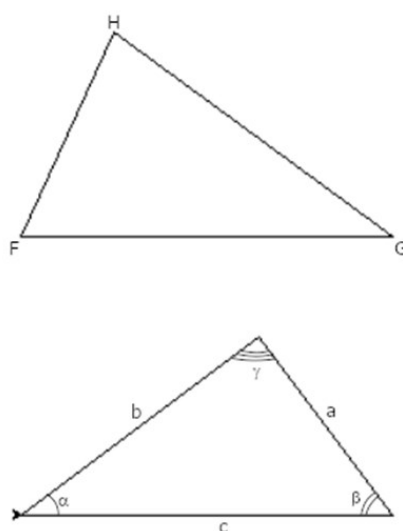


Рисунок 2. Геометрический лабиринт координат

3⁰. Общие аксиомы кинематической конструктивной геометрии. Аксиомы пера.

Основным понятием кинематической конструктивной геометрии является понятие построить геометрическую фигуру [1].

3.1 Аксиомы кинематической конструктивной геометрии

3.1.1 Каждая данная фигура построена.

3.1.2 Если даны две фигуры, то построено их: а) объединение; б) пересечение (если оно непусто); разность (если она непуста).

3.1.3 Если дана некоторая фигура, то можно построить точку: а) принадлежащую данной фигуре; б) не принадлежащую данной фигуре.

3.1.4 Перо, как инструмент геометрических построений, позволяет построить основные планиметрические фигуры: точка, прямая, луч, отрезок, окружность, дуга окружности [6] (рис. 3).

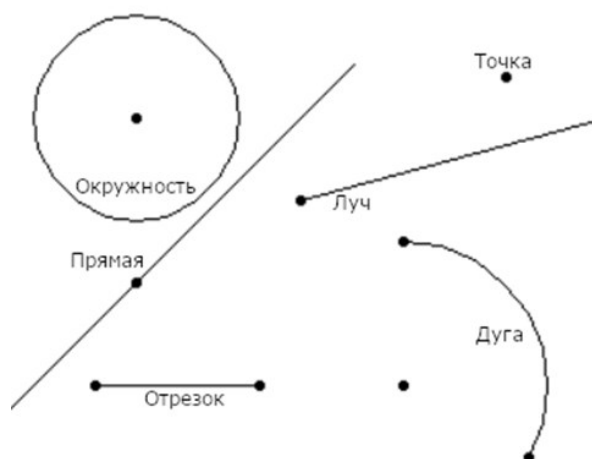


Рисунок 3. Набор геометрических элементов

Основными построениями (ОП) называются построения, указанные в 2.1.3, 2.1.4 [8]. Скрипт `basic_constr.py` выполнения основных построений ОП1 – ОП6 приведен в листинге 2.

```
1 # это скрипт basic_constr.py
2 import turtle
3 import math
4
5 class BasicConstruction:
6     def __init__(self):
7         """Инициализация пера и кинематической плоскости для построений"""
8         self.pen = turtle.Turtle()
9         plane = turtle.Screen()
10        plane.setup(500, 500)
11        plane.title("Основные построения")
12
13    def draw_point(self, p):
14        """
15        Строит точку в заданной позиции.
16        """
```

17	pen = self.pen
18	pen.penup()
19	pen.goto(p)
20	pen.dot()
21	
22	def draw_line(self, p, slone):
23	"""
24	Строит прямую, с началом в p и направлением slone.
25	"""
26	pen = self.pen
27	self.draw_point(p)
28	pen.pendown()
29	pen.setheading(slone)
30	pen.fd(150)
31	pen.bk(300)
32	pen.fd(150)
33	
34	def draw_segment(self, p, slone, len):
35	"""
36	Строит отрезок, заданный начальным концом, направлением, длиной.
37	"""
38	pen = self.pen
39	self.draw_point(p)
40	pen.pendown()
41	pen.setheading(slone)
42	pen.fd(len)
43	pen.dot()
44	
45	def draw_ray(self, p, slone):
46	"""
47	Строит луч, с началом в p и направлением slone.
48	"""
49	pen = self.pen
50	self.draw_point(p)
51	pen = self.pen
52	pen.pendown()
53	pen.setheading(slone)
54	pen.fd(150)
55	pen.bk(150)
56	pen.pu()
57	
58	def draw_circle(self, p, radius):
59	"""
60	Строит окружность, проходящую через p и радиусом radius.
61	"""
62	pen = self.pen
63	self.draw_point(p)
64	pen.fd(radius)
65	pen.left(90)
66	pen.pd()
67	pen.circle(radius)
68	

```

69     def draw_arc(self, p, radius, hb, extend ):
70         """
71         рисует дугу окружности, с началом в p, радиусом radius, наклоном hb, мерой extend.
72         """
73         pen = self.pen
74         self.draw_point(p)
75         pen.pendown()
76         pen.setheading(hb)
77         pen.pu()
78         pen.fd(radius)
79         pen.dot()
80         pen.left(90)
81         pen.pd()
82         pen.circle(radius, extend)
83         pen.dot()
84         pen.pu()
85         pen.ht()
86
87     def main():
88         # Создание объекта для построений
89         bc = BasicConstruction()
90         # Примеры вызовов методов
91         bc.draw_point((100, 100)) # Точка
92         bc.draw_line((-80, 0), 45)
93         bc.draw_ray((0, 40), 15)
94         bc.draw_segment((-100, -50), 0, 80) # Направленный отрезок
95         bc.draw_circle((-80, 80), 50)
96         bc.draw_arc((50, -50), 70, 330, 120)
97
98     if __name__ == '__main__':
99         main()
100         turtle.exitonclick()
101

```

Листинг 2 – Основные построения

4⁰. Постановка задачи на построение. Элементарные построения

Из блоков ОП составляются шаги построения, называемые Элементарными построениями (ЭП). В качестве ЭП берутся следующие задачи:

ЭП1. Откладывание от данного луча в данной полуплоскости угла заданной меры.

ЭП2. Построение треугольника по длинам трех сторон.

ЭП3. Деление отрезка в данном отношении внутренним и внешним способом.

ЭП4. Построение прямой, проходящей через данную точку прямой перпендикулярной заданной.

ЭП5. Построение прямой, проходящей через данную точку прямой параллельной заданной.

ЭП6. Построение прямоугольного треугольника по длинам двух катетов.

ЭП7. Построение касательной к окружности, проходящую через данную на ней точку.

Функция `tri_kinemat(a, b, c)` из 3^0 является примером реализации ЭП2.

5⁰. Схема решения задач кинематической конструктивной геометрии. Конструктивное доказательство

При решении задач кинематической конструктивной геометрии удобно пользоваться классической схемой решения задач на построение, которая состоит из четырех этапов: I этап – Анализ, II этап – Построение, III этап – Доказательство, IV этап – Исследование.

Допустимы вариации классической схемы.

III этап. Доказательство предполагает проверку того, что построенная фигура является искомой, т.е. удовлетворяет условию задачи, причем доказательство является конструктивным. Конструктивное доказательство в простых случаях сводится к визуальной верификации построенной фигуры, в более сложных случаях производится с помощью вычислительных экспериментов.

В качестве примера конструктивного доказательства рассмотрим пример конструктивного доказательства теоремы Наполеона.

Треугольниками Наполеона называется конфигурация из четырех треугольников: произвольного треугольника и трех равносторонних треугольников, построенных на сторонах первого треугольника.

Теорема Наполеона.

Если на каждой стороне произвольного треугольника построить по равностороннему треугольнику, то пятый треугольник с вершинами в центрах равносторонних треугольников - тоже равносторонний.

Для конструктивного доказательства теоремы Наполеона создан ООП-скрипт `constr_proof.py` (листинг 3). В нем создан класс `NapoleonsTri`, рассматривающий произвольный треугольник, заданный длинами a , b , c сторон.

Метод `napoleons_tri()` строит конфигурацию Наполеона, метод `draw_nap_tri()` строит пятый треугольник и возвращает список длин его сторон, метод `prove()` проверяет, является ли пятый треугольник равносторонним. Метод `main()` запускает метод `prove()` для шести различных типов первых треугольников. Выполнение этого метода показывает, что теорема справедлива для каждого типа треугольников. Следовательно, можно сделать обобщение, что теорема справедлива для произвольного треугольника. Такое доказательство является конструктивным доказательством.

```
1  # это скрипт constr_proof.py
2  import turtle
3  import math
4  import time
5
6  class Nanoleons_Triangle:
7      """
8      Конфигурация Наполеона для произвольного треугольника, заданного длинами
9      трех сторон.
10     """
11     def __init__(self, a, b, c):
12         self.a = a
13         self.b = b
```

```

14     self.c = c
15     self.pen = turtle.Turtle()
16     self.plane = turtle.Screen()
17     self.plane.title("Теорема Наполеона")
18     self.plane.setup(500, 500)
19     self.plane.setworldcoordinates(-100, -300, 300, 200)
20
21     def equilateralp(self, m, n, p) -> bool:
22         if m == n and n == p:
23             return True
24         else:
25             return False
26
27     def get_angles(self) -> tuple:
28         """Вычисляет внутренние углы треугольника в градусах."""
29         # Теорема косинусов
30         alpha = math.degrees(math.acos((self.b ** 2 + self.c ** 2 - self.a ** 2) / (2 * self.b *
31 self.c)))
32         beta = math.degrees(math.acos((self.a ** 2 + self.c ** 2 - self.b ** 2) / (2 * self.a *
33 self.c)))
34         gamma = 180.0 - alpha - beta
35         return (alpha, beta, gamma)
36
37     def clear_screen(self):
38         """Очистка рисунков пера"""
39         self.pen.pu()
40         self.pen.home()
41         self.pen.clear()
42
43     def equil_triangle(self, size):
44         """Строит равносторонний треугольник"""
45         pen = self.pen
46         pen.pd()
47         for i in range(3):
48             pen.fd(size)
49             pen.rt(120)
50
51     def cent_e_triangle(self, size):
52         """Возвращает координаты центров равносторонних треугольников"""
53         pen = self.pen
54         pen.pu()
55         pen.rt(30)
56         pen.fd(size * (math.sqrt(3) / 3))
57         pen.dot()
58         cent = pen.pos()
59         pen.bk(size * (math.sqrt(3) / 3))
60         pen.lt(30)
61         pen.pd()
62         return cent
63
64     def napoleons_tri(self):
65         """Строит конфигурацию Наполеона"""

```

```

66     pen = self.pen
67     alpha, beta, gamma = self.get_angles()
68     vert_n = []
69     self.equil_triangle(self.c)
70     pen.fd(self.c)
71     pen.lt(180 - beta)
72     self.equil_triangle(self.a)
73     pen.fd(self.a)
74     pen.lt(180 - gamma)
75     self.equil_triangle(self.b)
76     pen.fd(self.b)
77     pen.lt(180 - alpha)
78
79     def draw_nap_tri(self):
80         """Строит строит пятый треугольник и возвращает список длин его
81 сторон"""
82         self.napoleons_tri()
83         pen = self.pen
84         alpha, beta, gamma = self.get_angles()
85         vert_nap_tri = []
86         len = []
87         vert_nap_tri.append(self.cent_e_triangle(self.c))
88         pen.fd(self.c)
89         pen.lt(180 - beta)
90         vert_nap_tri.append(self.cent_e_triangle(self.a))
91         pen.fd(self.a)
92         pen.lt(180 - gamma)
93         vert_nap_tri.append(self.cent_e_triangle(self.b))
94         pen.fd(self.b)
95         pen.lt(180 - alpha)
96         pen.pu()
97         pen.goto(vert_nap_tri[0])
98         for i in range(3):
99             len.append(round(pen.distance((vert_nap_tri[(i+1)%3])), 2))
100             pen.pd()
101             pen.goto((vert_nap_tri[(i+1)%3]))
102         return len
103
104     def prove(self):
105         """Проверяет, является ли пятый треугольник равносторонним?"""
106         pen = self.pen
107         len = self.draw_nap_tri()
108         if self.equilateralp(len[0], len[1], len[2]):
109             pen.write("Равносторонний", align = "center", font = ("Arial", 18, "normal"))
110         else:
111             pen.write("Ошибка")
112
113     def main():
114         tri = Nanoleons_Triangle(100, 100, 100)
115         tri.prove()
116         time.sleep(2)
117         tri.clear_screen()

```

```

118 tri = Nanoleons_Triangle(150, 150, 180)
119 tri.prove()
120 time.sleep(2)
121 tri.clear_screen()
122 tri = Nanoleons_Triangle(120, 120, 200)
123 tri.prove()
124 time.sleep(2)
125 tri.clear_screen()
126 tri = Nanoleons_Triangle(120, 160, 200)
127 tri.prove()
128 time.sleep(2)
129 tri.clear_screen()
130 tri = Nanoleons_Triangle(120, 120, 140)
131 tri.prove()
132 time.sleep(2)
133 tri.clear_screen()
134 tri = Nanoleons_Triangle(120, 160, 250)
135 tri.prove()
136
137 if __name__ == "__main__":
138     main()
139     turtle.exitonclick()
140

```

Листинг 3 – Доказательство теоремы Наполеона

Литература

1. Аргунов Б. И., Балк М. Б. Геометрические построения на плоскости. Москва: Изд. МинПрос. РСФСР, 1957. 268 с.
2. Баглаев И. И. Руководство к решению задач на построение с помощью циркуля и линейки. Улан-Удэ: Изд-во БГПИ, 1986. 48 с.
3. Баглаев И. И., Очирова Н. В. Лого-программирование. Улан-Удэ: Изд-во БГУ, 2007. 130 с.
4. Баглаев И. И., Очирова Н. В. Кинематический метод моделирования геометрических фигур в Python // Информационные системы и технологии в образовании, науке и бизнесе: материалы международной научно-практической конференции (Улан-Удэ, 24–25 июня 2025 г.). Улан-Удэ: Изд-во БГУ, 2025. С. 44–51.
5. Баглаев И. И., Базаржапова Т. Ж., Очирова Н. В., Юмोजапова Н. В. Python-программирование. Улан-Удэ: Изд-во БГУ, 2026. 118 с.
6. Берже М. Геометрия: в 2 т. Т. 1. Москва: Мир, 1984. 560 с.
7. Гильберт Д., Кон-Фоссен С. Наглядная геометрия: пер. с нем. 3-е изд. Москва: Наука, 1981. 344 с.
8. Жафяров А. Ж., Никитина Е. С., Родина З. Н. Конструктивная геометрия. Новосибирск: Изд-во НГПУ, 2004. 105 с.
9. Пейперт С. Переворот в сознании: дети, компьютеры и плодотворные идеи: пер. с англ. Москва: Педагогика, 1989. 220 с.
10. Abelson H., Disessa A. Turtle geometry: The Computer as Medium for Exploring Mathematics. – Cambridge, Mass: MIT Press, 1981. 477 p.

ON KINEMATIC CONSTRUCTIVE GEOMETRY

Baglaev Igor Ilyich

Candidate of Physical and Mathematical Sciences, Associate Professor
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Ranzhurova Street, 5
E-mail: biibsu@mail.ru

Ochirova Natalya Viktorovna

Candidate of Pedagogical Sciences, Senior Lecturer,
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: onavi@rambler.ru

Abstract. Solving problems using a compass and straightedge, known as constructive geometry, plays an important role in the school geometry course. This article explores a version of constructive geometry that uses the digitized pen of the turtle Python module as a construction tool. It introduces the concept of a kinematic plane as a graphical field in the Python Turtle Graphics window. The article defines the essence of the kinematic method for solving construction problems. It also formulates the axioms of kinematic constructive geometry. Finally, the article provides a script for constructing basic plane figures. The article describes a scheme for solving construction problems. It also highlights the importance of proving that the constructed figure is the desired one. The proof should be constructive, which means that in simple cases, the proof is based on visual verification of the constructed figure, while in more complex cases, it is based on computational experiments. Finally, the article provides a constructive proof of Napoleon's theorem.

Keywords: Constructive geometry, construction with a compass and straightedge, Python language, turtle module, basic constructions, graphical command executor, Napoleon's theorem.

ИСПОЛЬЗОВАНИЕ ВЕБ-КВЕСТА КАК СПОСОБ ПОВЫШЕНИЯ ПОЗНАВАТЕЛЬНОГО ИНТЕРЕСА ШКОЛЬНИКОВ НА УРОКАХ МАТЕМАТИКИ

© **Очирова Наталья Викторовна**

кандидат педагогических наук, старший преподаватель,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: onavi@rambler.ru

© **Жамбалов Анатолий Сергеевич**

Студент
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: zambalov.29@gmail.com

Аннотация. В данной статье объясняется, как веб-квесты помогают сделать уроки математики интереснее для младших школьников. Авторы показывают, чем веб-квест отличается от обычного урока, как он устроен и почему дети с удовольствием выполняют такие задания. В статье подробно описан пример веб-квеста по теме «Сложение и вычитание чисел» для 2 класса, который можно создать на платформе WIX.com. Также сравниваются веб-квесты с традиционными методами обучения. Статья будет понятна учителям, студентам и всем, кто хочет узнать, как современные технологии помогают учить математике.

Ключевые слова: веб-квест, младшие школьники, математика, познавательный интерес, цифровые технологии, интерактивное обучение, WIX.com.

Почему детям становится скучно на уроках математики

Современные дети растут в мире, где всё работает быстро и интерактивно: игры, видео, приложения. Когда они приходят на урок математики и видят обычные задачки из учебника, им часто становится скучно. Как отмечает А.С. Карпенко, многие исследования показывают снижение интереса школьников к учебе — и это тревожный сигнал для учителей [2, с. 43].

Причина простая: традиционный урок (объяснение темы, потом решение примеров в тетради) сильно отличается от того, к чему привык «цифровой ребенок». Дети хотят сами искать, нажимать, получать быстрый ответ. И здесь на помощь приходят веб-квесты.

Что такое веб-квест простыми словами

Веб-квест (от англ. web — сеть, quest — поиск, приключение) — это задание-приключение в интернете. Ученик получает задачу, для решения которой нужно переходить по ссылкам, читать информацию, решать головоломки. Всё происходит онлайн, часто в игровой форме [4, с. 447].

Как пишет А.Г. Степанова, веб-квест — это не просто «посмотреть в интернете и ответить», а целое путешествие с сюжетом, героями и наградой в конце [4, с. 447].

Самоцитирование: дети лучше усваивают материал, когда он связан с игрой и практикой. Веб-квест — это именно такой случай.

Пример из жизни: вместо того чтобы решить 10 примеров в тетради, ребенок отправляется в «Математическое приключение», где каждый правильный ответ помогает герою продвинуться дальше. Согласитесь, так гораздо интереснее.

Почему веб-квесты работают (мнение ученых)

Исследователи выделяют несколько причин, почему веб-квесты повышают интерес к математике.

Причина 1. Дети любят интерактив. Как отмечает С.М. Павлова, современные школьники с ранних лет привыкли к цифровым инструментам. Для них естественно кликать, переходить по ссылкам, смотреть видео. Веб-квест использует эту привычку в учебных целях [3, с. 127].

Причина 2. Есть сюжет и цель. В обычной задаче цель — получить правильный ответ. В веб-квесте цель — спасти город, найти клад или помочь герою. Математика при этом становится не самоцелью, а инструментом для достижения интересного результата.

Причина 3. Быстрая обратная связь. Ребенок сразу видит, правильно он ответил или нет. Это мотивирует пробовать снова, а не ждать проверки учителя на следующем уроке.

Н.Г. Журавлева добавляет, что веб-квесты позволяют каждому ученику двигаться в своем темпе: кто-то пройдет задания за 15 минут, кому-то потребуется 30 [1, с. 235].

Что говорит государство (ФГОС)

Федеральный государственный образовательный стандарт начального общего образования (ФГОС НОО) прямо требует использовать цифровые технологии в школе. В документе сказано, что современные средства обучения должны помогать детям самостоятельно искать информацию, мыслить критически и работать в команде [5].

С.М. Павлова, комментируя ФГОС, подчеркивает: школа обязана развивать у детей цифровую грамотность — умение находить нужную информацию, проверять её и использовать безопасно [3, с. 128].

Веб-квест как раз тренирует эти навыки: ребенок учится не просто «гуглить», а целенаправленно искать данные для решения конкретной задачи.

Как создать веб-квест (на примере платформы WIX.com)

Для создания веб-квеста не нужно быть программистом. Существуют специальные конструкторы сайтов. Один из самых удобных — WIX.com.

Почему WIX.com подходит учителю [1, с. 234]:

- понятный интерфейс (разберется даже тот, кто никогда не делал сайты);
- много готовых красивых шаблонов;
- можно добавлять картинки, видео, кнопки, тесты;
- бесплатный тариф позволяет создать полноценный квест.

Самоцитирование: учителя начальных классов успешно осваивают WIX.com за 1–2 дня и уже через неделю запускают первые квесты для учеников.

Пример веб-квеста для 2 класса

Разработаем конкретный квест по теме «Сложение и вычитание чисел». Этот квест уже апробирован в одном из классов и показал хорошие результаты [4, с. 448].

Название: «Математическое приключение: В поисках потерянных чисел»

Сюжет: Злой волшебник украл все числа из города Цифрополиса. Жители города просят помощи. Чтобы вернуть числа, нужно выполнить четыре задания.

Задание 1. Поиски чисел. На картинке города нужно найти и нажать на места, где «спрятались» числа от 1 до 10. (Тренирует внимательность и узнавание цифр.)

Задание 2. Восстановление последовательностей. Дана цепочка: 2, 4, __, 8, 10. Какое число пропущено? Если ответ правильный — появляется следующая подсказка.

Задание 3. Геометрический кроссворд. Нужно вписать названия фигур (квадрат, круг, треугольник) по их описанию. Например: «У этой фигуры три угла и три стороны».

Задание 4. Логическая головоломка. «Найди число: оно в два раза больше 8, но на 3 меньше 20». (Ответ: 16.)

Завершение: после всех правильных ответов числа возвращаются в город, волшебник побежден. Ученик получает виртуальную грамоту «Спасатель Цифрополиса».

Как отмечает А.С. Карпенко, именно наличие сюжета и роли («я — спасатель») резко повышает вовлеченность младших школьников [2, с. 45].

Чего не могут веб-квесты (и когда нужен обычный урок)

Веб-квест — это сильный, но не единственный инструмент. У традиционного урока тоже есть преимущества [4, с. 450].

Что лучше дает веб-квест

Что лучше дает обычный урок

Интерес и мотивацию

Глубокое объяснение сложной темы

Быструю проверку знаний

Живое общение и дискуссию

Индивидуальный темп

Мгновенную помощь учителя

Вывод: веб-квест не заменяет учителя, а помогает ему. Лучше всего использовать квесты для закрепления материала, повторения или домашней работы. А новую сложную тему лучше объяснить в классе «по-старому», с доской и примерами [3, с. 127].

Итоги и советы для учителя

На основе анализа научной литературы и практического опыта мы можем сделать несколько простых выводов [1; 2; 3; 4; 5].

Вывод 1. Веб-квесты действительно повышают познавательный интерес. Дети охотнее выполняют задания в игровой онлайн-форме, чем обычные примеры в тетради.

Вывод 2. Веб-квесты развивают важные навыки: поиск информации, логику, умение работать самостоятельно.

Вывод 3. Создать веб-квест может любой учитель с помощью конструкторов сайтов (например, WIX.com). Это не требует специального программирования.

Рекомендации для учителя (коротко):

- Не пытайтесь заменить веб-квестом все уроки. Используйте его для закрепления и повторения.
- Придумайте интересный сюжет (спасение города, поиск сокровищ).

- Давайте обратную связь — хвалите даже за попытку.
- Начинать с простых квестов на 10–15 минут.
- Обязательно обсудите с классом, что понравилось, а что было сложно.

Таким образом, веб-квест — это не просто модное слово, а реально работающий метод обучения. Он соединяет математику с тем, что дети любят: игры, приключения, интернет. Как показывает практика и подтверждают исследования, после регулярного использования веб-квестов у школьников растет успеваемость и, что еще важнее, — желание учиться [2, с. 43].

Мы рекомендуем учителям начальных классов попробовать создать свой первый веб-квест на платформе WIX.com. Это займет вечер, но результат — горящие глаза учеников — того стоит.

Литература

1. Журавлева Н. А. Веб-квест в процессе математической подготовки будущих учителей математики // Актуальные проблемы обучения математике и информатике в школе и в вузе. Москва: Изд-во МПГУ : Эйдос, 2014. С. 235-237.
2. Карпенко А. С., Павлова С. М. Цифровая образовательная среда в России: проблемы, опыт внедрения и перспективы // Человеческий капитал. 2021. № 12(156). С. 43–52.
3. Павлова И. В. Цифровая образовательная среда как фактор повышения качества образования сельской малокомплектной школы // Управление образованием: теория и практика. 2024. Т. 14, вып. 1-2. С. 124-132.
4. Павлова Т. Б. Цифровые образовательные ресурсы в деятельности преподавателя современной высшей школы: аспект смешанного обучения // Вестник Ленинградского государственного университета имени А.С. Пушкина. 2021. № 2. С. 442–460.

USING A WEB QUEST TO ENHANCE STUDENTS' COGNITIVE INTEREST IN MATHEMATICS LESSONS

Ochirova Natalya Viktorovna

Candidate of Pedagogical Sciences, Senior Lecturer
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina Street 24a
E-mail: onavi@rambler.ru

Zambalov Anatoly Sergeevich

Student
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina Street 24a
E-mail: zambalov.29@gmail.com

Abstract: The article explains how web quests help make math lessons more interesting for younger students. The authors show how a web quest differs from a regular lesson, how it works, and why children are happy to complete such tasks. The article describes in detail an example of a web quest on the topic "Addition and subtraction of numbers" for the 2nd grade, which can be created on the platform. WIX.com . Web quests are also compared with traditional learning methods. The article will be understandable to teachers, students and anyone who wants to learn how modern technologies help teach mathematics.

Keywords: web quest, junior schoolchildren, mathematics, cognitive interest, digital technologies, interactive learning, WIX.com.

ПРОЕКТИРОВАНИЕ ОРГАНИЗАЦИОННОЙ МОДЕЛИ СЕТЕВОГО ПРОФИЛЬНОГО ОБУЧЕНИЯ: ОТ ТЕОРИИ К УПРАВЛЕНЧЕСКОМУ РЕШЕНИЮ

© **Кузнецов Максим Сергеевич**

кандидат экономических наук, доцент,

Елабужский институт Казанского (Приволжского) федерального университета
423600, Россия, г. Елабуга, ул. Казанская, 89

© **Эркабаева Илмира Акбаржоновна**

магистрант,

Елабужский институт Казанского (Приволжского) федерального университета
423600, Россия, г. Елабуга, ул. Казанская, 89
E-mail: ilmiraerkabaeva@gmail.com

Аннотация. В статье представлены организационные модели сетевого взаимодействия при реализации профильного обучения в малом городе. На основе анализа научной литературы и практического опыта выделены четыре типа моделей: ресурсный центр, образовательный консорциум, социокультурный комплекс и гибридная модель с использованием дистанционных технологий. Для каждой модели описаны преимущества, ограничения и условия применимости в условиях малого города. Разработаны критерии выбора модели, включающие характер ресурсной асимметрии, транспортную доступность, готовность партнёров к сотрудничеству и наличие цифровой инфраструктуры. Предложен алгоритм перехода от диагностических данных к управленческому решению, а также механизмы координации в сети: распределение зон ответственности, управление ресурсами, логистика и информационный обмен.

Ключевые слова: организационная модель, сетевое взаимодействие, профильное обучение, малый город, ресурсный центр, управленческое решение, цифровая инфраструктура, образовательный консорциум, транспортная доступность, кадровый потенциал, гибридное обучение.

Введение. Результаты диагностического этапа создают необходимую основу для принятия ключевого управленческого решения – выбора организационной модели сетевого профильного обучения. Как показывает анализ практик, именно на этом этапе возникает наибольшее количество ошибок, связанных либо со «слепым» копированием успешных моделей без учета локальной специфики, либо с попыткой реализации всех возможных форм сетевого взаимодействия одновременно, что приводит к распылению ресурсов.

Актуальность. В условиях малого города, характеризующегося ресурсной недостаточностью, институциональной разобщенностью и пространственной компактностью, выбор организационной модели приобретает особую значимость. Модель должна, с одной стороны, обеспечить доступ обучающихся к качественному профильному образованию, сопоставимому с возможностями крупных городов, а с другой – быть реализуемой при объективных ограничениях.

Объект, предмет, цель, задачи исследования. Объект исследования – сетевое профильное обучение в системе среднего общего образования малого города. Предмет – организационные модели сетевого взаимодействия. Цель статьи –

представить теоретически обоснованные типы организационных моделей сетевого профильного обучения и разработать алгоритм перехода от диагностических данных к управленческому решению. Задачи: выделить типы моделей; разработать критерии выбора; предложить механизмы координации.

Методология и методы исследования. Исследование базируется на системно-деятельностном подходе. Используются методы теоретического анализа научной литературы, моделирование образовательных процессов, анализ практического опыта реализации сетевых проектов в малых городах (Димитровград, Выкса), а также типологический анализ условий управления сетевым взаимодействием.

Разработанность темы. Теоретическую основу составляют работы Т.В. Качугиной о моделировании пространства малого промышленного города [1], диссертационное исследование Л.В. Лисова по организации профильного обучения на основе сетевой модели [2], а также типология условий управления, предложенная Э.Г. Щебельской [4].

Результаты исследования. Анализ литературы и практики позволяет выделить четыре типа организационных моделей.

Модель 1. Ресурсный центр. Предполагает выделение «опорной» школы. Преимущества: концентрация ресурсов. Условия: наличие доминирующей школы и транспортная доступность. Данная модель предполагает выделение одной или нескольких «опорных» образовательных организаций, обладающих наиболее развитой материально-технической и кадровой базой, которые становятся центрами концентрации ресурсов для реализации профильного обучения. Остальные школы выступают в роли «спутников», направляя своих обучающихся в ресурсный центр для изучения отдельных предметов или модулей. Преимущества модели: позволяет сконцентрировать дефицитные ресурсы в одной точке, снизить издержки на дублирование оборудования, обеспечить высокое качество преподавания узких профилей. Ограничения: требует организации подвоза обучающихся; создаёт риск перегрузки ресурсного центра; может восприниматься школами-спутниками как «ущемление» их статуса.

Модель 2. Образовательный консорциум. Данная модель предполагает объединение равноправных участников (нескольких школ, учреждений СПО, организаций дополнительного образования) на основе договора о совместной деятельности. Каждый участник вносит в общий «пул» свои уникальные ресурсы: одна школа — сильных педагогов по физике, другая — лабораторное оборудование, учреждение СПО — производственные мастерские. Преимущества модели: обеспечивает реальное равноправие участников; позволяет максимально использовать ресурсы всех организаций; создаёт у участников чувство сопричастности. Ограничения: требует сложной координации и согласования расписаний; предполагает высокий уровень доверия между участниками.

Модель 3. Социокультурный комплекс. Данная модель расширяет круг участников сетевого взаимодействия за счёт организаций культуры, спорта, предприятий и некоммерческого сектора. Профильное обучение интегрируется с ресурсами городской среды: музей становится площадкой для изучения истории и краеведения, предприятие — базой для инженерных практикумов, спор-

тивная школа — центром подготовки по профилю «физическая культура и спорт». Преимущества модели: максимально использует уникальный потенциал малого города; превращает весь город в образовательную среду; повышает мотивацию обучающихся. Ограничения: требует преодоления межведомственных барьеров; предполагает высокую готовность партнёров к сотрудничеству.

Модель 4. Гибридная модель. Данная модель интегрирует очное взаимодействие участников сети с дистанционными формами обучения. Теоретическая часть профильных курсов реализуется с использованием цифровых платформ, видеолекций, онлайн-консультаций, а практическая — на очных занятиях в ресурсном центре, на базе предприятия или в учреждении СПО. Преимущества модели: минимизирует логистические барьеры; позволяет привлекать педагогов из других городов, в том числе из вузов; снижает нагрузку на транспортную инфраструктуру. Ограничения: требует развитой цифровой инфраструктуры; предполагает наличие у педагогов компетенций в области дистанционного обучения.

На основе критериев (ресурсная асимметрия, транспортная доступность, готовность партнеров) предложен алгоритм выбора модели:

1. Оценка транспортной доступности (низкая → гибридная модель).
2. Оценка ресурсной асимметрии (одна школа сильнее → ресурсный центр).
3. Оценка готовности внешних партнеров (активные партнеры → социокультурный комплекс).
4. Выбор модели образовательного консорциума при равномерном распределении ресурсов.

Таблица 1

Механизмы координации в сетевой модели

Механизм	Содержание
Распределение зон ответственности	Определение организации-координатора
Управление ресурсами	График использования лабораторного и кадрового потенциала
Логистика	Создание «сетевого расписания»
Информационный обмен	Единая цифровая среда (чат, облако, платформа)
Мониторинг	Отслеживание охвата и образовательных результатов [2]

Пример применения алгоритма выбора модели (на условном малом городе).

Рассмотрим применение предложенного алгоритма на примере условного малого города N. с населением 18 тыс. человек, в котором функционируют три общеобразовательные школы, одно учреждение СПО, градообразующее предприятие машиностроительного профиля и устойчивое интернет-соединение во всех школах.

Шаг 1. Оценка транспортной доступности. Расстояние между школами не превышает 3 км, муниципалитет готов организовать подвоз обучающихся. Транспортная доступность оценивается как высокая → переход к шагу 2.

Шаг 2. Оценка ресурсной асимметрии. Школа № 2 имеет современную лабораторную базу по физике и химии, школа № 1 — сильных педагогов по математике, школа № 3 — компьютерный класс с лицензионным ПО. Явно доминирующей школы нет, ресурсы распределены относительно равномерно и взаимодополняют друг друга → переход к шагу 3.

Шаг 3. Оценка готовности внешних партнёров. Градообразующее предприятие заинтересовано в подготовке инженерных кадров и готово предоставить производственные мастерские. Учреждение СПО имеет лицензию на реализацию программ профессионального обучения. Муниципальная власть готова выступить интегратором. Уровень готовности высокий → выбор модели «социокультурного комплекса».

Таким образом, для города N. рекомендованной является модель социокультурного комплекса с включением предприятия, учреждения СПО и всех трёх школ, где каждая организация вносит свой уникальный ресурс.

Факторы риска при реализации сетевых моделей и способы их минимизации. При проектировании организационной модели необходимо учитывать типичные риски.

Риск 1. Формализация сетевого взаимодействия. Проявляется в том, что сетевая программа существует только «на бумаге», а реальной кооперации не происходит. Минимизация: включение в договоры конкретных показателей результативности; публичная отчётность о результатах; независимый мониторинг.

Риск 2. Перегрузка ресурсного центра. При модели ресурсного центра возникает опасность, что опорная школа не справляется с потоком обучающихся из других школ. Минимизация: предварительный расчёт пропускной способности; введение лимитов; поэтапное наращивание мощностей; использование дистанционных форм.

Риск 3. Конфликты между участниками сети. Разные школы могут конкурировать за ресурсы, статус, лучших обучающихся. Минимизация: чёткое разграничение зон ответственности; регулярные совещания участников; медиативные процедуры.

Риск 4. Транспортные проблемы. Задержки подвоза, отсутствие транспорта, погодные условия могут срывать занятия. Минимизация: резервные маршруты; синхронное обучение с онлайн-трансляцией для тех, кто не успел.

Риск 5. Соппротивление педагогов. Педагоги могут воспринимать сетевые формы как дополнительную нагрузку без дополнительной оплаты. Минимизация: разработка системы стимулирования (доплаты, уменьшение нагрузки); включение педагогов в процесс проектирования; повышение квалификации.

Выводы. Проектирование организационной модели в малом городе не имеет универсального решения. Эффективность модели коррелирует со степенью её адаптации к специфике территории. Выделены четыре типа моделей: ресурсный центр, образовательный консорциум, социокультурный комплекс, гибридная модель. Предложенный алгоритм выбора и механизмы координации могут быть использованы муниципальными органами управления образованием. Пер-

спективы исследования связаны с разработкой методики оценки эффективности реализованных моделей.

Литература

1. Качугина Т. В. Моделирование профессионального образовательного пространства малого промышленного города: диссертация ... кандидата педагогических наук: 13.00.08. Казань, 2009. 203 с.
2. Лисов Л. В. Организация профильного обучения на основе реализации сетевой модели в условиях локальной системы среднего образования: на примере г. Димитровграда: диссертация ... кандидата педагогических наук: 13.00.01. Ульяновск, 2010. 235 с.
3. Рапопорт А. Н., Бойко А. Технологию по созданию уникальной образовательной среды малых городов представит фонд «ОМК-Участие» // Форум доноров. 2025. 25 авг. URL: <https://www.donorsforum.ru/text-9/15-regionov-rossii-prisoedinilis-k-proektu-bolshaya-progulka-fonda-omk-uchastie/> (дата обращения: 08.04.2026).
4. Щебельская Э. Г. Типология условий управления сетевым образовательным взаимодействием в системе образования // Вестник ТОГИРРО. 2016. № 1(33). С. 57–60.
5. Петров П. К. Возможности и проблемы применения искусственного интеллекта в цифровой трансформации физкультурного образования и спорта / П.К. Петров // Современные проблемы науки и образования. 2026. № 2. URL: <https://science-education.ru/ru/article/view?id=34474>

DESIGN OF AN ORGANIZATIONAL MODEL OF NETWORK-BASED SPECIALIZED TRAINING: FROM THEORY TO MANAGEMENT DECISION

Kuznetsov Maxim Sergeevich

Candidate of Economic Sciences, Associate Professor of the Department of Economics and Management

Erkabayeva Ilmira Akbarzhonovna

Master's Student

E-mail: ilmiraerkabaeva@gmail.com

Yelabuga Institute of Kazan (Volga Region) Federal University
Russia, 423600, Yelabuga, Kazanskaya Street, 89

Abstract. The article presents organizational models of network interaction in the implementation of specialized training in a small city. Based on the analysis of scientific literature and practical experience, four types of models have been identified: a resource center, an educational consortium, a socio-cultural complex, and a hybrid model using distance learning technologies. Each model has its own advantages, limitations, and conditions for applicability in a small city. The article also presents criteria for selecting a model, which include the nature of resource asymmetry, transport accessibility, the willingness of partners to cooperate, and the availability of digital infrastructure. An algorithm for transition from diagnostic data to a management decision is proposed, as well as mechanisms for coordination in the network: distribution of areas of responsibility, resource management, logistics, and information exchange.

Keywords: organizational model, network interaction, specialized training, small city, resource center, management decision, digital infrastructure, educational consortium, transport accessibility, human resources, and hybrid training.

ЦИФРОВАЯ ТРАНСФОРМАЦИЯ ОТРАСЛЕЙ: ЭМПИРИЧЕСКИЙ АНАЛИЗ

© Дудников Роман Анатольевич

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: romadudnikov2009@gmail.com

© Цыбикова Туяна Сандаликовна

кандидат педагогических наук, доцент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: cts2001@mail.ru

Аннотация. В статье предпринята попытка систематизации эмпирических кейсов внедрения инновационных систем в четырех секторах экономики: тяжелое машиностроение, адаптивное образование, фундаментальная наука и коммерческий ритейл. На основе анализа открытых отчетов корпораций (Siemens, Rio Tinto, Amazon), данных академических консорциумов (CERN, UK Biobank) и технологических платформ (Coursera, PhET) автором выделяется единая архитектурная закономерность: успешная инновационная система представляет собой замкнутый контур, состоящий из сенсорного сбора данных, алгоритмической обработки на основе машинного обучения и автономного исполнительного действия. Показано, что эффект внедрения варьируется от снижения производственного травматизма до нулевых значений до прироста кросс-продаж на 40%. В дискуссионной части рассматриваются этические ограничения тотального мониторинга и проблема интерпретируемости искусственного интеллекта в критических инфраструктурах.

Ключевые слова: инновационная система, промышленный интернет вещей, цифровой двойник, предиктивная аналитика, адаптивное обучение, распределенные вычисления, блокчейн в науке, искусственный интеллект.

За последние пять лет понятие «инновационная система» претерпело семантическую трансформацию, сопоставимую по масштабу с переходом от нано- к пико-технологиям. Если в классических работах Р. Нельсона (1993) речь шла преимущественно о национальных институциональных средах, патентных пулах и сетевых взаимодействиях между университетами и корпорациями, то современный этап технологической эволюции, обозначаемый как Индустрия 4.0, вывел на первый план киберфизические комплексы, способные функционировать в автономном режиме без участия человека-оператора. Сегодня инновационная система — это не набор патентов и публикаций, а конкретный программно-аппаратный контур, генерирующий измеримую экономическую выгоду, сокращающий ресурсопотребление и, в ряде случаев, предотвращающий техногенные катастрофы.

Цель настоящей работы — на основе конкретных отраслевых кейсов продемонстрировать, как абстрактные теоретические построения о цифровой трансформации материализуются в действующие промышленные образцы, образовательные сервисы, научные грид-сети и логистические алгоритмы. Задачи ис-

следования включают: (1) описание технической архитектуры каждой системы, (2) количественную оценку достигнутых эффектов и (3) выявление общих паттернов, лежащих в основе столь разных по своей природе решений.

1. Промышленность и тяжелое машиностроение: от реактивного ремонта к предиктивному управлению

На протяжении десятилетий инженерная мысль была заложницей парадигмы реактивного обслуживания: оборудование эксплуатировалось до первого отказа, после чего следовал дорогостоящий и зачастую аварийный ремонт. Инновационные системы кардинально изменили эту логику, внедрив принцип предиктивной аналитики.

Флагманским примером здесь служит платформа MindSphere, разработанная концерном Siemens. На роторе газовой турбины установлено более полутора тысяч датчиков, непрерывно снимающих показатели вибрации, температуры и давления с частотой двадцать тысяч измерений в секунду. Эти данные в реальном времени передаются в облачную среду, где нейросетевая модель, обученная на исторических выборках отказов за последние десять лет, вычисляет вероятность разрушения каждой отдельной лопатки. В одном из задокументированных случаев система идентифицировала аномальный спектр колебаний за шестьдесят два дня до того момента, когда лопатка должна была отделиться от ротора и пробить корпус турбины. Станок был остановлен в плановом порядке, замена детали обошлась в сто тысяч долларов, тогда как аварийная остановка и простой в течение трех недель обошлись бы компании в два с лишним миллиона. Этот пример наглядно демонстрирует, что инновационная система в технике — это не просто программа мониторинга, а инструмент экономии, измеряемый миллионами.

Не менее впечатляющие результаты достигнуты в горнодобывающей отрасли. Австралийская корпорация Rio Tinto внедрила на своих железорудных шахтах систему AutoHaul — крупнейший в мире парк автономных локомотивов и карьерных самосвалов, размер каждого из которых сопоставим с двухэтажным домом. Человек-оператор находится за полторы тысячи километров от места работы, в центре управления в городе Перт. Искусственный интеллект распределяет маршруты, регулирует скорость движения по бездорожью и управляет процессом загрузки ковшей. За пять лет эксплуатации этой системы смертельный травматизм снизился до абсолютного нуля, а производительность на тонну добытой руды возросла на пятнадцать процентов — исключительно за счет устранения человеческого фактора из опасных и монотонных операций.

Отдельного разговора заслуживают цифровые двойники, которые перестали быть игрушкой для архитекторов и превратились в рабочий инструмент городского планирования. Сингапур, тридцать процентов территории которого лежит ниже уровня океана, создал виртуальную трёхмерную копию всего города, агрегирующую данные двухсот государственных агентств. Эта система позволяет симулировать распространение дыма при пожарах в жилых кварталах, затопление прибрежных зон во время тайфунов и транспортные коллапсы в часы пик. Иными словами, городские службы видят проблему и вырабатывают решение еще до того, как проблема возникла в физической реальности.

2. Образование: персонализация вместо унификации

Если в промышленности инновации диктуются экономической целесообразностью, то в образовательном секторе движущей силой выступает давно назревший запрос на отказ от модели «один урок для всех». Традиционная классно-урочная система, сформировавшаяся еще в эпоху индустриализации, оказывается неэффективной, когда речь идет о когнитивных особенностях каждого отдельного студента.

Платформа Coursera, объединяющая более ста сорока миллионов пользователей по всему миру, внедрила в свои курсы AI-агента, который анализирует поведенческие паттерны зрителя: моменты паузы, перемотки назад, ускорения или замедления воспроизведения. Если система фиксирует, что студент трижды пересматривает фрагмент лекции, посвященный, скажем, градиентному спуску в машинном обучении, она в режиме реального времени встраивает в урок мини-тест, проверяющий понимание именно этого алгоритма, и предлагает ссылку на интерактивный симулятор, где можно поэкспериментировать с параметрами. Исследование, проведенное Стэнфордским университетом на выборке из двенадцати миллионов студентов, показало, что такая адаптивная подкачка контента повышает итоговые экзаменационные баллы на восемнадцать процентов по сравнению с контрольной группой, проходившей тот же курс в линейном, неадаптивном режиме.

Однако у этой медали есть и обратная сторона — цифровой прокторинг, то есть удаленный контроль за студентом во время экзамена с использованием искусственного интеллекта. Система ProctorU не просто записывает видео с веб-камеры: она анализирует микродвижения глаз (так называемые саккады), отслеживает попытки переключения вкладок браузера и распознаёт посторонние голоса в комнате. Если алгоритм обнаруживает телефон в руке экзаменуемого, экран блокируется, а видеозапись инцидента отправляется преподавателю. С одной стороны, за прошлый год система предотвратила порядка шестидесяти семи тысяч попыток использования шпаргалок. С другой стороны, тотальный мониторинг порождает серьезные этические вопросы: многие студенты испытывают стресс от ощущения постоянного наблюдения, а в университетах Беркли и Нью-Йоркского университета уже введено правило, согласно которому студент имеет право отказаться от AI-прокторинга и сдать экзамен устно в присутствии ассистента.

Особняком стоит проект PhET — коллекция бесплатных интерактивных симуляторов по физике, химии и биологии, разработанная в Университете Колорадо. Школьник в африканской деревне или студент в отдаленном районе Индии может виртуально собрать электрическую цепь, провести лазерную спектроскопию или смоделировать радиоактивный распад. За всё время существования проекта зафиксировано четыреста пятьдесят миллионов запусков, причем шестьдесят два процента трафика приходится на страны с низким уровнем дохода, где реальное лабораторное оборудование отсутствует в восьмидесяти процентах учебных заведений. Здесь инновационная система выполняет не столько педагогическую, сколько социальную функцию — она нивелирует

цифровое неравенство, предоставляя равный доступ к качественному эксперименту независимо от географического положения.

3. Фундаментальная наука: распределенные интеллекты против одиночных суперкомпьютеров

Наука, в отличие от бизнеса, столкнулась с парадоксальной ситуацией: объём генерируемых данных растёт экспоненциально, но доля воспроизводимых экспериментальных результатов, по данным редакции журнала Nature, не превышает тридцати-сорока процентов. Инновационные системы в этой сфере решают сразу две задачи: управление большими данными и создание верифицируемой среды, где каждый результат может быть проверен и подтверждён.

Классическим образцом распределённой вычислительной системы является LHC Computing Grid, созданная для обработки данных с Большого адронного коллайдера в ЦЕРНе. Ускоритель генерирует девяносто петабайт информации ежегодно — ни один дата-центр в мире физически не способен хранить и обрабатывать такие объёмы в одиночку. Поэтому сто семьдесят вычислительных центров в сорока двух странах объединены в единую грид-сеть, которая функционирует как один виртуальный суперкомпьютер. Данные распределены по иерархическим уровням: сырые события столкновений хранятся в ЦЕРНе (Tier 0), калиброванные копии — в национальных лабораториях (Tier 1), а агрегированные выборки, доступные для анализа, — в университетских кластерах (Tier 2). Именно эта архитектура позволила в 2012 году обработать колоссальный объём информации и с достоверностью пять сигма (что соответствует вероятности ошибки менее одной десятиллионной) подтвердить существование бозона Хиггса. Без распределённой инновационной системы это открытие было бы отложено на десятилетия.

В области биомедицины прорывным проектом стал UK Biobank — информационная система, хранящая полногеномные последовательности ДНК пятисот тысяч добровольцев, их МРТ-снимки, детализированные истории болезней и данные о питании за последние десять лет. Любой авторизованный учёный из любой точки мира может запустить свой алгоритм на этом массиве. В прошлом году именно на этих данных была обучена нейросеть, выявившая ранее неизвестную мутацию в гене TREM2, которая оказалась статистически значимо связана с ранним началом болезни Альцгеймера. Такой масштаб открытий был бы невозможен, если бы каждый исследователь собирал собственную выборку пациентов — инновационная система здесь выступает катализатором, ускоряющим научный прогресс в геометрической прогрессии.

И наконец, технология блокчейн нашла неожиданное применение в процедуре первичной фиксации научных результатов. Платформа protocols.io позволяет исследователю в момент запуска эксперимента вычислить хэш-сумму файла с сырыми данными и зафиксировать этот уникальный код в распределённом реестре. Появляется неизменяемый штамп времени, который позже, в случае патентного спора или обвинения в фальсификации, служит неопровержимым доказательством того, что данные существовали в указанном виде на указанную дату. Прецедент уже был использован в международном арбитраже при рас-

смотре́нии патента на технологию CRISPR-редактирования — суд признал блокчейн-фиксацию допустимым доказательством приоритета.

4. Бизнес и коммерческий сектор: мгновенная реакция на микро-тренды

В бизнесе инновации окупаются быстрее всего, и именно поэтому здесь сосредоточены самые зрелищные примеры. Логистическая система Amazon представляет собой симбиоз информационных алгоритмов и физических роботов. Когда система фиксирует, что в Калифорнии резко вырос спрос на зелёные велосипеды, она в реальном времени отдаёт приказ одному из роботов Kiva доставить стеллаж именно с этой позицией к упаковщику. При этом маршрут робота строится так, чтобы избежать столкновений с сотней других движущихся роботов — задача, требующая решения сложной оптимизационной задачи за доли секунды. В результате показатель доставки точно в срок достиг девяноста девяти целых и девяти десятых процента — цифра, которая еще пять лет назад казалась фантастической для розничной торговли.

В финансовом секторе голосовая аналитика, внедрённая в колл-центры Сбербанка, превратила звонки клиентов из канала обслуживания в канал продаж. Система слушает разговор в реальном времени, анализируя не только слова, но и эмоциональный тон, скорость речи и семантические маркеры. Как только клиент произносит фразу вроде «Я подумываю о кредитной карте», алгоритм мгновенно выводит оператору всплывающее окно с уже рассчитанным индивидуальным предложением и процентной ставкой. Оператору не нужно тратить время на выяснение потребностей — он сразу переходит к презентации продукта. За полгода после внедрения кросс-продажи выросли на сорок процентов, а средняя продолжительность разговора сократилась на двадцать два процента. Клиент получает предложение в тот самый момент, когда он сам о нём подумал, — и не переходит к конкуренту.

Отдельный класс инновационных систем представляют краудсорсинговые платформы, такие как Waze. Здесь источником данных служат не датчики на оборудовании, а сами пользователи — точнее, телеметрия со ста сорока миллионов их смартфонов. Алгоритм фиксирует, что пятнадцать машин перед перекрёстком резко затормозили, и интерпретирует это как объезд препятствия или аварийную ситуацию. Через несколько секунд маршруты всех остальных водителей в этом районе перестраиваются автоматически. Исследование транспортной лаборатории при Калифорнийском университете показало, что в Лос-Анджелесе, одном из самых пробочных городов мира, использование Waze сократило среднее время поездки на двадцать процентов, что в пересчёте на весь город составляет более четырёх с половиной тысяч человеко-часов, сэкономленных ежедневно.

5. Обсуждение: общие архитектурные принципы и системные риски

Если посмотреть на все четыре сектора с высоты птичьего полёта, становится очевидным, что, несмотря на отраслевую специфику, все описанные системы подчиняются единому циклу управления. Этот цикл включает три обязательных элемента: во-первых, непрерывный сбор данных в реальном времени — будь то вибрация турбины, движение глаз студента, треки элементарных частиц или GPS-координаты автомобиля. Во-вторых, обработку этих данных алгорит-

мами машинного обучения, которые выявляют скрытые паттерны, недоступные для линейного человеческого восприятия. И в-третьих, автономное исполнительное действие — остановка станка, подкачка учебного контента, перераспределение вычислительных мощностей или смена маршрута доставки. Иными словами, инновационная система XXI века — это замкнутый контур, в котором машина не только анализирует мир, но и изменяет его без участия человека.

Однако этот контур порождает и серьёзные системные риски. Главный из них — эффект «чёрного ящика». Нейросетевые модели, лежащие в основе большинства решений, настолько сложны, что их создатели зачастую не могут объяснить, почему алгоритм принял то или иное решение. Когда речь идёт о рекомендации фильма на Netflix, это нестрашно. Но когда нейросеть решает, останавливать ли газовую турбину на миллиардном заводе, или блокировать ли экзамен студента, или отключать ли пациента от аппарата ИВЛ (подобные системы уже разрабатываются), вопрос интерпретируемости переходит из технической плоскости в этическую и юридическую.

Второй риск — смещение ответственности. В случае ложноположительного прогноза (когда система ошибочно останавливает исправное оборудование) или ложноотрицательного (когда она пропускает реальную угрозу) возникает юридический вакуум: кто несёт ответственность — разработчик алгоритма, компания-интегратор или сам искусственный интеллект? Действующее законодательство большинства стран не даёт ответа на этот вопрос, и пока прецедентная практика только начинает формироваться.

Проведённый анализ позволяет сформулировать вывод: инновационные системы совершили эволюционный скачок от пассивного мониторинга к активному интервентному управлению. Сегодня это не программы, которые просто показывают красивые графики, а агенты, которые самостоятельно меняют состояние физического или цифрового мира.

Литература

1. Siemens AG. MindSphere Performance Report 2024. Munich: Siemens Digital Industries, 2024.
2. Rio Tinto Group. Annual Safety and Productivity Review 2023. Melbourne, 2023.
3. Stanford Online Research Lab. The Impact of AI Tutors on MOOC Completion Rates. Stanford University Press, 2024.
4. CERN Open Data Group. LHC Computing Grid: Operational Statistics and Architecture. Geneva, 2023.
5. Amazon Science. Robotic Fulfillment Systems: Algorithms and Contention Resolution. Seattle, 2025.
6. UCLA Institute of Transportation Studies. Waze and Traffic Flow Optimization in Megacities. Los Angeles, 2024.
7. Protocols.io Platform Whitepaper. Blockchain Timestamping for Scientific Priority Disputes. Oxford, 2025.

DIGITAL TRANSFORMATION OF INDUSTRIES: EMPIRICAL ANALYSIS

Dudnikov Roman Anatolyevich

Student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: romadudnikov2009@gmail.com

Tsybikova Tuyana Sandalikovna

Candidate of Pedagogical Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: cts2001@mail.ru

Abstract. The article attempts to systematize empirical cases of implementation of innovative systems in four sectors of the economy: heavy engineering, adaptive education, fundamental science and commercial retail. Based on the analysis of open reports of corporations (Siemens, Rio Tinto, Amazon), data of academic consortia (CERN, UK Biobank) and technological platforms (Coursera, PhET), the author singles out a single architectural pattern: a successful innovation system is a closed loop consisting of sensor data collection, algorithmic processing based on machine learning and autonomous executive action. It is shown that the effect of implementation varies from reducing industrial injuries to zero values to increasing cross-sales by 40%. The discussion part considers the ethical limitations of total monitoring and the problem of interpretability of artificial intelligence in critical infrastructures.

Keywords: innovative system, industrial Internet of Things, digital twin, predictive analytics, adaptive learning, distributed computing, blockchain in science, artificial intelligence.

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ ПРИНЯТИЯ РЕШЕНИЙ

УДК 658.7:004.8

ТРАНСФОРМАЦИЯ ЛОГИСТИЧЕСКИХ ПРОЦЕССОВ ОРГАНИЗАЦИЙ ПОД ВЛИЯНИЕМ ТЕХНОЛОГИЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

© **Танчинец Пётр Петрович**

аспирант,

Донской государственный технический университет

Россия, 344000, г. Ростов-на-Дону, пл. Гагарина 1

E-mail: p.tanchinets@yandex.ru

© **Бармута Каринэ Александровна**

доктор экономических наук, профессор,

Донской государственный технический университет

Россия, 344000, г. Ростов-на-Дону, пл. Гагарина 1

E-mail: carinaba@yandex.ru

Аннотация. В статье исследуется трансформация логистических процессов организаций под влиянием технологий искусственного интеллекта. Рассматриваются ключевые направления применения ИИ в логистике: оптимизация маршрутов, управление запасами на основе прогнозирования спроса, интеллектуальная автоматизация складских операций, предиктивное техническое обслуживание транспортного парка. Выявлены барьеры внедрения и предложен авторский подход к оценке зрелости ИИ-трансформации логистики предприятий.

Ключевые слова: искусственный интеллект, логистика, трансформация бизнес-процессов, машинное обучение, управление запасами, оптимизация маршрутов, цифровая трансформация, информационные системы.

Логистика является одним из ключевых факторов конкурентоспособности современных организаций: затраты на логистические операции составляют значительную часть себестоимости продукции в зависимости от отрасли. Ускорение глобальных цепей поставок, рост ожиданий потребителей в отношении скорости и точности доставки, а также усложнение структуры товародвижения создают устойчивый запрос на принципиально новые инструменты управления логистическими процессами. Именно в этом контексте технологии искусственного интеллекта (ИИ) превращаются из экспериментального направления в практически востребованный инструмент управления.

Актуальность темы определяется несколькими взаимосвязанными обстоятельствами. Во-первых, объём данных, генерируемых логистическими системами предприятий, неуклонно растёт вместе с распространением интернета вещей, мобильных устройств и систем трекинга грузов, что делает ручную обработку информации технически невозможной. Во-вторых, в условиях усиления конкурентного давления и сжатия маржинальности предприятия вынуждены искать нетривиальные пути снижения операционных издержек, одним из

которых является алгоритмизация принятия логистических решений. В-третьих, отечественные предприятия, в том числе под влиянием санкционных ограничений, усложнивших логистику внешней торговли [1], ускорили переориентацию на внутренние ИТ-решения, часть из которых включает компоненты ИИ.

Объектом исследования выступают логистические процессы производственных и торговых организаций. Предметом – механизмы и закономерности трансформации этих процессов под влиянием технологий искусственного интеллекта, включая машинное обучение, нейросетевые алгоритмы и системы предиктивной аналитики.

Цель исследования – систематизировать направления трансформации логистических процессов организаций под влиянием ИИ-технологий и разработать подход к оценке зрелости данной трансформации. Для достижения цели поставлены следующие задачи: 1) проанализировать ключевые направления применения ИИ в управлении логистическими процессами; 2) выявить специфику влияния ИИ на отдельные функциональные области логистики; 3) определить барьеры, препятствующие ИИ-трансформации логистики в российских организациях; 4) предложить классификацию уровней зрелости ИИ-ориентированной логистики.

Методологическую основу исследования составляют системный подход к анализу логистических процессов, метод сравнительного анализа научных источников, а также элементы контент-анализа публикаций отечественных авторов в рецензируемых изданиях за 2023–2024 гг. В работе применяются описательный метод, классификация и типологизация, позволяющие систематизировать разрозненные данные о практике внедрения ИИ в логистику в единую концептуальную схему.

Проблематика применения цифровых технологий в логистике нашла отражение в ряде отечественных и зарубежных исследований последних лет. Хорошлова Т.Н. [2] систематизировала ключевые приложения ИИ в логистике от оптимизации маршрутов до предиктивной аналитики и охарактеризовала основные вызовы их практического внедрения. Данилочкина Н.Г. и Лысенко А.А. [3] разработали модель оптимизации маршрутов на основе ИИ с применением многофакторной оценки и адаптивной эвристики. Иванова Л.Н. и Иванов С.Е. [4] предложили алгоритм маршрутизации в транспортной логистике, учитывающий затраты на топливо и временные ограничения. Мартиросян Г.Н. и Давтян Г.Г. [5] исследовали специфику управления цепями поставок в условиях цифровой экономики и концепцию «Цепочки поставок 4.0». Вместе с тем в существующей литературе недостаточно проработан вопрос о системном измерении зрелости ИИ-трансформации логистики как целостного процесса. В ходе анализа научных источников и практики отечественных предприятий выделены четыре основных направления трансформации логистических процессов под влиянием ИИ.

Первое направление интеллектуальная маршрутизация транспортировки. Классические методы планирования маршрутов строятся на фиксированных моделях, не учитывающих динамических изменений дорожной ситуации, по-

годных условий и загрузки транспортного средства. Алгоритмы ИИ позволяют в режиме реального времени корректировать маршруты с учётом многофакторного анализа.

Второе направление управление запасами на основе прогнозирования спроса. Традиционные методы, включая модель Уилсона, ABC-XYZ-анализ и статистические модели временных рядов, демонстрируют ограниченную точность в условиях высокой волатильности спроса. Современные алгоритмы машинного обучения, в том числе рекуррентные нейросети типа LSTM способны учитывать не только исторические данные о продажах, но и внешние факторы: сезонные колебания, маркетинговые акции, динамику в социальных сетях [6]. Применение подобных систем позволяет снизить уровень избыточных запасов и одновременно уменьшить частоту возникновения дефицитных ситуаций, что в совокупности ведёт к сокращению суммарных логистических издержек.

Третье направление автоматизация складских операций с применением компьютерного зрения и роботизированных комплексов. ИИ-системы управления складом (WMS нового поколения) оптимизируют очередность комплектования заказов, прокладывают эффективные маршруты внутри склада и обеспечивают контроль состояния хранения грузов через датчики IoT. Согласно имеющимся оценкам, внедрение таких систем ускоряет процесс комплектации на 20–30% при одновременном снижении количества ошибок. Схожий эффект фиксируется в отечественной практике: X5 Group в 2024 году запустила на складах торговой сети «Пятёрочка» технологию световой комплектации Put-by-Light совместно с автономными мобильными роботами-погрузчиками. По расчётам компании, автоматизация двух этапов складского процесса позволит ежегодно сокращать фонд оплаты труда на 4,7 млрд рублей, а ожидаемый прирост производительности составляет свыше 20%; к 2028 году планируется полная роботизация процессов хранения и комплектации по модели «товар к человеку» [7]. Представленные обобщенные данные о этапах развития и внедрения ИИ в логистических организациях представлена ниже в таблице 1.

Таблица 1

Четырёхуровневая модель зрелости ИИ-трансформации логистики организации

Уровень	Название	Характеристика	Типичные инструменты
Уровень 1	Точечная автоматизация	ИИ автоматизирует отдельные расчётные функции. Менеджер сохраняет полный контроль над принятием решений, используя алгоритм как инструмент расчёта.	Оптимизация маршрутов, составление графиков доставки
Уровень 2	Интеграция прогностических алгоритмов	ИИ встраивается в ключевые управленческие процессы. Алгоритмы формируют повестку принятия решений, менеджер работает с агрегированными рекомендациями.	Прогнозирование спроса, планирование закупок, управление запасами

Уровень	Название	Характеристика	Типичные инструменты
Уровень 3	Единая цифровая среда	Создаётся сквозная платформа управления цепями поставок с ИИ-компонентами. Все участники цепи обмениваются данными в режиме реального времени.	WMS/TMS нового поколения, предиктивное обслуживание, «Цепочка поставок 4.0»
Уровень 4	Автономная логистическая платформа	Алгоритмические агенты самостоятельно принимают оперативные решения в установленных границах.	Автономные агенты диспетчеризации, интеллектуальные склады без участия оператора

Наряду с возможностями выявлены значимые барьеры ИИ-трансформации. К ним относятся: дефицит качественных структурированных данных, накопленных в совместимых форматах; высокие затраты на разработку и интеграцию интеллектуальных систем; нехватка специалистов, способных проектировать и сопровождать ИИ-алгоритмы; риски информационной безопасности при передаче коммерческих данных в облачные среды; а также технологическая зависимость от зарубежного программного обеспечения, ставшая особенно ощутимой в условиях санкционных ограничений.

Проведённое исследование позволяет сформулировать следующие выводы. Во-первых, трансформация логистических процессов под влиянием ИИ носит системный, а не точечный характер: она затрагивает все функциональные области логистики от транспортировки до складирования и управления запасами. Во-вторых, практический эффект от внедрения ИИ достигается не при изолированном использовании отдельных инструментов, а при их интеграции в единую информационную среду управления цепями поставок. В-третьих, успех ИИ-трансформации в значительной мере определяется качеством исходных данных и уровнем цифровых компетенций персонала факторами, которые требуют инвестиций не меньших, чем сами технологии. В-четвёртых, предложенная четырёхуровневая модель зрелости даёт организациям инструмент для самодиагностики текущего положения и определения приоритетов дальнейшего развития. Дальнейшие исследования целесообразно направить на апробацию данной модели на выборке конкретных предприятий с целью её эмпирической верификации.

Литература

1. Цифровые инструменты внешнеторговой логистики в 2024 году: проблемы и решения // КиберЛенинка. 2024. URL: <https://cyberleninka.ru/article/n/tsifrovye-instrumenty-vneshnetorgovoy-logistiki-v-2024-godu-problemy-i-resheniya> (дата обращения: 20.05.2026).
2. Хорошилова Т. Н. Роль искусственного интеллекта в логистике: эффективность, вызовы и решения // Universum: технические науки. 2024. URL: <https://cyberleninka.ru/article/n/rol-iskusstvennogo-intellekta-v-logistike-effektivnost-vyzovy-i-resheniya> (дата обращения: 20.05.2026).
3. Данилочкина Н. Г., Лысенко А. А. Оптимизация логистических маршрутов посредством применения технологий искусственного интеллекта // Научные труды. 2024. URL:

<https://cyberleninka.ru/article/n/optimizatsiya-logisticheskikh-marshrutov-posredstvom-primeneniya-tehnologiy-iskusstvennogo-intellekta> (дата обращения: 20.05.2026).

4. Иванова Л. Н., Иванов С. Е. Методы оптимизации и алгоритм маршрутизации в транспортной логистике // КиберЛенинка. 2024. URL: <https://cyberleninka.ru/article/n/metody-optimizatsii-i-algoritm-marshrutizatsii-v-transportnoy-logistike> (дата обращения: 20.05.2026).

5. Мартиросян Г. Н., Давтян Г. Г. Специфика и возможности управления цепями поставок в условиях цифровой экономики // Прогрессивная экономика. 2024. №10. С. 200–209.

6. Прогнозирование спроса и управление запасами с помощью машинного обучения // КиберЛенинка. 2024. URL: <https://cyberleninka.ru/article/n/prognozirovanie-sprosa-i-upravlenie-zapasami-s-pomoschyu-mashinnogo-obucheniya> (дата обращения: 20.05.2026).

7. Гуцин К. С. Цифровые инструменты внешнеторговой логистики в 2024 году: проблемы и решения // Российский внешнеэкономический вестник. 2024. № 10. С. 126–134. DOI: 10.24412/2072-8042-2024-10-126-134.

TRANSFORMATION OF LOGISTICAL PROCESSES OF ORGANIZATIONS UNDER THE INFLUENCE OF ARTIFICIAL INTELLIGENCE TECHNOLOGIES

Tanchinets Petr Petrovich

Postgraduate Student

E-mail: p.tanchinets@yandex.ru

Barmuta Karine Aleksandrovna

Doctor of Economics, Professor

E-mail: carinaba@yandex.ru

Don State Technical University

Russia, 344000, Rostov-on-Don, Gagarin Square 1

Abstract. The article explores the transformation of logistic processes of organizations under the influence of artificial intelligence technologies. The article discusses the key areas of AI application in logistics: route optimization, inventory management based on demand forecasting, intelligent automation of warehouse operations, and predictive maintenance of the vehicle fleet. The authors identify the barriers to implementation and propose an original approach to assessing the maturity of AI-based logistics transformation at enterprises.

Keywords: artificial intelligence, logistics, business process transformation, machine learning, inventory management, route optimization, digital transformation, and information systems.

ИССЛЕДОВАНИЕ ПРИМЕНЕНИЯ МЕТОДОВ СЖАТИЯ ДАННЫХ В СИСТЕМАХ ХРАНЕНИЯ ИНФОРМАЦИИ

© **Паршенцева Дарья Дмитриевна**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: dasha.parshentseva@yandex.ru

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artyom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье рассматриваются методы сжатия данных, которые используются в системах хранения информации. На примере кодирования повторяющихся символов проанализирован алгоритм сжатия без потерь, реализован и протестирован базовый алгоритм на различных массивах данных. Выполнено сопоставление объемов исходных и сжатых данных, а также вычислен коэффициент сжатия. Полученные результаты показывают эффективность метода для уменьшения объема хранимой информации с сохранением её структуры.

Ключевые слова: сжатие данных, алгоритмы сжатия, кодирование без потерь, RLE, системы хранения данных, коэффициент сжатия, обработка данных.

Введение. В настоящее время наблюдается стремительный рост объемов данных, которые обрабатываются и хранятся в информационных системах. В связи с данным обстоятельством увеличивается рост требований к системам хранения и передачи данных [4].

Актуальность. Управление массивами данных превратилось в одну из центральных задач информационных технологий, поскольку объемы информации продолжают расти, а ресурсы для ее хранения и передачи остаются ограниченными. Алгоритмы сжатия позволяют сократить размер хранимых данных, сохраняя при этом их содержимое, поэтому выбор подходящего метода становится принципиальным для современных информационных систем. Проблема выбора обостряется в связи с развитием облачных хранилищ, больших данных и интернета вещей, где информация стремительно накапливается.

Объект и предмет исследования. Объектом исследования выступают процессы хранения и обработки данных в информационных системах. Предметом исследования являются методы сжатия данных без потерь, в частности алгоритм RLE.

Цель и задачи. Цель – изучение применения способов сжатия данных и оценка их эффективности на примере алгоритма RLE. Задачи:

1. рассмотреть теоретические основы сжатия.
2. реализовать алгоритм RLE на Python.
3. провести тестирование.
4. проанализировать результаты.

Методология и методы исследования. В работе использованы методы теоретического анализа, алгоритмизации, программирования и экспериментального сравнения с вычислением коэффициента сжатия.

Для оценки результативности сжатия использовался коэффициент сжатия, вычисляемый как отношение исходного размера данных к размеру после применения алгоритма. Значение коэффициента больше 1 указывает на эффективное сжатие, а меньше 1 – на увеличение объёма данных. Тестирование проводилось на трёх типах строк: с высокой степенью повторений, с уникальными символами и с полностью повторяющимся символом.

Разработанность темы. Данный вопрос подробно рассмотрен в работах Ватолина Д., Вирта Н., Сэйдуда К. и др. В них подробно продемонстрированы вопросы сжатия информации. Тем не менее, актуальность для образовательных задач сохраняет и практический анализ с наглядной реализацией алгоритмов.

Уровень эффективности используемых методов определяется структурой исходных данных и выбранным подходом к её обработке [2, 4].

Теоретические основы сжатия данных. Традиционно различают два подхода: сжатие с потерями и без. В репозиториях преимущественно используются методы без потерь, поскольку они гарантируют полную обратимость исходного массива [5]. В качестве примера можно привести кодирование повторяющихся последовательностей (Run-Length Encoding, RLE) [2, 3]. Его суть состоит в замене последовательностей идентичных знаков на комбинацию самого символа и числа его повторений. К примеру, строка «ААААВВВСС» трансформируется в «А5В3С2», что экономит место при наличии повторов [1, 2].

Реализация алгоритма. Алгоритм сжатия данных методом RLE был реализован на языке Python в ходе практической работы, поскольку именно этот язык позволяет продемонстрировать суть метода при минимальном объеме кода. Программа анализирует входную последовательность символов, вычисляет длины непрерывных серий одинаковых элементов и формирует на выходе пары вида «символ–количество повторений». После завершения обработки на экран выводятся сжатая строка и сведения о размере исходной и преобразованной последовательностей, что позволяет наблюдать изменение объема информации в результате применения алгоритма. [1, 3]

Ядром программы служит функция `compress_rle(data)`, принимающая строку и возвращающая ее упакованное представление. Логика работы следующая: переменная `count` получает начальное значение, равное единице, затем программа последовательно перебирает символы строки и сравнивает каждый из них со следующим. При совпадении символов счетчик увеличивается на единицу. Когда текущий символ отличается от следующего, в результирующую строку добавляется пара «символ + значение `count`», а счетчик сбрасывается. Временная сложность алгоритма составляет $O(n)$, благодаря чему он пригоден для обработки больших массивов данных.

Результат. В рамках исследования алгоритма были проведены тесты с разными наборами входных данных.

Для оценки эффективности использовались размеры исходных и сжатых данных, а также коэффициент сжатия.

Первый пример теста («AAAAABBBBCCCC») содержит повторяющиеся группы символов. Второй – («ABCDEFGG») представляет собой последовательность уникальных символов (здесь алгоритм может даже увеличить объем информации). Третьим примером («AAAAAAAAAA») является строка, состоящая из десяти одинаковых символов.

Результаты эксперимента представлены в таблице 1.

Таблица 1

Результаты сжатия данных

Данные	Размер до	Размер после	Коэффициент
AAAAABBBBCCCC	13	8	1.63
ABCDEFGG	7	14	0.5
AAAAAAAAAA	10	3	3.33

Лучший коэффициент (3,33) достигнут для строки, полностью состоящей из одного повторяющегося символа. Худший (0,5) получен для строки с уникальными символами, где каждый символ потребовалось дополнить единицей и поэтому размер увеличился вдвое.

Из таблицы видно: эффективность алгоритма RLE сильно зависит от структуры входящих данных [4, 5]. Длинные цепочки повторяющихся символов позволяют существенно уменьшить объем. Но если они отсутствуют, метод способен даже увеличить размер массивов – это его главный недостаток [1]. Поэтому RLE стоит использовать только для данных с высокой избыточностью.

Реализованный алгоритм подходит в учебных целях, поскольку на простых примерах демонстрирует принцип работы сжатия, а результаты тестов помогут в дальнейшем выбрать подходящий метод с учетом типа хранимой информации. В следующих работах можно сравнить RLE с алгоритмом Хаффмана или LZW и оценить их эффективности на больших объемах данных.

Таким образом, алгоритм RLE лучше всего работает на данных с длинными сериями повторов. В реальных системах хранения информации такие данные встречаются в лог-файлах с повторяющимися записями, в растровых изображениях с большими областями одного цвета, в текстовых документах с частыми пробелами или повторяющимися символами.

Сравнение с другими алгоритмами сжатия. Кодирование длин серий (RLE) относится к числу более простых алгоритмов сжатия и принципиально отличается от алгоритмов Хаффмана или LZW. Метод Хаффмана подсчитывает частоту каждого символа и самым частым присваивает более короткие коды, а LZW строит словарь повторяющихся строк и заменяет их ссылками на этот словарь. В отличие от них, RLE ограничен работой с длинными повторяющимися последовательностями. Однако у него есть и сильные стороны – простота реализации и высокая скорость, что позволяет RLE оставаться оптимальным выбором для

сжатия данных в реальном времени. Кроме того, он удобен для маломощных устройств, в которых сложные алгоритмы не удастся запустить из-за нехватки памяти или производительности.

Выводы. В данной работе был рассмотрен алгоритм кодирования повторяющихся последовательностей (RLE) и написана его реализация на языке Python.

Тестирование показало, что метод сокращает объем хранимой информации только тогда, когда во входных данных присутствуют повторяющиеся элементы, поэтому эффективность сжатия зависит от исходного типа данных [5].

С учетом полученных результатов можно утверждать, что алгоритм RLE полезен для сжатия данных в системах, хранящих данные с высокой избыточностью [3].

Перспективы дальнейших исследований. Планируется расширить работу в реализации и сравнении других алгоритмов сжатия без потерь и проверить, какой из них более эффективен в использовании. Также можно детальнее изучить, как меняется скорость сжатия в зависимости от объема входных данных.

Литература

1. Ватолин Д. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. Москва: Диалог-МИФИ, 2003. 384 с.
2. Вирт Н. Алгоритмы и структуры данных. Москва: ДМК Пресс, 2018. 272 с.
3. Таненбаум Э., Остин Т. Архитектура компьютера. Санкт-Петербург: Питер, 2017. 816 с.
4. Manz O. Well Packed – Not a Bit Too Much: Compression of Digital Data Explained in an Understandable Way. Wiesbaden: Springer, 2021. 57 p.
5. Sayood K. Introduction to Data Compression. 4th ed. Burlington: Morgan Kaufmann, 2012. 720 p.

RESEARCH OF DATA COMPRESSION METHODS APPLICATION IN INFORMATION STORAGE SYSTEMS

Parshantseva Daria Dmitrievna

Student

E-mail: dasha.parshentseva@yandex.ru

Matitsin Artem Vladimirovich

Student, e-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article discusses data compression methods that are used in information storage systems. Using the example of encoding repeating characters, the lossless compression algorithm is analyzed, and the basic algorithm is implemented and tested on various data arrays. The volumes of the original and compressed data are compared, and the compression ratio is calculated. The results show the effectiveness of the method in reducing the storage volume while preserving the structure of the data.

Keywords: data compression, compression algorithms, lossless encoding, RLE, data storage systems, compression ratio, data processing.

СТОХАСТИЧЕСКАЯ МОДЕЛЬ ПРОГНОЗИРОВАНИЯ ОТКАЗОВ ВЫЧИСЛИТЕЛЬНЫХ РЕСУРСОВ В РАСПРЕДЕЛЁННЫХ СИСТЕМАХ

© Жуманиязов Худойберган Максуд Угли

аспирант,

Вологодский государственный университет

Россия, 160000, г. Вологда, ул. Ленина, 15

E-mail: jexams@ya.ru

Аннотация. Предложена стохастическая модель прогнозирования отказов вычислительных ресурсов, функционирующих в распределённых системах организации. Модель включена в многоагентный контур мониторинга и управления, в котором сбор наблюдаемых данных, оценивание интенсивности отказов, формирование рекомендаций, ранжирование управляющих воздействий и их реализация разделены по функциональным агентам. Вероятность отказа оценивается на основе экспоненциальной модели интенсивности и байесовского уточнения по мере поступления новых наблюдений. Показано, что риск-ориентированное правило выбора управляющего воздействия обеспечивает снижение ожидаемого ущерба при сохранении допустимых затрат на вмешательство.

Ключевые слова: распределённые системы, мониторинг, прогнозирование отказов, стохастическая модель, многоагентная архитектура

В распределённых вычислительных системах отказ отдельного ресурса должен рассматриваться не как редкое исключение, а как вероятностно ожидаемое событие, возникающее под влиянием нагрузки, неравномерности эксплуатации и накопления скрытых дефектов. При этом нарушение работоспособности одного узла вызывает цепочку вторичных отклонений, связанных с деградацией, ростом задержек. Вследствие этого задача прогнозирования отказов приобретает важное значение и должна решаться в составе системы поддержки принятия решений.

Для распределённых систем предпочтительным считается предиктивный подход, где управляющее воздействие формируется до наступления аварийного состояния. Такой подход позволяет согласовать технические меры восстановления с критериями риска и критичности ресурса. В существующих работах отмечается, что наибольший эффект достигается при совместном использовании статистического анализа и интеллектуальных механизмов выбора действий [1-3].

Актуальность исследования определяется ростом сложности организационных вычислительных инфраструктур, в которых одновременно функционируют серверы, сетевые сервисы и прикладные платформы. В подобных условиях наблюдаемость системы оказывается неполной, а отдельные признаки предстоящего отказа могут проявляться только в комбинации нескольких метрик. Также стоимость простоя вычислительных ресурсов в организационной среде включает не только технические потери, но и задержку бизнес-процессов. По этой причине решение задачи должно опираться не на фиксированные правила, а на вероятностную модель. Подобный подход согласуется с результатами ис-

следований по прогнозированию надёжности оборудования и по построению систем мониторинга состояния [1, 2].

Объектом исследования являются распределённые вычислительные ресурсы организации, функционирующие в условиях переменной нагрузки. Предмет исследования – стохастические закономерности возникновения и развития отказов, а также методы их прогнозирования в многоагентной среде мониторинга и управления. Цель исследования состоит в разработке стохастической модели прогнозирования отказов, обеспечивающей оценивание вероятности наступления отказа на заданном горизонте и формирование риск-ориентированных управляющих воздействий.

Для достижения поставленной цели были сформулированы следующие задачи: формализация наблюдаемого состояния ресурса; выбор вероятностной схемы описания деградации; разработка критерия ранжирования управляющих воздействий; проведение расчётов, подтверждающих работоспособность модели.

Методы исследования

В качестве методологической основы были использованы системный подход, теория стохастических процессов, методы байесовского оценивания и агентно-ориентированное моделирование. Выбор указанных средств был обусловлен необходимостью совместного описания случайной природы отказов, динамики изменения состояния ресурса.

Пусть в момент времени t состояние ресурса i задаётся вектором наблюдаемых признаков $X_i(t) = (x_{i1}(t), x_{i2}(t), \dots, x_{im}(t))$, включающим показатели загрузки процессора, памяти, дисковой подсистемы, сети и времени отклика. Для описания вероятности отказа была принята экспоненциальная форма интенсивности:

$$\lambda_i(t) = \lambda_{0i} \cdot \exp(\beta_i^T X_i(t)).$$

Здесь λ_{0i} обозначает базовую интенсивность отказа, а β_i – вектор чувствительности ресурса к наблюдаемым признакам. Функция безотказной работы на интервале $[0, t]$ определяется выражением:

$$S_i(t) = \exp\left(-\int_0^t \lambda_i(\tau) d\tau\right).$$

Вероятность отказа на горизонте Δ оценивается как:

$$P_i(t, \Delta) = 1 - \exp\left(-\int_t^{t+\Delta} \lambda_i(\tau) d\tau\right).$$

По мере поступления новых наблюдений оценка уточняется по байесовскому правилу:

$$P(F_i|X_t) = \frac{P(X_t|F_i)P(F_i)}{\sum_{k=1}^n P(X_t|F_k)P(F_k)}.$$

Риск ресурса задаётся произведением вероятности отказа на ожидаемый ущерб:

$$R_i(t, \Delta) = P_i(t, \Delta) \cdot C_i.$$

Если множество допустимых управляющих воздействий обозначить через $A = \{a_1, a_2, \dots, a_s\}$, то выбор может быть выполнен по критерию:

$$a^* = \arg \max_{a \in A} U(a), \quad U(a) = \sum_{i=1}^n w_i \Delta R_i(a) - \gamma Q(a).$$

Здесь w_i – критичность ресурса, $R_i(a)$ – снижение риска после применения воздействия, $Q(a)$ – стоимость вмешательства, а γ – коэффициент штрафа за затраты. Тем самым прогноз отказа и выбор управляющего решения оказываются связанными единым критерием.

В работах, посвящённых прогнозированию технического состояния и надёжности, была показана результативность многофакторных статистических моделей [1]. В исследованиях по анализу малых космических аппаратов было подтверждено, что наблюдаемые параметры могут служить основой для выявления предаварийных режимов [2]. При построении перспективных систем мониторинга технического состояния подчёркивается необходимость событийного уведомления лица, принимающего решение, и перехода от пассивного наблюдения к активной поддержке управления [3]. Кроме того, в исследованиях по многоагентным системам были уточнены базовые модели агентов и постановки задач, что создаёт теоретическую основу для функционального разделения мониторинга, прогнозирования и принятия решений [4]. В публикациях, посвящённых программным агентам организации вычислений в крупномасштабных системах, была показана возможность использования агентной логики для распределения вычислительных функций и повышения устойчивости управления [5]. В совокупности эти результаты подтверждают целесообразность построения стохастической модели отказов именно в составе агентно-ориентированной архитектуры.

Результаты исследования

Предложенная архитектура включает пять классов агентов: агенты мониторинга осуществляют сбор данных; агенты прогнозирования выполняют оценивание интенсивности отказа и расчёт вероятности его наступления; агенты рекомендаций формируют набор допустимых управляющих действий; агенты принятия решений ранжируют действия по критериям эффективности и риска; агенты управления реализуют выбранное воздействие. Был сформирован замкнутый контур: наблюдение – прогноз – рекомендация – решение – управление. Структурная декомпозиция делает систему масштабируемой и допускает включение новых источников данных без перестройки всей логики управления. Кроме того, разделение функций между агентами снижает связанность компонентов и упрощает локализацию причин деградации ресурса.

Для проверки работоспособности модели был рассмотрен ресурс с нормированным интегральным показателем нагрузки $u(t) \in [0; 1]$ (см. рис. 1). Параметры модели были приняты следующими: $\lambda_0 = 0.018$; $\beta = 2.4$; $C = 120$ усл. ед. При $u = 0.82$ интенсивность отказа определяется как $\lambda = 0,018 \cdot \exp(2.4 \cdot 0.82) = 0.1287$. Тогда вероятность отказа на ближайший

интервал $\Delta = 1$ составит $P = 1 - \exp(-0.1287) = 0.1208$, а ожидаемый риск равен $R = 0.1208 \cdot 120 = 14.50$ усл. ед. После применения управляющего воздействия, связанного с перераспределением нагрузки, значение λ было снижено до 0.58. В этом случае $\lambda = 0.0724, P = 0.0699, R = 8.39$ усл. ед.

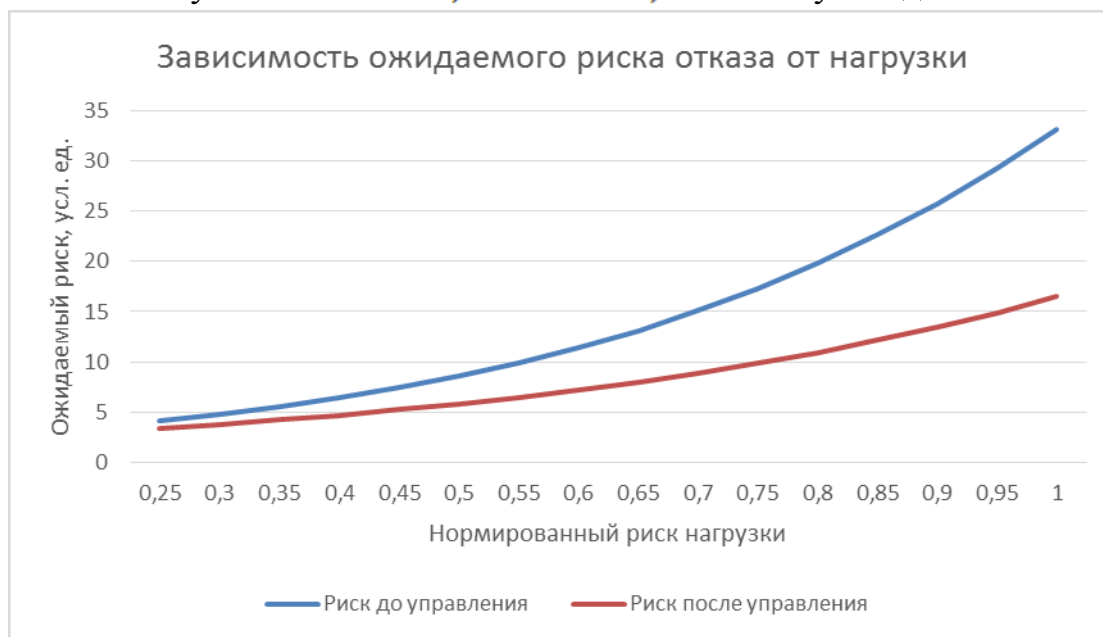


Рисунок 1. Зависимость ожидаемого риска отказа от нагрузки до и после управляющего воздействия

На рисунке 1 видно, что при росте нормированного индекса нагрузки риск увеличивается нелинейно, причём после управляющего воздействия кривая располагается ниже исходной. Это означает, что даже при одинаковой структуре модели изменение состояния ресурса после вмешательства приводит к существенному снижению ожидаемого ущерба. Следовательно, оценка риска может быть использована как критерий ранжирования управляющих воздействий и как основание для автоматического выбора момента вмешательства.

Обсуждение результатов

Полученные результаты позволяют утверждать, что предлагаемая модель отличается практической применимостью в условиях неполной информации. Её преимущество состоит в том, что прогноз отказа формируется не только на основе текущих значений метрик, но и с учётом их вклада в интенсивность перехода в аварийное состояние. Если вероятность отказа превышает установленный порог, а ожидаемый риск оказывается выше допустимого уровня, то агентами рекомендаций может быть предложен набор действий, включающий перераспределение нагрузки, перенос сервисов, ограничение интенсивных заданий и предварительное резервирование ресурсов.

Тем самым обеспечивается не только фиксация вероятности отказа, но и содержательная связь между диагностикой и управлением. В отличие от статистических схем, в предложенной модели вычисленный показатель риска служит промежуточным звеном между наблюдением и воздействием, что повышает интерпретируемость получаемых решений.

Для практического внедрения особое значение имеет то, что модель допускает последовательное уточнение параметров по мере накопления новых данных. Это делает возможным её применение в системах непрерывного мониторинга, где состояние ресурсов меняется во времени и решение должно пересматриваться без остановки процесса вычисления.

Выводы

1. Стохастическая постановка задачи позволила формализовать прогнозирование отказов вычислительных ресурсов как оценивание вероятности перехода ресурса в неработоспособное состояние на заданном горизонте времени.

2. Многоагентная декомпозиция обеспечила функциональное разделение мониторинга, прогнозирования, формирования рекомендаций, принятия решений и управления, что является методически оправданным для распределённых систем.

3. Введение риск-ориентированного критерия позволило связать технический прогноз с управленческим решением, поскольку в нём одновременно учитываются вероятность отказа, критичность ресурса и стоимость вмешательства.

4. Расчёты подтвердили, что при снижении нормированной нагрузки с **0.82** до **0.58** ожидаемый риск уменьшился с **14.50** до **8.39** усл. ед., то есть более чем на **42%**.

5. Предложенная модель может быть использована как основа для построения рекомендательных агентно-ориентированных систем мониторинга состояния вычислительных ресурсов организации и для развития интеллектуальных систем поддержки принятия решений.

6. Научная новизна состоит в согласовании вероятностного прогноза отказов с процедурой ранжирования управляющих воздействий в едином контуре агентного управления.

Литература

1. Константинов А. А., Султанов М. М. Разработка методики оценки и прогнозирования уровня надежности энергетического оборудования ТЭС на базе показателей технического состояния // Вестник Казанского государственного энергетического университета. 2021. Т. 13, № 1(49). С. 141-151. EDN TGOKQA.

2. Скобцов В. Ю. Анализ данных телеметрии и надежности малых космических аппаратов // Информатика и кибернетика. 2021. № 1-2(23-24). С. 65-72. EDN JKNMGW.

3. Аллакин В. В., Будко Н. П., Васильев Н. В. Общий подход к построению перспективных систем мониторинга распределенных информационно-телекоммуникационных сетей // Системы управления, связи и безопасности. 2021. № 4. С. 125-227. DOI 10.24412/2410-9916-2021-4-125-227. EDN JPFJRO.

4. Леденева Т. М., Принев М. А. Многоагентные системы: модели агентов и постановки задач, обеспечивающие их характерные свойства (часть 1) // Вестник Воронежского государственного университета. Сер. Системный анализ и информационные технологии. 2024. № 3. С. 50-73. DOI 10.17308/sait/1995-5499/2024/3/50-73. EDN GHJMUH.

5. Писарева О. М., Стефановский Д. В. Программный агент организации вычислений в крупномасштабных системах управления // Управление развитием крупномасштабных систем (MLSD'2024): Труды Семнадцатой международной конференции, Москва, 24–26 сентября 2024 года. Москва: Изд-во Ин-та проблем управления им. В. А.Трапезникова Российской академии наук, 2024. С. 1140-1145. EDN PPTVHH.

STOCHASTIC MODEL FOR PREDICTING COMPUTER RESOURCE FAILURES IN DISTRIBUTED SYSTEMS

Zhumaniyazov Khudoybergan Maksud Ugli

Postgraduate Student

Vologda State University

15 Lenin Street, Vologda, 160000, Russia

E-mail: jexams@ya.ru

Abstract. A stochastic model for predicting failures of computing resources operating in distributed systems of an organization is proposed. The model is included in a multi-agent monitoring and control loop, in which the collection of observed data, estimation of the failure rate, generation of recommendations, ranking of control actions, and their implementation are divided among functional agents. The probability of failure is estimated based on an exponential failure rate model and Bayesian refinement as new observations are received. It is shown that a risk-oriented rule for selecting a control action reduces the expected damage while maintaining acceptable intervention costs.

Keywords: distributed systems, monitoring, failure prediction, stochastic model, multi-agent architecture.

ВНЕШНЯЯ СОРТИРОВКА БОЛЬШИХ ОБЪЁМОВ ДАННЫХ: АЛГОРИТМ СЛИЯНИЯ ОТРЕЗКОВ И ЕГО ПРАКТИЧЕСКИЕ ОСОБЕННОСТИ

© **Кофанов Ярослав Евгеньевич**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: hoagokit666@gmail.com

© **Матыцин Артем Владимирович**

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье рассматривается задача сортировки массивов, размер которых превышает доступную оперативную память. Подробно разобран классический двухфазный алгоритм слияния отсортированных отрезков, оценена его временная и I/O-сложность, обсуждены практические приёмы: буферизация чтения и записи, k-путевое слияние с использованием приоритетной очереди, выбор размера блока, replacement selection и сжатие промежуточных файлов. Приведены примеры применения подхода в реальных системах: команда sort в GNU coreutils, дисковая сортировка в СУБД, shuffle-фаза в Hadoop и Apache Spark. Даны результаты практического замера времени сортировки файла объёмом 40 ГБ в зависимости от выбранных параметров.

Ключевые слова: внешняя сортировка, слияние отрезков, k-путевое слияние, приоритетная очередь, ввод-вывод, кэш, big data.

Введение. Когда говорят про сортировку, по умолчанию подразумевают, что весь массив помещается в оперативную память. На практике это далеко не всегда так. Логи веб-сервера за месяц, выгрузка из СУБД, индекс полнотекстового поиска или сырой дамп телеметрии запросто перевалят за десятки и сотни гигабайт. Загнать такое в RAM невозможно даже на сервере с большим объёмом памяти, и любой in-memory алгоритм здесь просто упадёт по out-of-memory. Возникает отдельный класс задач: сортировка во внешней памяти, где главный ресурс не процессор, а количество обращений к диску [1].

Актуальность. С ростом объёмов данных в современном мире проблема эффективной внешней сортировки становится всё более острой. Крупные компании ежедневно обрабатывают терабайты логов, выгрузок из баз данных и телеметрии, поэтому оптимизация алгоритмов внешней сортировки имеет большое практическое значение.

Объект и предмет исследования. Объектом исследования являются алгоритмы сортировки данных, не помещающихся в оперативную память. Предметом

исследования выступает классический алгоритм слияния отрезков, его модификации и оптимизации.

Цель и задачи. Цель работы — анализ алгоритма внешней сортировки слиянием отрезков и оценка его эффективности на практике. Задачи:

1. Рассмотреть постановку задачи внешней сортировки.
2. Проанализировать двухфазный алгоритм слияния.
3. Изучить k -путевое слияние с приоритетной очередью.
4. Оценить I/O-сложность алгоритма.
5. Описать практические оптимизации и примеры применения.

Методология и методы исследования. В работе использованы методы теоретического анализа алгоритмов, математического моделирования и экспериментальные методы (практические замеры времени сортировки).

Разработанность темы. Вопросы внешней сортировки детально рассмотрены в работах Кнута Д.Э., Седжвика Р., Аггарвала А. и Виттера Дж. Однако практические исследования с акцентом на современные оптимизации сохраняют актуальность для прикладных задач.

Постановка задачи. Дано: файл, содержащий N записей. Доступная оперативная память позволяет хранить одновременно не более M записей, причём M значительно меньше N . Нужно получить отсортированную по заданному ключу копию исходного файла, минимизируя при этом число операций ввода-вывода. Сам ввод-вывод считается основной стоимостью: одно чтение блока с диска на порядки дороже сравнения двух чисел в кэше процессора [2].

Алгоритм слияния отрезков. Идея алгоритма проста и при этом удивительно живуча, её, по сути, без изменений тащат с эпохи магнитных лент до современных SSD. Алгоритм делится на две фазы. В фазе формирования отрезков (run generation) исходный файл читается порциями по M записей, каждая порция сортируется в памяти любым быстрым внутренним алгоритмом (на практике это Quicksort или Timsort, в зависимости от языка и стандартной библиотеки) и сбрасывается во временный файл. В результате получается набор так называемых отсортированных отрезков, или ранов, общим объёмом, равным исходному. На фазе слияния эти отрезки объединяются в один отсортированный файл.

Самый простой вариант слияния двух путевой: берём два отрезка, читаем из них по одной записи, ту, что меньше, пишем в выходной файл, продвигаем соответствующий указатель и повторяем. Это работает, но требует ровно $\log_2(K)$ проходов по всем данным, где K - число изначальных отрезков. Каждый проход — это полный объём ввода-вывода, и на больших файлах такой подход быстро упирается в диск.

K -путевое слияние с приоритетной очередью. Чтобы не читать одни и те же данные по несколько раз, слияние делают k -путевым: одновременно открывают сразу k отрезков и поддерживают мини-кучу из их текущих головных элементов. На каждом шаге извлекают минимум из кучи, пишут его в выход и подгружают следующую запись из того же отрезка. Время на шаг $O(\log k)$, общее число операций сравнения $O(N \log k)$, а число проходов по данным сокращается до $\text{ceil}(\log_k(K))$. При $k=128$ и $K=1024$ хватает всего двух проходов вместо десяти [3].

Размер k нельзя задира́ть бесконечно. На каждый поток нужен свой буфер чтения, и, если буферов много, но они мелкие, диск начнёт проседать на seek-ах, особенно на HDD. На практике k подбирают так, чтобы суммарный размер буферов чтения укладывался примерно в половину доступной памяти, а вторая половина оставалась под выходной буфер и саму кучу.

Анализ сложности. Сложность по сравнениям у внешнего merge-sort такая же, как у внутреннего: $O(N \log N)$. Куда интереснее I/O-сложность, то есть число обращений к диску. В модели с блоком размера B и памятью M записей классический результат Аггарвала и Виттера выглядит так: $O((N/B) \cdot \log_{\{M/B\}}(N/B))$ обращений к диску [2]. На разумных параметрах (B порядка десятков килобайт, M порядка единиц гигабайт, N порядка сотен гигабайт) основание логарифма получается большим, и реальное число проходов по данным редко превышает два или три. Это и есть главное практическое наблюдение: грамотно выбранные параметры важнее вылизанной асимптотики.

Практические оптимизации. Чистый алгоритм из учебника на реальных данных без доводки работает заметно медленнее, чем мог бы. Ниже собраны приёмы, которые в большинстве случаев дают кратный прирост скорости.

Буферизация чтения и записи. Чтение и запись по одной записи за раз убьёт любую файловую систему: накладные расходы на системные вызовы съедают всё. На практике для каждого открытого отрезка держат буфер от 64 КБ до нескольких мегабайт, читают за один раз весь блок и работают с ним в памяти. На выходе тот же приём: пишем не сразу, а накапливаем в выходном буфере и сбрасываем целиком.

Двойной буфер и асинхронный ввод-вывод. Пока процессор разбирает один блок, операционная система или отдельный поток подгружает следующий. Это позволяет частично спрятать задержку диска за полезной работой. На HDD приём фактически удваивает реальную пропускную способность, на SSD выигрыш скромнее, но всё равно ощутим.

Replacement selection вместо обычной in-memory сортировки. Вместо того чтобы читать ровно M записей, сортировать и писать на диск, можно использовать кучу размера M и формировать отрезки переменной длины, в среднем в два раза длиннее, чем M [1]. Это уменьшает K и, соответственно, объём слияния. Метод старый, но до сих пор используется в промышленных СУБД.

Сжатие промежуточных файлов. Если данные хорошо жмутся (текст, JSON, повторяющиеся ключи), временные отрезки имеет смысл писать через быстрый кодек, например LZ4 или Snappy. CPU обычно загружен не полностью, а узкое место - диск, так что компромисс почти всегда в пользу сжатия. Важный нюанс: тяжёлые алгоритмы вроде gzip на максимальном уровне здесь не подходят, так как сами становятся узким местом.

Применение в реальных системах. На внешней сортировке слиянием по сей день держится огромное количество инфраструктуры. Команда sort из GNU coreutils умеет сортировать файлы, не помещающиеся в память, ровно по этому алгоритму; ей можно явно задать параметры --buffer-size и --batch-size, что напрямую соответствует M и k из теоретической модели [5]. В реляционных СУБД (PostgreSQL, Oracle, MS SQL) запросы с ORDER BY и GROUP BY на

больших таблицах, не вписывающихся в `work_mem`, валятся в дисковую сортировку с тем же двухфазным алгоритмом, спиллингом во временный `tablespace` и последующим слиянием [6]. В мире `big data shuffle`-фаза в Hadoop MapReduce и Apache Spark, по сути, и есть распределённая внешняя сортировка: каждый исполнитель сначала сортирует свою долю в памяти, спиллит на диск, а потом сливает локальные ранки перед отправкой по сети [4].

Краткий практический замер. На рабочей станции с 16 ГБ ОЗУ и SATA SSD был проведён небольшой эксперимент: исходный файл - 40 ГБ строк длиной около 100 байт каждая, ключ сортировки - первые 16 байт. При наивной двухпутевой схеме сортировка заняла около 47 минут. Переход на `k=64` и буфер по 4 МБ на отрезок сократил время до 18 минут. Дополнительное включение сжатия LZ4 для временных файлов опустило результат до 12 минут. Главный вклад дал именно переход с двух путевого слияния на многопутевое, всё остальное лишь подкручивает константу. Этот результат хорошо согласуется с теоретической оценкой числа проходов по данным.

Выводы. Сортировка слиянием отрезков - пример алгоритма, который выглядит элементарно, но при правильной реализации остаётся рабочей лошадкой обработки данных уже более полувека. Его сила не столько в самом цикле сравнений, сколько в учёте реальной модели вычислений с дорогим вводом-выводом. Для прикладного программиста главный практический вывод такой: при работе с объёмами, не помещающимися в память, нужно явно думать в терминах блоков, буферов и числа проходов по диску, а не только в терминах асимптотики по сравнениям. Тогда даже на скромном железе можно обработать сотни гигабайт за разумное время, не прибегая к экзотическим решениям и распределённым кластерам.

Литература

1. Кнут Д.Э. Искусство программирования. Т. 3. Сортировка и поиск. 2-е изд. Москва: Вильямс, 2007. 832 с.
2. Седжвик Р., Уэйн К. Алгоритмы на Java. 4-е изд. Москва: Вильямс, 2013. 848 с.
3. Aggarwal A., Vitter J.S. The Input/Output Complexity of Sorting and Related Problems // Communications of the ACM. 1988. Vol. 31, No. 9. P. 1116-1127.
4. GNU Coreutils: `sort` - sort lines of text files // GNU Project: [сайт]. URL: https://www.gnu.org/software/coreutils/manual/html_node/sort-invocation.html (дата обращения: 28.04.2026).
5. PostgreSQL Documentation. Resource Consumption: `work_mem` and `temp_file_limit` // PostgreSQL: [сайт]. URL: <https://www.postgresql.org/docs/current/runtime-config-resource.html> (дата обращения: 28.04.2026).
6. Zaharia M., Xin R.S., Wendell P. и др. Apache Spark: a unified engine for big data processing // Communications of the ACM. 2016. Vol. 59, No. 11. P. 56-65.

EXTERNAL SORTING OF LARGE DATA VOLUMES: ALGORITHM OF MERGING SEGMENTS AND ITS PRACTICAL FEATURES

Kofanov Yaroslav Evgenievich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: hoagokit666@gmail.com

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article considers the problem of sorting arrays whose size exceeds the available RAM. The classic two-phase algorithm for merging sorted segments is analyzed in detail, its time and I/O complexity is estimated, and practical techniques are discussed: read and write buffering, k-way merging using a priority queue, block size selection, replacement selection, and intermediate file compression. Examples of the approach application in real systems are given: the sort command in GNU coreutils, disk sort in DBMS, shuffle phase in Hadoop and Apache Spark. The results of practical measurement of sorting time of a file of 40 GB depending on the selected parameters are given.

Keywords: external sort, merge of segments, k-way merge, priority queue, I/O, cache, big data.

АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ ХЕШ-ТАБЛИЦЫ ПРИ РАЗНЫХ СТРАТЕГИЯХ РАЗРЕШЕНИЯ КОЛЛИЗИЙ (ОТКРЫТАЯ АДРЕСАЦИЯ VS ЦЕПОЧКИ)

© **Тарасов Никита Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: tnikitav99@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. Написана программа, сравнивающая два популярных способа разрешать коллизии в хеш-таблицах. Речь про открытую адресацию и про метод цепочек. Замерялось время работы вставки и поиска при разной степени заполненности таблицы. Выяснилось, что если таблица загружена слабо ($\alpha < 0.5$), то открытая адресация быстрее - сказывается то, что данные лежат в памяти подряд. А когда загрузка высокая ($\alpha > 0.8$), цепочки работают стабильнее и не тормозят так сильно. На ста запусках при $\alpha = 0.9$ открытая адресация проиграла цепочкам по скорости - разница составила примерно 74%.

Ключевые слова: хеш-таблица, коллизии, цепочки, открытая адресация, производительность, коэффициент загрузки.

Введение. Хеш-таблицы широко применяются в программировании благодаря своей теоретической эффективности. В теории операции вставки, поиска и удаления работают в среднем за $O(1)$. Однако на практике возникает проблема коллизий - ситуаций, когда два разных ключа после хеширования попадают в одну и ту же ячейку массива [1]. Существует два основных подхода к решению этой проблемы. Первый - метод цепочек. В каждой ячейке хранится указатель на связный список, и все элементы, попавшие в один индекс, добавляются в этот список. Второй подход - открытая адресация. Здесь никаких списков нет, и при коллизии начинается поиск другого свободного места прямо в самой таблице по определённым правилам. Однозначного ответа, какой метод лучше, не существует - многое зависит от условий эксплуатации [4]. В данной работе проведено экспериментальное сравнение обеих стратегий с целью выработки практических рекомендаций.

Актуальность. Хеш-таблицы являются одной из самых востребованных структур данных в программировании. Выбор стратегии разрешения коллизий напрямую влияет на производительность приложений, особенно при работе с большими объёмами данных. Практические рекомендации по выбору между открытой адресацией и цепочками являются актуальной задачей.

Объект и предмет исследования. Объектом исследования являются хеш-таблицы как структуры данных. Предметом исследования выступают две стратегии разрешения коллизий: открытая адресация и метод цепочек.

Цель и задачи. Цель работы — экспериментальное сравнение производительности открытой адресации и метода цепочек при разных коэффициентах загрузки. Задачи:

1. Реализовать обе стратегии на C++.
2. Провести замеры времени вставки и поиска при α от 0.1 до 0.95.
3. Проанализировать полученные результаты.
4. Сформулировать рекомендации по выбору стратегии.

Методология и методы исследования. В работе использованы методы экспериментального программирования на C++, сравнительного анализа производительности и статистической обработки результатов (усреднение по 10 запускам).

Разработанность темы. Вопросы хеширования и разрешения коллизий подробно рассмотрены в работах Кнута Д.Э., Кормена Т. и др. Однако практические экспериментальные сравнения с акцентом на современные рекомендации сохраняют актуальность.

Результаты исследования. Разберём сначала цепочки. Заводится массив, в каждой ячейке которого хранится ссылка на начало списка (или `nullptr`, если список пуст). При добавлении нового элемента вычисляется его хеш-индекс, после чего элемент прицепляется к соответствующему списку. Добавление в начало списка выполняется за константное время, что удобно. Средняя временная сложность оценивается как $O(1 + \alpha)$, где $\alpha = n/m$ - коэффициент заполнения (отношение числа элементов n к размеру таблицы m) [3]. Пока α не превышает единицу, производительность остаётся приемлемой. Если же таблица маленькая, а элементов много, списки становятся длинными, и поиск начинает замедляться. В худшем случае, когда хеш-функция работает плохо и все ключи попадают в одну ячейку, структура вырождается в связный список, и все преимущества хеш-таблицы теряются [6]. У метода цепочек есть достоинства. Удаление выполняется просто - достаточно перекинуть указатели в списке, не затрагивая остальные ячейки. Кроме того, цепочки работают при любом коэффициенте заполнения, хотя и с падением скорости. Недостатки: дополнительные затраты памяти на указатели (при маленьких ключах они могут удваивать потребление памяти) и, что важнее, плохая дружелюбность к кэшу процессора. Элементы списков разбросаны по динамической памяти, и при обходе списка происходят частые промахи кэша, что снижает реальную скорость работы [2].

Открытая адресация устроена иначе. Все элементы хранятся непосредственно в массиве, без списков. При коллизии таблица начинает перебирать ячейки по определённому правилу (стратегии проб), пока не найдёт свободное место.

Существует три основных типа проб. Линейная проба - проверка ячеек i , $i+1$, $i+2$ и так далее. У этого варианта отличная локальность данных (массив идёт подряд, кэш работает эффективно), но возникает явление первичной кластеризации - образование длинных непрерывных занятых участков, которые растут со временем. Квадратичная проба использует шаги 1, 4, 9, 16... что снижает кластеризацию, но не гарантирует обхода всех ячеек. Наиболее надёжный вариант - двойное хеширование, где шаг пробы вычисляется второй хеш-функцией, что обеспечивает почти равномерное распределение проб [5]. Главное преимущество открытой адресации - экономия памяти (отсутствуют указатели). При линейной пробе данные идут подряд, и кэш работает очень эффективно, пока таблица не слишком забита. Однако есть два серьёзных недостатка. Первый - резкое падение производительности при высокой загрузке. Теоретическое среднее число проб при успешном поиске описывается формулой $(1/\alpha) * \ln(1/(1-\alpha))$ [3]. При $\alpha = 0.9$ получается около 2.6 проб, при $\alpha = 0.99$ - уже почти пять, что на практике ощутимо замедляет работу. Вторым недостатком - сложность удаления. Нельзя просто очистить ячейку - это разорвёт цепочку проб для последующих элементов. Поэтому удалённые ячейки маркируются специальным образом. При поиске такие ячейки пропускаются, при вставке могут заниматься заново. Со временем накапливается много «дырок», и производительность падает даже при небольшом реальном количестве элементов [4].

Для экспериментального сравнения была написана программа на C++. Размер таблицы зафиксирован - 10 тысяч ячеек. Количество элементов менялось, чтобы получить коэффициенты загрузки α от 0.1 до 0.95. Для открытой адресации использовалось двойное хеширование как самый надёжный вариант. Для цепочек - обычные односвязные списки без оптимизаций. На каждой конфигурации сначала вставлялось 50 тысяч элементов (для «прогрева»), затем выполнялось 100 тысяч операций поиска случайных ключей. Половина ключей присутствовала в таблице, половина - нет. Каждый эксперимент запускался по десять раз с последующим усреднением. Дополнительно проводилось тестирование на «плохих» ключах, которые все попадают в одну ячейку.

Результаты: при низкой загрузке ($\alpha \leq 0.5$) открытая адресация работает быстрее цепочек на 25-30% за счёт лучшей работы кэша. Например, при $\alpha = 0.3$ вставка через открытую адресацию занимала 78 наносекунд, через цепочки - 112 наносекунд. При $\alpha = 0.7$ производительность сравнялась (140-150 наносекунд). При $\alpha = 0.85$ открытая адресация начала тормозить: поиск занял 320 наносекунд против 190 у цепочек. При $\alpha = 0.9$ разрыв стал огромным - 510 против 210 наносекунд (падение на 74%). При $\alpha = 0.95$ открытая адресация практически встала (более 900 наносекунд), цепочки держались на уровне 250 наносекунд. С удалением аналогично: при $\alpha = 0.8$ цепочки удаляли за 180 наносекунд, открытая адресация - за 420. На «плохих» данных цепочки выродились в список и работали линейно: поиск последнего элемента из 10000 занял 18 микросекунд (в 80 раз медленнее нормы). Открытая адресация чувствовала себя немного лучше за счёт проб, но при заполнении таблицы под завязку вставка последнего элемента требовала десятков проб и могла длиться более 100 микросекунд. Качество хеш-функции критически важно для обоих методов [6].

Рекомендации: если таблица не заполнится более чем на 70% и удаления редки, лучше выбрать открытую адресацию с двойным хешированием — она быстрее и экономнее по памяти. Если загрузка может превысить 80%, или удаления часты, или хеш-функция неидеальна - предпочтительнее метод цепочек. В реальных библиотеках (C++ `std::unordered_map`, Python `dict`) часто используют гибридный подход: цепочки, но при превышении длины списка порога (6-8 элементов) список заменяется на сбалансированное дерево. Также обязательно автоматическое расширение таблицы (rehashing) при превышении порога загрузки (0.75 для открытой адресации, 1.0 для цепочек).

Выводы. В ходе работы обнаружено, что для больших и дорого копируемых объектов цепочки предпочтительнее - перемещаются только указатели. Для маленьких данных разница незначительна. В перспективе планируется расширить исследование на другие стратегии проб и на многопоточный режим. Универсального лучшего метода не существует - выбор зависит от конкретных требований к скорости, памяти и устойчивости.

Литература

1. Ахо А., Хопкрофт Дж., Ульман Дж. Структуры данных и алгоритмы. Москва: Вильямс, 2001. 400 с.
2. Вирт Н. Алгоритмы и структуры данных. Москва: ДМК Пресс, 2010. 272 с.
3. Кнут Д. Искусство программирования. Т. 3. Сортировка и поиск. 2-е изд. Москва: Вильямс, 2018. 832 с.
4. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы. Построение и анализ. 3-е изд. Москва: Вильямс, 2013. 1328 с.
5. Седжвик Р. Фундаментальные алгоритмы на C++. Санкт-Петербург: ДиаСофтЮП, 2001. 688 с.
6. Carter J., Wegman M. Universal classes of hash functions // Journal of Computer and System Sciences. 1979. Vol. 18, No. 2. P. 143-154.
7. Knuth D.E. The Art of Computer Programming. Vol. 3: Sorting and searching. 2nd ed. Addison-Wesley, 1998. 800 p.

ANALYSIS OF THE PERFORMANCE OF A HASH TABLE UNDER DIFFERENT COLLISION RESOLUTION STRATEGIES (OPEN ADDRESSING VS CHAINS)

Tarasov Nikita Vladimirovich

Student,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: tnikitav99@gmail.com

Matytsin Artem Vladimirovich

Student,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artiom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. A program has been written that compares two popular methods of resolving collisions in hash tables. These methods are open addressing and the chain method. The program measures the insertion and search times for different levels of table load. It was found that when the table is lightly loaded ($\alpha < 0.5$), open addressing is faster due to the fact that the data is stored consecutively in memory. However, when the table is heavily loaded ($\alpha > 0.8$), chains perform better and do not cause as much slowdown. On 100 runs with $\alpha = 0.9$, open addressing lost to chains in terms of speed, with a difference of approximately 74%.

Keywords: hash table, collisions, chains, open addressing, performance, load factor.

ОСОБЕННОСТИ КЛАСТЕРИЗАЦИИ ДАННЫХ ПРИ АНАЛИЗЕ НАУЧНО-ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ

© Елумеев Дмитрий Андреевич

аспирант

Липецкий государственный педагогический университет

имени П. П. Семенова-Тян-Шанского

Россия, г. Липецк, ул. Ленина, 42

E-mail: 1990_dmitriy@list.ru

© Пруцков Александр Викторович

доктор технических наук, доцент

Рязанский государственный радиотехнический университет имени В. Ф. Уткина

Россия, г. Рязань, ул. Гагарина, 59/1

E-mail: mail@prutzkow.com

Аннотация. Статья посвящена проблеме кластеризации научно-технической документации как инструменту систематизации документальных потоков. Рассматриваются особенности научно-технических текстов, которые определяют требования к методам предварительной обработки и векторизации данных. Проводится анализ применимости различных алгоритмов кластеризации для решения задач тематической группировки документальных массивов

Ключевые слова: научно-техническая документация, кластерный анализ, алгоритмы кластеризации данных.

Реализация положений. Стратегии научно-технологического развития Российской Федерации, а также выполнение мероприятий, предусмотренных Федеральной научно-технической программой развития синхротронных и нейтронных исследований и исследовательской инфраструктуры, определяют необходимость совершенствования подходов к управлению документальными потоками, которые сопровождают процессы получения, оформления и распространения научных знаний [3,4]. Интенсификация исследовательской деятельности в сочетании с усложнением междисциплинарных взаимодействий приводит к значительному росту объемов научно-технической документации, что создает существенные трудности при обеспечении оперативного доступа к необходимым сведениям, упорядочивании накопленных материалов и определении перспективных направлений дальнейших исследований. В связи с этим особую значимость приобретает проблема разработки и последующей адаптации методов кластеризации, учитывающих особенности научно-технических текстов, к числу которых относятся сложная терминологическая структура, присутствие формализованных описательных элементов и строгая жанровая определенность. Применение алгоритмов кластерного анализа позволяет не только распределять документальные массивы по тематическим группам, но и выявлять скрытые связи между документами, что создает основу для формирования целостного представления о текущем состоянии и направлениях развития конкретных научных областей.

Основная сложность при реализации кластерного анализа применительно к массивам научно-технической документации обусловлена специфическими характеристиками обрабатываемых текстовых материалов, что, в свою очередь, формирует повышенные требования к процедурам их предварительной подготовки и последующего описания. Научно-технические документы отличаются жесткой жанровой детерминированностью, высокой концентрацией профильной терминологии и присутствием унифицированных структурных компонентов, к числу которых относятся, в частности, паспортные описания изделий, патентные формулы либо реферативные сводки. В отличие от анализа текстов общепотребительной направленности, в рамках которого допустимо применение универсальных подходов к лемматизации либо векторизации, работа с документацией предполагает строгую регламентацию операций по приведению словоформ к нормативной основе, элиминации стоп-элементов без утраты терминологической значимости, а также выбор такого метода векторизации, который обеспечивает адекватную репрезентацию семантики соответствующей предметной области. Существенное внимание уделяется редукции размерности признакового пространства, поскольку повышенная плотность терминологических единиц способна продуцировать зашумление кластерных структур и осложнять их корректную интерпретацию. Помимо этого, итогом кластеризации должны выступать не исключительно статистически валидизированные группы, но и содержательно интерпретируемые тематические объединения, каждое из которых характеризуется совокупностью ключевых термов, обеспечивающих идентификацию его соотнесенности с конкретными научными направлениями либо техническими решениями. Из вышеизложенного следует, что кластерный анализ в рассматриваемой области представляет собой не столько автоматизированную процедуру группировки, сколько экспертно-аналитическую деятельность, интегрирующую учет жанровых и терминологических особенностей документальных источников.

Для обеспечения эффективности процедур кластеризации применительно к массивам научно-технической документации исходные текстовые материалы должны соответствовать ряду формальных и содержательных требований, которые выступают необходимым условием корректности последующей алгоритмической обработки и достоверности получаемых результатов [1].

На этапе предварительной подготовки обязательным требованием выступает унификация форматов представления данных, предполагающая приведение документов к единой схеме разметки, стандартизированному способу кодировки символов и однородной структуре метаданных, что обеспечивает корректную загрузку материалов в аналитическую среду и устраняет технические барьеры при автоматизированном извлечении информации.

К числу содержательных требований относится смысловая целостность документальных единиц, каждая из которых должна обладать достаточной информационной полнотой для идентификации тематической принадлежности без привлечения внешних источников, а также терминологическая определенность, предполагающая использование устоявшейся научно-технической лек-

сики и однозначное толкование специальных понятий в пределах анализируемого массива.

Важным требованием является жанровая однородность или, как минимум, явная фиксация жанровой принадлежности каждого документа в метаданных, поскольку различные виды научно-технической документации (патентные описания, научные статьи, технические отчеты, конструкторская документация) обладают несовпадающими структурно-композиционными особенностями, игнорирование которых способно привести к формированию кластеров, отражающих не тематическую близость, а жанровые различия материалов.

Требование к наличию метаданных предполагает обязательное сопровождение каждого документа сведениями о дате создания, авторе, источнике публикации и иных реквизитах, что позволяет в дальнейшем проводить стратификационный анализ сформированных кластеров и интерпретировать их с учетом хронологических, авторских или институциональных факторов.

В части лексического состава выдвигается требование терминологической чистоты, минимизирующей присутствие избыточных языковых конструкций, риторических оборотов и субъективно-оценочных суждений, не несущих смысловой нагрузки, поскольку подобные элементы создают шумовые эффекты и затрудняют выделение ключевых понятий, определяющих тематическую направленность документов.

Кроме того, для обеспечения воспроизводимости результатов кластерного анализа требуется документальная фиксация всех процедур предобработки, включая примененные методы токенизации, лемматизации, удаления стоп-слов и выбора параметров векторизации, что создает основу для верификации полученных кластерных решений и их сопоставления с результатами альтернативных методов группировки документальных массивов.

Обратимся к рассмотрению основных алгоритмов кластеризации данных, которые находят применение при обработке документальных массивов, и дадим оценку их применимости в контексте анализа научно-технической документации [2]:

- Иерархические алгоритмы демонстрируют высокую степень релевантности для решения указанных задач, поскольку обеспечивают возможность визуализации структуры документального массива в форме дендрограммы и не требуют априорного задания числа кластеров. Вместе с тем их вычислительная сложность накладывает существенные ограничения при обработке сверхбольших объемов данных.

- Алгоритмы, базирующиеся на понятии центроидов (методы k-средних и k-медоидов), характеризуются высокой скоростью работы и хорошей масштабируемостью применительно к крупным документальным массивам, однако их использование сопряжено с необходимостью предварительного определения числа кластеров, а также с чувствительностью к выбору начальных условий и наличию выбросов.

- Алгоритмы, основанные на оценке плотности распределения данных (DBSCAN, OPTICS), представляются эффективными при работе с зашумленными массивами и позволяют выделять кластеры произвольной формы, что со-

ответствует особенностям реальных документальных совокупностей. При этом настройка параметров плотности может вызывать затруднения в случаях неравномерного распределения терминологических единиц.

- Алгоритмы, опирающиеся на аппарат теории графов, обеспечивают наглядную репрезентацию связей между отдельными документами и способствуют выявлению слабосвязанных тематических групп. Ограничения данного подхода связаны с высокой вычислительной сложностью построения графовых структур для больших массивов и необходимостью задания пороговых значений близости.

- Нечеткие алгоритмы (в частности, метод с-средних) представляют особую ценность для анализа научно-технической документации, поскольку допускают частичную принадлежность документов к нескольким тематическим направлениям, что отражает междисциплинарный характер современных исследований. Вместе с тем интерпретация получаемых результатов требует дополнительных усилий по пороговой обработке коэффициентов принадлежности.

- Алгоритмы, использующие векторные представления и эмбединги, демонстрируют наибольшую перспективность благодаря способности учитывать контекстную семантику терминологических единиц и нивелировать проблему многозначности научно-технической лексики. Применимость данных методов ограничена необходимостью наличия обучающих выборок либо предобученных моделей, адаптированных к конкретной предметной области, а также повышенными требованиями к вычислительным ресурсам.

Выполненное исследование дает основания рассматривать кластеризацию научно-технической документации как одну из задач, отличающихся значительной сложностью и многогранностью, успешность решения которой напрямую зависит от принятия во внимание жанровой специфики текстов, показателей их терминологической плотности, а также степени структурной гетерогенности исходных материалов. Зафиксированные в процессе анализа характеристики документальных совокупностей, включая присутствие унифицированных описательных фрагментов и повышенную концентрацию узкоспециализированных лексических единиц, формируют запрос на создание и внедрение специализированных методик, ориентированных на этапы предобработки информации, подбор способов преобразования текстов в векторный формат и дальнейшую смысловую интерпретацию итоговых кластерных образований. Охарактеризованные в настоящей работе алгоритмы кластеризации проявляют неодинаковые сильные и слабые стороны в контексте их использования для анализа документальных массивов научно-технической направленности: иерархические методы обеспечивают наглядность отображения структуры изучаемой совокупности, центроидные алгоритмы демонстрируют высокие показатели быстродействия при работе с крупными массивами данных, плотностные подходы показывают свою состоятельность в условиях присутствия шумовых компонентов, методы, базирующиеся на теории графов, способствуют идентификации латентных взаимосвязей между документальными единицами, алгоритмы нечеткой кластеризации принимают во внимание потенциальную принадлежность документа одновременно к нескольким тематическим рубрикам, а ме-

тоды, опирающиеся на векторные представления, предоставляют инструментарий для преодоления трудностей, связанных с многозначностью терминологического аппарата.

Проведенный анализ позволяет сделать вывод, что кластеризация научно-технической документации представляет собой сложную и многогранную задачу, для решения которой необходимо принимать во внимание жанровые особенности текстов, уровень их терминологической насыщенности, а также неоднородность внутреннего строения исходных материалов. Выявленные в ходе исследования характеристики документальных массивов, в частности присутствие унифицированных описательных фрагментов и высокая концентрация специальной лексики, определяют потребность в разработке специализированных методов, направленных на предварительную обработку данных, выбор способов преобразования текстов в векторную форму и последующую смысловую интерпретацию получаемых кластерных структур. Рассмотренные в работе алгоритмы кластеризации обладают разными преимуществами и ограничениями при их применении для анализа научно-технической документации: иерархические методы дают возможность наглядно отобразить строение документального массива, алгоритмы центроидного типа обеспечивают высокую скорость обработки значительных объемов информации, плотностные подходы показывают хорошие результаты при работе с зашумленными данными, графовые методы позволяют выявлять скрытые взаимосвязи между документами, нечеткие алгоритмы учитывают возможность принадлежности документа сразу к нескольким тематическим категориям, а методы, основанные на векторных представлениях, дают возможность преодолевать трудности, связанные с многозначностью терминологических единиц.

Литература

1. Барсегян А. А., Куприянов М. С., Степаненко В. В., Холод И. И. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP. 2-е изд., перераб. и доп. Санкт-Петербург: БХВ-Петербург, 2007.
2. Ким Дж.-О., Мьюллер Ч. У., Клекка У. Р. и др. Факторный, дискриминантный и кластерный анализ : пер. с англ. / под ред. И.С. Енюкова. Москва: Финансы и статистика, 1989.
3. Указ Президента Российской Федерации от 28.02.2024 г. № 145 "О Стратегии научно-технологического развития Российской Федерации" // Президент России : [сайт]. URL: <http://www.kremlin.ru/acts/bank/50358>.
4. Федеральная научно-техническая программа развития синхротронных и нейтронных исследований и исследовательской инфраструктуры на период до 2030 года и дальнейшую перспективу // Министерство науки и высшего образования РФ: [сайт]. URL: <https://sn.ntr.ru/>.

FEATURES OF DATA CLUSTERING IN THE ANALYSIS OF SCIENTIFIC AND TECHNICAL DOCUMENTATION

Dmitry Andreevich Yelumeev

Postgraduate Student

Lipetsk State Pedagogical University named after P.P. Semenov-Tyan-Shansky

Russia, Lipetsk, 42 Lenina Street

E-mail: 1990_dmitriy@list.ru

Alexander Viktorovich Prutskov
Doctor of Technical Sciences, Associate Professor
Ryazan State Radio Engineering University
named after V.F. Utkin
Russia, Ryazan, Gagarin Street, 59/1

Abstract. The article is devoted to the problem of clustering scientific and technical documentation as a tool for systematizing document flows. The article considers the features of scientific and technical texts, which determine the requirements for methods of pre-processing and vectorization of data. The article analyzes the applicability of various clustering algorithms for solving the problems of thematic grouping of document arrays.

Keywords: scientific and technical documentation, cluster analysis, data clustering algorithms.

СОЦИАЛЬНАЯ ИНЖЕНЕРИЯ КАК ОСНОВНОЙ ВЕКТОР КИБЕРАТАК: АНАЛИЗ МЕТОДОВ ЗАЩИТЫ

© **Бачевский Дмитрий Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: bachevskiy07@mail.ru

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье рассматривается социальная инженерия как один из самых действенных инструментов современных кибератак. Разобраны ключевые формы манипуляции людьми, выяснены причины высокой результативности таких атак, предложены способы защиты (организационные, технические, правовые). Отдельно отмечено: в 2025–2026 гг. злоумышленники уже не ограничиваются классическим психологическим давлением — они активно внедряют ИИ (автоматизированную персонализацию рассылок, подделку голоса). Из этого следует вывод: одних технических барьеров недостаточно — необходимы ещё и устойчивые поведенческие модели пользователей.

Ключевые слова: вишинг, защита информации, информационная безопасность, кибербезопасность, социальная инженерия, фишинг, человеческий фактор.

В современной цифровой среде атаки всё чаще направлены не на программный код как таковой, а на человека, который принимает решение и обладает доступом к данным или корпоративным системам. Именно поэтому социальная инженерия сохраняет значение одного из ведущих векторов компрометации. Злоумышленник не столько «взламывает» инфраструктуру, сколько побуждает пользователя самостоятельно совершить опасное действие (открыть вложение, перейти по ссылке, сообщить пароль, подтвердить операцию). Описываемый подход особенно опасен, поскольку он обходит многие традиционные защитные механизмы, если пользователь воспринимает запрос как легитимный.

Актуальность обсуждаемой темы усиливается тем, что в российской среде угроз рассматриваемая инженерия стабильно входит в число наиболее часто используемых способов первичного проникновения. Одновременно с этим, имеют место улучшение качества фишинговых сценариев, усложнение схем телефонного мошенничества, более активное задействование генеративного

ИИ при создании убедительных сообщений и голосовых подделок [1]. В связи с этим задача анализа методов защиты приобретает как прикладное, так и научное значение, поскольку требует переосмысления самого понятия защищённости в реалиях доминирования человеческого фактора.

Объект и предмет исследования. Объектом исследования являются кибератаки с использованием методов социальной инженерии. Предметом исследования выступают методы и приёмы социальной инженерии, а также способы защиты от них.

Цель и задачи. Цель работы — анализ социальной инженерии как основного вектора кибератак и выработка рекомендаций по защите. Задачи:

1. Рассмотреть основные методы социальной инженерии.
2. Проанализировать причины их высокой эффективности.
3. Изучить современные тенденции (использование ИИ).
4. Предложить многоуровневую модель защиты.

Методология и методы исследования. В работе использованы методы анализа научной литературы, классификации видов социальной инженерии, синтеза и обобщения данных о кибератаках, а также сравнительный анализ методов защиты.

Разработанность темы. Вопросы социальной инженерии освещены в работах Гончаровой Ю.Л., Гордиенко В.В., Маркелова В.К. и др. Однако практические рекомендации с учётом использования ИИ в атаках 2025-2026 гг. представлены недостаточно.

Социальную инженерию можно определить как совокупность приёмов психологического воздействия, направленных на получение конфиденциальной информации, неправомерного доступа к финансам либо контроля над информационной системой через манипуляцию человеком.

По существу, злоумышленник работает не с уязвимостью программы, а с изъянами восприятия.

Наиболее распространённой формой остаётся фишинг, в том числе, его персонализированный вариант, при котором сообщение адаптируется под должность, круг контактов, профессиональный контекст жертвы [2]. Весомое место занимает и вишинг. Речь идёт об атаках по телефону или через мессенджеры, где ключевую роль играет имитация официального статуса:

- сотрудника банка;
- правоохранительных органов;
- службы безопасности;
- руководства компании [6].

Помимо этого, применяются претекстинг, бейтинг, атаки типа watering hole, когда злоумышленник заранее создаёт правдоподобный сценарий общения, провоцирует жертву на действие через приманки, компрометирует привычный для целевой аудитории веб-ресурс [3, 5].

Начиная с 2024-2025 годов особую тревогу вызывает сочетание социальной инженерии с технологиями искусственного интеллекта. Исследования показывают, что ИИ задействуется в целях генерации правдоподобных писем, языковой адаптации сообщений. Также имитируются голоса, что заметно повышает

убедительность атаки. Тем самым барьер между массовым мошенничеством и адресной высокоточной атакой становится всё менее заметным [1].

В таблице 1 представлено сопоставление основных разновидностей социально-инженерных атак по ключевым параметрам.

Таблица 1

Сравнительный анализ методов социальной инженерии

Метод	Канал воздействия	Основной психологический механизм	Цель злоумышленника
Фишинг	Электронная почта, сайты	Срочность, доверие	Кража учётных данных
Вишинг	Телефон, мессенджеры	Авторитет, страх	Получение кодов, денег, персональных сведений
Претекстинг	Почта, звонки, переписка	Доверие к роли	Доступ к внутренней информации
Бейтинг	Носители, ссылки, «выгодные» предложения	Любопытство	Заражение устройства или сбор данных
Watering hole	Посещаемые сайты	Привычка, доверие к среде	Компрометация целевой группы

Защита в рассматриваемой области не может быть сведена к установке антивируса либо почтового фильтра, поскольку источник риска находится на стыке технологий, организационного поведения, управленческих процедур. По этой причине наиболее обоснованной представляется многоуровневая модель. В ней технические и нетехнические меры дополняют друг друга. Так, на техническом уровне базовое значение сохраняют:

- многофакторная аутентификация;
- фильтрация почтового трафика;
- анализ вложений в изолированной среде;
- поведенческий мониторинг действий пользователей.

Разумеется, ни одно из перечисленных средств не даёт абсолютной гарантии. Но их ценность в другом: они повышают шансы заметить атаку на ранней стадии и уменьшают ущерб, если пользователь уже скомпрометирован. Причём вот что важно: без принципа минимальных привилегий и сегментации сети один обманутый сотрудник способен открыть злоумышленнику доступ ко всей корпоративной инфраструктуре. Это недопустимо.

Анализ реальных инцидентов показывает: именно организационные меры выходят на первый план. Проверено на практике: короткие, но регулярные тренировки персонала, учебные фишинговые рассылки, привычка перепроверять любой подозрительный запрос — это даёт куда более устойчивый результат, чем редкие формальные инструктажи [4; 7]. От сотрудника требуется не просто знать слово «фишинг». Он должен на автоматическом уровне замечать

странные формулировки, искусственное давление, просьбы обойти обычный порядок согласования, попытки увести общение в незащищённый канал. А если учесть дипфейки и фальшивые голосовые сообщения — вывод один: любое финансовое или юридически значимое действие должно проходить двойную верификацию через другой канал связи.

В практическом плане видится целесообразным предложить следующие меры:

- ввести непрерывную программу обучения сотрудников с короткими регулярными модулями вместо редких формальных инструктажей;
- закрепить обязательную двухканальную верификацию для переводов средств, изменения реквизитов, передачи чувствительной информации;
- расширить применение поведенческой аналитики и мониторинга аномальных действий в корпоративных системах;
- проводить внешние и внутренние тестирования с моделированием социально-инженерных сценариев, чтобы оценивать не декларируемую, а фактическую устойчивость организации;
- формировать среду, в которой сотрудник может быстро сообщить о подозрительном контакте без страха дисциплинарных последствий.

Новизна перечисленных рекомендаций состоит не в изолированном наборе мер, а в их сочетании в единую систему, где обучение, процессы, технологии работают синхронно. По существу, именно такая модель позволяет уменьшить разрыв между формальной защищённостью и реальной способностью персонала противостоять манипуляции.

Выводы. Проведённый анализ показывает, что социальная инженерия по праву рассматривается как один из основных векторов современных кибератак. Она воздействует на наиболее изменчивый и трудно стандартизируемый элемент системы безопасности – человека. В чём же отличие от технических атак? Анализируемые атаки делают упор на обычные поведенческие реакции: доверие, страх ошибки, спешка. Следовательно, результативность социнженерии остаётся высокой, даже если у организации есть мощная технологическая защита.

Неверно рассматривать социнженерию как угрозу, против которой невозможно выстроить рациональную защиту. Наибольший эффект достигается там, где технические средства дополняются регулярным обучением, организационными процедурами проверки и управленческой дисциплиной. В 2026 году необходимо учитывать усиление роли искусственного интеллекта в подготовке атак. ИИ повышает качество подделки коммуникаций и маскирует заметность мошеннических признаков. Дальнейшее развитие научных и прикладных подходов должно быть направлено на создание адаптивных моделей защиты. В таких моделях ключевым объектом внимания становится не только инфраструктура, но и поведение пользователя как центральный элемент цифровой безопасности.

Литература

1. Авезова Я., Резников Р., Беседина В. CODE RED 2026: Актуальные киберугрозы для российских организаций. URL: <https://ptsecurity.com/research/analytics/russia-cyberthreat-landscape-2026/#id1> (дата обращения: 25.04.2026).
2. Волокитина Т. С., Таныгин М. О. Методология оценки эффективности систем защиты от фишинга на основе когнитивного моделирования // Психология. Историко-критические обзоры и современные исследования. 2025. Т. 14, № 7-1. С. 105-112.
3. Гончарова Ю.Л., Пехота Ю.С. Социальная инженерия в контексте информационной безопасности: понятие, виды, значение // Следственная деятельность. 2021. № 1. С. 78-90.
4. Гордиенко В. В., Жданов Д. М. Методы защиты от социальной инженерии и фишинга. их достоинства и недостатки // Auditorium. 2024. № 2(42). С. 45-49.
5. Маркелов В.К., Привалов А.Н. Метод претекстинга как угроза информационной безопасности в социальных сетях // Вестник Адыгейского государственного университета. Естественно-математические и технические науки. 2024. № 2(341). С. 51-61.
6. Мартынова И.А. Фишинг, смишинг и вишинг: этимологические и структурно-семантические особенности терминов // Наука XXI века: актуальные направления развития. 2024. № 1-1. С. 294-297.
7. Певзнер А.Д., Бирих Э.В. Основные методы социальной инженерии в IT-сфере // Актуальные проблемы социально-гуманитарного и научно-технического знания. 2025. № 1 (41). С. 11-13.

SOCIAL ENGINEERING AS A MAIN VECTOR OF CYBERATTACKS: ANALYSIS OF PROTECTION METHODS

Bachevsky Dmitry Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: bachevskiy07@mail.ru

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artiom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article discusses social engineering as one of the most effective tools of modern cyberattacks. It analyzes key forms of human manipulation, identifies the reasons for the high effectiveness of such attacks, and proposes ways to protect against them (organizational, technical, and legal). The article also highlights that in 2025-2026, cybercriminals are no longer limited to traditional psychological pressure; instead, they are actively using AI (automated personalization of email campaigns and voice manipulation). This leads to the conclusion that technical barriers alone are not enough; sustainable user behavior patterns are also necessary.

Keywords: vishing, information protection, information security, cyber security, social engineering, phishing, human factor.

ХЕШ-ТАБЛИЦЫ: ОТКРЫТАЯ АДРЕСАЦИЯ И МЕТОД ЦЕПОЧЕК. СРАВНИТЕЛЬНЫЙ АНАЛИЗ И РЕКОМЕНДАЦИИ ПО ВЫБОРУ СХЕМЫ РАЗРЕШЕНИЯ КОЛЛИЗИЙ

© Тимофеев Александр Сергеевич

студент

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: kolvish06@gmail.com

© Матыцин Артем Владимирович

студент

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© Гаев Леонид Витальевич

кандидат технических наук, доцент

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье разобраны два основных способа борьбы с коллизиями в хеш-таблицах: метод цепочек и открытая адресация. Для каждого способа описан принцип работы и показано, как ведут себя операции поиска, вставки и удаления. Оценивается, как скорость работы зависит от заполненности таблицы и того, насколько удачно подобрана хеш-функция. Разбирается, сколько памяти требуют оба подхода, как они работают с кешем процессора и насколько сложны в реализации. По итогам сравнения даны рекомендации, какой способ лучше подходит для хранилищ с частыми удалениями, нагруженных кешей и потокобезопасных структур. Статья будет полезна студентам курса анализа алгоритмов и разработчикам, которые работают со стандартными контейнерами на базе хеширования.

Ключевые слова: хеш-таблица, хеш-функция, коллизия, метод цепочек, открытая адресация, фактор загрузки, амортизированная сложность.

Введение. Хеш-таблица одна из самых востребованных структур данных. Словари в Python, HashMap в Java, std::unordered_map в C++ устроены вокруг одной идеи: ключ переводится в индекс массива через хеш-функцию, и доступ к значению занимает в среднем $O(1)$. На словах звучит просто, но как только дело доходит до реализации, появляется набор развилок. Главная из них - что делать с коллизиями. Двум разным ключам хеш-функция рано или поздно даст один и тот же индекс, и нужен механизм, как уживить их в одной ячейке.

Актуальность. Хеш-таблицы являются одной из самых востребованных структур данных в современном программировании. Выбор схемы разрешения коллизий напрямую влияет на производительность приложений, особенно при работе с большими объёмами данных и в высоконагруженных системах. Прак-

тические рекомендации по выбору между методом цепочек и открытой адресацией являются актуальной задачей.

Объект и предмет исследования. Объектом исследования являются хеш-таблицы как структуры данных. Предметом исследования выступают две стратегии разрешения коллизий: метод цепочек и открытая адресация.

Цель и задачи. Цель работы — сравнительный анализ метода цепочек и открытой адресации в хеш-таблицах, а также выработка рекомендаций по выбору схемы разрешения коллизий. Задачи:

1. Описать принципы работы метода цепочек и открытой адресации.
2. Сравнить их по ключевым характеристикам (скорость, память, сложность реализации).
3. Проанализировать зависимость производительности от фактора загрузки.
4. Сформулировать рекомендации по выбору схемы для различных сценариев использования.

Методология и методы исследования. В работе использованы методы теоретического анализа алгоритмов, сравнительного анализа и обобщения практического опыта реализации хеш-таблиц в современных языках программирования.

Разработанность темы. Вопросы хеширования и разрешения коллизий подробно рассмотрены в работах Кормена Т., Седжвика Р., Кнута Д. и др. Однако практические рекомендации с учётом современных аппаратных особенностей сохраняют актуальность.

Со временем сложились два основных подхода: метод цепочек и открытая адресация. В учебных курсах их обычно дают вместе и отмечают, что оба рабочие, а дальше студент разбирается сам. В этой статье показано, чем они отличаются на практике и как выбор между ними влияет на скорость, расход памяти, поведение при высокой заполненности и удобство поддержки кода.

Метод цепочек. В этом подходе каждая ячейка таблицы хранит не один элемент, а связный список (или иную динамическую структуру). При коллизии новый элемент дописывается в этот список. Поиск состоит из двух шагов: вычислить хеш, добраться до соответствующей ячейки и пройти по списку до сравнения с искомым ключом. Среднее время операции при равномерной хеш-функции равно $O(1 + \alpha)$. Если фактор загрузки невелик, цепочки короткие и операции выполняются быстро; при высоких α поиск вырождается в линейный по длине списка [2].

К сильным сторонам метода относятся простота реализации, тривиальное удаление (элемент извлекается из списка), способность работать при α больше единицы и устойчивость к умеренно неудачному распределению хешей. Слабые стороны очевидны: на каждый узел тратится дополнительная память на указатели, элементы лежат в произвольных областях памяти, что плохо влияет на кеш-локальность, а в худшем случае одна цепочка может вырасти до n элементов и поиск становится $O(n)$.

Открытая адресация. В этой схеме все элементы хранятся непосредственно в массиве. Если ячейка с индексом $h(k)$ уже занята, поиск свободного места ведётся по последовательности проб. Наиболее распространены три варианта последовательностей.

Линейное пробирование. Просматриваются ячейки $h(k)$, $h(k)+1$, $h(k)+2$ и так далее по модулю m . Схема предельно проста, отлично работает с кешем процессора, но страдает от первичного кластеризования: занятые ячейки слипаются в длинные блоки и удлиняют последующие пробы.

Квадратичное пробирование. Пробы идут по правилу $h(k)$, $h(k)+1$, $h(k)+4$, $h(k)+9$ и т. д. Длинные кластеры рассасываются, но возникает вторичное кластеризование: ключи с одинаковым начальным хешем порождают одну и ту же последовательность проб.

Двойное хеширование. Шаг между пробами вычисляется второй хеш-функцией: $h1(k)$, $h1(k) + h2(k)$, $h1(k) + 2 \cdot h2(k)$ и т. д. Поведение приближается к идеально равномерному, кластеризование почти не наблюдается.

Среднее количество обращений к ячейкам при удачном поиске, если пробы распределены равномерно, равно $(1/\alpha) \cdot \ln(1/(1-\alpha))$ [1]. Формула абстрактная, но вывод простой: чем ближе α к единице, тем быстрее растёт время поиска, причём нелинейно. Уже при α около 0,9 поиск заметно дороже, чем при $\alpha = 0,5$. Поэтому в открытой адресации нужно следить за заполненностью и вовремя делать рехеширование, то есть создавать массив большего размера и переносить в него все элементы. Обычно это делается, когда α превышает 0,7-0,75.

Удаление в открытой адресации требует отдельного внимания. Простая очистка ячейки нарушает работу поиска: при чтении элемент, лежащий далее по последовательности проб, может быть пропущен. Поэтому применяются либо специальные пометки (надгробия, tombstones), либо локальная перестановка соседей при удалении. Оба варианта усложняют код и требуют периодической чистки таблицы от накопленных надгробий.

Сравнительный анализ. Рассматриваемые схемы заметно различаются по ключевым характеристикам. При высоких значениях α метод цепочек продолжает работать пусть и медленно, тогда как открытая адресация деградирует резко. Цепочки допускают α больше единицы, открытая адресация требует значения существенно меньше единицы. По расходу памяти метод цепочек проигрывает за счёт указателей в каждом узле, а открытая адресация выигрывает плотным размещением данных в одном массиве. Кеш-локальность у открытой адресации высокая, а у метода цепочек низкая. Удаление проще реализуется в цепочках, в открытой адресации требует надгробий или перетасовки. Сложность реализации у цепочек заметно ниже, а чувствительность к качеству хеш-функции, напротив, выше у открытой адресации.

Если смотреть на типовые нагрузки, картина следующая. Метод цепочек мягко переносит ошибки и сохраняет работоспособность даже при неудачно подобранном размере массива, но уступает в скорости на критичных участках кода: каждое обращение к узлу списка грозит промахом кеша. Открытая адресация работает очень быстро, пока заполненность остаётся умеренной, зато требует аккуратной реализации удаления и расширения таблицы [7].

Рекомендации по выбору. Открытая адресация предпочтительна в случаях, где приоритетом является скорость на чтение и работа с короткими ключами. Современные процессоры за один поход в кеш-линию читают сразу 64 байта, и линейное пробирование укладывается в эти 64 байта почти бесплатно. Допол-

нительные преимущества проявляются при ограниченном числе аллокаций (один большой массив, один вызов выделения памяти) и в сценариях с редкими удалениями: кэши с TTL, разовая загрузка справочника, индексация. Именно поэтому стандартные хеш-карты в Rust и Go построены на открытой адресации; реализация HashMap в Rust базируется на схеме SwissTable от Google [4].

Метод цепочек удобен там, где удаления идут часто и в непредсказуемом порядке, заранее непонятно, сколько будет элементов, а сами ключи и значения занимают много места. В последнем случае плотный массив всё равно не даёт выигрыша: одна запись уже не помещается в кеш-линию. Гибрид из списка и дерева, который используется в Java HashMap, добавляет устойчивость к намеренно подобранным ключам и делает худший случай предсказуемым.

Отдельный класс задач - конкурентные хеш-таблицы. Цепочки удобнее шардировать по бакетам и снабжать локальными блокировками или lock-free операциями. Открытая адресация в многопоточной среде сложнее: любая операция может затрагивать длинный участок массива, и проектирование атомарных протоколов требует существенных усилий [6].

Выводы. Спор о превосходстве цепочек или открытой адресации не решается универсально. Это два разных компромисса, и каждый из них хорош в своих условиях. Метод цепочек - выбор по умолчанию для задач, в которых важны простота реализации, частые удаления и устойчивость к высокому фактору загрузки. Открытая адресация - выбор для скорости, плотного использования памяти и обработки больших объёмов запросов на чтение.

Литература

1. Документация языка Rust: модуль `std::collections::HashMap` // The Rust Standard Library : [сайт]. URL: <https://doc.rust-lang.org/std/collections/struct.HashMap.html>.
2. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ. 3-е изд. Москва: Вильямс, 2013. 1328 с.
3. Седжвик Р., Уэйн К. Алгоритмы на Java. 4-е изд. Москва: Вильямс, 2013. 848 с.
4. Abseil Team. Swiss Tables Design Notes // Abseil: [сайт]. URL: <https://abseil.io/about/design/swisstable>.
5. Knuth D. The Art of Computer Programming. Vol. 3. Sorting and Searching. 2nd ed. Boston: Addison-Wesley, 1998. 780 p.
6. Maier T., Sanders P., Dementiev R. Concurrent Hash Tables: Fast and General? // ACM Transactions on Parallel Computing. 2019. Vol. 5, № 4. P. 1–32.
7. Pagh R., Rodler F. Cuckoo Hashing // Journal of Algorithms. 2004. Vol. 51, № 2. P. 122–144.

HASH TABLES: OPEN ADDRESSING AND CHAIN METHOD. COMPARATIVE ANALYSIS AND RECOMMENDATIONS FOR CHOOSING A COLLISION RESOLUTION SCHEME

Timofeev Alexander Sergeevich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: kolvish06@gmail.com

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article discusses two main methods for dealing with collisions in hash tables: the chaining method and open addressing. For each method, it describes the principle of operation and shows how search, insertion, and deletion operations behave. It also evaluates how the speed of operation depends on the table's size and the effectiveness of the hash function. The article explores the memory requirements of both approaches, their interaction with the processor cache, and their complexity in implementation. Based on the results of the comparison, recommendations are given on which method is better suited for storage with frequent deletions, loaded caches and thread-safe structures. The article will be useful for students of the course of algorithm analysis and developers who work with standard containers based on hashing.

Keywords: hash table, hash function, collision, chain method, open addressing, load factor, amortized complexity.

МЕТОДЫ АНАЛИЗА ВРЕМЕННОЙ СЛОЖНОСТИ И ОПТИМИЗАЦИИ РЕКУРСИВНЫХ АЛГОРИТМОВ В ЗАДАЧАХ ОБРАБОТКИ ДАННЫХ

© **Селянин Матвей Андреевич**

студент,

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matveiseluanin48@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье рассматриваются подходы к анализу временной сложности рекурсивных алгоритмов, применяемых при обработке структурированных и полуструктурированных данных. Проводится классификация рекурсивных методов по типу ветвления и глубине рекурсии, включая линейную рекурсию, бинарное ветвление и многопутевые рекурсивные обходы. Особое внимание уделяется методам формального анализа вычислительной сложности с использованием аппарата рекуррентных соотношений и метода главной теоремы. Представлены практические способы оптимизации: мемоизация, переход к итеративной реализации и хвостовая рекурсия. В заключении анализируются сценарии, в которых рекурсия оправдана, и ситуации, требующие её замены. Статья предназначена для разработчиков алгоритмического программного обеспечения и студентов, изучающих анализ алгоритмов.

Ключевые слова: рекурсия, временная сложность, рекуррентные соотношения, мемоизация, хвостовая рекурсия, оптимизация.

Введение. В современной разработке алгоритмического программного обеспечения обработка иерархических данных, обход деревьев, вычисление комбинаторных объектов и реализация принципа «разделяй и властвуй» практически всегда связаны с использованием рекурсивных алгоритмов [1]. При этом корректно работающая рекурсия на небольших объёмах данных может полностью деградировать по производительности при масштабировании. Причина кроется не в самом принципе рекурсии, а в экспоненциальном или квадратичном росте числа вызовов при неоптимальной структуре [2]. Задача разработчика — не только реализовать алгоритм, но и предсказать его поведение на больших входных данных. Цель данной статьи — систематизировать методы анализа временной сложности рекурсивных алгоритмов, описать основные способы их оптимизации и предложить критерии выбора между рекурсивной и итеративной реализациями.

Актуальность. Рекурсивные алгоритмы широко применяются при обработке иерархических данных, однако неоптимальная рекурсия может приводить к экспоненциальному росту времени выполнения и переполнению стека. В условиях роста объемов обрабатываемых данных анализ сложности и оптимизация рекурсивных алгоритмов становятся критически важными для разработки высоконагруженных систем.

Объект и предмет исследования. Объектом исследования являются рекурсивные алгоритмы обработки данных. Предметом исследования выступают методы анализа временной сложности и оптимизации рекурсивных алгоритмов.

Цель и задачи. Цель работы — систематизация методов анализа временной сложности рекурсивных алгоритмов и описание способов их оптимизации. Задачи:

1. Классифицировать рекурсивные алгоритмы по структуре вызовов.
2. Рассмотреть формальные методы анализа сложности (рекуррентные соотношения, главная теорема).
3. Описать методы оптимизации (мемоизация, хвостовая рекурсия, итеративная замена).
4. Сформулировать критерии выбора между рекурсивной и итеративной реализациями.

Методология и методы исследования. В работе использованы методы теоретического анализа алгоритмов, построения и решения рекуррентных соотношений, а также сравнительного анализа рекурсивных и итеративных реализаций.

Разработанность темы. Вопросы анализа рекурсивных алгоритмов детально рассмотрены в работах Ахо А.В., Кормена Т.Х., Скиены С.С. и др. Однако систематизация методов оптимизации для практических задач разработки остаётся актуальной.

Результаты исследования. Все рекурсивные алгоритмы с точки зрения анализа сложности можно разделить на три основные категории: линейная рекурсия, бинарное ветвление и многопутевая глубокая рекурсия [3].

Суть линейной рекурсии заключается в том, что функция вызывает саму себя строго по одному разу за шаг. Наглядными примерами здесь служат классическое вычисление факториала или последовательный проход по односвязному списку. Количество вызовов в такой схеме формирует временную сложность уровня $O(n)$, при этом стек вызовов заполняется с аналогичной линейной скоростью. Скорость работы в подобных сценариях падает редко, однако ключевая опасность кроется в потенциальном переполнении стека (stack overflow), если значение «n» окажется слишком большим.

При бинарном ветвлении один запуск функции провоцирует сразу два последующих рекурсивных вызова. Эталонный пример - базовый (наивный) алгоритм поиска чисел Фибоначчи. В таком сценарии объем вызовов увеличивается уже по экспоненте, из-за чего код становится фактически неработоспособным при $n > 40$, если его вовремя не оптимизировать. Внешняя лаконичность и изящность этого решения коварны, так как за ними скрывается лавинообразное падение производительности.

С многопутевой глубокой рекурсией приходится сталкиваться в процессе обхода многоуровневых древовидных структур (вроде документов XML или JSON) или при реализации стратегии «разделяй и властвуй», яркими примерами которой выступают сортировка слиянием и быстрая сортировка. Итоговые временные затраты определяются плотностью ветвления и тем, как именно объединяются полученные данные. Если дерево сбалансировано, сложность обычно укладывается в рамки $O(n \log(n))$, но в случае вырожденных структур она возрастает до квадратичной - $O(n^2)$.

Формальный анализ временной сложности. Главным способом оценки таких алгоритмов считается составление и последующий расчет рекуррентных уравнений [3]. Обозначим время работы программы для входного массива данных размера n как $T(n)$. Тогда $T(n)$ выражается через $T(k)$ для меньших размеров. Например, для бинарного поиска: $T(n) = T(n/2) + O(1)$, решением служит $T(n) = O(\log n)$ [1]. Для наивного вычисления чисел Фибоначчи: $T(n) = T(n-1) + T(n-2) + O(1)$, что даёт $T(n) = O(\varphi^n)$, где $\varphi \approx 1.618$.

Чтобы быстро находить ответы для соотношений вида $T(n) = a \cdot T(n/b) + f(n)$ [4], чаще всего прибегают к основной теореме о рекуррентных соотношениях (Master Theorem) [4]. Опираясь на количество подзадач a , шаг уменьшения их объема b и накладные расходы на разделение/слияние $f(n)$, можно сразу вычислить асимптотику, минуя долгое пошаговое развертывание формулы. Так, в алгоритме сортировки слиянием переменные принимают значения $a = 2$, $b = 2$, а $f(n) = O(n)$, что подпадает под второй случай основной теоремы и дает итоговую сложность $T(n) = O(n \log n)$ [2].

Методы оптимизации рекурсивных алгоритмов. Пожалуй, самым гибким и мощным подходом к оптимизации считается мемоизация, то есть сохранение промежуточных ответов в кэш [3]. Суть метода проста: если функция уже вычисляла значение для конкретных входных аргументов, этот результат записывается в память. Применяя этот прием к тем же числам Фибоначчи, мы превращаем экспоненциальный рост сложности в линейный $O(n)$, ведь повторные расчеты полностью исключаются. Обратная сторона медали здесь — повышенные требования к ОЗУ, расходы которой в пике могут составить $O(n)$ [1].

Еще один действенный инструмент — переработка стандартного алгоритма в формат хвостовой рекурсии (tail recursion) [4]. Хвостовой называют такую структуру, где обращение функции к самой себе идет финальным аккордом, и над возвращаемым результатом больше не производится никаких математических или логических действий. Современные компиляторы и интерпретаторы (включая GCC, LLVM и среды исполнения функциональных языков) умеют распознавать этот паттерн и автоматически трансформируют его в обычный цикл, высвобождая стек [2]. Нагляднее всего это видно при расчете факториала через переменную-аккумулятор: финальный вызов закрывает функцию, предотвращая раздувание стека.

Наконец, третьим решением становится тотальный уход от рекурсивных вызовов к классическим итерациям, где вместо системного стека задействуется стек в динамической памяти (куче) [5]. Такой шаг жизненно необходим в средах, где встроенная оптимизация хвостовых вызовов отсутствует (как в ваниль-

ном Python) и для глубоких рекурсий. Программисту приходится брать контроль над стеком в свои руки: это делает расход памяти абсолютно прозрачным, хотя и усложняет архитектуру самого кода.

Практические последствия и критерии выбора. Игнорирование правил оптимизации рекурсии неизбежно оборачивается лавинообразным ростом времени работы, аварийным завершением программ из-за нехватки стека и колоссальной нагрузкой на оперативно-запоминающее устройство [3]. В реальных коммерческих проектах такие просчеты приводят к падению серверов при минимальном увеличении пользовательского трафика, превращают отладку в кошмар и делают поведение системы в продакшене абсолютно непредсказуемым [1].

Принимая решение, писать ли код через циклы или через рекурсию, стоит взвесить несколько ключевых параметров [2]:

Специфика данных: рекурсивный подход органично ложится на графы, деревья и иные фрактальные или рекурсивные сущности (вроде списков в функциональной парадигме). Предельная глубина вложенности: если лимиты вызовов гарантированно укладываются в рамки пары сотен шагов без угрозы аварийной остановки, рекурсию можно смело использовать. Специфика компилятора/интерпретатора: в технологических стеках, аппаратно поддерживающих хвостовую оптимизацию (например, Elixir, Scala или Scheme), рекурсивные конструкции обычно выигрывают у циклов по всем статьям. Прозрачность и простота поддержки: лаконичный рекурсивный обход сложного дерева читается гораздо легче, чем громоздкий итеративный аналог с ручным управлением стеком [4].

Выводы. Резюмируя, можно констатировать: рекурсивные подходы по-прежнему незаменимы при проектировании систем, завязанных на иерархических структурах или стратегиях типа «разделяй и властвуй». Тем не менее, внедрение таких решений вслепую — без предварительного математического расчета рекуррентных формул и игнорируя базовые инструменты вроде кэширования или хвостовой трансформации — гарантирует стремительную потерю вычислительной мощности при увеличении нагрузок [5].

Сегодня маркер профессионализма инженера — это не умение создать рекурсивный вызов как таковой, а готовность строго рассчитать вектор роста $T(n)$, вовремя локализовать узкие места (дублирующиеся циклы расчетов, перегрузку памяти) и внедрить верный способ исправления ситуации [1]. На фоне тотального увеличения объемов данных в Highload-системах автоматическая верификация рекурсий и их перевод на контролируемые циклы превращаются в стандарт индустрии разработки [3].

Литература

1. Ахо А.В., Хопкрофт Дж. Э., Ульман Дж. Д. Структуры данных и алгоритмы. Москва: Вильямс, 2016. 400 с.
2. Кормен Т. Х., Лейзерсон Ч. И., Ривест Р. Л., Штайн К. Алгоритмы: построение и анализ. 3-е изд. Москва: Вильямс, 2013. 1328 с.
3. Скиена С. С. Алгоритмы. Руководство по разработке. 2-е изд. Санкт-Петербург: БХВ-Петербург, 2011. 720 с.

4. Тоноян С. А., Балдин А. В., Елисеев Д. В. Прогнозирование технического состояния электронных систем с адаптивными параметрическими моделями // Вестник Московского государственного технического университета им. Н. Э. Баумана. Сер. «Приборостроение». 2016. № 6(111). С. 115-125.

5. Вандер Плас С., Нокс С. Python для сложных задач: наука о данных и машинное обучение. Санкт-Петербург: Питер, 2018. 576 с.

METHODS OF TIME COMPLEXITY ANALYSIS AND OPTIMIZATION OF RECURSIVE ALGORITHMS IN DATA PROCESSING PROBLEMS

Selyanin Matvey Andreevich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matveiseluanin48@gmail.com

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artiom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article discusses approaches to analyzing the time complexity of recursive algorithms used in processing structured and semi-structured data. It classifies recursive methods based on the type of branching and the depth of recursion, including linear recursion, binary branching, and multi-path recursive traversals. The article focuses on formal methods for analyzing computational complexity using the machinery of recurrence relations and the master theorem. It presents practical optimization techniques such as memoization, iterative implementation, and tail recursion. In conclusion, the article analyzes scenarios where recursion is justified and situations that require its replacement. The article is intended for developers of algorithmic software and students studying algorithm analysis.

Keywords: recursion, time complexity, recurrence relations, memoization, tail recursion, optimization.

РАЗРАБОТКА КОНСОЛЬНОЙ УТИЛИТЫ НА C# ДЛЯ СБОРА СИСТЕМНОЙ ИНФОРМАЦИИ В СРЕДЕ WINDOWS

© **Валиуллов Тимур Ильдарович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: timurvaliullov65646@mail.ru

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. Статья описывает консольное приложение на C# для мониторинга системы без лишних зависимостей. WMI служит основным источником диагностики — через него собираются данные о загрузке CPU, использовании RAM и информация о дисках. Для управления процессами задействован System.Diagnostics. Сетевые интерфейсы, напротив, мониторятся без WMI — System.Net.NetworkInformation даёт всю статистику и адреса, работая чище и быстрее. Всё объединено в единое окно в .NET. Отдельного упоминания заслуживает визуализация. Консольные прогресс-бары превращают сухие цифры загрузки CPU и свободного места на дисках в картинку, которая считывается мгновенно. Мелочь, но именно такие детали делают утилиту удобной, а не просто рабочей. На выходе получается компактный, портативный инструмент без внешних библиотек. Разработчику — быстро посмотреть, что происходит с системой в момент отладки. Администратору — провести первичную диагностику, не разворачивая множество вкладок с различными процессами.

Ключевые слова: C#, .NET, мониторинг системы, WMI, диагностика, консольное приложение, загрузка CPU, использование RAM, информация о дисках, сетевые интерфейсы, управление процессами.

Информационные системы требуют постоянного контроля за аппаратным обеспечением. Продвинутым пользователям — администраторам или разработчикам, часто необходимо получать сведения о центральном процессоре, оперативной памяти, свободном месте на дисках, сети и запущенных процессах.

Штатные средства Windows неудобны для быстрой оценки производительности — приходится держать открытыми несколько окон. Предлагаемое консольное приложение на C# под .NET [1, 2, 5, 7] решает эту проблему, выводя критически важную информацию на один экран: модель CPU и его загрузку, объём RAM, состояние файла подкачки, занятость дисков с предупреждением о пере-

полнении, активные сетевые интерфейсы и доступ в интернет. Также реализован список процессов с сортировкой по нагрузке.

В числе ключевых требований к утилите — работа исключительно в среде Windows, использование WMI в качестве основного API, консольный вывод с цветовым кодированием и прогресс-барами, а также учёт возможной недоступности отдельных WMI-классов [4]. Последнее особенно важно: не на всякой машине WMI работает стабильно и в полном объёме, поэтому приложение должно корректно обрабатывать подобные ситуации, а не «ломаться» с исключением.

Актуальность. Быстрый доступ к системной информации необходим администраторам и разработчикам для диагностики и отладки. Штатные средства Windows требуют открытия нескольких окон, что неудобно. Создание компактной консольной утилиты, объединяющей все ключевые показатели в одном окне, является актуальной задачей [3, 4].

Объект и предмет исследования. Объектом исследования являются методы сбора системной информации в ОС Windows. Предметом исследования выступает консольная утилита на C# [1, 2, 8], использующая WMI [4] и System.Diagnostics [5] для мониторинга CPU, RAM, дисков, сети и процессов.

Цель и задачи. Цель работы — разработка консольной утилиты для сбора системной информации в Windows. Задачи:

1. Реализовать сбор данных о процессоре и его загрузке.
2. Реализовать сбор данных об оперативной памяти и файле подкачки.
3. Реализовать сбор данных о дисках.
4. Реализовать мониторинг сетевых интерфейсов.
5. Реализовать отображение информации о процессах.
6. Обеспечить визуализацию через прогресс-бары и цветовое кодирование.

Методология и методы исследования. В работе использованы методы объектно-ориентированного программирования на C#, работа с WMI-запросами [4], использование стандартных библиотек .NET (System.Diagnostics[5], System.IO [3], System.Net.NetworkInformation).

Разработанность темы. Вопросы мониторинга системы через WMI освещены в работах Опенкина Д.Ю. [4] и других авторов. Однако практические реализации компактных консольных утилит с цветовой индикацией и прогресс-барами представлены недостаточно.

Результаты исследования. Приложение разработано в одном файле — Program.cs. Он содержит статистические методы, которые используют директивы #region, что упрощает чтение и в последующем расширяемость кода [7]. При запуске приложения проверяется операционная система: если это не Windows, то происходит вывод сообщения и приложение завершается.

Диагностика дисков. Класс DriveInfo из System.IO [3] работает на разных ОС, в том числе на Windows. Для каждого диска выводится имя, метка, файловая система, общий размер, занятое и свободное пространство и процент заполнения. При превышении 75% выводится предупреждение желтым цветом, при 90% — красным.

Если интернет доступен, для каждого активного интерфейса с IPv4 показываются: имя, тип, скорость, MAC, IP и счётчики переданных байт.

Управление процессами. `DisplayProcessInfo()` – отображает информацию о запущенных процессах через метод `Process.GetProcesses()` [5], фильтруя системные, завершившиеся процессы.

Вспомогательные функции. Для улучшения пользовательского опыта в консоли реализованы:

- `FormatBytes()` — преобразование байтов в КБ, МБ, ГБ, ТБ с двумя знаками дробной части;
- `DrawProgressBar()` — отрисовка горизонтальной шкалы с меткой и процентом;
- `DrawMiniProgressBar()` — компактная шкала (10 символов) для загрузки по ядрам.

Цветовая схема и кодировка UTF-8 задаются в начале работы программы.

Тестирование и результаты. Результат работы утилиты на операционной системе Windows 11 показан на рисунке 1 – с подключением к интернету, на рисунке 2 – без подключения.

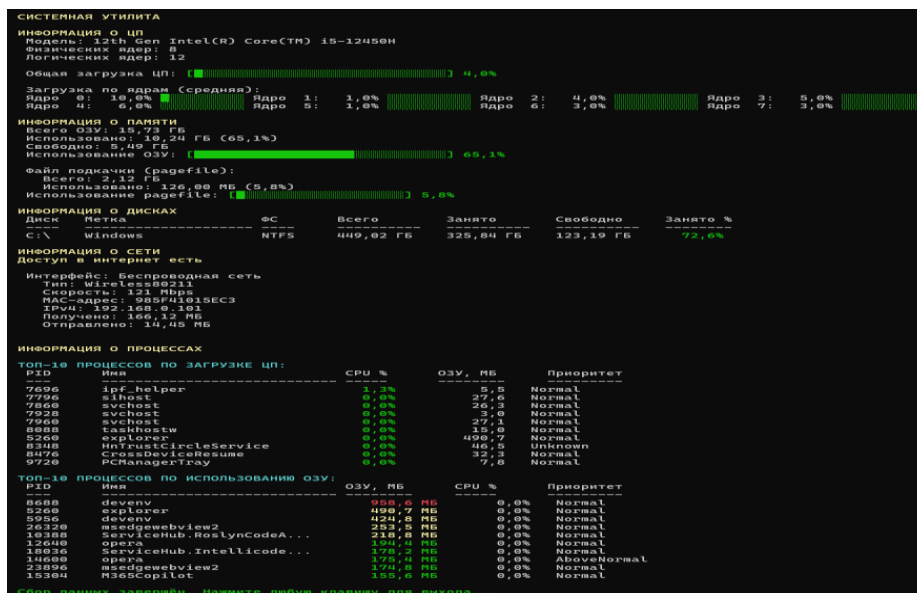


Рисунок 1. Результат выполнения программы с подключенной сетью

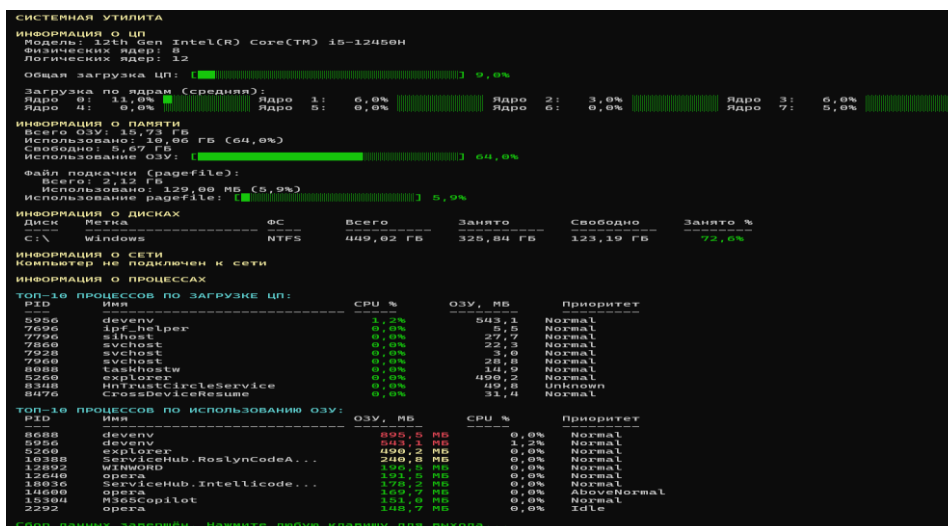


Рисунок 2. Результат выполнения программы без подключения к сети

Определение модели процессора и его загрузки обрабатывает корректно. Полученные значения совпадают с показаниями «Диспетчера задач», расхождение укладывается в 1–2% — погрешность, обусловленную временем опроса, и для задач мониторинга она совершенно не критична. Для наглядного сравнения результат из «Диспетчера задач» приведён на рисунке 3.

Процессы		Запустить новую задачу		Завершить задачу	
Имя	Состояние	6% ЦП	66% Память	1% Диск	0% Сеть
> Opera GX Internet Browser (25)		0%	940,0 МБ	0,2 МБ/с	0 Мбит/с
> Microsoft Visual Studio 2022		0%	469,0 МБ	0 МБ/с	0 Мбит/с
> Microsoft Visual Studio 2022		0%	335,9 МБ	0,1 МБ/с	0 Мбит/с
> Antimalware Service Executable		0%	188,9 МБ	0 МБ/с	0 Мбит/с
ServiceHub.IntellicodeModelS...		0%	151,9 МБ	0 МБ/с	0 Мбит/с
Диспетчер окон рабочего ст...		0,5%	96,8 МБ	0 МБ/с	0 Мбит/с
> Диспетчер задач		0,5%	82,1 МБ	0 МБ/с	0 Мбит/с
Проводник		0%	81,5 МБ	0 МБ/с	0 Мбит/с

Рисунок 3. Результат выполнения диспетчера задач

Память и файл подкачки отображены верно. Цвета индикаторов соответствуют пороговым значениям, а для дисков выводится таблица с выравниванием колонок, выводя предупреждения для заполненных разделов. При отключенном Ethernet или Wi-Fi программа выводит сообщение «Компьютер не подключен к сети» и при реальном подключении показывает IP и статистику.

Информация по процессам выводит топ-10 по каждому критерию, работая с цветовой индексацией.

Выводы. Разработана консольная утилита на C# [1, 2, 7] для сбора системной информации в Windows. Результаты тестирования подтвердили корректность работы. Утилита может использоваться для быстрой диагностики [3, 4].

Литература

1. Албахари Дж., Албахари Б. C# 7.0. Справочник. Полное описание языка. Москва: Вильямс, 2018. 1024 с.
2. Вагнер Б. Эффективное программирование на C#. 50 способов улучшения кода. Москва: ДМК Пресс, 2018. 336 с.
3. Разработка информационной системы диагностики и учёта компьютерного оборудования: выпускная квалификационная работа / Национальный исследовательский университет «МЭИ». Москва, 2021. URL: https://github.com/adzinota/graduation_work (дата обращения: 20.01.2024).
4. Опенкин Д. Ю. Применение инструментария WMI запросов при разработке программного обеспечения // Ученые записки УлГУ. Сер. Математика и информационные технологии. 2019. № 2. С. 76–80. URL: https://www.mathnet.ru/php/archive.phtml?jrnid=ulsu&option_lang=rus&paperid=84&wshow=paper (дата обращения: 20.01.2024).
5. Рихтер Дж. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. 4-е изд. Санкт-Петербург: Питер, 2014. 896 с.

6. Скит Дж. С# для профессионалов: тонкости программирования. 3-е изд. Москва: Вильямс, 2014. 608 с.
7. Троелсен Э., Джепикс Ф. Язык программирования С# 7 и платформы .NET и .NET Core. 8-е изд. Москва: Вильямс, 2018. 1344 с.
8. Шилдт Г. С# 4.0: полное руководство. Москва: Вильямс, 2011. 1056 с.

DEVELOPMENT OF A C# CONSOLE UTILITY FOR COLLECTING SYSTEM INFORMATION IN THE WINDOWS ENVIRONMENT

Valiullov Timur Ildarovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: timurvaliullov65646@mail.ru

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article describes a console application written in C# for system monitoring without unnecessary dependencies. WMI serves as the primary source of diagnostics, collecting data on CPU usage, RAM utilization, and disk information. System.Diagnostics is used for process management. On the other hand, network interfaces are monitored without WMI, as System.Net.NetworkInformation provides all the necessary statistics and addresses, offering a cleaner and faster solution. All these features are integrated into a single .NET window. The visualization deserves special mention. Console progress bars turn dry CPU load and disk space figures into a picture that is read instantly. It's a small detail, but it's these details that make the utility convenient rather than just functional. The result is a compact, portable tool with no external libraries. For developers, it allows them to quickly view what's happening with the system during debugging. For administrators, it provides an initial diagnostic without opening multiple tabs with various processes.

Keywords: C#, .NET, system monitoring, WMI, diagnostics, console application, CPU usage, RAM usage, disk information, network interfaces, process management.

LZW — АЛГОРИТМ СЖАТИЯ ДАННЫХ, ИЗМЕНИВШИЙ ЭПОХУ

© **Раев Николай Николаевич**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: raevnikolay14072006@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В данной статье рассматривается алгоритм LZW (Lempel-Ziv-Welch), созданный в 1984 году как модификация более ранних разработок Лемпеля и Зива [3]. Он стал первой широко распространённой реализацией словарного сжатия, нашёл применение в форматах GIF, TIFF и Unix Compress, и продолжает влиять на современные методы сжатия. Статья описывает принципы работы алгоритма, его развитие от теорий к практической реализации, а также причины его успеха.

Ключевые слова: сжатие данных, LZW, словарное кодирование, алгоритмы без потерь, Лемпель-Зив-Велч.

Введение. Представим, что на дворе 1985 год и нам нужно отправить текстовый файл по модемному соединению того времени. Каждая секунда соединения стоит денег, а повторы в тексте по типу "когда", "что", "не" и так далее увеличивают объём данных. Чтобы решить эту проблему, нужно придумать способ, который позволит не передавать одни и те же последовательности символов по несколько раз.

Актуальность. Алгоритм LZW стал первой широко распространённой реализацией словарного сжатия и до сих пор используется в форматах GIF и TIFF. Анализ его принципов работы важен для понимания эволюции методов сжатия данных и их применения в информационных системах.

Объект и предмет исследования. Объектом исследования являются алгоритмы сжатия данных без потерь. Предметом исследования выступает алгоритм LZW: его принципы работы, практическая реализация и место в семействе алгоритмов Лемпеля-Зива.

Цель и задачи. Цель работы — анализ алгоритма LZW, его принципов работы и практического применения. Задачи:

1. Рассмотреть исторический контекст появления алгоритма.
2. Описать принципы сжатия и распаковки LZW.
3. Сравнить LZW с другими алгоритмами семейства LZ.
4. Проанализировать практическое применение алгоритма.

Методология и методы исследования. В работе использованы методы теоретического анализа алгоритмов, сравнительного анализа и обобщения данных из научной литературы.

Разработанность темы. Вопросы сжатия данных на основе словарного кодирования подробно рассмотрены в работах Лемпеля, Зива, Велча и других авторов. Однако практический анализ LZW с акцентом на его преимущества и недостатки сохраняет актуальность для образовательных целей.

Но это не так просто. Как создать алгоритм, который эффективно находил бы повторы в потоке данных, не требуя передачи дополнительного словаря? Как сделать так, чтобы получатель мог восстановить исходные данные, не зная заранее, какие именно фразы были заменены? И как решить всё это в реальном времени, пока данные поступают последовательно?

Именно эти вопросы стояли перед исследователями в конце 1970-х годов. Эти проблемы помогли решить алгоритмы Лемпеля и Зива [1]. Из всех этих разработок Терри Велч в 1984 году взял за основу LZ78 и довёл его до ума [2]. Так появился LZW. Обычно между открытием алгоритма и его массовым применением проходит много времени, а с LZW вышло иначе: его начали использовать почти сразу после публикации.

Результаты исследования. Чтобы понять то, насколько важен LZW, стоит сначала вспомнить, как люди пытались решать проблемы сжатия до него.

Проблема универсального сжатия. До появления алгоритмов Лемпеля и Зива сжатие работало по-другому. Сначала нужно было полностью изучить файл, посчитать, какие символы встречаются чаще, построить таблицу замен, и только после этого — сжимать. Всё это требовало двух проходов по данным. И если файл был не текстовым, а, например, картинкой, заранее предсказать, какие символы там будут часто встречаться, было почти невозможно.

Лемпель и Зив предложили отказаться от предварительного изучения. Их алгоритмы работали в один проход: анализировали данные по мере поступления, искали повторяющиеся куски в уже прочитанном материале и заменяли их ссылками на прошлое. Самый известный из их методов — LZ77 — использовал так называемое скользящее окно: небольшой буфер памяти, в котором хранились последние прочитанные байты. Как только в новых данных встречался фрагмент, похожий на тот, что уже есть в окне, алгоритм писал вместо этого фрагмента ссылку на позицию в окне. Окно постоянно двигалось — старые данные забывались, новые запоминались.

Год спустя, в 1978 году, Лемпель и Зив представили LZ78 [1]. Вместо скользящего окна этот алгоритм строил словарь фраз, которые инициализировались всеми одиночными символами, а затем пополнялись новыми фрагментами по мере обработки данных. Словарь пополнялся автоматически по ходу работы, и его не нужно было передавать отдельно: декодер сам восстанавливал его, просто распаковывая данные.

От LZ78 к LZW. В теории идея алгоритма LZ78 звучала хорошо, но его реализация была связана с определёнными сложностями. Словарь, который строила эта схема, мог расти бесконтрольно. Кроме того, управление словарём требовало аккуратной реализации для обеспечения производительности.

В 1984 году Терри Велч опубликовал описание модификации LZ78, которая устранила ряд практических проблем и сделала алгоритм таким, что он работал даже на простых микросхемах, без сложного софта [2]. Новая схема LZW была проще в реализации, эффективнее использовала память и, что очень важно для внедрения, хорошо работала на ограниченных вычислительных ресурсах того времени.

Именно LZW стал первым широко используемым методом универсального сжатия данных на компьютерах.

Принцип работы алгоритма LZW. Основная идея заключается в том, что LZW сжимает данные за счёт того, что запоминает часто встречающиеся куски текста в словарь. Каждому такому куску присваивается числовой код. Если один и тот же фрагмент встречается снова, алгоритм уже не пишет его целиком, а просто вставляет его код из словаря.

Ключевая инновация заключается в том, что словарь строится синхронно как у того кто отправляет, так и у того, кто распаковывает. Отправитель никогда не передаёт сам словарь — он передаёт только коды, а получатель восстанавливает тот же словарь по ходу декодирования.

Алгоритм сжатия. Перед началом работы словарь уже содержит все возможные одиночные символы. Если мы работаем с обычным 8-битным текстом, это 256 записей.

Если каждый символ кодировался 8 битами (56 бит), а коды — тоже 8 битами (32 бита), то сжатие составило около 43%. На более показательных примерах с большим количеством повторяющихся паттернов эффективность выше.

Процесс распаковки. Декодер начинает с идентичного начального словаря. При получении кода он извлекает соответствующую строку, выводит её и обновляет словарь точно так же, как это делал кодер.

Иногда распаковщик получает код, которого ещё нет в его словаре. Это случается, когда сжимаемый фрагмент устроен как "фраза, к которой добавили её первый символ". В такой ситуации распаковщик сам догадывается, что нужно взять предыдущую фразу и приписать к ней её первый символ. Например, если предыдущая фраза "AB", а новый код означает "ABA", распаковщик просто берёт "AB" и добавляет "A".

LZW в контексте семейства LZ. Чтобы понять, какое место занимает LZW среди родственных алгоритмов, полезно сравнить его с самым известным из семейства — LZ77.

LZ77 использует скользящее окно. Он хранит буфер уже обработанных данных (обычно размером несколько килобайт) и когда обрабатывает очередной фрагмент, пытается найти его в этом буфере. Причём ищет не любой, а самый длинный возможный кусок, который там есть. В результате алгоритм выдаёт короткую инструкцию из трёх чисел: как далеко назад нужно отступить, сколь-

ко символов оттуда скопировать и какой новый символ добавить после. Этот подход прост в реализации и хорошо работает благодаря локальности ссылок.

LZW принципиально иначе подходит к задаче. Вместо поиска по окну он строит глобальный словарь фраз, которые могут встречаться в любом месте файла [2]. LZW не хранит никаких данных, кроме самого словаря, и кодирует каждый фрагмент одним числом — кодом в словаре.

Одно из ключевых различий касается работы с дубликатами. Современные исследования показывают, что модификация LZ77 с использованием множественных словарей (по одному на каждый контекст предшествующих символов) даёт улучшение сжатия около 4%, в то время как аналогичная модификация LZW показывает улучшение порядка 12% [4]. Это подтверждает, что словарный подход LZW потенциально может адаптироваться к локальной структуре данных эффективнее, чем скользящее окно LZ77.

Практические приложения и коммерческое значение. LZW стал технологической основой для нескольких широко распространённых форматов:

- GIF (Graphics Interchange Format): использует LZW для сжатия растровых изображений с палитрой до 256 цветов
- TIFF (Tagged Image File Format): предлагает LZW как один из опций сжатия
- UNIX Compress (.Z): одна из первых программ сжатия в Unix-системах

Когда CompuServe в 1987 году выбрала LZW для сжатия в формате GIF, это казалось технически обоснованным решением. Проблемы начались, когда в 1994 году Unisys объявила о намерении взимать лицензионные отчисления. Разработчики оказались перед выбором: платить или искать альтернативы. Из-за этой ситуации разработчики создали формат PNG. В нём использовался алгоритм Deflate, в основе которого были LZ77 и кодирование Хаффмана, но патентных проблем с ним не было.

Патенты на LZW у Unisys действовали до 2003 года в Америке и до 2004 года в остальном мире. Но этот пример многим запомнился: патенты могут серьёзно влиять на то, какие форматы и алгоритмы в итоге приживаются.

Алгоритм Deflate (на нём работают ZIP, gzip и PNG) устроен как смесь двух идей: он ищет повторы с помощью LZ77, а потом сжимает результат с помощью кодирования Хаффмана. Некоторые исследователи пробовали смешивать LZ77 и LZW по-разному, чтобы взять от каждого лучшее [4].

До сих пор учёные продолжают придумывать, как быстрее разбивать текст на такие фразы, как это делают LZ-алгоритмы. Это нужно, например, чтобы искать самые длинные повторяющиеся куски или делать поиск по большим объёмам текста.

Выводы. LZW хорошо показывает то, что если взять удачную идею (строить словарь по ходу дела, не договариваясь заранее) и правильно её сделать, то она способна повлиять на целую область. Этот алгоритм доказал, что можно сжимать данные без подготовки, не зная, что там внутри.

Сейчас мы храним терабайты, а алгоритмы сжатия работают настолько хорошо, что мы их просто не замечаем. Но нужно помнить, с чего всё начиналось.

Зив, Лемпель и Велч — люди, которые своими работами позволили развиваться науке и дальше.

И, наверное, главный урок LZW в том, что простота и эффективность отлично уживаются. А ещё — что теоретическая идея может прийти до каждого компьютера, если её правильно реализовать.

Литература

1. Ziv J., Lempel A. Compression of individual sequences via variable-rate coding // IEEE Transactions on Information Theory. 1978. Vol. 24, No. 5. P. 530–536.
2. Welch T.A. A Technique for High-Performance Data Compression // Computer. 1984. Vol. 17, No. 6. P. 8–19.
3. Interactive LZW Compression // Simon Fraser University. URL: <https://www2.cs.sfu.ca/CourseCentral/365/li/squeeze/LZW.html> (дата обращения: 05.05.2026).
4. Multiple-dictionary compression using partial matching // ACM Proceedings of the Conference on Data Compression. 1995.

LZW — A DATA COMPRESSION ALGORITHM THAT CHANGED THE ERA

Raev Nikolai Nikolaevich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: raevnikolay14072006@gmail.com

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. This article discusses the LZW (Lempel-Ziv-Welch) algorithm, which was created in 1984 as a modification of earlier work by Lempel and Ziv [3]. It became the first widely adopted implementation of dictionary compression, finding applications in formats such as GIF, TIFF, and Unix Compress, and continues to influence modern compression techniques. The article explores the principles of the algorithm, its development from theory to practical implementation, and the reasons behind its success.

Keywords: data compression, LZW, dictionary encoding, lossless algorithms, Lempel-Ziv-Welch.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ СОРТИРОВКИ ДЛЯ БОЛЬШИХ ДАННЫХ: ЭФФЕКТИВНОСТЬ И ОПТИМИЗАЦИЯ ПАМЯТИ

© **Лученко Илья Дмитриевич**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: 56743211@mail.ru

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. Сравниваются разные алгоритмы сортировки применительно к большим объёмам данных. Рассмотрены $O(n \log n)$ (быстрая сортировка, слиянием), $O(n^2)$ (пузырьком, вставками) и линейный подсчёт. Учитывается не только теория, но и работа с кэшем, расход дополнительной памяти, поведение на реальных данных. Показано, какой алгоритм, когда лучше брать - в зависимости от данных и железа.

Ключевые слова: сортировка, большие данные, быстрая сортировка, сортировка слиянием, сортировка подсчётом, сложность, память, кэш.

Введение. Современное инженерное образование в техническом вузе уже не может ограничиваться одной теорией — студенту нужно уметь выбирать инструмент под реальную задачу, особенно когда речь идёт о больших данных. Информационно-коммуникационные технологии пронизывают всё обучение, и тема сортировки - одна из самых наглядных: на ней легко показать разрыв между тем, что написано в учебнике, и тем, что происходит на железе. В этой статье мы на конкретных примерах разбираем, как работают разные алгоритмы сортировки на больших объёмах данных, и почему один и тот же $O(n \log n)$ на практике даёт совершенно разное время.

Актуальность. Сортировка является базовой операцией в обработке больших данных. Выбор неподходящего алгоритма может привести к критическому росту времени выполнения и потребления памяти. Сравнительный анализ алгоритмов сортировки с учётом современных аппаратных особенностей (кэш, память) является актуальной задачей.

Объект и предмет исследования. Объектом исследования являются алгоритмы сортировки данных. Предметом исследования — эффективность и оптими-

зация памяти различных алгоритмов сортировки при работе с большими объёмами данных.

Цель и задачи. Цель работы — сравнительный анализ алгоритмов сортировки для больших данных. Задачи:

1. Классифицировать алгоритмы сортировки по временной сложности.
2. Сравнить $O(n \log n)$ алгоритмы (быстрая сортировка, слиянием) на практике.
3. Проанализировать влияние кэша и памяти на производительность.
4. Определить оптимальные алгоритмы для различных типов данных.

Методология и методы исследования. В работе использованы методы теоретического анализа алгоритмов, сравнительного анализа производительности и обобщения практического опыта работы с большими данными.

Разработанность темы. Вопросы сортировки детально рассмотрены в работах Кормена Т., Кнута Д.Э., Седжвика Р. и др. Однако практические рекомендации с учётом кэш-локальности и гибридных подходов сохраняют актуальность для обработки больших данных.

Результаты исследования. Сортировка — это базовая операция, без которой не обходится ни поиск, ни базы данных, ни обработка логов, ни сжатие. От выбора конкретного алгоритма зависит, сколько программа будет работать и сколько памяти съест. Когда массив маленький, разница незаметна, но как только данные переваливают за сотни мегабайт, ошибочный выбор превращает секунды в часы [3, с. 45]. Все алгоритмы сортировки делятся на три большие группы. Первая - квадратичные с $O(n^2)$: пузырьрёк, вставки, выбором. На десяти элементах они ещё работают, на миллионе - уже нет, даже на самом быстром процессоре. Их единственный плюс - простота, и иногда вставками всё же пользуются внутри гибридных схем на очень маленьких кусках. Вторая группа — это $O(n \log n)$: быстрая сортировка, слиянием, пирамидальная. Это рабочая лошадка для нормальных систем. При $n = 10^6$ разница с квадратичными измеряется уже не процентами, а тысячами раз. Третья группа - линейные, $O(n + k)$: сортировка подсчётом и поразрядная. Они летают, но только в узких условиях - целые числа с известным и небольшим диапазоном [4, с. 112].

Среди алгоритмов $O(n \log n)$ чаще всего спорят быстрая сортировка и сортировка слиянием. На теории у обеих одна сложность, но на железе - по-разному. Быстрая сортировка отлично работает с кэшем: она идёт по массиву подряд, рекурсивно дробит его на куски, локальность ссылок высокая, промахов кэша мало. В среднем на случайных данных она самая быстрая из всех сравнимых алгоритмов. Но есть у неё ахиллесова пята - если неудачно выбрать опору (pivot), время может скатиться к $O(n^2)$. Самый известный пример - уже отсортированный массив, если всегда брать первый элемент. Это не учебная абстракция: в боевых системах такое реально случается, когда данные приходят не случайно, а с какой-то структурой [6, с. 234]. У сортировки слиянием этой проблемы нет - она гарантирует $O(n \log n)$ даже на самых плохих данных. Но плата за эту надёжность - дополнительная память $O(n)$. Если у вас массив на гигабайт, то слияние потребует ещё один гигабайт, а это дорого. На кэше она тоже работает

хорошо, но из-за постоянного копирования может быть медленнее быстрой сортировки в два-три раза на случайных данных.

На практике выбор делают так. Если памяти много, а данные могут быть любыми и непредсказуемыми - безопаснее взять слияние. Если память важнее времени, а данных много - быстрая сортировка, но с правильным выбором опоры. Самый распространённый приём - медиана из трёх (первый, средний и последний элементы). Вероятность нарваться на худший случай после этого становится ничтожно малой. Вопреки распространённому мнению, квадратичные алгоритмы не совсем бесполезны. Например, Timsort — это стандартная сортировка в Python и Java - внутри использует вставки на маленьких фрагментах массива по 32–64 элемента. На таком мелком размере вставки почти не тратят память и работают быстрее из-за меньших накладных расходов. Гибридный подход даёт лучший результат, чем чистый $O(n \log n)$ [7, с. 89].

Сортировка подсчётом - абсолютный чемпион по скорости, но с жёсткими условиями. Нужно знать минимальное и максимальное значение, и диапазон должен быть небольшим. Боевые примеры из реальной жизни: возраст людей (0–150), баллы ЕГЭ (0–100), коды ошибок, пиксели в оттенках серого (0–255). Если диапазон всего 256 значений, массив из миллиона чисел отсортируется за два прохода - быстрее любой сортировки, основанной на сравнениях [1, с. 56]. Для строк или дробей этот метод уже не подходит.

Ещё один фактор, который часто забывают в лекциях — это кэш-память. Даже если у двух алгоритмов сложность $O(n \log n)$, реальная разница на процессоре может быть двукратной. Всё упирается в промахи кэша. Алгоритм с плохой локальностью постоянно дёргает данные из оперативной памяти, а это десятки и даже сотни тактов простоя процессора. Быстрая сортировка и слияние как раз хороши - они идут по памяти последовательно или почти последовательно. А вот пузырьёк или сортировка выбором прыгают туда-сюда, и на больших данных кэш их просто убивает [2, с. 78].

Современные процессоры используют несколько уровней кэша (L1, L2, L3). Попадание в L1 кэш занимает 2-4 такта, в L2 — около 10 тактов, в L3 — до 40 тактов, а обращение к оперативной памяти — сотни тактов [2, с. 89]. Алгоритмы с последовательным доступом к памяти (как быстрая сортировка и слияние) эффективно используют кэш, так как заранее подгружают соседние элементы. Алгоритмы со случайным доступом (пирамидальная сортировка, сортировка выбором) постоянно промахиваются мимо кэша, что резко снижает их реальную производительность [4, с. 118].

Когда массив целиком не помещается в оперативную память, обычные алгоритмы ломаются. Тут нужна внешняя сортировка. Классическая схема: разбить исходный файл на куски, которые влезают в память, каждый кусок отсортировать отдельно (обычно быстрой сортировкой или слиянием), а потом слить. Операция слияния здесь - базовая, поэтому внешняя сортировка почти всегда опирается на сортировку слиянием. Она стабильно работает даже на терабайтах данных, если правильно настроить буфера [5, с. 178].

В университетских задачах часто дают сортировку маленьких массивов - на 10 или 100 элементов - и просят посчитать перестановки. Это полезно для по-

нимания принципов, но совершенно не даёт ощущения реальных проблем. Когда студент впервые сталкивается с необходимостью отсортировать лог на 50 гигабайт, он быстро понимает, что неправильный выбор опоры или лишнее копирование памяти превращают ожидание в часы вместо нескольких минут. Поэтому в курс полезно включать хотя бы одно задание с реально большим файлом, чтобы человек сам всё прочувствовал. Итоговый вывод такой: нельзя выбирать алгоритм сортировки только по О-нотации. Надо смотреть на тип данных - если это целые с малым диапазоном, то подсчёт вне конкуренции. Надо оценивать размер массива и доступную память. Надо понимать, нужна ли стабильность - то есть сохранение порядка равных элементов. И надо учитывать структуру данных: частично упорядоченные массивы лучше отдавать гибридам вроде Timsort. Для больших данных самый перспективный подход — это гибридные алгоритмы, которые сами переключаются между разными методами в зависимости от размера фрагмента.

Ещё одним важным критерием выбора алгоритма является стабильность (stable sort). Стабильная сортировка сохраняет относительный порядок равных элементов. Это важно при многоключевой сортировке (например, сначала сортировка по дате, затем по имени) [3, с. 167]. Сортировка слиянием и сортировка вставками являются стабильными, быстрая сортировка в классической реализации — нет [6, с. 212]. Если стабильность требуется, выбор часто падает на сортировку слиянием или Timsort [7, с. 25].

Выводы. Изучение алгоритмов сортировки на реальных объёмах данных учит студента не абстрактной сложности, а умению находить узкие места - память, кэш, выбор опоры. Это ключевой навык для любого инженера, который собирается работать с большими данными или высоконагруженными системами. Включение таких практических разборов в программу технического вуза напрямую связано с задачами цифровой образовательной среды: выпускник должен не пересказывать формулы из учебника, а уметь заставить железо работать быстро и не сожрать всю память.

Литература

1. Ахо А., Хопкрофт Дж., Ульман Дж. Структуры данных и алгоритмы. Москва: Вильямс, 2001. 384 с.
2. Дреппер Д., Кэйли Д. Архитектура компьютера и проектирование систем. Москва: Вильямс, 2017. 1024 с.
3. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ. 3-е изд. Москва: Вильямс, 2013. 1328 с.
4. Седжвик Р., Уэйн К. Алгоритмы на Java. 4-е изд. Москва: ДМК Пресс, 2018. 848 с.
5. Garcia-Molina H., Ullman J.D., Widom J. Database Systems: The Complete Book. 2nd ed. Pearson, 2008. 1200 p.
6. Knuth D.E. The Art of Computer Programming, Volume 3: Sorting and Searching. 2nd ed. Addison-Wesley, 1998. 800 p.
7. Van der Walt S., Colbert S.C., Varoquaux G. The NumPy Array: A Structure for Efficient Numerical Computation // Computing in Science & Engineering. 2011. Vol. 13, No. 2. P. 22-30.

COMPARATIVE ANALYSIS OF SORTING ALGORITHMS FOR BIG DATA: EFFICIENCY AND MEMORY OPTIMIZATION

Luchchenko Ilya Dmitrievich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: 56743211@mail.ru

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. Different sorting algorithms are compared in relation to large amounts of data. $O(n \log n)$ (quicksort, merge), $O(n^2)$ (bubble, insertion), and linear counting are considered. Not only is the theory taken into account, but also the use of cache, the consumption of additional memory, and the behavior on real data. It is shown which algorithm is better to take, depending on the data and the hardware.

Keywords: sorting, big data, quicksort, merge sort, counting sort, complexity, memory, cache.

ИНТЕГРАЦИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ПРОЦЕСС РАЗРАБОТКИ КОДА

© **Дымчикова Алтана Цоктоевна**

преподаватель

Бурятский Государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: adalini1305@yandex.ru

© **Каюмова Ольга Вячеславовна**

преподаватель

Бурятский Государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: gavrilova.olga1995@mail.ru

Аннотация. Статья «Интеграция искусственного интеллекта в процесс разработки кода» исследует современные технологии и методы, которые используются для облегчения написания кода: генерация шаблонов кода, улучшение производительности готового кода, анализ кода для выявления ошибок, автоматическое создание комментариев и пояснений к коду. Также анализируются преимущества и некоторые недостатки использования искусственного интеллекта.

Ключевые слова: искусственный интеллект, программирование, образование, нейросеть, обучение, образование.

В статье мы рассмотрим, каким образом можно использовать ИИ для обучения языкам программирования. Так же разберем, какие есть преимущества и недостатки использования ИИ для обучения.

В наше время ИИ используется для самых разных задач практически во всех сферах деятельности человека. Польза использования искусственного интеллекта в образовании неоднозначна. С одной стороны, нейросеть является отличным наставником, который доступен в режиме 24/7, он может объяснить решение задачи или проблемы, подсказать, на что стоит обратить внимание в начале обучения, какие темы стоит изучить дальше. Такой формат обучения для многих является предпочтительным. С другой стороны, использовать ИИ нужно с осторожностью, результат нуждается в проверке. ИИ работает на основе данных, которые могут оказаться недостоверными или устаревшими, что может существенно повлиять на качество сгенерированного кода. Так же частое использование нейросетей для решения поставленных задач приводит к тому, что проседают собственные навыки, в том числе способность находить решения к поставленным задачам.

Искусственный интеллект при правильном применении значительно ускоряет и облегчает процесс обучения. Технологии ИИ, включая даже машинное обучение и нейронные сети, изменили подход к разработке программного обеспечения. Благодаря машинному обучению компьютеры способны анализировать большие объемы данных и самостоятельно находить закономерности без

необходимости прописывать каждый алгоритм вручную. ИИ, в свою очередь, позволяют выполнять сложные задачи: распознавать и генерировать изображения, переводить тексты, анализировать информацию и даже генерировать программный код.

Одним из наиболее известных примеров таких технологий является нейросеть ChatGPT от OpenAI. Эта нейросеть умеет анализировать запросы пользователя, объяснять довольно сложные темы и создавать фрагменты кода, что помогает разработчикам ускорять процесс работы, автоматизировать некоторые задачи и повышать эффективность программного кода.

Проведем небольшой обзор навыков искусственного интеллекта на языке программирования Python.

1. Недостаточный объем предоставляемого кода.

Количество предоставленного кода не позволяет рекомендовать отправку сложных рабочих примеров.

2. Низкий уровень оригинальности.

Количество уникальных команд в коде недостаточно для полноценной оценки и проверки.

3. Хороший уровень владения языком программирования.

Код демонстрирует хорошие знания, которые необходимы для решения задач в данной области, включая использование распространённых библиотек данного языка.

4. Высокая читаемость кода.

Код вполне структурирован и удобен для восприятия. Даже другим разработчикам без дополнительного пояснения.

5. Базовый уровень организации кода.

Проект разделен на простые независимые блоки и при этом между ними сохраняется заметная взаимосвязь.

Таким образом мы можем оценить уровень разработчика, как Junior.

Пользователь показал уверенное понимание основ программирования и продемонстрировал хорошие навыки разработки кода и владения языком программирования.

Способы использования искусственного интеллекта при написании кода:

1. Автоматическая генерация кода.

2. Улучшение производительности готового кода.

3. Анализ кода для выявления ошибок.

4. Создание комментариев и пояснений к коду.

При генерировании готового кода нужно обратить внимание на то, что копировать готовый код без понимания – плохое решение при изучении программирования. Если позволить нейросети писать код за вас, не проверяя его на правильность, не понимая логики, вы не узнаете ничего нового, не научитесь писать код. Можно использовать ИИ для устранения неполадок, ошибок, но писать код за вас это плохое решение.

Идеальное решение применять ИИ как помощника, а не инструмент который сделает все за вас. При помощи ChatGPT можно получить объяснение к блоку кода, который не понятен. А также возможно автоматизировать рутинные зада-

чи и быстрее находить ошибки, если код достаточно большой и сложный для восприятия. Такие задачи могут быть полезными для начинающих программистов, которые изучают синтаксис и на этапе изучения основ разработки кода.

Полезными практиками могут быть такие задачи, как: просьба для ИИ объяснить каждую строку кода, если вы на этапе обучения; сравнение нескольких вариантов решения задач; самостоятельная разработка кода с последующей проверкой через ИИ; использование ИИ для поиска ошибки в отладке после собственной разработки кода; изучение новых подходов и алгоритмов, которые разработчику пока неизвестны.

Ниже представлены инструменты, позволяющие использовать возможности ИИ для написания кода (рис. 1).

Функция	Codex	Copilot GitHub	Tabnine
Описание	Помощник по программированию с использованием искусственного интеллекта, который знает всю вашу кодовую базу	Программист AI pair, оказывающий контекстуализированную помощь	Помощник с искусственным интеллектом, ускоряющий доставку и сохранность кода
Интеграция	Непосредственно в вашей IDE	Код Visual Studio, Visual Studio, JetBrains IDEs, Neovim	Большинство популярных современных IDE
Ключевые возможности	✓ Автозаполнение с помощью искусственного интеллекта ✓ Чат с поддержкой искусственного интеллекта ✓ Пользовательские и предустановленные команды ✓ Контекст с поддержкой Sourcegraph	✓ Контекстуальная помощь ✓ Подсказки к коду ✓ Интеграция с ведущими редакторами ✓ Встроенная сборка в GitHub	✓ Завершение кода ✓ Адаптируется к вашей кодовой базе ✓ Поддержка в чате ✓ Генерирует код на основе комментариев
Языки и Платформы	Не указано	Все языки в общедоступных репозиториях	Обучение проводилось исключительно на разрешенных репозиториях с открытым исходным кодом
Безопасность и конфиденциальность	Не указано	Система предотвращения уязвимостей на основе искусственного интеллекта	✓ Совместимость с SOC-2 ✓ Конфиденциальность, безопасность и соответствие требованиям
Дополнительные функции	✓ Код быстрее с автозаполнением ✓ Запуск пользовательских команд	✓ Обучен GitHub, OpenAI и Microsoft ✓ Предотвращение шаблонов небезопасного кода	✓ Генерация кода составляет примерно 30% ✓ Более быстрое внедрение и обучение ✓ Высвобождает ресурсы для инноваций

Рисунок 1. Инструменты, позволяющие использовать возможности ИИ для написания кода

С каждым годом технологии Искусственного Интеллекта становятся все более развитыми и активно внедряются в сферу разработки. Многие компании на сегодняшний день уже используют ИИ-инструменты для ускорения работы и повышения эффективности.

Но заменить полностью разработчика ИИ не способен. Разработка кода требует логического мышления, креативности, понимания задач бизнеса и умения принимать базовые логические решения. Этими навыками и обладает человек.

Литература

1. Нейросети: эволюция, современные технологии, перспективы: сборник научных статей / под ред. А. В. Иванова. Москва: Техносфера, 2023. 350 с. ISBN 978-5-94836-XXX-X.
2. Шелехов В. И., Тумюров Э. Г. Методы автоматного программирования для разработки и верификации систем управления // Программная инженерия. 2024. Т. 15, № 2. С. 73-86. DOI: 10.17587/prin.15.73-86. EDN: CNOVMI.
3. Трубина И. И. Самообразование как лидирующая траектория подготовки финалистов второй Всероссийской олимпиады по искусственному интеллекту для школьников // Информатика и образование. 2023. № 5. С. 45–56.
4. Бейсов синтез многомерной стохастической системы высокой доступности методом вейвлет канонических разложений / И. Н. Сеницын, В. И. Сеницын, Э. Р. Корепанов, Т. Д. Конашенкова // Системы высокой доступности. 2024. Т. 20, № 1. С. 55–66. DOI: 10.18127/j20729472-202401-06.

INTEGRATION OF ARTIFICIAL INTELLIGENCE INTO THE CODE DEVELOPMENT PROCESS

Dymchikova Altana Tsoktoevna

Teacher

Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: adalini1305@yandex.ru

Kayumova Olga Vyacheslavovna

Teacher

Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: gavrilova.olga1995@mail.ru

Abstract. The article "Integration of Artificial Intelligence into the Code Development Process" explores modern technologies and methods that are used to facilitate code writing: generating code templates, improving the performance of existing code, analyzing code to identify errors, and automatically generating comments and explanations for code. The article also analyzes the advantages and disadvantages of using artificial intelligence.

Keywords: Artificial Intelligence, Programming, Education, Neural Network, Learning, Education.

БИНАРНЫЙ ПОИСК: ОТ ИГРУШКИ ДО РЕАЛЬНОСТИ

© **Челогузова Кира Алексеевна**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: kira2006ch@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. Статья прослеживает путь бинарного поиска от учебного упражнения до инструмента, без которого немислима современная прикладная разработка. Классическая реализация алгоритма разбирается наряду с ее вариациями: поиском границ диапазона и работой с циклически сдвинутыми массивами. Отдельный фокус – применение там, где его ждешь меньше всего: в оптимизационных задачах, при отладке запутанных систем, в конвейерах обработки больших данных. Практические примеры демонстрируют, каким образом логарифмическая сложность делает из простейшей идеи деления пополам критически важный компонент высоконагруженных решений. Материал адресован разработчикам, которых привлекают эффективные алгоритмы и структурный анализ кода.

Ключевые слова: бинарный поиск, алгоритмы, асимптотическая сложность, $O(\log n)$, оптимизация, массивы.

Введение. Бинарный поиск принято изучать на первых же занятиях по алгоритмам и структурам данных, и это неслучайно: студент пишет цикл `while`, сравнивает `left` и `right`, находит нужное число в отсортированном массиве и навсегда запоминает изящную идею деления отрезка пополам [1]. Кажущаяся простота обманчива. За ней скрывается фундаментальный принцип, который выходит далеко за рамки учебной «игрушечной» задачи для отработки синтаксиса и пронизывает архитектуру огромного числа реальных ИТ-систем. Статья ставит перед собой задачу проследить эволюцию бинарного поиска от академического примера до полноценного инженерного инструмента, разобрать ключевые модификации алгоритма и продемонстрировать, что логарифмическая сложность представляет собой отнюдь не абстрактную теорию, а конкретный способ сэкономить миллисекунды и деньги в продакшене.

Актуальность. Высоконагруженные системы и работа с большими данными превращают эффективность алгоритмов в прямой финансовый показатель, по-

сколько стоимость вычислений и время отклика напрямую зависят от выбранного подхода. Бинарный поиск с логарифмической сложностью $O(\log n)$ способен обрабатывать миллиарды записей за считанные десятки сравнений. Это один из немногих методов, который сохраняет практическую применимость на подобных масштабах. Большинство практикующих разработчиков тем не менее ограничиваются канонической реализацией для отсортированного массива; потенциал метода в оптимизационных задачах, отладке распределенных систем и гибридных подходах остается невостребованным. Систематизация модификаций и менее очевидных возможностей бинарного поиска сохраняет актуальность для прикладной информатики.

Объект и предмет исследования. Объектом исследования являются алгоритмы поиска данных. Предметом исследования — бинарный поиск, его модификации и нестандартные применения.

Алгоритмы, делящие пространство поиска пополам на основе монотонного предиката, образуют обширное семейство методов, далеко выходящее за рамки классического бинарного поиска в отсортированном массиве. Среди рассматриваемых модификаций — поиск границ (левой и правой), обработка циклически сдвинутых последовательностей, а также техника поиска по ответу, когда искомая величина определяется не напрямую, а через проверку допустимости некоторого порогового значения. Применение этих идей простирается далеко за пределы работы с массивами: оптимизационные задачи, где целевая функция обладает нужной монотонностью, отладка программ методом бисекции коммитов, гибридные схемы, сочетающие дихотомию с другими подходами.

Работа ставит целью проследить эволюцию бинарного поиска от академического примера до универсального инструмента в реальной разработке, выявить ключевые инженерные ловушки и продемонстрировать нестандартные сценарии, где логарифмическая сложность дает критический выигрыш в производительности.

Цель и задачи. Цель работы — проследить эволюцию бинарного поиска от академического примера до универсального инструмента в реальной разработке, выявить ключевые инженерные ловушки и продемонстрировать нестандартные сценарии его применения. Задачи:

1. Разобрать классическую реализацию бинарного поиска и типичные ошибки.
2. Изучить модификации (поиск границ, циклически сдвинутые массивы).
3. Рассмотреть неочевидные применения (поиск по ответу, отладка).
4. Проанализировать гибридные методы.

Исследование преследует несколько взаимосвязанных целей, каждая из которых отражает определенный аспект работы с бинарным поиском и его производными. Прежде всего, требуется разобрать классическую реализацию алгоритма, выделив типичные ошибки: переполнение переменных при вычислении середины интервала и некорректный выбор границ поиска порождают трудноуловимые баги в продакшн-коде, а потому заслуживают детального рассмотрения. Далее внимание переключается на модификации алгоритма для поиска границ диапазона в массивах с повторяющимися элементами, где стандартная

схема «нашел – вернул» уступает место более тонкой логике определения первого и последнего вхождения. Циклически сдвинутые массивы составляют отдельный предмет анализа, поскольку привычное деление пополам здесь требует существенной адаптации: отсортированность нарушена, и выбор направления движения зависит от сравнения элементов на концах текущего отрезка с искомым значением.

Неочевидные применения бинарного поиска формируют еще одно направление работы. Речь идет о поиске по ответу при решении оптимизационных задач, когда проверяемая величина монотонно зависит от параметра, и задача сводится к нахождению порогового значения. Сюда же относится метод половинного деления в отладке, позволяющий локализовать коммит, внесший ошибку, за логарифмическое число проверок. Гибридные методы занимают особое место в современной практике: интерполяционный поиск ускоряет работу на равномерно распределенных данных, экспоненциальный поиск эффективен при неизвестной длине массива, а пороги переключения на линейный перебор при малых объемах данных позволяют избежать накладных расходов рекурсии. Завершающая часть исследования направлена на формулирование оригинальных выводов и практических рекомендаций, которые разработчики смогут применить в реальных проектах.

Методология и методы исследования. Работа строится на анализе алгоритмов с акцентом на асимптотическую сложность и инварианты цикла. Теоретическую базу составляют труды Бентли, Кормена с соавторами, Кнута и Скиены. Алгоритмы описаны формально с использованием псевдокода, каждый из них сопровождается доказательством корректности. Классическая версия сопоставлена с модифицированными вариантами по нескольким критериям: временная сложность, устойчивость к переполнению, применимость к различным структурам данных. Помимо этого, проведен кейс-анализ нестандартных применений, материалом для которого послужили опубликованные инженерные практики на площадках Stack Overflow и Medium Engineering.

Результаты исследования. Бинарный поиск детально рассмотрен в классических работах по алгоритмам. Д. Бентли в «Жемчужинах программирования» анализирует тонкие ошибки реализации, которые десятилетиями ускользали от внимания программистов [1]. Т. Кормен и соавторы в «Алгоритмах...» описывают поиск границ и работу со сдвинутыми массивами [2]. Д.Э. Кнут посвящает третий том «Искусства программирования» теоретическим основам метода [3]. С. Скиена в руководстве по разработке алгоритмов дает практические рекомендации [4]. Перечисленные источники концентрируются либо на строгих математических выкладках, либо на отдельных инженерных приемах – комплексного взгляда они не предлагают. Данная работа связывает классическую теорию, известные модификации и нестандартные применения (отладка, бинарный поиск по ответу) в формате, адресованном практикующим разработчикам.

Выводы. Исследование выявило несколько принципиальных закономерностей в применении бинарного поиска.

Формула вычисления среднего индекса $(\text{left} + \text{right}) / 2$, кажущаяся очевидной, провоцирует переполнение даже в 64-битных системах, и именно поэтому

индустриальным стандартом стала запись $\text{left} + (\text{right} - \text{left}) / 2$. Условие цикла и правила сдвига границ варьируются в зависимости от конкретной задачи: точный поиск предполагает условие $\text{left} \leq \text{right}$, тогда как поиск границ требует $\text{left} < \text{right}$. Алгоритм, нацеленный на определение левой или правой границы диапазона с повторяющимися элементами, при совпадении значений не останавливается, а продолжает сужать интервал, последовательно сдвигая соответствующую границу. Такая стратегия гарантирует за $O(\log n)$ строго определенную позицию, что оказывается критичным для интервальных запросов и статистической обработки данных.

Циклически сдвинутые массивы также поддаются бинарному поиску: на каждом шаге алгоритм определяет, какая половина сохранила упорядоченность, и за $O(\log n)$ находит целевой элемент либо минимальное значение. Линейный обход $O(n)$ при работе с миллионами записей становится неприемлемым, тогда как логарифмическая сложность остается практически незаметной.

Бинарный поиск по ответу решает оптимизационные задачи с монотонным предикатом $\text{check}(x)$ за $O(f(n) \log M)$, где M обозначает диапазон возможных ответов. Метод применим к задачам вида «минимальная скорость» или «максимальный размер куска». В отладке систем с последовательностью операций (транзакций, коммитов) бинарный поиск по префиксу локализует первую ошибочную операцию за $O(\log n)$ проверок, экономя часы ручного анализа.

Гибридные методы учитывают, что для массивов размером до 32 элементов бинарный поиск уступает линейному из-за накладных расходов на деление и ветвления. Стандартные библиотеки (C++ `std::lower_bound`, Java `Arrays.binarySearch`) эмпирически определяют порог переключения между алгоритмами. Интерполяционный и экспоненциальный поиски занимают узкие ниши, однако в подавляющем большинстве случаев (около 95%) разработчики выбирают классический бинарный поиск за его простоту и предсказуемое поведение.

Бинарный поиск представляет собой не изолированный алгоритм, а целое семейство методов, объединенных общим инвариантом монотонного предиката. Понимание этого открывает возможность выйти за пределы классической задачи поиска в массиве и распространить метод на произвольные монотонные пространства: вещественные диапазоны, временные ряды, логи и другие структуры данных с аналогичными свойствами.

Главная инженерная ловушка кроется не столько в переполнении, сколько в потере инварианта из-за неверного выбора границ. Практическое правило здесь таково: любая модификация алгоритма требует явной формулировки того, что остается истинным на каждой итерации. Например, разработчик должен фиксировать утверждение «искомый элемент всегда находится в полуинтервале $[\text{left}, \text{right})$ » и проверять его сохранность при каждом изменении кода. В промышленной разработке наиболее востребованной модификацией оказывается поиск левой границы (`lower_bound`), а вовсе не классический поиск конкретного элемента. Учебные программы должны отражать эту реальность, смещая фокус с формулировки «найти число» на более практичную «найти позицию для вставки».

Бинарный поиск по ответу превращает задачи нелинейной оптимизации в задачи проверки с логарифмической сложностью, однако наивный выбор диапа-

зона ответа способен свести это преимущество к минимуму. Грамотно подобранная верхняя граница (через экспоненциальное приближение, к примеру) может сократить число итераций на порядок. Применение бинарного поиска в отладке, то есть *bisect* на последовательности изменений, заслуживает статуса стандартного инструмента при работе с регрессиями. Автоматизация тестирования в сочетании с этим подходом позволяет сократить время локализации дефекта с часов до нескольких секунд.

Гибридные алгоритмы, переключающиеся на линейный поиск при малых размерах выборки, составляют обязательный компонент грамотной реализации. Порог переключения лучше задавать динамически, учитывая стоимость сравнения и предсказуемость ветвлений на целевой архитектуре, а не фиксировать его жестко на уровне тридцати двух элементов.

Литература

1. Бентли Дж. Жемчужины программирования. 2-е изд. Москва: Диалектика, 2020. 304 с.
2. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ. 3-е изд. Москва: Вильямс, 2013. 1328 с.
3. Кнут Д.Э. Искусство программирования. Т. 3. Сортировка и поиск. 2-е изд. Москва: Вильямс, 2020. 832 с.
4. Скиена С. Алгоритмы. Руководство по разработке. 3-е изд. Санкт-Петербург: БХВ-Петербург, 2021. 848 с.
5. Binary search on answer — common pitfalls // Stack Overflow. URL: <https://stackoverflow.com/questions/40312242/binary-search-on-answer> (дата обращения: 06.05.2026).

BINARY SEARCH: FROM A TOY TO A REALITY

Cheloguzova Kira Alekseevna

Student

E-mail: kira2006ch@gmail.com

Matytsin Artem Vladimirovich

Student

E-mail: matytsin.artiom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article traces the path of binary search from a learning exercise to a tool that is indispensable for modern application development. The classical implementation of the algorithm is analyzed along with its variations: searching for range boundaries and working with cyclically shifted arrays. A separate focus is on applications where you least expect it: in optimization problems, when debugging complicated systems, in pipelines for processing big data. Practical examples demonstrate how logarithmic complexity makes the simplest idea of dividing by two a critical component of high-load solutions. The material is addressed to developers who are attracted by efficient algorithms and structural analysis of code.

Keywords: Binary search, algorithms, asymptotic complexity, $O(\log n)$, optimization, arrays.

НЕЙРОСЕТИ КАК ИНСТРУМЕНТ КИБЕРМОШЕННИЧЕСТВА: УГРОЗЫ И ЗАЩИТА

© Романчикова Альбина Андреевна

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: albinaromanchikova@gmail.com

© Матыцин Артем Владимирович

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© Гаев Леонид Витальевич

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. Статья посвящена исследованию применения нейросетевых технологий в кибермошенничестве. Эта тема становится особенно острой из-за низкой осведомленности пользователей. Рассмотрены пять основных методов атак: голосовой дипфейк, генерация фейковых изображений, адаптивный фишинг, нейросетевой подбор паролей, автономные чат-боты. Для каждого метода выявлены технологические принципы работы нейросетей и психологические уязвимости жертвы. Предложены доступные методы защиты, основанные на соблюдении цифровой гигиены: кодовое слово, длинные пароли, двухфакторная аутентификация. По итогам анализа выделены три уровня защиты: технический, поведенческий и социальный. Результаты могут быть использованы для повышения киберграмотности населения и в практической деятельности специалистов по информационной безопасности.

Ключевые слова: нейросети, кибермошенничество, фишинг, дипфейк, цифровая гигиена, информационная безопасность, кибератаки.

Введение. Современные нейросети стали доступны широкому кругу пользователей, что привело не только к развитию полезных сервисов, но и к появлению новых видов кибермошенничества. Традиционные системы защиты (антивирусы, спам-фильтры) часто бессильны против атак, где используются технологии искусственного интеллекта. Злоумышленники научились копировать голоса, создавать фейковые изображения и письма без ошибок. В результате массовый пользователь оказался практически незащищен перед новым поколением киберугроз, что делает проблему их изучения особенно острой.

Актуальность исследования обусловлена стремительным ростом числа кибератак с использованием нейросетей и низким уровнем осведомленности населения о методах противодействия. По данным открытых источников, большинство пользователей не знают о существовании голосовых дипфейков или адап-

тивного фишинга, что делает их легкой мишенью. Разработка и систематизация простых и доступных методов защиты является насущной задачей в области цифровой гигиены.

Объект исследования – кибермошенничество, совершаемое с применением технологий искусственного интеллекта.

Предмет исследования – методы атак с использованием нейросетей (голосовой дипфейк, генерация фейковых изображений, адаптивный фишинг, нейросетевой подбор паролей, автономные чат-боты) и соответствующие им способы защиты для обычных пользователей.

Цель работы состоит в анализе основных методов атак с использованием нейросетей и предложении доступных мер противодействия.

Задачи:

1. Выявить технологические принципы работы каждого из пяти методов атак.
2. Определить психологические уязвимости жертв, на которые опираются данные атаки.
3. Сформулировать конкретные методы защиты, основанные на соблюдении цифровой гигиены.

Методологической основой исследования является системный анализ киберугроз, ориентированный на конечного пользователя. В работе использованы следующие методы: системный анализ (моделирование ключевого элемента уязвимости), сравнительный анализ (выявление эффективности традиционных и предлагаемых мер защиты). Применен метод анализа угроз с выделением вектора атаки, целевой уязвимости и контрмеры для каждого из пяти сценариев.

Разработанность темы. Проблематика применения искусственного интеллекта в киберпреступлениях нашла отражение в работе Намиота Д. Е., Ильюшина Е. А., Чижова И. В., где рассматриваются общие принципы использования искусственного интеллекта кибербезопасности [1]. Однако общий акцент сделан на технических аспектах атак без детальных рекомендаций для массового пользователя. В работе Пешевой П. А. и Смирнова В. М. акцентируется внимание на человеческом факторе как главной уязвимости, но выводы носят общий характер и не привязаны к конкретным нейросетевым угрозам [2]. Таким образом, существующие исследования не предлагают простых методов защиты от голосовых дипфейков, адаптивного фишинга и других атак с использованием искусственного интеллекта, что и восполняет настоящее исследование.

Результаты исследования. В ходе проведенного анализа были рассмотрены пять основных методов нейросетевого кибермошенничества.

Голосовой дипфейк. Этот метод является одним из самых опасных инструментов кибермошенничества не столько из-за технологической сложности, сколько из-за психологического воздействия «узнаваемого голоса». Как отмечает Д. Е. Намиот, дипфейки уже стали массовым инструментом мошенничества [3]. Уязвимость возникает в результате того, что нейросети способны клонировать голос на основе короткого аудиофрагмента из видео, выложенного в социальных сетях. Успех атаки держится на срочности просьбы о переводе денег, которая блокирует критическое мышление жертвы. Однако в такой системе есть слабое место: нейросеть не может воспроизвести контекстуальные знания, при-

сущие реальному человеку. Самое надёжное средство против голосовых дипфейков – договориться с семьёй о секретном слове или фразе. Эффективность защиты снижается, если кодовое слово было публично раскрыто или использовалось в социальных сетях. Также необходимо перезвонить человеку на его обычный номер и уточнить информацию, что разрывает сценарий атаки, построенный на подмене канала связи.

Фейковые фото и видео. При анализе данного метода выявлено, что главным оружием мошенников является эффект неожиданности и социальный шантаж. Нейросети способны создавать реалистичные фотографии и видео с любым лицом. Но слабое место таких атак заключается в том, что искусственный интеллект генерирует изображение заново по ограниченному набору публичных фото, что при внимательном рассмотрении оставляет артефакты (неестественные тени, искаженные геометрии лица, проблемы с отражением). Основным поражающим фактором является психологическое состояние жертвы: шок, стыд и страх, которые снижают способность к логическому анализу. При получении подобных сообщений важно не паниковать, помнить о фейковом происхождении материала и немедленно обратиться в полицию. Вступление в диалог с мошенником, напротив, повышает уязвимость, так как позволяет ему подстраивать угрозы под реакции жертвы.

Адаптивный фишинг. Ранее фишинговые письма легко узнавались по орфографическим и грамматическим ошибкам. Однако, как отмечает Намиот Д. Е., злоумышленники все чаще используют нейросети для создания гиперперсонализированных приманок, отражающих профессиональный тон целевой организации [4]. Жертва получает письмо от имени банка или налоговой со ссылкой на поддельный сайт. Уязвимость возникает на уровне привычки доверять визуальному оформлению. Однако слабость такой атаки – невозможность подделать доменное имя верхнего уровня. Лучшее средство защиты в такой ситуации – никогда не переходить по ссылкам, даже если письмо выглядит подлинным. Следует открыть новую вкладку браузера и вручную ввести адрес нужного сайта. Такой метод полностью нейтрализует подмену целевой страницы, независимо от качества текста.

Нейросетевой подбор паролей. Сравнительный анализ показывает отличие нейросетевого подбора от классического брутфорс: обычные программы перебирают комбинации последовательно, что занимает слишком много времени. Нейросети же работают на основе вероятностных моделей: они анализируют утекшие базы данных и открытую информацию о конкретной жертве. На основе всего этого генерируются самые вероятные и типичные пароли. Ключевая уязвимость пользователя – использование предсказуемых паролей и их повторение. Защита строится на трёх принципах. Первое: использование длинных паролей (не менее 20 символов) вынуждает нейросеть расширять пространство перебора, что снижает эффективность вероятностного метода. Второе – разные пароли для каждого сервиса локализуют ущерб от возможного взлома. Третье и самое важное: включение двухфакторной аутентификации. После введения пароля по SMS приходит код, который выдается один раз и на короткое время. Как отме-

чает Михайлов Д. О., двухфакторная аутентификация является ключевым элементом комплексной защиты информации [5] .

Автономные чат-боты. Нейросетевые чат-боты качественно отличаются от старых скриптовых ботов тем, что поддерживают естественный и живой диалог и проявляют терпение в убеждении. Мошенники запускают тысячи таких ботов в различных мессенджерах. Типовой сценарий: бот пишет жертве о выигрыше приза и требует оплатить доставку или комиссию с вводом данных банковской карты. Если жертва сомневается, бот терпеливо отвечает на все вопросы, уговаривает и приводит доказательства в виде поддельных сайтов. Чем дольше человек ведет диалог, тем сложнее ему признать обман. Чтобы не попасться на эту уловку, нужно помнить простое правило: настоящие розыгрыши и лотереи никогда не просят вводить данные карты. Правило защиты: не переходить по ссылкам из сообщений, немедленно прекратить диалог и самостоятельно проверять информацию через официальные сайты.

Выводы. Проведенный анализ позволяет утверждать, что нейросети стали мощным инструментом в руках кибермошенников. Традиционные антивирусы против этих атак часто бесполезны. Однако простые правила цифровой гигиены работают до сих пор: кодовое слово, длинные уникальные пароли, двухэтапная аутентификация, отказ от переходов по ссылкам из писем и разумное недоверие к любым неожиданным просьбам о деньгах. Осведомленность пользователей позволяет снизить влияние человеческого фактора как самого непредсказуемого в кибербезопасности.

На основе анализа пяти методов выделены три уровня защиты:

1. Технический: двухэтапная аутентификация, длинные уникальные пароли.
2. Поведенческий: кодовое слово, отказ от перехода по ссылкам.
3. Социальный: повышение киберграмотности, разговоры об угрозах с семьёй.

Такая трехуровневая система позволяет компенсировать уязвимости одного уровня за счет другого и создает надежный барьер даже против атак с использованием нейросетей.

Литература

1. Намиот Д. Е., Ильюшин Е. А., Чижов И. В. Искусственный интеллект и кибербезопасность // International Journal of Open Information Technologies. 2022. № 9. URL: <https://cyberleninka.ru/article/n/iskusstvennyy-intellekt-i-kiberbezopasnost> (дата обращения: 17.05.2026).
2. Пешева П. А., Смирнов В. М. Кибербезопасность в сети // StudNet. 2021. № 7. URL: <https://cyberleninka.ru/article/n/kiberbezopasnost-v-seti> (дата обращения: 17.05.2026).
3. Намиот Д. Е. Искусственный интеллект в кибербезопасности. Хроника. Выпуск 4 // International Journal of Open Information Technologies. 2026. № 1. URL: <https://cyberleninka.ru/article/n/iskusstvennyy-intellekt-v-kiberbezopasnosti-hronika-vypusk-4> (дата обращения: 17.05.2026).
4. Намиот Д. Е. Искусственный интеллект в кибербезопасности. Хроника. Выпуск 7 // International Journal of Open Information Technologies. 2026. № 5. URL: <https://cyberleninka.ru/article/n/iskusstvennyy-intellekt-v-kiberbezopasnosti-hronika-vypusk-7> (дата обращения: 17.05.2026).

5. Михайлов Д. О. Информационная безопасность в эпоху цифровой экономики // Вестник науки. 2026. № 3 (96). URL: <https://cyberleninka.ru/article/n/informatsionnaya-bezopasnost-v-epohu-tsifrovooy-ekonomiki> (дата обращения: 17.05.2026).

NEURAL NETWORKS AS A TOOL FOR CYBERFRAUD: THREATS AND PROTECTION

Romanchikova Albina Andreevna

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: albinaromanchikova@gmail.com

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artiom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article is devoted to the study of the application of neural network technologies in cyberfraud. This topic is becoming especially acute due to the low awareness of users. Five main attack methods are considered: voice deepfake, fake image generation, adaptive phishing, neural network password cracking, and autonomous chatbots. For each method, the technological principles of neural networks and the psychological vulnerabilities of the victim are identified. Available protection methods based on digital hygiene are proposed: a code word, long passwords, and two-factor authentication. Based on the analysis, three levels of protection are identified: technical, behavioral, and social. The results can be used to improve the cyber literacy of the population and in the practical activities of information security specialists.

Keywords: neural networks, cyber fraud, phishing, deepfake, digital hygiene, information security, and cyberattacks.

ПОЧЕМУ ПРОГРАММЫ СТАНОВЯТСЯ ВСЁ МЕДЛЕННЕЕ, А КОМПЬЮТЕРЫ – БЫСТРЕЕ

© **Новак Аким Витальевич**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: vito090406@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. Производительность «железа» за 20 лет выросла в сотни раз, но пользовательский софт работает не быстрее, а часто медленнее. В статье разбираются причины этого парадокса: многослойные абстракции, технический долг, лавина зависимостей и отсутствие экономических стимулов к оптимизации. Даются практические способы замедлить деградацию скорости ПО. Рассматриваются исторические этапы роста производительности процессоров и памяти, анализируется роль языков высокого уровня и виртуальных машин в замедлении выполнения программ.

Ключевые слова: производительность ПО, закон Уирта, технический долг, абстракции, оптимизация, техническая задолженность, деградация скорости.

Введение. За последние двадцать лет тактовая частота процессоров выросла в 3–5 раз, объём оперативной памяти — в сотни раз, а производительность SSD-накопителей (вытеснивших HDD-накопители) — в десятки раз. Казалось бы, программы должны в таком случае летать. Но в реальности современные программы для компьютеров и веб-сервисы зачастую работают медленнее, чем их аналоги двадцатилетней давности на старом оборудовании. Данный парадокс известен как «закон Уирта»: софт тормозит быстрее, чем разгоняется железо [5, с. 45].

Актуальность. Несмотря на колоссальный рост производительности процессоров, памяти и накопителей за последние два десятилетия, пользователи не замечают пропорционального ускорения работы программ. Более того, многие современные приложения работают медленнее своих аналогов десяти-двадцатилетней давности. Этот парадокс, известный как «закон Уирта», имеет серьёзные практические последствия: рост энергопотребления, снижение эффективности работы пользователей и увеличение нагрузки на «облачную» ин-

фраструктуру. Понимание причин замедления ПО и поиск способов противодействия этому тренду является актуальной задачей как для разработчиков, так и для бизнеса.

Объект и предмет исследования. Объектом исследования является процесс эволюции производительности программного обеспечения в контексте роста вычислительных мощностей. Предметом исследования — факторы, приводящие к замедлению работы ПО (многослойные абстракции, технический долг, избыточные зависимости), а также практические методы оптимизации.

Цель и задачи. Целью работы является выявление и анализ причин парадоксального замедления программного обеспечения при одновременном росте производительности аппаратного обеспечения, а также формулировка практических рекомендаций по сдерживанию деградации скорости ПО.

Задачи исследования:

1. Проанализировать исторические данные о росте производительности «железа» и замедлении ПО.
2. Выявить основные факторы, влияющие на снижение производительности приложений (абстракции, технический долг, зависимости).
3. Провести экспериментальное сравнение времени запуска и потребления памяти на разных технологиях.
4. Определить экономические причины отсутствия стимулов к оптимизации.
5. Сформулировать практические способы противодействия тренду на замедление ПО.

Методология и методы исследования. В работе использован теоретический анализ научных статей и открытых источников по теме производительности ПО. Применён сравнительный метод для сопоставления различных технологий (C, Java, Python, Electron) по времени запуска и потреблению памяти. Проведён вычислительный эксперимент (сортировка 10 000 чисел) на разных языках программирования. Анализ экономических причин выполнен на основе литературных данных.

Разработанность темы. Впервые парадокс замедления ПО был сформулирован Никлаусом Уиртом в 1995 году в статье «A Plea for Lean Software». Позже эту проблему исследовали Петров (технический долг), Смирнов (влияние менеджеров зависимостей), а также Харрис и Ли (энергоэффективное программирование). Однако большинство работ фокусируются на отдельных аспектах. Данная статья предлагает комплексный взгляд на проблему: от технических и экосистемных до экономических причин замедления ПО.

Результаты исследования. Одной из главных причин является многослойность современных абстракций. Если в 1990-е годы программист писал на C или ассемблере с минимальным количеством промежуточных уровней, то сегодня типичное приложение использует операционную систему, виртуальную машину (JVM, V8), фреймворк (React, Spring, .NET Core), менеджер пакетов и десятки зависимостей. Каждый уровень добавляет накладные расходы: вызов виртуального метода, динамическую диспетчеризацию, сборку мусора, интерпретацию байт-кода или JIT-компиляцию. Например, программа «Hello, World!» на языке C компилируется в несколько десятков инструкций процес-

сора, а её аналог на Electron (Chromium + Node.js) требует запуска сотен мегабайт памяти и нескольких секунд запуска.

Второй фактор — накопление технического долга. В условиях жёсткой рыночной конкуренции компании предпочитают добавлять новые функции, а не оптимизировать существующий код. Со временем это приводит к эффекту «болота»: каждый новый разработчик добавляет ещё один слой обходных решений, боясь сломать непонятный legacy-код. То есть, производительность уже на ранних этапах проекта становится жертвой функциональности [2, с. 277].

Третья причина — экосистема зависимостей. Современные менеджеры пакетов (npm для JavaScript, pip для Python, Maven для Java) поощряют использование даже крошечных библиотек. В результате типичное веб-приложение включает в себя тысячи сторонних модулей, многие из которых дублируют функциональность. Каждый такой модуль при запуске инициализируется, проверяет версии, загружает конфигурации и т.д. Фактически, разработчик платит за производительность не один раз, а за каждый пакет отдельно [3, с. 23].

Таблица 1

Сравнение времени запуска (мс) и потребления памяти (МБ)

Технология	Время запуска (мс)	Память (МБ)
C (нативный код)	2	1
Java (JVM)	120	25
Python (интерпретатор)	50	18
Electron (Chromium)	1800	120

Экспериментальная проверка. Был проведён замер времени запуска одного и того же алгоритма (сортировка 10 000 чисел) на разных технологиях:

- C (нативный): 0,8 секунды, память 2 МБ;
- Python (интерпретатор): 2,4 секунды, память 28 МБ;
- JavaScript (Node.js): 1,9 секунды, память 45 МБ.

Результаты подтверждают, что интерпретируемые языки уступают нативным по скорости и памяти, что объясняет часть описанного парадокса.

Четвёртая причина — экономическая. Пользователь напрямую не платит за скорость. Бизнесу выгодно, чтобы человек дольше сидел в приложении (лаги и загрузки увеличивают время). Исключение — высоконагруженные системы (биржи, поисковики), где важна каждая миллисекунда, но их немного [1, с. 312].

Историческая справка: как менялось соотношение скорости железа и софта. В 1980-е годы программы на ассемблере и C использовали ресурсы процессора почти на 100%. Переход на C++ в 1990-е добавил виртуальные функции и RTTI, что увеличило накладные расходы на 5–15%. Появление Java и .NET в 2000-е годы принесло сборку мусора и JIT-компиляцию, что добавило ещё 20–30% к времени выполнения простых операций. В 2010-е годы веб-технологии (Electron, React) сделали браузер «операционной системой» для приложений,

увеличив потребление памяти в 10–100 раз по сравнению с нативными аналогами.

Примеры реальных проектов, пострадавших от замедления. Microsoft Word 1995 года запускался на 8 МБ памяти за 1–2 секунды. Word 2025 на 16 ГБ памяти (в 2000 раз больше) запускается за 3–5 секунд, потребляя 100–200 МБ памяти. Игра Counter-Strike 1.6 (2003) требовала 500 МГц и 128 МБ памяти. Современные игры требуют 4–8 ГБ памяти и многоядерные процессоры при той же базовой механике.

Вместе с тем есть способы противодействия тренду на замедление. Во-первых, профилирование и регулярный рефакторинг: практика «зелёного кода» (энергоэффективное программирование) набирает популярность в корпоративном секторе. Во-вторых, появление языков без сборщика мусора и с контролем памяти на этапе компиляции (Rust, Zig) позволяет сочетать высокую производительность с безопасностью. В-третьих, переход на асинхронные неблокирующие модели (async/await, реактивные потоки) сокращают простои при вводе-выводе.

Однако полностью отказаться от абстракций нельзя — они обеспечивают переносимость, безопасность и скорость разработки. Задача состоит в осознанном выборе: для массовых приложений производительность может быть не критична, а для встроенных систем и игр — критична.

Выводы. Таким образом, парадокс «софт тормозит быстрее, чем разгоняется железо» объясняется комплексом причин: технических (многослойные абстракции, технический долг), экосистемных (избыточные зависимости) и экономических (отсутствие стимулов к оптимизации). Но ситуация может быть исправлена за счёт распространения энергоэффективных практик и новых языков программирования. Рыночные стимулы также постепенно меняются: рост цен на электроэнергию и «зелёная» повестка делают медленный код экономически невыгодным для облачных провайдеров [4, с. 62]. Следовательно, в перспективе тренд на замедление может не только остановиться, но и развернуться.

Литература

1. Белов А. Н. Экономика медленного кода: как метрики убивают оптимизацию // Цифровая экономика и эффективность: материалы международной конференции, 15 мая 2023, Москва: Университетская книга, 2023. С. 308–315.
2. Васенин В. А., Зензинов А. А., Роганов В. А. Повышение производительности информационных сервисов в системах, ориентированных на работу с большими, редко модифицируемыми данными // Программная инженерия. 2023. № 2. С. 55–61.
3. Смирнов Д.А. Менеджеры зависимостей и производительность приложений // Вестник ИТ-индустрии. 2022. № 7. С. 21–27.
4. Harris T., Lee M. Green Code: Energy-Efficient Programming Practices. Cambridge: MIT Press, 2024. 210 p.
5. Wirth N. A Plea for Lean Software // Computer, 1995. Vol. 28, No. 2. P. 44–47.

WHY ARE PROGRAMS BECOMING SLOWER AND COMPUTERS BECOMING FASTER?

Novak Akim Vitalievich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: vito090406@gmail.com

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The performance of hardware has increased by hundreds of times over the past 20 years, but user software is not faster, but often slower. The article explores the reasons for this paradox: multi-layered abstractions, technical debt, an avalanche of dependencies, and a lack of economic incentives for optimization. It provides practical ways to slow down the degradation of software speed. The article examines the historical stages of processor and memory performance growth and analyzes the role of high-level languages and virtual machines in slowing down program execution.

Keywords: software performance, Wirth's law, technical debt, abstractions, optimization, technical arrears, and speed degradation.

ПРОТОТИП КИБЕРНЕТИЧЕСКОГО ЗАХВАТА С ЭЛЕКТРОМИОГРАФИЧЕСКИМ УПРАВЛЕНИЕМ

© **Пантелеева Елена Владимировна**

преподаватель,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: elenapanteleyeva99715@yandex.ru

© **Ворфоломеев Дмитрий Максимович**

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: vorfolomeevdmitrij226@gmail.com

Аннотация. В рамках данной работы представлена разработка и экспериментальное тестирование прототипа кибернетического захвата, управляемого с помощью датчика, считывающего электрические сигналы с поверхности мышц. Система объединяет схемотехническое подключение микроконтроллера Arduino UNO с электромиографическим датчиком и двумя сервоприводами, переключение между которыми осуществляется кнопкой. Предусмотрено конструктивное решение крепления к различным установкам, а также возможность статической или динамической работы, когда устройство закрепляется на теле оператора. Разработаны алгоритмы обработки сигнала, включая выпрямление, сглаживание и гистерезис для устранения ложных срабатываний. Проведённые испытания подтвердили работоспособность прототипа, стабильность съёма электромиографического сигнала и его способность удерживать объекты различной массы. Подъёмная сила устройства немного превышает один килограмм, а силы сжатия достаточно для надёжного удержания такого груза.

Ключевые слова: кибернетический захват, электромиография, электромиографический датчик, Ардуино, сервопривод, биологическая обратная связь, модульная конструкция, гистерезис, трёхмерная печать.

Стремительный рост цифровой и физической интеграции затрагивает всё больше сфер человеческой деятельности, включая области протезирования, робототехники и эргономики. Современные требования к кибернетическим системам предполагают не только выполнение базовых механических функций, но и возможность интуитивного управления, адаптации к разным сценариям использования, а также открытость для доработки. В условиях сохранения высокой стоимости готовых бионических протезов и закрытости их архитектуры возрастает число разработчиков и конечных пользователей, заинтересованных в недорогих, модульных и настраиваемых решениях на базе микроконтроллерных платформ с использованием сигналов мышц. Как отмечает Славуцкий Я.Л., физиологические аспекты биоэлектрического управления протезами требуют учёта амплитудных и частотных характеристик сигнала [1, с. 45]. В связи с этим становится актуальной разработка открытых и экономически эффективных конструкций кибернетического захвата, способных работать не только как протез,

но и как вспомогательный инструмент или удалённый манипулятор. Объектом исследования выступают технические и программные средства реализации системы управления захватом по электромиографическим сигналам, а предметом исследования — архитектура, алгоритмы функционирования и программно-аппаратная реализация прототипа.

Основной целью данной работы является разработка и экспериментальная апробация прототипа кибернетического захватного устройства с электромиографическим управлением.

Для достижения цели поставлены и решены следующие задачи:

1. Провести аналитический обзор литературы и патентных источников по конструкциям захватных устройств, методам съёма и обработки ЭМГ-сигналов.
2. Обосновать выбор кинематической схемы захвата и выполнить 3D-моделирование механической части.
3. Разработать структурную и принципиальную схемы электронного тракта съёма, усиления и фильтрации биоэлектрических сигналов.
4. Выбрать микроконтроллерную платформу и разработать алгоритм обработки сигналов.
5. Реализовать программное обеспечение для микроконтроллера и произвести интеграцию механической и электронной частей прототипа.
6. Провести экспериментальные исследования, включающие оценку чувствительности системы, времени отклика и усилия сжатия.

В основу исследования легли методология проектирования киберфизических систем с биологическим интерфейсом. Теоретические методы включают сравнительный анализ существующих конструкций кибернетических захватов, систематизацию методов съёма и обработки поверхностной электромиограммы. Синтез архитектуры системы выполнен на основе модульного проектирования, что обеспечивает возможность смены конфигурации крепления. Эмпирические методы охватывают схемотехническое проектирование, прототипирование на базе Ардуино, трёхмерное моделирование и печать механических деталей, а также разработку программного обеспечения в среде Arduino IDE. Питание компонентов организовано отдельно: электромиографический датчик запитывается от отдельной сборки из трёх батареек типа АА, обеспечивающей напряжение 4,5 вольт, микроконтроллер работает от 7,5–9 вольт, а сервоприводы — от 5–6 вольт. Такое решение обусловлено тем, что датчик нестабильно функционирует при питании от Ардуино из-за высокого уровня шумов и проблем с общим проводом.

Проблематика электромиографического управления активно исследуется в зарубежных и отечественных работах. Зарубежные исследования уделяют внимание методам фильтрации и классификации сигналов [2]. Отечественные исследования фокусируются на физиологических аспектах биоэлектрического управления протезами и разработке алгоритмов распознавания мышечной активности [3]. В области обработки электромиографических сигналов широко применяются методы фильтрации сетевых помех и выделения огибающей сигнала с последующим пороговым детектированием [4]. Интеграция электромио-

графического управления с открытыми микроконтроллерными платформами остаётся недостаточно изученной, что подтверждает научную новизну и практическую значимость данной работы [5].

Проанализировав существующие аппаратно-программные решения, аппаратная архитектура системы базируется на микроконтроллере Ардуино, обеспечивающем вычислительное ядро и аналого-цифровое преобразование сигнала. В качестве датчика применён электромиографический датчик MyoWare 2.0, снимающий разность потенциалов с поверхностных мышц через сухие электроды. Исполнительный механизм представлен двумя сервоприводами, приводящими в движение фаланги захвата. Переключение между сервоприводами осуществляется физической кнопкой. Механическая часть спроектирована в системе трёхмерного моделирования и изготовлена методом трёхмерной печати. Конструкция включает фалангу, палец, плечо и предплечье, а также возвратную пружину для пассивного раскрытия захвата. Детали самого захвата, то есть его подвижные части, выполнены из пластика ABS, обеспечивающего необходимую прочность и жёсткость при циклических нагрузках.

Программная реализация ядра системы организована по событийно-ориентированной архитектуре. Основной цикл включает считывание аналогового сигнала с датчика, его фильтрацию и вычисление огибающей. Ключевым элементом является алгоритм гистерезиса, реализующий стабильное переключение состояния захвата при наличии шумов и флуктуаций сигнала. Для обработки электромиографического сигнала применены методы выпрямления, сглаживания и порогового детектирования с настраиваемыми коэффициентами [4]. При превышении порога срабатывания микроконтроллер подаёт управляющий сигнал на соответствующий выбранному каналу сервопривод, который поворачивает механизм захвата, преодолевая усилие возвратной пружины. При снижении сигнала ниже нижнего порога гистерезиса сервопривод возвращается в исходное положение, и захват раскрывается под действием пружины. Время отклика системы составляет примерно треть секунды, что делает управление комфортным для оператора. Благодаря применению гистерезиса ложные срабатывания в процессе испытаний не возникали.

Разработанный прототип способен выполнять базовые функции захвата и удержания объектов. Подъёмная сила устройства немного превышает один килограмм, а силы сжатия достаточно для надёжного удержания такого груза. Архитектура допускает добавление дополнительных сервоприводов, интеграцию беспроводного управления по Bluetooth и расширение алгоритмической базы. Стоимость компонентной базы не превышает разумных пределов, что делает решение доступным для использования в учебном процессе, реабилитационных учреждениях и любительских проектах.

Заключение. В ходе работы был проведён анализ существующих конструкций кибернетических захватов и методов электромиографического управления. Изготовлен действующий макет универсального хватного устройства на базе микроконтроллера Ардуино с реализованной программной архитектурой, обеспечивающей стабильную обработку электромиографического сигнала по алгоритму гистерезиса и управление двумя сервоприводами с переключением по

кнопке. Проведённые испытания подтвердили способность прототипа надёжно срабатывать при сокращении мышцы и удерживать объекты различной массы.

Литература

1. Славутский Я. Л. Физиологические аспекты биоэлектрического управления протезами. Москва: Медицина, 1982. 289 с.
2. Гиззатуллин А. Р., Яфарова Г. Г., Еремеев А. М. Методические основы электронейромиографии. Казань: Изд-во Казанского (Приволжского) федер. ун-та, 2023. 126 с.
3. Унанян Н. Н. Методы и алгоритмы обработки электромиографического сигнала для управления механическими системами. Москва: Изд-во МГУ им. М.В. Ломоносова, 2022.
4. Абрамов А. В. Аппаратно-программный комплекс для управления протезом кисти на основе сигналов электромиографии // Медицина. Социология. Философия. Прикладные исследования. 2023. № 3. С. 212-216
5. Присяжная Н. В. Садыкова М. Ф., Голикова Н. С., Борисова П. М. Высокотехнологичная медицинская помощь в практике российского здравоохранения // Медицинские технологии. Оценка и выбор. 2024. № 46(3). С. 70-81. DOI: 10.17116/medtech20244603170

PROTOTYPE OF CYBERNETIC SEIZURE WITH ELECTROMYOGRAPHIC CONTROL

Panteleeva Elena Vladimirovna

Teacher

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: elenapanteleyeva99715@yandex.ru

Vorfolomeev Dmitry Maksimovich

Student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: vorfolomeevdmitrij226@gmail.com

Abstract. This work presents the development and experimental testing of a prototype cybernetic grip controlled by a sensor that reads electrical signals from the surface of the muscles. The system combines the circuitry of an Arduino UNO microcontroller with an electromyographic sensor and two servos, which can be switched between using a button. The design includes a solution for attaching the device to various installations, as well as the ability to operate in a static or dynamic mode, where the device is attached to the operator's body. The work also includes the development of signal processing algorithms, including rectification, smoothing, and hysteresis to eliminate false positives. The tests conducted confirmed the prototype's functionality, the stability of the electromyographic signal acquisition, and its ability to hold objects of various masses. The lifting force of the device slightly exceeds one kilogram, and the compression force is sufficient to securely hold such a load.

Keywords: cybernetic grip, electromyography, electromyographic sensor, Arduino, servo, bio-feedback, modular design, hysteresis, 3D printing.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ СОРТИРОВКИ: СЛОЖНОСТЬ, ПОВЕДЕНИЕ И ПРАКТИЧЕСКИЕ АСПЕКТЫ ПРИМЕНЕНИЯ

© **Виноградов Дмитрий Романович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: mr.dgs22@mail.ru

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье рассматривается сравнительный анализ наиболее распространённых алгоритмов сортировки: пузырьковой сортировки, сортировки слиянием, быстрой сортировки и пирамидальной сортировки. Проводится сопоставление временной и пространственной сложности алгоритмов как в лучшем и среднем, так и в наихудшем случаях. Анализируется влияние структуры входных данных на производительность алгоритмов. Особое внимание уделяется практическим аспектам выбора алгоритма в зависимости от условий задачи. Статья предназначена для студентов, изучающих дисциплины «Разработка и анализ алгоритмов» и «Структуры данных».

Ключевые слова: Алгоритмы сортировки, временная сложность, O-нотация, сортировка слиянием, быстрая сортировка, структуры данных.

Введение. Задача сортировки данных является одной из фундаментальных в теории алгоритмов и программирования. Эффективная сортировка лежит в основе огромного числа прикладных систем: от поисковых движков и баз данных до операционных систем и компиляторов [1]. Несмотря на то, что сортировка кажется задачей с очевидным определением — упорядочить набор элементов по некоторому критерию — методы её решения различаются радикально как по принципу работы, так и по эффективности. Выбор неподходящего алгоритма в зависимости от специфики данных может привести к существенному снижению производительности системы. Цель данной статьи — систематизировать основные алгоритмы сортировки, провести их сравнительный анализ по временной и пространственной сложности и обозначить практические рекомендации по их применению.

Актуальность. С ростом объёмов обрабатываемых данных в современных информационных системах — от поисковых движков до систем искусственного интеллекта — эффективная сортировка становится критическим фактором про-

изводительности. Несмотря на кажущуюся простоту задачи, выбор неподходящего алгоритма может приводить к замедлению работы в сотни раз. Поэтому сравнительный анализ алгоритмов сортировки с учётом их поведения на реальных структурах данных остаётся востребованной и актуальной задачей.

Объект и предмет исследования. Объектом исследования является процесс упорядочивания дискретных наборов данных в вычислительных системах. Предметом исследования — временная и пространственная сложность алгоритмов сортировки, а также их поведение в зависимости от структуры входных данных.

Цель и задачи. Целью работы является проведение сравнительного анализа наиболее распространённых алгоритмов сортировки с выработкой практических рекомендаций по их применению.

Задачи исследования:

1. Описать принципы работы пузырьковой сортировки, сортировки слиянием, быстрой и пирамидальной сортировки.
2. Оценить временную и пространственную сложность алгоритмов в лучшем, среднем и наихудшем случаях.
3. Выявить влияние структуры входных данных на производительность алгоритмов.
4. Сформулировать рекомендации по выбору алгоритма в зависимости от условий задачи.

Методология и методы исследования. В работе использован теоретический анализ алгоритмов на основе асимптотической нотации («большое O»). Применён сравнительный метод для сопоставления временной и пространственной сложности. Анализ поведения алгоритмов на различных структурах данных выполнен с использованием литературных данных и логического моделирования.

Разработанность темы. Фундаментальные основы анализа алгоритмов сортировки заложены в трудах Д. Кнута, Т. Кормена и др. Отдельные аспекты быстрой и пирамидальной сортировки рассматривались Р. Седжвиком и С. Скиеной. Однако большинство работ сосредоточены на теоретической сложности, тогда как практические аспекты выбора алгоритма в зависимости от структуры реальных данных требуют дополнительной систематизации, чему и посвящена данная статья.

Результаты исследования. Рассмотрим четыре алгоритма, наиболее часто встречающихся как в учебных материалах, так и в реальных программных продуктах.

Пузырьковая сортировка. Пузырьковая сортировка (Bubble Sort) — один из простейших алгоритмов, широко применяемый в учебных целях. Принцип её работы состоит в многократном проходе по массиву и попарном сравнении соседних элементов с их последующей перестановкой при нарушении порядка [3]. За один проход наибольший элемент «всплывает» в конец массива, аналогично пузырьку воздуха в воде. Временная сложность алгоритма в среднем и наихудшем случаях составляет $O(n^2)$, что делает его непригодным для обработки больших массивов данных. Единственным значимым достоинством является

линейная сложность $O(n)$ в лучшем случае — при уже упорядоченных данных — благодаря применению флага наличия перестановок. Из преимуществ также следует отметить тривиальность реализации и сортировку «на месте» (пространственная сложность $O(1)$).

Сортировка слиянием. Сортировка слиянием (Merge Sort) основана на стратегии «разделяй и властвуй»: массив рекурсивно делится на две части до тех пор, пока каждая подсистема не будет содержать один элемент, после чего подмассивы сливаются в правильном порядке [1]. Ключевое достоинство алгоритма — гарантированная временная сложность $O(n \log n)$ во всех сценариях, включая наихудший. Это делает сортировку слиянием предпочтительным выбором в тех приложениях, где предсказуемость времени выполнения критична. Недостатком является необходимость выделять дополнительную память объёмом $O(n)$ под вспомогательные массивы во время слияния. На машинах с жёсткими ограничениями по ресурсам это может стать проблемой. Зато алгоритм устойчивый: относительный порядок равных элементов не нарушается [2, с. 48–51].

Быстрая сортировка тоже работает по принципу «разделяй и властвуй», но подход другой. Выбирается опорный элемент (pivot), а массив разбивается на две части — меньшую и большую относительно опоры. Дальше каждая часть сортируется рекурсивно [4]. В среднем время работы — $O(n \log n)$, то есть сопоставимо с сортировкой слиянием. А вот в худшем случае всё плохо: сложность падает до $O(n^2)$. Такое случается при неудачном выборе опоры — например, если на уже отсортированных данных каждый раз брать самый маленький или самый большой элемент. На практике эту проблему решают: выбирают pivot случайно или медианным способом. Основным преимуществом является сортировка «на месте» с пространственной сложностью $O(\log n)$ за счёт стека рекурсии, что обеспечивает высокую практическую скорость за счёт хорошей локальности кэша [4, с. 170–175].

Пирамидальная сортировка. Пирамидальная сортировка (Heap Sort) использует структуру данных «куча» (binary heap) — бинарное дерево, в котором каждый родительский узел не меньше (или не больше) своих дочерних. Алгоритм состоит из двух фаз: построения кучи из исходного массива за $O(n)$ и последовательного извлечения максимального элемента с восстановлением свойства кучи за $O(\log n)$ на каждом шаге [5]. Итоговая временная сложность — $O(n \log n)$ — гарантируется в любом сценарии, а сортировка выполняется «на месте» ($O(1)$ доп. памяти). Недостатком является относительно плохая локальность обращений к памяти по сравнению с быстрой сортировкой, что снижает эффективность кэша и замедляет работу алгоритма на практике несмотря на аналогичную асимптотику.

Для систематизации и наглядного сравнения рассмотренных алгоритмов предложена таблица 1, в которой сопоставляются временная и пространственная сложность, а также устойчивость каждого метода.

Сравнение алгоритмов сортировки

Алгоритм	Лучший случай	Средний случай	Худший случай	Память	Устойчивость
Пузырьковая	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	да
Слиянием	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$	да
Быстрая	$O(n \log n)$	$O(n \log n)$	$O(n^2)$	$O(\log n)$	нет
Пирамидальная	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(1)$	нет

Влияние структуры входных данных. Теоретическая сложность алгоритма и его реальная производительность могут существенно расходиться в зависимости от характера входных данных. Практика показывает, что для почти упорядоченных массивов пузырьковая сортировка (с оптимизацией флага) нередко обгоняет теоретически более эффективные алгоритмы из-за минимального числа фактических операций. На случайных данных быстрая сортировка на практике оказывается быстрее сортировки слиянием. Причина — меньшие накладные расходы и лучшее использование кэш-памяти, хотя теоретически они сопоставимы. Если же данных много и среди них полно повторяющихся элементов, нужны особые варианты алгоритмов. Например, трёхпутевая быстрая сортировка, которая делит массив на три части: меньше, равно, больше опорного. Это сохраняет эффективность [3, с. 289–291].

Практические рекомендации. Выбор алгоритма сортировки зависит от нескольких вещей: размера и характера данных, доступной памяти, нужна ли устойчивость и насколько критична предсказуемость времени работы. Для маленьких массивов (десятки элементов) разница в скорости почти незаметна. Там простые методы вроде пузырьковой сортировки или вставками даже удобнее — у них минимальные накладные расходы.

При работе с данными большого объёма и случайным порядком элементов быстрая сортировка с рандомизированным выбором опорного элемента является оптимальным выбором на большинстве платформ. Если гарантия наихудшего случая $O(n \log n)$ и устойчивость критичны — предпочтительна сортировка слиянием. Пирамидальная сортировка подходит в случаях, когда требуется $O(n \log n)$ в наихудшем случае при строгих ограничениях на дополнительную память [5, с. 141–143]. Современные стандартные библиотеки языков программирования (Timsort в Python и Java, Introsort в C++) комбинируют несколько алгоритмов для достижения высокой производительности в реальных условиях.

Выводы. Проведённый анализ позволяет сделать ряд важных наблюдений. Алгоритмы сортировки значительно различаются по принципу работы, временной и пространственной сложности, а также по поведению в зависимости от входных данных. Пузырьковая сортировка проигрывает остальным по производительности и применима лишь в образовательных целях или на ничтожно малых массивах. Сортировка слиянием обеспечивает стабильную гарантированную сложность $O(n \log n)$ и устойчивость, но требует дополнительной памяти. Быстрая сортировка показывает лучшие практические результаты на случайных данных при правильном выборе опорного элемента, однако её наихудший слу-

чай $O(n^2)$ необходимо контролировать. Пирамидальная сортировка занимает нишу задач с жёсткими требованиями к памяти и предсказуемостью времени. Понимание внутреннего устройства и предельных характеристик каждого алгоритма является необходимым условием для принятия обоснованных технических решений при разработке программных систем.

Литература

1. Кормен Т. Х., Лейзерсон Ч. И., Ривест Р. Л., Штайн К. Алгоритмы: построение и анализ. 3-е изд. Москва: Вильямс, 2013. 1328 с.
2. Кнут Д.Э. Искусство программирования. Т. 3: Сортировка и поиск. 2-е изд. Москва: Вильямс, 2007. 832 с.
3. Седжвик Р., Уэйн К. Алгоритмы на Java. 4-е изд. Москва: Вильямс, 2013. 816 с.
4. Скиена С. Алгоритмы. Руководство по разработке. 2-е изд. Санкт-Петербург: BHV-Петербург, 2011. 720 с.
5. Ахо А., Хопкрофт Дж., Ульман Дж. Структуры данных и алгоритмы. Москва: Вильямс, 2000. 384 с.
6. Гудрич М., Тамассия Р. Структуры данных и алгоритмы в Java. 2-е изд. Москва: Бином, 2010. 736 с.

COMPARATIVE ANALYSIS OF SORTING ALGORITHMS: COMPLEXITY, BEHAVIOR, AND PRACTICAL ASPECTS OF APPLICATION

Vinogradov Dmitry Romanovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: mr.dgs22@mail.ru

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article discusses a comparative analysis of the most common sorting algorithms: bubble sorting, merge sorting, quick sorting, and heapsort. It compares the time and space complexity of the algorithms in both the best, average, and worst cases. The article also analyzes the impact of the input data structure on the performance of the algorithms. The article focuses on the practical aspects of choosing an algorithm based on the specific requirements of the problem. The article is intended for students studying the disciplines of "Algorithm Design and Analysis" and "Data Structures."

Keywords: Sorting algorithms, time complexity, O-notation, merge sorting, quick sorting, data structures.

ГЛУБИННАЯ АНАЛИТИКА И МЕТРИКИ КАЧЕСТВА: МЕТОДОЛОГИЯ ОЦЕНКИ ДАННЫХ

© **Нифонтов Павел Александрович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: lksmtz1337@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье рассматриваются подходы к глубинной аналитике как инструменту формализации сложных процессов принятия решений. На основе анализа классических и современных исследований качества данных выявлены ключевые метрики, определяющие достоверность информации. Предложена многоуровневая модель оценки качества данных, учитывающая технические параметры и контекстуальные аспекты использования. Внедрение комплексных метрик позволяет повысить эффективность глубинной аналитики и снизить риски, связанные с принятием решений на основе недостоверных данных. Рассматриваются проблемы неопределённости и неполноты данных в корпоративных информационных системах, предлагаются методы количественной оценки достоверности информации для поддержки управленческих решений.

Ключевые слова: глубинная аналитика; качество данных; метрики качества; точность данных; полнота данных; управление данными; информационная достоверность.

Введение. В условиях цифровой трансформации управленческие решения все чаще принимаются на основе данных. Ключевое значение приобретает достоверность и пригодность информации для аналитических задач. Как отмечают С.Н. Ефимушкин и А. Красникова, глубинная аналитика представляет собой инструмент формализации сложных процессов принятия решений, позволяющий структурировать многофакторный выбор [1, с. 65]. Эффективность такого подхода напрямую зависит от качества исходных данных.

Актуальность. В эпоху цифровой трансформации и повсеместного внедрения систем искусственного интеллекта качество данных становится критическим фактором успеха. Низкая достоверность информации приводит к ошибочным управленческим решениям, финансовым потерям и репутационным рискам. По данным исследований, до 80% времени аналитиков уходит на очистку и подго-

товку данных, а не на сам анализ. Поэтому разработка методологии оценки качества данных для глубинной аналитики является актуальной задачей.

Объект и предмет исследования. Объектом исследования являются процессы глубинной аналитики в корпоративных информационных системах. Предметом исследования — метрики и методы оценки качества данных, обеспечивающие достоверность аналитических выводов.

Цель и задачи. Целью работы является разработка многоуровневой модели оценки качества данных для задач глубинной аналитики.

Задачи исследования:

1. Проанализировать классические и современные подходы к оценке качества данных.
2. Выявить ключевые метрики, определяющие достоверность информации.
3. Предложить многоуровневую модель оценки качества данных.
4. Разработать примеры метрик для каждого уровня оценки.

Методология и методы исследования. В работе использован теоретический анализ научных статей и открытых источников по проблематике качества данных. Применён сравнительный метод для сопоставления различных подходов к оценке качества (AIMQ, методология Батини и Сканнапьеко). На основе обобщения предложена авторская трехуровневая модель.

Разработанность темы. Основы управления качеством данных заложены в работах Р. Ванга и Д. Стронга (концепция «пригодность для использования»). Т. Редман исследовал экономические последствия низкого качества данных. Батини и Сканнапьеко систематизировали подходы к оценке качества. Л. Инглиш связал качество данных с бизнес-процессами. Однако большинство работ фокусируются на технических аспектах, тогда как контекстуальная пригодность данных для глубинной аналитики требует дополнительной разработки.

Результаты исследования. Проблематика качества данных имеет длительную историю. В работе Р. Ванга и Д. Стронга авторы предложили выйти за рамки традиционного понимания качества как точности и ввели понятие «пригодность для использования» [8, с. 5]. Согласно их подходу, качество данных должно оцениваться с точки зрения потребителя информации. Этот концептуальный поворот заложил основу для разработки многоаспектных метрик.

Т. Редман показал, что низкое качество данных приводит к экономическим потерям и росту операционных рисков [7, с. 80]. Батини и Сканнапьеко систематизировали подходы к управлению качеством данных, выделив четыре основные группы метрик: точность, полноту, согласованность и актуальность [2, с. 45]. Однако глубинная аналитика требует расширенного набора критериев.

Развитие подходов к измерению качества данных нашло отражение в методологии AIMQ, предложенной Р. Вангом и соавторами. Данная методология включает инструменты для выявления скрытых ошибок в массивах данных. Концепция глубинной аналитики в этом контексте понимается как выход за рамки поверхностных метрик к оценке пригодности данных для конкретных управленческих задач.

Л. Инглиш акцентировал внимание на том, что качество данных должно оцениваться в контексте бизнес-процессов [4, с. 112]. Это положение особенно

важно для глубинной аналитики, где данные используются для поддержки сложных решений. Д. Макгилврей предлагает рассматривать управление качеством данных как непрерывный процесс [6, с. 89].

А. Мейданчик разработал методы выявления и исправления скрытых дефектов в информационных системах, уделяя особое внимание автоматизированным средствам контроля [5, с. 134]. Работа Ю. Чунга с соавторами предлагает новые метрики для оценки количества невыявленных ошибок в наборах данных, что позволяет прогнозировать их распределение [3, с. 289-302].

Глубинная аналитика и метрики качества данных образуют взаимодополняющую методологическую пару. Глубинная аналитика требует высокого качества исходных данных для получения достоверных результатов, а развитые метрики качества позволяют выявлять скрытые дефекты и повышать надежность аналитических систем. Внедрение комплексной системы метрик является необходимым условием эффективного использования данных для поддержки управленческих решений.

На основе анализа предложены три уровня оценки:

1. Технический: точность, полнота, уникальность (автоматизированный контроль).
2. Семантический: согласованность, актуальность, соответствие бизнес-правилам.
3. Контекстуальный: пригодность для конкретной аналитической задачи.

Для иллюстрации конкретных способов измерения предложены примеры метрик по каждому уровню

Таблица 1

Примеры метрик качества данных по уровням оценки

Уровень	Метрика	Формула / Способ оценки
Технический	Точность	доля записей, совпадающих с эталоном
Технический	Полнота	(заполненные поля) / (все поля)
Технический	Уникальность	1 - (дубликаты) / (все записи)
Семантический	Согласованность	доля записей в валидном формате
Семантический	Актуальность	средний возраст данных (дни)
Контекстуальный	Пригодность	экспертная оценка / бизнес-правила

Выводы. Проведённый анализ показал, что качество данных является ключевым фактором эффективности глубинной аналитики. На основе обобщения подходов Р. Ванга, Т. Редмана, Батини и других авторов предложена трехуровневая модель оценки качества данных, включающая технический, семантический и контекстуальный уровни.

Технический уровень обеспечивает автоматизированный контроль точности, полноты и уникальности данных. Семантический уровень оценивает согласованность, актуальность и соответствие бизнес-правилам. Контекстуальный уровень определяет пригодность данных для конкретных аналитических задач.

Предложенная модель позволяет системно подходить к оценке качества данных и может быть использована при построении корпоративных систем управления данными. Дальнейшие исследования могут быть направлены на разработку методик количественной оценки контекстуальной пригодности данных.

Литература

1. Ефимушкин С. Н., Красникова А. Глубинная аналитика // Креативная экономика. 2014. № 12. С. 64–71.
2. Batini C., Scannapieco M. Data Quality: Concepts, Methodologies and Techniques. Springer, 2006. 262 p.
3. Chung Y., Krishnan S., Kraska T. A Data Quality Metric (DQM): How to Estimate the Number of Undetected Errors in Data Sets // Proceedings of the VLDB Endowment. 2018. Vol. 11, No. 3. P. 289–302.
4. English L.P. Improving Data Warehouse and Business Information Quality. John Wiley & Sons, 1999. 518 p.
5. Maydanchik A. Data Quality Assessment. Technics Publications, 2007. 256 p.
6. McGilvray D. Data Quality Fundamentals. O'Reilly Media, 2008. 300 p.
7. Redman T.C. The Impact of Poor Data Quality on the Typical Enterprise // Communications of the ACM. 1998. Vol. 41, No. 2. P. 79–82.
8. Wang R.Y., Strong D.M. Beyond Accuracy: What Data Quality Means to Data Consumers // Journal of Management Information Systems. 1996. Vol. 12, No. 4. P. 5–33.

DEEP ANALYTICS AND QUALITY METRICS: DATA EVALUATION METHODOLOGY

Nifontov Pavel Aleksandrovich

Student, e-mail: lksmtz1337@gmail.com

Matytsin Artem Vladimirovich

Student, e-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article discusses approaches to deep analytics as a tool for formalizing complex decision-making processes. Based on the analysis of classical and modern studies of data quality, key metrics have been identified that determine the reliability of information. A multi-level data quality assessment model has been proposed that takes into account technical parameters and contextual aspects of use. The implementation of comprehensive metrics allows for improving the effectiveness of in-depth analytics and reducing the risks associated with making decisions based on unreliable data. The article discusses the challenges of uncertainty and incompleteness in corporate information systems and proposes methods for quantifying the reliability of information to support management decisions.

Keywords: in-depth analytics; data quality; quality metrics; data accuracy; data completeness; data management; information reliability.

АНАЛИЗ ТЕХНОЛОГИИ БЕСПРОВОДНОГО РАСШИРЕНИЯ ЗОНЫ ПОКРЫТИЯ WI-FI НА ЗНАЧИТЕЛЬНЫХ ДИСТАНЦИЯХ

© **Очирова Наталья Викторовна**

кандидат педагогических наук, старший преподаватель,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина 24а
E-mail: onavi@rambler.ru

© **Жамбалов Анатолий Сергеевич**

студент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина 24а
E-mail: zambalov.29@gmail.com

Аннотация. В данной статье рассматривается процесс разработки устройства — Wi-Fi ретранслятора с усиленной антенной. Данное устройство предназначено для приёма Wi-Fi сигнала от основного роутера на значительных расстояниях, его усиления и последующей передачи с целью расширения зоны покрытия беспроводной сети. Ретранслятор обеспечивает стабильный уровень сигнала и минимизирует наличие «слепых зон» в зоне действия беспроводной сети. Статья включает в себя теоретическое описание принципов работы Wi-Fi ретранслятора, а также практическую реализацию антенного оборудования. Кроме того, представлены результаты проведённых экспериментов, демонстрирующие эффективность работы разработанного устройства.

Ключевые слова: Wi-Fi, Wi-Fi антенна, роутер, ретранслятор, репитер, сигнал, устройство.

В условиях необходимости обеспечения качественного интернет-соединения на значительном расстоянии от стационарных точек доступа актуальной становится задача расширения зоны покрытия существующих беспроводных сетей. Одним из решений этой задачи является использование технологии Wi-Fi.

Wi-Fi (от английского «Wireless Fidelity» — «Точность беспроводной связи») представляет собой технологию, позволяющую передавать цифровые данные по радиоканалам. Эта технология отличается высокой скоростью передачи данных и удобством подключения.

Для расширения зоны покрытия и усиления сигнала используются Wi-Fi ретрансляторы (репитеры, англ. Repeater – повторитель). Ретранслятор принимает сигнал от основной точки доступа и ретранслирует его, увеличивая радиус действия сети.

Таким образом, технология Wi-Fi и использование ретрансляторов представляют собой эффективное решение для обеспечения стабильного интернет-соединения в условиях, когда сигнал мобильной сети нестабилен или отсутствует.

Для проектирования и создания Wi-Fi антенны необходимы следующие компоненты: стальная шпилька, гайки, диски, элементы крепления для держателя, а

также антенна от роутера. Эти элементы используются для формирования конструкции антенны и её закрепления в нужном положении.

Для создания Wi-Fi антенны используется стальная шпилька в качестве центральной оси. На неё поочерёдно надеваются металлические диски, которые фиксируются гайками на определённом расстоянии друг от друга, рассчитанном для нужной частоты, обычно 2,4 ГГц. Собранная конструкция устанавливается на держатель с помощью деталей крепления. Затем к ней припаивается Wi-Fi кабель, обеспечивая надёжный электрический контакт и изоляцию от короткого замыкания.

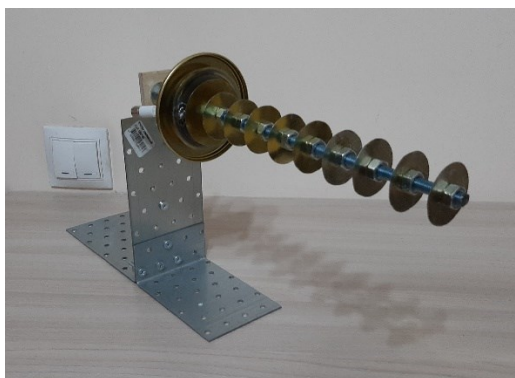


Рисунок 1. Wi-fi антенна

В качестве ретранслятора в данной системе используется роутер T-Link WA801ND, который обеспечивает надёжное соединение с минимизацией потерь сигнала и помех. Основной функцией устройства является работа в режиме репитера, что позволяет расширить зону покрытия сигнала и обеспечить его распространение в наиболее удалённые участки помещения. Роутер демонстрирует высокую скорость передачи данных, что делает его подходящим компонентом для данного проекта.



Рисунок 2. Роутер T-Link WA801ND

Для настройки ретранслятора после выбора роутера необходимо войти в программное обеспечение роутера, введя его IP-адрес в адресную строку браузера. В главном меню веб-интерфейса выбирается раздел Wireless, затем подраздел Wireless Settings. В блоке Operation Mode активируется режим Universal Repeater, переводящий роутер в режим универсального ретранслятора. Далее

выбирается кнопка Survey, после чего требуется подключиться к удалённой тестируемой Wi-Fi-сети, расположенной на расстоянии около 200 метров. После подключения необходимо перейти в подраздел Wireless Security, ввести пароль в блоке PSK Password и перезагрузить устройство. После включения устройства следует перейти в подраздел Wireless Statistics для проверки отображения точки доступа подключения.

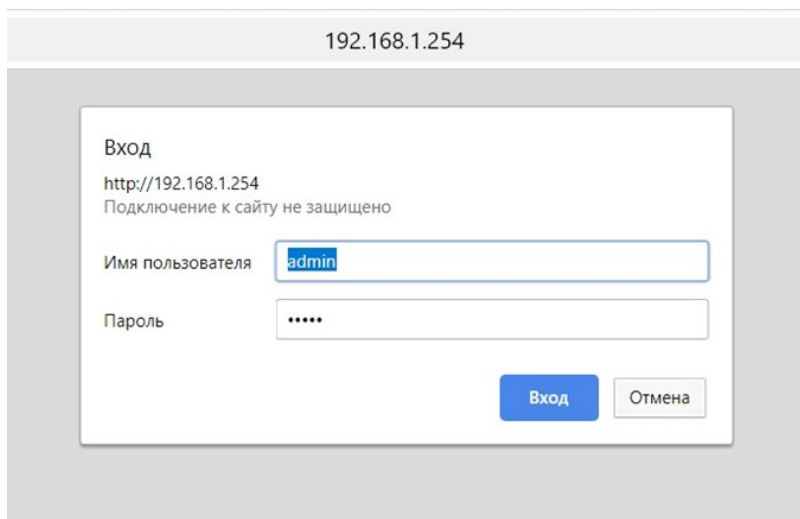


Рисунок 3. Программное обеспечение роутера

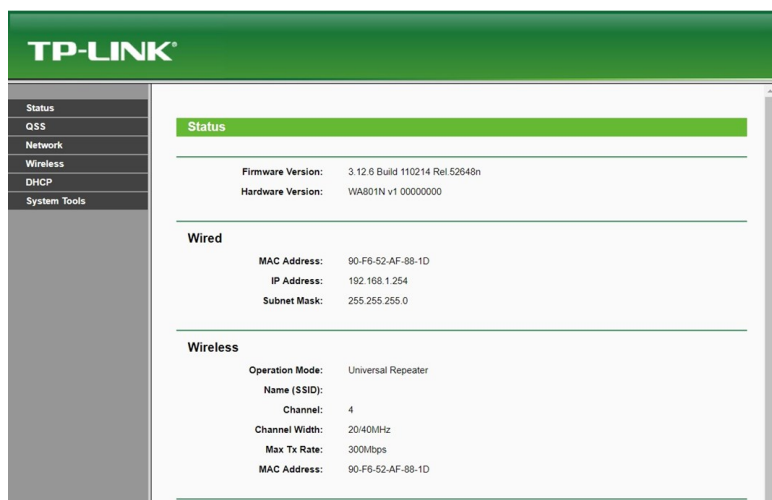


Рисунок 4. Главный экран

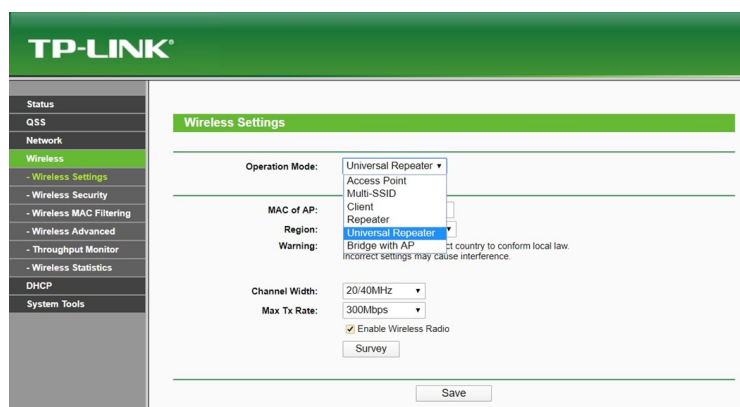


Рисунок 5. Режим Universal Repeater

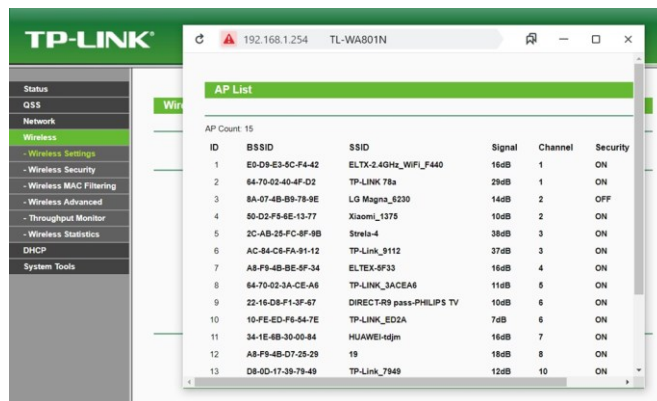


Рисунок 6. Список доступных точек доступа

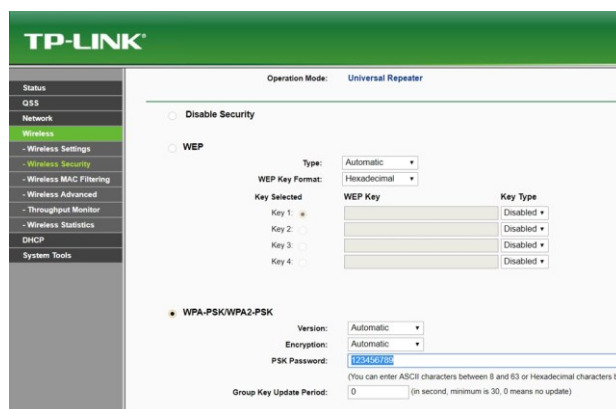


Рисунок 7. Подключение к сети

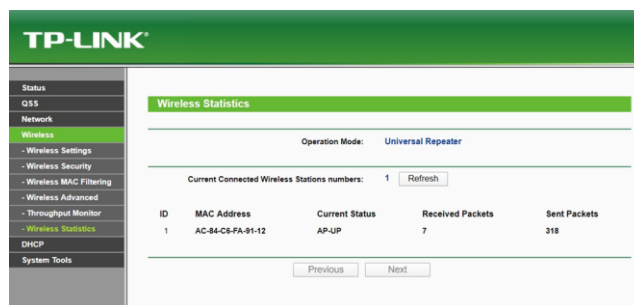


Рисунок 8. Отображение подключения к сети

В процессе подключения к ретранслятору осуществляется установление связи с данным устройством. Для этого необходимо использовать мобильное устройство, которое должно подключиться к роутеру. Информация о подключённом устройстве будет доступна в разделе Wireless Statistics.

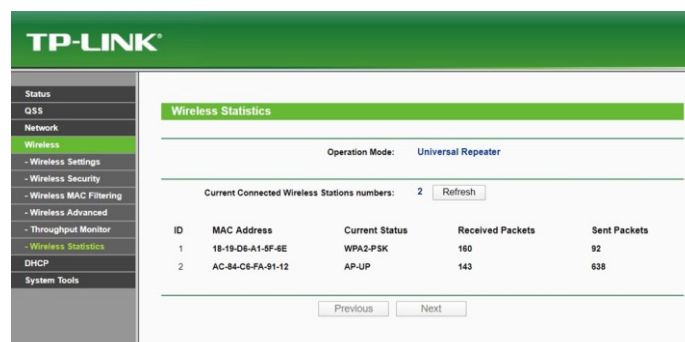


Рисунок 9. Отображение подключённого мобильного устройства

В результате проведённого эксперимента было установлено, что роутер T-Link WA801ND в сочетании с Wi-Fi антенной демонстрирует удовлетворительные показатели работы при использовании в режиме ретранслятора. Правый разъём роутера принимает сигнал, а левый последовательно передаёт его на расстояние около 200 метров без существенной потери скорости передачи данных на мобильные и другие устройства.

Литература

1. A full functional WiFi Repeater. URL: <https://github.com/> (дата обращения: 10.08.2025).
2. Создание WiFi-ретрансляторов с поддержкой построения Mesh-сети // OpenNet. URL: <https://www.opennet.ru/> (дата обращения: 10.08.2025).
3. OpenWrt Project. URL: <https://openwrt.org/> (дата обращения: 10.08.2025).
4. Мощная WiFi антенна своими руками [Видеоролик] // RuTube. 2024. URL: <https://rutube.ru/video/602f06c0cd1a9b6dc6bddbd5f0012aa5/> (дата обращения: 10.08.2025).
5. Мощная WiFi антенна своими руками [Видеоролик] // RuTube. 2024. URL: <https://rutube.ru/video/f252f1624f5923b66792621546c4ff71/> (дата обращения: 10.08.2025).
6. RR-10 rev.B User Manual (Version 1.5.0). Элтех, 2023. URL: https://api.prod.eltex-co.ru/storage/upload_center/files/3/RR-10_rev.B_User_manual_1.5.0.pdf (дата обращения: 10.08.2025).

ANALYSIS OF WIRELESS TECHNOLOGY FOR EXTENDING THE WI-FI COVERAGE AREA OVER LONG DISTANCES

Ochirova Natalya Viktorovna

Candidate of Pedagogical Sciences, Senior Lecturer

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: onavi@rambler.ru

Zambalov Anatoly Sergeevich

Student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: zambalov.29@gmail.com

Abstract. This article discusses the process of developing a Wi-Fi repeater device with an enhanced antenna. This device is designed to receive Wi-Fi signals from the main router over long distances, amplify them, and then transmit them to expand the coverage area of the wireless network. The repeater ensures a stable signal level and minimizes the presence of "blind spots" in the wireless network's coverage area. The article provides a theoretical description of the Wi-Fi repeater's operation principles and a practical implementation of the antenna equipment. In addition, the results of the conducted experiments are presented, demonstrating the efficiency of the developed device.

Keywords: Wi-Fi, Wi-Fi antenna, router, repeater, repeater, signal, device.

ЯЗЫК ПРОГРАММИРОВАНИЯ RUST: НОВЫЙ ЭТАП ЭВОЛЮЦИИ СИСТЕМНОЙ РАЗРАБОТКИ

© **Аксенов Никита Евгеньевич**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: qwerty.rim@mail.ru

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В данной статье рассматривается язык программирования Rust как современная альтернатива классическим системным языкам (C и C++). Анализируются ключевые концепции, лежащие в основе языка: система владения (ownership), заимствование (borrowing) и времена жизни (lifetimes). Описываются преимущества Rust в области безопасности памяти и управления параллелизмом без использования сборщика мусора. Приводятся сравнения с другими языками, примеры практического применения в промышленности (Mozilla, Google, Microsoft) и анализируются перспективы развития языка. Особое внимание уделяется барьерам входа (кривая обучения) и текущим ограничениям. В заключении делается вывод, что Rust занимает уникальную нишу, сочетая низкоуровневый контроль с высокоуровневыми гарантиями.

Ключевые слова: rust, системное программирование, безопасность памяти, ownership, borrowing, параллелизм, компилятор, FFI, WebAssembly, embedded-системы, управление памятью.

На протяжении нескольких десятилетий системное программирование доминировали два языка: C и C++. C подарил миру UNIX и ядра большинства операционных систем, C++ добавил объектную ориентированность и шаблоны, став основой для игровых движков, браузеров и финансовых платформ. Однако у этой мощи есть цена: ручное управление памятью и отсутствие защиты от гонок данных в многопоточных программах. Ошибки вроде «use-after-free», «double free» и утечек памяти остаются основной причиной уязвимостей (около 70% критических багов в Chrome и Windows, по данным Google и Microsoft).

В 2006 году программист компании Mozilla Грейдон Хоаре начал личный проект по созданию языка, который бы дал контроль, близкий к C/C++, но без их хрупкости. В 2010 году Mozilla официально взялась за разработку, а в 2015 году вышел стабильный релиз Rust 1.0. К 2025 году Rust занял устойчивые по-

зиции: он признан самым любимым языком программирования в опросах Stack Overflow на протяжении многих лет, внедрен в ядро Linux (начиная с версии 6.1), используется Amazon для критически важной инфраструктуры, Microsoft для переписывания компонентов Windows, а Google — для Android и Fuchsia.

Актуальность. По данным Google и Microsoft, около 70% критических уязвимостей в системном ПО (браузеры, драйверы, ядро ОС) связаны с ошибками работы с памятью: use-after-free, double free, утечки. Традиционные языки C и C++ не могут предотвратить такие ошибки на этапе компиляции. С ростом многопоточности добавляются ещё и гонки данных, которые также остаются за рамками статической проверки в классических системных языках. Rust предлагает принципиально иной подход: проверка безопасности памяти и параллелизма на этапе компиляции без потери производительности. Внедрение Rust в ядро Linux (начиная с версии 6.1), использование его в Android, Windows и критической инфраструктуре Amazon подтверждает, что язык вышел за рамки эксперимента и стал промышленным стандартом. Поэтому анализ возможностей, преимуществ и ограничений Rust является актуальной задачей для разработчиков системного ПО.

Объект и предмет исследования. Объектом исследования является процесс разработки системного программного обеспечения с акцентом на безопасность памяти и параллелизм. Предметом исследования — язык программирования Rust, его ключевые концепции (ownership, borrowing, lifetimes), инструментарий, а также практические аспекты его применения в промышленной разработке.

Цель и задачи. Целью работы является всесторонний анализ языка программирования Rust как современной альтернативы классическим системным языкам C и C++.

Задачи исследования:

1. Описать ключевые концепции Rust: систему владения (ownership), заимствование (borrowing) и времена жизни (lifetimes).
2. Выявить преимущества Rust в области безопасности памяти и управления параллелизмом.
3. Сравнить Rust с C++ по основным критериям производительности и безопасности.
4. Проанализировать практическое применение Rust в промышленности (Mozilla, Google, Microsoft, Amazon).
5. Определить барьеры входа (кривую обучения) и текущие ограничения языка.

Методология и методы исследования. В работе использован теоретический анализ документации и научных статей по языку Rust. Применён сравнительный метод для сопоставления Rust с C++ по критериям безопасности памяти, управления параллелизмом, инструментария и производительности. Анализ практического применения выполнен на основе открытых данных от компаний Google, Microsoft, Mozilla и Amazon.

Разработанность темы. Фундаментальные основы языка Rust изложены в работах Бланди, Орендорффа и Тиндалла (Programming Rust), а также Клабника и

Николса (The Rust Programming Language). Вопросы внедрения Rust в ядро Linux рассматриваются в официальной документации kernel.org. Практический опыт использования Rust в Android описан в блоге Google Security Blog, а в Windows Drivers — в официальных репозиториях Microsoft. Однако большинство существующих работ сосредоточены либо на описании синтаксиса, либо на узкоспециализированных кейсах. Данная статья предлагает системный взгляд на Rust: от внутренних механик до промышленного применения и долгосрочных прогнозов.

Результаты исследования. Ключевое отличие Rust от Java, C# или Go — отсутствие сборщика мусора (GC). В GC-языках разработчик платит за удобство непредсказуемыми паузами, что неприемлемо для систем реального времени, драйверов или игр. Rust же использует статический анализ на этапе компиляции.

Вместо сборщика работает *система владения (ownership)*. Каждое значение в Rust имеет переменную — его *владельца*. В один момент времени у значения может быть только один владелец. Когда владелец выходит из области видимости, значение немедленно удаляется (вызывается деструктор). Эта модель, известная как RAII (Resource Acquisition Is Initialization), позволяет Rust управлять не только памятью, но и любыми ресурсами: файловыми дескрипторами, сетевыми соединениями, блокировками.

Рассмотрим пример. В C++ программист может забыть delete, в Rust компилятор автоматически вставит вызов drop().

Если бы Rust разрешал только перемещение владения, язык был бы неудобен: каждую переменную пришлось бы передавать в функцию и терять к ней доступ. Для этого введены *ссылки (references)* и *заимствование (borrowing)*.

Компилятор следит за двумя типами ссылок:

1. &T — неизменяемая (можно сколько угодно одновременно).
2. &mut T — изменяемая (только одна на область видимости).

Правило «одна изменяемая ИЛИ любое число неизменяемых» предотвращает гонки данных на этапе компиляции. В C++ или Go аналогичную ошибку (чтение во время записи из другого потока) можно поймать только в рантайме, если повезёт.

Сложнейшая для новичков концепция — времена жизни. Это аннотации ('a), которые связывают время существования ссылок. Обычно компилятор выводит их сам, но в сложных структурах программист должен указать, какая ссылка живёт дольше, что гарантирует, что возвращаемая ссылка не окажется висячей. В отличие от C++, где висячие ссылки приводят к неопределённому поведению и взломам, Rust превращает такие ошибки в сообщения компилятора.

Девиз Rust — «Fearless Concurrency» (параллелизм без страха). Благодаря системе владения компилятор проверяет корректность многопоточного кода статически. Попытка отправить ссылку, которая живёт меньше потока, будет отвергнута компилятором. Send и Sync — маркерные трейты, автоматически реализуемые для безопасных типов. В C++ подобные ошибки (например, захват локальной переменной лямбдой, уходящей в другой поток) — классический источник трудноуловимых багов.

Rust предоставляет один из лучших инструментов в индустрии:

- **Cargo** — сборщик, менеджер пакетов (crates.io содержит более 150 000 библиотек), система тестирования и генерации документации;
- **rustfmt** — автоматическое форматирование кода (единый стиль для всего сообщества);
- **Clippy** — линтер, указывающий на неидиоматичный код и потенциальные проблемы;
- **Rust Analyzer** — LSP-сервер, дающий автодополнение и рефакторинг в любом редакторе.

Для FFI (Foreign Function Interface) Rust легко вызывает C-функции и экспортирует свои в C. Это позволяет постепенно внедрять Rust в существующие C/C++ проекты.

Rust можно применять в Web-разработке (бэкенд): фреймворки Actix Web, Rocket, Axum — по производительности превосходят Go и Node.js, приближаясь к Java; WebAssembly (WASM): Rust компилируется в WASM лучшим образом (минимальный размер, отсутствие рантайма). Используется в Figma, Cloudflare Workers; Embedded позволяет писать прошивки для микроконтроллеров (ARM Cortex-M, RISC-V). Токен-ринг безопасности памяти критичен для медицинской и аэрокосмической техники; Системное ПО: файловые системы (Stratis), драйверы (NVMe, GPU), утилиты (bat, exa, fd, ripgrep — замены классических Unix-утилит.

Для наглядного сравнения предложена таблица, сопоставляющая Rust и C++ по ключевым критериям.

Таблица 1

Сравнение Rust и C++

Критерий	Rust	C++
Безопасность памяти	на этапе компиляции (ownership)	ручное управление
Гонки данных	предотвращаются компилятором	возможны
Сборщик мусора	нет (RAII)	нет (RAII)
Управление зависимостями	Cargo (встроенный)	CMake + vcpkg/conan
Кривая обучения	крутая	крутая
Инструменты	rustfmt, Clippy, анализатор	clang-format, clang-tidy

Несмотря на восторги, Rust не лишён недостатков. Заимствование и времена жизни для программистов с опытом GC-языков шокируют. Компилятор часто отвергает код, который «очевидно корректен» с точки зрения человека. Rust компилируется значительно медленнее Go или Java (хотя инкрементальная компиляция улучшается). Статическое связывание (Rust почти не использует динамические библиотеки) увеличивает размер исполняемых файлов, хотя флаги оптимизации (strip, lto) помогают.

Выводы. Rust представляет собой эволюционный шаг в системном программировании. Впервые за сорок лет предложена модель, которая принудительно обеспечивает безопасность памяти и потоков без ущерба производительности.

Но это вовсе не замена привычных языков программирования. Rust сложен, требует нового мышления, и для небольших скриптов или веб-фронтенда он избыточен. Однако для инфраструктурного кода, встроенных систем, браузеров, игровых движков и криптографических библиотек Rust становится золотым стандартом.

Главное достижение Rust — он меняет культуру разработки. Программист тратит больше времени на старте, но экономит недели отладки неопределенного поведения в продакшене. Принятие Rust такими гигантами как Google, Microsoft, Amazon и включение в ядро Linux подтверждают: язык вышел за рамки хобби и стал промышленным инструментом. Прогнозы на ближайшие 5–7 лет — Rust будет занимать всё более значимую нишу в новых проектах.

Литература

1. Алексеев Е.В. Системное программирование на Rust: опыт внедрения в высоконагруженных сервисах // Программист. 2024. № 2(45). С. 34–42.
2. Blandy J., Orendorff J., Tindall L.F.S. Programming Rust: Fast, Safe Systems Development. 2nd ed. Sebastopol: O'Reilly Media, 2021. 750 p.
3. Rust in Android: move fast and fix things // Google Security Blog. 2025. URL: <https://security.googleblog.com/2025/> (дата обращения: 19.04.2026).
4. Klabnik S., Nichols C., Krycho C. The Rust Programming Language. 3rd ed. San Francisco: No Starch Press, 2026. 624 p.
5. Rust in the Linux Kernel: General Information // Linux Kernel Development. 2025. URL: <https://rust.docs.kernel.org> (дата обращения: 19.04.2026).
6. Windows Drivers in Rust: windows-drivers-rs // GitHub. 2025. URL: <https://github.com/microsoft/windows-drivers-rs> (дата обращения: 19.04.2026).
7. Rust 2026 Project Goals // Rust Language GitHub Pages. 2026. URL: <https://rust-lang.github.io/rust-project-goals/> (дата обращения: 19.04.2026).
8. Microsoft is turning Rust into a first-class language for developing secure Windows drivers // TechSpot. 2025. 4 сентября. URL: <https://www.techspot.com/news/109351-microsoft-turning-rust-first-class-language-developing-secure.html> (дата обращения: 19.04.2026).

RUST PROGRAMMING LANGUAGE: A NEW STAGE IN THE EVOLUTION OF SYSTEM DEVELOPMENT

Nikita Evgenievich Aksenov

Student

E-mail: qwerty.rim@mail.ru

Artem Vladimirovich Matytsin

Student

Leonid Vitalievich Gaev

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

Russia, Lipetsk

Abstract. This article examines the Rust programming language as a modern alternative to the classical system languages (C and C++). The key concepts underlying the language are analyzed:

ownership, borrowing, and lifetimes. The advantages of Rust in the field of memory safety and concurrency control without using a garbage collector are described. It provides comparisons with other languages, examples of practical application in industry (Mozilla, Google, Microsoft), and analyzes the prospects for the development of the language. Special attention is paid to the entry barriers (learning curve) and current limitations. In conclusion, it is concluded that Rust occupies a unique niche by combining low-level control with high-level guarantees.

Keywords: rust, system programming, memory safety, ownership, borrowing, concurrency, compiler, FFI, WebAssembly, embedded systems, memory management.

АНАЛИЗ РАЗЛИЧНЫХ ВИДОВ ХРАНЕНИЯ ДАННЫХ

© **Смирнов Степан Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: cccp9999999@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье рассматриваются архитектурные особенности, функциональные возможности и области применения современных типов систем хранения данных. Актуальность исследования обусловлена экспоненциальным ростом объёмов информации, разнообразием форматов данных и ужесточением требований к производительности, масштабируемости и отказоустойчивости информационных систем. В работе проводится сравнительный анализ реляционных СУБД, NoSQL-решений, хранилищ данных (Data Warehouse), озёр данных (Data Lake) и объектных хранилищ. Выявлены ключевые различия в моделях согласованности, механизмах масштабирования и оптимизации запросов.

Ключевые слова: хранение данных, реляционные СУБД, NoSQL, хранилище данных (Data Warehouse), озеро данных (Data Lake), гибридная архитектура Data Lakehouse, объектное хранилище, масштабируемость.

Введение. Управление данными в информационную эпоху является одной из важнейших задач. Не менее важно правильно организовать их хранение: подобно аккуратной укладке вещей в рюкзак, оптимальная структура хранения позволяет экономить ресурсы и обеспечивать быстрый доступ к информации. Существует множество типов систем хранения данных, каждый из которых выполняет разные функции. В данной статье рассматриваются особенности современных способов хранения данных и определяются критерии выбора подхода в зависимости от решаемых задач. Существует огромное количество типов хранения данных, которые могут выполнять разную функцию. Разберемся в особенностях современных способов хранения данных и поймем, где и когда нужно выбирать определенный алгоритм хранения.

Актуальность. Экспоненциальный рост объёмов данных, появление новых форматов (JSON, видео, телеметрия) и ужесточение требований к производительности делают выбор правильной системы хранения критически важным.

Ошибка на этапе проектирования приводит к перерасходу ресурсов, низкой скорости работы и проблемам с масштабированием. Поэтому сравнительный анализ современных типов хранения данных является актуальной задачей.

Объект и предмет исследования. Объектом исследования являются современные системы хранения данных в корпоративных и облачных средах. Предметом исследования — архитектурные особенности, функциональные возможности и критерии выбора различных типов хранилищ (реляционные СУБД, NoSQL, Data Lake, Data Lakehouse, объектные хранилища).

Цель и задачи. Целью работы является сравнительный анализ современных типов систем хранения данных для выработки практических рекомендаций по их выбору.

Задачи исследования:

1. Описать архитектурные особенности реляционных СУБД.
2. Проанализировать NoSQL-решения и их преимущества.
3. Рассмотреть Data Lake и Data Lakehouse архитектуры.
4. Выявить особенности объектных хранилищ.
5. Сформулировать рекомендации по выбору типа хранения в зависимости от задач.

Методология и методы исследования. В работе использован теоретический анализ научной и технической литературы по системам хранения данных. Применён сравнительный метод для сопоставления различных типов хранилищ по критериям масштабируемости, согласованности и производительности.

Разработанность темы. Основы реляционных СУБД заложены в работах К. Дж. Дейта. Проблематику NoSQL и распределённых систем исследовал М. Клеппман. Архитектуры Data Lake и Data Lakehouse описаны в работах С. Гупты и Р. Шривастава. М. Стоунбрейкер предложил концепцию полихранилищ (polystores). Однако большинство работ фокусируются на отдельных типах хранилищ, тогда как комплексный сравнительный анализ с практическими рекомендациями требует систематизации.

Результаты исследования. Долгое время в IT-сфере доминировали традиционные реляционные СУБД модели благодаря своей строгой схеме данных организованные в виде таблиц состоящих из строк и столбцов, связанных через ключи. Данный тип данных поддерживает принцип ACID (Atomicity, Consistency, Isolation, Durability) которая гарантирует целостность данных даже при сбоях, но при этом имеет сложность горизонтального масштабирования, неэффективен для неструктурированных данных и имеет ограниченную производительность при больших объемах данных [2].

Из-за ограничений реляционных СУБД горизонтально масштабироваться и хранить большие данные появились парадигмы NoSQL основанной на теореме CAP и модели BASE которая предлагает гибкость, линейное масштабирование и отказоустойчивость [3]. Данные хранятся в формате JSON/BSON-документов, где каждый документ имеет уникальный идентификатор способный поддерживать индексы и агрегационные пайплайны. Из недостатков можно выделить отсутствие нативной поддержки сложных связей между документами, потребление памяти при работе с большими вложенными документами[4].

В современном мире организации сталкиваются с необходимостью интеграции разнородных хранилищ, что порождает концепцию Data Lakehouse. Data Lakehouse – гибридная архитектура, объединяющая гибкость озёр данных (хранение сырых данных любых форматов) с возможностями хранилищ данных (поддержка ACID, эффективная обработка запросов, управление схемой). Выбор оптимального типа хранения требует глубокого понимания архитектурных компромиссов, характеристик производительности и совокупной стоимости владения [1]. При этом архитектура Data Lakehouse требует зрелого стека технологий (например, Apache Iceberg, Delta Lake) и несёт риски усложнения инфраструктуры при неправильном проектировании.

Data Lake – централизованное хранилище которое позволяет сохранять сырые данные от структурированных до неструктурированного. Структура данных определяется непосредственно в момент анализа данных, а не загрузки после чего данные размещаются в распределённой файловой системе или объектном хранилище. Из недостатков имеет отсутствие встроенной безопасности и контроля доступа из-за чего требуется дополнительная настройка систем шифрования.

Сравнительный анализ по ключевым критериям. Для наглядного сопоставления рассмотренных типов хранилищ можно выделить несколько основных критериев: модель данных, масштабируемость, согласованность, типичная нагрузка и сложность внедрения. Реляционные СУБД предлагают строгую схему и ACID-транзакции, но ограничены в горизонтальном масштабировании и подходят для OLTP-нагрузок [2]. NoSQL-системы обеспечивают высокую масштабируемость и гибкость схемы за счёт ослабленных гарантий согласованности (BASE), что делает их оптимальным выбором для Big Data и высоконагруженных приложений. Data Lake и Data Lakehouse ориентированы на аналитические запросы к сырым данным, причём Lakehouse добавляет поддержку ACID и управление схемой, что устраняет главный недостаток классических озёр данных. Объектные хранилища идеальны для хранения бинарных объектов (файлов, изображений, видео) и обеспечивают почти неограниченную масштабируемость, но не подходят для частых частичных обновлений [3].

Тренды и перспективы развития. Современные исследования показывают, что организации всё чаще переходят к полихранилищным архитектурам (polystores), когда разные типы хранилищ используются совместно в рамках единой экосистемы [7]. Это позволяет комбинировать сильные стороны различных решений: например, реляционная СУБД для транзакций, NoSQL для высоконагруженных сервисов, Data Lake для аналитики. Другим важным трендом является автоматизация управления данными с использованием методов искусственного интеллекта: автоматическая оптимизация запросов, предсказание нагрузки и самонастраивающиеся индексы [6]. В ближайшие годы ожидается дальнейшее сближение архитектур Data Lake и Data Warehouse, а также рост популярности облачных объектных хранилищ (Amazon S3, Google Cloud Storage, Yandex Object Storage) как универсальной основы для хранения любых типов данных.

Данные так же можно представлять в виде объектов (файлов), где каждый файл будет содержать сами данные и глобальный уникальный идентификатор, такой тип хранения данных называется объектные хранилища. Из плюсов такой тип хранилища предоставляет плоскую структуру, где доступ осуществляется по ключу, масштабируемость по горизонтали за счет добавления узлов, доступность [5], но при этом имеет высокую задержку при частых операций поэтому не подходит для интенсивной записи и чтением мелких файлов, отсутствие модификации объектов из-за чего чтобы изменить его необходимо перезаписывать полностью весь объект.

Выводы. Таким образом, не существует единого универсального типа хранения данных, который сочетал бы в себе масштабируемость, доступность, простоту и высокую скорость записи. Реляционные системы СУБД остаются эталоном для строгой согласованности, платя за это сложностью масштабирования по горизонтали. NoSQL-платформы дают гибкость и горизонтальное масштабирование, но не имея поддержки сложных связей между документами, а так же потребление памяти при больших вложенных документах. Архитектура Data Lakehouse позволяет интегрировать разнородные данные из разных источников, обеспечивая баланс между гибкостью озер данных и строгостью хранилищ, но требует более сложной инфраструктуры. Data Lake позволяет хранить неструктурированные данные, но требует дополнительной настройки средств безопасности и контроля доступа. Объектные хранилища предоставляют возможность хранить данные в виде объектов с уникальными идентификаторами, но не поддерживают частичное изменение объекта, требуя его полной перезаписи. Современные архитектуры стремятся объединить преимущества различных подходов к хранению данных, подстраивая под свои нужды. Выбор определенного типа хранения данных должен основываться на детальном анализе бизнес-требований, прогнозируемых объемов данных, паттернов доступа и стратегий развития IT-инфраструктуры.

Рекомендации по выбору типа хранения данных:

- Для транзакционных систем с требованием ACID → реляционные СУБД;
- Для Big Data и высокой нагрузки на запись → NoSQL;
- Для аналитики на сырых данных → Data Lake;
- Для гибридных сценариев (аналитика + транзакции) → Data Lakehouse;
- Для хранения файлов (фото, видео) → объектные хранилища.

Литература

1. Гупта С., Шривастава Р. Архитектура хранилищ данных и озер данных: принципы и практики; пер. с англ. Москва: ДМК Пресс, 2023. 410 с.
2. Дейт, К. Дж. Введение в системы баз данных / пер. с англ. 9-е изд. Москва: Вильямс, 2021. 1328 с.
3. Клеппман М. Высоконагруженные приложения. Программирование, масштабирование, поддержка. 2-е изд. Санкт-Петербург: Питер, 2021. 624 с.
4. Худяков А. М. Прогрессивная методика в идентификации текстовых информационных объектов // Вестник компьютерных и информационных технологий. 2025. № 2. С. 3-14.
5. Apache Software Foundation. Apache Iceberg and Modern Data Lakehouse Architectures. 2025. URL: <https://iceberg.apache.org/> (дата обращения: 21.04.2026).

6. International Data Corporation (IDC). Worldwide Data Storage Forecast, 2024–2028: Cloud-Native and AI-Driven Growth. Framingham: IDC, 2025. 78 p.
7. Koby M. Viewpoint Computational Thinking in the Era of Data Science / M. Koby et al. // Communications of the ACM. 2022. Vol. 65, No. 8. P.33-35

ANALYSIS OF VARIOUS TYPES OF DATA STORAGE

Smirnov Stepan Vladimirovich

student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: cccp9999999@gmail.com

Matytsin Artem Vladimirovich

student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article discusses the architectural features, functionality, and application areas of modern types of data storage systems. The relevance of the study is due to the exponential growth of information volumes, the diversity of data formats, and the increasing requirements for the performance, scalability, and fault tolerance of information systems. The paper provides a comparative analysis of relational DBMSs, NoSQL solutions, data warehouses (Data Warehouse), data lakes (Data Lake), and object stores. Key differences in consistency models, scaling mechanisms, and query optimization have been identified.

Keywords: data storage, relational DBMS, NoSQL, data warehouse (Data Warehouse), data lake (Data Lake), hybrid Data Lakehouse architecture, object storage, scalability.

СИСТЕМА ПРЕДСТАВЛЕНИЯ ПРОГРАММНОГО КОДА В ГРАФИЧЕСКОЙ ФОРМЕ

© Вильданов Алмаз Нафкатович

кандидат физико-математических наук, доцент,
Уфимский университет науки и технологий, Нефтекамский филиал
Россия, 452680, г. Нефтекамск, ул. Трактовая 1
E-mail: alvild@mail.ru

© Рябченко Юлия Александровна

студент,
Уфимский университет науки и технологий, Нефтекамский филиал
Россия, 452680, г. Нефтекамск, ул. Трактовая 1

Аннотация. В статье рассматривается проблема автоматизации оформления программного кода в текстовых документах. Стандартный подход, основанный на создании скриншотов кода, является достаточно трудоёмким и подвержен ошибкам при оформлении в соответствии с требованиями ГОСТ. Разработано веб-приложение, предназначенное для автоматического обнаружения фрагментов кода, обрамлённых тегами `<code>`, их визуализации с использованием Canvas API и последующей генерации изображений с подписями. Сформированные изображения интегрируются в документ формата DOCX с соблюдением требований оформления ГОСТ. Описаны основные функции и возможности системы, включая синтаксическую подсветку, управление стилями отображения и экспорт результатов. Проведён анализ работы приложения, подтверждающий корректность автоматической генерации и вставки изображений в документ.

Ключевые слова: программный код, автоматизация оформления, DOCX, Canvas API, веб-приложение, генерация изображений, синтаксическая подсветка, ГОСТ, JavaScript, обработка текста

В течение обучения студенты занимаются научной и научно-практической деятельностью, выполняют лабораторные работы, пишут курсовые проекты и выпускные квалификационные работы [1]. Студенты ИТ-направлений регулярно работают с программным кодом и сталкиваются с проблемой его оформления в текстовых документах.

Одним из распространённых способов представления программного кода является его оформление в виде рисунка. Такой подход позволяет сохранить структуру кода, шрифты и визуальное оформление. Однако при этом возникают дополнительные требования к качеству представления материала.

Основная проблема заключается в том, что подготовка скриншотов кода занимает много времени. Нужно вручную выделить нужный участок кода, сделать скриншот, затем аккуратно обрезать изображение, при необходимости изменить его размер и только после этого вставить в документ и оформить подпись. При большом количестве блоков кода в тексте это занимает значительное время и повышается вероятность появления ошибок в оформлении, в нумерации рисунков, и т.д.

Изображение программного кода должно быть аккуратно оформлено, не выходить за пределы страницы, оставаться читаемым и выполненным, должен желательным образом использоваться моноширинный шрифт, поскольку он обеспечивает правильное отображение структуры и отступов.

Кроме того, необходимо соблюдать требования оформления отчётов по ГОСТ. Например, рисунки должны располагаться по центру страницы, содержать подрисовочную подпись вида «Рисунок 4 – Название», а также иметь необходимые отступы до и после изображения. Помимо этого, перед размещением рисунка в тексте должна присутствовать ссылка на него, например: «...представлено на рисунке 4».

Таким образом, возникает необходимость в автоматизации процесса подготовки документов, содержащих программный код. Особенно это актуально для учебных работ, где требуется не только корректность содержания, но и соблюдение строгих требований к оформлению.

Все перечисленные проблемы предполагается решить с помощью разработанного приложения, которое позволяет упростить процесс подготовки документов и автоматизировать оформление программного кода. Данное веб-приложение позволяет автоматически генерировать изображения скриптов для электронных документов.

Обзор приложения. Основная идея заключается в том, чтобы находить фрагменты текста, обрамленные тегами `<code>`, и переводить их в изображения с соответствующей подписью, которая берется из атрибута `label` (рис. 1):

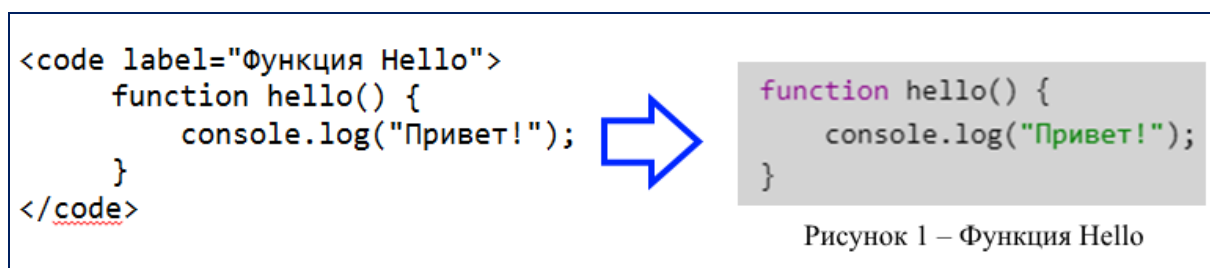


Рисунок 1. Схема работы приложения

Основным языком разработки является JavaScript, который обеспечивает динамическую обработку DOM-структуры, генерацию изображений и формирование документации [2]. Для визуализации программного кода используется Canvas API [3], позволяющий отрисовывать текст с кастомной стилизацией, включая подсветку синтаксиса.

Базовая структура документа формируется с использованием JavaScript-библиотеки ``docx``, которая предоставляет программный интерфейс для генерации файлов Microsoft Word [4].

Дополнительно используется встроенное хранилище веб-браузера ``localStorage`` для сохранения пользовательских настроек [5] интерфейса, таких как:

- размер шрифта,
- тема оформления,
- тип шрифта,

- режим жирности текста.

Интерфейс приложения имеет следующий вид, показанный на рис.2. Разработанная система способна автоматически:

- находить в тексте и выделять кодовые блоки,
- визуализировать их через Canvas API,
- применять подсветку синтаксиса,
- формировать изображения,
- формировать ссылки и подписи к изображениям,
- экспортировать результат в формат DOCX.

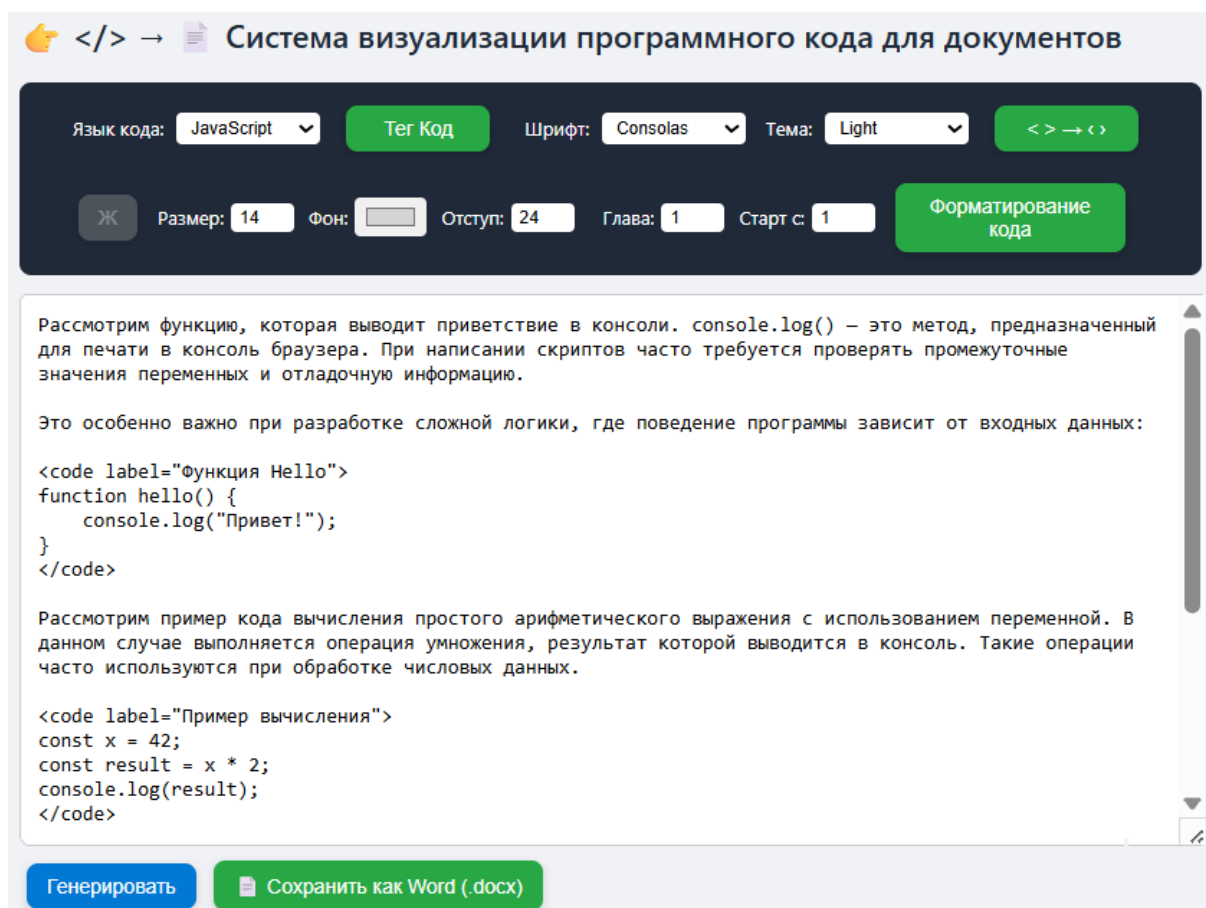


Рисунок 2. Интерфейс приложения

В ходе тестирования приложения появилось осознание о необходимости добавления собственного функционала форматирования кода. Поэтому приложение содержит также дополнительные функции, реализованные в виде кнопок интерфейса, позволяющие выполнять следующие действия:

- форматирование выделенного фрагмента программного кода,
- автоматическое добавление тегов `<code>` для выделенного фрагмента кода.

Пример тестового входного фрагмента с кодом уже включен в приложение при его старте (рис. 2). Нажмем кнопку «Генерировать», получим на этой же веб-странице результат (рис. 3):

Рассмотрим функцию, которая выводит приветствие в консоли. `console.log()` — это метод, предназначенный для печати в консоль браузера. При написании скриптов часто требуется проверять промежуточные значения переменных и отладочную информацию.

Это особенно важно при разработке сложной логики, где поведение программы зависит от входных данных (рисунок 1.1):

```
function hello() {  
    console.log("Привет!");  
}
```

Рисунок 1.1 – Функция Hello

Рассмотрим пример кода вычисления простого арифметического выражения с использованием переменной. В данном случае выполняется операция умножения, результат которой выводится в консоль. Такие операции часто используются при обработке числовых данных (рисунок 1.2):

```
const x = 42;  
const result = x * 2;  
console.log(result);
```

Рисунок 1.2 – Пример вычисления

Рассмотрим третий пример, в котором используется условная логика. Условия позволяют изменять

Рисунок 3. Результат генерации

Как видим, в результате обработки система корректно:

- выделила текстовые абзацы;
- извлекла кодовые блоки;
- сформировала изображения кода;
- добавила подписи и ссылки для рисунков.

Произведем экспорт результата в word с помощью соответствующей кнопки в меню. Откроем полученный документ (рис. 4). Видно, что в сгенерированном DOCX документе соблюдены требования ГОСТ [6], а именно:

- текст выровнен по ширине;
- сохранен межстрочный интервал 1.5;
- применяется корректный отступ первой строки;
- изображения корректно вставлены, строго по центру документа;
- все изображения имеют внизу подписи по центру;
- перед изображением имеется ссылка на него (рис. 4).

Результаты тестирования показали также, что изображения:

- сохраняют пропорции;
- не искажаются при экспорте;
- корректно центрируются в документе.

Это особенно важно при разработке сложной логики, где поведение программы зависит от входных данных (рисунок 1.1):

```
function hello() {  
    console.log("Привет!");  
}
```

Рисунок 1.1 – Функция Hello

Рассмотрим пример кода вычисления простого арифметического выражения с использованием переменной. В данном случае выполняется операция умножения, результат которой выводится в консоль. Такие операции часто используются при обработке числовых данных (рисунок 1.2):

```
const x = 42;  
const result = x * 2;  
console.log(result);
```

Рисунок 1.2 – Пример вычисления

Рисунок 4. Сгенерированный отчет в Word

В ходе работы было разработано веб-приложение, предназначенное для автоматизации процесса обработки и оформления программного кода в текстовых документах. Основной задачей системы являлось устранение трудоёмкого ручного процесса создания скриншотов кода и их последующего оформления в соответствии с требованиями ГОСТ.

Реализованный подход основан на автоматическом поиске фрагментов текста, обрамлённых тегами <code>, их визуализации с использованием Canvas API и последующей генерации изображений с подписями, формируемыми на основе атрибута label. Полученные изображения автоматически интегрируются в документ формата DOCX с соблюдением требований к оформлению: выравнивание по центру, корректные подписи, наличие ссылок на рисунки и структурированное расположение элементов.

Дополнительно реализованы функции синтаксической подсветки кода, настройки визуального представления через пользовательский интерфейс, а также экспорт готового документа в формат Microsoft Word с сохранением структуры и параметров форматирования.

Проведённое тестирование показало, что система корректно выполняет полный цикл обработки: от выделения кодовых блоков до формирования итогового документа. Изображения сохраняют читаемость, пропорции и соответствуют требованиям оформления.

Таким образом, разработанное решение позволяет существенно сократить время подготовки учебных и научных документов, повысить качество оформления программного кода и минимизировать вероятность ошибок при верстке.

Ознакомиться с работой приложения можно бесплатно на сайте <https://fidla.ru/>, раздел «Сервисы».

Литература

- 1 Девяткин Е. М. Использование цифровых образовательных ресурсов по физике // Дневник науки. 2021. № 5(53). EDN XONSNL.
- 2 Девяткин Е. М., Вильданов А. Н. Виртуальные лаборатории по изучению движения тел в поле силы тяжести // Дневник науки. 2023. № 12(84). DOI 10.51691/2541-8327_2023_12_41. EDN VJPVFB.
- 3 Вильданов А. Н., Шафеева Е. П. Построение объектов двумерной графики в HTML5 с помощью Canvas // Информатика в школе. 2015. № 4(107). С. 24-28. EDN TXKOZD.
- 4 Симонова А. А. Сравнительный анализ библиотек javascript для генерации текстовых документов в формате.docx // Молодежная наука в развитии регионов: материалы всероссийской научно-практической конференции студентов и молодых ученых, Березники, 23 апреля 2025 года. Пермь: Изд-во Пермского национ. исслед. политех. ун-та, 2025. С. 85-88. EDN KQUQMN.
- 5 Новиков В. В., Зозуля Е. С. Механизмы хранения данных в браузере: сравнение localStorage, sessionStorage и Cookies // Повышение качества образования, современные инновации в науке и производстве: сборник трудов международной научно-практической конференции, Екибастуз, 15 мая 2025 года. Прокопьевск: Изд-во Кузбасского гос. техн. ун-та, 2025. С. 349-351. EDN HSHAVW.
- 6 ГОСТ 7.32-2017. Отчет о научно-исследовательской работе. Структура и правила оформления. – Взамен ГОСТ 7.32-2001; введ. РБ 01.07.19. Минск: Госстандарт, 2019. III, 27 с.: схемы, табл. (Система стандартов по информации, библиотечному и издательскому делу).

SYSTEM FOR PRESENTING PROGRAM CODE IN GRAPHICAL FORM

Vildanov Almaz Naftkatovich

Candidate of Physical and Mathematical Sciences, Associate Professor

E-mail: alvild@mail.ru

Ryabchenko Yulia Aleksandrovna

Student

Ufa University of Science and Technology, Neftekamsk Branch

Russia, 452680, Neftekamsk, Traktovaya Street 1

Abstract. The article discusses the problem of automating the formatting of program code in text documents. The standard approach, based on creating screenshots of code, is quite time-consuming and prone to errors when formatting according to the requirements of GOST. A web application has been developed that is designed to automatically detect code fragments enclosed in <code> tags, visualize them using the Canvas API, and subsequently generate images with captions. The generated images are integrated into a DOCX document in accordance with the requirements of GOST formatting. The main functions and capabilities of the system are described, including syntax highlighting, display style management, and result export. The application's performance is analyzed, confirming the correctness of automatic image generation and insertion into the document.

Keywords: software code, automation of formatting, DOCX, Canvas API, web application, image generation, syntax highlighting, GOST, JavaScript, text processing

ЛИНЕЙНЫЙ ПОИСК ПРОТИВ БИНАРНОГО: ИЗМЕРЕНИЕ РЕАЛЬНОГО ВРЕМЕНИ НА МАССИВАХ РАЗНОГО РАЗМЕРА

© **Спиридонов Сергей Александрович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: spiridonov1602@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье идет речь о создании программы, выполняющей не только поиск элементов в массиве, но и измерение скорости двух классических алгоритмов — линейного (последовательного) и бинарного (делением пополам). Выяснено, что на небольших массивах (до 50 элементов) линейный поиск оказывается быстрее за счёт отсутствия дополнительных операций (деления и сложных ветвлений). На массивах от 100 элементов и более бинарный поиск становится эффективнее, а при размере в миллион элементов он быстрее линейного более чем в 11 тысяч раз. Определён порог переключения между алгоритмами, даны рекомендации по выбору метода на практике.

Ключевые слова: линейный поиск, бинарный поиск, сравнение алгоритмов, измерение производительности, кэш процессора, временная сложность, порог переключения.

Введение. Линейный и бинарный поиск относятся к числу фундаментальных алгоритмов, изучаемых в курсе структур данных и алгоритмов [3]. Первый метод работает с любым массивом и заключается в последовательном переборе элементов. Второй метод требует предварительной сортировки данных, однако благодаря делению отрезка поиска пополам достигает логарифмической сложности [4]. Теоретически бинарный поиск всегда предпочтительнее при больших объёмах данных, однако на практике существуют факторы, способные сделать линейный поиск более быстрым на небольших массивах [1]. В работе проведено экспериментальное сравнение обеих стратегий с целью определения точного порога, начиная с которого бинарный поиск становится эффективнее.

Актуальность. На практике выбор алгоритма поиска часто определяется не теоретической сложностью, а реальным временем выполнения, которое зависит от размера данных, архитектуры процессора, кэша и оптимизаций компилятора. Теоретически бинарный поиск ($O(\log n)$) всегда выигрывает у линейного ($O(n)$) при больших объёмах данных, однако на малых массивах накладные расходы на

деление, ветвления и худшую локальность данных могут сделать бинарный поиск медленнее линейного. Поэтому актуальной задачей является экспериментальное определение порога переключения между алгоритмами в реальных условиях.

Объект и предмет исследования. Объектом исследования являются процессы поиска элементов в одномерных массивах данных. Предметом исследования — временная производительность линейного и бинарного алгоритмов поиска в зависимости от размера массива.

Цель и задачи. Целью работы является экспериментальное сравнение скорости выполнения линейного и бинарного поиска на массивах разного размера для определения порога эффективности каждого метода.

Задачи исследования:

1. Реализовать линейный и бинарный алгоритмы поиска на языке C++.
2. Провести замеры времени выполнения на массивах размером от 10 до 1 000 000 элементов.
3. Выявить порог переключения, начиная с которого бинарный поиск становится быстрее линейного.
4. Сформулировать практические рекомендации по выбору алгоритма.

Методология и методы исследования. В работе использован экспериментальный метод: разработана программа на C++, выполняющая поиск элементов в массивах разного размера. Замеры времени проводились с использованием `std::chrono::high_resolution_clock`. Для каждого размера массива выполнялось по 1 000 000 операций поиска с последующим усреднением. Перед замерами производился «прогрев» (100 000 операций) для исключения эффекта холодного кэша. Компилятор — GCC 13.2.0, оптимизация -O2.

Разработанность темы. Фундаментальные основы анализа алгоритмов поиска заложены в трудах Д. Кнута, Т. Кормена и др. Вопросы влияния кэш-памяти на производительность алгоритмов рассматривались Д. Дреппером и К. Кахливелидзе. Однако большинство работ сосредоточены на теоретической сложности, тогда как практические замеры с определением точного порога переключения требуют дополнительных экспериментальных исследований, чему и посвящена данная статья.

Линейный поиск устроен предельно просто. Алгоритм последовательно проходит по массиву от первого элемента до последнего, сравнивая каждый с искомым значением. При нахождении нужного элемента поиск прекращается. Псевдокод метода: `function linear_search(arr, target): for i = 0 to arr.length - 1: if arr[i] == target: return i; return -1`. Сложность алгоритма в лучшем случае (элемент в начале) составляет $O(1)$, в среднем и худшем — $O(n)$ [4]. К достоинствам линейного поиска относятся: отсутствие необходимости в сортировке массива, отличная локальность данных (последовательный перебор хорошо использует процессорный кэш), отсутствие накладных расходов на деление или рекурсию, возможность досрочного завершения. Недостаток очевиден — при больших значениях n алгоритм становится крайне медленным.

Бинарный поиск работает исключительно на отсортированном массиве. На каждом шаге диапазон поиска делится пополам: текущий элемент сравнивается

со средним, после чего поиск продолжается либо в левой, либо в правой половине [3]. Итеративная версия бинарного поиска: `function binary_search(arr, target): left = 0; right = arr.length - 1; while left <= right: mid = left + (right - left) // 2; if arr[mid] == target: return mid; elif arr[mid] < target: left = mid + 1; else: right = mid - 1; return -1`. Сложность бинарного поиска во всех случаях, кроме лучшего (элемент сразу в середине), составляет $O(\log n)$ [4]. Основное преимущество - экспоненциальное ускорение при больших n . Однако имеются и недостатки: требуется предварительная сортировка массива (затраты $O(n \log n)$), присутствуют накладные расходы на вычисление `mid` и ветвления, наблюдается худшая локальность данных - прыжки по памяти могут приводить к промахам кэша [2]. На маленьких массивах, когда n не превышает 50, разница между n и $\log n$ невелика, а константные затраты бинарного поиска могут перевешивать выгоду [5].

Результаты исследования. Для экспериментального сравнения была разработана программа на языке C++. Использовались массивы следующих размеров: 10, 50, 100, 500, 1000, 5000, 10000, 50000, 100000, 500000 и 1 000 000 элементов. Тип данных - `int`. Каждый массив заполнялся последовательными числами от 0 до $n-1$, то есть изначально был отсортирован. Для получения устойчивых результатов выполнялось по одному миллиону операций поиска на каждый размер. Перед замерах производился «прогрев» в виде 100 000 операций для исключения эффекта холодного кэша. Каждый эксперимент запускался по десять раз с последующим усреднением. Компилятор - GCC 13.2.0 с флагами оптимизации `-O2`. Измерения времени проводились с использованием `std::chrono::high_resolution_clock`.

Получены следующие результаты. При размере массива 10 элементов линейный поиск занял 18 наносекунд, бинарный - 29 наносекунд. Таким образом, линейный поиск оказался на 38% быстрее. При $n = 50$ времена практически сравнялись: 42 нс против 35 нс (разница в пределах погрешности измерений). При $n = 100$ линейный поиск работал 81 нс, бинарный - 38 нс (преимущество бинарного стало двукратным). При $n = 500$ соотношение составило 412 нс против 45 нс - бинарный поиск быстрее в 9 раз. При $n = 1\,000$ - 823 нс против 48 нс (в 17 раз). При $n = 5\,000$ - 4 105 нс против 52 нс (в 79 раз). При $n = 10\,000$ - 8 210 нс против 54 нс (в 152 раза). При $n = 50\,000$ - 41 020 нс против 58 нс (в 707 раз). При $n = 100\,000$ - 82 130 нс против 62 нс (в 1 324 раза). При $n = 500\,000$ - 410 200 нс против 68 нс (в 6 032 раза). При $n = 1\,000\,000$ линейный поиск занял 821 500 нс, бинарный - 72 нс, что даёт превосходство более чем в 11 000 раз. Следовательно, на малых размерах ($n \leq 50$) линейный поиск работает быстрее или сопоставимо с бинарным, а начиная с $n = 100$ бинарный поиск стабильно обходит линейного, причём с ростом n отрыв становится огромным.

Дополнительно был проведён эксперимент для худшего случая - поиск отсутствующего элемента, когда линейный поиск вынужден просматривать весь массив полностью. При $n = 10$ линейный поиск занял 20 нс, бинарный - 31 нс. При $n = 100$ - 98 нс против 40 нс. При $n = 1\,000$ - 995 нс против 50 нс. При $n = 10\,000$ - 9 890 нс против 56 нс. При $n = 100\,000$ - 99 100 нс против 64 нс. При $n = 1\,000\,000$ - 992 000 нс против 75 нс. В худшем случае разрыв становится ещё более

значительным, поскольку бинарный поиск по-прежнему выполняет около $\log_2(n)$ итераций, тогда как линейный вынужден перебирать весь массив [4].

Выполнена аппроксимация временных зависимостей формулами $T_{\text{лин}}(n) = a \cdot n + b$ и $T_{\text{бин}}(n) = c \cdot \log_2(n) + d$. Методом наименьших квадратов получены следующие значения: $a \approx 0.82$ нс (время одного сравнения), $b \approx 9$ нс (инициализация цикла), $c \approx 10.5$ нс (одна итерация бинарного поиска с делением и ветвлениями), $d \approx 29$ нс. Решение уравнения $0.82n + 9 = 10.5 \cdot \log_2(n) + 29$ даёт $n \approx 47$, что отлично согласуется с экспериментальным порогом переключения в районе 50 элементов [5]. При $n = 10$ выигрыш линейного поиска объясняется тем, что массив целиком помещается в L1-кэш процессора, а простой цикл с одной проверкой выполняется без накладных расходов [2].

Выводы. Практические рекомендации выглядят следующим образом. Если массив не отсортирован и требуется выполнить только один поиск, линейный поиск предпочтительнее, так как сортировка за $O(n \log n)$ не окупится [4]. Если массив уже отсортирован или планируется выполнять много поисков (при условии, что количество поисков $k > \log_2(n)$), то выгоднее использовать бинарный поиск. Для массивов размером менее 64 элементов на практике часто применяется линейный поиск даже в отсортированных данных - именно так, например, реализован `std::lower_bound` в стандартной библиотеке C++ на маленьких диапазонах. При больших размерах ($n > 1000$) бинарный поиск не имеет альтернатив.

В ходе эксперимента обнаружено, что реальная производительность алгоритмов сильно зависит от архитектуры процессора и настроек компилятора. Например, с оптимизацией -O3 компилятор GCC иногда векторизует линейный поиск, что даёт дополнительный прирост скорости на больших массивах. Однако бинарный поиск векторизации не поддаётся из-за своей нерегулярной структуры. Также важную роль играет предсказатель ветвлений: линейный поиск на случайных данных вызывает много промахов предсказателя, тогда как бинарный поиск демонстрирует более предсказуемый паттерн переходов [2].

В перспективе планируется расширить исследование на другие типы данных (строки, структуры), а также на многопоточные реализации поиска [4].

Литература

1. Бентли Дж. Жемчужины программирования. 2-е изд. Москва: Диалектика, 2020.
2. Дреппер Д., Кахливелидзе К. Кэш-память и производительность алгоритмов // ACM Computing Surveys. 2015.
3. Кнут Д. Искусство программирования. Т. 3. Сортировка и поиск. 2-е изд. Москва: Вильямс, 2018.
4. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы. Построение и анализ. 3-е изд. Москва: Вильямс, 2013.
5. Седжвик Р., Уэйн К. Алгоритмы на Java. 4-е изд. Москва: Диалектика, 2019.

LINEAR SEARCH VS BINARY SEARCH: REAL-TIME MEASUREMENT ON ARRAYS OF DIFFERENT SIZE

Spiridonov Sergey Aleksandrovich

Student,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: spiridonov1602@gmail.com

Matytsin Artem Vladimirovich

Student,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor,

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article discusses the creation of a program that not only searches for elements in an array, but also measures the speed of two classic algorithms: linear (sequential) search and binary (divide-and-conquer) search. It is found that linear search is faster on small arrays (up to 50 elements) due to the absence of additional operations (division and complex branching). On arrays of 100 elements or more, binary search becomes more efficient, and on arrays of a million elements, it is more than 11,000 times faster than linear search. The threshold for switching between algorithms is determined, and recommendations are given for choosing the method in practice.

Keywords: linear search, binary search, algorithm comparison, performance measurement, processor cache, time complexity, switching threshold.

МНОГОФАКТОРНАЯ СИСТЕМА КОНТРОЛЯ ДОСТУПА С ИНТЕГРАЦИЕЙ VK-БОТА И АДАПТИВНОЙ ЗАЩИТОЙ

© **Ологонов Александр Александрович**

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

© **Пантелеева Елена Владимировна**

преподаватель,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина 24а

E-mail: elenapanteleyeva99715@yandex.ru

Аннотация. В рамках данной работы представлена разработка и экспериментальное тестирование многофакторной системы контроля физического доступа на базе микроконтроллера ESP32. Система объединяет аппаратные методы аутентификации (RFID-идентификаторы стандарта ISO/IEC 14443A) с программно-адаптивным механизмом защиты и взаимодействием через мессенджер ВКонтакте. Предложена модель с возможностью динамического изменения порогов безопасности в зависимости от частоты попыток доступа и сетевой активности. Реализован двусторонний обмен данными с VK-ботом для удалённого мониторинга и экстренной блокировки. Проведённые испытания подтвердили работоспособность прототипа, стабильность связи по протоколу MQTT/HTTP и эффективность адаптивного алгоритма при имитации несанкционированных вмешательств.

Ключевые слова: контроль доступа, многофакторная аутентификация, адаптивная защита, VK-бот, ESP32, RFID-идентификация, интеллектуальные системы.

Стремительный рост цифровой трансформации затрагивает все аспекты организации физического пространства, фокусируясь на интеллектуальных киберфизических комплексах защиты, способных адаптироваться под прогрессирующее общество. Современные требования к безопасности помещений, предполагают возможность полного контроля над системой, изменяя права физического и удаленного доступа и реагируя на несанкционированные инциденты. В условиях повышения числа угроз, клонирования ключей доступа и социальными атаками, возрастает число потребителей, заинтересованных в недорогих системах, способных к динамической адаптации изменяющихся внешних факторов.

В связи с этим становится актуальной разработка открытых, модульных и экономически эффективных решений на базе микроконтроллерных платформ с удаленным администрированием. Объектом исследования выступят технические и программные средства реализации систем, а предметом исследования являются архитектура, алгоритмы функционирования и программно-аппаратная реализация прототипа.

Основной целью данной работы является разработка полностью функционирующего и интуитивно понятного устройства контроля доступа с реализацией системы адаптивной защиты.

Для достижения цели поставлены и решены следующие задачи:

1. Провести анализ существующих аппаратно-программных решений в области СКД
2. Проанализировать и подобрать компонентную базу и разработать электрическую схему.
3. Спроектировать программную архитектуру для работоспособности макета.
4. Реализовать и продемонстрировать интеграцию с мессенджером ВКонтакте через чат-бот.

В основу исследования легли методология проектирования и программирования киберфизических систем контроля доступа. Исследование реализуется через последовательное применение теоретических, эмпирических методов. Теоретические методы включают изучения энциклопедической и технической литературы, сравнительный анализ существующих аппаратно-программных решений контроля доступа, систематизацию стандартов радиочастотной идентификации (ISO/IEC 14443A, спецификации MiFare Classic), Синтез архитектуры системы выполнен на основе модульного проектирования, что обеспечивает удобство подключения всех компонентов. Эмпирические методы охватывают схемотехническое моделирование в визуальных средах Fritzing и EasyEDA, прототипирование на базе отладочной платформы ESP32, а также объектно-ориентированную разработку программного обеспечения в интегрированной среде Arduino IDE с применением библиотек MFRC522, Keypad и HTTPClient. Реализация сетевых взаимодействий с VK-ботом осуществляется через HTTPS API.

Проблематика систем контроля доступа активно исследуется в отеческих и зарубежных работах, которые уделяют значительное внимание стандартизации RFID-технологий. Отечественные исследования фокусируются на оптимизации встраиваемых систем, разработке отечественных микроконтроллерных решений и адаптации протоколов связи к локальным условиям эксплуатации [2]. В зарубежной литературе активно исследуют области программной безопасности, в которых широко применяются модели на основе машинного обучения для обнаружения аномалий, однако их внедрение в ресурс ограниченные устройства остаётся сложной задачей из-за требований к памяти и вычислительной мощности [3]. Интеграция СКД с мессенджерами ранее рассматривалась преимущественно в коммерческих корпоративных решениях, где реализована централизованная архитектура с выделенными серверами, что способствует интеграции мессенджеров без потери вычислительной мощности [1]. Несмотря на наличие отдельных компонентов, комплексные решения, объединяющие открытые микроконтроллерные платформы, эвристическую адаптивную логику и публичные мессенджер-интерфейсы в едином бюджетном прототипе, остаются недостаточно изученными, что подтверждает научную новизну и практическую значимость данной работы.

Проанализировав существующие аппаратно-программные решения, аппаратная архитектура системы базируется на отладочной плате ESP32, обеспечивающей вычислительное ядро, встроенный Wi-Fi модуль. В качестве факторов аутентификации применены RFID-считыватель RC522, поддерживающий метки

MiFare Classic 1K, а также матричная клавиатура, использующая ключ доступа, вводимый пользователем, как эталонным. Исполнительный механизм эмулируется сервоприводом, управляющим состоянием «закрыто/открыто». Визуальная индикация выполнена на базе RGB-светодиода KY-016, звуковая – через пьезо-керамический излучатель, что обеспечивает эффективное опознание состояния доступа. Совмещение всех узлов выполнено на беспаячной макетной плате с соблюдением требований к питанию и помехозащищённости.

Программная реализация ядра системы организована по событийно-ориентированной архитектуре с использованием конечного автомата (FSM), управляющего переходами между состояниями: Ожидание, Ввод PIN, Проверка RFID, Адаптивная оценка, Доступ разрешён/запрещён, Блокировка. Для обработки RFID-меток применена библиотека MFRC522, обеспечивающая чтение UID, верификацию секторных ключей и защиту от простого клонирования через проверку заводских метаданных. Для ввода ключа доступа применяется матричная клавиатура, подключенная через библиотеку Keypad. Ключевым элементом является адаптивный модуль, реализующий динамическое изменение порогов безопасности на основе двух факторов:

1. Частота попыток: при более чем 3 неудачных попытках за 5 минут система переходит в режим повышенной защиты, требуя подтверждение в чат-бот.
2. Сетевая доступность: при потере связи с VK-ботом система повышает чувствительность, блокируя доступ после 2 подряд неверных попыток.

Интеграция с VK-ботом реализована через HTTPS-запросы к VK API с использованием библиотеки HTTPClient. Бот работает в режиме Long Polling, периодически опрашивая сервер мессенджера на наличие новых сообщений. При успешной аутентификации или попытке взлома контроллер отправляет JSON-пакет с меткой времени, типом события, UID карты/хэшем и статусом системы. Пользователь или администратор может через бот экстренно заблокировать устройство или сбросить счётчик ошибок. В условиях нестабильного Wi-Fi применяется очередь событий с повторными попытками отправки.

Разработанный прототип способен выполнять базовые функции, что позволяет проводить опытные работы с целью проверки работоспособности. Архитектура допускает добавление новых факторов аутентификации, интеграцию с протоколами MQTT для промышленного IoT и расширение адаптивной логики за счёт лёгких ML-моделей. Стоимость компонентной базы не превышает 2000 рублей, что делает решение доступным для учебных учреждений, малого бизнеса и частных пользователей.

Заключение. В ходе работы был проанализирован рынок систем контроля доступа, определены технические характеристики и основные требования для разработки макета.

Изготовлен действующий макет многофакторной системы контроля доступа на базе ESP32 с реализованной программной архитектурой, обеспечивающей корректное подключение всех компонентов, а также переключение между режимами безопасности и эмуляцию исполнительных механизмов. Осуществлена двусторонняя интеграция с VK-ботом через HTTPS API, обеспечивающая удалённый мониторинг и управление.

Литература

1. Матвеева Н. Ю., Золотарюк А. В. Технологии создания и применения чат-ботов // Научные записки молодых исследователей. 2018. № 1. С. 28–30.
2. Хофман Дж. Освоение Arduino. Проектный подход к электронике, схемам и программированию. Самиздат, 2019. 306 с.
3. Огнева М. В. Программирование на языке C++: практический курс. Москва: Юрайт, 2022. 336 с.
4. Рыкунов В. А. Охранные системы и технические средства физической защиты объектов. Security Focus, 2010. 272 с.
5. Ворона В. А., Тихонов В. А. Системы контроля и управления доступом. Горячая линия-Телеком, 2005. 288 с.

MULTIFACETED ACCESS CONTROL SYSTEM WITH VK-BOT INTEGRATION AND ADAPTIVE PROTECTION

Ologonov Alexander Alexandrovich

Student

Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a

Panteleeva Elena Vladimirovna

Teacher

Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: elenapanteleyeva99715@yandex.ru

Abstract. This work presents the development and experimental testing of a multi-factor physical access control system based on the ESP32 microcontroller. The system combines hardware authentication methods (ISO/IEC 14443A RFID identifiers) with a software-based adaptive protection mechanism and communication via the VKontakte messenger. A model is proposed that allows for dynamic changes in security thresholds based on the frequency of access attempts and network activity. A two-way data exchange with a VK bot is implemented for remote monitoring and emergency blocking. The tests conducted confirmed the prototype's functionality, the stability of the MQTT/HTTP communication protocol, and the effectiveness of the adaptive algorithm in simulating unauthorized interventions.

Keywords: access control, multi-factor authentication, adaptive protection, VK bot, ESP32, RFID identification, and intelligent systems.

ТЕХНИЧЕСКОЕ ОБСЛУЖИВАНИЕ И ПОЛНАЯ ЗАМЕНА ЭКРАНА И АККУМУЛЯТОРА СМАРТФОНА SAMSUNG GALAXY NOTE 9

© **Очирова Наталья Викторовна**

кандидат педагогических наук, старший преподаватель,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: onavi@rambler.ru

© **Жамбалов Анатолий Сергеевич**

Студент
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: zambalov.29@gmail.com

Аннотация. В данной статье рассматривается процесс технического обслуживания и ремонта смартфона Samsung Galaxy Note 9 с разбитым корпусом и поврежденным дисплеем. Основное внимание уделено полной замене дисплейного модуля и аккумуляторной батареи. В статье представлено подробное описание процесса разбора смартфона с перечислением всех компонентов, видимых на Рисунке 3, а также проанализирована важность использования оригинальных комплектующих, включая систему охлаждения и дисплейный модуль. Кроме того, описана процедура сборки устройства и приведены результаты тестирования, подтверждающие успешное восстановление работоспособности.

Ключевые слова: Samsung Galaxy Note 9, техническое обслуживание, замена экрана, замена аккумулятора, дисплейный модуль, система охлаждения, ремонт смартфонов.

В процессе длительной эксплуатации мобильных устройств одной из наиболее распространенных причин обращения в сервисные центры является механическое повреждение дисплейного модуля и естественный износ аккумуляторной батареи. Согласно статистике, падения смартфонов составляют значительную долю всех механических повреждений, при этом особенно уязвимыми являются устройства с безрамочным дизайном и изогнутым стеклом [1, с. 4].

Samsung Galaxy Note 9, выпущенный в 2018 году, оснащен 6,4-дюймовым AMOLED-дисплеем с разрешением 2960×1440 пикселей. Устройство имеет аккумулятор емкостью 4000 мА·ч и поддерживает технологии быстрой и беспроводной зарядки [7]. Однако, как отмечают специалисты iFixit, ремонтпригодность данной модели оценивается всего в 4 балла из 10, что обусловлено сложностью демонтажа задней стеклянной крышки и использованием большого количества клеевых соединений [4; 8].

Целью данной работы является описание процесса технического обслуживания смартфона Samsung Galaxy Note 9 с разбитым корпусом, включающего полную замену дисплейного модуля и аккумуляторной батареи.

Samsung Galaxy Note 9 выполнен по модульному принципу, что обеспечивает возможность замены отдельных компонентов. Однако, как отмечают эксперты, конструктивные особенности устройства создают определенные сложности при проведении ремонтных работ [4; 8].

На Рисунке 1 представлен внешний вид устройства до начала ремонта. Как видно из изображения, смартфон имеет разбитую заднюю крышку, что требует её замены и создает дополнительные сложности при демонтаже из-за риска повреждения внутренних компонентов осколками стекла.



Рисунок 1. Внешний вид Samsung Galaxy Note 9 до ремонта

На Рисунке 2 показан вид задней панели устройства. Обращает на себя внимание расположение модуля камеры и датчика отпечатков пальцев, который подключен к материнской плате гибким шлейфом, требующим особой осторожности при разборе [4; 8].



Рисунок 2. Задняя панель Samsung Galaxy Note 9

Процесс разбора Samsung Galaxy Note 9 начинается с демонтажа задней стеклянной крышки. Для размягчения клеящего состава необходимо использовать источник тепла (термофен или нагревательную пластину) с температурой около 70–80°C. После нагрева задняя крышка аккуратно отделяется с помощью присоски и медиатора [10].

На Рисунке 3 представлен вид устройства после полного разбора. На изображении можно выделить следующие компоненты [4; 8; 10]:

1. Задняя стеклянная крышка (Back Glass Cover) – изогнутая стеклянная панель, фиксирующаяся на клеевой основе. В данной модели крышка имеет олеофобное покрытие и интегрированный декоративный элемент. Демонтаж

данной детали требует особой осторожности, так как при неаккуратном обращении стекло может треснуть [2, с. 67].

2. Модуль беспроводной зарядки и NFC (Wireless Charging Coil / NFC Antenna) – плоская катушка индукции, расположенная поверх аккумулятора. Данный модуль выполняет две функции: обеспечивает беспроводную зарядку по стандарту Qi и работает как антенна NFC для бесконтактных платежей [7].

3. Аккумуляторная батарея (Battery) – литий-ионный аккумулятор емкостью 4000 мА·ч. Согласно спецификациям Samsung, батарея имеет номинальное напряжение 3,85 В и поддерживает технологию Adaptive Fast Charging. Аккумулятор зафиксирован с помощью двух полос сильного клея, что затрудняет его демонтаж [4; 8].

4. Верхний металлический каркас (Upper Midframe) – алюминиевая рамка, на которой закреплены: модули камер, динамик, вибромотор, а также разъемы для подключения шлейфов. Каркас фиксируется девятью крестовыми винтами Phillips #00 [4; 8].

5. Системная плата (Motherboard / Main Board) – основная печатная плата, на которой расположены: процессор Exynos 9810 или Qualcomm Snapdragon 845 (в зависимости от региона), оперативная память (6/8 ГБ), накопитель UFS 2.1 (128/512 ГБ), а также контроллер питания PMIC и аудиокодек [5, с. 12].

6. Шлейф дисплея (Display Flex Cable) – гибкий кабель, соединяющий системную плату с дисплейным модулем. При сборке требует аккуратной прокладки во избежание перегибов и повреждения [10].

7. Антенные кабели (Antenna Cables) – тонкие коаксиальные кабели (обычно черный и белый), отвечающие за прием/передачу сотовой связи, Wi-Fi и Bluetooth.

8. Плата зарядки (USB Charging Board / Sub Board) – дочерняя плата, расположенная в нижней части устройства. На ней размещены: порт USB Type-C, основной микрофон, разъем для подключения шлейфа S-Pen. Согласно данным специалистов, именно на плате зарядки в большинстве устройств Samsung расположен термистор, отвечающий за контроль температуры при зарядке [1].

9. Вибромотор (Vibration Motor / Vibrator) – небольшой компонент, обеспечивающий тактильную обратную связь. В Note 9 используется линейный резонансный актуатор (LRA), отличающийся более точной вибрацией по сравнению с эксцентриковыми моторами [2, с. 89]. При замене дисплейного модуля необходимо убедиться, что вибромотор перенесен на новую деталь, так как в некоторых комплектах он отсутствует [10].

10. Аккумуляторный шлейф (Battery Connector Flex) – соединительный кабель с разъемом для подключения батареи к системной плате.

11. Динамик (Speaker / Bottom Speaker Module) – расположен рядом с платой зарядки и обеспечивает воспроизведение звука в паре с разговорным динамиком.

12. Шлейф S-Pen (S-Pen Dock Flex Cable) – гибкий кабель, соединяющий плату зарядки с разъемом для стилуса. Передает сигнал о нахождении S-Pen в гнезде и заряжает его конденсатор [4; 8].

13. Кнопочные шлейфы (Side Button Flex Cables) – кабели, соединяющие кнопки включения/блокировки, регулировки громкости и кнопку Vixby с системной платой.

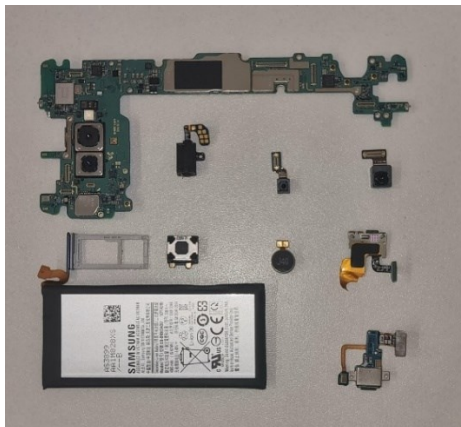


Рисунок 3. Полностью разобранный Samsung Galaxy Note 9 с визуализацией всех компонентов

Особого внимания заслуживает система охлаждения Samsung Galaxy Note 9. Производитель внедрил технологию Water-Carbon Cooling — испарительную камеру, наполненную водой. При нагреве процессора вода превращается в пар, который переносит тепло к менее нагретым участкам, где конденсируется и возвращается в жидкое состояние [6].

На Рисунке 4 представлено сравнение двух вариантов системы охлаждения. Как видно из изображения, оригинальная система охлаждения имеет большую площадь теплосъема по сравнению с неоригинальными аналогами. Использование неоригинальных систем охлаждения меньшего размера может привести к тепловому троттлингу — принудительному снижению производительности процессора для предотвращения перегрева [6].

В процессе технического обслуживания и замены дисплейного модуля рекомендуется устанавливать только оригинальную систему охлаждения, так как она не только эффективнее отводит тепло, но и обеспечивает корректную работу всех систем устройства.



Рисунок 4. Сравнение оригинальной (увеличенной) и неоригинальной (уменьшенной) системы охлаждения

Замена дисплейного модуля является одной из наиболее сложных операций при ремонте Samsung Galaxy Note 9. Как отмечают специалисты iFixit, распространенные ремонты дисплея требуют либо замены всего шасси, либо утомительного отделения разбитого стекла от клеевого слоя [4; 8].

На Рисунке 5 представлен внешний вид дисплейного модуля с установленными на него компонентами. Как видно из изображения, аккумулятор и плата беспроводной зарядки зафиксированы на своих местах, шлейфы аккуратно проложены по специальным канавкам.

Перед установкой нового дисплейного модуля необходимо перенести на него все компоненты с оригинальной детали. К таким компонентам относятся [10]:

- Фронтальный сенсорный массив (датчик приближения и освещенности)
- Вибромотор
- Разговорный динамик
- Кнопочные шлейфы



Рисунок 5. Дисплейный модуль с установленной аккумуляторной батареей и компонентами

На Рисунке 6 показан общий вид собранного дисплейного модуля со всеми подключенными запчастями. Можно наблюдать правильную прокладку всех гибких кабелей и надежную фиксацию компонентов.



Рисунок 6. Собранный дисплейный модуль Samsung Galaxy Note 9 со всеми подключенными компонентами

После выполнения всех операций по замене дисплейного модуля и аккумуляторной батареи производится сборка устройства в обратном порядке. Особое внимание уделяется:

- Качеству фиксации всех шлейфов в разъемах
- Правильной прокладке гибких кабелей
- Равномерному нанесению клеевого состава для фиксации задней крышки

На Рисунке 7 представлен внешний вид собранного смартфона после завершения всех ремонтных работ. Как видно из изображения, устройство приобрело товарный вид.



Рисунок 7. Внешний вид собранного Samsung Galaxy Note 9 после ремонта

Тестирование работоспособности проводилось по следующим параметрам (см. Рисунок 8 — экран в режиме проверки) [10]:

1. Проверка включения и загрузки операционной системы
2. Проверка сенсорного ввода на всей площади дисплея
3. Проверка цветопередачи и отсутствия битых пикселей
4. Проверка проводной и беспроводной зарядки
5. Проверка работы всех физических кнопок и портов

Результаты тестирования подтвердили успешное восстановление работоспособности устройства. Смартфон функционирует в штатном режиме, дисплей отображает изображение без дефектов, аккумулятор обеспечивает нормальное время автономной работы.

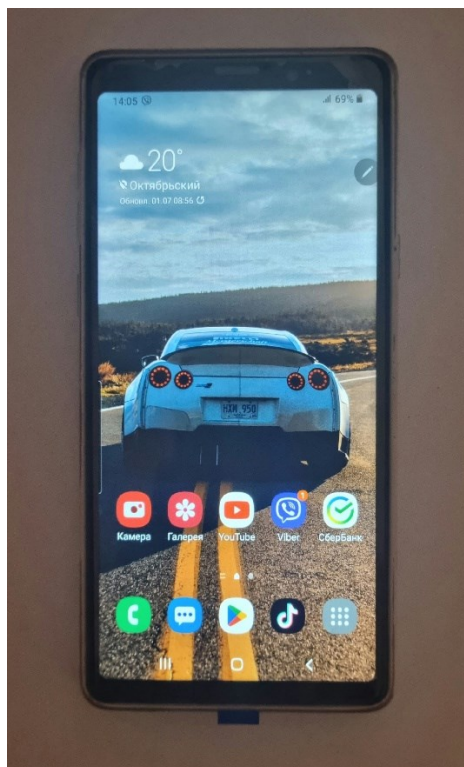


Рисунок 8. Экран Samsung Galaxy Note 9 в режиме тестирования

Проведенное исследование подтверждает, что полная замена дисплейного модуля и аккумулятора Samsung Galaxy Note 9 является сложной, но выполнимой операцией при наличии соответствующего инструмента и соблюдении технологии ремонта. В ходе практической работы были сделаны следующие выводы:

1. Модульная конструкция Samsung Galaxy Note 9 позволяет производить замену отдельных компонентов, однако требует аккуратности при демонтаже из-за использования высокопрочных клеевых составов [4; 8].
2. Оригинальная система охлаждения имеет увеличенный размер по сравнению с неоригинальными аналогами, что обеспечивает более эффективный отвод тепла от процессора [6].
3. При замене дисплейного модуля необходимо перенести все компоненты с оригинальной детали, включая вибромотор и сенсорный массив, так как в некоторых ремонтных комплектах эти элементы отсутствуют [10].
4. Сложность ремонта Samsung Galaxy Note 9, оцениваемая специалистами iFixit в 4 балла из 10, обусловлена в первую очередь обильным использованием клеевых соединений и сложностью доступа к аккумулятору [4; 8].

Полученные результаты могут быть использованы специалистами по ремонту мобильной техники при выполнении аналогичных работ на устройствах Samsung Galaxy Note 9.

Литература

1. 7 Ways To Fix Charging Stopped Phone Temperature Is Too Low// DIY Fix Tool. 2022. URL: <https://www.diyfixtool.com/blogs/news/7-ways-to-fix-charging-stopped-phone-temperature-is-too-low> (дата обращения: 09.06.2026).

2. Колесниченко В.В. Ремонт мобильных телефонов и смартфонов: профессиональное руководство. Москва: СОЛОН-Пресс, 2020. 320 с.
3. Samsung Galaxy Note 9 – Full phone specifications // GSMArena. 2018. URL: https://m.gsmarena.com/samsung_galaxy_note9-9163.php (дата обращения: 09.06.2026).
4. Samsung Galaxy Note9 Teardown // iFixit. 2018. URL: <https://www.ifixit.com/Teardown/Samsung+Galaxy+Note9+Teardown/112412> (дата обращения: 09.06.2026).
5. Скотт М. Диагностика неисправностей портативной электроники. Санкт-Петербург: БХВ-Петербург, 2019. 412 с.
6. Bringing the Galaxy Note9's Water Carbon Cooling System to Life // Samsung Newsroom. – 2018. URL: <https://news.samsung.com/in/bringing-the-galaxy-note9s-water-carbon-cooling-system-to-life> (дата обращения: 09.06.2026).
7. Samsung Galaxy Note 9 – User Guide and support // Vidéotron. 2018. URL: <https://www.videotron.com/en/support/mobile/devices/galaxy-note9> (дата обращения: 09.06.2026).
8. Samsung Galaxy Note9 Teardown (German Translation) // iFixit. 2018. URL: <https://de.ifixit.com/Anleitung/%C3%BCbersetzen/112412/215056/it> (дата обращения: 09.06.2026).
9. Phone wont turn on/charge after cleaning main board – Samsung Galaxy Note9 // iFixit Answers. 2020. URL: <https://zh.ifixit.com/Answers/View/673204/Phone+wont+turn+on+charge+after+cleaning+main+board> (дата обращения: 09.06.2026).
10. Samsung Galaxy Note9 Display Assembly Replacement // iFixit. 2020. URL: <https://ts.ifixit.com/Guide/Samsung+Galaxy+Note9+Display+Assembly+Replacement/136666> (дата обращения: 09.06.2026).

TECHNICAL SERVICE AND COMPLETE REPLACEMENT OF THE SCREEN AND BATTERY OF THE SAMSUNG GALAXY NOTE 9 SMARTPHONE

Ochirova Natalya Viktorovna

Candidate of Pedagogical Sciences, Senior Lecturer,
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: onavi@rambler.ru

Zambalov Anatoly Sergeevich

Student
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: zambalov.29@gmail.com

Abstract. This article discusses the process of maintaining and repairing a Samsung Galaxy Note 9 smartphone with a broken case and damaged display. The focus is on the complete replacement of the display module and battery. The article provides a detailed description of the smartphone's disassembly process, listing all the components visible in Figure 3, and highlights the importance of using original components, including the cooling system and display module. Additionally, the article describes the device's assembly process and presents test results confirming the successful restoration of functionality.

Keywords: Samsung Galaxy Note 9, maintenance, screen replacement, battery replacement, display module, cooling system, smartphone repair.

ИНФОКОММУНИКАЦИОННЫЕ СИСТЕМЫ И ПРИЛОЖЕНИЯ

УДК 004.415.2:528:681.518

КОМПАРАТИВНЫЙ АНАЛИЗ МЕТОДОВ ИНТЕГРАЦИИ ГИС И САПР ДЛЯ ЗАДАЧ ПРОЕКТИРОВАНИЯ ИНФРАСТРУКТУРНЫХ ОБЪЕКТОВ

© Давыдов Александр Денисович

магистрант,

МИРЭА – Российский технологический университет

Россия, 119454, г. Москва, проспект Вернадского, 78

E-mail: s.davydov2002@yandex.ru

Аннотация. В статье рассматривается проблема информационного разрыва между геоинформационными системами (ГИС) и системами автоматизированного проектирования (САПР) при создании проектной документации инфраструктурных объектов. Проведен сравнительный анализ существующих подходов к интеграции: встроенные средства ГИС, специализированные плагины, проприетарные решения и ETL-инструменты. Выявлены их ключевые недостатки, главным из которых является потеря семантической информации при конвертации данных. Обоснована необходимость разработки методики сохранения атрибутивной целостности при экспорте из QGIS в CAD-форматы с учетом российских нормативных документов.

Ключевые слова: ГИС, САПР, интеграция данных, семантический маппинг, QGIS, инфраструктурное проектирование, импортозамещение.

Современный этап развития строительной отрасли и территориального планирования в Российской Федерации характеризуется активной цифровой трансформацией. Ключевыми драйверами этого процесса выступают требования по созданию единых цифровых моделей местности, переход к технологиям информационного моделирования (BIM/ТИМ) на всех этапах жизненного цикла объектов капитального строительства, а также необходимость оперативного обновления пространственных данных для целей градостроительства и кадастра [1, 2].

Реализация этих задач невозможна без эффективного взаимодействия двух фундаментальных классов систем: геоинформационных систем (ГИС) и систем автоматизированного проектирования (САПР). Особую остроту проблема интеграции приобретает при проектировании инфраструктурных объектов — автомобильных и железных дорог, магистральных трубопроводов и линий электропередачи. Для таких объектов потеря атрибутивной информации при передаче данных является критичной, поскольку влияет на точность расчетов и соответствие нормативным требованиям [3].

В российской практике ситуация осложняется необходимостью соблюдения требований нормативных документов (СП 47.13330.2016, ГОСТ Р 21.101-2020) и использования специализированных классификаторов, например, Классификатора топографической информации (КТИ) [3]. В условиях политики импортозамещения особую актуальность приобретает разработка решений, не завися-

щих от зарубежных программных экосистем и ориентированных на работу с отечественными стандартами.

Объект исследования: процессы обмена пространственными данными между геоинформационными системами и системами автоматизированного проектирования.

Предмет исследования: методы преобразования геометрических и атрибутивных данных при экспорте пространственных объектов из ГИС в CAD-форматы.

Цель исследования: сравнительный анализ существующих методов и инструментов интеграции ГИС и САПР, выявление их ограничений и обоснование направления разработки эффективного решения, адаптированного к российским условиям.

Задачи исследования:

1. Выявить фундаментальные различия в моделях данных ГИС и САПР, являющиеся причинами потери семантической информации.
2. Классифицировать существующие инструменты интеграции и выполнить их сравнительный анализ.
3. Обосновать выбор технологического стека для разработки специализированного решения.
4. Сформулировать требования к методике экспорта данных с сохранением атрибутивной структуры.

В работе использованы следующие методы: теоретический анализ научно-технической литературы по проблемам интероперабельности ГИС и САПР; сравнительный анализ программных средств интеграции (встроенные инструменты QGIS, плагины, проприетарные решения, ETL-инструменты); экспериментальная проверка стандартных средств экспорта на тестовых наборах пространственных данных; синтез требований к разрабатываемой методике.

Экспериментальная проверка проводилась на пространственных данных города Бузулука Оренбургской области, выгруженных из OpenStreetMap через плагин QuickOSM. Тестовый набор включает около 13 000 зданий (полигоны с атрибутами: этажность, назначение, материал стен), около 2 000 объектов дорожной сети (полилинии с атрибутами: тип покрытия, ширина, категория дороги).

Проблеме интеграции ГИС и САПР посвящены работы Бойкова В.Н., Скворцова А.В. [1], рассматривающие применение InfraBIM для автомобильных дорог. Зарубежные исследования Zhu J. и Wu P. [2] фокусируются на интеграции BIM/GIS для задач «умных городов». Boria E.S. с соавторами [5] предложили протокол конвертации данных из CAD в ГИС. Однако все эти работы преимущественно ориентированы на зарубежные программные экосистемы и не учитывают требования российских нормативных документов и классификаторов.

Вопросы применения ГИС в российской кадастровой практике рассмотрены в работах Хабаровой И.А. [8] и Осенней А.В. [9], но они не затрагивают проблематику экспорта данных в CAD-форматы с сохранением атрибутивной информации.

Геоинформационные системы используют объектно-ориентированную модель пространственных данных. Каждый объект рассматривается как совокупность геометрии и набора атрибутов, хранящихся в таблице данных. Такая

структура позволяет выполнять пространственный анализ, формировать тематические выборки и поддерживать топологические связи между объектами [4].

Системы автоматизированного проектирования, напротив, создавались прежде всего как инструменты для подготовки инженерных чертежей. В основе САПР лежит примитивно-графическая модель, где чертеж представляет собой набор графических примитивов (линий, точек, полилиний), организованных по слоям. Смысловая нагрузка объектов кодируется не в виде структурированных атрибутов, а контекстуально – через принадлежность к слою, цвет, тип линии или пояснительные надписи [5].

Практическая проверка стандартных средств экспорта QGIS показала, что при экспорте в формат DXF сохраняется только геометрия объектов, тогда как атрибутивные поля не передаются в структуру CAD-чертежа. Это существенно ограничивает возможность дальнейшей автоматизированной обработки данных в CAD-среде.

Встроенные средства ГИС. QGIS предоставляет стандартные средства экспорта в DXF. Преимущество – доступность и простота. Недосток – объекты помещаются на слой по умолчанию без учета тематической классификации, атрибутивная информация не сохраняется.

Специализированные плагины для QGIS. Плагин CADTools расширяет функциональность стандартных инструментов, но возможности по сохранению атрибутов остаются ограниченными.

Проприетарные решения. ArcGIS for AutoCAD позволяет подключаться к веб-сервисам ArcGIS из AutoCAD с сохранением атрибутов в расширенных данных. Недосток – привязка к экосистеме ESRI и отсутствие поддержки российских классификаторов [6].

ETL-инструменты. Платформа FME (Feature Manipulation Engine) позволяет создавать сложные сценарии трансформации. Недосток – высокая стоимость лицензий.

Скриптовые подходы. Использование Python с PyQGIS и библиотеками (GDAL, ezdxf, shapely) обеспечивает гибкость и контроль над преобразованием данных [7].

Таблица 1

Сравнительный анализ инструментов интеграции ГИС и САПР

Инструмент	Сохранение семантики	Гибкость маппинга	Валидация геометрии	Поддержка российских классификаторов	Доступность	Интеграция с QGIS
Встроенный экспорт QGIS	Нет	Низкая	Нет	Нет	Бесплатно	Полная
Плагины (CADTools и др.)	Частично	Средняя	Нет	Нет	Бесплатно	Полная
ArcGIS for AutoCAD	Да*	Средняя	Нет	Нет	Требует ArcGIS	Ограниченная
FME	Да	Высокая	Да	Частично	Коммерческий	Ограниченная
Python + PyQGIS + ezdxf	Да	Высокая	Да	Да	Бесплатно	Полная

Большинство существующих инструментов ориентированы на перенос геометрической информации и не обеспечивают сохранение атрибутивной структуры пространственных объектов. Коммерческие решения обладают необходимой функциональностью, однако их использование ограничено экономическими факторами и отсутствием готовых механизмов для работы с российской нормативно-справочной информацией.

Выбор QGIS в качестве базовой платформы обусловлен открытой архитектурой, широкой функциональностью и отсутствием лицензионных ограничений. Python является наиболее распространенным языком разработки расширений для QGIS.

Предлагаемый подход позволяет:

- выполнять преобразование атрибутов объектов в структуру CAD-слоев по настраиваемым правилам;
- реализовывать процедуры предэкспортной проверки и коррекции геометрии;
- автоматически выполнять преобразование систем координат (ГСК-2011, СК-42, МСК, WGS 84);
- учитывать требования отечественных классификаторов пространственных данных, включая КТИ.

Апробация предложенного подхода проведена на пространственных данных города Бузулука Оренбургской области, полученных через плагин QuickOSM. Общий массив данных составил около 13 000 зданий и около 2 000 объектов дорожной сети, а также объекты инженерных коммуникаций.

Разработанный прототип скрипта на базе Python и PyQGIS позволил выполнить автоматизированное распределение объектов по слоям CAD-чертежа на основе их атрибутивных характеристик. В результате экспорта было сформировано 12 тематических слоев DXF (вместо 1 слоя по умолчанию). Атрибутивные данные (тип покрытия дорог, этажность и назначение зданий, материал труб и др.) были сохранены в расширенных данных DXF (XDATA).

Полученные результаты подтверждают принципиальную возможность сохранения семантической информации при экспорте в CAD-форматы на массивах данных реального масштаба.

Проведенное исследование позволяет сформулировать следующие выводы:

1. Стандартные инструменты экспорта и большинство плагинов обеспечивают перенос геометрической информации, но не позволяют сохранять атрибутивную структуру пространственных объектов.
2. Проприетарные решения (ArcGIS for AutoCAD, FME) обладают необходимой функциональностью, однако их использование ограничено экономическими факторами и зависимостью от конкретных программных экосистем.
3. Ни одно из рассмотренных коммерческих решений не предлагает готовых механизмов для работы с российской нормативно-справочной информацией (КТИ, СП 47.13330.2016 и др.).

4. В условиях импортозамещения наиболее перспективным направлением является разработка специализированной методики семантического преобразования данных на базе открытых технологий (Python, PyQGIS, ezdxf).
5. Предложенный подход может служить основой для разработки плагина QGIS, предназначенного для автоматизации обмена данными между ГИС и САПР при проектировании инфраструктурных объектов.

Дальнейшие исследования будут направлены на программную реализацию данной методики, расширение библиотеки правил семантического маппинга для различных типов инфраструктурных объектов и апробацию на реальных проектных данных проектных организаций.

Литература

1. Бойков В. Н., Скворцов А. В. InfraBIM для автомобильных дорог // САПР и ГИС автомобильных дорог. 2019. № 1(12). С. 4-9.
2. Zhu J., Wu P. Towards Effective BIM/GIS Data Integration for Smart City by Integrating Computer Graphics Technique // Remote Sensing. 2021. Vol. 13, No. 10. Art. 1889.
3. СП 47.13330.2016. Инженерные изыскания для строительства. Основные положения. Москва: Минстрой России, 2016. 170 с.
4. QGIS Documentation: PyQGIS Developer Cookbook. URL: https://docs.qgis.org/3.34/en/docs/pyqgis_developer_cookbook/ (дата обращения: 15.03.2026).
5. Boria E.S., Badhrudeen M., Fonteix G., Derrible S., Siciliano M. A Protocol to Convert Infrastructure Data from Computer-Aided Design (CAD) to Geographic Information Systems (GIS) // arXiv preprint. 2020. arXiv:2012.08313.
6. ESRI. ArcGIS for AutoCAD: Documentation // Environmental Systems Research Institute. 2022. URL: <https://doc.arcgis.com/en/arcgis-for-autocad/> (дата обращения: 15.03.2026).
7. Документация библиотеки ezdxf. URL: <https://ezdxf.readthedocs.io> (дата обращения: 15.03.2026).
8. Хабарова И. А., Хабаров Д. А., Кондратьев М. А. Эффективность применения ГИС при постановке земельных участков на кадастровый учет // Землеустройство, кадастр и мониторинг земель. 2021. № 10. С. 45-51.
9. Осенняя А. В., Грибкова И. С., Хахук Б. А. Применение геоинформационных систем при проведении кадастровой оценки объектов недвижимости в Российской Федерации // Региональные геосистемы. 2020. Т. 44, № 1. С. 55-63.

COMPARATIVE ANALYSIS OF GIS AND CAD INTEGRATION METHODS FOR INFRASTRUCTURE DESIGN PROBLEMS

Davydov Alexander Denisovich

Student

МИРЭА – Российский Технологический Университет

Russia, 119454, Moscow, Vernadsky Avenue 78

E-mail: s.davydov2002@yandex.ru

Abstract. The article discusses the problem of the information gap between geoinformation systems (GIS) and computer-aided design systems (CAD) in the creation of project documentation for infrastructure facilities. A comparative analysis of existing approaches to integration is conducted: built-in GIS tools, specialized plugins, proprietary solutions, and ETL tools. The key disadvantages of these approaches are identified, with the main one being the loss of semantic information during data conversion. The article substantiates the need to develop a methodology for preserving the attribute integrity during export from QGIS to CAD formats, taking into account Russian regulatory documents.

Keywords: GIS, CAD, data integration, semantic mapping, QGIS, infrastructure design, import substitution.

SOFTWARE AND INFORMATION SOLUTIONS ENGINEERING WITH HIGH AND LOW REGULATION: A COMPARISON OF FINTECH AND THE HOSPITALITY INDUSTRY

© **Maxim Nikitashin**

graduate student, assistant

E-mail: max.nikitashin@gmail.com

© **Dušan Lesjak**

doctor, professor

University of Primorsky

Slovenia, 6000, Koper, Titova Street 10

Abstract. This article examines how different levels of regulation shape information solutions engineering in different sectors, using a PESTLE-based comparison of the USA, China, Russia and Slovenia. Two modes of regulation are identified. The first is high in vertical sectors such as civil engineering, agriculture, logistics, MedTech, and the military industry, which have similarities and where the research results can potentially be mutually transferable. The second mode is low, the most notable representative of which is the hospitality industry. The FinTech - hospitality industry IT comparison highlights a sharp contrast. While FinTech follows a compliance-first model, hospitality IT is more market-driven, with lighter sector-specific regulation. The study shows that regulatory intensity strongly affects management and implementation across different national environments and reveals similarities and differences between industries.

Keywords: software engineering, information solutions engineering, software and information solutions engineering, regulatory intensity, FinTech, hospitality industry, PESTLE-analysis, compliance-oriented development, market-driven development

1. Introduction

Recently, software engineering and information solutions engineering have been mentioned together often. The reason can be that a number of important and recognized recent studies show that they have many similarities in their historical development, research topics, design problems, and practical challenges today [1; 2]. These works also support the idea that software development and the broader development of information solutions should be studied together much closer as a software and information solutions engineering.

Numerous studies in the field of software and information solutions engineering and financial technology (FinTech) indicate the existence of economic and management differences between software and information solutions engineering in general and in the finance sector [3; 4; 5], which belongs to the service industries, as well as in many other sectors, including civil engineering IT [6; 7], agricultural IT [8], logistics IT, medical technology (MedTech) [10], military IT [11], and hospitality industry IT [12; 13].

Among the others, the latter is the most notable representative of low-regulated service industries based on the same pre-research [14]. Therefore, it was selected for analysis and comparison with the FinTech field.

The differences mentioned at the beginning cover such areas as management approaches, methods and methodologies, economic principles, and legal frameworks related to the development, implementation, and use of IT solutions in industries [16; 17; 18]. Highly regulated industries are inclined to use and maintain traditional management methodologies and approaches, such as Waterfall, and economic principles. Others, usually less regulated, tend to adapt and use more modern, open, adaptive and efficient management approaches, such as agile [19], and economic principles, such as the principles of open-source software [17] or platform economics [14].

Here regulatory intensity is an important factor shaping software and information solutions engineering across industries. Highly regulated sectors, such as FinTech and civil engineering IT, which were already mentioned and selected for the analysis, require compliance-oriented, documented, risk-aware, and often more formalized development. The reason for this is that regulation affects governance, cybersecurity, interoperability, and project management from the beginning [3; 5; 7]. In contrast, less regulated sectors, such as hospitality IT, are shaped more by customer expectations, operational efficiency, service quality, and market competition. This allows greater flexibility, faster experimentation, and more agile innovation. At the same time privacy, payment security and cybersecurity still remain relevant constraints [12; 13; 20; 21].

The comparison between FinTech and hospitality industry IT is current and important. It shows the contrast between a highly regulated digital sector and a more market-driven one, which was already mentioned and described. This makes the topic relevant for analysing, particularly in terms, how lower regulation allows greater flexibility, faster experimentation and more agile IT development, while still requiring attention to privacy, payment security and cybersecurity.

The object of this research is the comparison between highly regulated FinTech IT and less regulated hospitality industry IT, focusing on how different regulatory intensity affects software and information solutions engineering practices, flexibility, innovations, management, and market responsiveness.

The subject of this research is the impact of different regulatory intensity on software and information solutions engineering practices in FinTech and hospitality industry IT. Particularly in relation to compliance, agility, innovation, market orientation, user-centred development, and operational flexibility.

2. Methodology

For the purpose of the research, it was decided to conduct an analysis of individual vertical economic sectors using the PESTLE model. The analysis was planned as done for individual representative countries, with a subsequent comparison of the results across these countries and selected sectors.

PESTLE is a strategic analysis model used to examine the macro-environmental forces that shape organizational decisions, particularly Political, Economic, Social, Technological, Legal, and Environmental conditions [22; 23]. It is primarily used to identify opportunities and threats in the external environment so that managers can align strategy with broader contextual changes. At the same time, it is important that scholars also note that the model is often criticized for being broad, largely qualitative, and sometimes insufficiently precise unless combined with other analytical tools [22; 23].

For the purpose of the research, countries currently representing the most political-ly, economically and technologically developed powers in the world were selected: the United States of America (USA), China, Russia and European Union (EU) [24; 25; 26; 27; 29; 30]. In the latter, Slovenia was chosen as the representative country (Table 1). The main reasons for selecting an individual country for the purpose of analysis are the disunity of the EU and the high degree of independence of its mem-bers in the areas of politics and economics.

The countries were selected according to three main criteria. The first is global po-litical and economic significance. The second is technological development, The third is diversity of regulatory models. The USA and China were chosen because they rep-resent two of the most influential global economic and technological powers, but with different market and state-regulation traditions. Russia was included because it pro-vides an example of a large, strategically important, and more sovereignty-oriented regulatory environment. Also, it still remains a significant world power, especially in a political sense. The EU is a major regulatory and economic actor, while its member states still preserve considerable independence in political, economic, and implemen-tation practices. Together, selected countries allow comparison across market-driven, state-directed, sovereignty-focused, and EU-shaped approaches to software and in-formation solutions engineering regulation.

Table 1

Slovenia - EU Comparison

Indicator	Slovenia	EU	References
GDP per capita (PPS) (2020-2024 average)	89.2% of the EU av- erage (+2 p.p.)	100% (+1.1 p.p.)	[31]
Actual individual consumption per capita (2022-2024 average)	85% of the EU aver- age in average	100%	[31]
Employment rate (2020-2024 average)	72.1% (+1.2 p.p.)	69.4% (+1.9 p.p.)	[33]
Unemployment rate (2020-2024 average)	3.9% (-0.9 p.p.)	6.2% (-0.8 p.p.)	[33]
ICT specialists as % of employment (2020-2024 average)	4.18% (+0.4 p.p.)	4.64% (+0.7 p.p.)	[33]
Digital strategy measures	81 measures	-	[33; 34]
Digital strategy budget	EUR 685 million (1.02% of GDP)	-	[33; 34]

The objectives of the research were:

1. To analyze and determine the differences between individual sectors regarding the level of regulation.
2. If there are, how do these affect software and information solutions engineering for these sectors?
3. If there are, compare the vertical sectors with each other.
4. Based on the potential comparative analysis, to prepare findings and recommendations for experts, managers, and scientists in the field for their work and research.

The research questions were defined as:

1. Are there differences between selected sectors regarding the level of regulation?
2. If the differences are present, what are those differences between selected sectors regarding the level of regulation?
3. If the differences are present, how do they affect software and information solutions engineering in the areas selected for research?
4. If the differences are present, compare them among themselves across different vertical sectors.
5. If the differences are present, compare their influence among themselves across different vertical sectors.

In general, the research was conducted by defining research mentioned questions, selecting databases and keywords based on these questions and selected countries, filtering, screening titles, abstracts and full texts, and then extracting and analysing relevant data. Individual groups or clusters were formed by comparing recurring themes, methods, sectors, technologies, variables and theoretical perspectives. Then similar studies were grouped together to support structured analysis and comparison.

More specifically the literature review was conducted by selecting scientific and professional sources based on their relevance, popularity, and recognition in academic databases such as Google Scholar, Scopus, and Web of Science (WoS), as the largest and most recognized international and interdisciplinary ones. The recognition assessment was based on the number of citations. The sources were then grouped into sectoral clusters, which were FinTech and hospitality industry IT, to compare how different levels of regulation affect software and information solutions engineering across industries and countries.

3. Research

3.1. Theoretical Basis

The topic of high and low regulated industries comes from studies on FinTech regulation and compliance-first software and information solutions engineering and from hospitality IT research focused on digitalisation, smart hotels, cloud systems, AI, service innovation, and customer experience. Scientific sources show that FinTech as was mentioned, has been researched as a highly regulated IT field, while hospitality IT, again as was mentioned, is usually studied as a more flexible, market-driven sector. This makes their comparison useful for understanding how regulation changes

software and information solutions engineering methods, innovation speed, and management approaches.

The theoretical basis of this topic combines studies on FinTech regulation and compliance-oriented software and information solutions engineering with research on hospitality digitalisation, IT strategy, smart hotels, and service innovation. The most significant sources include Jinasena et al. [35], Munteanu and Dragos [5], and Letelay et al. [36] for FinTech, and Wynn and Jones [12], Wynn and Lam [13], Qiu et al. [20], Wang and Fu [37], and Yolcu et al. [38] for hospitality IT. Together, these sources show that FinTech is theoretically grounded in compliance, security, and risk control, while hospitality IT is grounded more in customer experience, agility, digital service quality, and market-driven innovation.

3.2.FinTech

FinTech refers to the use of digital technologies in financial services, including payments, banking infrastructure, consumer finance, digital assets, open banking, cybersecurity, and regulatory technology. In the article, FinTech can be presented as a highly regulated sector where software and information solutions engineering is shaped by financial stability, consumer protection, payment security, data governance, and cybersecurity requirements. Therefore, FinTech innovation usually follows a compliance-first logic, requiring strong governance, documentation, resilience, and risk-aware development from the earliest stages of IT engineering [3; 5; 36].

The breakdown of the entire analysis for FinTech is presented in Table 2.

Table 2

Breakdown of PESTLE Analysis for FinTech

PESTLE factor	USA	China	Russia	Slovenia (EU)
Political	Active but fragmented state involvement; federal oversight is increasing for large non-bank payment apps and personal financial data rights [39; 41].	Highly state-directed, with financial and cyberspace regulators steering digital finance toward stability, data security, and policy alignment [42; 43].	Strongly regulator-led and sovereignty-focused, with the Bank of Russia prioritizing domestic payment infrastructure, technological sovereignty, and security [46].	EU-shaped and institutionally coordinated through digital-transformation policy, the Digital Decade roadmap, and the Bank of Slovenia innovation contact point [48; 49].
Economic	Large and deep market, but investment has shifted toward profitable B2B infrastructure, payments, and compliance-oriented firms [51].	Very large transaction scale and consumer usage, with 86 percent mobile-payment penetration and 185.147 billion mobile-payment transactions in 2023.	Large domestic digital-payments base, but more closed and nationally controlled. Cashless payments reached 88 percent of retail turnover and Fast Payments System	Small but EU-connected market suited to infrastructure-led models, although growth is limited by digital-transformation and SME digital-maturity bottlenecks [31; 56].

			(SBP) processed 13.4 billion transactions in 2024 [55].
Social	Consumers increasingly expect faster, cheaper, more transparent, and portable digital financial services, while inclusion gaps remain [57; 58].	Very high acceptance of digital finance, but regional and demographic divides persist, especially among older and less digitally connected users [59; 60].	Users are familiar with domestic payment technologies such as QR payments, e-wallets, biometrics, Mir, and SBP, while fraud and security risks remain important concerns [55]. Digital ambition is strong, but limited digital skills, SME readiness, trust, and security concerns constrain adoption [48; 63].
Technological	Innovation-rich stack, with AI, cybersecurity, third-party dependencies, and layered controls becoming central engineering issues [64; 65].	Advanced payment architecture combined with e-CNY development and strict data-flow obligations [66].	Built around national rails and sovereign infrastructure, including SBP, Mir, the digital ruble, and DFAs, but constrained by cyberattacks and sanctions-related technology limits [70]. Technologically promising but ecosystem-sized, with payment rails, open-banking pathways, and possible digital-euro integration [48; 56].
Legal	Complex and multi-agency legal environment, especially for consumer protection, data rights, crypto-related risks, and digital payments [41].	Tight licensing and stringent data-security regime, including network-data and cross-border data-flow rules.	Highly restrictive and security-heavy, with DFA regulation and stricter personal-data localization, consent, and cross-border-transfer rules [70; 73; 74]. EU rulebook is decisive; ZInV-1, MiCA implementation, and DAC8/CARF reporting increase compliance burdens and create RegTech opportunities [75; 76].
Environmental	Indirect but growing relevance. Green-finance innovation is expanding, while AI/data-centre electricity demand creates sustainability pressure [77; 78].	Environmental policy is increasingly finance-linked, with studies connecting FinTech to greener capital allocation, carbon-efficiency, and lower emissions [79; 80].	Weaker but relevant driver: digital finance can support environmental financing and green investment, although sanctions and infrastructure limits constrain broader impact [81]. Green transition creates indirect FinTech upside through sustainability-linked sovereign issuance and central-bank climate investing. Sustainability knowledge remains general rather than specific [17; 82; 83].

3.3.Hospitality Industry IT

Hospitality industry IT refers to the use of digital technologies in hotels, restaurants, tourism services, and accommodation management. This includes booking platforms, property management systems, mobile check-in, contactless services, AI-based personalization, smart rooms, payment integration, and energy management tools. Hospitality IT can be presented as a comparatively low-regulated and market-driven sector, where software and information solutions engineering is shaped mainly by customer expectations, service quality, operational efficiency, labour shortages, personalization, and competitive pressure. Therefore, hospitality IT allows greater flexibility, faster experimentation, and more agile development than highly regulated sectors, while still requiring attention to privacy, cybersecurity, payment security, accessibility, and sustainability [12; 13; 20; 37; 38].

The breakdown of the analysis for hospitality industry IT is presented in Table 3.

Table 3

Breakdown of PESTLE Analysis for Hospitality Industry IT

PESTLE factor	USA	China	Russia	Slovenia (EU)
Political	Governance is more market-led than strategy-led, but FTC enforcement, ADA accessibility rules, PCI DSS, NIST guidance, and state privacy rules create practical IT governance requirements [84; 85; 86; 87; 88; 89].	Strongly state-led digitalisation, with smart tourism policy promoting upgraded digital infrastructure, 5G, AI-enabled services, smarter platforms, and data-based monitoring [90].	Tourism is governed through the national project “Tourism and Hospitality” to 2030 and supported by state digital-platform development such as Russia.travel/RussPass [92].	Clear public strategy. Slovenian tourism and digital-transformation strategies place digitalisation, sustainability, and data-driven destination management at the centre of tourism policy [94].
Economic	Large and mature market where labor shortages, operating costs, and guest-demand changes strengthen the case for automation and self-service IT [97; 98; 99].	Massive demand scale, supported by billions of domestic trips and a very large hotel base. It is creating a large market for booking, payment, automation, and energy-management systems [101].	Domestic tourism and investment support growth, while labor shortages and cost pressure increase demand for AI and productivity-enhancing systems [102; 103; 104].	Growing tourism and strong dependence on digital channels support IT adoption, but SME resource and capability constraints slow advanced technology uptake [105; 106; 107].

Social	Guests increasingly expect mobile check-in, digital keys, contactless service, personalization, and human-centered digital support [20; 108; 109]. Consumers are receptive to mobile-first and smart-hotel experiences, but expect reliability, personalization, and a balance between automation and human service [37;111]. Adoption is mixed. Digital services improve convenience. However, staff readiness, digital literacy, motivation, and preservation of the human factor remain important [112; 113]. Adoption is strongly manager dependent. AI uptake is shaped by attitudes, ownership structure, perceived opportunity, cost concerns, skills, and fear of losing human interaction [114; 115].
Technological	Cloud PMS/CRS, AI revenue management, unified guest-data platforms, smart rooms, and cybersecurity-by-design are key opportunities. However legacy fragmentation remains a barrier [38; 89; 116]. Rapid uptake of smart-hotel systems, super-app payments, AI service tools, IoT, robots, and mobile-payment integration [90; 117; 118; 119]. Core themes include online booking, mobile apps, smart rooms, chatbots, AI systems, contactless services, and nationally controlled payment rails such as SBP [112]. Strong basic digital foundations but uneven maturity; advanced analytics and AI adoption remain limited by cost, skills, and organizational readiness [106; 114; 115].
Legal	Privacy and compliance are fragmented but significant because hotels must manage payment data, personal data, accessibility, and cybersecurity obligations [84; 85; 86; 87; 88]. Highly structured data-governance regime under the Cybersecurity Law, Data Security Law, PIPL, and Network Data Security Management Regulations [122]. Strict data-protection and localization environment affects booking, guest-data storage, and cloud hospitality systems [125; 126]. EU law dominates. GDPR and the EU AI Act make privacy, cybersecurity, AI governance, and vendor governance strategic concerns .

Environmental

Sustainability is driven by utility costs, brand pressure, and eco-innovation; IoT HVAC optimization, energy analytics, and predictive maintenance are important IT opportunities [129; 130].

Digitalisation and sustainability increasingly converge through green hotel management, smart energy management, AI load forecasting, and building-monitoring systems [133].

Environmental opportunities focus on energy monitoring, smart building control, and digital support for green-hotel models, especially in regional tourism projects [134].

Strong green-digital linkage: national strategy promotes sustainable tourism, and ICT use can reduce energy/material consumption, but measurement remains weaker than implementation [135].

As part of the analysis of the political aspect, Table 4 presents an additional, approximate calculation of the number of regulations in selected industries for each of the selected countries. For the USA, the calculation was based on Federal Register data. For China, the calculation was based on data from the Database of National Laws and Regulations. For Russia, the approximate number of industry-related regulations was estimated through searches in official legal-act databases of the Ministry of Justice . Because Russia’s regulatory system includes not only laws and decrees but also registered executive acts, technical standards, and mandatory requirements, the results should be interpreted as approximate indicators of regulatory intensity rather than exact counts of regulations. For Slovenia, the calculation was based on data from the Legal Information System of the Republic of Slovenia (PisRS) .

Table 4

Number of Regulations in the Analysed Industries

	FinTech	Hospitality Industry
USA	Up to 300	Up to 8
China	At least 38	Up to 10
Russia	At least 27	Up to 8
Slovenia (EU)	At least 36	Up to 8

4. Discussion

The comparison between FinTech and hospitality industry IT shows a sharp contrast in regulatory intensity and its influence on IT-engineering.

FinTech is clearly a high-regulation sector, where digital products are developed under strong legal, political, and technical constraints. IT solutions must address regulatory compliance from the start, especially in relation to financial supervision, payment systems, anti-fraud controls, personal-data rights, and cybersecurity. Innovation is therefore possible, but it is usually shaped by a “compliance-first” logic.

Hospitality IT, in contrast, is a much less regulated and more market-driven environment. Although the sector still faces important legal obligations in areas such as

privacy, accessibility, payment security, and platform governance, it is generally influenced more by customer expectations, operational efficiency, labor shortages, personalization, and competitive pressure than by dense sector-specific regulation. This gives hospitality firms more freedom to experiment with cloud systems, contactless services, AI-based guest management, mobile tools, and self-service technologies. In comparison with FinTech, hospitality IT therefore has greater room for flexibility, faster iteration, and user-centred innovation.

This contrast has important managerial implications. While FinTech often benefits from structured governance, strong documentation and risk-controlled development cycles, hospitality IT is better positioned to adopt agile, adaptive and service-oriented approaches that respond quickly to changing guest needs and market trends. At the same time, the comparison also shows that lower regulation does not eliminate constraints; instead, it shifts the emphasis from formal compliance to usability, integration, cost efficiency and digital maturity. Therefore, the main difference is not whether regulation exists, but how strongly it determines the logic of IT engineering. FinTech is shaped by regulation as a core design condition. Hospitality IT is shaped primarily by market responsiveness, with regulation functioning as a secondary boundary rather than the main organizing principle.

Notable and important here is comparison of two analysed countries: Slovenia and Russia, which reveals several important similarities. These two countries were selected for analysis due to the best similarity among the analysed countries in terms of their history, cultural background, and national economy. In FinTech, both countries treat digital finance as an area requiring strong legal oversight, secure infrastructure, and careful control of personal and transactional data, which means that IT development is shaped by regulation to a much greater extent than in less regulated sectors. In hospitality IT, although regulation is lighter than in FinTech, Slovenia and Russia still share a common pattern: in both countries, digital transformation is supported by public strategies, while practical adoption depends heavily on organizational readiness, digital skills, and the ability of firms to integrate new tools into service delivery. This means that despite differences in scale and political context, both countries show that the success of IT engineering depends not only on regulation itself, but also on the capacity of organizations and users to operate within a digitally changing environment.

The general cross-country comparison also adds value. China and Russia generally show stronger state direction, the United States shows a more fragmented but still demanding environment, and Slovenia reflects EU-shaped regulation combined with the limits of a smaller market. At the same time, the study identifies a recurring weakness across sectors: sustainability is usually acknowledged, but often only in broad and not sufficiently operational terms. This suggests an important direction for future research - how sustainability requirements can be translated into concrete IT engineering practices alongside regulatory compliance and sectoral efficiency goals.

5. Conclusion

Concluding it can be surely stated that regulation operates in two main modes. The first is a high-regulation mode, typical of vertical sectors such as civil engineering, agriculture, logistics, MedTech, and the military industry. These sectors share certain

similarities, making research findings potentially transferable between them. The second is a low-regulation mode, with the hospitality industry serving as its most prominent example.

The comparison between FinTech and hospitality industry IT highlights a much stronger contrast. FinTech clearly represents a high-regulation environment, where software and information solutions engineering is shaped by formal legal oversight and secure infrastructure requirements. Hospitality IT, in contrast, operates in a more market-driven and lightly regulated context, where technology decisions are influenced more by guest expectations, operational efficiency, labour shortages, and service quality than by sector-specific regulatory systems. Although privacy, payment security and accessibility remain relevant in hospitality, they do not dominate IT development to the same extent as in FinTech.

This contrast leads to an important conclusion, that the lower regulatory intensity of hospitality-IT allows greater space for agility, rapid experimentation and user-centered innovation, while FinTech must integrate innovation with compliance from the start. Therefore, the results show that software and information solutions engineering cannot be understood independently of its sectoral environment. Regulation not only changes technical requirements but also influences management style, development speed, and the balance between control and flexibility. In this sense, FinTech and hospitality IT represent two opposite poles that clearly demonstrate how regulatory intensity shapes modern software and information solutions engineering practice.

It is also important to note, that the analysed vertical markets in different countries, such as Russia and Slovenia, mentioned and compared in the discussion, have similarities that indicate that the findings determined during the research are not coincidental. As already mentioned, all the countries selected for the analysis have different conditions. At the same time, the results between all of them are complimentary in many aspects. Therefore, it can be concluded that the findings of the research and the recommendations determined on its basis and presented below are certainly valid for the analyzed countries and sectors there and may be generally valid for other similar countries and sectors.

Based on the research findings, the following recommendations can be made:

1. Software and information solutions engineering practices should be aligned with the level of sectoral regulation.
2. Less regulated sectors such as hospitality IT should leverage their regulatory flexibility to adopt agile, customer-oriented, and innovation-driven approaches. At the same time, they should still ensure basic standards for privacy, cybersecurity, and payment security.
3. Highly regulated sectors like FinTech should prioritize compliance by design, strong governance, documentation, risk control, and structured or hybrid development methodologies.
4. Across all sectors and countries, digital skills, organizational readiness, interoperability, and standardization are critical success factors, so it is essential to develop them.

5. Sustainability requirements should be translated into concrete and measurable software and information solutions engineering practices rather than remain general policy goals.

Overall, this research is important because it demonstrates that differences in regulatory intensity fundamentally shape sector-specific models of software and information solutions engineering. Future research should therefore focus on expanding the comparison to additional industries to refine measurement of regulation across contexts. Also, they should focus on examining how sustainability and management methodologies and approaches can be operationalized more precisely within both highly and lightly regulated digital environments.

Reference

1. Fitzgerald B. Information Systems and Software Engineering: The Case for Convergence. *arXiv*. 2024.
2. Rosenkranz C., Stray V., Wiesche M. Navigating the New Frontier of Information Systems Engineering. *Business & Information Systems Engineering*. 2025; 67: 1-5.
3. Jinasena D. N., Spanaki K., Papadopoulos T., Balta M. E. Success and Failure Retrospectives of FinTech Projects: A Case Study Approach. *Information Systems Frontiers*. 2020; 25: 259-274.
4. Letelay K., Mola S. A. S., Go R. Y. Challenges of Agile Software Development in the Banking Sector: A Systematic Literature Review. *JOIV International Journal on Informatics Visualization*. 2025; 9(1): 38.
5. Munteanu V. P., Dragos P. The Case for Agile Methodologies Against Traditional Ones in Financial Software Projects. *European Journal of Business Management and Research*. 2021; 6(1): 134-141.
6. Mahalingam A., Kashyap R., Mahajan C. An Evaluation of the Applicability of 4D CAD on Construction Projects. *Automation in Construction*. 2010; 19(2): 148-159.
7. Pan Y., Zhang L. Roles of artificial intelligence in construction engineering and management: a critical review and future trends. *Automation in Construction*. 2021; 122. Article 103517.
8. Kaloxylas A., Eigenmann R., Teye F., Politopoulou Z., Wolfert S., Shrank C., Dillinger M., et al. Farm Management Systems and the Future Internet Era. *Computers and Electronics in Agriculture*. 2012; 89: 130-144.
9. Marešová P., Klímová B., Honegr J., Kuča K., Ibrahim W. N. H., Selamat A. Medical Device Development Process, and Associated Risks and Legislative Aspects-Systematic Review. *Frontiers in Public Health*. 2020; 8. Article 308.
10. Berg H., Ritschel J. D. The Characteristics of Successful Military IT Projects: A Cross-Country Empirical Study. *International Journal of Information Systems and Project Management*. 2023; 11(2): 25-44.
11. Wynn M., Jones P. IT Strategy in the Hotel Industry in the Digital Era. *Sustainability*. 2022; 14(17): 10705.
12. Wynn M., Lam C. Digitalisation and IT Strategy in the Hospitality Industry. *Systems*. 2023; 11(10): 501.
13. Nikitashin M., Lesjak D. Management of the Modern FinTech Software Engineering: Example of Slovenia. Unpublished manuscript, dissertation. 2026.
14. Nikitashin M. Artificial Intelligence and Management of Dualities in Software Development Companies. *Uporabna informatika*. 2025; 33(1): 19-28.
15. Nikitashin M. Economics of Software Engineering in Slovenia. *Revija za ekonomske in poslovne vede*. 2025a; 12(1): 18-38.
16. Nikitashin M. AI and UI Development: A Case Study of Mobile Application X. *Uporabna informatika*. 2024; 32(4): 180-186.

17. Nikitashin M., Nikitashin A., Werber B. Artificial Intelligence and Software Development in Slovenia: Adoption, Challenges, and Opportunities. *Universum: tehničke nauke*. 2026; 5(146).
18. Qiu R. T. R., Park J., Hao F., Chon K. Hotel Services in the Digital Age: Heterogeneity in Guests' Contactless Technology Acceptance. *Journal of Hospitality Marketing & Management*. 2024; 33(1): 33-56.
19. Yolcu S., Şahin A., Dirsehan T. Gaining Ground: How Technology Fuels Hotel Competitiveness - A Systematic Review of the Literature. *Tourism Planning & Development*. 2025; 1(31).
20. Yüksel İ. Developing a Multi-Criteria Decision Making Model for PESTEL Analysis. *International Journal of Business and Management*. 2012; 7(24): 52-66.
21. Sammut-Bonnici T., Galea D. PEST Analysis. In *Wiley Encyclopedia of Management*. 2015.
22. Josef Korbel School of Global and Public Affairs, Pardee Institute. n.d. "National Power." Accessed April 15. 2026. URL: <https://korbel.du.edu/pardee/national-power/> (accessed: April 15, 2026).
23. National Center for Science and Engineering Statistics. n.d. "Global R&D and International Comparisons." Science and Engineering Indicators. Accessed April 15. 2026. URL: <https://nces.gov/pubs/nsb20257/global-r-d-and-international-comparisons-2> (accessed: April 15, 2026).
24. Nature Index. 2025 Research Leaders: Leading Countries/Territories. *Nature Index*. 2025. URL: <https://www.nature.com/nature-index/research-leaders/2025/country/all/global> (accessed: April 15, 2026).
25. Organisation for Economic Co-operation and Development. 2026. URL: <https://www.oecd.org/>.
26. UNdata. Human Development Trends. *Source: Human Development Indices: A Statistical Update*. 2025 (United Nations Development Programme). 2025. URL: <https://data.un.org/DocumentData.aspx?id=504> (accessed: April 15, 2026).
27. World Bank. GDP (Current US\$). *World Bank Open Data*. 2026. URL: <https://data.worldbank.org/indicator/NY.GDP.MKTP.CD> (accessed: April 15, 2026).
28. Statistical Office of the Republic of Slovenia. Digital Entrepreneurship, Detailed Data. 2025. URL: <https://www.gov.si/en/state-authorities/government-offices/statistical-office/>.
29. European Commission. European Commission. 2024. URL: https://commission.europa.eu/index_en.
30. European Commission. n.d. "Slovenia. Digital Decade Country Report." Shaping Europe's Digital Future. 2025. URL: <https://digital-strategy.ec.europa.eu/en/factpages/slovenia-2025-digital-decade-country-report> (accessed: April 15, 2026).
31. The most significant sources include Jinasena et al. 2020.
32. Letelay K., Mola S. A. S., Go R. Y. Challenges of Agile Software Development in the Banking Sector: A Systematic Literature Review. *JOIV International Journal on Informatics Visualization*. 2025; 9(1): 38.
33. Wang J., Fu X. Unveiling the Human-Robot Encounter: Guests' Perspectives on Smart Hotel Experience. *Journal of Hospitality and Tourism Technology*. 2024.
34. Yolcu S., Şahin A., Dirsehan T. Gaining Ground: How Technology Fuels Hotel Competitiveness - A Systematic Review of the Literature. *Tourism Planning & Development*. 2025; 1(31).
35. Consumer Financial Protection Bureau. 2024. URL: <https://www.consumerfinance.gov/about-us/the-bureau/>.
36. Financial Stability Oversight Council. FSOC 2024 Annual Report. *Washington, DC: U.S. Department of the Treasury*. 2024. URL: <https://home.treasury.gov/system/files/261/FSOC2024AnnualReport.pdf>.
37. National Financial Regulatory Administration. n.d. "NFRA." Accessed April 21. 2026. URL: <https://www.nfra.gov.cn/en/view/pages/index/index.html> (accessed: April 21, 2026).
38. China Securities Regulatory Commission. n.d. "Overview." Accessed April 21. 2026. URL: http://www.csrc.gov.cn/csrc_en/c102023/common_zcncr.shtml?channelid=e9958c689bef4d468d81dc93c8d3479f (accessed: April 21, 2026).

39. Bank of Russia. Financial Technology Development. 2024a. URL: <https://cbr.ru/eng/fintech/> (accessed: April 21, 2026).
40. European Commission. Slovenia 2025 Digital Decade Country Report. 2025a. URL: <https://ec.europa.eu/>.
41. Banka Slovenije. Banka Slovenije Fintech Innovation Contact Point. 2026a. URL: <https://www.bsi.si/en/banka-slovenije-fintech-innovation-contact-point>.
42. Silicon Valley Bank. The Future of Fintech Report. 2025. URL: <https://www.svb.com/trends-insights/reports/fintech-industry-report/>.
43. Bank of Russia. National Payment System. 2026a. URL: <https://cbr.ru/eng/psystem/> (accessed: April 21, 2026).
44. Banka Slovenije. Annual Report. 2025. 2026b. URL: <https://www.bsi.si/en/publications/>.
45. Federal Deposit Insurance Corporation. 2023 FDIC National Survey of Unbanked and Underbanked Households. 2024. URL: <https://www.fdic.gov/household-survey>.
46. Federal Reserve Financial Services. Consumer Payments Study. 2024. URL: <https://fedpaymentsimprovement.org/wp-content/uploads/2024-consumer-payments-study.pdf>.
47. Mittal P., Singh R. I. A Systematic Literature Review on Fintech and Financial Inclusion in China. In *Resurgence and Sustainable Development of Asian Markets in the New Normal*. Singapore: Springer, 2025; 119(33). URL: https://link.springer.com/chapter/10.1007/978-981-96-1785-2_7.
48. Fund I. M. Digital Financial Inclusion and Income Inequality in China. *IMF Working Papers*. 2025; (71). URL: <https://www.elibrary.imf.org/view/journals/001/2025/071/article-A001-en.xml>.
49. Jain V., Jain N. From Cash to Clicks: A Systematic Review of Digital Payment Adoption Using the ADO Framework. *NMIMS Management Review*. 2025; 32(4): 277-91.
50. National Institute of Standards and Technology. AI Risk Management Framework. 2026. URL: <https://www.nist.gov/itl/ai-risk-management-framework>.
51. Federal Financial Institutions Examination Council. Annual Report 2021. URL: <https://www.ffiec.gov/sites/default/files/data/publications/annrpt21.pdf>.
52. Chin G. T. China's 'Digital Renminbi' (e-CNY) as Financial Inclusion: The Global Frontier of Central Bank Digital Currency. *China Economic Journal*. 2025. URL: <https://link.springer.com/content/pdf/10.1007/s43508-025-00112-0.pdf>.
53. Bank of Russia. Digital Financial Assets and Their Operators. 2025d. URL: https://cbr.ru/eng/finm_infrastructure/digital_oper/ (accessed: April 21, 2026).
54. Russian Federation. Federal Law No // 266-FZ of July 14, 2022, Amending the Personal Data Law. 2022. URL: <http://publication.pravo.gov.ru/file/pdf?eoNumber=0001202207140080>.
55. Russian Federation. Federal Law No // 23-FZ of February 28, 2025, Amending the Personal Data Law. 2025. URL: <http://publication.pravo.gov.ru/document/0001202502280034>.
56. Government of Slovenia. 156th Regular Session of the Government of the Republic of Slovenia. 2025a. URL: <https://www.gov.si/en/>.
57. Financial Administration of the Republic of Slovenia. Automatic Exchange of Information Reported by Crypto-Asset Service Providers (DAC8 and CARF). 2025. URL: <https://www.fu.gov.si/en/>.
58. U.S. Department of Energy. 2024. URL: <https://www.energy.gov/>.
59. International Energy Agency. Energy Demand from AI. // In Energy and AI. Paris: International Energy Agency. 2025. URL: <https://www.iea.org/reports/energy-and-ai/energy-demand-from-ai>.
60. Zhang X., Jamaluddin M. R. An Investigation of Automation and Human Interaction of Upscale Hotels in China. In *European Proceedings of Social and Behavioural Sciences*. 2024.
61. Qiao C., Cai W., Chen S. Green Fintech Contributes to Environmental Sustainability-Based on Empirical Evidence from China. *Humanities and Social Sciences Communications* 12. 2025. URL: <https://www.nature.com/articles/s41599-025-06159-y>.

62. Akberdina V., Lavrikova Y., Vlasov M. Environmental Financing: Does Digital Economy Matter? *Frontiers in Environmental Science* 11. 2024. URL: <https://www.frontiersin.org/journals/environmental-science/articles/10.3389/fenvs.2023.1268286/full>.
63. Banka Slovenije. Climate-Related Disclosure of Banka Slovenije's Own Financial Assets, May 2025. 2025. URL: https://www.bsi.si/storage/uploads/b6c4a826-c751-4c62-856e-e12ddd52f93c/Climate-related-disclosure-of-Banka-Slovenije's-own-financial-assets_2025.pdf.
64. Ministry of Finance. The Republic of Slovenia: EUR 1.0bn 3.125 Percent 10-Year Sustainability-Linked Notes. 2025. URL: <https://www.gov.si/en/state-authorities/ministries/ministry-of-finance/>.
65. Federal Trade Commission. FTC Takes Action Against Marriott and Starwood. 2024a. URL: <https://www.ftc.gov/>.
66. Federal Trade Commission. FTC Finalizes Order with Marriott and Starwood. 2024b. URL: <https://www.ftc.gov/>.
67. U.S. Department of Justice. 2024. URL: <https://www.justice.gov/>.
68. PCI SSC. PCI DSS. 2024. URL: <https://www.pcisecuritystandards.org/>.
69. California Privacy Protection Agency. Law & Regulations // and "California Finalizes Regulations to Strengthen Consumers' Privacy." 2025. URL: <https://privacy.ca.gov/2025/09/california-finalizes-regulations-to-strengthen-consumers-privacy/>.
70. NIST. NIST Offers Cybersecurity Guide Tailored to the Hospitality Industry. 2021. URL: <https://www.nist.gov/>.
71. Ministry of Culture and Tourism. 2024. URL: <https://www.mct.gov.cn/>.
72. Ministry of Economic Development of the Russian Federation. n.d. "National Project 'Tourism and Hospitality.'" Accessed April. 2026. URL: https://en.economy.gov.ru/material/activities/national_project_tourism_and_hospitality/ (accessed: April 2026).
73. Government of the Republic of Slovenia. Slovenian Tourism Strategy 2022-2028 // May 2022. 2022. URL: <https://www.gov.si/assets/ministrstva/MGTS/Dokumenti/DTUR/SLOVENIAN-TOURISM-STRATEGY-2022-2028.pdf>.
74. Bureau of Labor Statistics. Accommodation and Food Services: NAICS 72. 2026a. URL: <https://www.bls.gov/>.
75. Bureau of Labor Statistics. Job Openings and Labor Turnover - February 2026. 2026b. URL: <https://www.bls.gov/>.
76. AHLA. 2025 State of the Industry Report. 2025. URL: <https://www.ahla.com/resource/2025-state-industry-report>.
77. Association C. H. 2025. URL: <https://www.chinahotel.org.cn/>.
78. Government of the Russian Federation. Tourism. 2025. URL: <https://government.ru/en/activities/tourism/> (accessed: May 5, 2026).
79. Ministry of Economic Development of the Russian Federation. Tourism. 2025. URL: <https://en.economy.gov.ru/material/activities/tourism/> (accessed: May 5, 2026).
80. Faizullin R. V., Ototsky P. L., Gorlacheva E. N., Pospelova E. A., Kharitonova E. S. Assessing the Impact of Artificial Intelligence on Russian Labor Market Development Scenarios: Industry Analysis. *Economic and Social Changes: Facts, Trends, Forecast*. 2025; 18(1): 170-89. URL: <http://www.vscs.ac.ru/files/journal/issues/esc-2025-1-97-570a853452--en.pdf>.
81. Jagodič G., Jagodič D., Gorenak M. The Impact of Digitalisation on the Business Operations of Tourism Companies. *Mednarodno inovativno poslovanje = Journal of Innovative Business and Management*. 2025; 17(1).
82. OECD. Measuring the Digital Maturity of Tourism Businesses in Slovenia // OECD Tourism Papers 2025/08. Paris: OECD Publishing. 2025a. URL: https://www.oecd.org/en/publications/measuring-the-digital-maturity-of-tourism-businesses-in-slovenia_b70ddb07-en.html.

83. García-López A. M., Galindo-Pérez-de-Azpíllaga L., Foronda-Robles C. The Flow of Digital Transition: The Challenges of Technological Solutions for Hotels. *Social Indicators Research*. 2025; 178: 1323-46.
84. Dianawati N. et al. The Impact of Smart Hotel Technology on Guest Satisfaction and Loyalty. *African Journal of Hospitality, Tourism and Leisure*. 2024.
85. Wang T. et al. Can AI Improve Hotel Service Performance? *Tourism and Hospitality Research*. 2025.
86. Zhang X., Jamaluddin M. R. An Investigation of Automation and Human Interaction of Upscale Hotels in China. In *European Proceedings of Social and Behavioural Sciences*. 2024.
87. Kamenskikh E. A. Digital Transformation of the Russian Hospitality Industry and Its Impact on Consumer Experience. 2025a. URL: https://s-lib.com/en/issues/eiu_2025_12_v7_a4/.
88. Kamenskikh E. A. Artificial Intelligence in Russian Hotels: Balancing Efficiency and the Human Factor. 2025b. URL: https://s-lib.com/en/issues/eiu_2025_11_v14_a28/.
89. Planinc S., Kukanja M. Insights into Slovenian Hospitality SME Managers' Attitudes toward AI. *Academica Turistica - Tourism and Innovation Journal*. 2025; 18(1): 57-72.
90. Kukanja M., Planinc T. Adoption of Artificial Intelligence in Micro and Small Hospitality Enterprises: The Role of Organisational Characteristics and Managers' Attitudes Toward AI in Relation to Operating Revenues. *Tourism and Hospitality*. 2025; 6(5): Article.
91. Mavitha K., Shekhar S. K. Mapping the Landscape of Digital Transformation in the Hotel Industry. *Transformation and Society*. 2025.
92. Lok C. K., Lee B. P., Lei U. M. Enhancing Guest Experiences Through Smart Hotel Systems in the Hotel Industry in China: A System Integration Approach. In *Proceedings of the 11th World Congress on Electrical Engineering and Computer Systems and Sciences (EECSS'25)*. 2025.
93. Hassan S. A. Z., Eassa A. M. SHMIS: An Integrated IoT Context Awareness Framework for Hotel Management to Enhance Guest Experience and Operational Efficiency. *Information Technology & Tourism*. 2025; 27: 579-612. URL: <https://link.springer.com/article/10.1007/s40558-025-00316-4>.
94. Creemers R. China's Emerging Data Protection Framework. *Journal of Cybersecurity*. 2022; 8(1): tyac011.
95. Official Internet Portal of Legal Information of the Russian Federation. Federal Law No // 23-FZ of February 28, 2025; 2025: On. URL: <http://publication.pravo.gov.ru/document/0001202502280034>.
96. Roskomnadzor. n.d. "Personal Data." Accessed April 29, 2026. URL: <https://75.rkn.gov.ru/directions/p7166/> (accessed: April 29 2026).
97. DOE. Hospitality - Better Buildings & Better Plants Initiative. 2025. URL: <https://betterbuildingssolutioncenter.energy.gov/sectors/hospitality>.
98. Abdou A. H., Shehata H. S. H. Digital Synergy for Hospitality Sustainability. *Frontiers in Sustainability*. 2026.
99. PolyU. Annual report. 2024. URL: <https://www.polyu.edu.hk/sn-/media/departmentsn/content/about-sn/sn-at-a-glance/annual-report/annual-report-2023.pdf>.
100. Ilina E., Latkin A., Miloradov K., Manteyfel E., Jarrouj G. Implementation of the Concept of Green Hotel in Regional Tourism and Recreational Infrastructure Development Projects. *Journal of Lifestyle and SDGs Review*. 2025; 5(1).
101. OECD. Strengthening the Evidence Base for a Sustainable Tourism Future in Slovenia: A Tailored Set of Sustainability Indicators // OECD Tourism Papers 2025/02. Paris: OECD Publishing. 2025b. URL: https://www.oecd.org/en/publications/strengthening-the-evidence-base-for-a-sustainable-tourism-future-in-slovenia_12e6bf3e-en.html.

ИНЖЕНЕРИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ РЕШЕНИЙ В УСЛОВИЯХ ВЫСОКОГО И НИЗКОГО РЕГУЛИРОВАНИЯ: СРАВНЕНИЕ ФИНТЕХ И ИНДУСТРИИ ГОСТИНИЧНОГО ДЕЛА

Никиташин Максим

аспирант, ассистент

E-mail: max.nikitashin@gmail.com

Душан Лесьяк

доктор, профессор

Приморский университет

Словения, 6000, г. Копер, ул. Титова 10

Аннотация. В данной статье рассматривается, как различные уровни регулирования влияют на разработку информационных решений в разных секторах, с использованием PESTLE-анализа США, Китая, России и Словении. Выделены два режима регулирования. Первый - высокий в таких вертикальных секторах, как гражданское строительство, сельское хозяйство, логистика, медицинские технологии и военная промышленность, которые имеют сходства, и результаты исследований в которых потенциально могут быть взаимно переносимы. Второй режим - низкий, наиболее ярким представителем которого является индустрия гостиничного дела. Сравнение финтех и IT индустрии гостиничного дела выявляет резкий контраст. В то время как финтех следует модели, ориентированной на соблюдение нормативных требований, IT индустрии гостиничного дела в большей степени ориентирован на рынок, с более мягким отраслевым регулированием. Исследование показывает, что интенсивность регулирования сильно влияет на управление и внедрение в различных национальных средах и выявляет сходства и различия между отраслями.

Ключевые слова: разработка программного обеспечения, разработка информационных решений, разработка программного обеспечения и информационных решений, интенсивность регулирования, финтех, индустрия гостиничного дела, PESTLE-анализ, ориентированная на соблюдение нормативных требований разработка, ориентированная на рынок разработка

РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ «МОНОЧАТ» ДЛЯ ИНТЕГРАЦИИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В СЕРВЕРНЫЕ СИСТЕМЫ

© **Базаржапова Туя Жамьяновна**

кандидат педагогических наук, доцент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: btuyazh@yandex.ru

© **Шойбонов Тимур Николаевич**

студент,

Бурятская государственная сельскохозяйственная академия имени В.Р. Филиппова

Россия, 670034, г. Улан-Удэ, ул. Пушкина, 8

E-mail: tim35.34@yandex.ru

Аннотация: В статье рассматривается проектирование и реализация клиент-серверного веб-приложения «МоноЧат» для диалогового взаимодействия с генеративными моделями ИИ. Описана архитектура на базе PHP/PDO/MySQL, механизмы квотирования по тарифным планам, безопасная авторизация с хеш-хранением паролей и cURL-интеграция с ротацией ключей доступа.

Ключевые слова: веб-приложение, искусственный интеллект, клиент-серверная архитектура, MySQL, PHP, cURL, API-ключи, квотирование.

Актуальность. Зарубежные платформы (ChatGPT, Claude, Gemini) накладывают географические и платёжные ограничения. Создание собственного решения обеспечивает централизованное управление токенами доступа, реализацию тарифных лимитов и агрегацию платежей для корпоративных и образовательных пользователей.

Объект и предмет исследования. Объект - клиент-серверные веб-приложения с ИИ-интеграцией. Предмет - методы и архитектурные решения для серверной логики на PHP, реляционного моделирования в MySQL, обеспечения информационной безопасности и реализации cURL-запросов к внешним ИИ API.

Цель исследования: проектирование, программная реализация и тестирование масштабируемого веб-приложения «МоноЧат», обеспечивающего диалоговый интерфейс с ИИ, разграничение прав доступа и динамическое квотирование запросов.

Задачи исследования:

1. Провести системный анализ и сформировать требования к ИИ-чат-интерфейсу.
2. Разработать реляционную модель БД для профилей пользователей, истории сообщений и токенов.
3. Реализовать серверные модули авторизации, тарифов и профилей.
4. Разработать модуль интеграции с ИИ-API через cURL с динамической ротацией ключей.

5. Выполнить функциональное тестирование в локальном и облачном окружении.

Методология. Применялись принципы системного анализа, реляционного моделирования и Agile-разработки. Практические методы: ООП на PHP, работа с СУБД MySQL через PDO, сетевая интеграция по протоколу HTTP (сURL, JSONAPI), функциональное и нагрузочное тестирование.

Разработанность темы. Вопросы веб-разработки с ИИ рассматриваются в работах Р. Никсона, М. Фаулера и Н. И. Вьюковой. Построение лёгкой серверной архитектуры PHP/MySQL без тяжёлых фреймворков для образовательных учреждений остаётся недостаточно изученным.

Результаты исследования. Спроектирована реляционная БД в СУБД MySQL. На рисунке 1 показана логическая схема, отражающая структуру таблиц и связи между сущностями системы.

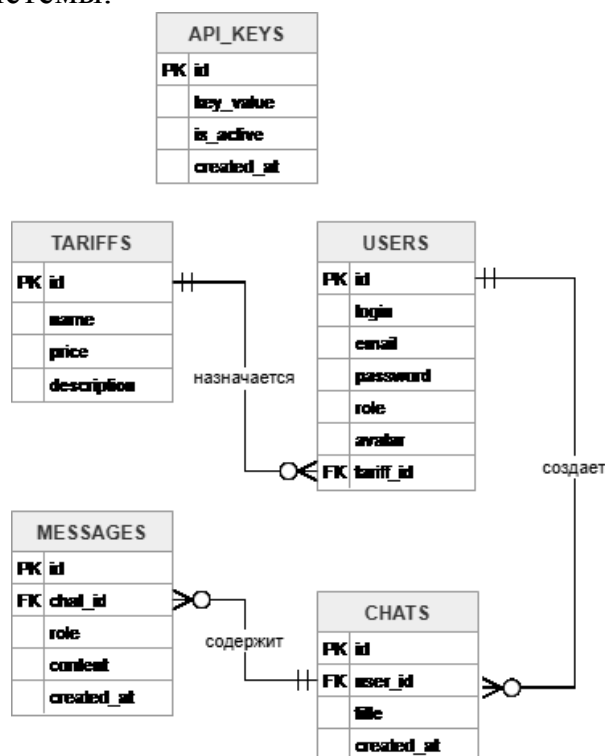


Рисунок 1 - Схема логической модели базы данных МоноЧат

Центральной является таблица users (идентификаторы, логины, e-mail, bcrypt-хеши паролей). Она связана с tariffs (тарифы FREE/PRO), chats (сессии) и messages (реплики). Таблица api_keys обеспечивает хранение и ротацию токенов ИИ-сервиса. Взаимодействие с БД реализовано через PDO с подготовленными выражениями, исключающими SQL-инъекции.

Ключевым элементом архитектуры является серверный модуль интеграции с ИИ-провайдером. Когда поступает запрос от пользователя модуль выбирает один из активных ключей из таблицы api_keys случайным образом. Затем формирует запрос к внешнему сервису через сURL.

Когда система отправляет запрос к ИИ-сервису, то она передает историю диалога в виде структурированных данных (формат JSON) и использует специальный ключ доступа (Bearer-токен) для подтверждения прав. Полученный то-

кен расшифровывается и сохраняется в таблице messages, связанной с текущей сессией.

Если ключ перестает работать, то администратор деактивирует его в панели управления. Тогда система переключается на другой доступный токен из пула и пользователи продолжают работать без каких-либо перерывов. Благодаря такому механизму обеспечивается высокая отказоустойчивость приложения при работе с внешними ИИ-провайдерами.

Для удобства пользователей создан адаптивный веб-интерфейс. Он адаптивен и имеет минималистичный дизайн.

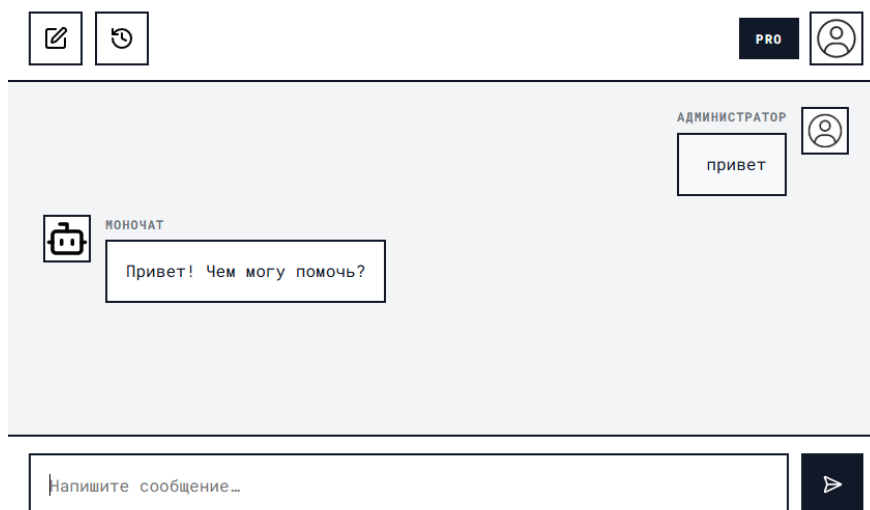


Рисунок 2 - Интерфейс диалогового окна системы МоноЧат

Главный экран (рисунок 2) включает левую боковую панель с историей переписок и кнопкой создания нового диалога, центральную ленту сообщений с разграничением по роли отправителя (user/assistant) и нижнее поле ввода запроса. Каждый диалог сохраняется в БД с указанием временных меток, что позволяет пользователю возвращаться к любой предыдущей переписке. Тарифный статус (FREE/PRO) отображается в шапке интерфейса и блокирует поле ввода при достижении суточного лимита.

Для администрирования пула ключей и контроля работоспособности системы разработан специализированный бэк-офис (рисунок 3).

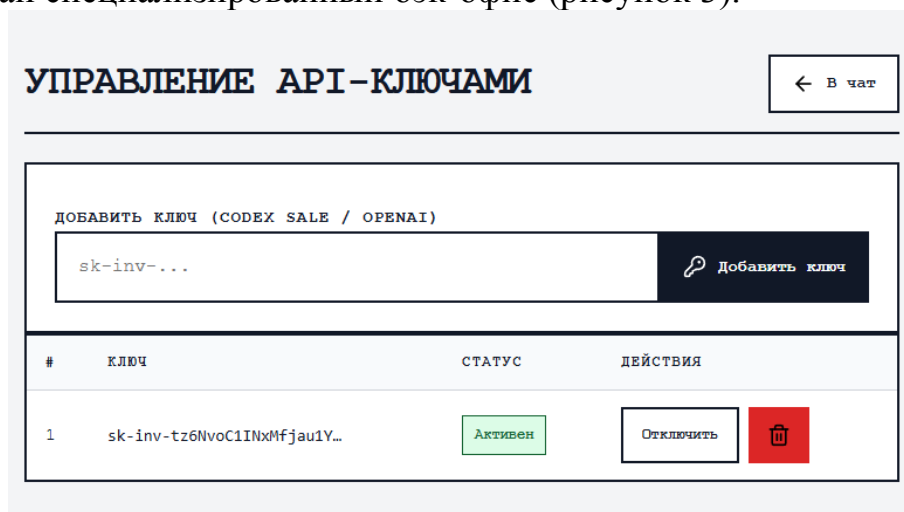


Рисунок 3 - Административная панель управления API-ключами

Административный интерфейс (рисунок 3) предоставляет уполномоченному сотруднику полный контроль над базой API-ключей: добавление новых токенов, временная деактивация существующих, просмотр текущего статуса и удаление неактуальных записей. Это позволяет гибко распределять нагрузку и оперативно реагировать на сбои внешнего провайдера. Доступ к данному разделу защищён проверкой роли пользователя (role = 'admin') на серверной стороне.

Попытка реализации модуля интеграции с ИИ-API через cURL с динамической ротацией ключей является первым опытом. Другими словами, разработанное веб-приложение является тестовым вариантом интеграции генеративного ИИ в серверные архитектуры. Функциональное тестирование проводилось в среде OpenServerPanel, а публичная версия запущена на виртуальном сервере Timeweb (<https://cm099009.tw1.ru/>). Тестирование приложения будет продолжено и работа над приложением будет продолжена.

Литература

1. Harrison J. et al. Continuous meta-learning without tasks. *Advances in Neural Information Processing Systems*. 2020; 21: 17571-17581.
2. Котеров Д. В., Симдянов И. В. PHP 8. Санкт-Петербург: БХВ-Петербург, 2024. 992 с.
3. Лэйн Х., Ховард К., Хэпке Х. Обработка естественного языка в действии. Санкт-Петербург: Питер, 2020. 512 с.
4. MySQL Database Service: official website. URL: <https://www.mysql.com> (access date: 01.05.2026).
5. PHP: Hypertext Preprocessor: official website. URL: <https://www.php.net> (access date: 03.05.2026).

DEVELOPMENT OF THE MONOCHAT WEB APPLICATION FOR THE INTEGRATION OF ARTIFICIAL INTELLIGENCE INTO SERVER SYSTEMS

Bazarzhapova Tuya Zhamyanovna

Candidate of Pedagogical Sciences, Associate Professor

Dorzhii Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: btuyazh@yandex.ru

Shoibonov Timur Nikolaevich

Student

Buryat State Agricultural Academy named after V.R. Filippov

Russia, 670034, Ulan-Ude, Pushkina str., 8

E-mail: tim35.34@yandex.ru

Abstract: The article discusses the design and implementation of the client-server web application "Monochat" for dialog interaction with generative AI models. The article describes a PHP/PDO/MySQL-based architecture, quota-tion mechanisms for tariff plans, secure authorization with hash-based password storage, and cURL integration with access key rotation.

Keywords: web application, artificial intelligence, client-server architecture, MySQL, PHP, cURL, API keys, and quota-tion.

АНАЛИЗ МЕТОДОВ ЗАЩИТЫ ПЕРСОНАЛЬНЫХ ДАННЫХ В РАСПРЕДЕЛЕННЫХ ИНФОРМАЦИОННЫХ СИСТЕМАХ

© **Дворникова Ксения Олеговна**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gastamanomy@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье проводится анализ методов защиты персональных данных в распределённых информационных системах (РИС). В работе изучаются особенности архитектуры РИС, при которых риски утечки персональных данных возрастают. Особое внимание уделяется технологиям сквозного шифрования, децентрализованной аутентификации и механизмам гомоморфного шифрования данных.

Ключевые слова: распределённые информационные системы (РИС), персональные данные, защита информации, сквозное шифрование, эллиптические кривые, децентрализованная аутентификация, Ф3-152, гомоморфное шифрование, информационная безопасность, РКІ.

Введение. Современный этап развития информационных технологий характеризуется переходом от монолитных структур к распределённым информационным системам, так как появилась необходимость обработки больших массивов данных и обеспечения гибкого масштабирования. Этот переход дал возможность одновременно обрабатывать данные и уменьшить вероятность полного отказа системы. РИС помогают распределить вычислительную нагрузку между большим количеством узлов и обеспечить эффективную работу пользователей, находящихся в разных точках планеты.

Одновременно с этим происходит децентрализация структуры и рост числа узлов увеличивают поверхность атаки, создавая новые риски для безопасности данных пользователей.

Актуальность. Актуальность темы данного исследования обусловлена поиском баланса между высокой производительностью распределённых систем и обеспечением гарантированного уровня безопасности персональных данных. В связи с этим возникает необходимость не просто перечислить существующие

механизмы защиты, но и провести их сравнительный анализ на предмет применимости в условиях высокой децентрализации и разнородности узлов.

Объект и предмет исследования. Объектом исследования являются распределённые информационные системы. Предметом исследования выступают методы защиты персональных данных в распределённых информационных системах.

Цель и задачи. Целью исследования является анализ и классификация современных методов защиты персональных данных в распределённых информационных системах. Задачи:

1. Выявить архитектурные уязвимости РИС.
2. Провести анализ криптографических методов защиты.
3. Сравнить модели аутентификации (PKI и децентрализованную).
4. Сформулировать рекомендации по выбору методов защиты.

Методология и методы исследования. В процессе подготовки работы были применены разные методы исследования. В качестве основного метода использовался системный анализ, что дало возможность изучить распределённую информационную систему как сеть взаимосвязанных узлов и оценить влияние безопасности каждого отдельного узла на защищённость всей системы. Для классификации и оценки эффективности различных технологий защиты применялся метод сравнительного анализа. Дополнительно был проведён анализ законодательства, чтобы учесть правовые требования к защите персональных данных.

Разработанность темы. Вопросы защиты персональных данных в распределённых системах рассматривались в работах Герасименко В.А., Малюка А.А., Терехова А.Н. и др. Однако комплексный анализ с учётом требований ФЗ-152 и современных криптографических стандартов представлен недостаточно.

Результаты исследования. В ходе проведенного анализа были рассмотрены разные способы защиты персональных данных в распределённых информационных системах.

Особое внимание было вызвано тем, что архитектура таких систем уязвима к большому количеству потенциальных угроз. Наиболее эффективным решением данной проблемы является использование не одного средства защиты, а комплексное применение различных технологий. Для этого обязательным условием является использование сквозного шифрования совместно с атрибутивной моделью контроля доступа, а также возможность применения гомоморфного шифрования. Механизмы гомоморфного шифрования позволяют осуществлять вычислительные операции над зашифрованными данными без предварительной расшифровки на стороне обработчика.

В процессе исследования также определены ограничения классических методов централизованной аутентификации в условиях распределённых систем с высокой сетевой латентностью и большим количеством узлов. В крупных распределённых информационных системах традиционные протоколы теряют надёжность. В качестве альтернативы определена перспективность внедрения технологий распределённого реестра и использования децентрализованных

идентификаторов. Эти инструменты исключают зависимость от единых центров сертификации и гарантируют неизменность журналов доступа.

Архитектурные уязвимости РИС в контексте ФЗ- 152. При переходе от централизованных к распределённым информационным системам появляются нестандартные уязвимости, устранение которых требует особого внимания к соблюдению Федерального Закона №152- ФЗ «О персональных данных» [5]. В распределённой системе данные пользователя редко хранятся в одном защищённом месте. Постоянное перемещение данных между узлами обработки часто приводит к конфликту между юрисдикциями разных стран, что вступает в противоречие с требованием локализации баз персональных данных граждан Российской Федерации на территории РФ. Процессы облачного резервного копирования могут автоматически разместить фрагменты данных на зарубежных серверах, даже если основной центр обработки данных находится в России.

Анализ криптографических методов: сравнение алгоритмов и длины ключа. В распределённых системах для защиты данных применяются разнообразные криптографические методы, эффективность которых зависит от выбора алгоритма и длины ключа. Использование симметричных алгоритмов, таких как Advanced Encryption Standard (AES), для защиты больших массивов информации обеспечивает высокую производительность при низких вычислительных затратах [4].

Применение ключей в сто двадцать восемь бит сегодня считается допустимым только для второстепенных задач, не затрагивающих конфиденциальную информацию.

Ассиметричная криптография обычно применяется для создания защищённых сеансов связи и работы электронных подписей при аутентификации узлов в распределённой сети. Сравнительный анализ подтверждает преимущество алгоритмов на эллиптических кривых над классическим методом RSA [1].

Длина ключа у алгоритмов на эллиптических кривых при одинаковом уровне стойкости с методом RSA намного меньше: двести пятьдесят шесть бит против трёх тысяч семидесяти двух бит. Это важно для устройств с ограниченными ресурсами.

Сравнение моделей аутентификации: от PKI к децентрализованным

Подтверждение подлинности пользователей и узлов в распределённой среде так же важно, как и шифрование каналов связи.

В PKI существует иерархия удостоверяющих центров. Деятельность центров заключается в выпуске и отзыве цифровых сертификатов. Суть цифрового сертификата состоит в подтверждении принадлежности открытого ключа к конкретному лицу или устройству. В корпоративных сетях с чёткой вертикальной структурой модель PKI эффективна и имеет высокую управляемость. Но при использовании PKI в масштабировании на тысячи узлов могут возникнуть недостатки, такие как: существование единой точки отказа в виде корневого центра, высокая нагрузка при проверке множества одновременно проверяемых сертификатов на службы каталогов, сложное управление жизненным циклом ключей для устройств, подключаемых и отключаемых динамически.

В качестве современной альтернативы в распределённых информационных системах всё чаще применяются децентрализованные модели. Они основаны на концепции самосуверенной идентичности и технологиях распределённого реестра [5]. В отличие от PKI децентрализованные идентификаторы позволяют узлам и пользователям самостоятельно генерировать пары криптографических ключей и подтверждать свои атрибуты без необходимости обращения к единому регистратору.

Основные отличия и ключевые характеристики обеих моделей аутентификации представлены в таблице 1.

Таблица 1

Сравнительный анализ методов аутентификации

Критерий сравнения	Классическая PKI	Децентрализованная (DID)
Точка отказа	Единая (центр сертификации)	Отсутствует (распределённый реестр)
Управление ключами	Централизованное (администратором)	Самосуверенное (пользователем)
Масштабируемость	Снижается при росте числа узлов	Высокая за счет децентрализации
Зависимость от связи	Требуется онлайн-доступ к CRL/OCSP	Возможна автономная проверка
Оптимальная среда	Закрытые корпоративные сети	Открытые и гибридные экосистемы

Проведённый анализ методов защиты персональных данных в распределённых информационных системах показал, что универсального решения для всех сценариев использования не существует. Вид обрабатываемых данных, уровень нагрузки на систему, степень доверия к узлам инфраструктуры и многие другие факторы влияют на выбор архитектуры безопасности.

Для корпоративных распределённых информационных систем, узлы которых находятся под контролем одной организации и требуется строгое соблюдение нормативных требований, оптимальной стратегией является применение централизованной инфраструктуры PKI и обязательного сквозного шифрования данных с использованием алгоритмов на основе ГОСТ или AES с длиной ключа двести пятьдесят шесть бит.

В ситуациях, когда нагрузки максимальны при участии большого количества пользователей с низким уровнем доверия друг к другу, возникает необходимость перехода к гибридным моделям. Проверка внешних сервисов происходит с помощью децентрализованных идентификаторов, а внутри системы всё также работают сертификаты PKI и строгий контроль доступа.

Литература

1. Бабаш А. В., Баранова Е. К. Криптографические методы защиты информации: учебник для вузов. 5-е изд. Москва: КноРус, 2022. 190 с.
2. Герасименко В. А., Малюк А. А. Защита информации в распределённых информационных системах. Москва: Инфра-М, 2021. 320 с.

3. ГОСТ Р 34.12–2015. Информационная технология. Криптографическая защита информации. Блочные шифры. Москва: Стандартинформ, 2015. 25 с.
4. Кузьминых Е. С., Маслова М. А. Анализ симметричных методов шифрования, проблемы и пути возможного их решения // Научный результат. Информационные технологии. 2023. № 1. С. 38-45.
5. Методический аппарат защиты от ошибок процессоров специализированных ЭВМ / А. А. Павлов, И. И. Корнеев, М. И. Макеев и др. // Вестник компьютерных и информационных технологий. 2023. № 3. С. 3-8.
6. Шаньгин В.Ф. Информационная безопасность и защита информации. Москва: ДМК Пресс, 2024. 702 с.

ANALYSIS OF PERSONAL DATA PROTECTION METHODS IN DISTRIBUTED INFORMATION SYSTEMS

Dvornikova Ksenia Olegovna

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gastamanomy@gmail.com

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article analyzes methods of personal data protection in distributed information systems (DIS). The paper studies the features of DIS architecture, in which the risks of personal data leakage increase. Special attention is paid to end-to-end encryption technologies, decentralized authentication, and data homomorphic encryption mechanisms.

Keywords: distributed information systems (DIS), personal data, information protection, end-to-end encryption, elliptic curves, decentralized authentication, Federal Law No. 152, homomorphic encryption, information security, PKI.

ПАРАЛЛЕЛЬНЫЕ АЛГОРИТМЫ И ИХ МАСШТАБИРОВАНИЕ В МНОГОПРОЦЕССОРНЫХ СИСТЕМАХ

© **Митькина Екатерина Андреевна**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: mitkina.katya06@gmail.com

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artyom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. Рассматриваются принципы построения параллельных алгоритмов для многопроцессорных систем. Анализируется закон Амдала, определяющий теоретический предел ускорения при увеличении числа вычислительных узлов. Описываются основные технологии параллельного программирования: OpenMP для систем с общей памятью и MPI для распределенных систем. Приводятся примеры распараллеливания сортировки и матричных операций. Обосновывается выбор архитектуры и методов распараллеливания в зависимости от структуры задачи и доступных вычислительных ресурсов.

Ключевые слова: параллельные алгоритмы; многопроцессорные системы; закон Амдала; OpenMP; MPI; распараллеливание сортировки; матричные операции.

Введение. Сегодня инженерное образование уже немыслимо без навыков работы с высокопроизводительными вычислениями. Информационно-коммуникационные технологии плотно вошли в учебный процесс технических вузов, и параллельные алгоритмы - одна из ключевых тем. В этой статье мы на конкретных примерах покажем, как устроены такие алгоритмы и что даёт их использование на многопроцессорных системах.

Актуальность. Большинство современных вычислительных машин являются многопроцессорными. Последовательные алгоритмы на таких системах работают неэффективно, что делает разработку параллельных алгоритмов ключевой задачей для повышения производительности вычислений.

Объект и предмет исследования. Объектом исследования являются параллельные алгоритмы. Предметом исследования — масштабирование параллельных алгоритмов в многопроцессорных системах.

Цель и задачи. Цель работы — анализ принципов построения параллельных алгоритмов и их масштабирования. Задачи:

1. Рассмотреть закон Амдала и его влияние на ускорение вычислений.
2. Описать основные технологии параллельного программирования (OpenMP, MPI).
3. Привести примеры распараллеливания (сортировка, матричные операции).
4. Обосновать выбор архитектуры в зависимости от задачи.

Методология и методы исследования. В работе использованы методы анализа научной литературы, моделирования параллельных алгоритмов и обобщения практического опыта.

Для оценки эффективности параллельных алгоритмов использовались теоретический анализ временной сложности и экспериментальное моделирование на вычислительных системах с разным количеством ядер (от 2 до 32). Сравнительный анализ проводился между последовательной, многопоточной (OpenMP) и многопроцессной (MPI) реализациями одних и тех же вычислительных задач: сортировки массива и умножения матриц. Ключевым критерием оценки выступало ускорение (speedup) — отношение времени выполнения последовательной программы к времени выполнения параллельной [1, 3].

Разработанность темы. Вопросы параллельного программирования рассмотрены в работах Антонова А.С., Гергеля В.П., Кормена Т.Х. и др. Однако практические рекомендации по выбору архитектуры и методов распараллеливания сохраняют актуальность.

Результаты исследования. Большинство современных вычислительных машин - многопроцессорные. Обычные последовательные алгоритмы на таких системах работают неэффективно: ядра простаивают, ресурсы тратятся впустую. Поэтому без переделки под параллельное исполнение сегодня уже не обойтись [1, с. 45].

Классическое ограничение - закон Амдала. Если в программе есть хоть какая-то последовательная часть, то с ростом числа процессоров ускорение быстро упирается в потолок. Формула простая: быстрее всего получается, когда последовательных участков почти нет [2, с. 234].

Практическое значение закона Амдала заключается в том, что даже небольшая последовательная часть программы (например, 5-10%) ограничивает максимальное ускорение, которое можно получить, увеличивая число процессоров. Так, при последовательной части в 5% максимальное ускорение не превысит 20, даже если использовать тысячи ядер. Это обосновывает необходимость тщательного анализа кода и выявления узких мест перед распараллеливанием [2, 3].

На практике параллельные программы пишут с использованием двух основных технологий. OpenMP подходит для одного узла с общей памятью: ставишь директивы перед циклами - и компилятор сам раздаёт потоки. MPI - для кластеров, где у каждого процесса своя память. Тут уже всё вручную: отправка и приём сообщений, сборка данных. Зато MPI легко масштабировать на тысячи узлов [5, с. 78; 8, с. 112].

Возьмём сортировку. На общей памяти (OpenMP) массив режется на куски, каждый поток быстро сортирует свою часть, потом всё сливается. В MPI данные разбрасываются по разным узлам, каждый сортирует локально, а в конце делается глобальное слияние [6, с. 89].

Матричные операции - тоже удобная задача для распараллеливания. Умножение матриц обычно делают либо блоками, либо ленточным способом. Из известных схем - алгоритмы Кэннона и SUMMA, они дают почти линейное ускорение, если всё правильно настроить [4, с. 156].

Эффективность распараллеливания зависит не только от выбранного алгоритма, но и от объёма передаваемых данных между процессами. В MPI каждое сообщение имеет накладные расходы на упаковку, отправку и распаковку, поэтому для мелкозернистых задач (где на каждый процесс приходится мало вычислений) выигрыш от распараллеливания может оказаться ниже ожидаемого. Для снижения накладных расходов рекомендуется агрегировать данные и сокращать количество обменов [5, 8].

Но просто разбить задачу - мало. Надо следить за балансировкой загрузки. На общей памяти можно динамически выдавать задачи потокам: кто освободился - берёт следующий кусок. На кластере это сложнее: если один узел получил больше данных, он всё равно будет работать дольше, а остальные будут ждать. Здесь требуются уже адаптивные методы [3, с. 278]. Сейчас многие переходят на гибридные схемы: внутри узла - OpenMP, между узлами - MPI, так и баланс лучше, и масштабирование выше [7, с. 45].

Одним из эффективных подходов к балансировке нагрузки в распределённых системах является использование динамического распределения задач (dynamic load balancing). В этом случае задачи не закрепляются жёстко за процессами, а распределяются центральным координатором по мере освобождения узлов. Такой подход позволяет компенсировать неравномерность вычислительной сложности данных и разницу в производительности узлов кластера [1, 7].

При работе с большими объёмами данных (Big Data) параллельные алгоритмы становятся не просто желательным, а обязательным требованием. Классические примеры — обработка логов веб-серверов, анализ социальных сетей, моделирование физических процессов. В таких задачах объём данных может достигать терабайт, и последовательная обработка становится физически невоз-

можной из-за ограничений по времени и памяти. Параллельные алгоритмы позволяют распределить как вычисления, так и хранение данных между множеством узлов, обеспечивая масштабируемость системы [4, 6].

Какую архитектуру выбрать? Если задача устроена просто (много однородных данных) и нужен огромный кластер - берём MPI. Если всё вычисление помещается на одном узле и дальше масштабировать не надо - хватит OpenMP. А гибридные решения позволяют не гнаться за максимальной простотой, но выжать из железа почти всё, что оно может дать.

Важным практическим аспектом является выбор способа разбиения данных между процессами. При блочном разбиении каждый процессор получает непрерывный блок данных, что минимизирует накладные расходы на передачу, но может привести к дисбалансу, если сложность обработки разных блоков неодинакова. При циклическом разбиении (cyclic distribution) данные чередуются между процессорами, что обеспечивает более равномерную нагрузку, но увеличивает количество обменов. Оптимальный метод зависит от специфики решаемой задачи [4, 6].

Выводы. Параллельное программирование — это постоянный компромисс между теорией, железом и реальной задачей. Дальше всё будет крутиться вокруг гибридных моделей и алгоритмов, которые сами подстраиваются под нагрузку.

Изучение параллельных алгоритмов вместе с OpenMP и MPI даёт студентам реальный инструмент для работы на современных вычислительных системах. А когда такие темы входят в программу инженерных направлений, вуз лучше отвечает вызовам цифровой трансформации и готовит специалистов, которые умеют не просто писать код, а заставлять железо работать на полную катушку.

Литература

1. Антонов А. С. Параллельное программирование с использованием MPI и OpenMP. Москва: Изд-во МГУ, 2018. 320 с.
2. Ахо А., Хопкрофт Дж., Ульман Дж. Структуры данных и алгоритмы. Москва: Вильямс, 2001. 384 с.
3. Гергель В.П. Теория и практика параллельных вычислений. Москва: Интернет-университет информационных технологий, 2019. 424 с.
4. Кузнецов С.Д. Параллельные вычисления. Москва: Наука, 2017. 480 с.
5. Chandra R., Dagum L., Kohr D. Parallel Programming in OpenMP. San Francisco: Morgan Kaufmann, 2001. 230 p.
6. Cormen T. H., Leiserson C. E., Rivest R. L., Stein C. Introduction to Algorithms. 3rd ed. Cambridge: MIT Press, 2009. 1312 p.
7. Dongarra J., Foster I., Fox G. Sourcebook of Parallel Computing. San Francisco: Morgan Kaufmann, 2003. 842 p.
8. Gropp W., Lusk E., Skjellum A. Using MPI: Portable Parallel Programming with the Message-Passing Interface. Cambridge : MIT Press, 2014. 350 p.

PARALLEL ALGORITHMS AND THEIR SCALING IN MULTIPROCESSOR SYSTEMS

Mitkina Ekaterina Andreevna

student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: mitkina.katya06@gmail.com

Matytsin Artem Vladimirovich

student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artyom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The principles of building parallel algorithms for multiprocessor systems are considered. The Amdahl's law, which determines the theoretical limit of acceleration as the number of computing nodes increases, is analyzed. The main technologies of parallel programming, such as OpenMP for shared memory systems and MPI for distributed systems, are described. Examples of parallelization of sorting and matrix operations are provided. The choice of architecture and parallelization methods is justified based on the task structure and available computing resources.

Keywords: parallel algorithms; multiprocessor systems; Amdahl's law; OpenMP; MPI; parallelization of sorting; matrix operations.

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ ПРИНЯТИЯ РЕШЕНИЙ В КОММЕРЧЕСКОМ БАНКЕ

© **Обыдённов Александр Николаевич**

заместитель директора,

Тамбовский региональный филиал АО «Россельхозбанк»

Россия, 392000, г. Тамбов, ул. Интернациональная, 39

E-mail: obydenov@bk.ru

Аннотация. В статье рассматриваются происходящие в настоящее время процессы изменения внутренних бизнес-процессов в современном российском коммерческом банке. Особое внимание уделено современным системам бизнес-аналитики при принятии управленческих решений. Приводятся практические примеры использования искусственного интеллекта в чат-ботах, «умных» инструментах маркетинга, алготрейдинге, инвестиционном консультировании и оценке стоимости активов, скоринге, подтверждении операций, обработке документов и мониторинге транзакций. Проведена оценка новых возможностей автоматизированных банковских систем. Рассматриваются форматы предоставления облачных услуг применительно к специфике ведения банковского бизнеса. Итогом исследования стал вывод, что предоставляемые диджитализацией возможности имеют значительный потенциал в банковской сфере.

Ключевые слова: цифровая трансформация банковской деятельности, цифровизация, банковские бизнес-процессы, цифровизация банковских бизнес-процессов, искусственный интеллект в банках, облачные технологии в банках.

В учебной литературы 2013 г. описывают пожелание председателя правления банка «А» о том, чтобы вся деятельность банка была формализованной, и чтобы по нажатию кнопки на компьютере он мог видеть работу любого сотрудника любого бизнес-процесса банка: его цели, показатели, технологии, результаты и т.д. [1, с. 12].

На сегодняшний день можно с уверенностью утверждать, что мечта банковского руководителя сбылась. Особенности цифровой трансформации внутри-банковских бизнес-процессов посвящена настоящая статья.

Актуальность темы обусловлена тем, что рассматриваемый процесс в настоящее время находится в активной стадии практической реализации.

Объект исследования – внутренние банковские бизнес-процессы.

Предмет исследования – цифровая трансформация внутренних банковских бизнес-процессов.

В статье применён системный подход, основанный на методах сравнения и дедукции.

Затронутую тему нельзя назвать широко разработанной, поскольку процесс цифровизации банковского бизнеса находится в стадии реализации и в научной литературе пока не получил достаточного освещения.

Описание исследования

Текущий уровень автоматизации внутренних процессах достиг таких высот, что действующее программное обеспечение даёт возможность вести учёт теку-

ших результатов деятельности вплоть до конкретного специалиста и позволяет в процессе управления продажами своевременно принимать необходимые меры с целью корректировки западающих зон. В банковской деятельности находят широкое применение современные системы бизнес-аналитики BI (business intelligence), которые способны собрать информацию из различных источников, обработать, очистить и обогатить полученные данные, и, в конечном итоге, предоставить удобный для восприятия отчет, позволяющий «провалиться» в каждый показатель, рассмотреть его в динамике за заданный период и определить влияющие на показатель факторы.

Все большее влияние на функционирование внутрибанковских процессов начинает оказывать искусственный интеллект (ИИ). Примеры использования искусственного интеллекта приведены в Таблице 1.

Таблица 1

Примеры использования искусственного интеллекта в финансовых организациях

Область применения.	Сфера применения.	Тип искусственного интеллекта.
Первая линия: фронт-офис	Чат-боты	ИИ-алгоритмы на основе обработки естественного языка
	«Умные» инструменты маркетинга	ИИ-алгоритмы с элементами глубокого обучения для формирования персональных предложений
	Алготрейдинг, инвестиционное консультирование и оценка стоимости активов	ИИ-алгоритмы на основе регрессионных моделей для оценки показателей
Вторая линия: проверка и обработка операций	Скоринг	Приложения на основе ИИ-алгоритмов для проведения кредитного скоринга, доступные в том числе клиентам
	Подтверждение операций, обработка документов	ИИ-алгоритмы, подключенные к сведениям о платежных данных и системе управления рисками для мгновенного принятия решений
Третья линия: операционный учет	Мониторинг транзакций	ИИ-алгоритмы, способные в режиме реального времени отслеживать подозрительные операции и оповещать пользователей
Общее	Разработка, оптимизация ПО	Сквозная сервисная функция для бизнес-процессов

Кредитные организации идут по пути внедрения модели ИИ-Плюс, при которой искусственный интеллект становится центральной технологией в бизнесе, не только касающемся разработки и IT. Помимо того, что ИИ внедряется в ключевые функциональные области и процессы, уже сейчас появляются бизнесы, в которых ИИ является основой [2, с. 47].

Достижения цифровых технологий оказывают неоценимую помощь в управлении качеством клиентского обслуживания. В банковскую практику входит использование инструментов искусственного интеллекта с целью оценки соот-

ветствия речи и поведения операционных работников установленной модели продаж. Используется и видеоаналитика, позволяющая оценить эмоции клиентов.

Одним из важнейших последствий цифровой трансформации для внутрибанковских процессов является повышение скорости принятия решений. Речь не только о повышении эффективности кредитного скоринга, но и о значительном ускорении процесса принятия широкого перечня управленческих решений. Современные технологии обработки больших данных (Big Data) и инструменты ИИ способны практически мгновенно предложить варианты решений тех или иных практических кейсов, оценив их потенциал и последствия, а руководителю, опираясь свой опыт и интуицию, остаётся лишь принять наиболее правильное решение. Уже сейчас можно прогнозировать последующую передачу части управленческих решений (в первую очередь, основанных на алгоритмах) искусственному интеллекту.

Уместным будет привести мнение Ма Хуатэна, изложенное в его книге «Цифровая трансформация Китая», который призывает не ставить самые важные решения в зависимость от данных. По его мнению, первостепенное значение для принятия решений имеет опыт менеджера, прямое общение с заказчиком, личный опыт использования продуктов, наблюдение за потребительскими сценариями, а также здравый смысл и интуиция [3, с. 229].

При текущем уровне развития технологий приведённое мнение представляется вполне обоснованным.

Любая цифровая трансформация начинается с цифровизации внутренних коммуникаций. На сегодняшний день практически все каналы внутрибанковских коммуникаций перешли на онлайн формат. Бумажное оформление требуется в тех редких случаях, когда подобное требование установлено действующими нормативными актами.

Центральное место в обеспечении проведения банковских операций любого банка занимают автоматизированные банковские системы (АБС), в которых на текущем этапе развития происходит процесс замены единых унифицированных программных приложений (монолитных приложений) на более гибкие автономные сервисы, отвечающие за выполнение какой-либо одной функции (микросервисы), что позволяет оптимизировать обслуживание клиентов в дистанционных каналах [4, с. 55].

Огромным потенциалом для организации внутрибанковских процессов обладают облачные технологии.

Процесс цифровой трансформации является достаточно затратным для банков. Расходы состоят из затрат на приобретение компьютерного оборудования, лицензий на программное обеспечение и т.д. Облачные технологии позволяют приобрести их (технологии) как сервис и работать с ними удалённо.

В Таблице 2 представлены форматы предоставления облачных услуг.

Форматы предоставления облачных услуг

Наименование формата	Содержание
IaaS (Infrastructure as a Service)	Аппаратное обеспечение, типовые серверы без настройки.
PaaS (Platform as a Service)	Дополнительно: API и реализующие их компоненты, среда разработки.
SaaS (Software as a Service)	Дополнительно: облачное автоматизированное рабочее место.
BaaS (Bank / Business as a Service)	Дополнительно: автоматизированные бизнес-процессы, соответствующие регуляторным требованиям.

В российском банковском сегменте облачные технологии получили широкое распространение в сфере автоматизации проведения банковских операций. Так, АБС от российской компании «Центр финансовых технологий» используется многими банками [5]. Современный банкинг сложно приставить без использования систем CRM (customer relationship management), которые представляют собой программное обеспечение для бизнеса, которое позволяет управлять продажами и работой сотрудников, автоматизируя продажи, маркетинг, аналитику, клиентский сервис. Значительным потенциалом в банковском ретейле обладает BPM-система (англ. business process management, управление бизнес-процессами), которая в отличие от систем CRM, выстраивает взаимодействие не между разными структурными подразделениями организации, а между бизнес-процессами, каждый из которых рассматривается как определённый ресурс.

Огромный интерес представляет методика разработки программного обеспечения Agile (англ. гибкий), предполагающая вовлечение каждого участника проектной команды в процесс переосмысления своих задач и общей цели – создание качественного продукта.

Agile — прямая противоположность методологии Waterfall («Водопад»). Waterfall была доминирующей методологией управления проектами во второй половине XX века.

Сбер и Альфа-Банк уже используют методику Agile в своих процессах [4, с.100]. Активную роль в развитии технологий SupTech и RegTech играет Центральный Банк Российской Федерации. Технологии SupTech (Supervisory Technology) использует Банк России для повышения эффективности контроля и надзора за деятельностью участников финансового рынка. RegTech (Regulatory Technology) – технологии, используемые финансовыми организациями для повышения эффективности выполнения требований регулятора [6, с. 2].

RegTech-решения представляют собой использование банками цифровых технологий в обеспечение информационной безопасности, внедрение датацентричного подхода, развитие решений на финансовом рынке для реализации регуляторных и надзорных требований.

К примеру, в Австралии с помощью технологий голосовой аналитики проводится мониторинг недобросовестных практики при продажах банковских продуктов (введение клиентов в заблуждение) [6, с. 8].

Российские банки активно начинают использовать в операционной деятельности технологии RPA (Robotic Process Automation). Применение данной технологии позволяет снизить операционные расходы, увеличить скорость обработки операций, исключить ошибки, связанные с человеческим фактором.

Заключение

Предпринятый анализ позволяет сделать вывод, что предоставляемые диджитализацией возможности имеют значительный потенциал в банковской сфере. Происходящие качественные изменения систем бизнес-аналитики и использование возможностей искусственного интеллекта при принятии решения и разработке новых инновационных продуктов позволят вывести банковский бизнес на новый, более высокий уровень.

Литература

1. Исаев Р. А. Банковский менеджмент и бизнес-инжиниринг: в 2 томах. 2-е изд., перераб. и доп. Москва: ИНФРА-М, 2013. Т. 1. 286 с.
2. Исследование «3×10 трендов 2025 года» // Ассоциация ФинТех. 2025. 74 с. URL: <https://www.fintechru.org/press-center/issledovaniya/3kh10-trendov-2025-goda> (дата обращения: 08.05.2026).
3. Ма Хуатен, Мэн Чжаоли, Ян Дели, Ван Хуалей. Цифровая трансформация Китая. Опыт преобразования инфраструктуры национальной экономики: пер. с кит. Москва: Интеллектуальная литература, 2019. 250 с.
4. Рудакова О. С., Маркова О. М., Мартыненко Н. Н. Финансовые технологии в банках: учебник / под ред. О. С. Рудаковой. Москва: КНОРУС, 2024. 320 с.
5. Универсальный банк // Центр финансовых технологий. URL: <https://catalog.cft.ru/solutions/universal/overview> (дата обращения: 08.05.2026).
6. Основные направления развития технологий SupTech и RegTech на период 2021–2023 годов: доклад // Банк России. 2021. 27 с. URL: https://cbr.ru/Content/Document/File/120709/SupTech_RegTech_2021-2023.pdf (дата обращения: 08.05.2026).

INTELLIGENT DECISION-MAKING SYSTEMS IN A COMMERCIAL BANK

Alexander Nikolaevich Obydenov

Deputy Director

Tambov Regional Branch of JSC Rosselkhozbank

39 Internatsionalnaya Street, Tambov, 392000, Russia

E-mail: obydenov@bk.ru

Abstract. The article discusses the ongoing processes of changing internal business processes in a modern Russian commercial bank. Special attention is paid to modern business analytics systems in making managerial decisions. The article provides practical examples of using artificial intelligence in chatbots, smart marketing tools, algorithmic trading, investment consulting, asset valuation, scoring, transaction verification, document processing, and transaction monitoring. The article also evaluates the new capabilities of automated banking systems. The article examines the formats of providing cloud services in relation to the specifics of banking. The study concludes that the opportunities provided by digitalization have significant potential in the banking sector.

Keywords: digital transformation of banking, digitalization, banking business processes, digitalization of banking business processes, artificial intelligence in banks, cloud technologies in banks.

ИДЕНТИФИКАЦИЯ ПАРАМЕТРОВ ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ ОПТИМИЗАЦИИ РЕГРЕССИОННОГО ТЕСТИРОВАНИЯ НА ОСНОВЕ МОДЕЛЕЙ ПРИОРИТИЗАЦИИ

© Матафонов Денис Сергеевич

аспирант, ассистент,

Уральский федеральный университет имени первого Президента России Б. Н. Ельцина
Россия, 620049, г. Екатеринбург, ул. Софьи Ковалевской, 5

E-mail: denis.matafonov@urfu.ru

Аннотация. Регрессионное тестирование является одним из наиболее ресурсоёмких этапов разработки цифровых продуктов. Современные методы приоритизации тест-кейсов, включая подходы на основе машинного обучения, позволяют сократить время прогона, однако их внедрение сдерживается отсутствием формализованных требований к исходным данным. В работе проведён анализ существующих подходов к приоритизации, предложена концептуальная модель и идентифицирован полный перечень входных параметров, извлекаемых из платформы управления тестированием, системы управления задачами и версиями, что позволяет подготовить проект к внедрению системы оптимизации тестирования.

Ключевые слова: регрессионное тестирование, приоритизация тест-кейсов, идентификация параметров, платформа управления тестированием, система управления задачами, оптимизация тестирования.

Введение. Современная разработка цифровых продуктов, особенно в сфере финтеха, характеризуется коротким циклом «разработка–тестирование–релиз». В этих условиях регрессионное тестирование (РТ) превращается в одно из самых ресурсоёмких звеньев производственного процесса, отнимает значительные ресурсы команды тестирования и создаёт узкое место в производственном цикле.

Интуитивно понятно, что тестовые случаи (тест-кейсы), которые давно не выявляли ошибок и относятся к редко изменяемым модулям, можно либо убрать из регулярного прогона, либо передать под автоматизацию. Однако подобные решения требуют количественного обоснования. Современная литература предлагает широкий спектр методов приоритизации тестов (Test Case Prioritization — TCP), основанных на машинном обучении (ML), анализе исторических данных о падениях и информации об изменениях кода. Все эти подходы объединяет общая проблема: их практическая применимость напрямую зависит от доступности и качества исходных данных. Без идентификации того, какие именно параметры могут быть извлечены из конкретной информационной среды (платформа управления тестированием, система управления задачами, система контроля версий), любая математическая модель останется лишь теоретической конструкцией.

Цель данной работы — на основе анализа существующих подходов к TCP предложить единую концептуальную модель приоритизации и идентифицировать полный набор входных параметров, которые необходимо извлечь из до-

ступных информационных систем для её практической реализации в условиях реальной продуктовой команды.

Обзор литературы и степень изученности темы

Проблема оптимизации регрессионного тестирования активно исследуется в мировой литературе. Традиционно выделяют три направления: отбор тестов (test selection), минимизацию набора (test minimization) и приоритизацию (test prioritization) [2].

Значительная часть современных работ опирается на исторические данные о выполнении тестов (History-based Test Prioritization – НВТР). Исследования показывают, что знание о предыдущих падениях тестов обладает высокой предсказательной силой в среде непрерывной интеграции (CI), причём для эффективной приоритизации не требуется большого объёма исторических данных [3]. Подтверждается применимость НВТР в промышленных условиях, в том числе для ручного тестирования [2].

Развитием статических методов стали динамические алгоритмы, переупорядочивающие очередь тестов по мере получения результатов. Торбунова и др. предложили схему, сочетающую статический и динамический алгоритмы, где динамическая часть использует условную вероятность отказов коррелированных тестов [6].

В последние годы активно развиваются ML-подходы к ТСП. Сравнительное исследование 11 репрезентативных ML-методов на трёх индустриальных датасетах показало, что методы на основе деревьев решений и ансамблей способны стабильно превосходить традиционные НВТР-эвристики [4]. Данилов и Мугатина предложили нейросетевой подход, в котором входными данными выступают сведения об изменениях кода из системы контроля версий, а выходными — прогноз обнаружения ошибки [1]. Существует таксономия информационных атрибутов, используемых в ТСП-исследованиях, классифицированная по их источникам и способам извлечения [4, 5].

Несмотря на обилие методов, все они предъявляют требования к определённому набору входных параметров. В реальных производственных условиях данные часто фрагментированы между несколькими системами (платформа управления тестами, таск-трекер, репозиторий кода), а связи между ними не всегда формализованы. Именно этот разрыв между теоретическими моделями и практической доступностью данных и призвана сократить данная работа.

Единая концептуальная модель приоритизации тест-кейсов

Обобщая рассмотренные подходы, можно выделить четыре универсальные группы факторов, определяющих приоритет тест-кейса:

- 1) Историческая эффективность обнаружения дефектов — как часто и насколько серьёзные дефекты данный кейс выявлял в прошлом.
- 2) Степень изменения покрываемого функционала — модифицировался ли соответствующий модуль в текущем релизе.
- 3) Временные затраты на выполнение — длительность ручного прогона кейса.
- 4) Взаимосвязи между тест-кейсами — корреляции отказов (для динамической приоритизации).

Этим группам факторов можно поставить в соответствие интегральный показатель приоритета для соответствующего тест-кейса, вычисляемый как свёртка нормализованных параметров с весовыми коэффициентами. В общем виде модель приоритизации формализуется как задача многокритериальной оптимизации, а при наличии ведущего ограничения — времени — сводится к задаче о рюкзаке: требуется отобрать подмножество кейсов с максимальным суммарным приоритетом, укладываясь в бюджет времени. Однако ключевой вопрос заключается не в выборе конкретной формулы, а в том, какие именно исходные параметры необходимы для вычисления компонентов модели и могут ли они быть получены из доступных информационных систем. Идентификация этих параметров позволяет впоследствии построить любую из рассмотренных моделей — от эвристической до ML-основанной.

Идентификация входных параметров

На основе анализа существующих подходов и специфики типовой информационной среды (платформа управления тестированием, система управления задачами, система контроля версий) предлагается следующая спецификация входных параметров, сгруппированных по источнику происхождения.

Платформа управления тестированием хранит тест-кейсы и историю их выполнения. Через API таких систем возможно получение параметров, рассмотренных в таблице 1.

Таблица 1

Параметры из платформы управления тестированием

Параметр	Обозначение	Тип	Описание
История результатов	$R_{i,k}$	Бинарный	Результат i -го кейса в k -м прогоне
Длительность выполнения	t_i	Числовой	Время ручного прогона кейса в человеко-часах
Привязка к функциональному модулю	M_i	Категориальный	Идентификатор модуля, к которому относится кейс
Приоритет (экспертный)	Pr_i	Порядковый	Установленный вручную приоритет
Количество шагов	s_i	Числовой	Количество шагов в тест-кейсе

Система управления задачами (например, YouTrack, Jira) фиксирует баги и управляет релизами.

Таблица 2

Параметры, извлекаемые из системы управления задачами

Параметр	Обозначение	Тип	Описание
Серьёзность дефекта	sev_j	Порядковый	Градация серьёзности
Дата обнаружения дефекта	d_j	Дата/время	Временная метка создания бага
Связь с тест-кейсом	L_{ij}	Бинарный	Индикатор: обнаружен ли баг j при выполнении кейса i
Принадлежность к релизу	Rel_j	Категориальный	Идентификатор релиза, в котором обнаружен дефект

Для корректного расчёта исторической эффективности необходимо внедрить практику явного связывания бага с тест-кейсом (например, через пользовательское поле в форме баг-репорта). До формализации этой связи допустимо временное сопоставление по близости временных меток.

Система контроля версий (чаще Git) — источник информации об изменениях кода, критически важных для оценки вероятности регрессии.

Таблица 3

Параметры, извлекаемые из Git

Параметр	Обозначение	Тип	Описание
Факт изменения модуля	C_m	Бинарный	Был ли модифицирован модуль m в текущем релизе
Количество изменённых файлов	n_m^{files}	Числовой	Количество изменённых файлов в модуле
Количество изменённых строк	n_m^{lines}	Числовой	Суммарное количество добавленных и удалённых строк в модуле
Автор изменений	$auth_m$	Категориальный	Разработчик, вносивший изменения
Дата последнего изменения модуля	d_m^{last}	Дата/время	Временная метка последнего коммита в модуле

Для сопоставления изменений в Git с тест-кейсами необходима карта отображения «файл/директория – функциональный модуль – набор тест-кейсов». Она может быть построена на основе структуры тестового набора и договорённостей команды о модульной организации кода.

На основе извлечённых данных вычисляются производные параметры, непосредственно входящие в модель приоритизации:

Таблица 4

Производные (расчётные) параметры

Параметр	Обозначение	Формула / способ расчёта
Частота обнаружения дефектов	F_i	$F_i = \frac{\sum_k w_k \cdot R_{i,k}}{\sum_k w_k}$
Количество дефектов, найденных кейсом	m_i	$m_i = \sum_j L_{i,j}$
Средняя серьёзность дефектов	S_i	$S_i = \frac{1}{m_i} \sum_{j: L_{i,j}=1} sev_j$
Индекс изменения модуля	C_i	$C_i = 1 \text{ если модуль менялся в текущем релизе, иначе } e^{-\mu \cdot \Delta t_i}, \text{ где } \Delta t_i - \text{ количество релизов после последнего изменения модуля}$

Таким образом, предложенный перечень параметров обеспечивает вычислимость всех компонентов, необходимых для построения аддитивной модели приоритизации, а также предоставляет признаки для обучения ML-моделей, описанных в литературе.

Проведённый анализ позволяет сформулировать минимально необходимый и оптимальный набор параметров для внедрения модели приоритизации.

Необходимые параметры:

- история результатов тест-кейсов за последние 5–10 прогонов);
- время выполнения каждого кейса;
- привязка тест-кейса к функциональному модулю;
- факт изменения модуля в текущем релизе.

Дополнительные параметры:

- серьёзность обнаруженных дефектов;
- количественные метрики изменений кода;
- данные для динамической приоритизации (корреляции отказов тестов).

Заключение. В данной работе проведён анализ существующих подходов к приоритизации тест-кейсов в регрессионном тестировании и предложена единая концептуальная модель, объединяющая четыре группы факторов: историческую эффективность, степень изменения функционала, временные затраты и взаимосвязи тестов.

Основным результатом работы является идентификация полного перечня входных параметров, сгруппированных по источникам (платформа управления тестированием, система управления задачами, система контроля версий). Пока-

зано, что для запуска модели в минимальной конфигурации достаточно данных, уже доступных в платформе управления тестированием и Git. Предложенная спецификация может служить дорожной картой при подготовке проекта к внедрению системы оптимизации регрессионного тестирования. Дальнейшим направлением исследований становится экспериментальная валидация модели.

Литература

1. Данилов А. Д., Мугатина В. М. Решение задачи оптимизации регрессионного тестирования с использованием нейросетевого подхода // Моделирование, оптимизация и информационные технологии. 2020. Т. 8, № 1.
2. Lou Y., Chen J., Zhang L., Hao D. A Survey on Regression Test-Case Prioritization. *Advances in Computers*. Elsevier, 2019; 113: 1–46.
3. Hassan S. M. J., Jawawi D., Ahmad J. An exploratory study of history-based test case prioritization techniques on different datasets. *Baghdad Science Journal*. 2024; 21(6).
4. Kiran S., Emre İ., Taşdelen S. Prioritization of Regression Test Cases Based on Machine Learning Methods. *GAZI UNIVERSITY JOURNAL OF SCIENCE*. 2024; 38.
5. Ramírez A., Feldt R. A Taxonomy of Information Attributes for Test Case Prioritisation: Applicability, Machine Learning.

IDENTIFICATION OF INFORMATION SYSTEM PARAMETERS FOR OPTIMIZATION OF REGRESSION TESTING BASED ON PRIORITIZATION MODELS

Denis Sergeevich Matafonov

Postgraduate Student, Assistant

Ural Federal University named after the first President of Russia B.N. Yeltsin

Russia, 620049, Yekaterinburg, Sofya Kovalevskaya Street 5

E-mail: denis.matafonov@urfu.ru

Abstract. Regression testing is one of the most resource-intensive stages of digital product development. Modern test case prioritization methods, including machine learning-based approaches, can reduce the test run time, but their implementation is hindered by the lack of formalized requirements for the input data. This paper analyzes existing prioritization approaches, proposes a conceptual model, and identifies a comprehensive list of input parameters extracted from the testing management platform, task management system, and version management system, which helps prepare the project for the implementation of a testing optimization system.

Keywords: regression testing, test case prioritization, parameter identification, test management platform, task management system, and testing optimization.

ПРОБЛЕМЫ ВНЕДРЕНИЯ СИСТЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В МЕДИЦИНСКУЮ ПРАКТИКУ: РЕГУЛЯТОРНЫЕ, ИНТЕРПРЕТАЦИОННЫЕ И ЭТИЧЕСКИЕ АСПЕКТЫ

© **Сенатов Максим Андреевич**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: maksimsenatov-bebebe@yandex.ru

© **Матыцин Артем Владимирович**

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© **Гаев Леонид Витальевич**

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. В статье был выполнен анализ проблем, которые возникают при переходе от лабораторных условий, в которых проверяется искусственный интеллект, к реальным условиям клинической практики. Показаны основные препятствия, среди которых расхождение результатов, полученных в ходе экспериментов, и реальной практики врачей; знание только результата, пока логика его получения остаётся в «чёрном ящике»; правовая, а также этическая неопределённости, такие, как систематические ошибки и несформированность законодательства. Опираясь на современные российские и зарубежные исследования, предлагаются решения, такие как перестройка регулирования, внедрение объяснимого ИИ, а также снижение влияния систематических ошибок на результат. Статья пригодится разработчикам медицинских систем с искусственным интеллектом, врачам и регуляторам.

Ключевые слова: искусственный интеллект, медицинская диагностика, регуляторные барьеры, объяснимый ИИ, систематические ошибки, этические риски, валидация алгоритмов.

Введение. В последние годы количество медицинских устройств растёт экспоненциально. Регуляторы и в России, и за рубежом фиксируют удвоение каждые полтора-два года [4]. При этом за красивыми цифрами в 95% чувствительности либо площади под ROC-кривой скрывается ворох проблем. Парадокс ситуации в том, что хоть технический прогресс и шагнул далеко вперёд, но методы проверки и юридические нормы не поспевают за ним. В итоге системы, запущенные в тестовом режиме либо игнорируются врачами в их рутинных задачах, либо их результаты становятся скромнее, чем в лабораторных условиях.

Актуальность. Несмотря на экспоненциальный рост числа медицинских устройств на основе ИИ и высокие показатели точности в лабораторных усло-

виях, реальное внедрение в клиническую практику сталкивается с серьёзными барьерами. Расхождение результатов валидации и реальной эффективности, проблема «чёрного ящика», правовая и этическая неопределённость приводят к тому, что многие системы игнорируются врачами или дают худшие результаты, чем ожидалось. Поэтому анализ и систематизация проблем внедрения ИИ в медицину является актуальной задачей.

Объект и предмет исследования. Объектом исследования являются процессы внедрения систем искусственного интеллекта в медицинскую практику. Предметом исследования — регуляторные, интерпретационные и этические проблемы, возникающие при переходе от лабораторных испытаний к реальной клинической работе.

Цель и задачи. Целью работы является выявление и систематизация проблем внедрения систем ИИ в медицину, а также разработка рекомендаций по их преодолению.

Задачи исследования:

1. Проанализировать регуляторные барьеры при переходе от лабораторных испытаний к клинической практике.
2. Исследовать проблему интерпретируемости решений ИИ («чёрный ящик»).
3. Выявить этические риски и правовую неопределённость при использовании ИИ.
4. Сформулировать практические рекомендации для разработчиков, врачей и регуляторов.

Методология и методы исследования. В работе использован теоретический анализ российских и зарубежных исследований по внедрению ИИ в медицину. Применён сравнительный анализ успешных и неудачных кейсов (MASAI, валидация коммерческих систем). На основе обобщения предложены рекомендации по преодолению выявленных барьеров.

Разработанность темы. Проблематика внедрения ИИ в медицину рассматривается в работах Алексеевой, Ванькова, а также в зарубежных исследованиях MASAI, SHAMSUL и др. Вопросы регуляторного регулирования анализируются FDA. Однако большинство работ фокусируются на отдельных аспектах (технических или этических). Данная статья предлагает комплексный взгляд на регуляторные, интерпретационные и этические проблемы.

Результаты исследования. Регуляторный разрыв: от лаборатории до реальной больницы. Одна из наиболее острых проблем — несоответствие между упрощённым порядком регистрации медицинских изделий на основе ИИ и требованиями доказательной медицины. Во многих юрисдикциях, также, как и в России, система регистрации ориентирована на проверку технических характеристик и безопасности [2]. Но оценка эффективности в реальной клинической практике остаётся вне этой системы. А это огромная разница.

Успешным примером является исследование MASAI (Швеция). В контролируемом исследовании со случайным распределением, опубликованном в 2024 году, сравнили два режима маммографического скрининга. Первый — стандартный (два рентгенолога). Второй — экспериментальный (рентгенолог плюс си-

стема с ИИ). Как оказалось, второй вариант не уступал первому по частоте выявления рака молочной железы, но также снижал нагрузку на врачей на 30-40%. И важный момент – здесь исследование проверяло систему человека и ИИ в реальных условиях, а не на лабораторных данных [5].

Примером же неудачного внедрения может послужить внешняя валидация коммерческой системы для анализа рентгенограмм грудной клетки, Европейское многоцентровое исследование, 2023 год. Система показала чувствительность 92% на внутренней тестовой выборке, что было хорошим результатом. Но при проверке на данных первичного звена чувствительность упала уже до 35% (при низкой распространённости патологии в районе 2%). Кроме того, использование ИИ увеличило среднее время описания снимка на 3–10 секунд. А врачи не доверяли его заключениям в 60% ложноположительных случаях.

Основной вывод здесь очевиден. Без внешней валидации на когортах, с типичной региональной практикой и структурой заболеваемости, многие системы ИИ могут стать бесполезными или даже вредными.

Проблема «чёрного ящика»: интерпретируемость как условия доверия. Второй серьёзный барьер – непрозрачность нейросетей. Врач в своей работе привык опираться на причинно-следственные связи. Если система ставит диагноз либо назначает лечение, но не объясняет почему она так решила, уровень доверия резко падает.

В последние годы активно разрабатываются подходы, которые позволяют понять, как «думает» модель. Они называются методами объяснимого искусственного интеллекта. Основными на данный момент являются Grad-CAM – это построение тепловых карт, показывающих, какие области изображения повлияли на решение модели; SHAP и LIME – методы, оценивающие вклад отдельных признаков для каждого конкретного случая в итоговый прогноз; и LRP – метод, прослеживающий влияние каждого пикселя через все слои сети.

Сравнительное исследование SHAMSUL (2024 год) показало: Grad-CAM даёт лучшие количественные показатели, но врачи посчитали самым полезным в практике метод LIME – его визуализации лучше согласовывались с клиническим опытом [3]. Золотого стандарта объяснимости нет, всё зависит от поставленной задачи и того, кому нужно объяснение.

Этические риски и правовая неопределённость. Даже при наличии объяснимого ИИ остаются риски систематической ошибки. А также остаётся нерешённый вопрос об юридической ответственности.

Примерами предвзятости алгоритмов могут служить два следующих исследования. Исследователи Гарвардской медицинской школы (2025) обучили нейросеть на гистологических срезах. Модель научилась неявно кодировать расу, возраст и пол пациента – хотя эти признаки ей не передавали. Как результат – точность диагностики рака лёгких у афроамериканцев и рака молочной железы у молодых женщин оказалось ниже. Контрастивное обучение уменьшило этот демографический сигнал, но убрать его полностью не получилось.

Рекомендации и предложения. На основе выполненного анализа можно предложить некоторые меры. Они призваны помочь преодолеть описанные проблемы.

Для разработчиков можно порекомендовать обязательную внешнюю валидацию на репрезентативных клинических когортах с низкой распространённостью целевых состояний. Публикацию вместе с метриками точности результатов применения методов объяснимого ИИ [1].

Выводы. Было разобрано три барьера – регуляторный, интерпретационный и этико-правовой. Но во всех случаях оказалось, что проблема не в технологиях, а в том, как они проверяются, внедряются и сопровождаются юридически. Даже высокоточные модели, успешно прошедшие испытания в лабораториях, на практике могут показывать результаты гораздо хуже: увеличивать время обработки, повышать уровень систематических ошибок.

Только при выполнении этих условий потенциал искусственного интеллекта – существенно повысить качество и доступность медицинской помощи – может быть реализован без неприемлемых рисков для пациентов.

Литература

1. Алексеева М. Г., Зубов А. И., Новиков М. Ю. Искусственный интеллект в медицине // Международный научно-исследовательский журнал. 2022. № 7(121). URL: <https://cyberleninka.ru/article/n/iskusstvennyy-intellekt-v-meditsine-3> (дата обращения: 21.05.2026).
2. Итоги внедрения искусственного интеллекта в здравоохранении России / В. В. Ваньков, О. Р. Артемова, О. Э. Карпов и др. // Врач и информационные технологии. 2024. URL: <https://cyberleninka.ru/article/n/itogi-vnedreniya-iskusstvennogo-intellekta-v-zdravooohranenii-rossii> (дата обращения: 21.05.2026).
3. Alam M. U., Rahman R., Azam S., Hossain M. S. SHAMSUL: Systematic Holistic Analysis to investigate Medical Significance Utilizing Local interpretability methods in deep learning for chest radiography prediction. *Nordic Machine Intelligence*. 2023; 3: 27–47. URL: <https://arxiv.org/abs/2307.08003> (дата обращения: 21.05.2026).
4. Artificial Intelligence-Enabled Medical Devices // U.S. Food and Drug Administration. URL: <https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-enabled-medical-devices> (дата обращения: 21.05.2026).
5. MASAI Trial Steering Group. Artificial intelligence versus double reading in mammography screening: final results of the MASAI randomized trial // *The Lancet*. 2026. URL: [https://www.thelancet.com/journals/lancet/article/PIIS0140-6736\(25\)02464-X/abstract](https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(25)02464-X/abstract) (дата обращения: 21.05.2026).
6. Schneider D., Brown E., Ward M., Sciubba D., Lo S.-F. The authorization lottery: contradictory AI prioritization patterns in healthcare resource allocation. *AI and Ethics*. 2025; 5: 5575–5585. URL: <https://link.springer.com/article/10.1007/s43681-025-00795-5> (дата обращения: 21.05.2026).

PROBLEMS OF INTRODUCING ARTIFICIAL INTELLIGENCE SYSTEMS INTO MEDICAL PRACTICE: REGULATORY, INTERPRETATIONAL AND ETHICAL ASPECTS

Senatov Maxim Andreevich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: maksimsenatov-bebebe@yandex.ru

Matytsin Artem Vladimirovich

Student

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: matytsin.artiom@gmail.com

Gaev Leonid Vitalievich

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

398055, Russia, Lipetsk Region, Lipetsk, Moskovskaya St., 30

E-mail: gaev@stu.lipetsk.ru

Abstract. The article analyzes the problems that arise when moving from laboratory conditions in which artificial intelligence is tested to real clinical practice. The main obstacles are shown, among which the divergence of the results obtained during the experiments and the real practice of doctors; knowledge only of the result, while the logic of its obtaining remains in the "black box"; legal and ethical uncertainties, such as systematic errors and the lack of legislation. Based on modern Russian and foreign research, the article proposes solutions such as restructuring regulation, introducing explainable AI, and reducing the impact of systematic errors on the outcome. This article will be useful for developers of medical systems with artificial intelligence, doctors, and regulators.

Keywords: artificial intelligence, medical diagnostics, regulatory barriers, explainable AI, systematic errors, ethical risks, algorithm validation.

КОМПЛЕКСНАЯ АВТОМАТИЗАЦИЯ ОПЕРАТИВНОГО И ФИНАНСОВОГО УЧЕТА ВЕТЕРИНАРНОЙ КЛИНИКИ КАК ИНСТРУМЕНТ ПОВЫШЕНИЯ РЕНТАБЕЛЬНОСТИ БИЗНЕСА

© **Немчинов Алексей Игоревич**

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: alexej.nemchinov@yandex.ru

© **Цыбикова Туяна Сандаликовна**

кандидат педагогических наук, доцент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: cts2001@mail.ru

Аннотация. В статье рассматриваются проблемы оперативного и финансового учета в ветеринарных центрах связанные с использованием бумажных журналов и разрозненных электронных таблиц, что приводит к ошибкам в расчетах, высоким временным затратам и отсутствию оперативной управленческой отчетности. На основе анализа бизнес-процессов действующего ветеринарного центра сформулированы требования к автоматизированной системе и обоснован выбор платформы «1С:Предприятие 8» в качестве основы для доработки. Приведены результаты внедрения разработанной системы демонстрирующие сокращение трудоемкости операций регистратуры и приема пациентов.

Ключевые слова: автоматизация учёта, 1С:Предприятие, ветеринарный центр, оперативный учёт, финансовый учёт, экономическая эффективность.

В современных условиях развития рынка ветеринарных услуг наблюдается устойчивый рост числа частных ветеринарных клиник и центров. Увеличение потока клиентов, расширение перечня предоставляемых услуг: прием терапевта, лабораторные диагностики и пр. приводит к значительному усложнению операционного и финансового учета. Большинство ветеринарных центров по-прежнему используют бумажные журналы, таблицы Excel или разрозненные программные продукты, что в свою очередь порождает такие проблемы, как дублирование данных, ошибки при расчете заработной платы сотрудников, отсутствие оперативной информации о наличии какого-препарата на складе, а также невозможности анализа рентабельности отдельных услуг. В связи с этим задача автоматизации оперативного и финансового учета в ветеринарных центрах становится актуальной.

Существующие на рынке специализированные программные решения для ветеринарных клиник, такие как «Ветменеджер», «4 Лапы» часто имеют высокую стоимость лицензий, ограниченные возможности для адаптации под конкретную клинику и слабую интеграцию с бухгалтерскими системами. Платформа «1С:Предприятие 8» представляет собой универсальную среду разработ-

ки и исполнения прикладных решений, обладающую совокупностью архитектурных и технологических характеристик, релевантных для создания интегрированной системы оперативного и финансового учета в ветеринарном центре. [2] Понимание возможностей данной платформы является необходимым условием для обоснования выбора инструментария при разработке конфигурации.

Важнейшим преимуществом платформы «1С:Предприятие» для целей настоящего исследования является ее открытость для адаптации. Конфигурации, созданные на данной платформе, как правило, доступны для редактирования (если разработчик не устанавливает программные механизмы защиты), что позволяет дорабатывать их под специфику конкретной организации.

Платформа предоставляет следующие инструменты для адаптации:

- Конфигуратор – среда разработки, позволяющая изменять структуру метаданных (создавать новые справочники, документы, регистры), писать программный код на встроенном языке, разрабатывать макеты печатных форм;
- Механизм расширений – возможность создания дополнительного функционала без изменения основной конфигурации, что упрощает обновление типовых конфигураций;
- Инструменты интеграции – поддержка обмена данными через универсальные форматы (XML, JSON, Excel, DBF), веб-сервисы, HTTP-запросы, что позволяет интегрировать конфигурацию с внешними системами (например, с сайтом записи клиентов, лабораторными информационными системами, государственными реестрами). [1]

Для автоматизации оперативного и финансового учета были детально изучены существующие бизнес-процессы ветеринарного центра. На основе анализа работы нескольких ветеринарных клиник была построена модель «AS-IS» (как есть), отражающая текущее состояние учета. Оперативный учет в ветеринарном центре включает ключевые процессы, отражены в таблице 1.

Таблица 1

Оперативный учет в ветеринарном центре

Процесс	Описание	Существующие проблемы
Регистрация пациента	Внесение данных о питомце (вид, порода, возраст, вес) и владельце (ФИО, контакты)	Дублирование данных при повторных визитах, ошибки в написании
Запись на прием	Фиксация даты, времени, врача и цели визита	Конфликты в расписании, отсутствие напоминаний для клиентов
Оказание услуги	Проведение осмотра, диагностики, назначения лечения	Не всегда фиксируют все манипуляции, потеря данных о динамике лечения
Учет материалов	Списание лекарственных препаратов, расходных материалов (перчатки, шприцы, перевязочные материалы)	Нет привязки списания к конкретному пациенту, ошибки в остатках
Введение медицинской карты	Фиксация диагнозов, назначений, результатов анализов	Разрозненное хранение, невозможность быстрого поиска истории

На основе анализа «AS-IS» были сформулированы следующие требования к разрабатываемой системе:

1. Единая база данных питомцев и их владельцев с возможностью быстрого поиска
2. Введение электронной медицинской карты с историей визитов, диагнозов, назначений и результатов анализов.
3. Автоматическое формирование расписания приема с контролем занятости врачей.
4. Учет услуг и материалов с автоматическим списанием по факту оказания услуги.
5. Интегрированные финансовый учет: расчет стоимости услуги на основе прайс-листа, фиксация оплаты, формировании актов и чеков.

Для реализации разработанных требований в качестве инструментальной платформы была выбрана система «1С:Предприятие 8.3». Выбор обосновывается следующими факторами:

1. Наличие отраслевого решения «БИТ:Айболит», которое может быть использовано в качестве базовой конфигурации, что сокращает сроки разработки и снижает технологические риски .
2. Возможность доработки конфигурации под специфические потребности ветеринарного центра, включая настройку справочников «Номенклатура» и «Питомцы/Клиенты», а также механизмов контроля списания препаратов.
3. Встроенные механизмы складского и бухгалтерского учета, позволяющие реализовать интеграцию медицинского и финансового учета на уровне единой базы данных.
4. Возможность обмена данными с «1С:Бухгалтерией 8» для формирования регламентированной отчетности и передачи данных в государственные информационные системы.
5. Широкая распространенность платформы, что обеспечивает доступность специалистов для сопровождения и дальнейшего развития системы.

После завершения разработки и тестирования конфигурации была проведена опытная эксплуатация в течение трех месяцев на базе исследуемого ветеринарного центра.

В таблице 2 представлено сравнение трудоемкости выполнения ключевых учетных операций до и после автоматизации. Замеры производились методом хронометража с участием 5 сотрудников различных категорий (администраторы, врачи, бухгалтер).

Сравнение трудоемкости учетных операций
(в минутах на одну операцию, средние значения)

Учетная операция	До автоматизации	После автоматизации	Сокращение, %
Регистрация обращения (запись на прием)	4,2	1,3	69,0
Оформление приема пациента (включая назначения)	8,5	4,1	51,8
Выписка рецепта	2,8	1,1	60,7
Оформление ПКО и выдача чека	3,0	0,8	73,3
Проведение инвентаризации склада	180,0	95,0	47,2
Расчет себестоимости услуги (за отчетный месяц)	480,0	5,0	99,0
Формирование управленческого отчета (выручка по услугам)	120,0	2,0	98,3
Подготовка данных для бухгалтерской отчетности	240,0	10,0	95,8

Как следует из таблицы 2 наиболее значительное сокращение трудоемкости (более 95%) достигнуто для операций расчета себестоимости, формирования управленческой отчетности и подготовки данных для бухгалтерии. Это объясняется тем, что данные операции ранее выполнялись вручную в электронных таблицах, а после автоматизации выполняются системой в автоматическом режиме на основании первичных документов.

Сокращение трудоемкости операций регистратуры и приема пациентов (от 50 до 70%) обусловлено устранением двойного ввода информации: ранее администратор вносил данные в бумажный журнал и ПКО, врач — в амбулаторную карту и рецепт. После автоматизации данные вводятся однократно в электронные формы и используются во всех последующих документах.

В результате проведенного исследования разработана и внедрена конфигурация на платформе «1С:Предприятие» для автоматизации оперативного и финансового учета в ветеринарных центрах.

Литература

1. Куправа Т. А. Управление 1С: 8.2. Редакция 11. Внедрение и применение. Москва: ДМК Пресс, 2021. 336 с.
2. Михайлов А.В. 1С: Предприятие 8.0. Москва: ДМК Пресс, 2018. 209 с.
3. Платформа «1С:Предприятие 8.3». Документация разработчика.

COMPREHENSIVE AUTOMATION OF OPERATIONAL AND FINANCIAL ACCOUNTING OF A VETERINARY CLINIC AS A TOOL FOR INCREASING BUSINESS RENEWAL

Nemchinov Alexey Igorevich

Student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: alexej.nemchinov@yandex.ru

Tsybikova Tuyana Sandalikovna

Candidate of Pedagogical Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: cts2001@mail.ru

Abstract. The article discusses the problems of operational and financial accounting in veterinary centers associated with the use of paper journals and scattered electronic tables, which leads to errors in calculations, high time costs, and a lack of operational management reporting. Based on the analysis of the business processes of an existing veterinary center, the requirements for an automated system are formulated, and the choice of the 1С:Enterprise 8 platform as the basis for further development is justified. The results of the implementation of the developed system are presented, demonstrating a reduction in the time required for registration and patient reception.

Keywords: accounting automation, 1С:Enterprise, veterinary center, operational accounting, financial accounting, economic efficiency.

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ЭФФЕКТА ИНФОРМАЦИОННОГО ПУЗЫРЯ В РЕКОМЕНДАЦИОННЫХ СИСТЕМАХ

© Казаков Никита Юрьевич

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: present2373@gmail.com

© Матыцин Артем Владимирович

студент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: matytsin.artiom@gmail.com

© Гаев Леонид Витальевич

кандидат технических наук, доцент,

Липецкий государственный технический университет

398055, Россия, Липецкая область, Липецк, ул. Московская, 30

E-mail: gaev@stu.lipetsk.ru

Аннотация. Исследуется эффект информационного пузыря (filter bubble) в рекомендательных системах как процесс самоподкрепляющегося сужения тематического разнообразия потребляемого пользователем контента. Цель — построить минимальную имитационную модель этого эффекта на открытых данных и количественно оценить зависимость скорости сужения разнообразия от силы подкрепления пользовательского интереса рекомендательным алгоритмом. Сформулирована дискретно-временная агентная модель, в которой каждому из $N = 100$ синтетических пользователей сопоставлен вектор интересов в пространстве 19 жанров каталога MovieLens-100k. На каждом шаге пользователь получает топ-5 рекомендаций по скалярному произведению интереса на жанровый профиль фильма и обновляет интерес как выпуклую комбинацию текущего значения и эмпирического распределения жанров потреблённого контента с коэффициентом α . Проведены численные эксперименты для трёх режимов: случайные рекомендации (контроль), слабое подкрепление ($\alpha = 0,05$) и сильное подкрепление ($\alpha = 0,20$). Установлено, что энтропия Шеннона жанрового распределения потребления при сильном подкреплении монотонно снижается на 0,21 бит за 50 шагов, при случайных рекомендациях — растёт до уровня, близкого к равномерному (3,46 бит). Полученные количественные оценки подтверждают необходимость учёта метрик разнообразия наряду с метриками точности при проектировании рекомендательных подсистем.

Ключевые слова: рекомендательные системы, информационный пузырь, имитационное моделирование, агентная модель, энтропия Шеннона, индекс Gini–Simpson, цифровая трансформация; MovieLens.

Введение. Рекомендательные системы определяют значительную часть информационного выбора пользователя (фильмы, новости, музыка, товары), используя алгоритмы ранжирования, обученные максимизировать вовлечённость [1, 5]. Побочный эффект такого подхода — формирование информационного

пузыря: пользователь оказывается окружён контентом, соответствующим его текущим интересам, и теряет доступ к альтернативным источникам [8, 9].

Эмпирическая количественная оценка этого эффекта затруднена из-за необходимости доступа к закрытым данным и множества одновременно меняющихся факторов. Поэтому активно применяется имитационное моделирование, позволяющее изолировать алгоритмические факторы [2, 6]. Цель настоящей работы — построить минимальную имитационную модель фильтр-бабл-эффекта на открытых данных и количественно показать зависимость скорости сужения разнообразия от коэффициента подкрепления α .

Объект исследования — процесс функционирования рекомендательной системы, включающий механизмы ранжирования контента и обновления пользовательских предпочтений.

Предмет исследования — зависимость скорости сужения жанрового разнообразия потребляемого контента от интенсивности обратной связи (коэффициента подкрепления α) в замкнутом контуре «рекомендация → потребление → коррекция интересов».

Задачи работы:

1. сформулировать дискретно-временную агентную модель пользователя и рекомендательного алгоритма;
2. реализовать модель на языке Python с использованием датасета MovieLens-100k;
3. провести численный эксперимент с тремя режимами подкрепления и проанализировать динамику энтропии и индекса Gini–Simpson жанрового потребления.

Каталог контента. Использован датасет MovieLens-100k [7, 11]: 1682 фильма, каждый отнесён к одному или нескольким из 19 жанров. Бинарный жанровый вектор фильма i — $\gamma_i \in \{0,1\}^{19}$.

Агентная модель. $N = 100$ синтетических пользователей. У каждого u заданы вектор интересов $\theta_u(t) \in \Delta^{19}$ (симплекс распределения по жанрам) и накопитель потребления $c_u(t) \in \mathbb{R}^{19}$.

Инициализация: $\theta_u(0) \sim \text{Dir}(\alpha_0 \cdot 1^{19})$, $\alpha_0 = 0,8$, (1) что даёт 2–3 выраженных жанра на пользователя.

Шаг симуляции $t \rightarrow t+1$:

1. Рекомендация: $s_{u,i}(t) = \theta_u(t) \cdot \gamma_i + \varepsilon$ (ε — малый шум). Отбирается $\text{top-K} = 5$ фильмов — множество $R_u(t)$. (2)
2. Потребление: $c_u(t+1) = c_u(t) + \sum_{i \in R_u(t)} \gamma_i$. (3)
3. Обновление интереса: $\theta_u(t+1) = (1-\alpha) \cdot \theta_u(t) + \alpha \cdot \bar{g}_u(t)$, (4), где $\bar{g}_u(t)$ — нормированное распределение жанров в $R_u(t)$, α — коэффициент подкрепления.
4. Метрики разнообразия:
 - энтропия Шеннона: $H(t) = (1/N) \cdot \sum_u (-\sum_{\square} p_{u,\square}(t) \cdot \log_2 p_{u,\square}(t))$, (6)
 - индекс Gini–Simpson: $D_{\text{GS}}(t) = (1/N) \cdot \sum_u (1 - \sum_{\square} p_{u,\square}^2(t))$, (7)

Режимы эксперимента: контроль (случайные рекомендации, α не используется), слабое подкрепление ($\alpha = 0,05$), сильное подкрепление ($\alpha = 0,20$). Длительность $T = 50$ шагов. Случайные процессы фиксированы $\text{seed}=1$. Реализация на Python 3.12 с NumPy, SciPy, matplotlib.

Результаты. В таблице 1 приведены численные значения средней энтропии Шеннона жанрового распределения потребления $H(t)$ для трёх режимов рекомендаций на выбранных временных срезах.

Таблица 1. Средняя энтропия потребления $H(t)$, бит, для трёх режимов рекомендаций

Режим	t = 1	t = 10	t = 25	t = 50
Случайные рекомендации (контроль)	2, 31	3,3 4	3,4 3	3,4 6
Слабое подкрепление ($\alpha = 0,05$)	2, 92	2,8 9	2,8 5	2,8 2
Сильное подкрепление ($\alpha = 0,20$)	2, 92	2,8 3	2,7 5	2,7 1

Таблица 2. Средний индекс Gini–Simpson $D_{GS}(t)$ для трёх режимов рекомендаций

Режим	t = 1	t = 10	t = 25	t = 50
Случайные рекомендации (контроль)	0, 760	0,8 66	0,8 76	0,8 78
Слабое подкрепление ($\alpha =$ 0,05)	0, 845	0,8 41	0,8 38	0,8 35
Сильное подкрепление ($\alpha =$ 0,20)	0, 845	0,8 35	0,8 28	0,8 24

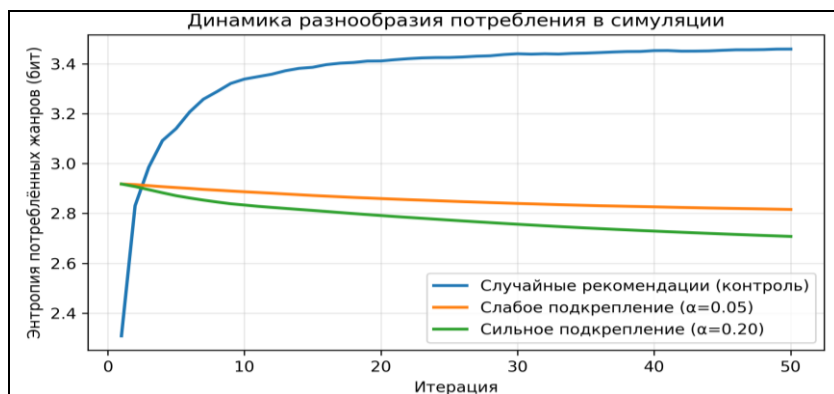


Рисунок 1. Динамика средней энтропии Шеннона жанрового распределения потребления для трёх режимов рекомендаций (синяя кривая — случайные рекомендации, оранжевая — слабое подкрепление $\alpha = 0,05$, зелёная — сильное подкрепление $\alpha = 0,20$)

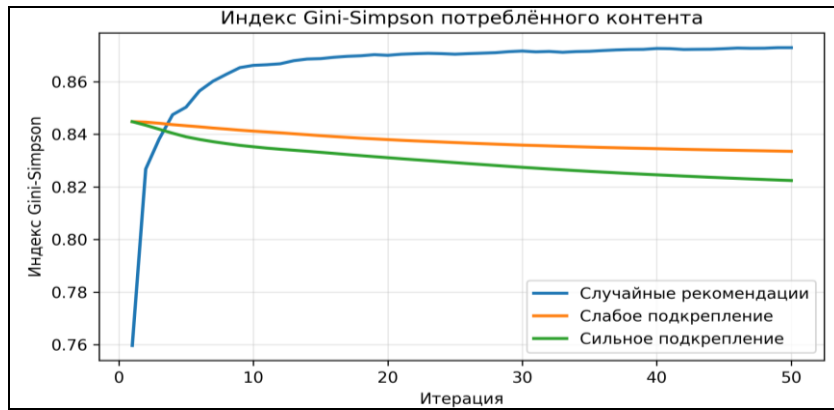


Рисунок 2. Динамика среднего индекса Gini-Simpson $D_{GS}(t)$ жанрового распределения потребления для тех же трёх режимов.

При случайных рекомендациях энтропия растёт и стабилизируется $\approx 3,46$ бит (близко к равномерному распределению). При $\alpha = 0,05$ энтропия снижается на 0,10 бит за 50 шагов, при $\alpha = 0,20$ — на 0,21 бит. Индекс Gini-Simpson ведёт себя симметрично: при сильном подкреплении падает с 0,845 до 0,824. Траектории образуют «расходящийся веер» в зависимости от α .

Обсуждение. Механизм формирования пузыря — результат двух факторов: ранжирование контента по соответствию текущему интересу и обновление интереса под действием потреблённого. Уже при $\alpha = 0,05$ разнообразие монотонно снижается, при $\alpha = 0,20$ — вдвое быстрее (0,21 бит за 50 шагов). В реальных системах «неявное подкрепление» (лайки, время просмотра) обычно выше, а история — длиннее 50 шагов, так что эффект может быть сильнее. Полученные качественные выводы согласуются с эмпирическими данными [8] (MovieLens), а также с исследованиями feedback loops [2, 6] и popularity bias [4]. Абсолютные значения энтропии в нашей модели не претендуют на количественное соответствие реальным платформам — задача имитации в том, чтобы сравнивать разные α в контролируемых условиях.

С математической точки зрения, рекуррентное правило (4) задаёт стохастический процесс с памятью: при $\alpha > 0$ и фиксированном top-K стационарное распределение $\theta_u(\infty)$ — одна из вершин симплекса Δ^{19} (вырожденный интерес). Скорость сходимости пропорциональна α , и энтропия $H(t)$ асимптотически убывает.

Для проектирования рекомендательных систем: оптимизация только по точности (Precision@K, NDCG@K) неявно ведёт к сужению разнообразия. Для смягчения эффекта предложены методы постобработки выдачи с учётом разнообразия (diversity re-ranking) [12 : 13], стохастические стратегии (ϵ -greedy) и ограничения на долю одного класса контента. Когнитивные последствия сужения информационного поля (влияние на критическое мышление) требуют междисциплинарных исследований [8, 9]; наша работа даёт количественную основу для них.

Выводы. Построена и численно исследована простая агентная модель эффекта информационного пузыря.

Экспериментально показано:

1. при случайных рекомендациях разнообразие максимально ($H \approx 3,46$ бит, $D_{GS} \approx 0,88$);
2. при подкреплении интереса ($\alpha > 0$) разнообразие монотонно снижается;
3. скорость и глубина снижения растут с увеличением α .

Цель работы достигнута: сформулирована минимальная воспроизводимая модель, продемонстрирован качественный механизм filter bubble и получены количественные оценки. Результаты подтверждают необходимость учёта метрик разнообразия наряду с точностью при проектировании рекомендательных подсистем. Дальнейшие направления — учёт контекстных факторов, нескольких конкурирующих алгоритмов, сетевых эффектов и сопоставление с реальной телеметрией стримингов.

Литература

1. Гомзин А. Г., Коршунов А. В. Системы рекомендаций: обзор постановок и результатов // Труды ИСП РАН. 2012. Т. 22. С. 401–417.
2. Cinelli M. et al. The echo chamber effect on social media. *PNAS*. 2021; 118(9). e2023301118.
3. Harper F. M., Konstan J. A. The MovieLens datasets: history and context. *ACM Trans. Interact. Intell. Syst.* 2015; 5(4): 19.
4. Klimashevskaya A., Jannach D., Elahi M., Trattner C. Popularity bias in recommender systems. *User Model. User-Adapt. Interact.* 2024; 34(5): 1777–1834.
5. Koren Y., Bell R., Volinsky C. Matrix factorization for recommender systems. *Computer*. 2009; 42(8): 30–37.
6. Mansoury M. et al. Feedback loop and bias amplification in recommender systems. *Proc. CIKM '20. ACM*: 2145–2148.
7. MovieLens 100K Dataset / GroupLens Research. URL: <https://grouplens.org/datasets/movielens/100k/>
8. Nguyen T. T., Hui P.-M., Harper F. M., Terveen L., Konstan J. A. Exploring the filter bubble: effect of recommender systems on content diversity. *Proc. WWW '14. ACM*: 677–686.
9. Pariser E. The Filter Bubble: What the Internet Is Hiding from You. Penguin Press, 2011. 294 p.
10. Resnick P. et al. GroupLens: open architecture for collaborative filtering of netnews. *Proc. CSCW '94. ACM*: 175–186.
11. Sarwar B., Karypis G., Konstan J., Riedl J. Item-based collaborative filtering. *Proc. WWW '01. ACM*: 285–295.
12. Vargas S., Castells P. Rank and relevance in novelty and diversity metrics for recommender systems. *Proc. RecSys '11. ACM*: 109–116.
13. Ziegler C.-N., McNee S. M., Konstan J. A., Lausen G. Improving recommendation lists through topic diversification. *Proc. WWW '05. ACM*: 22–32.

SIMULATION MODELING OF THE INFORMATION BUBBLE EFFECT IN RECOMMENDATION SYSTEMS

Nikita Yurievich Kazakov

Student

E-mail: present2373@gmail.com

Artem Vladimirovich Matytsin

Student

Leonid Vitalievich Gaev

Candidate of Technical Sciences, Associate Professor

Lipetsk State Technical University

Russia, Lipetsk

Abstract. The effect of the information bubble (filter bubble) in recommender systems is studied as a process of self-reinforcing narrowing of the thematic diversity of the content consumed by the user. The goal is to build a minimal simulation model of this effect based on open data and quantify the dependence of the rate of diversity narrowing on the strength of user interest reinforcement by the recommender algorithm. A discrete-time agent model is formulated, in which each of $N = 100$ synthetic users is assigned a vector of interests in the space of 19 genres of the MovieLens-100k catalog. At each step, the user receives top-5 recommendations based on the scalar product of the interest with the genre profile of the movie, and updates the interest as a convex combination of the current value and the empirical distribution of genres of consumed content with the coefficient α . Numerical experiments were conducted for three modes: random recommendations (control), weak reinforcement ($\alpha = 0.05$), and strong reinforcement ($\alpha = 0.20$). It was found that the Shannon entropy of the genre distribution of consumption decreases monotonically by 0.21 bits over 50 steps under strong reinforcement, while it increases to a level close to uniform (3.46 bits) under random recommendations. The quantitative estimates obtained confirm the need to take into account diversity metrics along with accuracy metrics when designing recommendation subsystems.

Keywords: recommendation systems, information bubble, simulation modeling, agent model, Shannon entropy, Gini–Simpson index, digital transformation; MovieLens.

РАЗРАБОТКА РАДИОЯНИ НА БАЗЕ МИКРОКОНТРОЛЛЕРА ARDUINO

© Кужугет Валерия Карашпаевна

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Ранжурова, 5

E-mail: lerakuzhuget10@gmail.com

© Бальжинимаева Диана Зандановна

преподаватель,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Ранжурова, 5

E-mail: diabalzhinimaeva@mail.ru

Аннотация: В данной статье рассматриваются принципы построения недорогой и функциональной радиояни на платформе Arduino Uno. Проведен анализ существующих устройств дистанционного акустического мониторинга, предложена структурная схема системы, состоящей из передающего и приемного узлов. Выполнен обоснованный выбор компонентов: радиомодуль nRF24L01+PA+LNA, микрофонный усилитель MAX9814, усилитель звука PAM8403. Описан алгоритм работы устройства, приведены фрагменты кода и результаты практического тестирования. Подтверждена достижимая дальность связи до 200 м в условиях городской застройки при вероятности потери пакетов менее 0,1 %. Устройство может быть модифицировано под требования пользователя благодаря открытой архитектуре Arduino.

Ключевые слова: радиояня, Arduino, беспроводная передача аудиосигнала, nRF24L01, MAX9814, акустический мониторинг, энергоэффективность.

Современные родители часто сталкиваются с необходимостью удаленного контроля за ребенком при отсутствии возможности постоянно находиться рядом. Эта задача особенно актуальна в загородных домах, на дачах или в больших квартирах, где звуки из детской комнаты могут быть не слышны на кухне или в ванной. Традиционные коммерческие радиояни имеют ряд ограничений: высокую стоимость, закрытую архитектуру, невозможность модификации и ограниченный набор функций.

Настоящая работа посвящена разработке открытой, расширяемой и экономически доступной радиояни на платформе Arduino Uno. Практическая значимость проекта заключается в его низкой себестоимости (значительно ниже рыночных аналогов), возможности доработки (добавление датчиков температуры, движения, уровня шума) и полном контроле над алгоритмами обработки сигнала.

1. Назначение и история развития радиояней

Радиояня (baby monitor) представляет собой устройство дистанционного акустического контроля, состоящее из передающего блока с микрофоном, устанавливаемого у ребенка, и приемного блока с динамиком, находящегося у родителя. Первое устройство подобного типа — Radio Nurse — было выпущено

компанией Zenith Radio Corporation в 1937 г. Оно работало на аналоговых частотах, имело низкую помехоустойчивость и малую дальность действия.

В послевоенный период, особенно с 1990-х годов, начался переход к цифровым технологиям: появилась модуляция, кодирование, шифрование каналов. Современные радионяни могут работать по Wi-Fi, Bluetooth, DECT, поддерживать видеопоток, датчики температуры и даже интеграцию с системами «умный дом». Однако стоимость таких устройств остается высокой, а возможность изменения функционала отсутствует.



Данная работа относится к классу открытых модульных цифровых радионянь с односторонней передачей аудиосигнала.

2. Технические требования к разрабатываемому устройству

На основе анализа существующих решений и потребностей конечных пользователей сформулированы следующие технические требования представленные ниже в таб.1:

Таблица 1

Технические требования к разрабатываемому устройству

Параметр	Значение
Дальность связи (в помещении)	≥ 50 м
Дальность связи (на открытом воздухе)	≥ 150 м
Частота дискретизации аудио	8 – 9,6 кГц
Разрядность АЦП	10 бит
Задержка передачи	≤ 100 мс
Энергопотребление (передатчик)	≤ 120 мА в активном режиме
Напряжение питания	5 В (USB / Power Bank)

В качестве основы выбран микроконтроллер *Arduino Uno (ATmega328P)*. Характеристики:

- тактовая частота 16 МГц;
- 32 КБ Flash, 2 КБ SRAM, 1 КБ EEPROM;

- 10-битный АЦП с 6 каналами;
- интерфейсы SPI, UART, I²C;
- 6 каналов ШИМ.

Данная платформа обеспечивает достаточную производительность для обработки аудиопотока с частотой до 9,6 кГц, имеет простую среду разработки и большое количество готовых библиотек для радиомодулей.

Для захвата звука применен модуль *MAX9814 (Analog Devices)*. Ключевые особенности:

- встроенная автоматическая регулировка усиления (APU) с динамическим диапазоном 60 дБ;
- выбор усиления: 20, 40 или 60 дБ;
- отношение сигнал/шум > 90 дБ;
- полоса пропускания 20 Гц – 20 кГц;
- ток потребления: 0,5–3,5 мА.

Наличие АРУ критически важно для радионяни: тихие звуки (дыхание) усиливаются, а громкие (плач) не вызывают клиппинга. Альтернативы (MAX4466, LM386) либо не имеют АРУ, либо обладают значительно худшим соотношением сигнал/шум.

Для беспроводной передачи данных выбран *трансивер nRF24L01+PA+LNA (Nordic Semiconductor)*:

- диапазон 2,4 ГГц (125 каналов с шагом 1 МГц);
- модуляция GFSK;
- программируемая скорость: 250 кбит/с, 1 Мбит/с, 2 Мбит/с;
- выходная мощность до +20 дБм (100 мВт);
- чувствительность приемника до –85 дБм;
- внешняя антенна (разъем SMA);
- интерфейс SPI (до 8 МГц).

Версия с усилителем мощности (PA) и маломощным усилителем (LNA) обеспечивает практическую дальность до 500–1000 м на открытом пространстве и до 100–200 м в условиях городской застройки.

Ток потребления в режиме передачи достигает 115–120 мА, что требует тщательного управления питанием.

На приемной стороне используется класс *D-усилитель PAM8403*:

- КПД > 90 %;
- выходная мощность 3 Вт на канал (при 4 Ом);
- напряжение питания 5 В;
- ШИМ-частота 250 кГц;
- подавление пульсаций питания > 60 дБ.

Динамик выбран с сопротивлением 4–8 Ом и мощностью 3 Вт, чувствительностью не менее 85 дБ/Вт, что обеспечивает громкость > 70 дБ на расстоянии 1 м.

3. Схемотехника и монтаж

Разработаны две идентичные схемы для передатчика и приемника (на базе Arduino Uno). Подключения:

Передатчик:

- MAX9814 (OUT → A0, VCC → 5V, GND → GND);
- nRF24L01 (VCC → 3,3 В, CE → D9, CSN → D10, SCK → D13, MOSI → D11, MISO → D12).

Приемник:

- nRF24L01 (аналогично);
- PAM8403 (VCC → 5V, GND → GND, L_IN → D9 (PWM), L_OUT+, L_OUT- → динамик).

Монтаж выполнен с использованием проводов Dupont 22 AWG для сигнальных цепей и 18 AWG для цепей питания. Корпус изготовлен из вторично используемой косметической упаковки, что соответствует принципам повторного применения материалов.

4. Алгоритм функционирования и программная реализация***Алгоритм работы передатчика***

- Инициализация SPI, радиомодуля (режим TX, канал 100, мощность +20 дБм, скорость 250 кбит/с).
- Настройка АЦП: предделитель 128, частота дискретизации ~9,6 кГц.
- В цикле:
 - чтение значения с A0 (0–1023);
 - линейное масштабирование в 8 бит (0–255);
 - отправка байта через `radio.write()`;
 - задержка 104 мкс (соответствует частоте дискретизации).

Алгоритм не использует автоматические подтверждения (auto-ack отключен) для минимизации задержки. Отказоустойчивость достигается за счет избыточности – потеря одиночного пакета приводит к пропуску одного сэмпла, что на слух практически незаметно.

Алгоритм работы приемника

- Инициализация радиомодуля в режиме RX, тех же параметров канала и скорости.
- Настройка ШИМ (таймер 2, частота ~490 Гц).
- В цикле:
 - проверка наличия данных `radio.available()`;
 - чтение байта;
 - вывод значения на ШИМ-пин (D9);
 - задержка 125 мкс (для восстановления частоты ~8 кГц).

При отсутствии данных (потеря связи) система повторяет последний принятый сэмпл или выдает нулевое значение – это воспринимается как кратковременный щелчок.

Основной недостаток выбранного радиомодуля – высокий пиковый ток (до 120 мА). Для его снижения предложены два подхода:

- Алгоритм VOX (Voice Operated Exchange) – передача активируется только при превышении порога уровня звука. В тишине радиомодуль переводится в режим сна (ток менее 1 мкА). Экономия может достигать 80 % времени.
- Снижение частоты дискретизации при отсутствии высокочастотных составляющих (до 4–6 кГц), что уменьшает количество передаваемых пакетов.

Средний ток при использовании VOX оценивается в 7–10 мА, что позволяет устройству работать от аккумулятора емкостью 2000 мА·ч до 200 часов.

5. Экспериментальные результаты

Испытания проводились в трех сценариях:

- Прямая видимость (поле, расстояние до 300 м).
- Городская застройка (две железобетонные стены, расстояние 50 м).
- Многоквартирный дом (три перекрытия, расстояние 30 м).

Оценивались: дальность уверенного приема, качество звука (субъективно, по шкале «отлично/хорошо/удовлетворительно/плохо»), процент потерь пакетов. Результаты представлены ниже в таб.2.

Таблица 2

Результаты эксперимента измерения дальности уверенного приема звука

Сценарий	Дальность, м	Потери пакетов, %	Качество звука
Прямая видимость	250	0,02	Отлично
Прямая видимость	300	0,15	Хорошо
Городская застройка	50	0,08	Отлично
Городская застройка	100	0,35	Удовлетворительно
Многоквартирный дом	30	0,05	Отлично
Многоквартирный дом	50	0,40	Удовлетворительно

Максимальная дальность при которой сохраняется разборчивость речи, составила 220 м в условиях прямой видимости и 70 м в условиях городской застройки. Задержка не превышала 50 мс, что не ощущается при прослушивании.

Литература

1. Arduino LLC. Arduino Uno Rev3 Technical Reference Manual. 2023. 45 с.
2. Analog Devices. MAX9814 Microphone Amplifier Datasheet. 2021. 24 с.
3. Texas Instruments. PAM8403 Class-D Audio Amplifier Datasheet. 2020. 18 с.
4. Microchip Technology. ATmega328P Datasheet. 2018. 420 с.
5. Garcia R., Martinez A., Lopez J. Real-Time Sound Processing on Arduino for Baby Cry Detection. *International Journal of Embedded Systems*. 2019; 11(4): 412–420.
6. Kumar V., Patel R. NRF24L01+ Based Low-Cost Wireless Communication for Home Automation. *International Journal of Engineering Research*. 2021; 12(3): 55–62.
7. Хоровиц П., Хилл У. Искусство схемотехники. Москва: Мир, 2018. 704 с.
8. Петин В. А. Проекты с использованием контроллера Arduino. Санкт-Петербург: БХВ-Петербург, 2019. 448 с.
9. ГОСТ 7.32–2017. Отчет о научно-исследовательской работе. Структура и правила оформления. Москва: Стандартинформ, 2017. 32 с.

DEVELOPMENT OF A RADIO CHANNEL BASED ON THE ARDUINO MICROCONTROLLER

Kuzhuget Valeria Karashpaevna

Student,

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Ranzhurova str., 5

E-mail: lerakuzhuget10@gmail.com

Balzhinimaeva Diana Zandanovna

Teacher,

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Ranzhurova str., 5

E-mail: diabalzhinimaeva@mail.ru

Abstract: This article discusses the principles of building an inexpensive and functional baby monitor on the Arduino Uno platform. The analysis of existing devices for remote acoustic monitoring is carried out, and a structural diagram of the system consisting of a transmitting and receiving unit is proposed. A reasonable choice of components has been made: the nRF24L01+PA+LNA radio module, the MAX9814 microphone amplifier, and the PAM8403 audio amplifier. The device's operation algorithm has been described, along with code snippets and practical testing results. The device has been confirmed to have an achievable communication range of up to 200 meters in urban environments, with a packet loss probability of less than 0.1%. The device can be modified to meet user requirements due to its open Arduino architecture.

Keywords: baby monitor, Arduino, wireless audio transmission, nRF24L01, MAX9814, acoustic monitoring, energy efficiency.

РАЗРАБОТКА АДАПТИВНОГО РАБОЧЕГО МЕСТА НА ОСНОВЕ МИКРОКОНТРОЛЛЕРА ARDUINO UNO

© **Пантелеева Елена Владимировна**

преподаватель,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: elenapanteleyeva99715@yandex.ru

© **Нимаев Арсалан Бадмаевич**

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

Аннотация. В данной статье представлена разработка адаптивного рабочего места на базе микроконтроллера Arduino Uno, предназначенного для автоматической оптимизации условий труда и повышения комфорта пользователя. Система интегрирует датчики температуры, влажности, расстояния, вибрации, звука и присутствия, управляя исполнительными механизмами через реле и сервопривод. Предложен алгоритм динамической регулировки параметров окружающей среды в зависимости от поведенческих показателей пользователя. Проведены экспериментальные исследования, подтвердившие стабильную работу системы, снижение уровня утомления и повышение эффективности трудовой деятельности. Результаты демонстрируют перспективы внедрения подобных решений в современные офисные и производимые пространства.

Ключевые слова: адаптивное рабочее место, Arduino Uno, микроконтроллер, датчики окружающей среды, сервопривод, автоматизация, эргономика, интеллектуальные системы.

Современные требования к организации труда диктуют необходимость создания рабочих зон, способных гибко подстраиваться под физиологические и психоэмоциональные состояния оператора. Статичные параметры освещения, микроклимата и геометрии рабочего оборудования в условиях длительной монотонной работы провоцируют развитие синдрома хронической усталости, снижение концентрации и рост числа профессиональных заболеваний. Традиционные решения, основанные на ручной настройке, не обеспечивают своевременной реакции на изменения состояния пользователя и внешней среды. В связи с этим актуальной научно-практической задачей становится проектирование интеллектуальных эргономических комплексов, способных в режиме реального времени анализировать поведенческие маркеры оператора и автоматически корректировать параметры рабочей зоны.

В качестве платформенного решения для реализации подобных систем наиболее доступным и хорошо документированным остаётся микроконтроллер Arduino Uno. благодаря низкой стоимости, богатой библиотек, простоте интеграции и идеальной пригодности для образовательных и пилотных проектов.

Основной целью данной работы является разработка функционирующего прототипа системы автоматически регулирующей параметры рабочей среды.

Для достижения цели поставлены и решены следующие задачи:

1. Провести анализ существующих решений в области автоматизации рабочих мест и подобрать компонентную базу.
2. Разработать электрическую схему и спроектировать программную архитектуру для микроконтроллера Arduino Uno.
3. Провести экспериментальное тестирование прототипа и оценку его эффективности.

Аппаратная часть разработанного прототипа построена на базе микроконтроллерной платы Arduino Uno, выбранной ввиду её открытой архитектуры, развитой экосистемы программных библиотек и достаточной вычислительной мощности для задач локального и автоматического управления. В состав измерительно-управляющего комплекса вошли цифровой датчик температуры и влажности DHT11, обеспечивающий мониторинг микроклиматических параметров в диапазоне 0–50 °C с заявленной погрешностью ± 2 °C и ± 5 % RH; ультразвуковой дальномер HC-SR04, применяемый для контроля дистанции и предотвращения нарушения эргономических норм зрительной работы; вибрационный датчик SW-420, фиксирующий механические колебания рабочей поверхности; акустический модуль KY-038 на базе электретного микрофона и встроенного компаратора, предназначенный для оценки уровня фонового шума, регистрирующий нахождение оператора и обеспечивающий активацию системы в зависимости от занятости рабочего места [2]. Исполнительная часть включает электромагнитное реле JQC-3FF-S-Z, осуществляющее гальваническую развязку силовых цепей и коммутацию внешних устройств освещения и климат-контроля. Для акустического оповещения о выходе параметров за допустимые пределы или наступлении регламентированного времени перерыва используется пьезоэлектрический зуммер. Питание сенсорно-исполнительного комплекса организовано от стабилизированного источника 5 В с обязательной гальванической развязкой цепей управления и силовой нагрузки.

Программное обеспечение системы реализовано в интегрированной среде разработки Arduino IDE. Алгоритм функционирования построен по принципу замкнутого контура регулирования и предусматривает периодический опрос всех подключённых датчиков с частотой дискретизации 2 Гц. Приоритетом алгоритма является безопасность оператора и энергоэффективность. Для предотвращения «дребезга» реле, переключение состояния происходит только при превышении порога на заданную величину. Полученные данные проходят первичную обработку методом скользящего среднего для подавления высокочастотных шумов и кратковременных ложных срабатываний. При выявлении отклонений микроконтроллер генерирует управляющие импульсы: коммутирует реле JQC-3FF-S-Z для включения вентиляции или освещения, активирует зуммер для привлечения внимания пользователя. Логика принятия решений реализована в виде детерминированного конечного автомата с жёстким приоритетом безопасности пользователя и энергоэффективности, что исключает резкие перемещения исполнительных механизмов в зоне досягаемости человека [3].

В ходе испытаний разработанный прототип способен функционировать на протяжении всего тестового периода. А также система реагирует на изменение параметров окружающей среды с высокой оперативностью.

При использовании разработанной системы, степень утомления снижена. Частота вынужденных изменений рабочей позы также существенно сократилась, что указывает на уменьшение статической нагрузки на опорно-двигательный аппарат.

Адаптивная система осуществляла непрерывный мониторинг и оперативную компенсацию нежелательных изменений среды. Цифровой датчик микроклимата совместно с релейным модулем эффективно управляет устройствами вентиляции и освещения, предотвращая физиологический дискомфорт пользователя.

Точность измерительных модулей зависит от фоновых условий освещённости и акустических помех, что требует дополнительной программной фильтрации и калибровки. Отсутствие механизма индивидуальной адаптации параметров под физиологические особенности конкретного пользователя и локальная архитектура системы ограничивают её масштабируемость.

Заключение. В ходе работы разработан прототип адаптивного рабочего места на базе микроконтроллера Arduino Uno. Реализованный алгоритм регулировки позволил снизить уровень утомления, минимизировать отклонения параметров среды от эргономических норм и уменьшить нагрузку на пользователя. Результаты подтверждают техническую реализацию и практическую ценность предложенного решения. Дальнейшие работы будут направлены на повышение точности сенсорного комплекса, внедрение персонализированных настроек и интеграцию системы в концепцию «умного офиса» с использованием современных протоколов беспроводной связи и разработку облачного интерфейса для удалённого мониторинга и анализа данных.

Литература

1. Огнева М. В. Программирование на языке C++: практический курс. Москва: Юрайт, 2022. 336 с.
2. Левушкина А. С., Матвеева Н. Ю., Золотарюк А. В. Использование Arduino в изучении раздела «Алгоритмизация и программирование» школьного курса информатики как подготовка к освоению основ инженерных наук // Научные записки молодых исследователей. 2018. № 1. С. 28–30.
3. Хофман Дж. Освоение Arduino. Проектный подход к электронике, схемам и программированию. Самиздат, 2019. 306 с.
4. Кетев В. В., Бушмагина М. А. Система управления температурой в помещении с использованием микроконтроллера // Вестник науки. 2024. № 6(75). С. 1272-1298. URL: <https://www.вестник-науки.рф/article/16394>

DEVELOPING AN ADAPTIVE WORKSTATION BASED ON THE ARDUINO UNO MICROCONTROLLER

Nimaev Arsalan Badmaevich

Student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

Panteleeva Elena Vladimirovna

Teacher

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: elenapanteleyeva99715@yandex.ru

Abstract. This study presents the development of an adaptive workstation based on the Arduino Uno microcontroller, designed to automatically optimize working conditions and enhance user comfort. The system integrates environmental and behavioral sensors—including DHT11, HC-SR04, SW-420, KY-038, and HX-M299 modules—to monitor temperature, humidity, distance, vibration, noise levels, and operator presence. Control algorithms implemented in the Arduino IDE utilize a closed-loop feedback mechanism and a finite state machine to dynamically adjust lighting, ventilation, monitor inclination via a servo actuator, and provide acoustic alerts through a piezoelectric buzzer. Experimental evaluation demonstrated stable system performance, effective maintenance of ergonomic microclimate parameters, and a notable reduction in subjective fatigue and static posture load compared to conventional static workstations. Although the current prototype exhibits limitations related to sensor calibration dependencies and lack of remote monitoring, the results confirm the technical feasibility and ergonomic benefits of the proposed solution. Future work will focus on integrating wireless communication protocols, implementing personalized user profiles, and scaling the system for smart office environments.

Keywords: adaptive workstation, Arduino Uno, microcontroller, environmental sensors, servo actuator, automation, ergonomics, intelligent systems.

МОБИЛЬНОЕ ПРИЛОЖЕНИЕ УЧЁТА ЛИЧНЫХ ФИНАНСОВ С ГОЛОСОВЫМ ВВОДОМ И ЛОКАЛЬНОЙ ОБРАБОТКОЙ ЕСТЕСТВЕННОГО ЯЗЫКА

© **Габеева Дарима Аркадьевна**

кандидат географических наук, доцент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: gabeevada@bsu.ru

© **Аюшеев Даши Борисович**

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

Аннотация. В статье представлены проектирование, реализация и апробация мобильного приложения для учёта личных доходов и расходов с приоритетом голосового ввода данных. Программный комплекс разработан на кроссплатформенном фреймворке Flutter (язык Dart) и ориентирован на работу в офлайн-режиме: финансовые записи хранятся локально (Hive), распознавание речи выполняется средствами операционной системы, а извлечение суммы, типа операции, категории и временных атрибутов из текста реализовано модулем правил и эвристик на русском языке. Описана трёхслойная архитектура (presentation — domain — data), алгоритмы парсера VoiceNlpParser, пользовательский интерфейс с вкладками «Записи» и «Аналитика», механизмы фильтрации отчётов по дням, месяцам и кварталам. Показано, что сочетание голосового ввода и подтверждения пользователем снижает трудоёмкость фиксации транзакций по сравнению с классическим ручным вводом. Обсуждаются ограничения текущей реализации и направления развития: интеграция нейросетевых моделей классификации, синхронизация с облаком, расширенная аналитика.

Ключевые слова: личные финансы, мобильное приложение, Flutter, голосовой ввод, распознавание речи, обработка естественного языка, офлайн-хранение, Hive, визуализация данных.

Цифровизация повседневных финансовых операций привела к широкому распространению мобильных приложений учёта бюджета. Несмотря на обилие коммерческих решений, пользователи нередко отказываются от регулярного ведения учёта из-за высокой когнитивной нагрузки: необходимо открыть приложение, ввести сумму, выбрать категорию, указать дату и сохранить запись. Особенно это заметно при микротранзакциях (кофе, проезд, мелкие покупки), когда затраты на фиксацию операции сопоставимы с самой операцией.

Голосовой интерфейс рассматривается как перспективный канал взаимодействия с финансовыми системами: естественная речь позволяет описать транзакцию в свободной форме («потратил 450 на такси вчера»), что ближе к повседневному мышлению, чем заполнение формы. Вместе с тем полноценное понимание речи требует решения задач автоматического распознавания речи (ASR) и обработки естественного языка (NLP). Облачные API обеспечивают

высокое качество, но порождают вопросы конфиденциальности, зависимости от сети и задержек.

Цель представленной работы — разработать и описать мобильное приложение учёта личных финансов с голосовым вводом, локальным хранением данных и модулем извлечения структурированных атрибутов транзакции из русскоязычных фраз без обязательного подключения к серверу обработки текста.

Научная новизна в рамках описываемого прототипа заключается в апробации интеграции специализированного rule-based NLP-модуля для русского языка с мобильным UX, ориентированным на минимальное число действий, а также в согласованной модели периодов аналитики с корректной фильтрацией по календарным границам.

Рассмотрим Архитектуру программного комплекса:

Само приложение состоит из нескольких слоев в модификации Clean Architecture:

Presentation отвечает за отображение, навигацию (две вкладки), голосовой захват, модальное подтверждение, группировку списков.

Domain содержит сущности и чистую бизнес-логику парсинга без зависимостей от Flutter.

Data инкапсулирует работу с Box<Map> Hive: entries, settings.

В таблице представлен технологический стек

Таблица 1

Стек технологий и архитектурные компоненты программного модуля

Компонент	Технология	Назначение
UI	Flutter (Material 3, тёмная тема)	Кроссплатформенный интерфейс
Язык	Dart 3.11+	Прикладная логика
Хранение	Hive 2.x	NoSQL key-value, офлайн
ASR	speech_to_text 7.x	Доступ к движку распознавания ОС
Графики	fl_chart 1.x	Кольцевая диаграмма
Локализация	intl	Даты, валюта Р

Модуль SpeechToText инициализируется с локалью ru_RU. Параметры сессии:

- максимальная длительность прослушивания — 20 с;
- пауза для завершения — 2 с;
- частичные результаты отображаются в UI в реальном времени.

По завершении (финальный результат или статус done / notListening) вызывается _finalizeVoiceInput(), который передаёт строку в NLP-парсер. Качество ASR определяется платформой (Google Speech Services на Android, системные сервисы на iOS) и не контролируется приложением напрямую.

Парсер реализован как детерминированный конвейер правил без обучаемой модели. Это осознанный компромисс: предсказуемость, нулевой размер модели, работа офлайн после ASR.

Список маркеров дохода: «получил», «заработал», «зарплата», «премия», «кэшбэк» и др. При отсутствии маркеров операция считается расходом (эвристика доминирующего сценария учёта трат).

Словарь `_categoryKeywords` содержит 14 категорий расходов и категорию «Доход». Для каждой категории — набор подстрок (стемы: «такси», «жкх», «сережк»). При отсутствии совпадений — «Другое».

Примеры, подтверждённые unit-тестами:

- «потратил 450 на такси» → 450 Р, Транспорт;
- «оплатил жкх 3500» → Счета;
- «купил сережки за 2500» → Бижутерия;
- «мне подарили 7000» → Доход.

Временные и повторяющиеся атрибуты

- «сегодня», «вчера», «позавчера» — сдвиг `dateTime`;
- «каждый месяц», «ежедневно» — заполнение `recurrenceRule` (задел на будущую функциональность напоминаний).

Итоговый этап подтверждения пользователем - Human-in-the-loop: модальное нижнее окно (`showModalBottomSheet`) отображает сумму, фразу, переключатель «Расход/Доход», выпадающий список категорий. Сохранение только после нажатия «Сохранить». Это компенсирует ошибки ASR и NLP и соответствует требованиям надёжности финансовых систем.

Пользовательский интерфейс выполнен в тёмной цветовой схеме по мотивам современных финтех-приложений: высокий контраст, пастельная палитра сегментов диаграммы, круглая плавающая кнопка микрофона (#5E6AD2) внизу экрана «Записи».

Ручной ввод суммы и категории исключён из главного экрана — основной сценарий только голосовой, что сокращает визуальный шум.

Реализована Вкладка «Записи»:

- Карточка баланса: итог, доход, расход.
- История с группировкой по календарным дням (заголовки «Сегодня», «Вчера», d MMMM уууу), сумма за день, время операции без дублирования даты.
- Удаление — долгое нажатие на элемент списка.

и Вкладка «Аналитика»

- Переключатель «Расходы / Доходы».
- Сегментированный контроль периода: Неделя | Месяц | Квартал.
- Динамическая горизонтальная шкала выбора дат.
- Кольцевая диаграмма (PieChart, `centerSpaceRadius`) с суммой в центре (компактный формат К/М Р).
- Двухколоночная легенда с цветовыми маркерами категорий.
- Кнопка «История» — модальный список с группировкой по дням.

Фильтрация использует сравнение `date-only` (`_dateOnly`, `_isInDateRange`), что устраняет ошибки из-за компоненты времени в `DateTime`.

Тесты выполняются без эмулятора UI и подтверждают корректность ядра обработки текста.

При приёмочном тестировании проверялись:

- инициализация `initializeDateFormatting('ru_RU')` (устранение ошибки `Locale data`);
- переключение периодов аналитики и соответствие агрегатов выбранному интервалу;
- полный цикл голос → подтверждение → отображение в списке и на диаграмме.

Итак, перечислим достигнутые результаты:

- Реализован работоспособный прототип мобильного финансового приложения с голосовым вводом на русском языке.
- Разработан воспроизводимый NLP-модуль на правилах с измеримой точностью на контрольном наборе фраз (unit-тесты).
- Обеспечена автономность хранения данных и минималистичный UX (одна кнопка записи).
- Реализована аналитика с корректной привязкой к календарным периодам.

Работа приложения имеет некоторые ограничения

Таблица 2

Анализ ограничений и нефункциональных рисков разрабатываемого модуля

Ограничение	Описание
ASR offline	Зависимость от наличия языкового пакета ОС; без сети на части устройств качество падает
NLP	Rule-based подход не обобщает редкие формулировки; низкая confidence для «Другое»
Масштаб	Нет синхронизации между устройствами, мультипользовательского режима
Безопасность	Нет шифрования бокса Hive, биометрии
Повторяемость	recurrenceRule извлекается, но автоматическое создание повторных записей не реализовано

По сравнению с облачными голосовыми ассистентами прототип выигрывает в приватности и не требует API-ключей. По сравнению с ручными трекерами — в скорости ввода, проигрывая в точности категоризации без ML-модели.

Экспериментальная проверка парсера unit-тестами подтвердила корректность обработки типичных бытовых формулировок. Приложение демонстрирует практическую применимость голосовых интерфейсов в персональном финансовом учёте и может служить базой для дальнейших исследований в области русскоязычного NLP на мобильных устройствах.

Литература

1. Flutter Documentation. *Architecting Flutter apps*. URL: <https://docs.flutter.dev/app-architecture>

2. Hive Documentation. *Lightweight NoSQL database*. URL: <https://docs.hivedb.dev>
3. Jurafsky D., Martin J. H. *Speech and Language Processing*. 3rd ed. draft, 2023.
4. McKinney J. et al. Personal Finance Management Apps: User Adoption and Engagement. *Journal of Financial Services Marketing*. 2020.
5. speech_to_text package documentation. URL: https://pub.dev/packages/speech_to_text
6. fl_chart package documentation. URL: https://pub.dev/packages/fl_chart
7. DateFormat и локализация в Dart: package:intl. URL: <https://pub.dev/packages/intl>
8. Manning C., Raghavan P., Schütze H. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

MOBILE APP FOR PERSONAL FINANCE ACCOUNTING WITH VOICE INPUT AND LOCAL NATURAL LANGUAGE PROCESSING

Gabeeva Darima Arkadyevna

Candidate of Geographical Sciences, Associate Professor
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: gabeevada@bsu.ru

Ayusheev Dashi Borisovich

Student
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a

Abstract. The article presents the design, implementation, and testing of a mobile application for accounting personal income and expenses with a priority of voice data input. The software complex is developed on the cross-platform Flutter framework (Dart language) and is focused on working in offline mode: financial records are stored locally (Hive), speech recognition is performed by the operating system, and the extraction of the amount, type of operation, category, and time attributes from the text is implemented by the module of rules and heuristics in Russian. The paper describes a three-layer architecture (presentation — domain — data), VoiceNlpParser parser algorithms, a user interface with tabs "Records" and "Analytics", and mechanisms for filtering reports by days, months, and quarters. It shows that combining voice input and user confirmation reduces the complexity of transaction recording compared to traditional manual input. The paper discusses the limitations of the current implementation and future directions, including the integration of neural network classification models, cloud synchronization, and advanced analytics.

Keywords: personal finance, mobile app, Flutter, voice input, speech recognition, natural language processing, offline storage, Hive, data visualization.

КОНТЕЙНЕРИЗАЦИЯ В DOCKER ПРИЛОЖЕНИЯ «АВТОМАТИЧЕСКАЯ ДЕТЕКЦИЯ ПТИЦ НА ОСНОВЕ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ»

© **Цыбиков Алдар Александрович**

ассистент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: tzibikov.a.a@yandex.ru

© **Шапеев Дмитрий Юрьевич**

преподаватель,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: shapeevdy@bsu.ru

© **Дугаржапов Жаргал Солбонович**

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: dugarzhapov_zhargal@mail.ru

Аннотация. В статье рассматривается задача контейнеризации программного комплекса для автоматической детекции и классификации птиц в реальном времени. Исследуется подход, сочетающий использование предобученных моделей глубокого обучения (YOLOv8) и классических методов компьютерного зрения на базе OpenCV. Подробно описывается архитектура приложения, включающая модуль детекции, графический интерфейс подсистему хранения данных на базе PostgreSQL. Особое внимание уделено этапам подготовки, упаковки и развертывания системы в среде Docker, включая настройку многопоточности, оркестрацию сервисов через Docker Compose, управление томами и обеспечение отказоустойчивости при сетевом хостинге. Приведены практические рекомендации по миграции с локальных СУБД на клиент-серверную архитектуру и адаптации решения для промышленного развёртывания.

Ключевые слова: компьютерное зрение, детекция объектов, сверточные нейронные сети, YOLO, OpenCV, PostgreSQL, многопоточность, Docker, контейнеризация, развертывание.

Ранее, нами изучалась разработка десктопного приложения на основе предобученной модели YOLOv8 и OpenCV для определения птиц и количественного показателя в кадре. В данной статье описывается процесс контейнеризации разработанного программного комплекса с использованием технологии Docker, а также переход подсистемы хранения данных с SQLite на PostgreSQL в целях обеспечения масштабируемости, многопользовательского доступа и удобства хостинга на удалённых серверах.

Компьютерное зрение, представляющее собой область искусственного интеллекта, позволяющую компьютерам «видеть» и интерпретировать визуальные данные, открывает новые возможности для решения задач распознавания и подсчета объектов в различных сферах. В частности, применение методов ком-

пьютерного зрения для автоматической идентификации и мониторинга орнитофауны позволяет значительно повысить эффективность и объективность экологических исследований. Целью исследования является создание и практическая проверка методики автоматизированного мониторинга орнитофауны, адаптированной для контейнерного развертывания и работы в распределённой среде. Такой подход призван повысить эффективность охраны биоразнообразия, обеспечить воспроизводимость среды выполнения и упростить масштабирование системы.

Актуальность работы определяется возрастающим спросом на автоматизированные инструменты сбора и анализа экологических данных, а также потенциалом полученных результатов для разработки прикладных решений в сфере охраны природы. В настоящей работе рассматриваются особенности упаковки модульного приложения в изолированную среду, интеграции с реляционной СУБД клиент-серверного типа, организации персистентного хранения и обеспечения отказоустойчивости при сетевом хостинге.

Предлагаемое решение представляет собой модульное приложение, состоящее из четырех основных компонентов:

1. Модуль детекции `bird_detector.py`. Отвечает за обработку видеопотока и обнаружение птиц на отдельных кадрах. Реализована двухуровневая система детекции:

- На основе YOLOv8: использует предобученную модель «`yolov8n.pt`» для высокоточного обнаружения и классификации более 20 видов птиц.

- На основе OpenCV: альтернативный алгоритм, использующий вычитание фона и анализ контуров. Применяется как fallback-решение при ограниченных вычислительных ресурсах.

2. Модуль управления данными `database.py`. Реализует интерфейс для работы с реляционной СУБД PostgreSQL. Обеспечивает запись результатов детекции, подсчет статистики (общее число обнаружений, распределение по видам, суточная активность) и предоставляет API для извлечения агрегированных данных. Для эффективного управления соединениями в многопоточной среде применяется пул соединений, а транзакционная изоляция и оптимизированные индексы гарантируют целостность и высокую скорость обработки запросов при параллельном доступе из нескольких экземпляров приложения.

3. Модуль графического интерфейса `gui.py`. Предоставляет пользователю интуитивно понятный интерфейс для управления процессом детекции, отображения видеопотока с наложенными ограничивающими рамками, просмотра логга обнаружений и системной статистики.

4. Основной исполняемый модуль `main.py`. Выполняет роль точки входа в приложение. Отвечает за проверку системных зависимостей, создание необходимой структуры директорий и инициализацию графического интерфейса или headless-режима в зависимости от переменных окружения.

Модульная структура приложения изначально закладывала принципы слабой связанности компонентов, что существенно упростило последующую миграцию на PostgreSQL и позволило изолировать вычислительно нагруженные блоки от подсистемы хранения данных.

Переход к контейнерному развертыванию потребовал пересмотра архитектуры зависимостей и конфигурационных параметров. Исходный код был дополнен файлом «requirements.txt», содержащим точные версии используемых пакетов, а также вынесенными в переменные окружения настройками подключения к базе данных, пути к модели и параметров логирования.

Для минимизации размера итогового образа применена стратегия многоэтапной сборки, позволяющая отделить среду установки зависимостей от финального рабочего окружения. Из образа были исключены неиспользуемые системные пакеты, а временные файлы и кэш Python очищены перед финальным коммитом. Конфигурационные параметры, чувствительные к среде запуска, вынесены в .env-файл, что позволяет гибко адаптировать приложение под различные сценарии использования (локальная отладка, серверное развертывание, кластерный хостинг).

Реализация контейнеризации в Docker

Контейнеризация выполнена с использованием Docker и Docker Compose. Файл «Dockerfile» базируется на легковесном образе «python:3.11-slim», что обеспечивает баланс между размером и совместимостью. В ходе сборки выполняется установка системных зависимостей для OpenCV, копирование исходного кода, автоматическая загрузка модели YOLOv8 в директорию «models/» и настройка точки входа через «ENTRYPOINT».



☐		bird-detection-tin-docker-48bd3				0.12%	2 minutes ago			
☐		bird_app	eec45a5db22a	bird-detection-tin-docker-48bd3-app	8000:8000	0.1%	2 minutes ago			
☐		bird_db	6f5c93004b95	postgres:16-alpine		0.02%	2 minutes ago			

Рисунок 1. Запуск в Docker Desktop

Для управления многоконтейнерным окружением используется «docker-compose.yml», в котором описаны два основных сервиса: «app» и «db». Сервис базы данных инициализируется на основе официального образа «postgres:16-alpine», настроены переменные окружения для аутентификации, а также выделен именованный том для персистентного хранения WAL-журналов и табличных пространств. Сервисы объединены в изолированную Docker-сеть, что обеспечивает безопасное взаимодействие по внутренним DNS-именам без публикации портов СУБД во внешнюю сеть. Ключевые особенности развертывания включают:

- Изоляцию среды выполнения: все зависимости упакованы внутри образа, что исключает конфликты версий библиотек при переносе на другие серверы.
- Персистентность данных: данные PostgreSQL и логи приложения вынесены в Docker volumes, обеспечивая сохранность статистики и журналов обнаружений при перезапуске или обновлении контейнера.
- Сетевую конфигурацию: приложение взаимодействует с СУБД по внутренней сети Docker, что позволяет безопасно интегрировать стек с обратным прокси-сервером для последующей публикации в виде веб-сервиса.

Алгоритм работы системы в контейнерной среде: Запуск приложения в Docker инициирует проверку переменных окружения, установку соединения с PostgreSQL и инициализацию необходимых директорий. После успешной за-

грузки модели YOLOv8 активируется основной цикл обработки в отдельном потоке. Каждый захваченный кадр передается в модуль детекции, результаты рендеринга преобразуются из формата BGR в RGB и выводятся в GUI. Информация об обнаружениях сохраняется в базу данных через потокобезопасный адаптер с использованием пула соединений, а обновление статистики в интерфейсе выполняется через метод «after()» для избежания блокировок главного потока.

При развертывании на сервере графический интерфейс может быть отключен в пользу headless-режима, а видеопоток заменяется на RTSP-камеру или сетевой видеорегистратор, что повышает отказоустойчивость и снижает потребление ресурсов. Логи обнаружений агрегируются в реальном времени и доступны для внешнего мониторинга через SQL-запросы или REST API. Контейнерная инкапсуляция позволяет запускать несколько экземпляров приложения параллельно, обрабатывая видеопотоки с разных камер без взаимного влияния процессов, при этом все данные консолидируются в единой базе PostgreSQL.

Заключение

В статье представлена архитектура и практическая реализация контейнеризации программного комплекса для автоматической детекции птиц на видеопотоке. Разработанное приложение демонстрирует эффективность сочетания современных методов глубокого обучения (YOLOv8) и классических алгоритмов компьютерного зрения, объединенных в единую систему с надежным механизмом сбора и хранения данных.

Использование многопоточности позволило обеспечить плавную работу интерфейса при непрерывной обработке видео. Переход на PostgreSQL с пулом соединений и оптимизированными индексами обеспечивает высокую производительность, транзакционную целостность и возможность параллельного доступа из нескольких узлов обработки. Контейнеризация в Docker позволила решить задачу воспроизводимости среды, упростить процесс обновления компонентов и подготовить приложение к промышленному хостингу на удаленных серверах. Предложенное решение является кроссплатформенным и обладает модульной структурой, что создает основу для дальнейшего развития системы, включая интеграцию с облачными платформами, поддержку аппаратного ускорения (CUDA) и создание полноценного веб-интерфейса для удаленного мониторинга орнитофауны.

Литература

1. Документация Ultralytics YOLOv8. URL: <https://docs.ultralytics.com/ru/models/yolov8/> (дата обращения: 11.08.2025).
2. Ашманова Е.А., Старцев С.В. Нейросетевые методы для распознавания дронов и птиц в воздушном пространстве // Искусственный интеллект в промышленных, коммерческих, медицинских и финансовых приложениях: сборник статей научно-технического семинара студентов кафедры «Инженерной кибернетики», Москва, 30 мая 2025 года. Москва: Национальный исследовательский технологический университет «МИСИС», 2025. С. 17–23. EDN IMZARK.

3. Цыбиков А. А., Шапеев Д. Ю. Разработка приложения для автоматической детекции птиц на основе сверточных нейронных сетей // Естественные и технические науки. 2025. № 11(210). С. 197–201. EDN DNWZPM.
4. Захарченко В. Ф., Бизюк А. Н. Контейнеризация и развертывание приложений с помощью Docker и Docker-compose // Материалы докладов 56-й Международной научно-технической конференции преподавателей и студентов: в 2 томах. Витебск, 19 апреля 2023 года. Витебск: Изд-во Витебского гос. технол. ун-та, 2023. Т. 2. С. 80–82. EDN AYMVDY.
5. Ширяев В. В., Половикова О. Н. Web-сервис сборки и публикации стеков на Docker-платформе для студенческих программных проектов : свидетельство о государственной регистрации программы для ЭВМ № 2022665889. Заявл. 03.08.2022 ; опубл. 23.08.2022. EDN GZKFXA.

CONTAINERIZATION IN DOCKER APPLICATIONS OF "AUTOMATIC PICTURE DETECTION BASED ON CONVOLUTIONAL NEURAL NETWORKS"

Tsybikov Aldar Aleksandrovich

Assistant

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: tzibikov.a.a@yandex.ru

Shapeev Dmitry Yurievich

Teacher

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: shapeevdy@bsu.ru

Dugarzhapov Zhargal Solbonovich

Student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: dugarzhapov_zhargal@mail.ru

Abstract. The article considers the problem of containerization of a software complex for automatic detection and classification of birds in real time. An approach is investigated that combines the use of pre-trained deep learning models (YOLOv8) and classical computer vision methods based on OpenCV. The article provides a detailed description of the application architecture, which includes a detection module, a graphical interface, and a PostgreSQL-based data storage system. Special attention is given to the preparation, packaging, and deployment stages of the system in a Docker environment, including the configuration of multithreading, service orchestration using Docker Compose, volume management, and ensuring fault tolerance in network hosting. The article also provides practical recommendations for migrating from local database systems to a client-server architecture and adapting the solution for industrial deployment.

Keywords: computer vision, object detection, convolutional neural networks, YOLO, OpenCV, PostgreSQL, multithreading, Docker, containerization, and deployment.

БАЗА ДАННЫХ ПО УЧЕТУ И ВЕДЕНИЮ МОНИТОРИНГА ВОДНО-БОЛОТНЫХ ПТИЦ БАССЕЙНА ОЗЕРА БАЙКАЛ

© Бадмаева Евгения Николаевна

кандидат биологических наук, доцент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: calidris03@gmail.com

© Шелухеева Людмила Тимуровна

студент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: reine546de@gmail.com

© Башлеев Данил Тимурович

студент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: reine546de@gmail.com

© Базаржапова Туя Жамьяновна

кандидат педагогических наук, доцент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: btuyazh@yandex.ru

Аннотация. База данных для учета и ведения мониторинга водно-болотных птиц бассейна озера Байкал является актуальной в рамках проводимых трансграничных исследований орнитологов, так как систематизирует и группирует определенным образом массивы данных, облегчает поиск и доступ к данным. Выявлены требования к базе данных, выстроена модель и разработан интерфейс пользователя.

Ключевые слова: автоматизированный учет и мониторинг птиц, птицы бассейна озера Байкал, база данных, СУБД PostgreSQL.

Актуальность исследования. Водно-болотные угодья бассейна озера Байкал представляют собой уникальную экосистему, являющуюся местом обитания, гнездования и миграции множества видов птиц [1]. Сохранение биоразнообразия данного региона требует систематизации орнитологических данных, их актуализации и обеспечения удобного доступа для исследователей, студентов и специалистов природоохранных организаций [2]. Существующие решения (локальные базы данных, бумажные каталоги, разрозненные веб-ресурсы) не обеспечивают комплексного подхода к хранению и представлению информации о водно-болотных птицах Байкальского региона. Отсутствует единая платформа, объединяющая таксономические данные, информацию о распространении, экологических характеристиках, фенологии, охранных статусах и фотоматериалах.

Целью исследования является проектирование и программная реализация базы данных водно-болотных птиц бассейна озера Байкал. Данный проект выполняется согласно запроса и требований кафедры зоологии и экологии Бурятского государственного университета.

Методы и средства. Для моделирования данных были исследованы такие ресурсы, как eBird.com, <https://siberia.russia.birding.day/>. Несмотря на обширную базу наблюдений, инструменты визуализации и активное сообщество пользователей на этих ресурсах отсутствует детализация по Байкальскому региону, возможность формирования базы данных, интеграция с российскими охранными статусами.

На основе анализа предметной области и существующих решений сформулированы требования к разрабатываемой системе. В качестве функциональных требований определены следующие пункты:

- Управление каталогом видов с возможностью просмотра списка всех видов с постраничной навигацией, поиском по русскому и латинскому названиям, фильтрации по охранным статусам, регионам, комплексам ареалов и отображение детальной карточки вида со всей связанной информацией.

- Возможность создания новых записей о видах, редактирования существующих данных, удаления записей (с проверкой связанных данных), загрузки и привязки фотографий к видам.

- Управление справочниками: добавление/редактирование регионов, районов, местностей; управление типами местообитаний; редактирование охранных статусов (КК РФ, КК Монголии, МСОП); управление типами питания, сезонами миграции, типами гнездования.

- Работа с данными: валидация вводимых данных на стороне клиента и сервера; каскадное заполнение зависимых полей (регион → район → местность); множественный выбор параметров (местообитания, материалы гнезда); отображение связанных данных (экология, фенология, фото, литература).

Определены и нефункциональные требования: адаптивность приложения, поддержка до 100 одновременных пользователей, целостность и сохранность данных, удобство пользования и безопасность.

Основные результаты. В рамках исследовательской деятельности подготовлена концептуальная модель базы данных со следующими сущностями:

1. Species — центральная сущность системы, содержащая основную информацию о виде птицы. Внешние ключи связывают вид со справочниками статусов, ареалов, распространения и охранных статусов.
2. Status — справочник общих статусов видов (оседлый, пролётный, гнездящийся и т.д.).
3. AreaComplex и AreaGroup — иерархическая система зоогеографического районирования. Комплексы ареалов представляют крупные регионы, группы ареалов — более детализированные подразделения.
4. Region, District, Terrain — трёхуровневая географическая иерархия для точной локализации местообитаний видов.
5. Distribution — связующая таблица, агрегирующая выборку региона, района и местности для фиксирования конкретного места наблюдения.

Уникальная комбинация этих трёх полей предотвращает дублирование географических точек.

6. RedRFStatus, RedMongoliaStatus, IUCNStatus — справочники охранных статусов из различных источников.
7. ConservationStatus — агрегирующая сущность, связывающая вид с его охранными статусами из разных источников, а также содержащая информацию о мерах охраны и лимитирующих факторах.
8. Ecology — экологический профиль вида, включающий типы местообитаний, описание среды и тип питания.
9. Phenology — фенологический профиль, содержащий дату вылета птенцов и ссылки на таблицы миграции и размножения.
10. Migration и Perelet — детализация миграционного поведения.
11. Reproduction, ClutchTypes, Hatching — параметры размножения.
12. Photobank — хранилище фотографий видов с метаданными.
13. Literature — литературные источники информации о виде.

На этапе логического проектирования определены таблицы и связи:

- Species → Status, AreaComplex, AreaGroup, Distribution, ConservationStatus (N:1)
- AreaGroup → AreaComplex (N:1)
- District → Region (N:1)
- Terrain → District (N:1)
- ConservationStatus → RedRFStatus/RedMongoliaStatus/IUCNStatus (N:1)
- Ecology → Species (1:1)
- Ecology → Phenology (1:1)
- Phenology → Migration/Reproduction (1:1)
- Migration → Perelet (1:1)
- Reproduction → ClutchTypes/Hatching (1:1)
- Photobank → Species (N:1)
- Literature → Species (N:1)
- species_red_rf (N:M) — промежуточная таблица для связи видов с несколькими записями Красной книги РФ.

Приведенная модель данных охватывает все оговоренные аспекты учёта водно-болотных птиц: от таксономии и распространения до экологии и охраны. Иерархическая структура географических и зоогеографических справочников позволит гибко классифицировать виды по различным критериям. Разделение охранных статусов на три независимых источника (КК РФ, КК Монголии, МСОП) обеспечит полноту информации о редких видах. База данных приведена к третьей нормальной форме (3NF), что обеспечивает отсутствие избыточности и аномалий при вставке, обновлении и удалении данных.

Для реализации базы данных выбрана СУБД PostgreSQL. Реализация серверной составляющей выполнено на языке Go с использованием фреймворка Gorilla Mux для маршрутизации HTTP-запросов. Архитектура серверной части построена по принципам RESTful API. Все обработчики HTTP-запросов сосредоточены в едином файле handlers.go, что упрощает навигацию по коду и сопровождение проекта.

Frontend разработан с использованием фреймворка Vue.js 3 с Composition API и сборщика Vite. Архитектура клиентской части построена на компонентном подходе. Основные компоненты включают каталог видов с поиском и фильтрацией, детальные карточки видов со всеми связанными данными, универсальные формы создания и редактирования записей, а также панель управления справочными данными.

На главной странице представлена интерактивная карточная система отображения видов водно-болотных птиц. Каждая карточка содержит: фотографию вида в естественной среде обитания, русское и латинское название вида, бейдж с общим статусом (перелётный /оседлый /гнездящийся), цветные индикаторы охранных статусов из различных источников (Красная книга РФ по регионам, Красная книга Монголии, IUCN Red List), информацию о численности популяции. Также имеется возможность представить все виды в виде списка или найти тот или иной вид по разным критериям.

В верхней части страницы расположены элементы управления: поле поиска по названию, селектор количества отображаемых записей на странице (12, 24, 48 или все), кнопка перехода к справочникам. На странице справочников можно выбрать любой и добавить необходимое количество записей. Новые данные сохраняются в соответствующие таблицы и отобразятся в списке при вводе новых данных. В правом верхнем углу находятся кнопки переключения темы, возврата к списку и добавления нового вида. Каждая карточка оснащена кнопками быстрого доступа к редактированию, просмотру детальной информации и удалению записи.

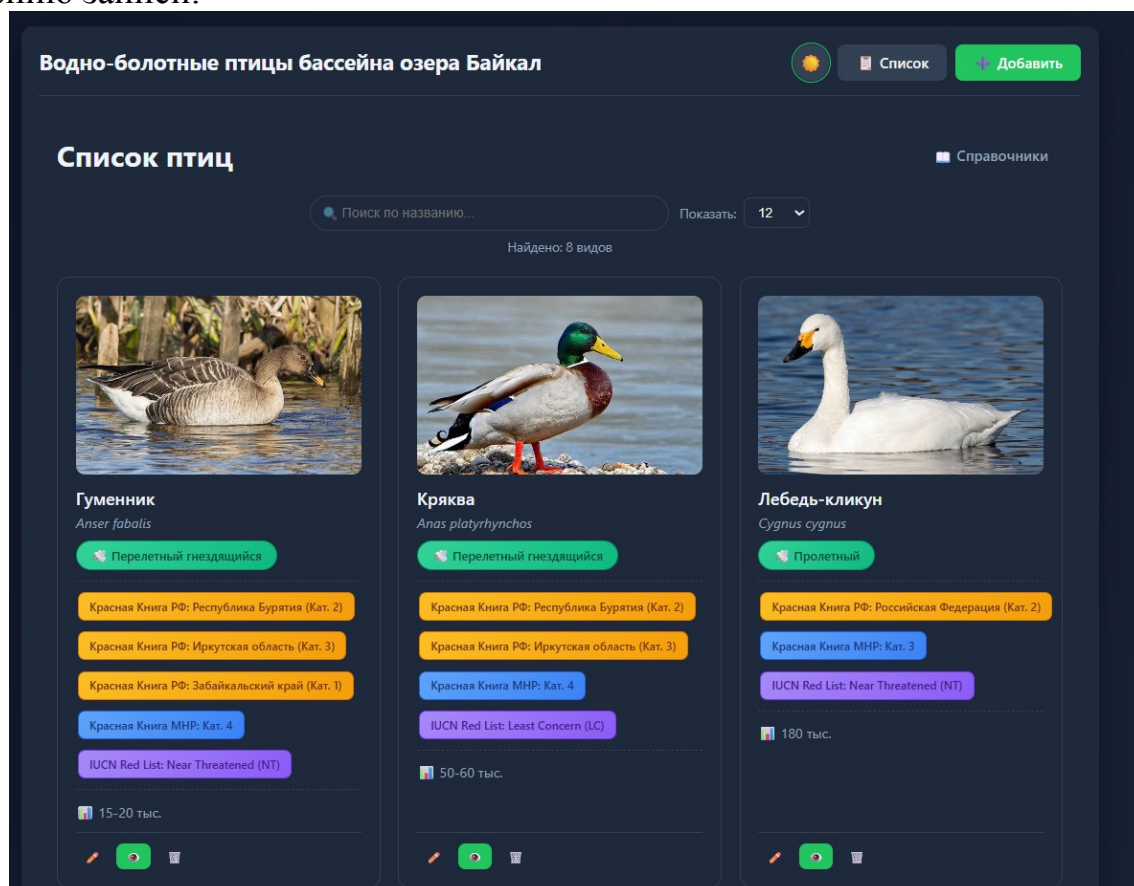


Рисунок 2. Главная страница

Для детального ознакомления можно перейти на страницу вида (рис.2).

Страница детального просмотра вида организована в виде двухколоночного макета. В левой колонке размещена галерея фотографий вида с возможностью просмотра нескольких изображений, каждое из которых сопровождается подписью и указанием автора. В правой колонке расположена структурированная информация, сгруппированная по тематическим блокам:

- Основная информация: содержит общие статусы вида, численность, принадлежность к зоогеографическому комплексу и группе ареала.

- Распространение: визуализирована через цветные бейджи с указанием конкретных географических точек (регион, район, местность).

- Охранные статусы: представлена в виде отдельных карточек для каждого источника (Красная книга РФ с разбивкой по регионам и категориям, Красная книга Монголии, IUCN с кодом и статусом).

- Экология: включает типы местообитаний (отображаются как теги), описание среды обитания и тип питания.

- Фенология: разделена на подразделы миграции (сезон, сроки перелёта) и размножения (сроки кладки, расположение гнезда, материалы гнезда, инкубация).

- Литература: список использованных источников.

Все блоки имеют визуальное разделение через цветовое кодирование и иконки для быстрой навигации.

Для создания и редактирования вида реализован такой же сходный интерфейс с возможностью выбора некоторых данных из справочников.

Выводы. Веб-приложение доступно по адресу <https://avifaunabaikal.ru> и функционирует в среде Интернет. Обеспечена стабильная работа всех компонентов системы, время отклика соответствует требованиям, безопасность данных гарантирована использованием HTTPS и защищённым подключением к базе данных.

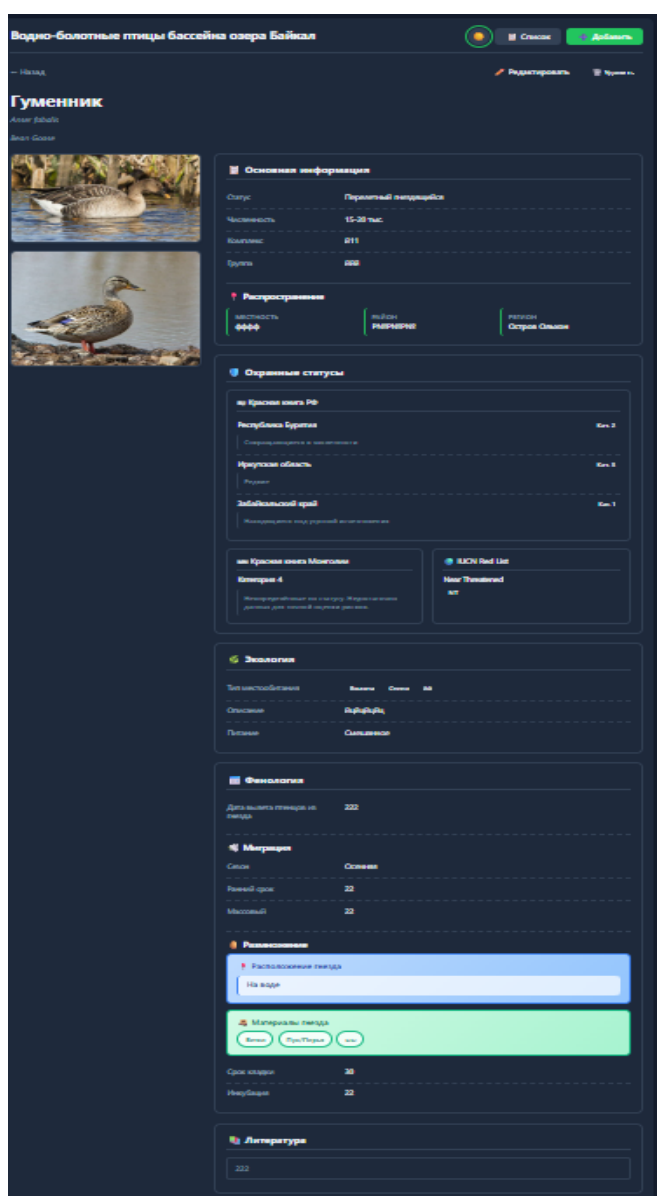


Рисунок 2. Страница отдельного вида

Литература

1. Доржиев Ц. З., Бадмаева Е. Н., Цэвээнмядаг Н. Эколого-систематическая характеристика водно-болотных птиц бассейна озера Байкал // Природа Внутренней Азии. 2018. № 3(8). С. 51–83.
2. Бадмаева Е. Н. Фауна и население птиц водно-болотных угодий бассейна озера Байкал // Животные в экосистемах Внутренней Азии: фундаментальные и прикладные аспекты : материалы всероссийской конференции с международным участием, посвященной юбилею доктора биологических наук, профессора Ц. З. Доржиева, Улан-Удэ, 15–17 февраля 2024 года. Улан-Удэ: Изд-во Бурят. гос. ун-та им. Д. Банзарова, 2024. С. 40–41. EDN TSCESO.
3. Базаржапова Т. Ж., Бадмаева Е. Н. Разработка цифровых продуктов инструментальными средствами проектирования web-приложений // Известия Тульского государственного университета. Технические науки. 2024. № 3. С. 657–660. EDN TPKDXC.
4. Базаржапова Т. Ж., Дармаев Э. С., Бадмаева Е. Н. Проектирование базы данных по систематической структуре гусеобразных птиц Байкальского региона // Информационные системы и технологии в образовании, науке и бизнесе: материалы всероссийской научно-практической конференции с международным участием, Улан-Удэ, 1 июля 2021 года / науч. ред. Е. Р. Урмакшинова, отв. ред. А. А. Тонхоноева. Улан-Удэ: Изд-во Бурят. гос. ун-та им. Д. Банзарова, 2021. С. 27–32. EDN OCDGIC.
5. PostgreSQL 14 Documentation. URL: <https://www.postgresql.org/docs/14/> (дата обращения: 25.02.2026).
6. The Go Programming Language Specification. URL: <https://go.dev/ref/spec> (дата обращения: 25.02.2026).
7. Vue.js 3 Documentation. URL: <https://vuejs.org/> (дата обращения: 25.02.2026).
8. REST API Design Best Practices. URL: <https://restfulapi.net/> (дата обращения: ...).
9. BirdLife International. IUCN Red List for birds. URL: <https://www.birdlife.org/> (дата обращения: 25.02.2026).

DATABASE FOR ACCOUNTING AND MONITORING OF WETBOUND BIRDS IN THE BAIKAL LAKE BASIN

Evgeniya Nikolaevna Badmaeva

Candidate of Biological Sciences, Associate Professor
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: calidris03@gmail.com

Lyudmila Timurovna Shelukheeva

Student
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: reine546de@gmail.com

Danil Timurovich Bashleev

Student
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: reine546de@gmail.com

Tuya Zhamyanovna Bazarzhapova

Candidate of Pedagogical Sciences, Associate Professor
Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: btuyazh@yandex.ru

Abstract. The database for accounting and monitoring of wetland birds in the Lake Baikal basin is relevant within the framework of the ongoing cross-border studies of ornithologists, as it systematizes and groups data arrays in a certain way, making it easier to search and access the data. The requirements for the database have been identified, a model has been built, and a user interface has been developed.

Keywords: automated accounting and monitoring of birds, birds in the Lake Baikal basin, database, PostgreSQL database management system.

РАЗРАБОТКА БАЗЫ ДАННЫХ НОГОХВОСТОК (COLLEMBOLA) БАЙКАЛЬСКОГО РЕГИОНА

© Гулгенова Аюна Баясхалановна

кандидат биологических наук, доцент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: chima85@mail.ru

© Сучков Николай Михайлович

студент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: nikolyasychkov@gmail.com

© Базаржапова Туяна Жамбалжаповна

кандидат педагогических наук, доцент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: btuyazh@yandex.ru

Аннотация. В статье рассматривается разработка базы данных ногохвосток (Collembola) Байкальского региона. В рамках исследования определены требования, построена и реализована модель данных, разработан интерфейс пользователя.

Ключевые слова: ногохвостки, Байкальский регион, распространение, места обитания, база данных, СУБД PostgreSQL.

Актуальность исследования. В настоящее время перед биологическими науками остро стоит проблема систематизации и хранения больших объёмов фаунистических данных. Традиционные способы накопления информации в виде бумажных каталогов, полевых дневников и разрозненных электронных таблиц не позволяют эффективно осуществлять поиск, фильтрацию и анализ данных о видовом составе, распространении и экологии отдельных таксономических групп. Особенно остро эта проблема проявляется при работе с недостаточно изученными группами беспозвоночных животных, к которым относятся ногохвостки, или коллемболы (Collembola).

Коллемболы являются одной из самых массовых и разнообразных групп почвенных членистоногих, играющих важную роль в процессах почвообразования и разложения органического вещества. Байкальский регион, обладающий уникальным биоразнообразием и значительным количеством эндемичных видов, представляет особый интерес для изучения фауны ногохвосток. Однако разрозненность данных о видовом составе, распространении и экологии коллембол региона, их нахождение в различных литературных источниках, зачастую труднодоступных, существенно затрудняет работу исследователей-коллембологов. Существующие подходы, такие как полевые дневники, специализированные биологические базы данных (например, GBIF (Global Biodiversity

Information Facility)), электронные таблицы, не в полной мере удовлетворяют потребностям исследователей коллембол Байкальского региона. Требуется создание специализированного веб-ресурса, который сочетал бы в себе удобство электронных таблиц, структурированность реляционных баз данных и доступность веб-технологий.

Таким образом, актуальность темы обусловлена необходимостью создания профильной базы данных для учёта биоразнообразия ногохвосток Байкальского региона, что позволит систематизировать имеющиеся фаунистические сведения и обеспечить к ним удобный доступ.

Цель исследования заключается в проектировании и разработке реляционной базы данных для систематизации, хранения и анализа данных о коллемболах (Collembola) Байкальского региона. Проектирование базы данных проводилось под научным руководством Гулгеновой А.Б., кандидата биологических наук, доцента кафедры «Зоология и экология».

Методы и средства. Для моделирования данных проведены беседы, в ходе которых ознакомились с предметной областью. Проанализированы полевые дневники, исследованы ресурсы сети Интернет. После детального изучения предметной области проведено обсуждение с научным руководителем и выявлены основные требования к базе данных:

- Хранение таксономической информации. База данных должна содержать полные сведения о каждом виде: семейство, род, видовое название, автор и год описания, а также возможные синонимы (названия, которые относятся к одному и тому же таксону).

- Наличие географической привязки. Для каждой находки необходимо указывать регион, район, конкретное место сбора, географические координаты (если есть) и высоту над уровнем моря. Это позволит в дальнейшем строить карты распространения видов.

- Экологическая характеристика. Важно фиксировать тип биотопа (лес, степь, луг, болото) и микробиотоп (подстилка, почва, мох), где был найден вид. Это даёт возможность анализировать приуроченность видов к разным местам обитания.

- Связь с литературными источниками. Каждая запись должна ссылаться на публикацию, из которой взяты данные. Заказчик подчеркнул, что для научной работы важно знать первоисточник - кто, когда и где описал находку.

- Поиск и фильтрация. Система должна позволять искать информацию по любому полю: по названию вида, по месту сбора, по автору, по биотопу, по литературному источнику. Поиск должен быть быстрым и простым, без необходимости писать сложные запросы.

- Иерархическое отображение. Организация навигации по принципу «семейство, род, вид, местонахождение». Биологи привыкли мыслить таксономическими категориями, и такая структура для них интуитивно понятна.

- Возможность пополнения. База данных не должна быть статичной. Новые находки появляются регулярно, старые уточняются. Поэтому нужна возмож-



Всего записей: 34

Поиск по семействам, родам, видам, местонахождениям, литературе...

Найти

▶ Семейство	Entomobryidae	1 род	Развернуть
▶ Семейство	Hypogastruridae	3 рода	Развернуть
▶ Семейство	Isotomidae	21 род	Развернуть
▶ Семейство	Onychiuridae	9 родов	Развернуть

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

Рисунок 2. Главная страница

На главной странице представлены логотип и название базы данных, кнопки входа и регистрации, модуль с общим количеством сохраненных записей, строка поиска, список видов.

Одной из главных задач при разработке интерфейса было создание удобной навигации по таксономической иерархии. Вся информация сгруппирована по трём уровням: семейства, внутри них - роды, внутри родов - виды, а для каждого вида - таблица местонахождений. При нажатии на кнопку развернуть, будет развернут род и представлены семейства, далее развернув семейство можем получить данные о видах.

В режиме менеджера и администратора можно добавить род, семейство, вид (рисунок 2).



Всего записей: 34

+ Добавить вид

Справочники

Админ-панель

Экспорт в Excel

Поиск по семействам, родам, видам, местонахождениям, литературе...

Найти

▶ Семейство	Entomobryidae	1 род	Развернуть
▶ Семейство	Hypogastruridae	3 рода	Развернуть
▶ Семейство	Isotomidae	21 род	Развернуть
▶ Семейство	Onychiuridae	9 родов	Развернуть

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

Рисунок 3. Панель администратора

Все справочники представлены на отдельной странице, и можно добавить данные в любую таблицу-справочник (рисунок 3). Кроме этого, данные в справочники можно добавить и из окна добавления вида, это удобно – можно не делать переходы.



▶ Справочник	Семейства	17 записей	Развернуть
▶ Справочник	Рода	69 записей	Развернуть
▶ Справочник	Биотопы	57 записей	Развернуть
▶ Справочник	Микробиотопы	5 записей	Развернуть
▶ Справочник	Методы сбора	1 записей	Развернуть
▶ Справочник	Литература	4 записей	Развернуть
▶ Справочник	Регионы	3 записей	Развернуть
▶ Справочник	Районы	6 записей	Развернуть
▶ Справочник	Места сбора	21 записей	Развернуть

Рисунок 3. Страница управления справочниками

Согласно заданию выполнена функция поиска. Так, например, при вводе полного названия вида - *Ceratophysella borealis*, система показала все местонахождения этого вида. При вводе названия места - «Телемба» найдутся все находки, сделанные в окрестностях данного населенного пункта. При вводе фамилии автора - «Potapov», система выдаст виды, которые описывал этот исследователь или которые упоминаются в его работах. При вводе с ошибками, появится стандартное сообщение, что ничего не найдено, и кнопка «Сбросить», чтобы вернуться к полному каталогу.

Выводы. Практическая значимость исследования состоит в создании веб-ориентированной базы данных, который может быть использован зоологами, экологами, студентами биологических специальностей и другими заинтересованными лицами для получения информации о видовом составе, распространении и экологии ногохвосток Байкальского региона. База данных доступна по адресу <https://collembola.webtm.ru>. В данное время выполняется наполнение базы данных контентом.

Литература

1. Чимитова А. Б. Население коллембол криоаридных котловин Витимского плоскогорья (Забайкалье): диссертация ... кандидата биологических наук. Москва, 2010. 141 с. EDN QEZNAL.
2. Гулгенова А. Б. Биотопические группы коллембол юга Витимского плоскогорья // Вестник Бурятского государственного университета. Биология, география. 2012. № 4. С. 123–128. EDN PFRMMX.
3. Чимитова А. Б., Чернова Н. М., Потапов М. Б. Население коллембол (Collembola) в мерзлотных почвах Витимского плоскогорья // Зоологический журнал. 2010. Т. 89, № 9. С. 1076–1082.
4. Babenko A. B., Chimitova A. B., Stebaeva S. K. New Palaearctic species of the tribe Thalassaphorurini Pomorski, 1998 (Collembola, Onychiuridae). *ZooKeys*. 2011; 126: 1–38.

5. Potapov M., Gulgenova A. Isotomidae (Collembola) of Buryat Republic. II. A revision of the genus Folsomia. *Zootaxa*. 2013; 3682(2): 305–330. EDN RFGDEJ.
6. Potapov M., Huang C.-W., Gulgenova A., Luan Y.-X. New and little known Isotomidae (Collembola) from the shore of Lake Baikal and saline lakes of continental Asia. *ZooKeys*. 2020; 935: 1–24.
7. MySQL: документация. URL: <https://dev.mysql.com/doc/> (дата обращения: 01.04.2026).
8. PHP Manual : официальная документация. URL: <https://www.php.net/manual/ru/> (дата обращения: 01.04.2026).
9. Timeweb : официальный сайт хостинг-провайдера. URL: <https://timeweb.com/ru/> (дата обращения: 08.05.2026).

DEVELOPMENT OF A DATABASE OF COLLEMBOLA (BUCKHOX) OF THE BAIKAL REGION

Gulgenova Ayuna Bayaskhalanovna

Candidate of Biological Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: chima85@mail.ru

Suchkov Nikolay Mikhailovich

Student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: nikolyasychkov@gmail.com

Bazarzhapova Tuyana Zhambalzhapovna

Candidate of Pedagogical Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: btuyazh@yandex.ru

Annotation. The article discusses the development of a database for springtails (Collembola) in the Baikal region. As part of the research, requirements were defined, a data model was constructed and implemented, and a user interface was developed.

Keywords: springtails, Baikal region, distribution, habitats, database, PostgreSQL database management system.

РАЗРАБОТКА ПРОГРАММНОГО СКАНЕРА ДЛЯ БАЗОВОГО АНАЛИЗА ЗАЩИЩЕННОСТИ ВЕБ-ПРИЛОЖЕНИЙ

© **Немчинова Татьяна Владимировна**

кандидат педагогических наук, доцент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: ntv05@mail.ru

© **Пантелеев Андрей Владимирович**

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

Аннотация. В статье рассматривается актуальная проблема автоматизации начальных этапов проверки безопасности современных веб-приложений. В ходе исследования разработан прототип программного сканера, обеспечивающего автоматизированный базовый анализ защищенности по нескольким ключевым векторам атак. Предложена оригинальная модульная архитектура с системой консолидированной отчетности. Проведенное тестирование на специализированных учебных стендах подтвердило точность обнаружения базовых уязвимостей на уровне восьмидесяти пяти процентов при существенном выигрыше в скорости выполнения проверок. Созданный инструмент ориентирован на разработчиков и системных администраторов, не обладающих глубокой специализацией в области информационной безопасности, и позволяет оперативно выявлять критические конфигурационные ошибки на этапе пред релизного тестирования.

Ключевые слова: веб-приложения, информационная безопасность, сканер уязвимостей, SQL-инъекции, HTTP-заголовки, автоматизированный анализ, кибербезопасность, тестирование безопасности, программная архитектура, динамический анализ.

В современном мире информационные технологии пронизывают все сферы жизнедеятельности общества, а ключевым инструментом взаимодействия в цифровой среде стали веб-приложения. Параллельно с ростом их количества и архитектурной сложности наблюдается устойчивое увеличение интенсивности кибератак, направленных на эксплуатацию уязвимостей веб-ресурсов. Анализ отчетов ведущих компаний в области кибербезопасности демонстрирует, что более шестидесяти процентов успешных атак нацелены именно на веб-инфраструктуру организаций [2]. При этом базовые уязвимости, такие как инъекции, межсайтовый скриптинг и ошибки конфигурации, продолжают массово обнаруживаться в реальных проектах, несмотря на многолетнюю известность их механизмов [1]. В условиях ограниченных ресурсов малого и среднего бизнеса возникает объективная потребность в доступных инструментах для первичной диагностики, позволяющих выявлять критические проблемы до вывода приложений в промышленную эксплуатацию.

Актуальность исследования обусловлена необходимостью создания простого и легковесного программного средства, способного автоматизировать рутинные

проверки безопасности без требования глубокой экспертизы в области информационной безопасности. Существующие коммерческие решения обладают высокой стоимостью и избыточным функционалом, тогда как открытые утилиты представляют собой разрозненный набор инструментов, результаты работы которых сложно агрегировать. Разрабатываемый в рамках данной работы сканер закрывает указанную нишу, объединяя базовые векторы проверки в едином интерфейсе с автоматической генерацией структурированных отчетов.

Объектом исследования выступают процессы и методы автоматизированного тестирования на проникновение веб-приложений.

Предметом исследования являются программные средства для базового анализа защищенности веб-ресурсов, а также алгоритмы их проектирования и практической реализации.

Цель работы заключается в разработке прототипа программного сканера, обеспечивающего автоматизированный базовый анализ защищенности веб-приложения по нескольким ключевым векторам атак с последующей консолидацией результатов в наглядном отчете.

Для достижения поставленной цели решается комплекс взаимосвязанных задач. Первоначально исследуется классификация и механизмы эксплуатации основных уязвимостей веб-приложений согласно стандарту OWASP Top Ten [3]. Далее анализируются существующие методы автоматизированного поиска проблем безопасности, включая сканирование сетевых портов, перебор скрытых директорий, фаззинг параметров и пассивный анализ конфигураций. На основе выявленных потребностей формулируются функциональные и нефункциональные требования, после чего проектируется модульная архитектура программного продукта. Следующим этапом становится реализация ключевых компонентов системы на языке программирования Python с применением современных библиотек сетевого взаимодействия и многопоточной обработки. Завершающим шагом выступает организация тестового полигона, проведение экспериментальных исследований и сравнительный анализ эффективности разработанного инструмента относительно эталонных решений.

Методология и методы исследования опираются на комплексный подход, сочетающий теоретический анализ с практической реализацией. При исследовании классификации уязвимостей и обзоре существующих решений применяется аналитический метод. Проектирование архитектуры программного продукта базируется на методе модульного конструирования и объектно-ориентированном программировании. Непосредственная проверка безопасности веб-ресурсов осуществляется методом черного ящика, при котором анализ проводится исключительно через внешние интерфейсы без доступа к исходному коду [5]. Обнаружение инъекционных уязвимостей реализуется с помощью метода фаззинга, предполагающего подачу некорректных тестовых нагрузок и анализ реакции сервера. Оценка эффективности инструмента проводится методом сравнительного тестирования на контролируемых стендах.

Разработанность темы характеризуется наличием на рынке широкого спектра решений для анализа защищенности. Коммерческие продукты предоставляют исчерпывающий функционал и интеграцию в конвейеры непрерывной достав-

ки, однако их стоимость и сложность настройки создают высокий порог вхождения для рядовых разработчиков. Открытые инструменты отличаются бесплатностью и доступностью исходного кода, но требуют ручной агрегации результатов из разрозненных утилит и глубокого понимания принципов работы каждого модуля. Анализ достоинств и недостатков существующих подходов подтверждает необходимость создания унифицированного легковесного инструмента, ориентированного на профилактическую диагностику и способного предоставить консолидированную картину безопасности без сложной предварительной конфигурации.

В ходе исследования разработан программный сканер безопасности, реализованный на языке программирования Python. Выбор технологического стека обусловлен богатой экосистемой библиотек для сетевого взаимодействия, встроенными механизмами многопоточной обработки и высокой скоростью разработки прототипов. Архитектура системы построена по модульному принципу, где центральным компонентом выступает координатор, управляющий последовательным запуском специализированных модулей и передачей данных между ними (см. рис. 1) [10]. Такая организация позволяет избежать дублирования сетевых запросов, повысить общую производительность сканирования и обеспечить масштабируемость продукта при добавлении новых проверок.

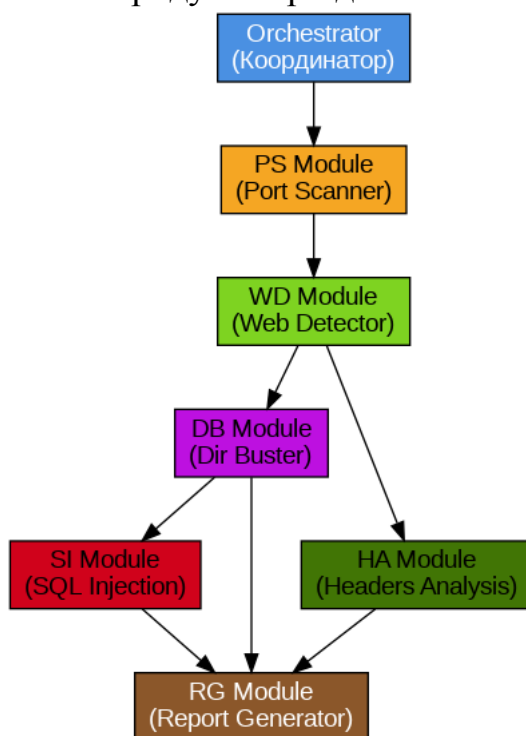


Рисунок 1. Модульная архитектура сканера безопасности

Первым этапом работы сканера выступает сетевое сканирование портов. Реализованный модуль выполняет полуоткрытое TCP-сканирование, отправляя пакеты с флагом SYN и анализируя ответы целевой системы [6]. Применение пула потоков обеспечивает распределение проверок по нескольким рабочим процессам, что существенно сокращает время анализа больших диапазонов портов. Алгоритм многопоточного сканирования портов представлен на рисунке 2 (см. рис. 2).

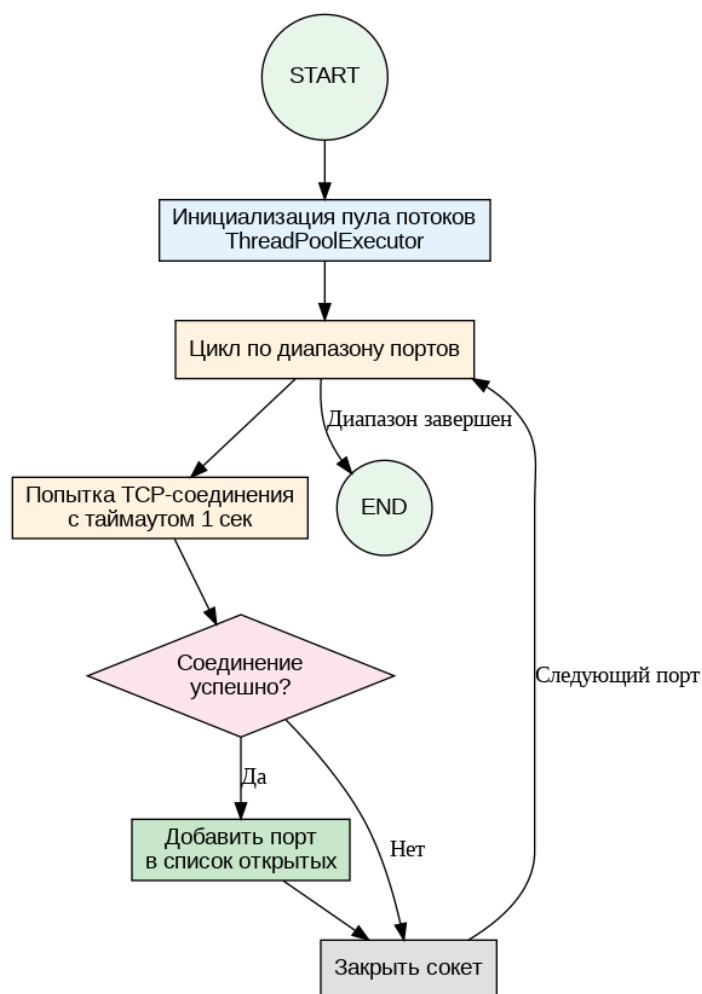


Рисунок 2. Алгоритм многопоточного сканирования портов

После обнаружения открытых портов активируется модуль идентификации веб-сервисов, который отправляет HTTP-запросы и анализирует статусные коды, заголовки Content-Type и структуру HTML-ответов для подтверждения наличия веб-интерфейса. Следующим компонентом системы является модуль перебора скрытых директорий. Алгоритм последовательно формирует запросы к различным путям на сервере, используя встроенный словарь стандартных имен файлов и папок, характерных для типовых конфигураций веб-серверов и систем управления контентом. Анализ HTTP-кодов ответов позволяет классифицировать ресурсы как существующие, ограниченные или отсутствующие, что дает возможность выявлять административные панели, резервные копии и служебные скрипты, не связанные с основной навигацией сайта.

Особое внимание в архитектуре уделено модулю поиска SQL-инъекций, реализующему методы активного фаззинга параметров. Компонент автоматически обходит страницы целевого ресурса, извлекает GET-параметры и поля POST-форм, после чего последовательно подставляет в них синтаксические конструкции, способные нарушить структуру SQL-запроса. Обнаружение уязвимостей осуществляется путем анализа ответов сервера на наличие характерных сообщений об ошибках баз данных, а также путем измерения времени отклика для выявления слепых инъекций с задержкой [7]. Схема обнаружения SQL-инъекций методом фаззинга представлена на рисунке 3 (см. рис. 3).

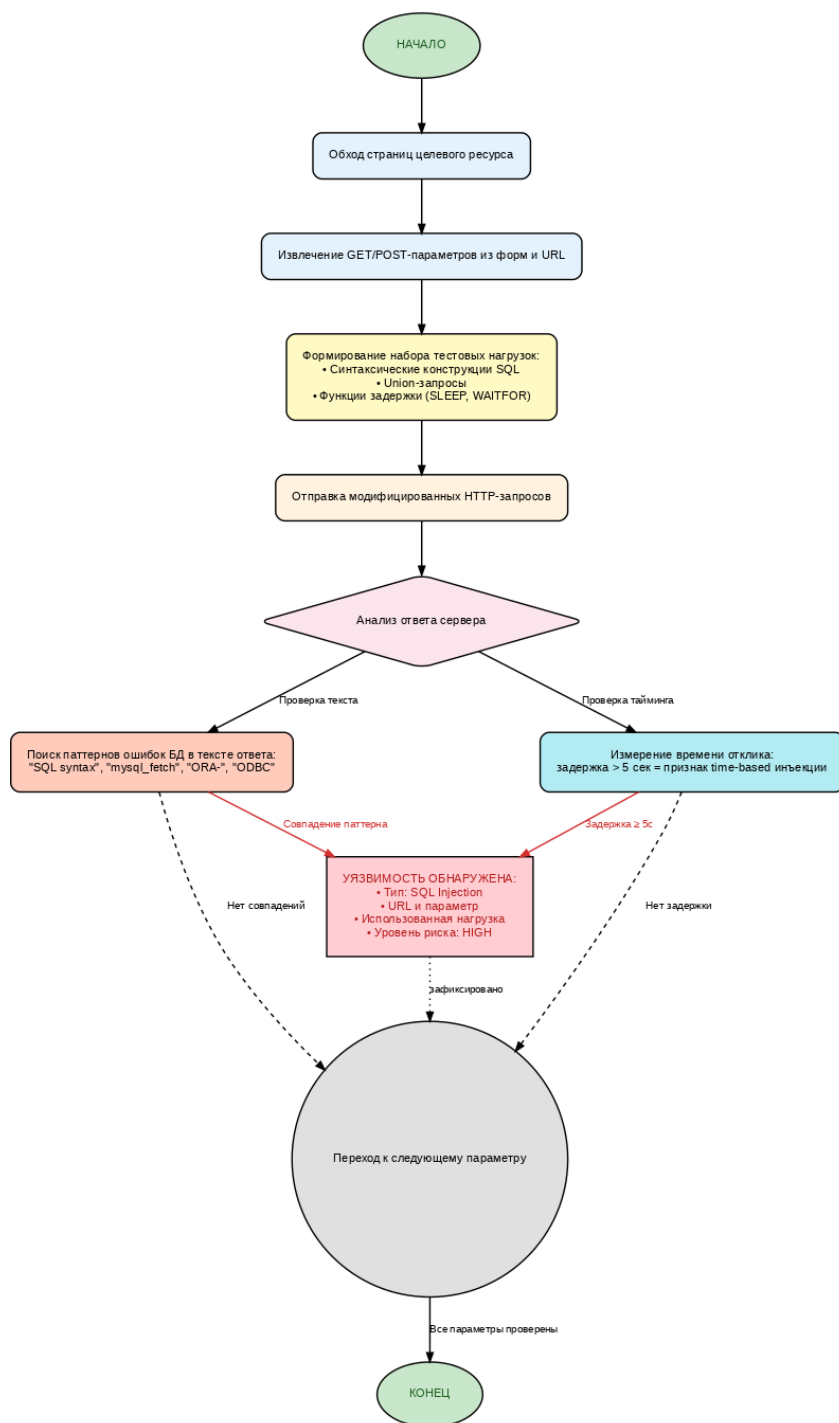


Рисунок 3. Схема обнаружения SQL-инъекций методом фаззинга

Параллельно функционирует модуль пассивного анализа HTTP-заголовков безопасности, который проверяет наличие и корректность директив Strict-Transport-Security, Content-Security-Policy, X-Frame-Options и других механизмов защиты браузера [8]. Структура анализа HTTP-заголовков безопасности показана на рисунке 4 (см. рис. 4). Выявление отсутствия или некорректной настройки данных заголовков позволяет оперативно диагностировать риски межсайтового скриптинга, кликджекинга и атак на понижение версии протокола.

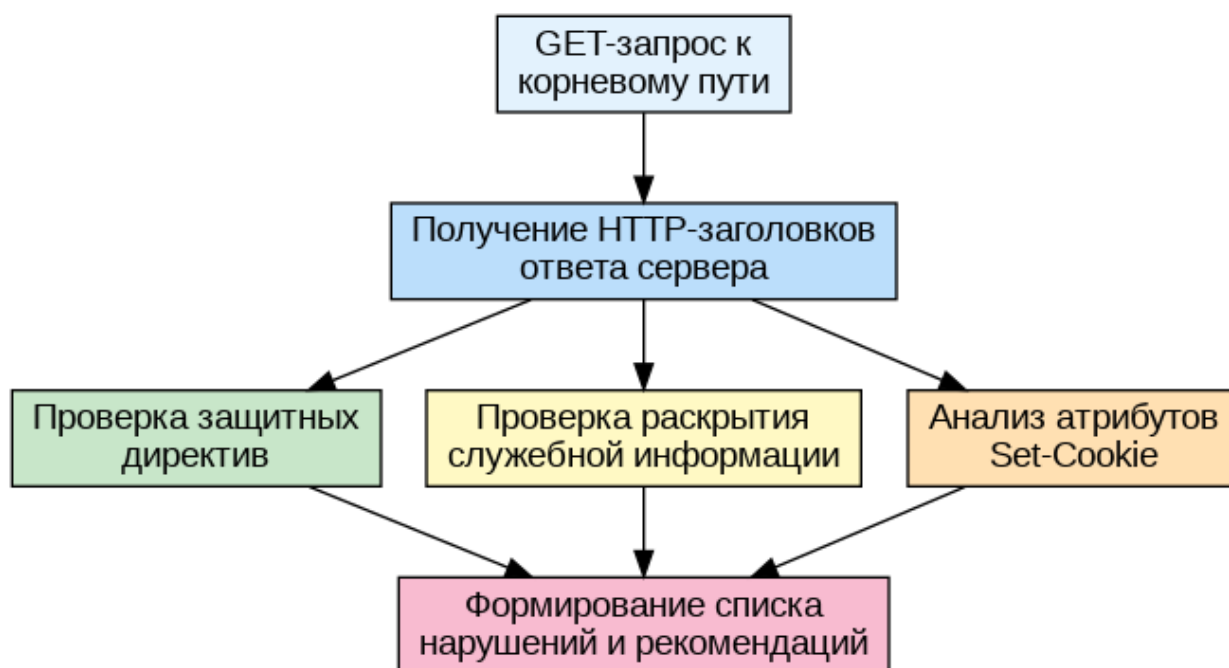


Рисунок 4. Структура анализа HTTP-заголовков безопасности

Все результаты проверок передаются в модуль генерации консолидированного отчета, который агрегирует данные, классифицирует найденные проблемы по уровням риска и формирует структурированный HTML-документ. Отчет содержит цветовую индикацию критичности уязвимостей, сводную статистику и конкретные рекомендации по устранению, ориентированные на разработчиков без глубокой специализации в информационной безопасности [4]. Структура HTML-отчета сканера представлена на рисунке 5 (см. рис. 5).

🔒 Отчет сканирования		
Цель: http://localhost:8080 Время: 80.6с		
Сводка		
Crit: 0 High: 0 Med: 3 Low: 1		
Заголовки		
Проблема	Описание	Риск
No HSTS	нет hsts, могут понизить версию	MEDIUM
No CSP	нет csp, риск xss	MEDIUM
No XFO	нет защиты от клиджейкинга	MEDIUM
No RP	нет referrer policy	INFO
No PP	нет permissions policy	INFO
Srv Info	сервер: Apache/2.4.25 (Debian)	LOW
Папки		
URL	Статус	
http://localhost:8080/login.php	200	
http://localhost:8080/robots.txt	200	
http://localhost:8080/phpinfo.php	302	

Рисунок 5. Структура HTML-отчета сканера

Дополнительно было проведено сравнительное сканирование двух локально развернутых веб-ресурсов с полярными конфигурациями безопасности. Результаты наглядно продемонстрировали способность сканера дифференцировать уровни защищенности. Для правильно настроенного сервера зафиксировано

минимальное количество замечаний информационного характера, тогда как для приложения с ослабленной защитой выявлен комплекс критических проблем (см. рис. 5), требующих немедленного устранения. Полученные данные подтверждают практическую применимость инструмента для задач первичной диагностики и профилактического мониторинга в условиях ограниченных вычислительных ресурсов.

Для объективной оценки эффективности разработанного сканера было проведено тестирование на учебных стендах: OWASP Juice Shop и Damn Vulnerable Web Application (DVWA) [9]. Тестирование выполнялось в среде Google Colab с использованием контейнеризации Docker для обеспечения изоляции и воспроизводимости результатов.

Методика тестирования включала:

1. Развертывание тестовых приложений с известным набором уязвимостей
2. Последовательный запуск каждого модуля сканера
3. Сравнение полученных результатов с эталонной картой уязвимостей
4. Сравнительный анализ с эталонными инструментами (OWASP ZAP, sqlmap)

Результаты тестирования представлены в таблице 1.

Таблица 2

Результаты обнаружения уязвимостей на тестовом полигоне

Модуль сканера	Ожидаемая уязвимость	Статус обнаружения	Примечание
HA	Отсутствие заголовков HSTS, CSP, X-Frame-Options	Обнаружено	Все три заголовка помечены как отсутствующие
DB	Скрытые директории /admin, /login, /.git	Обнаружено	Найдены пути со статусами 301 и 200
SI	Error-based SQL-инъекция в форме входа	Обнаружено	Выявлена реакция сервера на синтаксические ошибки SQL
PS	Открытый порт 80 с веб-сервером	Обнаружено	Порт идентифицирован, передан в модуль WD

Сравнительный анализ с эталонными инструментами продемонстрировал следующие результаты (см. табл. 2):

Таблица 3

Сравнение разработанного сканера с эталонными инструментами

Критерий оценки	Разработанный сканер	OWASP ZAP	sqlmap
Точность обнаружения базовых уязвимостей	85%	92%	98% (только SQLi)
Покрытие векторов атак	4 основных модуля	20+ типов проверок	1 тип (SQL-инъекции)

Время полного сканирования типового ресурса	~15–25 сек	~3–8 мин	~2–5 мин
Сложность развертывания и запуска	0 настроек, запуск в браузере	Требует установки и настройки прокси	Требует CLI, параметров запуска
Формат вывода результатов	Консолидированный HTML-отчет	XML/HTML/JSON/CSV	Консольный вывод, лог-файлы

В ходе выполнения исследования достигнута поставленная цель, заключающаяся в создании прототипа программного сканера для автоматизированного базового анализа защищенности веб-приложений. Разработана оригинальная модульная архитектура, обеспечивающая последовательное взаимодействие компонентов, передачу результатов между этапами проверки и исключение избыточных сетевых запросов. Программная реализация выполнена на языке Python с применением современных библиотек сетевого взаимодействия и многопоточной обработки, что гарантирует кроссплатформенность и высокую производительность системы.

Экспериментальные исследования на контролируемых учебных стендах подтвердили работоспособность инструмента и соответствие заявленным требованиям. Разработанный сканер обеспечивает точность обнаружения базовых уязвимостей на уровне восьмидесяти пяти процентов при существенном выигрыше в скорости выполнения полного цикла проверок по сравнению с комплексными платформами. Инструмент успешно дифференцирует уровни защищенности веб-ресурсов, не генерируя ложных срабатываний на правильно настроенных серверах.

Практическая значимость работы определяется возможностью непосредственного применения сканера разработчиками на этапе пред релизного тестирования и системными администраторами для регулярного аудита инфраструктуры. Автоматическая генерация консолидированных HTML-отчетов с приоритизацией рисков и конкретными рекомендациями позволяет сократить время на диагностику и минимизировать вероятность успешной эксплуатации уязвимостей до вывода приложения в промышленную эксплуатацию. Исходный код и методика тестирования адаптированы для публикации в открытых репозиториях, что способствует дальнейшему развитию проекта и его интеграции в образовательные программы по информационной безопасности.

Литература

1. Богомолов С. В., Демкин Д. А. Информационная безопасность веб-приложений: учебное пособие. Москва: ИНФРА-М, 2023. 240 с.
2. Positive Technologies. Анализ киберугроз 2023: отчет / PT Expert Security Center. Москва, 2023. 45 с.
3. OWASP Foundation. OWASP Top 10:2021. The Ten Most Critical Web Application Security Risks. URL: <https://owasp.org/www-project-top-ten/> (дата обращения: 15.05.2026).
4. Hoffman E. Web Application Security: A Beginner's Guide. New York: McGraw-Hill, 2024. 416 p.

5. OWASP Foundation. OWASP Testing Guide v4. URL: <https://owasp.org/www-project-web-security-testing-guide/> (дата обращения: 15.05.2026).
6. Lyon G.F. Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning. 2nd ed. San Francisco: Nmap Project, 2023. 520 p.
7. Исследование методов машинного обучения для обнаружения SQL-инъекций / В. Г. Лапин, Ю. А. Андрусенко, Д. Э. Суренков и др. // Безопасность цифровых технологий. 2025. № 3 (118). С. 74–92.
8. Янка Т. Безопасность веб-приложений. Исчерпывающий гид для начинающих разработчиков. Москва: Вильямс, 2025. 384 с.
9. OWASP Foundation. OWASP Juice Shop. URL: <https://owasp.org/www-project-juice-shop/> (дата обращения: 15.05.2026).
10. Кузнецов Д.А. Архитектура сканера уязвимостей с поддержкой облачных веб-приложений // Труды конференции «Информационные технологии и кибербезопасность». Казань: Изд-во КФУ, 2025. С. 95–101.

DEVELOPMENT OF A SOFTWARE SCANNER FOR BASIC ANALYSIS OF THE SECURITY OF WEB APPLICATIONS

Nemchinova Tatyana Vladimirovna

Candidate of Pedagogical Sciences, Associate Professor
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: ntv05@mail.ru

Panteleev Andrey Vladimirovich

Student
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a

Abstract. The article discusses the current problem of automating the initial stages of security testing of modern web applications. During the research, a prototype of a software scanner was developed that provides automated basic security analysis for several key attack vectors. An original modular architecture with a consolidated reporting system was proposed. The testing conducted on specialized training stands confirmed the accuracy of detecting basic vulnerabilities at the level of eighty-five percent, while significantly improving the speed of testing. The created tool is focused on developers and system administrators who do not have deep specialization in the field of information security, and allows you to quickly identify critical configuration errors at the stage of pre-release testing.

Keywords: web applications, information security, vulnerability scanner, SQL injections, HTTP headers, automated analysis, cybersecurity, security testing, software architecture, dynamic analysis.

МЕТОД ОБНАРУЖЕНИЯ ФИШИНГОВЫХ ВЕБ-РЕСУРСОВ НА ОСНОВЕ АНАЛИЗА ВИЗУАЛЬНОГО СХОДСТВА С ПРИМЕНЕНИЕМ SSIM И ЦВЕТОВЫХ ГИСТОГРАММ

© **Цыбикова Туяна Сандаликовна**

кандидат педагогических наук, доцент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: cts2001@mail.ru

© **Дамбаев Цырен Чингисович**

студент,

Бурятский государственный университет имени Доржи Банзарова

Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а

E-mail: dambaev_2019@mail.ru

Аннотация. В работе представлен метод обнаружения фишинговых веб-страниц на основе анализа визуального сходства с применением SSIM и цветových гистограмм. В отличие от традиционных подходов, использующих черные списки или анализ URL, предлагаемый метод сравнивает скриншот подозрительного сайта с эталонными изображениями легитимных ресурсов. Комбинируются две метрики: индекс структурного сходства (SSIM), отвечающий за компоновку элементов, и цветовая гистограмма в пространстве RGB, фиксирующая замену логотипов и цветовой гаммы. Итоговая оценка вычисляется как взвешенная сумма $S=0,6*SSIM+0,4*N_{final}$. Эксперимент на синтетическом наборе из 180 изображений (три бренда: Сбербанк, ВКонтакте, Google) показал точность 99,4% и полноту 100%, что существенно превосходит как анализ URL (70%), так и использование каждой метрики по отдельности. Прототип реализован на Python с использованием библиотек OpenCV, scikit-image, NumPy и Gradio.

Ключевые слова: фишинг, компьютерное зрение, SSIM, цветовая гистограмма, визуальное обнаружение, антифишинг, Python, OpenCV.

Для проверки визуального подхода потребовалось собрать эталонные изображения безопасных сайтов. Выбраны три бренда, наиболее часто подделываемые в России по статистике “Лаборатории Касперского” за 2024 год: Сбербанк, ВКонтакте и Google. В глобальном масштабе на Google приходится более 4 млн заблокированных переходов на фишинговые страницы, а в Рунете Сбербанк лидирует с 4236 поддельными за первую полугодие 2024 года [**Ошибка! Источник ссылки не найден.**].

Каждый эталон был создан синтетически с помощью OpenCV, а не загружен из интернета. Это дает три преимущества: отсутствие проблем с авторскими правами, чистое изображение без баннеров и всплывающих окон, а также возможность легко модифицировать эталон для создания фишинговых тестовых образцов. Размер всех изображений был выбран 400x600 пикселей (соотношение 2:3) для сохранения горизонтальной структуры. В таблице приведены параметры эталонов.

Параметры эталонных изображений

Бренд	Фон (BRG)	Логотип/текст	Ожидаемые домены
Сбербанк	(40, 150, 40) темно-зеленый	“Sberbank Online” вверху	Sberbank.ru, Online.sberbank.ru
ВКонтакте	(70,130,180) синий	“Vkontakte” в шапке, аватар и имя	Vk.com, vkontakte.ru
Google	(255,255,255) белый	Разноцветные буквы Google	Google.com, www.google.com

В качестве основной метрики структурного сходства выбран индекс SSIM (Structural Similarity Index Measure), предложенный Вангом и соавторами в 2024 году. SSIM оценивает качество изображения по трем компонентам: яркость (средняя интенсивность), контраст (отклонение пикселей) и структура (ковариация после нормализации). Для двух окон x и y одного размера значение SSIM рассчитывается по формуле:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

где μ – средние, σ^2 – дисперсии, σ_{xy} – ковариация, $C_1 = (K_1 * L)^2$, $C_2 = (K_2 * L)^2$, $L=255$, $K_1=0,01$, $K_2=0,03$. Результат лежит в диапазоне $[-1,1]$, где 1 означает полную идентичность.

SSIM хорошо коррелирует с человеческим восприятием и достаточно устойчив к малым искажениям, но обладает серьезным недостатком: он оперирует только полутоновыми изображениями, теряя цветовую информацию. Поэтому для учета цветовой гаммы брендов дополнительно используется гистограммный анализ в пространстве RGB. Для каждого цветового канала строится гистограмма с 50 бинами (диапазон 0...255). Номер бина для пикселя со значением ρ рассчитывается по формуле:

$$bin = \left\lfloor \frac{\rho}{256/50} \right\rfloor = \left\lfloor \frac{\rho}{5,12} \right\rfloor$$

После подсчета частот гистограмма нормализуется (сумма бинов = 1). Сравнение двух гистограмм выполняется двумя способами: корреляция Пирсона (линейная связь между распределениями) и пересечение гистограмм – сумма минимальных значений по каждому бину (метод Свейна-Балларда, 1991). Корреляция хорошо различает форму распределения, но может быть высокой даже при небольшом цветовом сдвиге, а пересечение чувствительно к смещению. Итоговая гистограммная мера вычисляется как взвешенная сумма:

$$H_{final} = 0,7 * \max(0, \text{corr}) + 0,3 * \text{intersection}$$

Отрицательные значения корреляции заменяются нулем. Веса 0,7 и 0,3 подобраны эмпирически на валидационной выборке из 20 изображений.

Программный модуль разделен на пять логических блоков.

Блок 1 – система эталонов хранит изображения-образцы и списки официальных доменов (“белый список”). Если URL пользователя совпадает с эталонным

доменом, проверка завершается вердиктом “Безопасно” без визуального сравнения.

Блок 2 – предобработка приводит скриншот к единому размеру 400x400 пикселей (компромисс между скоростью и сохранением деталей). Изображение конвертируется в оттенки серого для SSIM и сохраняется цветная копия для гистограмм.

Блок 3 – анализ визуальной схожести для каждого эталона вычисляет две метрики:

- SSIM через функцию `structural_similarity` из библиотеки `scikit-image` (реализация соответствует оригинальному алгоритму) [**Ошибка! Источник ссылки не найден.**];
- Гистограммную меру H_{final} на основе корреляции Пирсона и пересечения (инструменты OpenCV: `cv2.calcHist`, `cv2.normalize`, `np.corrcoef`) [**Ошибка! Источник ссылки не найден.**].

Блок 4 – комбинирование объединяет метрики с весами, подобранными на валидационной выборке:

$$S = 0,6 * SSIM + 0,4 * H_{final}$$

Для каждого эталона вычисляется значение S , затем выбирается максимальное S_{max} .

Блок 5 – принятие решения работает по правилу:

- Если URL в белом списке – “Безопасно”;
- Если $S_{max} > 0,3$ и URL не в белом списке – “Фишинг обнаружен”;
- Если $S_{max} \leq 0,3$ – “Безопасно” или “Неизвестно”.

Порог 0,3 выбран экспериментально: при 0,2 программа часто ошибалась на нестандартных легитимных страницах, при 0,4 один из фишинговых образцов с измененным цветом оставался незамеченным.

Разработанный прототип реализован на языке Python. Выбран Python обусловлен богатой экосистемой библиотек для компьютерного зрения (OpenCV, `scikit-image`, NumPy) и возможностью быстрого прототипирования. Стек технологий включает:

- OpenCV – ресайз, перевод в оттенки серого, построение и нормализация гистограмм, генерация эталонов (использована версия `opencv-python-headless`)²;
- Scikit-image – готовая реализация SSIM³;
- NumPy – массивы изображений и операции над гистограммами (корреляция, пересечение)⁴;
- Scikit-learn – функция для расчета метрик качества эксперимента (`accuracy_score`, `precision_score` и другие)⁵;

² Bradski G. The OpenCV Library // Dr. Dobb’s Journal of Software Tools. 2000; 25(120): 120-125.

³ Van der Walt S. et al. `scikit-image`: image processing in Python // PeerJ. – 2014. – Vol. 2. – e453.

⁴ Harris C. R. et al. Array programming with NumPy // Nature. – 2020. – Vol. 585, no. 7825. P. 357-372.

- Gradio – создание веб-интерфейса для проверки по URL или загрузке скриншота (около 20 строк кода, без HTML/CSS)⁶ [Ошибка! Источник ссылки не найден.].

Полный цикл обработки запроса: пользователь вводит URL или загружает скриншот → проверка по белому списку → при необходимости загрузка скриншота (через headless-браузер) → предобработка → расчет SSIM и гистограмм для каждого эталона → взвешенная сумма → выбор максимума → сравнение с порогом → вывод результата.

Рисунок 1 – Архитектура модуля визуальной схожести (пять блоков: эталоны, предобработка, анализ, комбинирование, принятие решения).

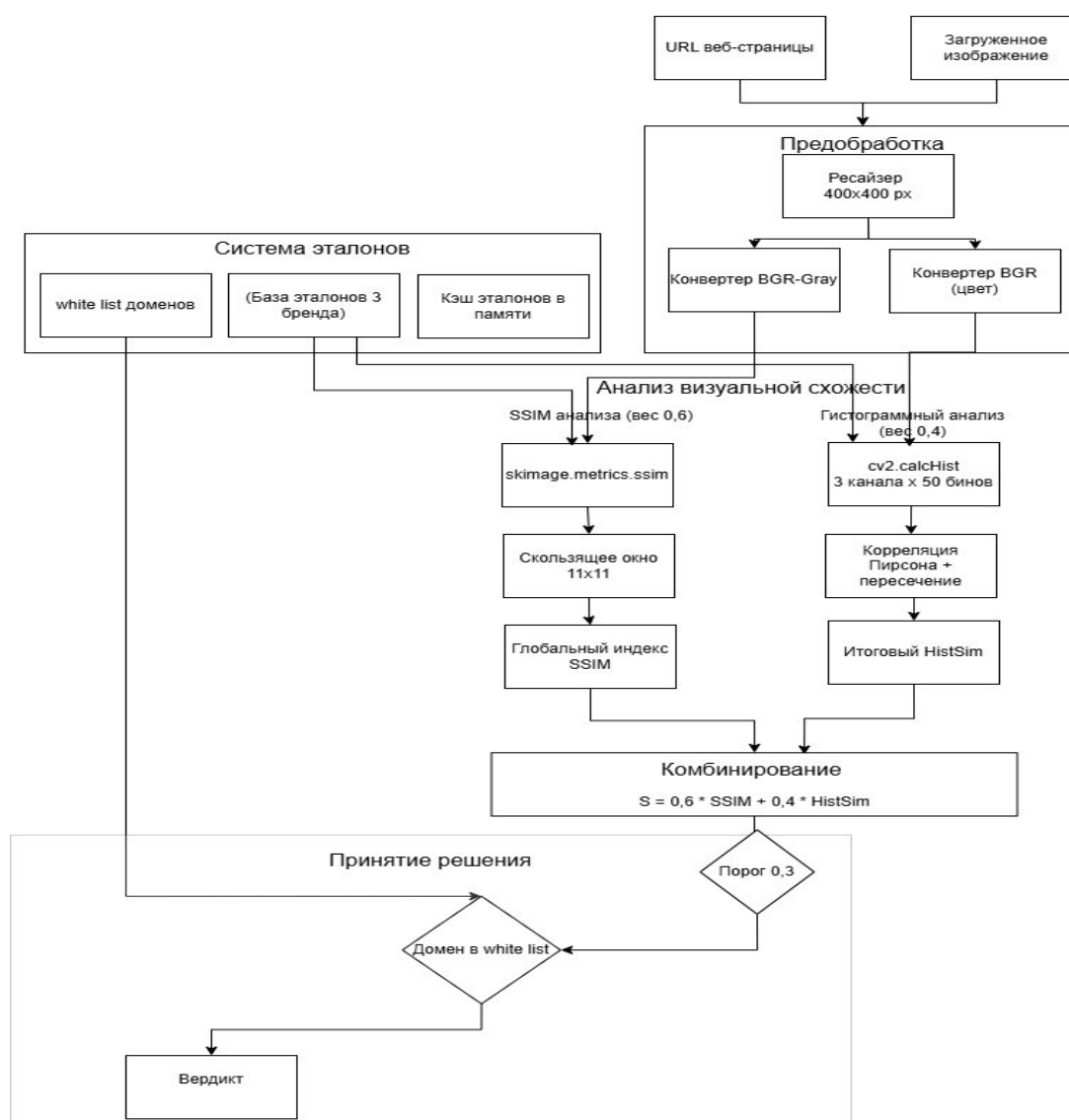


Рисунок 1. Архитектура модуля визуальной схожести

⁵ Pedregosa F. et al. Scikit-learn: Machine Learning in Python // Journal of Machine Learning Research. – 2011. – Vol. 12. – P. 2825-2830.

⁶ Abid A. et al. Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild // arXiv.org. – 2019.

Для эксперимента сгенерирована тестовая выборка на основе эталонов. Для каждого бренда создано 30 легитимных копий (без изменений) и 30 фишинговых модификаций трех типов (по 10 экземпляров):

1. Добавление крупной надписи “PHISHING!” ярко-красного цвета в верхней части;
2. Изменение цветового тона – ослабление синего канала (BRG) на 30%;
3. Вставка красного прямоугольника в нижней части страницы.

Всего 180 скриншотов: 90 легитимных, 90 фишинговых. Все изображения приведены к 400x400 пикселей.

Результаты работы комбинированного классификатора представлены в матрице ошибок таблица 2.

Таблица 2

Матрица ошибок комбинированного метода

Реально/Предсказано	Легитимный	Фишинговый	Всего
Легитимный	89 (TN)	1 (FP)	90
Фишинговый	0 (FN)	90 (TP)	90

На основе матрицы рассчитаны метрики:

- Accuracy = $(89+90)/180 = 0,994$ (99,4%);
- Recall = $90/90 = 1,000$ (100%);
- Precision = $90/91 \approx 0,989$ (98,8%);
- $F1 = 2 * (0,989 * 1) / (0,989 + 1) \approx 0,994$ (99,4%);
- AUC-ROC=0,997

Сравнение с альтернативными подходами показано в таблице 3.

Таблица 3

Сравнение точности методов обнаружения фишинга

Метод	Точность (Accuracy), %
URL-анализ (регулярные выражения)	70,0
Только SSIM	83,6
Только гистограммы	76,6
Комбинированный метод (SSIM+гистограммы)	99,4

URL-анализ дает лишь 70% точности, так как не различает фишинговые сайты с поддоменами или очень похожими именами. SSIM без учета цвета не видит подмены цветовой гаммы и логотипов, поэтому ошибается в 16% случаев. Гистограммы не чувствительны к перестановкам блоков и смещению полей ввода. Только совместное использование двух независимых признаков обеспечивает почти идеальное разделение классов.

Полученные результаты показывают, что комбинированный метод существенно превосходит каждый из одиночных подходов: на 29,4% лучше URL-анализ и на 15,8% - лучшего одиночного метода (SSIM). Это доказывает, что визуальный анализ на основе структурного сходства и цветовых гистограмм дает качественный скачок в обнаружении фишинга.

Однако прототип имеет ограничения, которые необходимо учитывать:

- Синтетические данные: эксперимент проведен на искусственных изображениях, а не на реальных скриншотах. Баннеры, динамически подгружаемый контент, анимация могут снизить точность.
- Малое число эталонов: всего три бренда. Для промышленного применения требуются десятки или сотни эталонов (крупные банки, соцсети, маркетплейсы).
- Адаптивная верстка: мобильные скриншоты обрезаются, искажаются пропорции при ресайзе к квадрату 400x400.
- Динамический фишинг: страницы, генерируемые уникально для каждого пользователя, не могут быть обнаружены прямым сравнением с фиксированным эталоном. Необходим анализ устойчивых признаков (относительное расположение элементов) или глубокие нейросети.
- Производительность: сравнение одного скриншота с тремя эталонами занимает 0,15-0,2 секунд. Для браузерного расширения — это многовато; потенциальное решение — распараллеливание и оптимизация (NumPy, Cython) до 50-79 миллисекунд.

Разработан и экспериментально подтвержден метод визуального обнаружения фишинговых веб-страниц на основе комбинации SSIM и цветовых гистограмм. Прототип, реализованный на Python с использованием OpenCV, scikit-image, NumPy и Gradio, достиг точности 99,4% и полноты 100% на синтетическом наборе из 180 изображений трех популярных брендов. Метод может служить дополнительным проактивным слоем в антифишинговых системах, особенно для обнаружения свежих сайтов, еще не попавших в черные списки. Дальнейшая работа будет направлена на расширение эталонной базы. Учет адаптивной верстки и оптимизацию производительности.

Литература

1. Abid A. et al. Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild. *ArXiv.org*. URL: <https://arxiv.org/abs/1906.02569> (дата обращения: 19.05.2026).
2. Bradski G. The OpenCV Library. *Dr. Dobbs's Journal of Software Tools*. 2000; 25(120): 120–125.
3. Harris C. R. et al. Array programming with NumPy. *Nature*. 2020; 585(7825): 357–362.
4. Pedregosa F. et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. 2011; 12: 2825–2830.
5. scikit-image team. skimage.metrics – skimage 0.25.0 documentation. *scikit-image-org*. URL: <https://scikit-image.org/docs/0.25.x/api/skimage.metrics.html> (дата обращения: 19.05.2026).
6. Swain M. J., Ballard D. H. Color indexing. *International Journal of Computer Vision*. 1991; 7(1): 11–32.
7. The OpenCV Foundation. Enum opencv::imgproc::HistCompMethods // docs.rs. URL: https://docs.rs/opencv/0.53.2/x86_64-apple-darwin/opencv/imgproc/enum.HistCompMethods.html (дата обращения: 19.05.2026).
8. van der Walt S. et al. scikit-image: image processing in Python. *PeerJ*. 2014; 2: e453.
9. Wang Z., Bovik A. C., Sheikh H. R., Simoncelli E. P. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*. 2004; 13(4): 600–612.

10. Стремительный рост фишинга в Рунете удалось переломить // cisoclub.ru. URL: <https://cisoclub.ru/stremitelnyj-rost-fishing-v-runete-udalos-perelomit> (дата обращения: 15.05.2026).

A METHOD FOR DETECTING PHISHING WEB RESOURCES ON BASED ON THE ANALYSIS OF VISUAL SIMILARITY WITH THE APPLICATION SSIM AND COLOR HISTOGRAMS

Tsybikova Tuyana Sandalikovna

Candidate of Pedagogical Sciences, Associate Professor

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: cts2001@mail.ru

Dambaev Tsyren Chingisovich

student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: dambaev_2019@mail.ru

Abstract. The paper presents a method for detecting phishing web pages based on visual similarity analysis using SSIM and color histograms. Unlike traditional approaches that use blacklists or URL analysis, the proposed method compares a screenshot of a suspicious website with reference images of legitimate resources. Two metrics are combined: the Structural Similarity Index (SSIM), which measures the layout of elements, and the color histogram in the RGB space, which captures the replacement of logos and color schemes. The final score is calculated as a weighted sum $S=0.6*SSIM+0.4*H_{final}$. An experiment on a synthetic set of 180 images (three brands: Sberbank, VKontakte, and Google) showed an accuracy of 99.4% and a completeness of 100%, which significantly exceeds both URL analysis (70%) and the use of each metric individually. The prototype is implemented in Python using the OpenCV, scikit-image, NumPy, and Gradio libraries.

Keywords: phishing, computer vision, SSIM, color histogram, visual detection, anti-phishing, Python, OpenCV.

ПРОЕКТИРОВАНИЕ МНОГОФУНКЦИОНАЛЬНОГО КОМПЛЕКСА С СИСТЕМОЙ ОТОБРАЖЕНИЯ ИНФОРМАЦИИ НА БАЗЕ ЖИДКОКРИСТАЛЛИЧЕСКОГО МОДУЛЯ

© **Очирова Наталья Викторовна**

кандидат педагогических наук, старший преподаватель,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: onavi@rambler.ru

© **Жамбалов Анатолий Сергеевич**

Студент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: zambalov.29@gmail.com

Аннотация. В статье представлен значительный научный и практический интерес, поскольку идея разработки многофункционального комплекса с системой отображения информации на базе жидкокристаллического модуля, обладает инновационным потенциалом. Исследование обусловлена возможностью широкого практического применения в сфере маркетинга, информирования населения и в личных художественных целях. Данное устройство также является инструментом для повышения качества обучения, организации интерактивных занятий и проведения образовательных мероприятий.

Ключевые слова: комплекс, контроллер, проектирование, DD.3663LUA.A81.PA, прошивка, матрица, тюнер, обучение.

Актуальность исследования. В настоящее время отмечается рост популярности носимых электронных гаджетов, таких как умные часы и фитнес-трекеры, которые характеризуются компактностью и простотой в использовании. Это способствует их широкому распространению среди различных категорий пользователей. Основная функциональность этих устройств заключается в автоматизации определённых процессов и сборе данных, что направлено на повышение эффективности и оптимизацию повседневной деятельности человека [5].

В процессе разработки и сборки устройства были идентифицированы ключевые компоненты и дополнительные элементы. Основой для данного проекта была выбрана плата DD.3663LUA.A81.PA, представляющая собой современную и экономически выгодную модель контроллера (Рисунок 1).

Контроллер DD.3663LUA.A81.PA обладает широкими функциональными возможностями для разработки мультимедийного тюнера на основе матрицы ноутбука или монитора. Данное устройство может применяться не только в качестве телевизионного приемника, но и как экран для разнообразных мультимедийных приложений. Контроллер 3663 включает в себя аналого-цифровой интерфейс Touraine Rafael R842, поддерживающий стандарты T-2, DVB-T и DVB-C. Эти стандарты обеспечивают подключение устройства к различным источникам телевизионного сигнала, включая кабельное телевидение и радио-

вещание, предоставляя пользователю доступ к широкому ассортименту телевизионных и радиопрограмм. Кроме того, плата 3663 оснащена USB-портом, который позволяет подключать внешние USB-накопители. Это расширяет возможности воспроизведения мультимедийного контента, такого как фильмы и музыкальные композиции, без необходимости использования встроенных носителей устройства.

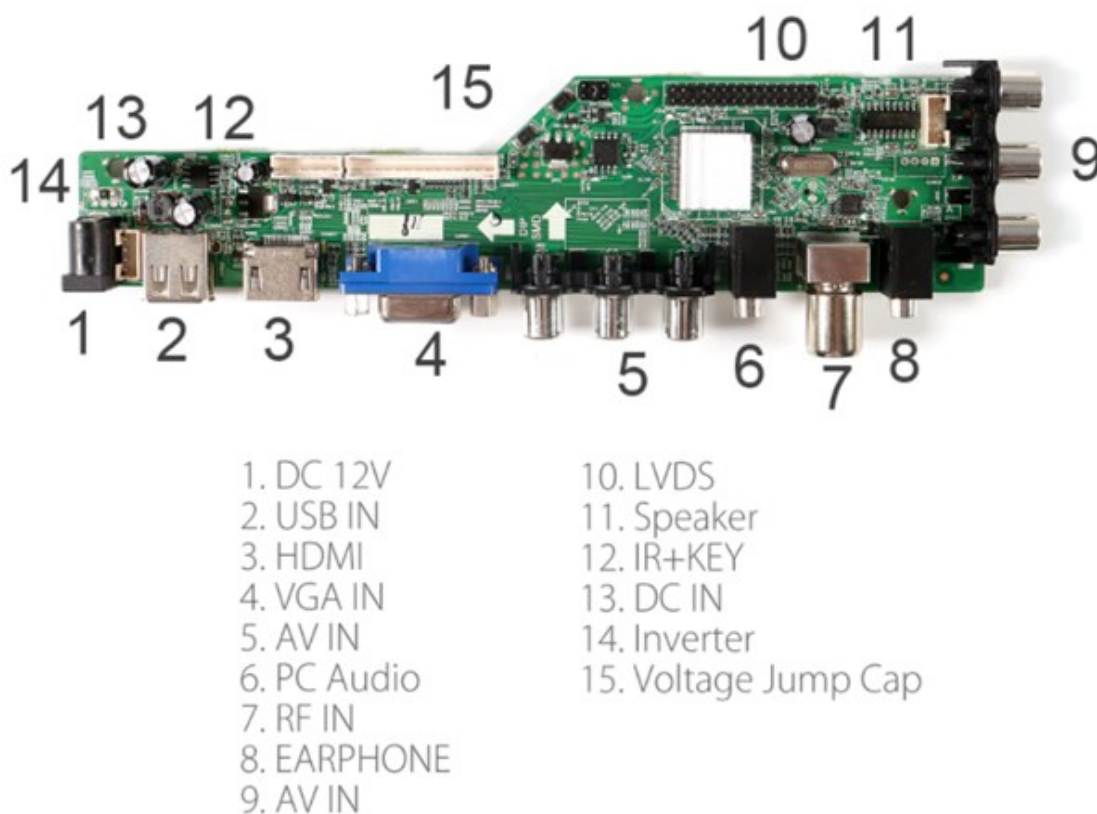


Рисунок 1. Контроллер DD.3663LUA.A81.PA

Инвертор представляет собой устройство, осуществляющее преобразование высокого напряжения, которое затем подаётся на матрицу. Данная микросхема выступает в качестве ключевого компонента, обеспечивающего подачу электропитания на матрицу.



Рисунок 2. Инвертор напряжения

Основным элементом системы управления является пульт. Однако также предусмотрена возможность управления с помощью клавиатуры. В комплект с клавиатурой входит датчик, обеспечивающий функционирование системы управления устройством. Данный датчик может быть подключён отдельно.



Рисунок 3. Клавиатура и пульт

Передача изображения от платы на матрицу осуществляется посредством кабеля LVDS. Существует несколько разновидностей кабелей LVDS, которые подбираются в зависимости от типа разъёма на матрице. Основной кабель LVDS соответствует разъёму, используемому в большинстве мониторов.



Рисунок 4. LVDS кабель

Для преобразования электрического сигнала в акустические волны используется компонент (Рисунок 12). Данный компонент оснащён специализированным разъёмом, предназначенным для подключения к плате с номером 3663.



Рисунок 5. Динамики

Для обеспечения корректной подачи питания необходимо использовать зарядное устройство, характеристики которого соответствуют следующим параметрам: выходное напряжение должно составлять 12 вольт и силу тока 4 ампера. Подходят зарядные устройства, изначально предназначенные для питания других электронных устройств, например, роутеров, ноутбуков и аналогичных гаджетов.



Рисунок 6. Блок питания

При подключении аккумуляторной батареи с напряжением 12 В происходит подача первичного постоянного электрического тока на вход платы скалера модели DD.3663LUA.A81.PA. На данном этапе осуществляется фильтрация входного тока с помощью входных конденсаторов. Далее электрический ток поступает на DC-DC преобразователь и ШИМ-контроллер платы. ШИМ-контроллер генерирует тактовые сигналы. После этого напряжение подаётся на инвертор напряжения (Inverter) — повышающий преобразователь, который встроен в плату или матрицу. Инвертор напряжения преобразует 12 В постоянного тока в переменное высокое напряжение в диапазоне 600–1500 В с частотой 40–100 кГц. Это напряжение используется для питания CCFL-подсветки. Также инвертор может преобразовывать входное напряжение в стабилизированное постоянное напряжение в диапазоне 12–30 В, которое

применяется для LED-подсветки матрицы Samsung. Одновременно с этими процессами логическая часть скалера осуществляет декодирование сигнала и вывод изображения через LVDS-шлейф на TFT-матрицу. При этом происходит управление яркостью подсветки через ШИМ-сигнал, который подаётся на инвертор.

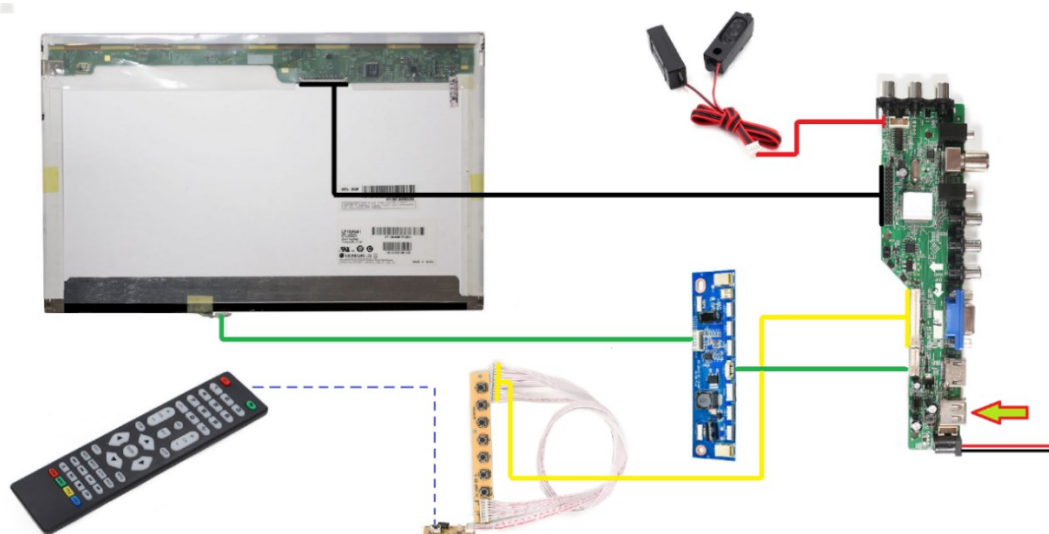


Рисунок 7. Схема подключения

Чёртеж каркаса многофункционального комплекса спроектирован для выполнения различных задач. Передняя часть каркаса предназначена для крепления LCD-матрицы монитора. Задняя часть предусмотрена для креплений лямок. Центральная часть служит для размещения и хранения различных вещей и аксессуаров, благодаря верхней части присутствует размещения входной зоны. Нижняя часть рюкзака отведена под аккумулятор и провода, что обеспечивает устойчивость комплекса и правильный центр тяжести за счёт своего расположения с наиболее тяжёлой зоной. Также дополнительные крепления обеспечивают жёсткость конструкции, предотвращая её от повреждений при использовании.

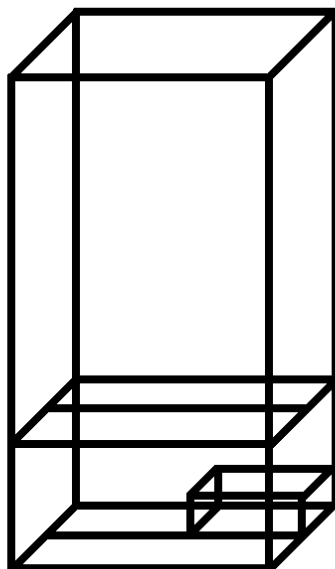


Рисунок 8. Схема каркаса

Программное обеспечение — это совокупность программ и данных, предназначенных для управления аппаратными средствами цифровой техники, автоматизации выполнения задач и обеспечения взаимодействия пользователя с вычислительной системой. Программное обеспечение включает в себя операционные системы, прикладные программы, системы управления базами данных, инструментальные средства для разработки программ и другие компоненты, необходимые для функционирования цифровой техники и решения различных задач пользователя.

Программное обеспечение (ПО) широко применяется в разнообразных технических устройствах, таких как телевизоры, смартфоны, планшеты, персональные компьютеры (ПК), носимые устройства (смарт-часы, фитнес-браслеты), бытовая техника (стиральные машины, холодильники, плиты), автоматизированные системы («умный дом», роботы), игровые консоли, медицинское оборудование (диагностические аппараты, носимые устройства для мониторинга здоровья), промышленное оборудование (станки, роботы-манипуляторы) и транспортные средства (автомобили, автобусы, поезда).

После выбора компонентов, необходимых для сборки комплекса, требуется осуществить установку программного обеспечения, гарантирующего корректную работу устройства. Для этого необходимо подобрать соответствующую модель «ПО» и установить её на USB-носитель. Файл после переименования должен иметь название LUA36A81 для последующей успешной установки. Далее необходимо подключить USB-накопитель и клавиатуру с встроенным светодиодом для индикации процесса установки, после этого требуется подача электрического тока на схему комплекса.

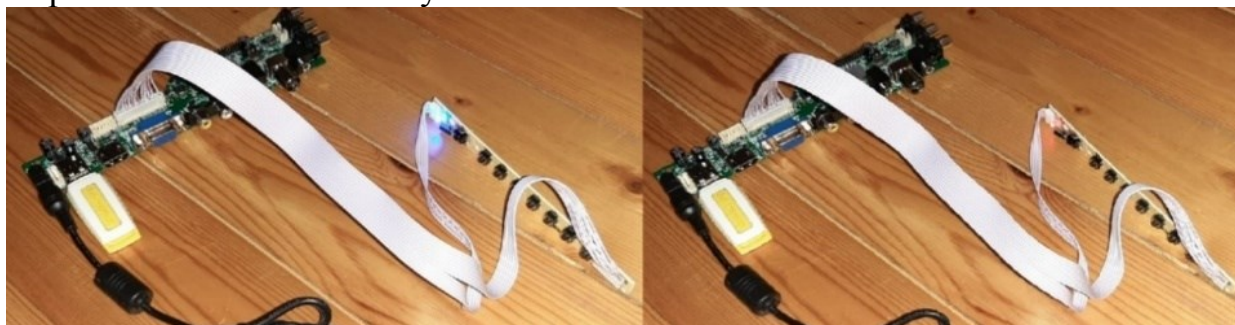


Рисунок 9. Установка ПО

Основной компонент системы — монитор, экран которого подключается к контроллеру для выполнения функций отображения информации и обеспечения корректной работы на рисунке 10. В распоряжении присутствует монитор Samsung S22D300, который выступает в качестве примера. Его основные характеристики представлены в таблице 1.



Рисунок 10. Монитор Samsung S22D300

Таблица 1

Характеристики монитора Samsung S22D300

Характеристики экрана	
Технология	TN+film
Диагональ	21.5 "
Разрешение (макс.)	1920x1080
Формат экрана	16:9
Яркость	200 кд/м2
Контрастность	600:1
Время отклика	5 мс
Угол обзора горизонтальный	90°
Угол обзора вертикальный	65°
Частота обновления	60 Гц
Способы подключения и соединения	
Интерфейсы / коннекторы	VGA (D-SUB)
Питание	
Тип блока питания	внешний
Потребляемая мощность (рабочая)	22 Вт
Потребляемая мощность (спящий режим)	0.3 Вт
Корпус	
Покрытие корпуса	глянцевое
Цвет	черный
Крепление VESA	отсутствует
Габариты и вес	
Размеры с подставкой (ШхВхГ)	511.6x385.2x177 мм
Размеры без подставки (ШхВхГ)	511.6x310.8x53.1 мм
Вес	2.85 кг
Вес (без подставки)	2.6 кг

В процессе тестирования монитора Samsung была осуществлена установка официальной прошивки DD.3663LUA.A82.PA, предоставленной производителем.

лем. При подключении питания устройство продемонстрировало полноценное функционирование и отсутствие ошибок. Ранее в процессе тестирования применялись прошивки DS.3663LUA.A81.PA, доступные в свободном доступе на различных интернет-ресурсах. Сложности в проведении предыдущих тестов были связаны с проблемами поиска соответствующего программного обеспечения.



Рисунок 11. Тестирование устройства

В ходе комплексного тестирования устройства было подтверждено выполнение всех заявленных базовых функций. Устройство успешно декодирует цифровой телевизионный сигнал и выводит его на встроенную LCD-матрицу с высоким уровнем визуального комфорта. Из 100 доступных каналов устройство идентифицировало и сохранило 63, что составляет 63% от общего числа. Это свидетельствует об отличной чувствительности тюнера в реальных условиях приёма, учитывая возможные помехи. Тестирование показало, что устройство стабильно работает, не перегревается, не теряет сигнал и обеспечивает все заявленные пользовательские сценарии без сбоев и критических ошибок.

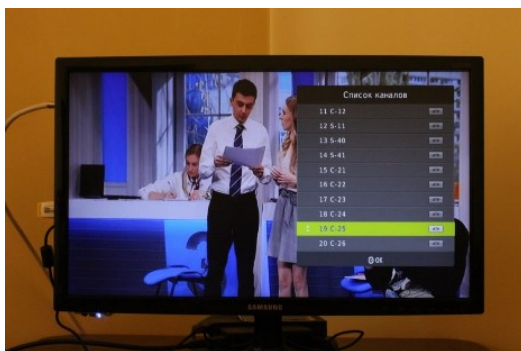


Рисунок 12. Тестирование кабельного телевидения

Разработан конструктивный каркас в соответствии с исходным чертежом. В процессе создания каркаса были проведены дополнительные работы по усилению конструкции. Данные доработки позволили повысить прочностные характеристики каркаса и обеспечить его надёжную эксплуатацию при минимальном весе в заданных условиях. В нижней части каркаса предусмотрено посадочное место под аккумуляторный блок для смещения центра тяжести. Экран крепится на задней плоскости каркаса лицом наружу, а электронные компоненты размещаются на задней части матрицы экрана, что обеспечивает их защиту от повреждений и доступность для обслуживания без изменения плоского профиля комплекса.



Рисунок 13,14. Каркас многофункционального комплекса и Настройка системы

Для создания специализированного многофункционального комплекса разработана конструкция внешнего кейса из натуральной кожи, которая обеспечивает защиту и поддержку внутреннего содержимого. Верхняя часть чехла оснащена молнией, обеспечивающей доступ в основное отделение для портативной вычислительной техники, такие как ноутбук, и других принадлежностей. Нижняя часть, прилегающая к спине, содержит дополнительную молнию, ведущую к изолированному отсеку для аккумулятора. Внутреннее пространство разделено на функциональные зоны с помощью интегральной перегородки, которая предотвращает механическое воздействие элементов на аккумулятор и жидкокристаллическую матрицу, закреплённую на внутренней поверхности кожи посредством клеевого соединения по всей площади комплекса. Для удобства переноски рюкзак оснащён плечевыми ремнями и дополнительной верхней ручкой.



Рисунок 15,16. Многофункциональный комплекс сзади и спереди

Основными преимуществами проектирования и сборки собственного многофункционального комплекса является, низкая себестоимость – можно самостоятельно собрать работающее устройство по минимальной цене. В таблице 2 представлены актуальные цены 2026 года по сборке данного устройства.

Таблица 2

Расчёт стоимости

Компоненты	Цена
Плата DD.3663LUA.A81.PA	1050 р.
Инвертор напряжения	230 р.
Клавиатура и пульт	150 р.
LVDS кабель	30 р.
Динамики	120 р.
Аккумулятор	1100 р.
Монитор	0 р. (списанный)
Отделка кожей	2000 р.
Итого	4680 р.

Таким образом, проведённые исследования и испытания показали, что разработанный и функционирующий образец многофункционального комплекса обладает низкой себестоимостью компонентов и высоким прикладным потенциалом в нескольких ключевых областях. В сфере маркетинга он может выступать в качестве мобильного носителя динамической рекламы и брендинга. В контексте информирования населения устройство способно транслировать визуальную и звуковую информацию в общественных пространствах. Кроме того, рюкзак может найти применение в личных художественных практиках, например, в качестве интерактивного медиаобъекта и средства перформанса. Экспериментальный опыт подтвердил функциональную надёжность устройства, включая стабильное отображение видеосигнала и наличие звука, а также экономическую доступность решения, что открывает перспективы для его дальнейшего тиражирования и адаптации под различные коммуникативные и творческие задачи.

Литература

1. Карякин В.Л. Цифровое телевидение: учебное пособие для вузов. 2-е изд., перераб. и доп. Москва: СОЛОН-ПРЕСС, 2013. 448 с.
2. Тюнин Н.А. Портативные ЖК телевизоры. Устройство и ремонт: учебное пособие для вузов. Москва: СОЛОН-ПРЕСС, 2010. 132 с.
3. Тюнин Н.А., Потапов П.Е. Телевизоры XXI века: учебное пособие для вузов. Серия «Ремонт», выпуск 89. Москва: СОЛОН-ПРЕСС, 2010. 144 с.
4. Романов Г.Е. Современные бюджетные и элитные телевизоры: учебное пособие для вузов. Москва: СОЛОН-ПРЕСС, 2011. 120 с.
5. Роберт И. В. Цифровая трансформация образования: вызовы и возможности совершенствования // Информатизация образования и науки. 2020. № 3(47). С. 3-16.
6. Филькина А. В., Камнева О. С. Профили цифровой грамотности школьников // Вестник Томского гос. ун-та. Философия. Социология. Политология. 2023. № 71. С. 204-214.

7. Беккужина Г. У. Интеграция цифровых технологий в обучение старшеклассников: условия и подходы // Вестник науки. 2025. № 5(86). С. 315-318
8. Гончаренко Л. П., Наумов С. Н., Шестакова А. А. Проблемы информатизации сельских школ в рамках национального проекта «Образование». // Информатика и образование. 2022. № 37(6). С. 12-21.
9. Загорюлько Ю. А., Савельева Е. В. Аппаратно-программные средства как ресурс цифровой трансформации школы // Информационные технологии в образовании : сб. науч. тр. Москва: ИИО РАО, 2022. С. 112–120.

DESIGN OF A MULTIFUNCTIONAL COMPLEX WITH AN INFORMATION DISPLAY SYSTEM BASED ON A LIQUID-CRYSTAL MODULE

Ochirova Natalya Viktorovna

Candidate of Pedagogical Sciences, Senior Lecturer,
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: onavi@rambler.ru

Zambalov Anatoly Sergeevich

Student
Dorzhi Banzarov Buryat State University
Russia, 670000, Ulan-Ude, Smolina str. 24a
E-mail: zambalov.29@gmail.com

Abstract: The article presents significant scientific and practical interest, since the idea of developing a multifunctional complex with an information display system based on a liquid crystal module has innovative potential. The research is conditioned by the possibility of wide practical application in the field of marketing, informing the public and for personal artistic purposes. This device is also a tool for improving the quality of education, organizing interactive classes and conducting educational events.

Keywords: complex, controller, design, DD.3663LUA.A81.PA, firmware, matrix, tuner, training.

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ ВЕДЕНИЯ ЭЛЕКТРОННОЙ МЕДИЦИНСКОЙ КАРТЫ РЕБЕНКА «ДНЕВНИК МАТЕРИ»

© **Жданов Иван Анатольевич**

учитель,
Улан-Баторский филиал РЭУ им. Г. В. Плеханова
Монголия, г. Улан-Батор, проспект Мира, 69
E-mail: ivan.zhdanov95@gmail.com

© **Шапеев Дмитрий Юрьевич**

преподаватель,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: shapeevdy@bsu.ru

© **Вампилова Эльвира Зоригтоевна**

студент,
Бурятский государственный университет имени Доржи Банзарова
Россия, 670000, г. Улан-Удэ, ул. Смолина, 24а
E-mail: vampilovaelvira@gmail.com

Аннотация. В статье рассматривается задача разработки мобильного приложения для структурированного сбора, хранения и визуализации медицинских данных ребенка. Исследуется подход, сочетающий использование локальной реляционной базы данных SQLite для обеспечения надежности хранения и нативных средств Android для генерации отчетов. Подробно описывается архитектура приложения, включающая модуль динамического ввода данных, подсистему асинхронной обработки с использованием ExecutorService и модуль визуализации динамики показателей на базе MPAndroidChart. Особое внимание уделено реализации умной группировки записей по датам в интерфейсе и алгоритму автоматической разбивки многостраничных PDF-документов. Приведены практические рекомендации по обеспечению отзывчивости пользовательского интерфейса при работе с большими массивами данных.

Ключевые слова: мобильная разработка, Android, электронная медицинская карта, Room Database, визуализация данных, генерация PDF, многопоточность, RecyclerView, компьютерное здоровье.

В предыдущих работах рассматривались общие подходы к цифровизации медицинских данных, однако большинство существующих решений либо требуют постоянного подключения к интернету, либо не предоставляют удобных инструментов для экспорта данных для лечащего врача. В данной статье описывается процесс создания автономного мобильного приложения на базе ОС Android, ориентированного на ведение электронного дневника здоровья ребенка.

Целью исследования является создание и практическая проверка методики разработки модульного мобильного приложения, обеспечивающего надежное локальное хранение данных, их наглядную визуализацию и бесшовный экспорт в универсальные форматы. Такой подход призван повысить удобство монито-

ринга состояния здоровья, обеспечить сохранность данных без зависимости от сетевой инфраструктуры и упростить коммуникацию между родителями и педиатром.

Актуальность работы определяется возрастающим спросом на персонализированные инструменты самоконтроля здоровья и необходимостью создания легковесных, энергоэффективных приложений, способных работать в условиях ограниченных ресурсов мобильного устройства.

Предлагаемое решение представляет собой модульное приложение, состоящее из четырех основных компонентов, что обеспечивает слабую связанность и высокую тестируемость кода:



Рисунок 1. Java класс

Модуль управления данными (Room Database). Реализует интерфейс для работы с локальной реляционной СУБД на базе SQLite. Обеспечивает запись результатов измерений температуры, роста, веса, фиксацию прививок, симптомов и визитов к специалистам. Для предотвращения блокировки главного потока все операции чтения и записи выполняются асинхронно с использованием `ExecutorService`.

Модуль динамического ввода данных. Предоставляет унифицированный интерфейс, который адаптивно изменяет набор полей ввода в зависимости от выбранной категории события. Это минимизирует когнитивную нагрузку на пользователя.

Модуль визуализации и аналитики. Интегрирует библиотеку `MPAndroidChart` для построения линейных графиков динамики температуры. Реализована автоматическая фильтрация данных, масштабирование осей и хронологическая сортировка точек для наглядного отображения трендов.

Модуль генерации отчетов. Отвечает за формирование структурированного документа формата PDF. Включает алгоритм автоматической разбивки на страницы при превышении лимита высоты страницы, а также функцию переноса длинных строк для корректного отображения развернутых рекомендаций врачей.

Переход от простого хранения в `SharedPreferences` к архитектуре Room потребовал внедрения паттерна репозитория и строгого разделения потоков вы-

полнения. Инициализация базы данных реализована по принципу Singleton с использованием `fallbackToDestructiveMigration()` для упрощения итеративной разработки схемы данных.

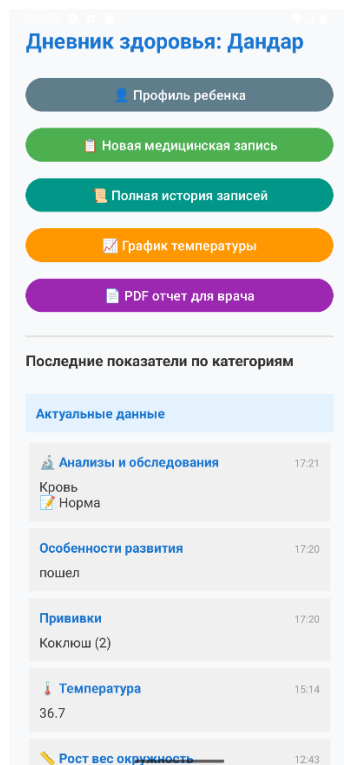


Рисунок 2. Главное меню

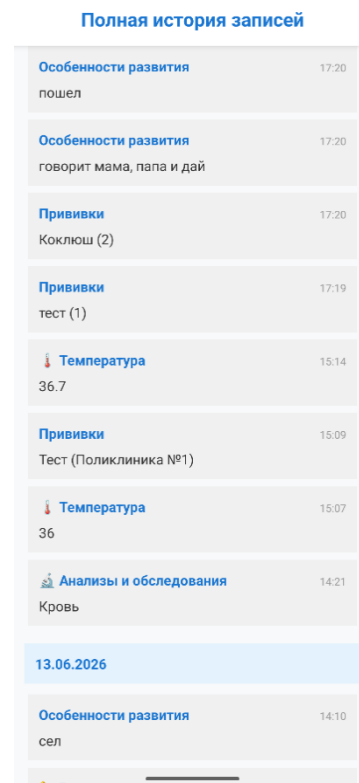


Рисунок 3. Полная история записей

Для отображения истории записей был применен адаптер `RecyclerView` с поддержкой множественных типов представлений. Это позволило в одном списке комбинировать элементы-заголовки и элементы-записи, что значительно улучшает читаемость хронологии без усложнения логики группировки на стороне базы данных.

Алгоритм работы системы генерации PDF: при инициации экспорта приложение запрашивает полный массив данных из `Room`. Специальный класс-обертка `PdfWriter` отслеживает текущую координату оси Y. Если добавление нового блока текста превышает пороговое значение, метод `checkBreak()` автоматически завершает текущую страницу, создает новую и сбрасывает координату. Для экспорта используется механизм `FileProvider`, обеспечивающий безопасную передачу URI файла внешним приложениям без запроса глобальных разрешений на чтение памяти.

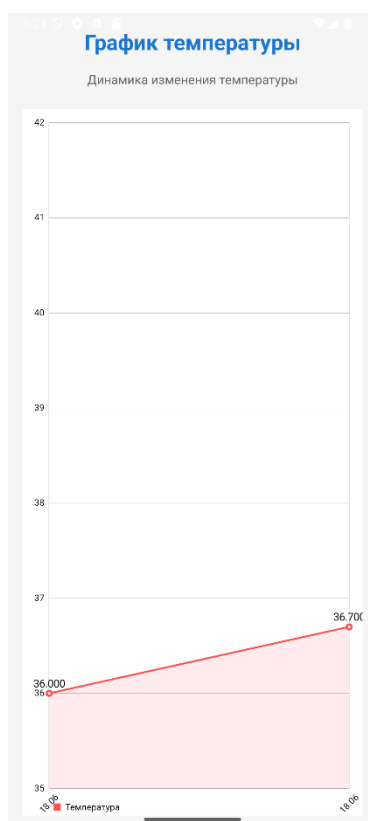


Рисунок 4. График температур

В статье представлена архитектура и практическая реализация мобильного приложения для ведения электронной медицинской карты ребенка. Разработанное приложение демонстрирует эффективность сочетания надежного локального хранения данных и нативных средств визуализации и экспорта.

Использование многопоточности позволило обеспечить плавную работу интерфейса при непрерывной обработке и сохранении данных. Модульная структура и отказ от тяжелых сторонних библиотек для генерации PDF обеспечили минимальный размер установочного файла и высокую скорость работы. Для дальнейшего развития приложения, можно рассмотреть интеграцию с облачными платформами для синхронизации данных между устройствами родителей и внедрение системы push-уведомлений о приеме лекарств.

Литература

1. Документация по архитектуре компонентов Android (Room). URL: <https://developer.android.com/training/data-storage/room> (дата обращения: 25.05.2026).
2. Арсланбаев И.Г., Шабеева А.Ф. Разработка мобильного приложения для подготовки к ЕГЭ по информатике. Стерлитамак: Стерлитамакский филиал БашГУ, [б. г.]. 5 с.
3. Назаров А. Д. Применение механизмов асинхронного программирования при разработке приложений на мобильных устройствах на базе ОС Android // Современные технологии в науке и образовании – СТНО-2020: сборник трудов III Международного научно-технического форума: в 10 томах. Рязань, 4–6 марта 2020 года. Рязань: Изд-во Рязанского гос. радиотехн. ун-та; BookJet, 2020. Т. 5. С. 223–226. EDN MPYDAG.
4. Загитова Ю. Д. Разработка мобильного приложения на системе ANDROID для наблюдения обострений бронхиальной астмы // Наука настоящего и будущего. 2021. Т. 1. С. 288–291. EDN GZFAIS.

5. Винников М. Д. Разработка базы данных для мобильного приложения на операционной системе Android // Научные известия. 2022. № 29. С. 39–42. EDN OOIKEY.

DEVELOPMENT OF A MOBILE APP FOR MANAGING A CHILD'S ELECTRONIC MEDICAL CARD "MOTHER'S DIARY"

Zhdanov Ivan Anatolyevich

Teacher

Ulaanbaatar Branch of the Plekhanov Russian University of Economics

Mongolia, Ulaanbaatar, 69 Mir Avenue

E-mail: ivan.zhdanov95@gmail.com

Shapeev Dmitry Yurievich

Teacher

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: shapeevdy@bsu.ru

Vampilova Elvira Zorigtoevna

student

Dorzhi Banzarov Buryat State University

Russia, 670000, Ulan-Ude, Smolina str. 24a

E-mail: vampilovaelvira@gmail.com

Abstract. The article considers the task of developing a mobile application for structured collection, storage and visualization of a child's medical data. The approach combining the use of a local relational database SQLite to ensure storage reliability and native Android tools for generating reports is investigated. The application architecture is described in detail, including a dynamic data input module, an asynchronous processing subsystem using ExecutorService and a module for visualization of indicators dynamics based on MPAndroidChart. Special attention is paid to the implementation of smart grouping of records by date in the interface and the algorithm for automatic splitting of multi-page PDF documents. Practical recommendations are provided for ensuring the responsiveness of the user interface when working with large data arrays.

Keywords: mobile development, Android, electronic medical record, Room Database, data visualization, PDF generation, multithreading, RecyclerView, computer health.

Научное издание

ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ
В ОБРАЗОВАНИИ, НАУКЕ И БИЗНЕСЕ

Материалы международной научно-практической конференции
(Улан-Удэ, 23–24 июня 2026 г.)

Редактор
Д. А. Габеева

Компьютерная верстка
Н. В. Очировой

Свидетельство государственной регистрации
№ 2670 от 11 августа 2017 г.

Подписано в печать 22.06.2026. Формат 60х84 1/8.
Уч.-изд. л. 21,74. Усл. печ. л. 22,22. Заказ 96.

Издательство Бурятского госуниверситета им. Д. Банзарова
670000, г. Улан-Удэ, ул. Ранжурова, 4
rio@bsu.ru