

ISSN 2227-6882

Министерство образования Республики Беларусь

Учреждение образования  
«Гомельский государственный университет  
имени Франциска Скорины»

**Новые математические методы  
и компьютерные технологии  
в проектировании, производстве  
и научных исследованиях**

Материалы XXIX Республиканской научной конференции  
студентов и аспирантов  
(Гомель, 23–25 марта 2026 года)

В двух частях

Часть 2

Гомель  
ГГУ им. Ф. Скорины  
2026

УДК 51:004:001.89

Сборник содержит материалы докладов XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях».

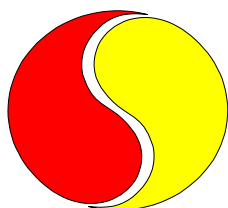
В издание включены материалы секций: «Современные информационные технологии» («Системное и программное обеспечение информационных технологий»), «Первые шаги в IT-сфере».

Адресуется студентам, магистрантам и аспирантам вузов, научным работникам.

**Редакционная коллегия:**

Л. Н. Марченко (главный редактор), А. Ф. Васильев,  
Д. С. Кузьменков, В. С. Смородин, О. В. Якубович,  
М. С. Белокурский, А. В. Воруев, Е. И. Сукач

© Учреждение образования  
«Гомельский государственный университет  
имени Франциска Скорины», 2026



## **СОВРЕМЕННЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ**

*Системное и программное  
обеспечение информационных технологий*

---

**А. Д. Абрамов, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

### **РАЗРАБОТКА ВЕБ-САЙТА ДЛЯ BACK-END РАЗРАБОТЧИКОВ “ROAD MAP” С ИСПОЛЬЗОВАНИЕМ DJANGO**

В условиях активного развития информационных технологий профессия программиста становится одной из наиболее востребованных. Особый интерес представляет направление back-end разработки, связанное с созданием серверной логики веб-приложений, обработкой запросов и взаимодействием с базами данных. Однако начинающие специалисты часто сталкиваются с проблемой отсутствия чёткой структуры обучения и понимания последовательности изучения необходимых технологий.

В связи с этим актуальной является разработка специализированного веб-сайта формата “Road Map”, представляющего собой интерактивный путеводитель по этапам становления back-end разработчика. Такой ресурс позволяет систематизировать знания, выделить ключевые технологии и выстроить логическую последовательность их освоения, выступая удобным ориентиром для студентов и людей, изучающих программирование самостоятельно.

При разработке веб-сайта использован фреймворк Django [1], реализованный на языке программирования Python. Django предоставляет встроенные средства для создания серверной логики, маршрутизации запросов, работы с базами данных и административной панели, что позволяет разрабатывать надёжные и масштабируемые веб-приложения в короткие сроки. Разработанный веб-сайт “Road Map” ориентирован на наглядное и структурированное представление учебных материалов.

Реализовано разделение технологий по уровням сложности, навигация по тематическим разделам и возможность редактирования структуры через административную часть. Пользовательский интерфейс разработан с учётом адаптивного отображения на различных устройствах. В рамках всего вышесказанного следует, что создаваемый веб-сайт направлен на упрощение процесса обучения и формирование целостного представления о профессиональном пути в области серверной разработки.

### **Литература**

1 Головатый, А. Подробное руководство / А. Головатый, Дж. Каплан-Мосс, пер. с. англ. – М.: Символ-Плюс, 2010. – 560 с.

**Н. В. Аксамит**

*(ГрГУ имени Янки Купалы, Гродно)*

## **АНАЛИЗ ПРИМЕНИМОСТИ ТЕХНОЛОГИЙ WORKFLOW-АВТОМАТИЗАЦИИ В СИСТЕМЕ АНАЛИЗА МЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ**

Реализация логики оркестрации исключительно на уровне прикладного кода усложняет сопровождение системы. В качестве альтернативного подхода рассматривается применение технологий workflow-автоматизации – специализированных средств формального описания и исполнения многоэтапных процессов.

Подсистема приёма данных взаимодействует с различными источниками (диагностическое оборудование, PACS-серверы, архивные хранилища), при этом сценарии извлечения и предварительной обработки могут существенно различаться в зависимости от интегрируемой системы. Наличие множества конфигурируемых и расширяемых сценариев делает данный узел подходящим для применения workflow-автоматизации, позволяющей декларативно описывать процессы интеграции и модифицировать их без изменения прикладной логики.

Процесс анализа исследования включает нормализацию данных, определение типа исследования, выбор набора моделей, последовательный или параллельный запуск нейронных сетей, агрегацию результатов и формирование отчета. Он характеризуется многоэтапностью, ветвлениями, повторными попытками при сбоях и

необходимостью аудита выполнения. При высокой вычислительной нагрузке и наличии долгоживущих процессов более обоснованным является использование отказоустойчивых и масштабируемых платформ оркестрации, обеспечивающих мониторинг, трассировку и сохранение состояния выполнения.

Таким образом, применение технологий workflow-автоматизации позволяет формализовать процессы интеграции и анализа медицинских изображений, повысить управляемость архитектуры и обеспечить её расширяемость при сохранении производительности критически нагруженных компонентов.

### Литература

1 Фаулер, М. Архитектура корпоративных программных приложений / М. Фаулер. – М.: Вильямс, 2019. – 544 с.

**К. М. Андросенко, Е. Ю. Кузьменкова**  
(ГГУ имени Ф. Скорины, Гомель)

### РАЗРАБОТКА САЙТА-ВИЗИТКИ ДЛЯ СПА-САЛОНА

В современном мире, где происходит всеобщая интеграция сети Интернет во все сферы жизнедеятельности человека, появление компаний в цифровой среде является не просто преимуществом, а необходимым условием для успешного развития бизнеса. Особенно это актуально для сферы услуг, где люди ищут специалиста быстро: решение принимается за пару минут после просмотра результатов поиска в Яндексe или ленты в соцсетях. Одним из наиболее эффективных инструментов представления бизнеса в сети является сайт-визитка. Главная цель такого сайта – создать положительный образ компании и обеспечить потенциальным клиентам удобную обратную связь. По сути, это цифровой офис, который работает круглосуточно и доступен из любой точки мира.

Для разработки сайта-визитки СПА-салона были выбраны технологии HTML, CSS и JavaScript. Этот выбор обусловлен несколькими причинами. HTML отвечает за структуру страницы. С его помощью создаются заголовки, тексты, списки услуг и форма обратной связи. Другими словами, HTML – это основа, на которой держится все содержимое сайта. CSS отвечает за внешний вид. Для СПА-салона

Материалы XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 23–25 марта 2026 г.

важно передать атмосферу уюта и расслабления, поэтому с помощью CSS задаются цвета, шрифты, отступы и фон. Также CSS позволяет сделать сайт адаптивным – чтобы он одинаково хорошо смотрелся на компьютере, телефоне или другом устройстве. JavaScript отвечает за интерактивность и оживляет сайт. При использовании современных сайтов пользователи ожидают не просто удобный пользовательский интерфейс, но и различные дополнительные возможности: открытие меню по нажатию кнопки, проверка ошибки заполнения формы до отправки, подсказки при наведении и другое.

В результате выполнения работы был разработан сайт-визитка для СПА-салона, который позволяет в лаконичной и удобной форме донести до клиентов ключевую информацию о компании и услугах, которые она предоставляет. Для индустрии красоты и здоровья важно не только проинформировать клиента, но и вызвать у него эмоциональный отклик, желание посетить салон. Готовый сайт решает обе задачи: он содержит полную информацию об услугах, ценах и контактах, а визуальное оформление создает нужную атмосферу.

**К. В. Андрусенко, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА FRONT-END ЧАСТИ ОБУЧАЮЩЕЙ ПЛАТФОРМЫ С ИНТЕГРАЦИЕЙ ФУНКЦИЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА**

Архитектура Front-end части представляет собой модульное одностраничное приложение (Single Page Application), построенное с использованием фреймворка Angular. Приложение реализовано в соответствии с принципами компонентного подхода и разделения ответственности (Separation of Concerns), что обеспечивает масштабируемость, повторное использование кода и удобство сопровождения.

При разработке использованы: HTML5, CSS3 и JavaScript, обеспечивающие структуру, стилизацию и программную логику интерфейса соответственно. Для организации реактивного взаимодействия между компонентами и асинхронной обработки данных применяется библиотека RxJS. При проектировании системы выделяются следующие основные функциональные модули.

«Интерфейс пользователя» – обеспечивает взаимодействие обучающегося с платформой. Предполагается реализация таких возможностей, как регистрация и авторизация пользователей, просмотр учебных материалов, прохождение тестов, загрузка заданий, отслеживание прогресса обучения и отображение рекомендаций.

«Модуль управления состоянием» – реализует централизованное хранение и обработку состояния приложения. Использование реактивного подхода на базе RxJS позволяет обрабатывать потоки данных, управлять событиями, обеспечивать подписку на изменения и синхронизацию интерфейса с сервером.

«Модуль интеграции искусственного интеллекта» – обеспечивает взаимодействие с Back-end-сервисами ИИ через RESTful API. Предполагается наличие таких функций, как: интеллектуальные рекомендации учебных материалов; автоматическая генерация тестовых заданий; анализ ответов пользователя; чат-ассистент для поддержки обучения; адаптивная подстройка сложности контента.

Взаимодействие с серверной частью осуществляется посредством HTTP-запросов к REST API. Передача данных происходит в формате JSON. Обработка ответов сервера, управление ошибками и повторные запросы реализуются с использованием механизмов RxJS.

**Т. А. Базар**

*(ГрГУ имени Янки Купалы, Гродно)*

## **О РАЗРАБОТКЕ МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ КИНОТЕАТРА**

В условиях цифровизации и роста конкуренции на рынке развлечений создание современного мобильного приложения является важным инструментом повышения качества обслуживания и вовлечённости аудитории кинотеатров.

Однако разработка такого приложения сопряжена с рядом задач: интеграцией с различными системами (расписание, бронирование, платежи), обеспечением высокой производительности, отзывчивостью интерфейса и адаптацией к изменяющимся требованиям пользователей. Для решения этих задач была выбрана архитектура MVVM (Model-View-ViewModel), обеспечивающая чёткое разделение логики представления, бизнес-логики и данных. В качестве

Материалы XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 23–25 марта 2026 г.

технологической основы использовались Swift и фреймворк SwiftUI, что позволило реализовать декларативный подход к построению интерфейсов, обеспечить реактивность и высокую модульность приложения.

При разработке сетевого слоя применялись URLSession и библиотека Alamofire для выполнения асинхронных запросов к серверу, а также механизмы кэширования для оптимизации загрузки данных. Особое внимание было уделено безопасности: передача конфиденциальных данных осуществляется по протоколу HTTPS, а аутентификация пользователей организована с использованием Supabase. Созданное приложение предоставляет пользователям возможность просмотра афиши, трейлеров, рейтингов, получения персонализированных уведомлений и удобного бронирования билетов. Оно также открывает для кинотеатра новые возможности по повышению лояльности клиентов и увеличению продаж за счёт системы рекомендаций и аналитики.

Таким образом, использование современных подходов и технологий в разработке мобильных приложений для кинотеатров позволяет не только удовлетворить запросы пользователей, но и создать конкурентное преимущество на рынке цифровых услуг.

**В. В. Балабан, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ВИЗУАЛЬНОЙ ЧАСТИ ИГРОВОГО ПРИЛОЖЕНИЯ “SKY FLAPPER” В СРЕДЕ UNITY**

Важным аспектом разработки игрового приложения “Sky Flapper” стала проработка визуальной составляющей, направленной на создание целостного и интуитивно понятного пользовательского опыта. Визуальная структура игры построена на системе экранов и меню, обеспечивающих логичную навигацию и последовательное погружение игрока в игровой процесс. Главное меню служит отправной точкой взаимодействия, предоставляя доступ к ключевым разделам приложения: запуску игры и внутриигровому магазину. Внутриигровой магазин был реализован как отдельный интерфейсный экран, содержащий коллекцию скинов персонажа.

Каждый скин представлен визуально и сопровождается кнопкой покупки, активируемой при наличии достаточного количества

игровой валюты. Такая реализация не только расширяет возможности персонализации, но и усиливает мотивацию пользователя к повторному прохождению уровней. Каждый уровень обладает уникальным визуальным оформлением, включая индивидуальный задний фон и стилистические отличия, что повышает разнообразие и удержание внимания игрока. В процессе прохождения уровней предусмотрены динамические интерфейсные элементы, отображающие результат игрового взаимодействия. При успешном завершении уровня появляется окно победы, визуально фиксирующее достижение игрока. В случае столкновения персонажа с препятствием, таким как труба или другая поверхность, игровой процесс останавливается. Дополнительно реализована система визуального поощрения: за достижение определённых порогов очков игрок получает монеты различной редкости. Чем выше итоговый результат, тем более редкая и ценная монета начисляется, что усиливает элемент прогрессии и повышает вовлечённость пользователя.

Таким образом, визуальная часть “Sky Flapper” ориентирована на сочетание функциональности и эстетики, обеспечивая удобную навигацию, визуальное разнообразие уровней и четкую систему обратной связи. Реализованные решения демонстрируют возможности Unity в создании структурированного пользовательского интерфейса и визуально привлекательных игровых приложений.

**А. П. Барабанов, Е. М. Березовская**  
(ГГУ имени Ф. Скорины, Гомель)

## **СОЗДАНИЕ ЧАТ-БОТА ДЛЯ УПРАВЛЕНИЯ СЕРВЕРОМ DISCORD С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА ПРОГРАММИРОВАНИЯ PYTHON И API DISCORD**

Современное образование характеризуется активной интеграцией цифровых платформ в учебный процесс. Мессенджеры и коммуникационные сервисы, в частности Discord, всё чаще используются как среда для взаимодействия студентов и преподавателей благодаря гибкой канальной структуре, поддержке реального времени и возможности создания программных расширений – ботов.

Однако стандартный функционал Discord не ориентирован на выполнение специфических образовательных задач, таких как сбор обратной связи, учёт посещаемости или автоматизированное администрирование учебных групп. Исследование в работе было направлено на разработку чат-бота, автоматизирующего администрирование Discord-сервера и адаптированного для использования в образовательном процессе. Цель работы – проектирование и разработка программного обеспечения на языке Python с использованием Discord API, автоматизирующего административные функции на сервере (модерация, управление ролями, реагирование на действия пользователей), а также определение сценариев его применения в обучении. Для разработки применялся язык Python с использованием асинхронной библиотеки `py-cord` (либо `discord.py`), что обеспечивало эффективную обработку событий и взаимодействие с Discord API.

Разработанный чат-бот представляет собой инструмент, расширяющий функциональные возможности Discord для решения конкретных педагогических задач: автоматизация управления учебной группой, проведение интерактивных опросов и сбора обратной связи, учёт посещаемости и активности, поддержка асинхронного обучения и обратной связи. Внедрение разработки позволит снизить административную нагрузку на преподавателя, повысить вовлечённость студентов за счёт интерактивности и оперативной обратной связи, а также организовать сбор аналитических данных для улучшения учебного процесса. Перспективы дальнейшего развития включают создание веб-интерфейса или мобильного приложения для управления ботом без использования текстовых команд, а также интеграцию с системами управления обучением (LMS).

**М. Б. Бельский, Д. С. Кузьменков**  
(ГГУ им Ф. Скорины, Гомель)

## **РАЗРАБОТКА ГЕЙМПЛЕЯ ИГРОВОГО ПРИЛОЖЕНИЯ 2D-ПЛАТФОРМЕРА В СРЕДЕ UNITY**

Последние десятилетия индустрия видеоигр демонстрирует стремительное развитие, глубоко интегрируясь в современную культуру, технологические процессы и глобальный бизнес.

В контексте этого роста 2D-платформеры остаются одним из наиболее востребованных и стабильных жанров. Популярность данного направления обусловлена сочетанием простоты освоения для конечного пользователя, динамичного геймплея и практически неограниченных возможностей для креативной реализации идей разработчика.

“Sky Flapper” – это игровое приложение, разработанное в среде Unity на языке C#. В игре присутствует динамичный геймплей, интуитивно понятный интерфейс. Для этого используются встроенные компоненты среды Unity, например, Rigidbody2D, который отвечает за логику прыжков, манипулируя вектором силы.

Одной из важных частей игрового приложения стала разработка внутриигровой валюты, которую можно получить, проходя через трубы, и которую впоследствии можно использовать во внутриигровом магазине. Эта валюта даёт возможность пользователю покупать скины, тем самым мотивируя его играть, чтобы получить тот скин, который нравится больше всего.

Таким образом “Sky Flapper” является примером эффективного сочетания классического геймплея и современных методов вовлечения игрока. Дальнейшие планы по развитию приложения включают разработку уровней и внедрение визуальной части для них. Разработанное игровое приложение демонстрирует Unity и C# как мощные и современные инструменты разработки, особенно в игровой индустрии.

**Г. А. Березин**

*(МГПУ имени И. П. Шамякина, Мозырь)*

**РАЗРАБОТКА МОБИЛЬНОГО  
ПРИЛОЖЕНИЯ-ПУТЕВОДИТЕЛЯ  
ПО РЕСПУБЛИКЕ БЕЛАРУСЬ  
НА ПЛАТФОРМЕ UNITY**

В Программе деятельности Правительства Республики Беларусь на 2025-2029 годы поставлена задача сделать туризм национальным проектом и максимально реализовать его экономический потенциал [1]. Однако не у всех есть возможность путешествовать – по финансовым причинам, из-за расстояний или состояния здоровья.

Кроме того, в системе образования не хватает современных визуальных форматов, создающих эффект присутствия.

BYWalk – это мобильное приложение-путеводитель нового поколения по Беларуси, которое объединяет 3D/360-туры, интерактивную карту, маршруты, аудиогид и «живой» каталог достопримечательностей в одном месте (рисунок 1).



Рисунок 1 – Интерфейс приложения BYWalk

Это продукт, который может стать цифровой инфраструктурой внутреннего туризма: удобно, красиво, технологично, с сильной образовательной составляющей.

### Литература

1 Пирожник, И. И. Современные тенденции развития туризма в Республике Беларусь: цифровизация, устойчивость, кластеризация / И. И. Пирожник, Н. Н. Бамбиза // Проблемы и перспективы развития туризма в странах с переходной экономикой. – 2022. – С. 112–118.

**А. В. Бондарева**

*(ГГУ имени Ф. Скорины, Гомель)*

### РАЗРАБОТКА DESKTOP-ПРИЛОЖЕНИЯ ДЛЯ УЧЁТА ЛИЧНЫХ ФИНАНСОВ С ИСПОЛЬЗОВАНИЕМ C# И SQLITE

В современной экономической реальности управление личными финансами становится критически важным навыком. Постоянный рост цен и необходимость планирования бюджета требуют доступных инструментов для учёта доходов и расходов. Работа посвящена разработке desktop-приложения «MyFinance» – автоматизированного рабочего места для комплексного учёта личных финансов.

Приложение реализовано на платформе .NET Framework с использованием языка C# и Windows Forms. Выбор данной технологии обусловлен высокой скоростью разработки и простотой развёртывания. В качестве встраиваемой СУБД выбрана SQLite, что обеспечило автономность приложения и надёжность благодаря ACID-транзакциям. Для проектирования базы данных использовался инструмент DB Browser for SQLite.

Архитектура приложения включает три основные сущности: счета (Account), категории (Category) и операции (Operation). Ключевой особенностью стало вынесение бизнес-логики на уровень базы данных в виде SQL-триггеров, автоматически пересчитывающих баланс счетов при добавлении, изменении или удалении операций. Это гарантирует целостность данных при любых действиях пользователя.

Пользовательский интерфейс реализован в виде главной информационной панели, который содержит выбор счёта, фильтрацию по периодам (день, неделя, месяц, год), отображение текущего баланса, суммы доходов и расходов. История операций представлена в двух ListView, реализованы круговые диаграммы для визуализации распределения доходов и расходов по категориям. Дополнительные формы обеспечивают добавление и редактирование операций.

В результате создано полнофункциональное приложение, автоматизирующее рутинные расчёты и предоставляющее пользователю инструменты для анализа финансов. Интуитивно понятный интерфейс для пользователей с возможностью связаться с разработчиком, при возникновении проблем. Приложение может применяться для ведения домашней бухгалтерии, планирования бюджета и в учебных целях.

**Ф. С. Боярин**

*(ГрГУ имени Янки Купалы, Гродно)*

## **РЕАЛИЗАЦИЯ РАБОТЫ МНОГОФУНКЦИОНАЛЬНОГО МЕССЕНДЖЕРА В РЕЖИМЕ ОФФЛАЙН**

Современные мессенджеры стали неотъемлемой частью повседневной коммуникации, однако их функциональность критически зависит от наличия стабильного интернет-соединения. В условиях

нестабильной связи или её отсутствия пользователи сталкиваются с ограничениями в работе приложений. Целью данной работы является представление концепции реализации оффлайн-режима для многофункционального мессенджера, обеспечивающего бесперебойную работу независимо от состояния сетевого подключения.

Оффлайн-режим мессенджера должен решать следующие задачи: обеспечение возможности создания и редактирования сообщений без подключения к сети, локальное хранение и синхронизация данных, сохранение истории переписки и медиафайлов, автоматическая отправка накопленных сообщений при восстановлении соединения.

Для реализации оффлайн-режима необходимы следующие функциональные компоненты: локальная база данных для хранения сообщений, контактов и медиафайлов, очередь отправки для накопления исходящих сообщений, механизм синхронизации данных между локальным хранилищем и сервером, система разрешения конфликтов при одновременном редактировании данных на разных устройствах, кэширование часто используемых данных для быстрого доступа.

При проектировании оффлайн-функциональности учитываются следующие технические аспекты: эффективное управление локальным хранилищем с учетом ограниченных ресурсов мобильных устройств, реализация алгоритмов сжатия и оптимизации хранения медиаконтента, обеспечение целостности данных при переключении между онлайн и оффлайн-режимами, индикация статуса сообщений (создано локально, в очереди отправки, доставлено, прочитано).

Ключевой особенностью архитектуры является использование паттерна «оффлайн-первый» (offline-first), при котором все операции сначала выполняются локально, а затем синхронизируются с сервером. Это обеспечивает мгновенный отклик интерфейса и бесшовный пользовательский опыт независимо от качества соединения.

Реализация полноценного оффлайн-режима в мессенджере является важным направлением развития коммуникационных приложений, позволяющим обеспечить непрерывность работы пользователей в любых условиях. Успешная реализация требует комплексного подхода к проектированию архитектуры, оптимизации хранения данных и эффективных алгоритмов синхронизации.

**Ф. С. Боярин, Г. С. Дудко**  
(ГрГУ имени Янки Купалы, Гродно)

## **СРАВНИТЕЛЬНЫЙ АНАЛИЗ ФУНКЦИОНАЛА ВЕБ-ВЕРСИИ И МОБИЛЬНОЙ ВЕРСИИ МЕССЕНДЖЕРА TELEGRAM**

Мессенджер Telegram предлагает пользователям два основных способа доступа: веб-версию и нативное мобильное приложение. Тестирование веб-клиента и мобильных приложений под управлением iOS и Android позволило выявить ряд существенных функциональных расхождений.

Мобильная версия обеспечивает прямую интеграцию с камерой и микрофоном, что позволяет записывать видео и отправлять голосовые сообщения «на лету». Веб-версия работает только с готовыми файлами с устройства. Функция геолокации в реальном времени, доступная в мобильном приложении, в вебе отсутствует полностью – передача координат возможна только как статический «пин». Аппаратное ограничение браузеров не позволяет веб-версии совершать и принимать звонки в фоновом режиме, тогда как мобильное приложение корректно работает в фоне.

Push-уведомления в веб-версии зависят от активности браузера и настроек энергосбережения операционной системы. При закрытом браузере или переводе компьютера в сон уведомления не доставляются. Мобильное приложение гарантирует получение сообщений независимо от того, активно приложение или нет.

Наиболее критичное различие выявлено в области безопасности. Веб-версия Telegram не поддерживает секретные чаты со сквозным шифрованием. Доступ к контактам устройства в вебе невозможен без предварительного импорта или сканирования QR-кода, мобильное приложение имеет полную интеграцию с телефонной книгой.

Веб-версия оптимизирована для работы с клавиатурой и мышью. Однако функции редактирования изображений в ней ограничены базовыми инструментами, а мобильное приложение предлагает расширенный редактор с фильтрами и обрезкой. Управление большими группами и каналами в вебе проще за счет наличия полноценного интерфейса, аналогичного десктопным приложениям.

Таким образом, веб-версия является усеченным клиентом, в котором доступны базовые функции обмена сообщениями и уступает мобильному приложению в использовании аппаратных возможностей, фоновой работе и безопасности.

**А. М. Булкина, И. А. Соболев**  
(ГГУ имени Ф. Скорины, Гомель)

## **ПРИМЕНЕНИЕ ASP.NET CORE ДЛЯ РАЗРАБОТКИ ИНТЕРНЕТ-МАГАЗИНА ТОВАРОВ ДЛЯ ДОМАШНИХ ЖИВОТНЫХ**

Разработка веб-приложения интернет-магазина PetShop направлена на создание современной платформы для онлайн-продажи товаров для домашних животных. Ключевой задачей проекта является обеспечение пользователям простого и удобного доступа к каталогу продукции, возможности формирования заказов и отслеживания их статуса, а также предоставление администраторам эффективных инструментов управления всем спектром бизнес-процессов – от контроля ассортимента до обработки заказов.

В рамках работы реализовано полнофункциональное приложение на основе технологического стека Microsoft. Архитектура решения построена на платформе ASP.NET Core с использованием шаблона проектирования Model-View-Controller (MVC), что обеспечивает модульность, масштабируемость и поддержку кода. Для взаимодействия с данными используется Entity Framework Core – современный ORM-фреймворк, позволяющий работать с реляционной базой данных MS SQL Server через объектную модель C#. Пользовательский интерфейс разработан с применением Razor Pages, что гарантирует адаптивность и кроссбраузерную совместимость.

Функциональность приложения разделена по ролям. Пользователи могут регистрироваться в системе, просматривать каталог товаров с фильтрацией по категориям и типам животных, добавлять товары в корзину, оформлять заказы с указанием адреса доставки и отслеживать историю своих покупок. Особенностью реализации является динамическое обновление корзины через AJAX-запросы и автоматическое резервирование товаров на складе при оформлении заказа.

Административная часть включает панель управления с возможностью полного контроля над контентом магазина. Администраторы могут выполнять CRUD-операции с товарами, управлять категориями, просматривать все заказы и изменять их статусы. Доступ к административным функциям защищен проверкой роли пользователя, хранящейся в сессии.

**А. С. Бурикин, Т. В. Процкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **СОЗДАНИЕ ОНЛАЙН-МАГАЗИНА ВИНИЛОВЫХ ПЛАСТИНОК С ИСПОЛЬЗОВАНИЕМ SPRING FRAMEWORK**

**Актуальность и проблематика** Проектирование высокопроизводительных, безопасных и легко масштабируемых веб-площадок остается одной из ключевых потребностей современной индустрии электронной коммерции. Опираясь на актуальные IT-методологии и современные подходы к программированию, в данном исследовании описывается процесс создания интернет-магазина для реализации виниловых пластинок с применением инструментов Spring Framework.

**Архитектурные решения** Центральным вектором работы выступает построение надежной многоуровневой архитектуры приложения. Значительное внимание уделяется конструированию REST API для бесшовной интеграции платформы с внешними сервисами, а также внедрению механизмов защиты конфиденциальной информации.

**Технологический стек** Для реализации поставленных задач был выбран следующий набор технологий: **Spring Boot** – выступает в качестве программного фундамента для быстрой настройки и запуска основного серверного приложения [1]; **Spring Data JPA** – применяется для эффективного взаимодействия приложения с базой данных [1]; **PostgreSQL** – используется в качестве надежной реляционной СУБД для хранения информации, при этом структура базы спроектирована с учетом всех необходимых связей между бизнес-сущностями; **Spring Security** – отвечает за разграничение прав доступа и комплексную защиту пользовательских данных от несанкционированного вмешательства; **Thymeleaf** – задействован для

Материалы XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 23–25 марта 2026 г.

генерации пользовательского веб-интерфейса, предоставляя посетителям интуитивно понятный и комфортный опыт взаимодействия с витриной.

**Функциональные возможности платформы** Разработанная система предназначена для автоматизации ключевых бизнес-процессов магазина. Она разделяет функционал по ролям: административная панель предоставляет инструменты для редактирования каталога товаров, в то время как функционал для покупателей включает возможности оформления заказов и мониторинга их текущего статуса.

**Обеспечение качества (Тестирование)** С целью повышения отказоустойчивости сервиса и минимизации вероятности возникновения багов, программный код системы прошел процедуру тестирования с использованием библиотек JUnit и Mockito [1].

### Литература

1 Шилдт, Г. Ш57 Java 8. Полное руководство / Г. Шилдт. – 9-е изд.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2015. – 1376 с.: ил. – Парал. тит. англ. ISBN 978-5-8459-1918-2 (рус.)

**А. В. Ваденкова, Е. Ю. Кузьменкова**  
(ГГУ имени Ф. Скорины, Гомель)

## РАЗРАБОТКА ХРАНИЛИЩА ДАННЫХ ДЛЯ ОБРАБОТКИ BIG DATA С ИСПОЛЬЗОВАНИЕМ DWH, APACHE SPARK И AIRFLOW

Современные системы анализа данных требуют end-to-end решений, в которых объединены сбор, хранение, обработка и визуализация больших объёмов информации. Важную роль играет слоистая архитектура хранилища данных и использование специализированных инструментов для оркестрации и распределённых вычислений.

Цель работы – разработка прототипа хранилища данных и аналитического конвейера, реализующего полный цикл обработки: от загрузки данных из внешних API и файлов до формирования витрин и дашбордов. В проекте применяется Medallion-архитектура Bronze-Silver-Gold, в которой данные последовательно очищаются, структурируются и агрегируются для бизнес-аналитики.

Хранилище реализовано на базе PostgreSQL и используется как центральный DWH для хранения «сырых», «очищенных» и агрегированных данных. Обработка выполняется в ApacheSpark и Python, что позволяет обрабатывать большие объёмы данных и выполнять сложные преобразования и агрегирование.

Оркестрация процессов осуществляется с помощью ApacheAirflow, где описан DAG, автоматизирующий загрузку, обработку и обновление витрин. Конвейер включает ветки работы с данными о погоде и о поездках такси, а также этап объединения этих данных в единую витрину для анализа влияния погодных условий на транспортную активность.

Инфраструктура развернута в контейнерах Docker с использованием DockerCompose, что обеспечивает воспроизводимость и удобство запуска. Для визуализации и отчётности используются Streamlit–дашборд и автоматически формируемые Excel–отчёты с различными графиками.

Практическая значимость работы заключается в демонстрации полного цикла построения аналитического решения – от загрузки и обработки больших данных до формирования витрин и визуализации.

Разработанный прототип может быть использован как основа для дальнейшего развития в направлении потоковой обработки данных, расширения источников и внедрения VI–инструментов.

**М. И. Василевский, И. Л. Стефановский**  
(ГГТУ имени П. О. Сухого, Гомель)

## **РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ПО ПЛАНИРОВАНИЮ И АВТОМАТИЗАЦИИ ЛАБОРАТОРНЫХ И ТЕХНИЧЕСКИХ ИСПЫТАНИЙ**

В современных условиях научно-технической и производственной деятельности эффективность проведения лабораторных и технических испытаний является приоритетной задачей. Работа специалистов испытательных лабораторий связана с обработкой больших объёмов экспериментальных и измерительных данных, а также с ручным планированием и оформлением результатов, что увеличивает трудоёмкость процессов и повышает вероятность ошибок.

Целью данной работы является разработка информационной системы (ИС), предназначенной для планирования и автоматизации лабораторных и технических испытаний. Программное обеспечение базируется на реляционной базе данных, объединяющей сведения об объектах испытаний, методиках, оборудовании и результатах измерений [1]. Архитектура разрабатываемой системы включает модули планирования испытаний, обработки данных и генерации отчётной документации.

Модуль планирования испытаний обеспечивает автоматизацию формирования графиков работ, распределения ресурсов и контроля соблюдения утверждённых методик. Взаимодействие с базой данных посредством *SQL*-запросов гарантирует целостность и высокую скорость обработки информации, позволяя выполнять оперативный анализ результатов и формировать необходимые отчёты.

Практическое внедрение информационной системы позволяет сократить временные затраты специалистов на обработку данных и документооборот приблизительно на 60 %. Автоматизация процессов проведения лабораторных и технических испытаний повышает точность и прозрачность выполнения работ, снижает риск ошибок и обеспечивает централизованный контроль всех этапов испытательного процесса.

### **Литература**

1 Громов, Ю. Ю. Информационные системы и технологии: учебное пособие для вузов / Ю. Ю. Громов, В. В. Беляев. – М.: ЮНИТИ-ДАНА, 2020. – 303 с.

**Е. П. Вегера**

*(ГГТУ имени П. О. Сухого, Гомель)*

## **РАЗРАБОТКА WEB-ПРИЛОЖЕНИЯ ДЛЯ АВТОМАТИЗАЦИИ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОГО УЧЕТА НА КАФЕДРЕ С МОДУЛЕМ ПОДКЛЮЧЕНИЯ К 1С**

В современном учреждении образования эффективное управление материально-техническими ресурсами играет важную роль в обеспечении качественного учебного процесса. На кафедрах вузов ведется учет большого количества оборудования, материалов,

компьютерной техники и других материальных ценностей. Традиционные способы учета, основанные на бумажной документации или разрозненных электронных таблицах, являются трудоемкими, подвержены ошибкам и не обеспечивают должной оперативности и контроля.

Эффективным решением данной задачи является разработка *web*-приложения для автоматизации материально-технического учета на кафедре как базовом структурном подразделении вуза. Такое приложение позволяет централизованно хранить данные о материальных ценностях, отслеживать их текущее состояние, местоположение, движение, а также оперативно формировать необходимые отчеты.

Разрабатываемое *web*-приложение предназначено для упрощения и повышения эффективности работы сотрудников кафедры, ответственных за учет материально-технических ресурсов. Система обеспечивает автоматизацию основных операций, включая добавление, редактирование и удаление информации об оборудовании, учет перемещений, контроль наличия и состояния материальных средств. Это позволяет снизить вероятность ошибок, связанных с человеческим фактором, повысить достоверность данных и сократить время на выполнение учетных операций. Особенностью *web*-приложения является наличие модуля подключения к системе 1С: Предприятие, которая широко используется в организациях для ведения бухгалтерского и управленческого учета. Такая интеграция обеспечивает обмен, синхронизацию и исключает необходимость повторного ввода информации о материальных ценностях в разные системы.

Внедрение предлагаемой разработки позволит оптимизировать рабочие процессы, повысить прозрачность учета и обеспечить надежное хранение информации, что в конечном итоге способствует повышению эффективности деятельности кафедры в целом.

**А. В. Вожейко**

*(БрГУ имени А. С. Пушкина, Брест)*

## **НЕЙРОСЕТЕВАЯ КЛАССИФИКАЦИЯ ПОВЕДЕНЧЕСКИХ ПАТТЕРНОВ ДЛЯ ЗАЩИТЫ ОТ АВТОМАТИЗИРОВАННЫХ СКРИПТОВ**

В современных информационных системах защита от автоматизированных скриптов (ботов) является критически важной задачей.

Традиционные методы фильтрации постепенно теряют эффективность из-за совершенствования алгоритмов имитации действий человека. Перспективным подходом является поведенческая биометрия, основанная на анализе динамики взаимодействия пользователя с манипулятором [1].

Целью работы является разработка нейросетевой модели для классификации сессий взаимодействия на классы «человек» и «бот».

В качестве первичных данных используются временные ряды координат курсора  $(X, Y)$  и моменты времени  $t$ , полученные в ходе мониторинга активности.

Для обучения нейронной сети сформированы две выборки. Первая группа содержит паттерны естественных движений человека, вторая – данные программной эмуляции (линейная интерполяция, имитация случайных блужданий).

В качестве математической модели классификатора используется многослойный перцептрон. На этапе предобработки данных вычисляются векторы признаков: мгновенная скорость  $V_i = \sqrt{\Delta X^2 + \Delta Y^2} / \Delta t$ , ускорение и кривизна траектории. Обучение проводится методом обратного распространения ошибки [2].

Применение модели позволяет в реальном времени оценивать вероятность того, что текущие действия совершаются программным скриптом.

Результаты экспериментов показали, что классификация на основе анализа динамических признаков обеспечивает точность распознавания более 90 %, что достаточно для систем обеспечения информационной безопасности [3].

### Литература

1 Хайкин, С. Нейронные сети: полный курс / С. Хайкин. – М.: Вильямс, 2019. – 1104 с.

2 Осовский, С. Нейронные сети для обработки информации / С. Осовский. – М.: Финансы и статистика, 2002. – 344 с.

3 Гудфеллоу, И. Глубокое обучение / И. Гудфеллоу, И. Бенджио, А. Курвилль. – М.: ДМК Пресс, 2018. – 652 с.

**В. Э. Войновская**  
(ГрГУ имени Янки Купалы, Гродно)

## **АВТОМАТИЗИРОВАННОЕ ТЕСТИРОВАНИЕ БЕЗОПАСНОСТИ ВЕБ-ПРИЛОЖЕНИЙ ДЛЯ ВЫЯВЛЕНИЯ УЯЗВИМОСТЕЙ**

В эпоху цифровых угроз автоматизированное тестирование безопасности веб-приложений становится ключевым щитом защиты информации. Основой подхода служит комплексное применение Selenium WebDriver, API-тестирования, а также методов статического и динамического анализа для моделирования атак и выявления брешей в архитектуре [1].

Подход базируется на многоуровневом сканировании приложения, сочетающем проверку клиентской и серверной частей. Selenium WebDriver адаптируется для ввода вредоносных данных и симуляции попыток инъекций, а API-тестирование проверяет бэкенд модифицированными запросами, чтобы убедиться в надежности защиты конечных точек.

Применяется гибридная методология, объединяющая статический анализ в CI/CD-пайплайнах с динамическим тестированием в рантайме. С помощью автоматизированных сценариев на Java и Python имитируем аутентификацию с подставными данными и проверяем обработку JSON и XML, выявляя ошибки конфигурации серверов.

Внедрение автоматизации позволило сократить время ревью и сделать тесты масштабируемыми, минимизировав человеческий фактор. Комбинация инструментов помогла выявить критические уязвимости в авторизации и незащищенные точки подключения на ранних стадиях, предотвратив потенциальные утечки данных до выхода продукта.

Автоматизация тестирования создает базу знаний о типовых уязвимостях, которая будет использоваться для обучения команд разработки. Это позволит сместить акцент с поиска и исправления чужих ошибок на их предотвращение еще на этапе написания кода.

### **Литература**

1 Майерс, Г. Искусство тестирования программ / Г. Майерс, Т. Баджетт, К. Сандлер. – М.: Диалектика, 2015. – 272 с.

**Д. Н. Гаврик**  
(БНТУ, Минск)

## **ПРАКТИЧЕСКАЯ ОПТИМИЗАЦИЯ И УСКОРЕНИЕ ИНФЕРЕНСА ДИФУЗИОННЫХ МОДЕЛЕЙ**

Показано прикладное сравнение способов ускорения инференса Stable Video Diffusion (image-to-video) при фиксированном входе и настройках генерации (1024x576, 25 кадров) на GPU с 16 ГБ видеопамати. Цель - оценить выигрыш во времени и памяти и определить методы, которые ускоряют расчет без заметной деградации результата.

Рассмотрены подходы, которые можно комбинировать: компрессия параметров (INT8 weight-only квантование UNet), оптимизация графа (torch.compile и TF32), снижение числа шагов диффузии, замена UNet на дистиллированные веса (включая LCM), полуструктурная разреженность 2:4, а также перенос части вычислений в постобработку (ключевые кадры с интерполяцией RIFE). Оценивались время, пик VRAM (allocated/reserved) и прокси-метрики: CLIP similarity к исходному изображению, tSSIM и tLPIPS между соседними кадрами.

Наибольшее ускорение показали ключевые кадры + RIFE: 99,4 с и 3,45 раза относительно базового режима при высокой привязке к исходнику (CLIP similarity 0,981). LCM с 4 шагами дает сбалансированное ускорение 1,92 раза при временной согласованности, близкой к базовому режиму (tSSIM 0,722; tLPIPS 0,167), и умеренной потере привязки (0,930). INT8 квантование ускоряет инференс в 1,27 раза и снижает пик allocated на 669 МБ. Разреженность 2:4 без sparsity-aware fine-tuning приводит к коллапсу результата и применима только при наличии подготовленных весов.

### **Литература**

1 Ho, J. Denoising Diffusion Probabilistic Models / J. Ho, A. Jain, P. Abbeel // arXiv:2006.11239. – 2020.

2 Blattmann, A. Stable Video Diffusion: Scaling Latent Video Diffusion Models to Large Datasets / A. Blattmann [и др.] // arXiv:2311.15127. – 2023.

3 Luo, S. Latent Consistency Models: Few-Step Inference / S. Luo [и др.] // arXiv:2310.04378. – 2023.

**Д. А. Гаврилова**  
(ГГТУ имени П. О. Сухого)

## **ОЦЕНКА ЭФФЕКТИВНОСТИ РАБОТЫ РАСПРЕДЕЛЁННЫХ СИСТЕМ**

Оценка эффективности современных распределенных систем (РС) базируется на трех фундаментальных моделях, определяющих пропускную способность и пределы масштабирования.

Во-первых, любая высоконагруженная РС представляет собой многоканальную систему массового обслуживания (СМО). Основной метрикой стабильности выступает коэффициент загрузки кластера:

$K = \frac{\lambda}{n\mu}$ , где  $\lambda$  – интенсивность входящего потока запросов,  $\mu$  – про-

изводительность одного узла, а  $n$  – количество серверов. Математика СМО доказывает: при приближении  $K \rightarrow 1$  время ожидания в очереди растет асимптотически. Система нелинейно деградирует, приводя к отказам в обслуживании [1].

Во-вторых, экстенсивное горизонтальное масштабирование строго ограничивается законом Амдала. Он описывает предел ускорения системы  $S$  при наличии доли последовательного кода  $\alpha$  на  $p$  вычислительных узлах:

$$S = \frac{1}{\alpha + \frac{1-\alpha}{p}}.$$

В-третьих, это ограничение нивелируется законом Густавсона-Барсиса. В реальной практике кластеры масштабируют для параллельной обработкикратно растущих объемов данных. В этом случае масштабированное ускорение  $S_p$  имеет вид:

$$S_p = p - \alpha \times (p - 1).$$

### **Литература**

1 Плескунов, М. А. Теория массового обслуживания: учебное пособие / М. А. Плескунов; М-во науки и высшего образования РФ, Урал. федер. ун-т. – Екб.: Изд-во Урал. ун-та, 2022. – 80 с.

**С. В. Гагалушко**

*(ГрГУ имени Янки Купалы, Гродно)*

## **АНАЛИЗ УЯЗВИМОСТЕЙ И ПРОЕКТИРОВАНИЕ БЕЗОПАСНОЙ СИСТЕМЫ АУТЕНТИФИКАЦИИ ДЛЯ КОРПОРАТИВНЫХ СЕТЕЙ НА ОСНОВЕ СЕТЕВОГО ПРОТОКОЛА АУТЕНТИФИКАЦИИ KERBEROS**

Проведён анализ уязвимостей протокола Kerberos и на его основе спроектирована защищённая система аутентификации для корпоративных сетей. Объектом исследования выступает процесс аутентификации в распределённых средах, предметом - проектирование и реализация безопасной системы с использованием алгоритма шифрования Magma. Цель работы заключается в разработке модульного прототипа, устойчивого к современным атакам и пригодного для практического применения.

Методология включает аналитический обзор угроз (Golden Ticket, Silver Ticket, Kerberoasting, replay-атаки, SQL-инъекции), моделирование протокольных процедур, программную реализацию на Python и PHP, а также тестирование на устойчивость. Разработанная система реализует трёхэтапный протокол Kerberos (AS, TGS, Service Request) с использованием блочного шифра Magma в режиме CBC. Ключевые архитектурные решения: двойное шифрование билетов, серверное хранение долговременных ключей, криптографически стойкая генерация сессионных идентификаторов, механизм защиты от повторного воспроизведения (replay) и параметризованные запросы к базе данных.

В ходе тестирования подтверждена корректность реализации протокола и эффективность встроенных защитных механизмов. Система продемонстрировала устойчивость к подделке билетов (Golden Ticket, Silver Ticket), офлайн-подбору ключей (Kerberoasting), перехвату сессий и SQL-инъекциям. Полученные результаты свидетельствуют о возможности построения криптографически стойкой инфраструктуры аутентификации на базе Kerberos с заменой стандартных шифров на алгоритм Magma. Разработанный прототип может использоваться как основа для внедрения в корпоративные сети, а также в качестве демонстрационного стенда для обучения специалистов методам защиты распределённых систем.

**В. В. Галькевич**  
(БГУ, Минск)

## **РАЗРАБОТКА ОТКАЗОУСТОЙЧИВОГО ОДНОРАНГОВОГО IN-MEMORY ХРАНИЛИЩА ДАННЫХ**

В работе был рассмотрен вопрос создания алгоритма хранения данных и реализации на его базе однорангового хранилища данных, работающего с данными в оперативной памяти, а также способного к восстановлению после отказа не более двух из машин-носителей.

Идея алгоритма состоит в следующем: необходимо завести два паритета, каждый из которых будет подсчитан по своему правилу и будет необходим для восстановления данных. Тем не менее, не любые два алгоритма подсчёта двух паритетов будут удовлетворять задаче.

Формализуем правило выбора алгоритмов для подсчёта паритетов. Пусть в одной кодовой группе есть  $k$  блоков данных  $D_0, \dots, D_{k-1}$  и 2 паритета  $P$  и  $Q$ , тогда паритеты можно характеризовать как функции от этих блоков данных. Первый паритет можно представить в виде  $P = f(D_0, \dots, D_{k-1})$ , а второй паритет в следующем виде  $Q = g(D_0, \dots, D_{k-1})$ . Тогда для того, чтобы выбранные паритеты удовлетворяли условию нашей задачи, необходимо чтобы для любых двух индексов  $u \neq v$  система уравнений, порождаемая  $P$  и  $Q$ , позволяла однозначно найти величины  $D_u, D_v$  по оставшимся данным.

Хранилище, разработанное в соответствии с описанным алгоритмом, позволяет получить выигрыш в размере избыточных данных порядка  $N^2$  раз, где  $N$  – число машин (дата-центров) в кластере.

### **Литература**

1 Галькевич, В. В. Разработка Распределенного In-Memory хранилища данных / В. В. Галькевич. – Мн.: 2025.

2 Petrov, A. Distributed data / A. Petrov. – O'Reilly Media, 2024.

3 RAID: Redundant array independent disks: [site]. – URL: <https://en.wikipedia.org/wiki/RAID/> (date of access: 22.05.2025).

4 Redis: The Real-time Data Platform: [site]. – URL: <https://redis.io/>. (date of access: 22.05.2025).

5 Cassandra: Apache Cassandra Documentation: [site]. – URL: <https://cassandra.apache.org/>. (date of access: 22.05.2025).

**К. С. Гапонова, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ И СТРУКТУРЫ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ WEB-САЙТА ОДО «АМЕЛИЯ»**

В рамках данного проекта в phpMyAdmin была спроектирована реляционная база данных для работы ОДО «Амелия». Таблица пользователей содержит в себе поля для хранения логинов, контактных данных и зашифрованных паролей, что обеспечивает персонализацию сервиса. Таблица продукции включает в себя исчерпывающий набор атрибутов, таких как уникальные идентификаторы, наименования, подробные текстовые описания, стоимостные показатели и пути к графическим файлам для визуализации каталога.

Отдельное внимание было уделено таблице заказов, которая выступает связующим звеном между покупателем и товарной позицией, фиксируя временные метки и статусы покупок.

Серверная часть разработана на языке программирования PHP, который обеспечивает эффективную коммуникацию между клиентским интерфейсом и сервером базы данных. Основные алгоритмы системы сосредоточены на обработке пользовательских запросов, таких как регистрация и авторизация. В процессе регистрации PHP-скрипты осуществляют валидацию вводимых данных, проверяют наличие существующих аккаунтов в базе и выполняют запись новой информации. Для реализации функционала интернет-магазина был использован механизм сессий, позволяющий сохранять состояние «корзины» пользователя при перемещении между различными разделами сайта. Алгоритм оформления заказа считывает данные из временного хранилища сессии, формирует итоговую структуру транзакции и производит запись в соответствующие таблицы базы данных.

Дополнительно в рамках серверной части был реализован модуль администратора, который позволяет сотрудникам ОДО «Амелия» осуществлять оперативное обновление каталога товаров, изменять цены и модерировать поступающие заказы через удобный web-интерфейс, не требующий прямого обращения к phpMyAdmin.

Особое внимание в программном коде уделено вопросам безопасности: все входящие данные проходят этап очистки от потенциально опасных символов, а использование встроенных функций РНР для работы с паролями гарантирует защиту конфиденциальной информации.

**П. С. Гарацкевич**  
(ГрГУ имени Янки Купалы, Гродно)

## **ПРОЕКТИРОВАНИЕ ЛОГИКИ СРАЖЕНИЯ В КОМПЬЮТЕРНОЙ ИГРЕ**

Логика сражения напрямую влияет на вовлечённость игрока в игровой процесс, баланс и общее восприятие компьютерной игры. Целью данного проекта является разработка и описание логики сражения для компьютерной игры, включающей расчёт потерь, защитных параметров, вероятностных факторов.

В рамках реализуемого проекта глобальной стратегии была введена механика сражений, базирующаяся на механиках расчета процесса боевого столкновения, расчетов потерь сторон, расчётов бонусов и модификаторов локаций и построек.

В основе расчёта урона для сторон используются формула:

$$C \cdot t \cdot k(A/D),$$

где  $C$  – кол-во человек в отряде,

$t$  – время отряда в сражении,

$k$  – отдельный коэффициент потери для атакующей и защищающейся стороны.

При потере людей в отряде, отряд ослабевает по параметрам. Логика сражения предусматривает, большие потери атакующих по отношению к защищающимся. Функция  $k(A, P)$  рассчитывает коэффициент потерь и рассчитывается как  $A/P$  для защищающейся стороны и  $1 - A/P$  для атакующей стороны соответственно, где  $A$  – сумма параметра атаки атакующих,  $P$  – сумма параметров атаки у атакующих и защиты у защищающихся.

Отдельное внимание уделяется коэффициенту  $t$  (времени отряда в бою) и подкреплениям. Чем дольше отряд находится в сражении, тем больше человек он теряет с каждым ходом. У отрядов присоединившихся недавно, данный коэффициент минимален и растёт параллельно с остальными отрядами.

Таким образом, проектирование логики сражения представляет собой комплексную задачу, которая объединяет математические расчёты и реалистичность. Грамотная реализация данной логики, улучшит взаимодействия игрока с игрой и повысит шансы на окупаемость продукта.

**А. А. Герасименко, Г. Л. Карасёва**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА МОБИЛЬНОЙ ИГРЫ В ЖАНРЕ SCI-FI ШУТЕРА В СРЕДЕ РАЗРАБОТКИ UNITY НА C#**

Данный проект посвящён разработке мобильной top-down action-игры в сеттинге зомби-апокалипсиса с акцентом на динамичный игровой процесс, управление большими группами противников и принятие тактических решений в условиях ограниченного пространства. Основной целью работы является создание играбельного прототипа и формирование масштабируемой архитектуры проекта, ориентированной на дальнейшее развитие игры и расширение её функциональных возможностей.

Игровой процесс строится вокруг core loop коротких игровых сессий, включающего исследование локации, сражения с ордами противников, сбор ресурсов и последующую прокачку персонажа. Ключевой игровой механикой выступает система динамического риска, при которой игрок самостоятельно определяет момент завершения миссии, выбирая между безопасной эвакуацией и продолжением прохождения с постепенно возрастающей сложностью ради получения более ценных наград.

Проектирование уровней осуществляется с применением модульного подхода, позволяющего создавать масштабируемые игровые пространства и гибко адаптировать их под различные игровые сценарии. Особое внимание уделяется обеспечению читаемости боевых ситуаций, что достигается за счёт продуманной структуры локаций и размещения интерактивных элементов.

Практическая часть проекта направлена на реализацию ключевых игровых систем, включая боевую механику, управление противниками, механику эвакуации, экономику ресурсов и пользовательский интерфейс. Разрабатываемый прототип демонстрирует взаимодействие основных компонентов игрового процесса и используется для тестирования баланса, сложности и технических решений.

Результатом работы является функционирующая игровая сборка, отражающая применённые принципы гейм-дизайна и программной архитектуры, а также подтверждающая возможность дальнейшего масштабирования проекта, добавления новых уровней, противников и игровых режимов.

**М. Д. Гребёнкин, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА FRONT-END ЧАСТИ ПРИЛОЖЕНИЯ ДЛЯ СОЗДАНИЯ И ПОИСКА МЕРОПРИЯТИЙ ПО ИНТЕРЕСАМ С ИСПОЛЬЗОВАНИЕМ REACT, TYPESCRIPT**

В условиях активного развития цифровых технологий особую актуальность приобретают веб-приложения, направленные на социальное взаимодействие и организацию совместной деятельности. Одним из таких направлений является создание платформ для поиска и организации мероприятий по интересам, позволяющих пользователям находить единомышленников и планировать совместные встречи. Использование современных front-end технологий обеспечивает удобный и отзывчивый пользовательский интерфейс, адаптированный под различные устройства.

Была разработана front-end часть веб-приложения для создания и поиска мероприятий по интересам с использованием библиотеки React [1] и языка программирования TypeScript [2]. React позволяет реализовать компонентный подход к построению пользовательского интерфейса, что повышает масштабируемость и упрощает сопровождение приложения. Применение языка программирования TypeScript обеспечивает строгую типизацию, снижает количество ошибок на этапе разработки и повышает надежность кода.

Разработанное веб-приложение предоставляет пользователям возможность просматривать список публичных мероприятий, осуществлять поиск по названию и дате, а также создавать собственные встречи. Реализован удобный интерфейс карточек мероприятий с отображением основной информации, включая дату, описание и автора события. Дополнительно в приложении предусмотрены функции фильтрации и рекомендации мероприятий, что повышает удобство навигации и персонализации пользовательского опыта. Особое внимание при разработке было уделено адаптивности интерфейса и корректному управлению состоянием приложения.

### Литература

1 Стефанов, С. React.js. Быстрый старт / С. Стефанов. – СПб.: Питер, 2016. – 322 с.

2 Вандеркам, Д. Эффективный TypeScript: 62 способа улучшить код / Д. Вандеркам. – СПб.: Питер, 2020. – 288 с.

**А. А. Гришко, Е. М. Березовская**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ОНЛАЙН-МАГАЗИНА С ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМОЙ РЕКОМЕНДАЦИЙ ТОВАРОВ НА ОСНОВЕ МАШИННОГО ОБУЧЕНИЯ**

Сегодня электронная торговля активно развивается, и всё большее значение в ней получают технологии искусственного интеллекта. Если раньше системы рекомендаций строились на заранее прописанных правилах и большом количестве условий, то сейчас появляется возможность использовать обучаемые модели, которые сами анализируют данные и находят закономерности.

Ранее для создания рекомендательной системы требовалось вручную продумывать и реализовывать множество алгоритмов и связей между товарами. Такой подход усложнял разработку и требовал постоянных доработок при изменении поведения пользователей. Кроме того, подобные решения плохо адаптировались к новым данным. Целью данной работы является показать, как методы машинного обучения могут использоваться в торговых системах в качестве интеллектуального помощника, который формирует

рекомендации автоматически на основе накопленной информации. По результатам проделанных исследований был разработан онлайн-магазин по клиент-серверной архитектуре на платформе .NET. Серверная часть реализована с использованием ASP.NET Core Web API, для работы с базой данных применяются SQL Server и Entity Framework Core. Модуль рекомендаций создан на базе библиотеки ML.NET и встроен в серверную часть приложения.

В отличие от классического подхода, логика рекомендаций не прописывается вручную в виде большого количества правил. Вместо этого модель обучается на данных о просмотрах и покупках пользователей и на их основе формирует персональные предложения. При накоплении новых данных модель может быть переобучена, что позволяет системе адаптироваться к изменению интересов пользователей. Практическая значимость работы заключается в демонстрации того, что современные инструменты машинного обучения позволяют упростить разработку интеллектуальных модулей и сделать их более гибкими по сравнению с традиционными алгоритмическими решениями. Применение таких технологий повышает качество рекомендаций и улучшает взаимодействие пользователя с системой.

Таким образом, использование искусственного интеллекта в электронной коммерции можно рассматривать как закономерный этап развития программных систем и перспективное направление дальнейшего совершенствования торговых платформ.

**В. В. Гришкова**

*(ГГУ имени Ф. Скорины, Гомель)*

## **РАЗРАБОТКА ИНТЕРНЕТ-МАГАЗИНА ОБОРУДОВАНИЯ И БЫТОВОЙ ТЕХНИКИ «ROBUST» С ИСПОЛЬЗОВАНИЕМ PHP**

Онлайн-продажа бытовой техники и оборудования с каждым годом набирает всё большие обороты. Домохозяйства и организации обеспечивают постоянно растущий спрос на технику и оборудование, удовлетворяющие их потребности. В условиях конкуренции за потребителя коммерческие сайты магазинов, реализующие устройства для повседневной жизни и бизнеса, перестают быть просто красивой витриной. Теперь веб-сайты предоставляют полный

цикл продажи: от продвижения, привлечения покупателей до формирования покупок в корзине и безналичной оплаты. Чтобы обеспечить многофункциональность сайта по продаже техники «Robust», использовался язык программирования PHP. Он позволяет обрабатывать на сервере различные сценарии поведения и работать с базой данных. PHP предусматривает работу с сессиями, на сайте это применяется для сохранения информации о товарах в корзине и их количестве. При регистрации на сайте и формировании заказа данные с форм запишутся и будут храниться в MySQL, универсальной и надежной СУБД.

На сайте присутствует разделение ролей на пользователя (покупателя) и администратора (продавца). Пользователю предоставляется удобный и интуитивно понятный интерфейс: просмотр товаров в каталоге и их подробное описание в карточках, сортировка по названию, цене, рейтингу, поиск, добавление в корзину, формирование заказа и оплата. Благодаря таким опциям даже крупные покупки теперь можно осуществить в пару кликов, и сумма в корзине высчитывается автоматически со всеми скидками и промокодами. Администратор получает доступ к базе данных и может просматривать данные покупателя и его заказы, а на сайте может отвечать на вопросы по товару, добавлять новые позиции в каталог, учитывать остатки, изменять цены, оповещать об акциях, осуществлять рекламную рассылку. Таким образом, администратор не теряет связь с покупателями и видит, какие товары наиболее востребованы в данный момент, что позволяет в будущем выстраивать персональные рекомендации и формировать лояльность к магазину.

**Н. С. Громько, А. В. Лубочкин**  
(ГГУ имени Ф. Скорины, Гомель)

## **НАСТОЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ АВТОМАТИЗАЦИИ СКЛАДСКОГО И БУХГАЛТЕРСКОГО УЧЁТА КОФЕЙНОГО БИЗНЕСА С ИСПОЛЬЗОВАНИЕМ PYTHON И POSTGRESQL**

В условиях современного развития малого и среднего бизнеса возрастает потребность в автоматизации процессов учёта и управления ресурсами предприятия. Актуальна данная задача и для сферы

розничной торговли и вендинговых услуг, в частности бизнеса, связанного с эксплуатацией кофейных аппаратов. Эффективное управление запасами ингредиентов, учёт партий поставок требуют применения специализированного программного обеспечения.

Данная работа представляет разработку настольного приложения для автоматизации складского и бухгалтерского учёта кофейного бизнеса. Приложение предназначено для ведения учёта ингредиентов и их партий, автоматического списания ингредиентов на основе рецептур напитков, а также создания резервных копий базы данных.

Для разработки использован язык программирования Python [1], обеспечивающий высокую скорость разработки, для реализации графического интерфейса – библиотека PyQt5 [2], позволяющая создавать удобные и функциональные настольные приложения для операционной системы Windows. В качестве СУБД выбрана PostgreSQL [3], которая обеспечивает надёжное хранение больших объёмов информации, поддержку транзакций и возможность масштабирования системы при увеличении количества точек продаж.

Разработанное приложение включает следующие основные модули: учёт ингредиентов и партий поставок; управление рецептурами напитков; учёт продаж и; резервное копирование и архивирование данных.

### Литература

- 1 Лутц, М. Изучаем Python / М. Лутц. – 5-е изд. – СПб.: Питер, 2020. – 832 с.
- 2 Summerfield, M. Rapid GUI Programming with Python and Qt / M. Summerfield. – Prentice Hall, 2007. – 648 p.
- 3 Документация PostgreSQL: [сайт]. – URL: <https://www.postgresql.org/docs/> (дата обращения: 16.02.2026).

**В. Д. Гузов, С. В. Киргинцева**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ БАНКА “АВАНК” С ИСПОЛЬЗОВАНИЕМ JAVASCRIPT**

Разработано веб-приложение для банка “АВанк” с использованием JavaScript, предназначенное для предоставления клиентам

удобного доступа к основным дистанционным банковским услугам через браузер. В современных условиях развития финансовых технологий одна из актуальных задач – создание интуитивно понятных веб-интерфейсов, обеспечивающих круглосуточное взаимодействие пользователей с банковскими продуктами и сервисами.

Основная логика интерфейса реализована на языке JavaScript с использованием событийной модели и работы с DOM, что позволяет динамически обновлять содержимое страниц без полной перезагрузки, повышая удобство и скорость работы пользователя [1–3].

Особое внимание уделено вопросам usability и адаптивной верстке: пользовательский интерфейс разработан таким образом, чтобы корректно отображаться на настольных и мобильных устройствах, обеспечивая единый сценарий работы клиента независимо от используемой платформы. Дополнительно реализованы элементы клиентской валидации вводимых данных средствами JavaScript, что позволяет сократить количество ошибок при формировании заявок и повысить общую надежность системы. Теоретической основой исследования являются современные подходы к разработке веб-приложений банковской тематики, использование JavaScript как основного языка для реализации интерактивного фронтенда.

### **Литература**

- 1 Дакетт, Дж. HTML и CSS. Разработка и дизайн веб-сайтов / Дж. Дакетт. – М.: Эксмо, 2021. – 512 с.
- 2 Никсон, Р. PHP, MySQL и JavaScript. Исчерпывающее руководство / Р. Никсон. – СПб.: Питер, 2020. – 768 с.
- 3 Флэнаган, Д. JavaScript. Подробное руководство / Д. Флэнаган. – СПб.: Символ-Плюс, 2020. – 1100 с.

**Н. С. Гулюк, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ИНЖЕНЕРНОГО КАЛЬКУЛЯТОРА НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ JAVA**

Современные инженерные и технические расчёты требуют использования инструментов, способных выполнять не только базовые

арифметические операции, но и сложные вычисления, включая работу с тригонометрическими, логарифмическими, степенными функциями, а также обработку выражений с несколькими уровнями вложенности.

Стандартные калькуляторы не всегда удовлетворяют этим требованиям, что обуславливает актуальность разработки инженерного калькулятора с расширенным функционалом.

Был разработан инженерный калькулятор на языке программирования Java [1]. Выбор языка обусловлен его платформенной независимостью, объектно-ориентированной парадигмой и широкими возможностями стандартной библиотеки.

Графический интерфейс реализован с использованием JavaFX, что обеспечивает удобство взаимодействия пользователя с программой. Разработанный калькулятор поддерживает выполнение базовых арифметических операций (сложение, вычитание, умножение, деление), вычисление степеней и корней, работу с тригонометрическими функциями ( $\sin$ ,  $\cos$ ,  $\tan$ ), логарифмами, факториалом.

Реализован механизм корректного разбора и вычисления математических выражений с учётом приоритета операций и правил ассоциативности.

Программная архитектура построена с применением принципов объектно-ориентированного программирования. Логика вычислений отделена от графического интерфейса, что повышает модульность и упрощает дальнейшее расширение функционала.

Обработка исключительных ситуаций (деление на ноль, некорректный ввод, выход за пределы допустимых значений функций) обеспечивает устойчивость работы приложения.

Разработанный инженерный калькулятор может использоваться студентами технических специальностей, школьниками старших классов и специалистами для выполнения учебных и прикладных расчётов.

### Литература

1 Блинов, И. Н. Java from ЕРАМ / И. Н. Блинов, В. С. Романчик. – Мн.: Четыре четверти, 2020. –560 с.

**Н. Д. Гурбо**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ПРИМЕНЕНИЕ МОДУЛЬНО-МОНОЛИТНОЙ И ЛУКОВОЙ АРХИТЕКТУРЫ В СИСТЕМЕ ОБРАБОТКИ ПОЛЬЗОВАТЕЛЬСКИХ ОБРАЩЕНИЙ**

Выбор архитектурных решений в backend-разработке определяет не только структуру программной системы, но и характер её дальнейшего развития и сопровождения. При проектировании прикладных сервисов важно находить баланс между простотой эксплуатации и возможностью постепенного расширения функциональности. В контексте систем с единой предметной областью и последовательной бизнес-логикой целесообразно рассматривать монолитную модель как базовую организацию приложения.

Снижение внутренней связанности достигается за счёт концентрической организации слоёв, при которой зависимости направлены к доменному ядру, содержащему основные сущности и бизнес-правила. Прикладной уровень инкапсулирует сценарии работы пользователей и агентов, включая создание обращений и обработку комментариев, не завися от конкретной технологии хранения данных. Инфраструктурные механизмы, такие как реализация доступа к данным через ORM и аутентификация на основе JWT, изолированы от бизнес-правил и могут быть заменены без переработки доменной модели.

Подобное разделение ответственности формирует модульную структуру монолитного приложения, ограничивает область распространения изменений и поддерживает инверсию зависимостей между его слоями. В результате монолит сохраняет целостность транзакционной модели и простоту развертывания, одновременно обеспечивая разделение внутренних компонентов, что позволяет развивать систему обработки пользовательских обращений без перехода к распределённой архитектуре.

### **Литература**

1 Распространенные архитектуры веб-приложений: [сайт]. – URL: <https://learn.microsoft.com/ru-ru/dotnet/architecture/modern-web-apps-azure/common-web-application-architectures> (дата обращения: 27.02.2026).

**Д. А. Гусаков, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА FRONT-END ЧАСТИ ВЕБ-ПРИЛОЖЕНИЯ «МАРКЕТПЛЕЙС» С ИСПОЛЬЗОВАНИЕМ HTML5, CSS3, JAVASCRIPT, FIGMA**

В современной веб-разработке стандартом становится архитектура SPA (Single Page Application), обеспечивающая мгновенный отклик интерфейса. Целью работы являлась реализация адаптивного клиентского приложения для маркетплейса цифровой техники.

Процесс проектирования веб-приложения «Маркетплейс» выполнен в редакторе Figma с применением подхода “Mobile First” и технологии Auto Layout, что обеспечило масштабируемость дизайн-системы.

Верстка реализована на HTML5 с использованием семантических тегов (<main>, <article>, <aside>) для улучшения SEO-показателей. Визуальная часть построена на CSS3 с применением Grid Layout и Flexbox.

Ключевой особенностью является использование функции clamp() для реализации адаптивной типографики и «резиновых» отступов, что позволило минимизировать количество медиа-запросов в коде.

Программная логика базируется на модульном JavaScript (ES6+) [1,2]. Реализована асинхронная архитектура:

1 Бэкенд: Интеграция с платформой Firebase (Authentication для авторизации и Cloud Firestore в качестве NoSQL базы данных товаров).

2 Корзина: Логика управления заказами реализована на клиенте с сохранением состояния в localStorage, обеспечивая персистентность данных.

3 UX: Внедрена технология «Skeleton Loading» для визуализации процесса загрузки контента.

В результате создано веб-приложение с удобным пользовательским интерфейсом, динамической фильтрацией и системой уведомлений, полностью готовое к эксплуатации.

### Литература

- 1 Флэнаган, Д. JavaScript. Полное руководство / Д. Флэнаган. – М.: Вильямс, 2021. – 720 с.
- 2 Симпсон, К. ECMAScript 6 и не только / К. Симпсон. – СПб.: Питер, 2017. – 336 с.

**Д. В. Дашкевич, Н. А. Аксёнова**  
(ГГУ имени Ф. Скорины, Гомель)

### **РАЗРАБОТКА ПРИЛОЖЕНИЯ ВЕБ-ПРИЛОЖЕНИЯ «ПЕРСОНАЛЬНЫЙ ОРГАНАЙЗЕР»**

Одним из перспективных направлений разработки является создание веб-приложений с поддержкой технологии Progressive Web App (PWA), позволяющей объединить преимущества веб-сайтов и нативных приложений.

Для разработки веб-приложения была использована библиотека React, которая позволяет создавать интерактивные одностраничные приложения (SPA). В качестве языков программирования применялись JavaScript и TypeScript. Для стилизации интерфейса использованы каскадные таблицы стилей CSS. В приложении реализованы следующие функциональные возможности:

- календарь с возможностью добавления заметок и напоминаний;
- система хранения текстовых заметок;
- встроенный калькулятор;
- модуль конвертации валют;
- отображение прогноза погоды;
- переключение светлой и тёмной темы интерфейса.

Хранение пользовательских данных осуществляется локально средствами браузера, что позволяет использовать приложение без регистрации и серверной части. Для реализации технологии PWA используется Service Worker, обеспечивающий кэширование статических ресурсов приложения и возможность его запуска в офлайн-режиме. В случае отсутствия подключения к сети модуль погоды выводит уведомление о недоступности данных. Возможность установки приложения на устройство обеспечивается использованием Web App Manifest.

**А. Д. Дебой**

*(ГГУ имени Ф. Скорины, Гомель)*

## **РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ СКРИНИНГА ФИБРИЛЛЯЦИИ ПРЕДСЕРДИЙ**

Фибрилляция предсердий (ФП) является одной из наиболее распространённых аритмий и существенно повышает риск ишемического инсульта. Значимая доля случаев ФП остаётся недиагностированной из-за бессимптомного или малосимптомного течения, поэтому раннее выявление нарушений ритма на этапе первичного контакта с пациентом рассматривается как приоритетная задача. В клинических рекомендациях подчёркивается целесообразность скрининга ФП у групп повышенного риска, что определяет актуальность создания доступных цифровых инструментов для предварительной оценки ритма сердца [1]. Цель работы – разработка мобильного приложения для скрининга ФП, обеспечивающее быстрое получение предварительного результата и формирование рекомендаций по дальнейшим действиям. В качестве основного источника данных рассматривается фотоплетизмография (PPG), получаемая с камеры смартфона, поскольку позволяет выполнить измерение без специализированного мед. оборудования.

Было разработано мобильное приложение, позволяющее выполнять скрининговые измерения в домашних условиях и повышающий доступность первичного выявления возможной ФП, что может способствовать более своевременному обращению за медицинской помощью и последующей верификации диагноза. В рамках разработки предусматриваются: модуль записи PPG-сигнала с контролем качества (устойчивость контакта, уровень шума); модуль предобработки (фильтрация артефактов, нормализация, выделение пульсовых интервалов); алгоритм классификации ритма (норма / подозрение на ФП / недостаточное качество записи) на основе признаков нерегулярности и/или моделей машинного обучения; пользовательский интерфейс с визуализацией результата, историей измерений и предупреждением о необходимости подтверждения результата клиническими методами (ЭКГ).

### **Литература**

1 Hindricks, G. et al. 2020 ESC Guidelines for the diagnosis and management of atrial fibrillation / G. Hindricks // European Heart Journal, 2021.

**П. В. Дедова**

*(ГрГУ имени Янки Купалы, Гродно)*

### **РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ КАФЕ «ПОЛЯНА СИТИ»**

На современном этапе достаточно большая роль отводится цифровизации общественного питания. Мобильные приложения становятся ключевым инструментом повышения конкурентоспособности заведений. Для кафе «Поляна Сити», расположенного в городе Гродно, разработка собственного мобильного приложения представляет стратегическую необходимость для удовлетворения растущих потребностей клиентов и оптимизации внутренних процессов.

Основными целями разработки являются: запуск собственной доставки заказов, увеличение лояльности клиентов через персонализированный сервис, минимизация ошибок при приеме заказов. Приложение проектируется как кроссплатформенное решение для iOS и Android с использованием фреймворка React Native, что позволяет сократить время разработки и затраты на поддержку.

Архитектура приложения включает несколько взаимосвязанных модулей. Модуль электронного меню предоставляет актуальную информацию о блюдах с фотографиями, составом и возможностью фильтрации по категориям. Система онлайн-заказа позволяет формировать корзину с расчетом итоговой суммы.

Личный кабинет пользователя хранит историю заказов, персональные данные и предпочтения. Push-уведомления информируют о статусе заказа. Административная панель предоставляет аналитику по продажам, управление меню и обработку заказов.

Технологический стек включает: React Native для фронтенда, Node.js с Express.js для бэкенда, PostgreSQL для хранения данных, Redis для кэширования и Firebase для push-уведомлений. API реализует RESTful архитектуру с JWT-аутентификацией. Для платежной интеграции используется система ЕРИП.

Внедрение мобильного приложения для кафе «Поляна Сити» позволит не только улучшить клиентский опыт, но и получить конкурентное преимущество на рынке общественного питания города Гродно.

**А. В. Дидюля**  
(ГрГУ имени Янки Купалы, Гродно)

## **ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ РАБОТЫ С ГРАФИЧЕСКИМ КОНТЕНТОМ ПРИ ПОМОЩИ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ**

При профессиональной работе с графикой, часто возникает необходимость не только создавать высококачественные визуальные образы, но и вносить правки в процессе работы над изображениями. Традиционные методы требуют значительных временных затрат и экспертных навыков. Внедрение генеративных нейросетевых моделей, в частности архитектуры Stable Diffusion, позволит автоматизировать творческие процессы. Целью данной работы является представление концепции веб-приложения для интеллектуальной генерации и модификации графического контента на основе диффузионных моделей.

Определен следующий функционал приложения: модуль генерации по тексту (txt2img) с выбором чекпоинтов моделей, модуль преобразования изображений (img2img) с регулировкой силы дещумизации, инструмент интеллектуальной перерисовки выделенных зон (inpainting) на базе редактора масок, а также сервис нейросетевого увеличения разрешения (upscale).

Диффузионные нейросетевые модели требуют немало количества видеопамяти и оптимизации процесса генерации графического контента. Поэтому учтены следующие технические аспекты: асинхронная клиент-серверная архитектура (разделение интерфейса на React.js и серверной части на FastAPI для обеспечения отзывчивости), распределенная обработка задач (использование брокера сообщений Redis и очереди Celery для управления тяжелыми вычислениями на GPU), оптимизация потребления ресурсов (внедрение методов FP16, Tiled VAE и CPU Offloading для работы на

видеокартах с объемом памяти до 4 Гб), а также масштабируемость (возможность динамического подключения дополнительных вычислительных воркеров).

Таким образом, реализация требует тщательного проектирования архитектуры очередей, оптимизации нейросетевых пайплайнов и обеспечения бесшовного взаимодействия пользователя с генеративными алгоритмами через понятный пользовательский интерфейс.

**С. К. Дмитрук**

*(БрГУ имени А. С. Пушкина, Брест)*

## **ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ СИСТЕМЫ УПРАВЛЕНИЯ ЛОКАЛЬНЫМИ БАЗАМИ ДАННЫХ В МОБИЛЬНОМ ПРИЛОЖЕНИИ-КАТАЛОГЕ**

В современных мобильных информационных системах одной из ключевых задач является обеспечение бесперебойного доступа к данным в условиях нестабильного сетевого соединения. При разработке приложений-агрегаторов, использующих методы веб-парсинга для получения информации с внешних ресурсов (например, сервиса Кинопоиск), возникает необходимость кэширования данных на стороне клиента для повышения производительности и улучшения пользовательского опыта [1].

Целью работы является проектирование и программная реализация реляционной базы данных для хранения и систематизации сведений о мультимедийном контенте в мобильном приложении. В качестве объекта исследования выступает структура данных, включающая такие атрибуты, как название фильма, год выпуска, жанр, авторский состав и краткое описание.

В ходе проектирования была разработана схема базы данных, реализованная средствами СУБД SQLite. Архитектура системы включает таблицы для хранения основной информации о фильмах, справочники жанров и списки пользовательских предпочтений («избранное»). Взаимодействие с базой данных осуществляется через уровень абстракции данных (DAO), что позволяет эффективно выполнять операции вставки, обновления и выборки информации, полученной в результате парсинга внешнего ресурса [2].

Реализованная система управления локальными данными позволяет минимизировать количество сетевых запросов, сокращая время отклика интерфейса и нагрузку на аппаратные ресурсы мобильного устройства. Результаты тестирования подтверждают стабильность работы приложения при объемах данных свыше 1000 записей, что делает предложенное решение пригодным для использования в реальных программных продуктах [3].

### Литература

- 1 Дейт, К. Дж. Введение в системы баз данных / К. Дж. Дейт. – М.: Вильямс, 2016. – 1328 с.
- 2 Фаулер, М. Архитектура корпоративных программных приложений / М. Фаулер. – М.: Вильямс, 2017. – 544 с.
- 3 Шилдт, Г. Java: руководство для начинающих / Г. Шилдт. – М.: Вильямс, 2019. – 816 с.

### З. И. Довидович

*(ГрГУ имени Янки Купалы, Гродно)*

## РАЗРАБОТКА БЕЗОПАСНОЙ СИСТЕМЫ МОНИТОРИНГА ИТ-ИНФРАСТРУКТУРЫ НА ОСНОВЕ ZABBIX С ИНТЕГРАЦИЕЙ HASHICORP VAULT

В условиях цифровой трансформации критически важных отраслей обеспечение безопасности и бесперебойности систем мониторинга становится ключевой задачей. Развертывание таких систем порождает проблему управления конфиденциальными данными: паролями, ключами API и токенами, хранение которых в конфигурационных файлах несет высокий риск утечки. Целью работы была разработка прототипа, в котором вопросы управления доступом и секретами решаются на архитектурном уровне. В качестве основы был развернут стек на базе Zabbix 7.4.6, Nginx и MariaDB. Ключевым решением стала интеграция со специализированным инструментом HashiCorp Vault для централизованного и безопасного хранения секретов. Архитектура предполагает, что Zabbix Server запрашивает секреты напрямую из Vault в момент необходимости через API с использованием ограниченных по времени токенов, что исключает их

Материалы XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 23–25 марта 2026 г.

---

постоянное хранение в конфигурации. Внутри Zabbix была настроена детальная ролевая модель доступа («Администраторы серверов», «Сетевые инженеры»), реализующая принцип минимальных привилегий и предотвращающая горизонтальное перемещение при компрометации. В результате был создан функционирующий прототип, демонстрирующий жизнеспособный подход. Использование специализированных хранилищ секретов должно стать обязательной практикой, гранулярное управление доступом внутри системы мониторинга – это критически важный слой защиты, а гибкость современных систем позволяет создавать безопасные и комплексные решения, адаптированные под задачи объектов с высокими требованиями к защите информации.

### Литература

1 Zabbix 7.4 Documentation. Zabbix LLC: [site]. – URL: <https://www.zabbix.com/documentation/current/en/manual>. (date of access: 26.01.2026).

**А. В. Донец, Е. М. Березовская**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА СИСТЕМЫ «1С: ЛОГИСТ» ДЛЯ УПРАВЛЕНИЯ АВТОМОБИЛЬНЫМИ ПЕРЕВОЗКАМИ И СЕРВЕРА ВЗАИМОДЕЙСТВИЯ ДЛЯ РАСЧЕТА И ОПТИМИЗАЦИИ МАРШРУТОВ**

Актуальность исследуемой тематики обусловлена необходимостью интеграции учетных и логистических функций в единой среде, а также потребностью в интеллектуальной поддержке принятия решений при планировании маршрутов. Традиционные подходы к управлению перевозками зачастую характеризуются разрозненностью данных, отсутствием сквозного контроля статусов и ручным расчётом маршрутов, что ведет к неэффективному использованию транспорта и увеличению затрат.

В результате проведённого исследования разработана система «1С: Логист», обеспечивающая полный цикл управления перевозками с возможностью автоматизированного расчета оптимальных

маршрутов на основе внешнего вычислительного модуля. Система реализована на платформе «1С: Предприятие 8» и включает в себя два ключевых компонента: учетно-управляющую конфигурацию и сервер взаимодействия. Учетная часть обеспечивает ведение нормативно-справочной информации, распределение заявок, назначение транспорта и контроль статусов движения грузов на всех этапах перевозки. Сервер взаимодействия, разработанный как внешний программный модуль, выполняет сложные вычислительные задачи: на основе исходных данных о грузах, транспорте и точках маршрута производится оптимизация по критериям времени, стоимости и загрузки.

Техническая реализация базируется на использовании платформы «1С: Предприятие» для построения клиент-серверной учетной системы. Сервер взаимодействия разработан с помощью ASP.NET Web Core API и интегрируется с 1С посредством HTTP-сервисов. Такой подход обеспечивает масштабируемость, разделение учетных и вычислительных функций, а также возможность дальнейшего расширения математического аппарата.

Новизна решения заключается в комплексной автоматизации транспортной логистики в среде 1С с выносом ресурсоемких оптимизационных расчетов на отдельный сервер взаимодействия. Внедрение системы позволит исключить ручное планирование маршрутов, повысить прозрачность перевозочного процесса, сократить затраты на топливо и амортизацию, а также обеспечить руководство инструментом для контроля и анализа эффективности логистики.

**А. В. Дудинский, Е. М. Березовская**  
(ГГУ имени Ф. Скорины, Гомель)

## **ИНФОРМАЦИОННОЕ ANDROID-ПРИЛОЖЕНИЕ “GSUAPP” НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ KOTLIN**

Университет – это динамичная образовательная среда, где эффективное взаимодействие между преподавателями и студентами, а также четкое планирование учебного процесса являются ключевыми факторами успеха. В условиях цифровизации всех сфер жизни возрастает потребность в специализированных инструментах, которые

не просто дублируют, а централизуют и оптимизируют учебные процессы. Существующие универсальные календари и мессенджеры часто не учитывают специфику образовательного учреждения, что приводит к рассогласованию расписаний, потере важных объявлений и снижению оперативности коммуникации.

Для решения вышеуказанных задач было разработано мобильное приложение “GSUAPP”. Оно представляет собой целенаправленное цифровое решение, призванное автоматизировать управление расписанием и создать единый, интуитивно понятный канал взаимодействия между преподавательским составом и учащимися.

Основные функции приложения:

1 Пользователи могут просматривать актуальное расписание занятий как для конкретной учебной группы, так и для отдельного преподавателя.

2 Просмотр встроенной ленты новостей для академических объявлений, оповещений о переносах занятий, контрольным срокам по проектам и другой важной информации.

3 Для преподавателей предусмотрена авторизация в приложении, которая открывает возможность создавать записи в ленте для эффективного взаимодействия со студентами.

Приложение разработано в среде разработки Android Studio с использованием языка Kotlin. Пользовательский интерфейс разработан с использованием языка программирования XML. Мобильное приложение использует базу данных Firebase Realtime Database и библиотеку Apache POI языка Java для работы с файлами.

**М. Д. Екимова**

*(ГрГУ имени Янки Купалы, Гродно)*

## **АВТОНОМНОЕ КРАСНОЕ ТЕСТИРОВАНИЕ СИСТЕМ ИИ: ПРИМЕНЕНИЕ LLM ДЛЯ СИМУЛЯЦИИ АТАК И ОЦЕНКИ ЗАЩИТНЫХ МЕХАНИЗМОВ**

В условиях широкого внедрения систем ИИ возрастает необходимость методов оценки их устойчивости к адаптивным атакам [2].

Для диалоговых систем на основе больших языковых моделей характерны уязвимости, связанные с обработкой контекста и вероятностной генерацией ответов [1].

Одним из перспективных направлений является автономное красное тестирование, при котором атакующие сценарии формируются автоматически.

Целью работы является построение формальной модели автономного LLM-агента, реализующего адаптивный поиск уязвимостей на основе истории взаимодействия с тестируемой системой.

Пусть  $S$  – тестируемая система, отображающая множество запросов  $Q$  в множество ответов  $R$ .

Пусть  $H_i = \{(Q_0, R_0), \dots, (Q_{i-1}, R_{i-1})\}$  – история взаимодействия к моменту шага  $i$ . Агент  $A$  формирует запрос как функцию истории:  $Q_i = A(H_i), R_i = S(Q_i)$ . Для количественной оценки вводится функция уязвимости  $F : R \rightarrow \mathbb{R}_{\geq 0}$ , где значение  $F(R_i)$  характеризует степень отклонения ответа от политики безопасности.

Процесс тестирования рассматривается как задача максимизации суммарного индекса уязвимости на конечном горизонте  $n$ :

$$\max_A \sum_{i=0}^n F(R_i), \text{ при } R_i = S(A(H_i)).$$

Метод реализуется посредством инициализации запроса, итеративной генерации воздействий на основе истории взаимодействия и адаптации стратегии до достижения критерия останова.

Предложенный подход формализует автономное красное тестирование как задачу адаптивного управления и может применяться при оценке защищенности интеллектуальных систем.

## Литература

1 Гудфеллоу, И. Глубокое обучение / И. Гудфеллоу, Й. Бенжио, А. Курвилль. – М.: ДМК Пресс, 2018. – 653 с.

2 Сомова, М. В. «Безопасность и уязвимости больших языковых моделей» / М. В. Сомова, Д. А. Олешко // Научные высказывания. – 2025. – Т. 77, № 9. – С. 35–39.

**Д. С. Ермак**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ПРОЕКТИРОВАНИЕ РАСЧЕТНО-СПРАВОЧНОГО МОДУЛЯ ДЛЯ ПРИЛОЖЕНИЯ «ПОМОЩНИК ВРАЧА-НЕОНАТОЛОГА»**

Работа врача-неонатолога характеризуется высокой ответственностью и необходимостью оперативного принятия решений на основе большого объема данных. В настоящее время неонатологи осуществляют поиск актуальных клинических протоколов и стандартов вручную через различные печатные и онлайн источники, а для расчетов критических параметров используются разные онлайн-калькуляторы, что увеличивает временные затраты и риск ошибок. Внедрение современных информационных технологий, объединенных в единый специализированный инструмент, является необходимым условием для повышения эффективности и безопасности работы.

Целью данной работы является проектирование и реализация расчетно-справочного модуля для приложения помощник врача-неонатолога, который объединит в себе в том числе функции клинического калькулятора и централизованной базы медицинских протоколов, соответствующих актуальным стандартам неонатологической помощи.

Разрабатываемый модуль позволит проводить точные расчеты дозировок лекарственных средств, параметров инфузионной и нутритивной поддержки с автоматическим учетом параметров новорожденного. Также встроенная справочно-протокольная система, предоставит быстрый структурированный доступ к клиническим рекомендациям, стандартам лечения вместо их ручного поиска в разных источниках. В результате планируется снижение рутинных ошибок за счет автоматизации вычислений, встроенной проверки вводимых значений на соответствие безопасным диапазонам и предоставления врачу единой точки доступа к актуальной нормативной информации.

Разработка модуля – ключевой элемент цифровизации неонатологии. Этот инструмент позволит повысить оперативность, точность и безопасность работы врача. Реализация требует тщательного проектирования алгоритмов, экспертной валидации, а также обеспечения надёжности, удобства и защиты данных.

**А. В. Захаров, Е. М. Березовская**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ИНТЕРАКТИВНОГО WEB-ПРИЛОЖЕНИЯ «РЕСТОРАННЫЙ ПОРТАЛ» С ИСПОЛЬЗОВАНИЕМ REACT, NODE.JS И POSTGRESQL**

В современном ресторанном бизнесе успех зависит не только от качества кухни, но и от удобства взаимодействия с гостями. Традиционные методы бронирования и заказа через телефонные звонки устарели и часто приводят к ошибкам, потерям времени и недовольству клиентов. Разработка полнофункционального веб-приложения, объединяющего систему бронирования столиков, онлайн-заказ блюд и личные кабинеты пользователей, позволит ресторанам автоматизировать процессы, повысить конверсию и улучшить клиентский опыт.

Разработанное веб-приложение «Ресторанный портал» ориентировано в первую очередь на рестораны среднего и премиум-сегмента, которые стремятся предложить гостям современный цифровой сервис. По сравнению с существующими агрегаторами доставки, приложение отличается глубокой интеграцией с внутренними процессами заведения, уникальной системой виртуального выбора столиков и полноценной системой лояльности.

Веб-приложение состоит из нескольких ключевых модулей: интерактивная система бронирования столиков; онлайн-меню с возможностью заказа; личные кабинеты пользователей; административная панель управления. Все модули логически связаны между собой для обеспечения seamless-опыта.

Например, при выборе столика через интерактивную карту зала пользователь сразу может сделать предзаказ блюд, которые будут готовы к его приходу. В личном кабинете сохраняется история посещений и предпочтений, что позволяет персонализировать предложения.

Проект разработан с использованием React для создания динамического пользовательского интерфейса, Node.js для серверной логики и PostgreSQL для надежного хранения данных. Применение современных веб-технологий обеспечивает высокую производительность, безопасность и кросс-платформенную доступность.

Разработанное решение позволяет ресторанам не только повысить операционную эффективность, но и создать уникальный цифровой опыт для гостей, что становится критически важным конкурентным преимуществом в современном рынке HoReCa.

**И. А. Зеленков, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ВЫСОКОПРОИЗВОДИТЕЛЬНОГО API-СЕРВИСА ПРОГНОЗИРОВАНИЯ ПОГОДЫ С ИНТЕГРАЦИЕЙ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ**

В современном web-программировании наблюдается устойчивая тенденция к интеграции алгоритмов машинного обучения в микросервисную архитектуру. Основной проблемой при этом становится обеспечение высокой производительности сервиса, так как выполнение инференса нейронных сетей является ресурсоёмкой операцией. Целью работы является создание API-сервиса для почасового прогноза погоды с использованием фреймворка FastAPI и предобученной LSTM-модели.

Архитектура разработанного приложения строится на базе асинхронного фреймворка FastAPI, который использует стандарт ASGI. Это позволяет обрабатывать множество запросов одновременно, не блокируя основной поток ввода-вывода. В качестве ядра системы прогнозирования используется рекуррентная нейронная сеть (RNN) архитектуры LSTM (Long Short-Term Memory), реализованная с помощью библиотеки TensorFlow/Keras. Модель обучается на исторических данных и способна предсказывать температуру на 24 часа вперед.

Ключевым аспектом реализации серверной части является оптимизация жизненного цикла приложения. Загрузка «тяжелой» модели машинного обучения и инициализация нормализатора данных (MinMaxScaler) вынесены в отдельный менеджер контекста lifespan. Это гарантирует, что модель загружается в оперативную память единожды при старте сервиса, а не при каждом входящем запросе. Для решения проблемы блокировки цикла событий (Event Loop) при выполнении синхронных вычислений (инференс модели) используется

механизм `run_in_thread-pool`, который позволяет выполнять CPU-bound операции в отдельном пуле потоков, сохраняя асинхронную природу обработки HTTP-запросов. Важным компонентом системы является подсистема кэширования, реализованная на базе легковесной реляционной СУБД SQLite и ORM SQLAlchemy. Для минимизации количества обращений к внешним источникам данных и снижения вычислительной нагрузки на сервер был внедрен механизм TTL (Time-To-Live).

**К. А. Зеленский, Л. К. Титова**  
(ГГТУ имени П. О. Сухого, Гомель)

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ «АВТОМАТИЗАЦИЯ ТЕХНИЧЕСКОГО ОБСЛУЖИВАНИЯ ПРОМЫШЛЕННОГО ОБОРУДОВАНИЯ» С ИСПОЛЬЗОВАНИЕМ ASP.NET CORE, REACT VITE, POSTGRESQL**

В условиях цифровизации производства растёт необходимость автоматизации процессов технического обслуживания оборудования. На крупных предприятиях ежедневно формируются заявки на ремонт, профилактику и диагностику, а использование бумажных журналов и разрозненных систем затрудняет контроль и увеличивает время обработки заявок.

Разработанное веб-приложение позволяет регистрировать и отслеживать заявки, назначать исполнителей, контролировать статус работ и хранить историю обслуживания, обеспечивая прозрачность процессов и удобство работы сотрудников.

Серверная часть реализована на *ASP.NET Core*, которая принимает и обрабатывает запросы, управляет бизнес-логикой и следит за правами доступа. Для хранения данных используется *PostgreSQL* – там сохраняются все заявки, информация о пользователях и оборудовании, при этом база надежная и легко масштабируется, если объем информации вырастет. Клиентская часть сделана с *React* и сборщиком *Vite*. Благодаря этому интерфейс работает быстро и плавно, страницы загружаются мгновенно, а сами элементы можно легко менять и добавлять новые функции без больших переделок. В итоге получился удобный и отзывчивый интерфейс, которым легко пользоваться на разных устройствах.

Особенностью проекта является модуль генерации технических отчетов на основе генеративных языковых моделей. После выполнения работ система формирует структурированный отчет с описанием неисправности, выполненных действий и использованных материалов, что сокращает время подготовки документации и обеспечивает единый стиль.

Веб-приложение поддерживает поиск и фильтрацию заявок, формирование статистики по видам неисправностей и оборудованию, а также предусматривает возможность расширения функционала. Решение способствует повышению эффективности работы технических служб и цифровизации процессов обслуживания.

**Е. Г. Землянский, И. А. Соболев**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ ФОРМИРОВАНИЯ И ТРЕКИНГА ПЕРСОНАЛЬНЫХ ПРИВЫЧЕК НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ KOTLIN**

В наше время существует постоянная необходимость в саморазвитии и поддержании дисциплины. И если раньше контроль за привычками подразумевал использование бумажных носителей, то на сегодняшний день широкое распространение получили мобильные трекеры – удобный и технологичный способ мониторинга прогресса. Такие приложения позволяют автоматизировать процесс контроля за повседневной деятельностью и обеспечивают мгновенный доступ к статистике личных достижений. Высокая популярность подобных решений обоснована их постоянным наличием под рукой и простотой взаимодействия. В связи с этим и было разработано мобильное Android-приложение «Atomic» для систематизации и закрепления полезных привычек.

При разработке использовался стек технологий Kotlin и Jetpack Compose. Kotlin – кроссплатформенный язык, являющийся стандартом разработки под Android благодаря лаконичности кода.

Jetpack Compose – декларативный фреймворк, позволяющий строить интерфейс из автономных компонентов, что упрощает проектирование и снижает количество ошибок.

Приложение обладает понятным интерфейсом и состоит из трех функциональных блоков. Компонент настройки обеспечивает ввод параметров привычки и их сохранение в локальной базе данных для работы офлайн. Компонент трекинга визуализирует прогресс в виде карточек и «серий» выполнения, позволяя фиксировать результат в одно касание. Модуль взаимодействия реализует систему push-уведомлений, а также функцию экспорта достижений в социальные сети через Android Intent API. Использование данного стека технологий позволило создать производительный инструмент с элементами геймификации для эффективного достижения персональных целей.

### Литература

1 Жемеров, Д. Kotlin в действии / Д. Жемеров, С. Исакова. – М.: ДМК Пресс, 2018. – 402 с.

**В. А. Зинович**

*(ГГУ имени Ф. Скорины, Гомель)*

## **РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ ОСНОВЫ ВЕБ-ПРИЛОЖЕНИЯ «ОНЛАЙН-КИНОТЕАТР» НА ОСНОВЕ RESTFUL АРХИТЕКТУРЫ И ТЕХНОЛОГИИ КОНТЕЙНЕРИЗАЦИИ DOCKER**

Стремительный рост популярности стриминговых сервисов диктует высокие требования к производительности и масштабируемости. Традиционные монолитные архитектуры уступают место контейнеризованным решениям, ускоряющим разработку и развертывание.

В докладе приводятся результаты разработки веб-приложения «Онлайн-кинотеатр» с использованием RESTful архитектуры и технологии контейнеризации. Программный комплекс построен на трехзвенной архитектуре и включает клиентскую часть, серверное приложение и базу данных. Серверная платформа реализована на фреймворке ASP.NET Core 8, обеспечивающем высокую производительность обработки HTTP-запросов и построение масштабируемого API.

Взаимодействие с PostgreSQL реализовано посредством ORM Entity Framework Core, что позволило использовать объектно-ориентированный подход к управлению данными [1].

Клиентская часть разработана с использованием библиотеки React, обеспечивающей создание одностраничного интерфейса (SPA). Для стилизации использован Tailwind CSS. Аутентификация пользователей реализована на основе JWT-токенов, гарантируя безопасность передачи данных.

Ключевой особенностью является полная контейнеризация компонентов с использованием Docker. Написаны Dockerfile для серверного и клиентского приложений с технологией многоэтапной сборки для минимизации размера образов. Docker Compose объединяет серверную часть, клиентскую часть (обслуживаемую Nginx) и базу данных в изолированную сеть, гарантируя идентичность окружения разработки и эксплуатации.

В результате создан программный продукт, реализующий регистрацию и авторизацию пользователей, навигацию по каталогу и непосредственный просмотр фильмов, а также управление личными списками просмотра.

### **Литература**

1 Троелсен, Э. Язык программирования C# 7 и платформы .NET и .NET Core / Э. Троелсен, Ф. Джепикс; пер. с англ. – 8-е изд. – СПб.: ООО «Диалектика», 2018. – 1328 с.

**Н. Д. Зубрик**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ПРИМЕНЕНИЕ СИСТЕМ ПРЕДОТВРАЩЕНИЯ УТЕЧЕК ДАННЫХ ДЛЯ МОНИТОРИНГА И ФИЛЬТРАЦИИ СЕТЕВОГО ТРАФИКА**

В современной практике защиты информации главное значение имеет контроль исходящих потоков данных. Основными объектами мониторинга и фильтрации в данном случае выступают протоколы передачи гипертекста, почтовые сервисы и протоколы передачи файлов. Одной из наиболее важных задач специалиста по информационной

безопасности является разработка алгоритмов автоматизированной классификации трафика для предотвращения случайного или умышленного разглашения персональных данных [1].

Основой современных систем предотвращения утечек конфиденциальных данных является комплексный анализ контента и контекста передаваемых сообщений. Процесс идентификации нарушений основывается на сопоставлении хеш-функций документов, применении технологий оптического распознавания символов и контекстном поиске по заданным словарям. Регулярное обновление цифровых отпечатков существенно повышает вероятность обнаружения скрытых каналов утечки. Благодаря комплексной оценке каждого информационного объекта возникает итоговый показатель риска. Это позволяет построить четкую границу между стандартными процессами и аномальной активностью, которая свидетельствует о наличии внутренней угрозы.

Таким образом, сетевой трафик, в котором присутствуют признаки передачи конфиденциальных данных, подвергается автоматической блокировке или помещается в карантин для ручного разбора. Параллельно с этим, легитимная информация проходит через защищенный шлюз без создания задержек в работе пользователей. Данный подход сочетает в себе высокую производительность сети с надежностью политик обеспечения информационной безопасности.

### Литература

1 Олифер, В. Г. Компьютерные сети. Принципы, технологии, протоколы: учебник для вузов / В. Г. Олифер, Н. А. Олифер. – 6-е изд. – СПб.: Питер, 2020. – 992 с.

**Е. В. Иванова, П. В. Бычков**  
(ГГУ имени Ф. Скорины, Гомель)

### **РАЗРАБОТКА ВЕБ-СЕРВИС ПО ДОСТАВКЕ ТОВАРА ПОТРЕБИТЕЛЮ**

В современных условиях наличие специализированного интернет-ресурса для предприятий доставки является обязательным требованием рынка: потребители ожидают оперативного дистанционного оформления заявок и мониторинга грузов в реальном времени.

Цель данной работы - разработка сервиса доставки с применением технологического стека JavaScript, C#, MongoDB и Angular, где Angular будет обеспечивать отзывчивость интерфейса, а MongoDB выступать в качестве базы, для управления данными. Система будет включать три модуля: личный кабинет заказчика, реестр отправок и трекер статусов.

Функционирование сервиса базируется на разграничении прав между четырьмя типами пользователей. Клиент после регистрации получает доступ к истории заявок (активные заявки будут расположены вверху списка), может создавать новые заказы с выбором товаров и способа оплаты, а также отслеживать статус. Курьер видит карточки активных заявок, при отсутствии закреплённого заказа, может принять свободный заказ кнопкой «Назначить на себя», и последовательно обновлять статус доставки. Диспетчер имеет доступ к общей таблице заявок с сортировкой (активные вначале), и кнопкой ручного создания заявки для случаев телефонного обращения клиента. Администратор управляет учётными записями: просматривает полный список, редактирует поля, удаляет записи с принудительной деавторизацией удаляемого пользователя.

Практическая значимость данного подхода заключается в сокращении временных затрат на коммуникацию между участниками логистической цепочки. Автоматизация назначения заявок и прозрачность статусов минимизируют необходимость в телефонных уточнениях. Данный сервис станет инструментом повышения качества обслуживания и оптимизации логистических процессов, при помощи разделения ролей, и создания функциональной системы для всех участников доставки.

**А. С. Исаченко, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ МОНИТОРИНГА И ВИЗУАЛИЗАЦИИ КОСМИЧЕСКОГО МУСОРА С ИСПОЛЬЗОВАНИЕМ PYTHON, FASTAPI, REACT**

В современном мире проблема засорения околоземного пространства становится всё более острой, создавая угрозы для функционирующих спутников и пилотируемых миссий. В связи с

Этим разработана доступных инструментов для мониторинга и анализа космического мусора является актуальной задачей, способствующей повышению осведомленности и безопасности космической деятельности. Данная работа посвящена созданию веб-приложения для отслеживания объектов космического мусора в реальном времени. Проект нацелен на обработку открытых данных с последующей их визуализацией в интуитивно понятном для пользователя интерфейсе.

В качестве серверной части используется язык Python [1] и фреймворк FastAPI, который обеспечивает высокую производительность и простоту создания асинхронных API для обработки и подачи данных. Клиентская часть реализована с использованием современной библиотеки JavaScript – React, что позволяет создавать динамичный и отзывчивый интерфейс. Архитектура системы построена по принципу разделения фронтенда и бэкенда. FastAPI выступает в роли промежуточного слоя: он получает данные из внешних открытых источников, обрабатывает их и предоставляет их REST API. React-приложение, в свою очередь, запрашивает эти данные и отображает их в браузере пользователя, предоставляя инструменты для фильтрации и наглядной визуализации.

В результате с помощью вышеуказанных технологий было разработано веб-приложение, позволяющее в реальном времени отслеживать положение космического мусора на орбите, предоставляя пользователям актуальную информацию из открытых источников в удобном графическом интерфейсе.

### **Литература**

1 Васильев, А. Программирование на Python в примерах и задачах / А. Васильев. – М.: Эксмо, 2021. – 616 с.

**П. П. Кавцевич**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ПРОЕКТИРОВАНИЕ МОДУЛЬНОЙ АРХИТЕКТУРЫ ЛОКАЛЬНОГО ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ СЕМАНТИЧЕСКОЙ ОБРАБОТКИ ТЕКСТОВ**

Рост объемов неструктурированной текстовой информации требует новых подходов в работе с ней, исключая зависимость

от облачных решений. В данной статье предлагается разработка архитектуры локального программного комплекса по принципу интеграции систем каталогизации, интеллектуального ядра, графовых баз данных и среды управления знаниями, объединенных сквозными потоками данных. На этапе агрегации используется Calibre для нормализации цифровых источников и Libib для учета печатных изданий. Это позволяет сформировать единый слой метаданных. Технологическим фундаментом системы является архитектура Retrieval-Augmented Generation, позволяющая AI-агенту извлекать релевантные контексты из локальной базы данных перед генерацией верифицированного ответа [1]. На основе этого механизма агент извлекает смыслы и сущности, преобразуя неструктурированные данные в систему логических объектов. Важным аспектом является использование графовых баз данных. На этапе обработки AI-агент не просто извлекает текст, а формирует топологию связей, где каждая идея становится узлом, а контекстуальное сходство – ребром графа. Использование графовой модели данных позволяет эффективно обрабатывать многомерные связи, которые теряются при классическом реляционном подходе.

Центральной операционной средой выступает Obsidian. Он функционирует не только как интерфейс, но и как система управления базой знаний, где результаты анализа преобразуются в сеть атомарных заметок. Преимущество такой архитектуры заключается в создании бесшовного конвейера, где использование графовых алгоритмов и открытых форматов обеспечивает независимость модулей.

### **Литература**

1 Retrieval-Augmented Generation (RAG): глубокий технический обзор: [site]. – URL: <https://habr.com/ru/articles/931396/>. (date of access: 23.01.2026).

**В. В. Казека, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

### **КЛИЕНТСКАЯ ЧАСТЬ WEB-ПРИЛОЖЕНИЯ «ИНТЕРНЕТ-МАГАЗИН МУЗЫКАЛЬНЫХ ИНСТРУМЕНТОВ»**

Онлайн-магазины музыкальных инструментов предоставляют покупателям широкий спектр возможностей по подбору и приобретению

желаемых товаров, что положительно сказывается на реализации продукции. Удобный, или, как сейчас актуально называть, «user-friendly» интерфейс позволяет пользователю быстро находить необходимые вкладки, разделы и потенциальные фильтры.

Взаимодействие пользователя и web-приложения, как правило, сопровождается пояснениями на соответствующих элементах страницы.

Клиентская часть сайта разработана с использованием:

- HTML5 и фреймворка Bootstrap – для создания структуры страницы, семантической разметки, доступности интерфейса;

- CSS3 и препроцессоров SASS и LESS – для стилизации и поддержки различных состояний интерфейса, адаптивной верстки с использованием Flexbox и Grid, оптимизации производительности рендеринга;

- JavaScript – для обработки событий, динамического обновления страницы, анимации интерфейса, управления историей браузера при фильтрации в SPA-подобных интерфейсах, кастомных слайдеров и каруселей, фильтрации и сортировки каталога на стороне клиента, работы с Fetch API. HTML5, CSS3 и JavaScript.

В приложении реализован следующий функционал:

- *каталог и навигация*: вывод товаров с пагинацией, фильтрация по параметрам, сортировка, поиск, история просмотров;

- *карточки товаров*: галерея изображений, переключение вариантов, отображение остатков и статуса «в наличии», таблица характеристик, блок отзывов и рейтинга;

- *корзина*: добавление, удаление и изменение количества товаров, пересчёт суммы и скидок, отложенные товары;

- *личный кабинет*: регистрация и авторизация, восстановление пароля, история заказов, статус доставки;

- *оформление заказа*: многошаговая форма, валидация полей в реальном времени, выбор способа доставки и оплаты, подстановка адреса по подсказкам, подтверждение и отправка заказа;

- *админ-панель*: управление заказами, добавление и редактирование товаров, управление акциями и промокодами, складской учёт.

**Е. О. Калинкова**

*(ГрГУ имени Янки Купалы, Гродно)*

## **МОДЕЛИРОВАНИЕ И ОЦЕНКА ЗАЩИЩЁННОСТИ СЕГМЕНТИРОВАННОЙ СЕТЕВОЙ ИНФРАСТРУКТУРЫ СИСТЕМ ВИДЕОНАБЛЮДЕНИЯ НА ОСНОВЕ ПРОГРАММНЫХ ЭМУЛЯТОРОВ**

Развитие концепций IoT в корпоративной среде повышает актуальность защиты распределенных систем видеонаблюдения (CCTV). Передача видеопотока по IP-сетям делает их уязвимыми к перехвату трафика, DoS-атакам и компрометации VMS-серверов. Согласно стандарту BS 7799-3, эффективное управление рисками требует систематической оценки мер защиты [1].

В работе предложен подход к моделированию защищенной инфраструктуры с использованием эмуляторов VirtualBox и GNS3. На Ubuntu Server развернута система ZoneMinder. В GNS3 реализована сегментация зон безопасности: серверы, рабочие места, внешние сети.

Ключевым аспектом исследования стала практическая реализация принципа «эшелонированной защиты» (Defense-in-Depth). Моделирование включало три взаимосвязанных уровня: базовый контроль сетевого доступа через настройку правил межсетевого экрана UFW; проактивное противодействие атакам на подбор паролей с помощью системы Fail2ban; и фундаментальный слой, состоящий из харденинга операционной системы и СУБД MySQL для устранения уязвимостей конфигураций по умолчанию.

Эксперименты на стенде включали оценку устойчивости к сетевому сканированию и имитации атак типа «brute-force». Результаты подтвердили, что сегментация в сочетании с настроенными средствами защиты нейтрализует большинство типовых угроз на уровне серверного периметра.

Дальнейшие исследования направлены на интеграцию модуля анонимизации персональных данных методами машинного обучения, что обусловлено требованиями Закона Республики Беларусь от 10 ноября 2008 г. № 455-З «Об информации, информатизации и защите информации».

## Литература

1 BS 7799-3:2006. Information security management systems. Part 3: Guidelines for information security risk management. – BSI, 2006. – 70 p.

**Г. Г. Каравацкий**

*(ГрГУ имени Янки Купалы, Гродно)*

## КАЧЕСТВЕННАЯ ОПТИМИЗАЦИЯ АГЕНТНЫХ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ СЕЛФ-ПРОМПТИНГА

Агентная система (ИИ-агент) представляет собой верхнеуровневую систему, объединяющую функциональные возможности искусственного интеллекта и классических автономных систем [1]. Основным предназначением ИИ-агентов выступает автоматизация задач, требующих методологическое вмешательство со стороны человека.

Настоящей проблематикой определена точность работы системы. Одним из множества факторов, оказывающих неблагоприятное влияние на качество результата, представлена неверная интерпретация промпта, вызванная некорректными семантическими ассоциациями, где промпт – установленная текстовая инструкция.

В качестве решения проблемы неверной интерпретации промпта выступает технология селф-промптинга. Технология основывается на принципе составления промпта самим искусственным интеллектом.

Предлагается использовать рассматриваемую технологию для оптимизации разработки ИИ-агентов (рисунок 1).

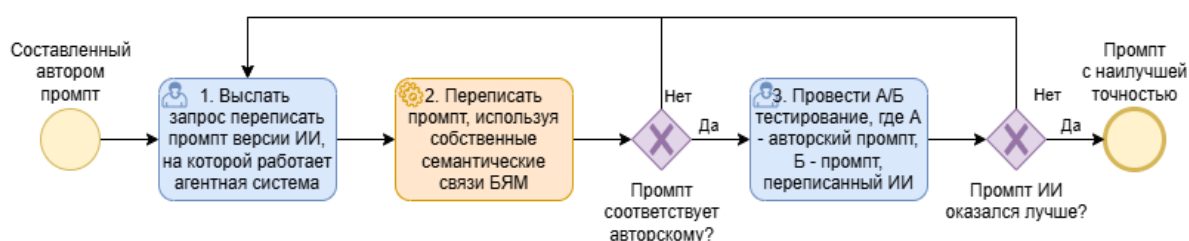


Рисунок 1 – Процесс имплементации селф-промптинга

Имплементация селф-промптинга в рамках разработки ИИ-агентов позволит повысить качество работы системы. Однако,

Материалы XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 23–25 марта 2026 г.

важно подчеркнуть, что при смене большой языковой модели (БЯМ) технология не будет работать в той же мере качественно, потому как каждая версия БЯМ имеет разные семантические ассоциации.

### Литература

1 ИИ агенты и Агентный ИИ: концептуальная таксономия, приложения и проблемы: [сайт]. – URL: <https://www.sciencedirect.com/science/article/pii/S1566253525006712?via%3Dihub> (дата обращения: 18.02.2026).

**Г. Л. Карасёва, З. Р. Тагангылыджова**  
(ГГУ имени Ф. Скорины, Гомель)

### **СОЗДАНИЕ ПРИЛОЖЕНИЯ «ДОСТОПРИМЕЧАТЕЛЬНОСТИ РЕСПУБЛИКИ БЕЛАРУСЬ» В СРЕДЕ C++ BUILDER С ИСПОЛЬЗОВАНИЕМ БД ACCESS**

Беларусь богата своими достопримечательностями. Природные чудеса, родовые замки, усадьбы: все они оставляют неизгладимое впечатление на граждан республики и наших гостей. Однако, туристический сервис в Республике Беларусь не очень развит на данный момент. Не хватка удобных маршрутов, вопросы с туристической инфраструктурой и т.д. всё это создаёт проблемы для потенциальных туристов. Приложение дает возможность узнать о прекрасных местах нашей Родины и проложить маршрут их посещения.

Работа посвящена разработке приложения «Достопримечательности Республики Беларусь» с использованием языка программирования C++, БД Microsoft Access в среде Builder 6.0 с помощью технологии ADO. На главной панели расположено изображение территории Республики Беларусь и элементы навигации, представленные в виде кнопок с названиями областных центров Республики Беларусь: «Брест», «Гродно», «Гомель», «Минск», «Могилёв», «Витебск» и кнопка «Закрыть».

После выбора одного из областных центров открывается список интересных мест рассматриваемого объекта. При нажатии на выбранный элемент списка появляется его изображение, а также описание и интересные факты, позволяющие пользователю узнать много интересного (рисунок 1).

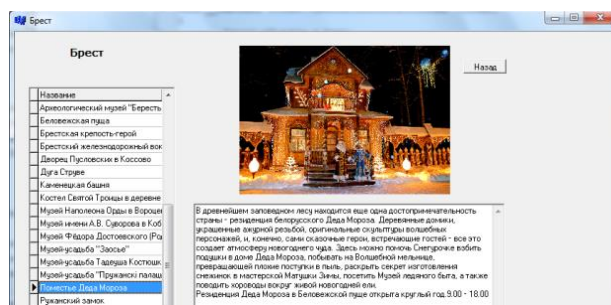


Рисунок 1 – Вид окна со списком достопримечательностей

Создано визуальное приложение для просмотра описания интересных мест Республики Беларусь, которое реализует отображение изображений и записей. Изображения меняются с интервалом в 8 секунд, для удобного просмотра.

**К. С. Караханова, Е. Ю. Кузьменкова**  
(ГГУ имени Ф. Скорины, Гомель)

## РАЗРАБОТКА ВЕБ-САЙТА СОЦИАЛЬНО-БЛАГОТВОРИТЕЛЬНОГО УЧРЕЖДЕНИЯ С ИСПОЛЬЗОВАНИЕМ JAVASCRIPT, REACT

В современном мире веб-приложение для благотворительной организации становится ключевым решением для формирования доверия и оперативного информирования общества, так как отсутствие качественного цифрового представительства ограничивает возможности фонда в привлечении пожертвований и освещении достигнутых успехов.

Разработанный веб-сайт социально-благотворительного учреждения предоставляет пользователям возможность ознакомиться с миссией фонда, изучить информацию о предстоящих мероприятиях и просмотреть отчеты о реализованных проектах. Неравнодушные люди могут находить актуальные реквизиты для совершения пожертвований и оставлять заявки на участие в волонтерских программах, в то время как администраторы учреждения могут оперативно добавлять новые информационные блоки, редактировать новости, загружать фотографии и отслеживать статус поступивших обращений.

Для разработки приложения была выбрана библиотека React.js, которая обеспечивает высокую производительность и гибкость интерфейса. Визуальная часть создана с помощью HTML и CSS, а для обеспечения интерактивности и динамического обновления контента используется JavaScript. Модульная архитектура приложения, характерная для React, позволяет эффективно разделять логику представления и управления данными.

При посещении приложения пользователи могут детально изучить историю создания фонда и текущие направления его деятельности. Ресурс содержит актуальную ленту новостей и подробные отчеты об успехах, а для удобства участников программ на сайте размещены верифицированные данные для совершения финансовых транзакций. Реализованные формы обратной связи позволяют волонтерам и нуждающимся получать необходимую информацию в реальном времени. Таким образом, наличие единой цифровой платформы с прозрачной отчетностью значительно улучшает качество взаимодействия с аудиторией и повышает уровень общественного доверия к благотворительному учреждению.

**Е. Ю. Карпович**

*(ГрГУ имени Янки Купалы, Гродно)*

## **РАЗРАБОТКА СИСТЕМЫ АВТОМАТИЗАЦИИ УПРАВЛЕНИЯ СКЛАДСКИМИ ЗАПАСАМИ**

Система автоматизации управления складскими запасами предназначена для оптимизации учёта товарно-материальных ценностей, контроля остатков и повышения эффективности логистических процессов предприятия с несколькими складами. В рамках дипломного проекта разрабатывается прикладная информационная система, обеспечивающая поддержку основных операций складского учёта и анализа движения запасов.

Функциональность системы включает ведение номенклатурного справочника, регистрацию операций поступления, перемещения и списания товаров, контроль остатков в разрезе складов и партий, а также автоматизированное формирование заявок на закупку на основе нормативов минимального и максимального уровня запасов.

Запланирована реализация инструментов анализа оборачиваемости, выявления дефицита и излишков, а также формирования аналитической отчетности, включая ABC/XYZ-классификацию.

Проектирование выполнено с использованием типовых объектов платформы 1С: Предприятие 8: справочников, документов, регистров сведений и накоплений. Для реализации интерфейс планируется использовать управляемые формы, а для реализации бизнес-логики – средства встроенного языка 1С. Для формирования отчетности будет применена система компоновки данных, которая обеспечивает гибкость настройки аналитических форм.

Для разработки используется платформа 1С: Предприятие версии 8.3. В итоге будет получена законченная прикладная конфигурация, ориентированная на использование в условиях малого и среднего бизнеса. Архитектура решения обеспечит масштабируемость, адаптацию под специфику складских процессов и возможность дальнейшего расширения, включая интеграцию с системами закупок и бухгалтерским учётом.

**В. Д. Качура, Е. М. Березовская**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ПРИЛОЖЕНИЙ “VIRTUAL PIANO” И “MUSICAL CALCULATOR” С ИСПОЛЬЗОВАНИЕМ ИНТЕГРИРОВАННОЙ СРЕДЫ РАЗРАБОТКИ DELPHI**

На сегодняшний день существует множество приложений для создания музыки – от профессиональных DAW до мобильных драм-машин. Однако подавляющее большинство из них направлено либо на запись готового звука, либо на симуляцию виртуальных студий. Доля приложений, которые объясняют музыкальную теорию через код и инженерию, крайне мала. При этом спрос на понимание основ музыки среди начинающих разработчиков и просто энтузиастов растет: многие хотят не просто нажимать кнопки, а понимать, как рождается нота, частота и аккорд.

Целью данной работы являлись анализ, проектирование и реализация серии приложений, формирующих у начинающих пользователей понимание музыкальной теории через инструменты

низкоуровневого программирования и классические языки разработки. Основная идея – воссоздать «прозрачную» среду обучения, где каждый этап генерации звука или расчета нотной последовательности доступен для изучения и модификации пользователем.

С помощью языка C и инструментов 8-битного синтеза было разработано приложение “Virtual piano”. Программа эмулирует работу простейшего звукового генератора, управляя частотами колебаний на уровне аппаратных абстракций. Пользователь получает возможность не только прослушивать ноты, но и наблюдать, как значение частоты преобразуется в звуковой сигнал. Это позволяет сформировать фундаментальное понимание природы музыкального тона.

На основе объектно-ориентированного подхода в среде Delphi было создано графическое приложение “Musical calculator”. Интерфейс построен с использованием стандартных компонентов визуальной среды разработки. Приложение выполняет функции построения интервалов и аккордов на основе вводимых пользователем нот, выступая в роли справочно-вычислительного инструмента для быстрого построения гармонических последовательностей. Для обеспечения доступности и кроссплатформенности разработана консольная версия музыкального калькулятора в среде Visual Studio Code. Консольное приложение, реализованное на языке Delphi, сохраняет полную функциональность графической версии, но работает в текстовом режиме. Это решение позволяет пользователям с ограниченными аппаратными ресурсами или предпочтением минималистичного интерфейса изучать музыкальную теорию без потери качества обучения.

Все приложения объединены общей педагогической целью: сделать музыкальную теорию доступной, наглядной и вычислительно прозрачной для начинающих музыкантов и разработчиков.

**Е. В. Киптик, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА КЛИЕНТСКОЙ ЧАСТИ ВЕБ-ПРИЛОЖЕНИЯ “DIGIHELPER” С ИСПОЛЬЗОВАНИЕМ ФРЕЙМВОРКА BLAZOR**

Современные веб-приложения предъявляют высокие требования к интерактивности, производительности и удобству пользовательского

интерфейса. Это особенно актуально для многопользовательских бизнес-систем, работающих в реальном времени, где важны быстрый отклик интерфейса, синхронизация состояния и безопасность данных. В работе рассматривается разработка клиентской части веб-приложения “DigiHelper»” с использованием фреймворка Blazor.

Blazor является фреймворком платформы .NET, позволяющим создавать интерактивные веб-интерфейсы на языке C# с применением компонентного подхода. В проекте использована модель Blazor Server, при которой выполнение интерфейса осуществляется на сервере, а изменения состояния передаются в браузер через постоянное соединение с использованием SignalR.

Клиентская часть приложения построена на основе Razor Components, объединяющих разметку, состояние и обработчики событий, что обеспечивает модульность интерфейса и упрощает сопровождение проекта. Взаимодействие с серверной частью осуществляется через HTTP-API с использованием типизированных клиентов на основе библиотеки Refit.

Разработанное веб-приложение “DigiHelper” предназначено для автоматизации работы онлайн-магазинов цифровых товаров. Реализованы функции регистрации и авторизации пользователей, управления аккаунтом и тарифным планом, настройки двухфакторной аутентификации для товаров, а также их автоматической выдачи и управления лимитами.

Приложение предоставляет инструменты аналитики и визуализации данных, включая календари выплат и рекламных расходов, отчёты о прибыли и показатели активности. Реализованы модули автоматизации рассылок сообщений покупателям, реферальная программа и система уведомлений.

**Н. Л. Киселев**  
(БГУ, Минск)

## **ИСПОЛЬЗОВАНИЕ НЕЙРОННЫХ СЕТЕЙ ПРИ УПРАВЛЕНИИ АВТОНОМНЫМ ТРАНСПОРТНЫМ СРЕДСТВОМ**

Цель исследования – создание и сравнительный анализ подходов машинного обучения с подкреплением для управления

Материалы XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 23–25 марта 2026 г.

автономным транспортным средством в симуляторе CARLA [1]. Задача поставлена как решение марковского процесса принятия решений (MDP).

Осуществлено внедрение двух алгоритмов: Deep Q-Network (DQN) с дискретным набором действий и метод Actor-Critic с возможностью непрерывного управления. Для DQN спроектирована свёрточная сеть с использованием механизмов остаточных связей, состоящая из пяти блоков (рисунок 1).



Рисунок 1 – Работа DQN-агента и структура сети ResidualCNN

Реализация выполнена на языке Python с применением фреймворков TensorFlow/Keras и API версии CARLA 0.9.16.

По результатам исследования установлено, что DQN проявляет большую устойчивость при движении по стандартным маршрутам, тогда как модель Actor-Critic эффективнее справляется с вызовами в сложных условиях – во время дождя и в ночное время.

### Литература

1 Dosovitskiy, A. CARLA: An Open Urban Driving Simulator / A. Dosovitskiy [et al.] // Proc. of the 1st Annual Conf. on Robot Learning. – 2017. – P. 1–16.

**И. К. Киселёв**

*(ГрГУ имени Янки Купалы, Гродно)*

## **МЕТОДЫ И АЛГОРИТМЫ ПЕРСОНАЛИЗАЦИИ КОНТЕНТА В ЭЛЕКТРОННОЙ КОММЕРЦИИ ДЛЯ РОСТА КЛЮЧЕВЫХ БИЗНЕС-ПОКАЗАТЕЛЕЙ**

В современном мире электронной торговли стратегии персонализации контента, основанные на технологиях искусственного интеллекта, становятся ключевым конкурентным преимуществом. Актуальность исследования связана с необходимостью прогнозирования потребительского поведения и формирования актуальных рекомендаций через автоматизированные системы. Согласно ряду исследований, персонализация позволяет значительно повысить эффективность взаимодействия с клиентами и улучшить бизнес-показатели [1; 2].

Использование алгоритмов машинного обучения и нейронных сетей играет важную роль в сегментации аудитории и создании точных персонализированных предложений. Эти методы позволяют анализировать большие массивы данных о поведении пользователей, что подтверждается современными работами в области big data и цифровой экономики [3; 4]. В результате компании получают возможность повышать конверсию, увеличивать продажи и снижать уровень отказов, что также отмечается в исследованиях по персонализированному маркетингу [5].

### **Литература**

1 Кузнецов, В. Н. Персонализация в электронной коммерции: подходы и технологии / В. Н. Кузнецов. – М.: Инфра-Инженерия, 2020. – 256 с.

2 Лебедев, А. В. Искусственный интеллект в маркетинге: тенденции и прогнозы / А. В. Лебедев, Т. С. Иванова. – СПб.: Питер, 2021. – 192 с.

3 Шкирюк, А. А. Большие данные и их влияние на бизнес: новые горизонты / А. А. Шкирюк, Э. И. Мосеев. – Каз.: КНИТУ, 2021. – 300 с.

4 Соловьев, И. Г. Анализ потребительского поведения в цифровой экономике / И. Г. Соловьев. – Екб.: УрФУ, 2019. – 220 с.

5 Устинов, А. Ю. Персонализированный маркетинг: от идеи к практике / А. Ю. Устинов. – М.: Альпина Паблишер, 2018. – 180 с.

**А. А. Климов**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА 2D-ШУТЕРА С ЭЛЕМЕНТАМИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА НА ОСНОВЕ ФРЕЙМВОРКА MONOGAME**

Современные игровые движки (Unity, Unreal Engine) скрывают от разработчика низкоуровневые процессы, что упрощает вход в индустрию, но ограничивает понимание базовых принципов их работы. Использование фреймворка MonoGame позволяет получить полный контроль над архитектурой приложения. Актуальность работы заключается в поиске баланса между гибкостью разработки и производительностью системы в условиях ограниченных ресурсов.

Для достижения максимальной производительности применен механизм плотного хранения со ссылочным доступом. Это оптимизирует данные для обработки процессором и исключает частые вызовы сборщика мусора за счет работы со значимыми типами и передачи данных по ссылке.

Реализована система виртуальной камеры с матричными преобразованиями координат. Интерфейс пользователя выделен в отдельный экземпляр ECS, что изолирует графику UI от игровой логики и упрощает масштабирование проекта.

Боевая система основывается на технологии Raycasting (бросания лучей), что обеспечивает мгновенную и точную регистрацию попаданий, а также хорошую производительность.

Искусственный интеллект (ИИ) реализован как реактивная система, сочетающая два алгоритма: патрулирования и стрельбы. Для патрулирования ИИ анализирует поверхность перед собой с помощью лучей. При обнаружении края платформы или препятствия направление движения инвертируется. Для стрельбы система проверяет прямую видимость между игроком и врагом с помощью бросания лучей. Кроме этого, в алгоритм стрельбы заложены погрешности, делая ИИ более естественным и щадящим.

Разработанный программный инструментарий подтверждает эффективность применения ECS-архитектуры в 2D-приложениях на C#. Удалось добиться высокой модульности: новые игровые механики добавляются через создание изолированных систем, что позволяет не изменять существующий код без необходимости.

**В. С. Коваленко, Е. Ю. Кузьменкова**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА КНИЖНОГО ИНТЕРНЕТ-МАГАЗИНА “LIBERTY” С ИСПОЛЬЗОВАНИЕМ REACT, NODE.JS**

Интернет-магазины уже давно занимают важное место в жизни человека. С каждым годом потребность в совершении покупок онлайн продолжает расти, в сети Интернет появляется множество новых интернет-магазинов. Они предлагают разнообразный выбор товаров для дома, одежду, технику, образовательные ресурсы и многое другое.

У многих людей любимым времяпровождением является чтение книг. Поэтому книжные интернет-магазины продолжают пользоваться популярностью. Они дают возможность быстро находить нужные книги, сравнивать товары разных производителей по стоимости и другим параметрам, знакомиться с отзывами и оценивать качество предлагаемых изданий. Кроме того, такие магазины дают доступ к зарубежным изданиям и позволяют совершать покупки, не покидая дома.

Для разработки клиентской части интернет-магазина был выбран фреймворк React, обеспечивший высокую производительность приложения и удобство создания интерфейса. Он позволил реализовать динамичные компоненты и эффективное обновление данных. Для оформления и адаптивности интерфейса использовался CSS.

Для реализации backend-части приложения был выбран Node.js, который предоставил возможность создать серверное приложение с использованием языка JavaScript. В рамках проекта Node.js использовался для обработки пользовательских запросов, работы с базой данных в SQL Server Management Studio.

Разработанный книжный интернет-магазин «Liberty» позволяет пользователям зарегистрироваться и создать аккаунт, просматривать каталог, искать книги и фильтровать их по различным параметрам. Понравившиеся издания можно сохранять и добавлять в корзину для оформления заказа. У каждой книги предусмотрена отдельная страница с аннотацией и дополнительными характеристиками. Также доступна страница с информацией о физическом книжном магазине.

**К. С. Кожемякин, С. И. Ленденкова**  
(ГГУ имени Ф. Скорины)

## **РАЗРАБОТКА ПЛАГИНА ДЛЯ INTELLIJ IDEA**

Современные инструменты разработки значительно изменили подходы к созданию ПО.

В условиях стремительного развития методологий разработки возникает потребность в эффективных средствах автоматизации рутинных операций и расширения функциональности сред программирования.

В частности, платформа IntelliJ IDEA становится незаменимой экосистемой для создания плагинов, которые позволяют разработчикам кастомизировать поведение IDE, внедрять поддержку новых языков и осуществлять глубокий анализ кода в реальном времени с высоким уровнем интеграции.

Основной целью работы является реализация плагина для IntelliJ IDEA, предназначенного для автоматизации генерации шаблонного кода и улучшения навигации по проекту. Такой плагин может предоставить разработчикам простые и удобные инструменты для создания типовых архитектурных компонентов.

Например, данное расширение будет полезно и удобно для автоматического прототипирования REST-контроллеров, генерации DTO-объектов или создания каркасов микросервисов.

В качестве инструментов разработки используются язык программирования Java, SDK IntelliJ Platform, а также система сборки Gradle.

Разрабатываемое расширение имеет интуитивно понятный интерфейс, вписывающийся в эргономику IntelliJ IDEA, который будет доступен для разработчиков любого уровня.

Актуальность работы заключается в необходимости создания удобного инструмента для разработчиков и тимлидов, который будет способствовать ускорению процессов написания кода, унификации решений в команде и поддержке инженеров в их повседневной работе по созданию сложных программных продуктов.

**В. В. Козликовская, Е. М. Березовская**  
(ГГУ имени Ф. Скорины, Гомель)

## **ПОДКЛЮЧЕНИЕ СЕРВИСА MAILJET ДЛЯ ОТПРАВКИ КОДА ПОДТВЕРЖДЕНИЯ ПРИ РЕГИСТРАЦИИ В МОБИЛЬНОМ ПРИЛОЖЕНИИ “FOR ALL INTO LIFE”**

Регистрация пользователей является важным этапом работы мобильных приложений, в том числе “For All Into Life” (далее “FAIL”), обеспечивающим безопасный доступ к его функциональным возможностям. В современных цифровых сервисах для подтверждения учетной записи широко используется механизм отправки одноразового кода на электронную почту пользователя, позволяющий повысить уровень защиты данных и предотвратить несанкционированный доступ.

В приложении “FAIL” процесс регистрации открывает доступ к персонализированным рекомендациям и функциональным модулям, связанным с управлением финансами, документами и правовыми аспектами взрослой жизни, что требует применения надежного механизма аутентификации. Использование одноразовых кодов подтверждения обеспечивает корректную идентификацию пользователей и снижает риски компрометации учетных записей.

Технология подтверждения регистрации на основе одноразовых кодов (ОТР – One-Time Password) предполагает генерацию уникального временного кода, предназначенного для однократного использования в течение ограниченного промежутка времени. Передача кода осуществляется по защищенному каналу связи, что исключает возможность его повторного применения и повышает общий уровень информационной безопасности системы.

Для реализации механизма отправки кодов подтверждения в мобильном приложении используется почтовый сервис Mailjet, предоставляющий API для автоматизированного формирования и отправки электронных сообщений. Интеграция может быть реализована с использованием REST-запросов или протокола SMTP, что позволяет гибко настраивать параметры доставки, управлять шаблонами писем и отслеживать статус отправленных сообщений. Поддержка мониторинга доставки и обработки ошибок обеспечивает

стабильную работу системы регистрации и минимизирует задержки при получении кодов пользователями. Интеграция сервиса Mailjet в “For All Into Life” представляет собой важный шаг к созданию надежной платформы для молодежи, предоставляющей пользователям защищенный доступ к необходимым инструментам для комфортного вступления во взрослую жизнь.

**Е. В. Козловская, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ УПРАВЛЕНИЯ И ХРАНЕНИЯ ПАРОЛЕЙ С ИСПОЛЬЗОВАНИЕМ C#, UWP**

В условиях растущего числа кибератак и утечек данных проблема безопасного управления учётными данными остаётся критической для конечных пользователей. Традиционные подходы к хранению паролей (заметки, документы) являются уязвимыми, а облачные менеджеры паролей нередко вызывают опасения относительно приватности данных. Разработанное приложение решает эту проблему, предоставляя локальное, автономное хранилище паролей с полным контролем пользователя над своими данными.

Приложение разработано на базе UWP (UI-фреймворк для Windows) с использованием архитектурного паттерна MVVM (Model-View-ViewModel). Основным языком программирования – C# с применением фреймворка ASP.NET Core [1] для backend-логики.

Для организации зависимостей в приложении применяется Dependency Injection.

Основной функционал приложения включает: управление паролями (создание, редактирование, удаление и поиск учётных данных с метаданными), генератор паролей, валидация паролей, управление профилем пользователя, импорт/экспорт данных (возможность резервного копирования и восстановления хранилища в защищённом формате), многоязычная поддержка (интерфейс адаптирован для работы на нескольких языках).

Разработанное приложение сочетает в себе высокий уровень конфиденциальности и современные архитектурные решения. Полная локальность хранения данных гарантирует, что информация

пользователя остается исключительно на его устройстве. Чистая архитектура на основе MVVM и внедрения зависимостей обеспечивает простоту расширения функционала и проведения модульного тестирования. Интуитивный пользовательский интерфейс, с удобной навигацией между разделами делает работу с приложением комфортной для пользователей любого уровня.

### **Литература**

1 Лок, Э. ASP.NET Core в действии: пер. с англ. / Э. Лок. – 3-е изд. – М.: Диалектика, 2023. – 912 с.

**А. Р. Коледа**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ГЕНЕРАЦИЯ РЕСУРСОВ ДЛЯ 2D-ИГР С ПРИМЕНЕНИЕМ УСЛОВНЫХ ГЕНЕРАТИВНО-СОСТЯЗАТЕЛЬНЫХ СЕТЕЙ**

Приложение представляет собой реализацию системы генерации 2D-ресурсов с применением условной генеративно-состязательной сети (сGAN). Первоначально подключаются необходимые библиотеки для работы с нейронными сетями, обработки изображений и выполнения математических операций. Далее настраиваются параметры обучения: размер изображений, размер батча, количество эпох и скорость обучения.

Программа включает два ключевых компонента – генератор (Generator) и дискриминатор (Discriminator). Генератор принимает на вход случайный вектор шума и дополнительное условие (например, класс или тип игрового объекта) и формирует изображение заданного размера. Его архитектура построена на последовательности слоёв, постепенно преобразующих входные данные в полноценное 2D-изображение.

Дискриминатор, в свою очередь, получает изображение и соответствующее условие и определяет, относится ли оно к реальным данным из обучающего набора или было создано генератором. Он выполняет извлечение признаков и формирует вероятность принадлежности изображения к реальным данным.

За процесс обучения модели отвечает отдельный блок программы. В ходе обучения дискриминатор минимизирует ошибку классификации реальных и сгенерированных изображений, а генератор стремится улучшить качество генерации так, чтобы дискриминатор не мог отличить ‘поддельные’ изображения от настоящих. После завершения обучения модель может использоваться для генерации новых 2D-игровых ресурсов по заданным параметрам.

Таким образом, программа представляет собой модульную систему генерации графических ассетов, которую можно адаптировать под различные типы 2D-игр и стили визуального оформления.

Дополнительно предусмотрена возможность сохранения и повторной загрузки обученной модели, а также гибкой настройки параметров, что обеспечивает удобство повторного использования и адаптацию системы под различные задачи генерации 2D-ресурсов.

**И. А. Колеснева, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **КЛИЕНТСКАЯ ЧАСТЬ БЛОГА «МОЯ КУЛИНАРНАЯ ЛАБОРАТОРИЯ»**

В условиях активного развития информационных технологий и повсеместного распространения сети Интернет web-приложения заняли важное место в повседневной жизни пользователей. Одним из востребованных направлений является разработка web-ресурсов, предназначенных для хранения, обработки и представления информации в удобном и доступном виде. Одним из таких ресурсов являются кулинарные блоги, которые дают пользователям возможность находить рецепты, изучать технологии приготовления блюд и обмениваться с другими пользователями кулинарным опытом.

Разработка клиентской части осуществлялась с использованием современных web-технологий: HTML5 для структурирования контента, CSS3 для стилизации и оформления интерфейса, а также фреймворк Bootstrap для создания адаптивного дизайна.

Клиентская часть реализует функционал просмотра рецептов, отображения подробной информации о каждом блюде и взаимодействия с системой аутентификацией. Данные динамически

загружаются из базы данных и выводятся пользователю в структурированном виде. Каждый рецепт содержит следующую информацию: название, перечень ингредиентов и способ приготовления. В системе реализована авторизация пользователей с разграничением прав доступа. После входа в систему интерфейс адаптируется в зависимости от роли пользователя. Обычные пользователи имеют возможность просматривать рецепты и взаимодействовать с дополнительным функционалом, администратор получает доступ к инструментам управления контентом.

Дополнительной функцией клиентской части является раздел «Избранное». Авторизованные пользователи могут добавлять понравившиеся рецепты в персональный список избранных блюд, что повышает удобство использования системы, позволяя сохранять рецепты для быстрого доступа. Информация об избранных рецептах сохраняется в базе данных и привязывается к конкретному пользователю, что обеспечивает персонализацию контента. В приложении реализована проверка корректности вводимых данных в формах, что позволяет минимизировать ошибки при передаче информации на сервер и повышает надежность работы приложения.

**Н. И. Кондратьев, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ ПЛАТФОРМЫ ДЛЯ ПРОДАЖИ И ПРОВЕДЕНИЯ ОБУЧАЮЩИХ КУРСОВ С ИСПОЛЬЗОВАНИЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА**

В современных условиях цифровизации образования наблюдается стремительный рост спроса на онлайн-обучение и автоматизированные образовательные решения. По данным аналитических исследований, мировой рынок электронного обучения демонстрирует ежегодный прирост более 20 %, что обусловлено развитием дистанционных технологий и изменением образовательных моделей. В этом контексте актуальной становится задача создания интеллектуальных платформ, способных не только предоставлять учебный контент, но и адаптивно взаимодействовать с обучающимися.

Целью данной работы является проектирование и реализация автоматизированной платформы для коммерциализации и организации обучающих курсов с применением технологий искусственного интеллекта. Платформа предназначена для решения комплекса задач: управление каталогом курсов, обработка платежных транзакций, организация образовательного процесса, интеллектуальная поддержка пользователей через ИИ-ассистента, аналитика учебной активности.

Для реализации системы выбран современный технологический стек. В качестве основы фронтенд-части используется фреймворк Next.js. Пользовательский интерфейс реализован с применением дизайн-системы Tailwind CSS. Для интеграции возможностей искусственного интеллекта задействованы внешние API больших языковых моделей, развёрнуты локальные LLM-решения для обработки чувствительных данных.

Разработанная автоматизированная платформа для организации и монетизации образовательного контента с интеграцией искусственного интеллекта позволит создателям курсов эффективно управлять образовательными программами, а обучающимся – получать персонализированную поддержку. Применение ИИ-технологий обеспечивает качественно новый уровень взаимодействия участников образовательного процесса, автоматизируя рутинные операции и концентрируя внимание на содержательных аспектах обучения.

**Д. О. Кононенко, В. А. Короткевич**  
(ГГУ имени Ф. Скорины, Гомель)

## **ПРИЛОЖЕНИЕ ДЛЯ ВИЗУАЛИЗАЦИИ ДАННЫХ О ПОДАЧЕ ЗАЯВЛЕНИЙ В ПРИЕМНУЮ КОМИССИЮ ВУЗА**

Целью данной работы являлось расширение функциональности программного комплекса «Приемная комиссия ГГУ им. Ф. Скорины». Реализованное с этой целью интернет-приложение обеспечивает просмотр обобщенной информации о ходе приемной комиссии в университет. Приложение предназначено для использования ответственными сотрудниками приемной комиссии и руководством университета. Авторизация пользователей приложения ведется по протоколу LDAP с использованием средств авторизации доменной сети ГГУ.

Приложение позволяет просматривать данные о планах приёма, поданных заявлениях и результатах предварительного зачисления. Поддерживается фильтрация по формам обучения (дневная/заочная), формам оплаты (бюджет/платно) факультетам и специальностям. Для специальностей отображается распределение абитуриентов по баллам, количество абитуриентов с различными видами льгот, текущие проходные баллы. Показываются данные как для целевого приёма, так и основного набора. Реализован просмотр текущих недоборов на специальности, для платного обучения недоборы показываются с прогнозом переноса документов с бюджетной на платную форму обучения. Дополнительно в приложении реализован просмотр поданных заявлений за текущие сутки, а также общий просмотр всех заявлений за период приёмной кампании. Предусмотрен просмотр анкетных данных абитуриентов, что позволяет анализировать контингент поступающих и оперативно получать необходимую справочную информацию.

Клиентская часть приложения реализована с использованием фреймворка Angular, основанного на языке программирования JavaScript. Для построения пользовательского интерфейса также используется CSS-фреймворк Bootstrap. Серверная часть разработана на основе фреймворка FastAPI, реализованного на языке программирования Python. Для работы с базой данных совместно с FastAPI применяется ORM-библиотека SQLAlchemy. В качестве системы управления базами данных используется СУБД MS SQL Server, обеспечивающий надёжное хранение, целостность и быстрый доступ к данным вступительной кампании. Реализовано взаимодействие между клиентской и серверной частями приложения по архитектуре «клиент–сервер».

**Л. Л. Конопко**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ГЕНЕРАЦИЯ ТЕСТ-КЕЙСОВ ДЛЯ ФУНКЦИОНАЛЬНОГО ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Функциональное тестирование остаётся ключевым этапом обеспечения качества программных систем, однако ручная подготовка

тест-кейсов трудоёмка и подвержена пропускам требований. В работе рассматривается подход к автоматизированной генерации тест-кейсов для функционального тестирования, ориентированный на воспроизводимость, полноту покрытия и сокращение времени регрессионных проверок. Метод сочетает модельно-ориентированные представления предметной области с анализом входных доменов и ограничений, а также применяет эвристики минимизации набора тестов при фиксированных метриках покрытия.

Постановка задачи формулируется так: из множества кандидатных тестов необходимо выбрать минимальный набор, который закрывает все функциональные требования. Для каждого теста заранее известен перечень требований, которые он проверяет. Получение точного оптимального решения является вычислительно сложной задачей (аналог задачи покрытия множества), поэтому используется эвристический отбор. На основе NLP-анализа и гибридной модели машинного обучения разработан интеллектуальный ассистент QA инженера, позволяющий ускорять и стандартизировать процесс функционального тестирования. Экспериментальная валидация проведена на трёх сервисных модулях (CRUD-API, модуль расчётов и модуль конфигураций) с суммарно около 120 функциональными требованиями. Базовая линия – ручные тест-кейсы по спецификации. Предложенный генератор повысил покрытие требований с 82 % до 96 % и сократил суммарное время регрессионного прогона на 28 % за счёт устранения дубликатов и автоматической приоритизации по риску изменений. Доля тестов, не дающих вклада в покрытие, снизилась с 23 % до 9 %.

В дальнейшем планируется добавить вероятностное ранжирование по историческим данным дефектов, критерии адекватности тестов (mutation score) для контроля «сильного» покрытия и интеграцию оракулов на основе контрактов и проверок инвариантов исполнения.

**Д. А. Копоть**

*(ГрГУ имени Янки Купалы, Гродно)*

## **О РАЗРАБОТКЕ ХОСТИНГ-ПЛАТФОРМЫ**

Хостинг является важной составляющей цифровой инфраструктуры, обеспечивающей функционирование веб-ресурсов,

информационных систем и онлайн-сервисов. Он играет значительную роль в развитии бизнеса, электронной коммерции и цифровых услуг, предоставляя доступность, надежность и масштабируемость веб-приложений. Качественный хостинг способствует стабильной работе сайтов, защите данных и улучшению пользовательского опыта.

В работе рассматриваются подходы к разработке эффективной хостинг-платформы, включающей надежную серверную инфраструктуру, системы мониторинга, резервного копирования и обеспечения информационной безопасности. Особое внимание уделяется использованию современных технологий виртуализации и контейнеризации, а также автоматизации процессов управления ресурсами и развертывания сервисов.

Разработка хостинг-платформы предполагает интеграцию инструментов управления, таких как панели администрирования, системы биллинга и технической поддержки. Важную роль играет обеспечение масштабируемости и отказоустойчивости, позволяющих адаптироваться к изменяющимся нагрузкам и требованиям клиентов. Использование аналитических инструментов дает возможность отслеживать производительность, выявлять узкие места и оптимизировать использование вычислительных ресурсов.

Таким образом, создание современной хостинг-платформы требует комплексного подхода, включающего надежность, безопасность, удобство администрирования и поддержку актуальных технологий. Это способствует развитию цифровой экономики, расширению возможностей для бизнеса и созданию благоприятных условий для разработки и эксплуатации веб-решений различной сложности.

**Д. А. Король**

*(ГрГУ имени Янки Купалы, Гродно)*

## **РАЗРАБОТКА ЗАЩИЩЕННОГО ВЕБ-СЕРВИСА ДЛЯ ПОИСКА СВЕДЕНИЙ ОБ ОРГАНИЗАЦИЯХ ПО УНП**

Справочные веб-сервисы для поиска организаций по УНП и наименованию востребованы при проверке контрагентов и получении регистрационных сведений. При открытом доступе возрастает риск злоупотреблений: автоматизированные массовые запросы могут приводить к исчерпанию вычислительных и сетевых ресурсов,

а некорректная интеграция с внешними источниками – к снижению доверия к данным. OWASP подчёркивает угрозу неограниченного потребления ресурсов через API, поэтому контроль нагрузки и доступа рассматривается как базовый элемент защиты.

Цель разработки – создание устойчивого веб-сервиса, реализующего сценарий «поиск → результаты → карточка организации». Оптимальна архитектура клиент–сервер: клиент отвечает за интерфейс и отображение результатов, сервер предоставляет REST API, валидирует параметры, выполняет запросы к базе данных и агрегирует сведения из внешних реестров. В качестве источника актуальных данных целесообразно использовать REST API ГРП и REST API ЕГР.

Защитный контур включает: аутентификацию по API-ключу в заголовке запроса; ограничение частоты обращений и таймауты при взаимодействии с внешними сервисами; строгую серверную валидацию УНП и параметров поиска; параметризованные запросы к БД; минимизацию информативности сообщений об ошибках для предотвращения раскрытия внутренней структуры. Реализация может опираться на типовые механизмы безопасности FastAPI [3]. Надёжность обеспечивается единым контрактом ответов и регламентированной обработкой состояний «не найдено», «внешний источник недоступен», «превышен лимит», а также журналированием для выявления аномалий и настройки ограничений.

### **Литература**

1 FastAPI. Security. Описание схем аутентификации и механизмов защиты API. FastAPI.: [сайт]. – URL: <https://fastapi.tiangolo.com/tutorial/security/>. (дата обращения: 05.02.2026).

**В. Д. Кузьменков, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

### **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ «КАРТОТЕКА ФИЛЬМОВ» С ИСПОЛЬЗОВАНИЕМ HTML5, CSS3, JAVASCRIPT**

В современном мире интернет является самым простым и доступным способом получения информации о различных аспектах жизни человека, используется для быстрого поиска различного контента, в том числе информации о фильмах.

Для конечного пользователя важным является не только информация о фильме, но и рецензии, оценки выбранного фильма другими пользователями. Для разработки веб-приложения «Картотека фильмов» были выбраны современные веб-технологии: HTML5 – для семантической вёрстки, поисковой оптимизации, CSS3 – для разработки привлекательного пользовательского интерфейса, JavaScript [1] – для адаптивной вёрстки, для организации взаимодействия с СУБД, для реализации необходимых функций веб-приложения. В качестве СУБД была использована MySQL.

Разработанное веб-приложение позволяет просматривать подробную информацию о выбранном фильме: описание, актеры, жанр, год выхода на экран, бюджет, режиссер и др.

Также реализованы функции поиска фильмов по заданным критериям, фильтрации, сортировки и др. Разработана панель администратора для добавления информации о фильмах, её изменения, удаления и др. Реализована возможность добавлять фильмы в избранное, оценивать фильмы, оставлять рецензии на фильмы, оценивать рецензии других пользователей.

Веб-приложение позволяет ранжировать фильмы по рейтингу. Также реализована возможность просмотра информации об актерах, режиссерах, для каждого фильма добавлены постеры и скриншоты для получения более подробной информации о фильме.

Веб-приложение «Картотека фильмов» обладает удобным пользовательским интерфейсом, адаптировано для использования на мобильных устройствах.

### Литература

1 Флэнаган, Д. ECMAScript 6. Подробное руководство. / Д. Флэнаган. – М.: Символ-Плюс, 2018. – 984 с.

**М. А. Лашкин, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА BACK-END ЧАСТИ ОБУЧАЮЩЕЙ ПЛАТФОРМЫ С ИНТЕГРАЦИЕЙ ФУНКЦИЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА**

Современные обучающие платформы все чаще выходят за рамки простого хранения учебных материалов и тестов. Для повышения

эффективности обучения используются системы, способные анализировать действия пользователей, учитывать результаты обучения и адаптировать образовательный процесс под конкретный случай. В связи с этим особый интерес представляет разработка обучающей платформы с поддержкой функций искусственного интеллекта, а именно back-end части, так как она отвечает за основной функционал продукта.

Разработка back-end части приложения выполнена с использованием языка программирования Java [1] и фреймворка Spring Boot, что позволило реализовать модульную и расширяемую архитектуру приложения.

Для работы с данными применяются Spring Data и система управления базами данных PostgreSQL. Взаимодействие с сервисами искусственного интеллекта осуществляется через REST-интерфейсы, а также Spring AI, а для обработки ресурсоёмких операций используется асинхронный подход. Развёртывание и сопровождение приложения упрощены за счёт использования средств контейнеризации таких как Docker и Docker-Compose.

Интеграция функций искусственного интеллекта позволила реализовать автоматизированную обработку результатов обучения и анализ активности пользователей. На основе собранных данных платформа формирует рекомендации, направленные на устранение пробелов в знаниях и оптимизацию учебного процесса.

Такой подход снижает нагрузку на преподавателей и повышает качество обратной связи для обучающихся. Разработанная часть платформы может быть использована в качестве основы для создания образовательных систем с элементами искусственного интеллекта.

Полученные результаты подтверждают целесообразность применения технологий искусственного интеллекта при проектировании современных обучающих сервисов.

### Литература

1 Блинов, И. Н. Java from EPAM / И. Н. Блинов, В. С. Романчик. – Мн.: Четыре четверти, 2020. – 560 с.

**Д. С. Лебедев, И. С. Войтешенко**  
(БГУ, Минск)

## **ОПТИМИЗАЦИЯ ПОСТОЯННЫХ СОЕДИНЕНИЙ В SERVICE MESH**

В распределённых системах широко применяется архитектура service mesh, которая помогает обеспечить наблюдаемость и контроль трафика, не изменяя код приложения. Однако в большинстве решений взаимодействие между сервисами в меш сети проходит через прокси, что увеличивает время передачи сообщений и потребляемые ресурсы. Целью данной работы является исследование проблем производительности service mesh и разработка библиотеки, которая позволит сократить время передачи данных с помощью постоянных соединений на примере WebSocket. Разработанная библиотека управляет постоянными соединениями на уровне приложения, избавляя от прохода трафика через прокси сервера. Также она обладает интерфейсом, совместимым со стандартным, что облегчает её внедрение в готовые приложения. В таблице 1 представлены результаты тестирования времени обработки межсервисных сообщений от интенсивности нагрузки (RPS – количество запросов в секунду) и размера тела запроса. Показаны 50-й, 90-й, 95-й и 99-й перцентили времени обработки. Эксперимент проводился на ноутбуке с процессором Apple M4 и 24 ГБ оперативной памяти. Во всех случаях наблюдается снижение времени выполнения запросов, эффект усиливается при передаче данных большего объёма.

Таблица 1 – Результаты нагрузочного тестирования

Запрос	RPS	Тело, KB	p50, мс	p90, мс	p95, мс	p99, мс
Стандартный	600	0	2	4	5	8
	400	100	7	11	14	22
	200	1024	21	31	36	48
С помощью библиотеки	200	0	1	3	3	5
	400	100	4	7	9	16
	200	1024	15	25	31	44

**И. А. Лебедев, А. Б. Демуськов**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ANDROID-ПРИЛОЖЕНИЯ “EASYGO” ДЛЯ ОПРЕДЕЛЕНИЯ МАРШРУТА НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ KOTLIN**

Сегодня активно развиваются различные технологии, призванные сделать жизнь людей комфортнее. Тем не менее, люди старшего возраста часто не успевают за техническим прогрессом и склонны отвергать новшества из-за их кажущейся сложности и непонятности. Горожане постоянно пользуются общественным транспортом, чтобы преодолевать как большие, так и совсем короткие расстояния, однако далеко не каждый может легко разобраться в схемах движения. В связи с этим была предложена идея создания Android-приложения, способного помочь пользователю быстро проложить удобный маршрут при поездке от одной остановки до другой.

Данное приложение разрабатывается в среде разработки Android Studio: здесь находятся все инструменты, которые позволят довольно легко создать приложение для Android.

Для создания Android-приложения можно использовать языки Java и Kotlin. Данное приложение разрабатывается на языке программирования Kotlin. Kotlin позволяет создавать объектно-ориентированный код, в котором используются классы, наследование и полиморфизм, как и в языке Java. Но Kotlin также поддерживает функциональное программирование.

На данный момент разработан пользовательский интерфейс на XML и частично реализована логика работы на Kotlin. В приложении используются **Google Maps** для отображения остановок. Пользователь может задать начальную и конечную остановку, после чего система подберет подходящий общественный транспорт для передвижения между ними.

Для реализации навигационной части приложение использует **Google Maps API** – набор программных интерфейсов, который позволяет встраивать интерактивные карты прямо в приложение и управлять их отображением. С помощью этого API разработчик может добавлять метки остановок, обрабатывать выбор пользователя и получать координаты выбранных точек, что делает взаимодействие с картой максимально гибким.

**А. М. Лисай, Т. А. Поть**  
(ГрГУ имени Янки Купалы, Гродно)

## **НЕЙРОСЕТЕВЫЕ МОДЕЛИ ДЛЯ ДИАГНОСТИКИ ГЕМАНГИОМ ПЕЧЕНИ НА МРТ-ИЗОБРАЖЕНИЯХ**

Применение методов глубокого обучения в медицинской визуализации позволяет повысить скорость и точность диагностики заболеваний внутренних органов. Особенно актуальной является автоматизация анализа МРТ-изображений печени, где сложная структура тканей и вариабельность патологических очагов затрудняют ручную диагностику.

В рамках работы разработана нейросетевая модель для сегментации и классификации печени и гемангиомы на МРТ-изображениях. В качестве основы использована свёрточная архитектура U-Net, ориентированная на сохранение пространственного контекста и выделение границ объектов. Обучение модели выполнялось на размеченном наборе медицинских изображений с применением аугментации данных и разделением выборки на обучающую и валидационную части. Использовались методы увеличения вариативности входных данных, которые позволяют повысить устойчивость модели к шумам, артефактам МРТ и различиям между пациентами. Реализация осуществлялась с использованием фреймворка PyTorch [1].

Качество работы модели оценивалось по стандартным метрикам семантической сегментации. По результатам тестирования достигнут коэффициент Dice 0,959 для сегментации печени и 0,909 для гемангиомы, при значениях метрики IoU 0,922 и 0,833 соответственно. Полученные показатели свидетельствуют о высокой точности классификации и сегментации печени и гемангиомы.

Разработанный подход может использоваться в системах поддержки принятия врачебных решений и представляет практический интерес для дальнейших исследований в области автоматизированной медицинской диагностики.

### **Литература**

1 Использование нейросети для анализа медицинских изображений. CSI-MED: [сайт]. – URL: <https://www.csi-med.ru/news/625/>. (дата обращения: 23.02.2026).

**В. А. Лисичкин, В. А. Короткевич**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ ПРИЛОЖЕНИЯ ДЛЯ ПОИСКА И ОРГАНИЗАЦИИ МЕРОПРИЯТИЙ**

В настоящее время поиск мероприятий, соответствующих личным интересам пользователей, зачастую сопряжен с анализом разрозненных источников, отсутствием персонализированных рекомендаций и несвоевременными уведомлениями. Для решения этих задач разработано веб-приложение, обеспечивающее интеграцию удобного интерфейса, интеллектуального поиска и системы уведомлений в единую экосистему.

Приложение позволяет пользователям находить мероприятия через два ключевых механизма: теговую фильтрацию с минималистичным дизайном, исключая информационный шум, и AI-движок, анализирующий данные о хобби и предпочтениях пользователя для формирования персонализированных рекомендаций. Это позволяет эффективно удовлетворять как конкретные запросы, так и открыть для пользователя новые, соответствующие его вкусам, события.

Архитектура системы построена на веб-фреймворке Django, Supabase в роли облачной базы данных, гарантирующей масштабируемость и отказоустойчивость, Redis для кеширования часто запрашиваемых данных, Celery и RabbitMQ для асинхронной обработки задач, включая отправку уведомлений о начале событий и обновлении подписок.

Особое внимание уделено системе оповещений:

- пользователи получают мгновенные уведомления при регистрации на мероприятие;
- за час до старта события приходит напоминание с деталями локации и актуальной информацией, генерируемой в фоновом режиме через Celery-воркеры.

Архитектура приложения спроектирована с фокусом на долгосрочную поддержку и простую адаптацию под растущие потребности пользователей. Интеграция с облачными сервисами, такими как Supabase, и использование очередей сообщений, обеспечивают бесперебойную работу и гибкое масштабирование даже при резких всплесках активности.

**А. Д. Маркелов, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА МОДУЛЯ АУТЕНТИФИКАЦИИ И АВТОРИЗАЦИИ ДЛЯ WEB-ПРИЛОЖЕНИЯ “DIGIHELPER”**

Проект «DigiHelper» представляет собой web-приложение с серверной частью в виде REST API, которое предназначено для автоматизации задач цифровых магазинов: работы с данными аккаунта, подключаемых интеграций и сервисных функций. Чтобы доступ к возможностям системы был управляемым, в серверной части реализован отдельный модуль, отвечающий за вход пользователей и проверку их прав при обращении к защищённым методам API.

Пользователь проходит стандартный путь: регистрация и последующий вход. На этапе регистрации, помимо основных учетных данных, в систему заносятся параметры, необходимые для работы магазина (в частности, идентификатор и ключ доступа), а данные формы проверяются. После создания учетной записи пользователь авторизуется по email и паролю: контролируется корректность email, а также задано ограничение на длину пароля (8–16 символов). При успешном входе выданный токен сохраняется в механизме хранения сессии, после чего открывается доступ к функциональности личного кабинета. Взаимодействие клиента с сервером после входа построено на токеной схеме. Сервер использует JWT Bearer-аутентификацию: токен формируется по заданным параметрам и далее проверяется при каждом запросе. В конфигурации задаются ключевые настройки, которые отвечают за валидность токена (включая проверку подписи, срока действия и принадлежности токена нужному приложению). Это позволяет сделать доступ к методам API единообразным: если токен отсутствует или неверен, запрос не проходит к бизнес-логике. Чтобы пользователь мог работать в системе без повторного ввода пароля, предусмотрен механизм обновления сессии через refresh-токены.

Для обработки сбоев применяется общий обработчик исключений: он перехватывает ошибки и возвращает корректные ответы API, не раскрывая лишние внутренние детали. Проверка корректности входных данных реализована через FluentValidation. Для наблюдаемости и анализа событий используется централизованное логирование через Serilog.

**В. Е. Марочкин**

(ГГТУ имени П. О. Сухого, Гомель)

## **РАЗРАБОТКА СЕТЕВОЙ СОЦИАЛЬНОЙ ИГРЫ «УБЕЖИЩЕ» С ИНТЕГРИРОВАННОЙ ЯЗЫКОВОЙ МОДЕЛЮ**

Социальная игра «Убежище» моделирует коллективный отбор в условиях ограниченных ресурсов. Участники получают персонажей с индивидуальными характеристиками (профессия, здоровье, фобии, навыки и др.) и в ходе раундов постепенно раскрывают их, аргументируя своё право остаться в системе. После обсуждения проводится голосование, по итогам которого один игрок исключается. Победителями считаются участники, сохранившие место к завершению игры.

Интеграция *LLM* расширяет функциональность системы за счёт автоматизации ряда интеллектуальных задач. Она может предоставлять справочную информацию по терминам или правилам, а также составлять отчёт по игре.

Система реализуется по клиент-серверной модели. Клиентское приложение отвечает за пользовательский интерфейс и локальное отображение состояния игры. Серверная часть выполняет функции:

- централизованного хранения игрового состояния;
- обработки голосований и переходов между раундами;
- синхронизации данных между участниками;
- взаимодействия с модулем *LLM*.

Обмен данными осуществляется по протоколу *TCP* с использованием постоянного соединения, что обеспечивает гарантированную доставку и корректную последовательность игровых событий.

Клиент-серверная архитектура и централизованное управление состоянием обеспечивают корректную синхронизацию игрового процесса, а модульная структура системы допускает дальнейшее масштабирование и расширение функциональности.

Программная реализация выполняется с использованием игрового движка *Unity* и языка *C#*. Доступ к языковой модели осуществляется посредством API-ключа, хранящегося на серверной стороне.

Разрабатываемая система демонстрирует возможность интеграции языковых моделей в многопользовательскую социальную игру. Предложенный подход носит универсальный характер и может применяться в иных интерактивных или образовательных системах.

**В. Е. Марочкин**  
(ГГТУ имени П. О. Сухого, Гомель)

## **ПРИМЕНЕНИЕ ЯЗЫКОВЫХ МОДЕЛЕЙ БОЛЬШОГО ОБЪЁМА В ПРИКЛАДНЫХ ПРОГРАММНЫХ СИСТЕМАХ**

Языковые модели большого объёма активно внедряются в прикладные программные системы в качестве интеллектуальных модулей обработки естественного языка. Их применение позволяет автоматизировать генерацию текстовых пояснений, аналитических выводов и отчётов без разработки специализированных алгоритмов для каждой задачи.

Интеграция LLM осуществляется через облачный API с использованием ключа авторизации. В архитектурном плане интеллектуальный модуль размещается на серверной стороне, что обеспечивает защиту ключей доступа и централизованный контроль запросов. Сервер формирует контекст запроса, передаёт его в модель и выполняет предварительную обработку полученного ответа.

Качество генерации во многом определяется подготовкой входных данных. Перед отправкой запроса выполняются сбор релевантной информации, нормализация и структурирование данных. Использование формализованных структур и явное указание требуемого формата ответа повышают предсказуемость результата и упрощают его автоматическую обработку.

Одним из типовых сценариев является предоставление справочной информации по специализированным терминам. По запросу пользователя модель может формировать краткое описание понятия и контекст использования. Это особенно актуально в системах, содержащих профессиональную лексику (медицинскую, техническую, юридическую и др.), где пользователю требуется оперативное пояснение терминов.

Другим направлением применения является автоматизированное формирование текстовых отчётов на основе структурированных параметров состояния системы. Модель генерирует связное аналитическое резюме, позволяя представить результаты работы в удобной текстовой форме.

Полученный ответ проходит этап валидации и синтаксического разбора. Проверяется соответствие заданной структуре, извлекаются необходимые поля, при необходимости выполняется повторная генерация.

Таким образом, внедрение LLM расширяет функциональные возможности прикладных программных систем, обеспечивая интеллектуальную обработку текста при сохранении масштабируемости и архитектурной устойчивости решения.

**И. В. Марушкевич**

*(ГГУ имени Ф. Скорины, Гомель)*

## **ПРИМЕНЕНИЕ ТЕХНОЛОГИЙ .NET ДЛЯ СОЗДАНИЯ ИНСТРУМЕНТА ОБРАБОТКИ ГОЛОСОВЫХ ДАННЫХ**

Развитие человеко-машинного взаимодействия требует специализированных инструментов, способных не только фиксировать аудиоинформацию, но и преобразовывать её в различные форматы. Особую ценность приобретают локальные решения, функционирующие в условиях ограниченного или отсутствующего интернет-соединения и гарантирующие конфиденциальность обрабатываемых данных. В ответ на этот запрос разработано приложение “VoiceAppFree”, объединяющее запись звука и синтез речи в едином интерфейсе.

Архитектурной основой приложения выбран паттерн MVVM, реализованный средствами WPF. Интерфейс включает три функциональные вкладки: «Запись звука» с возможностью выбора длительности записи через ползунок (по умолчанию 5 секунд), «TTS Синтез речи» с редактором текста и тестированием системы, а также «История записей» для просмотра и управления сохранёнными данными. Визуальные индикаторы и система уведомлений обеспечивают понятную обратную связь на всех этапах работы.

Ключевым функциональным компонентом стала реализация модуля синтеза речи. При инициализации SpeechSynthesisService автоматически сканирует установленные в системе голоса (GetInstalledVoices) и выбирает голос, соответствующий русскому языку. Процесс синтеза включает настройку параметров скорости и

громкости, перенаправление вывода в `MemoryStream` и последующую упаковку сырых аудиоданных в корректный WAV-формат с генерацией RIFF-заголовка. Реализована функция тестирования TTS-системы для проверки работоспособности перед использованием.

Модуль записи звука использует событийно-ориентированную модель `NAudio`: асинхронно поступающие от системного аудиодрайвера фрагменты данных последовательно записываются в `MemoryStream` через `WaveFileWriter`. После завершения записи формируется детализированная информация: размер файла в удобочитаемом формате, длительность, дата создания. Реализована возможность досрочной остановки записи, сохранения в файл и добавления в историю с автоматическим уведомлением об успешной операции.

**Е. И. Маслак, В. В. Комраков**  
(ГГТУ имени П. О. Сухого, Гомель)

## **ТЕХНОЛОГИЯ RAG В ИНФОРМАЦИОННО-АНАЛИТИЧЕСКИХ СИСТЕМАХ ДЛЯ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА**

Современная концепция генерации с дополненной выборкой, известная как *RAG*, представляет собой инновационный архитектурный метод, который качественно меняет подход к интеллектуальному анализу данных о персонале [1]. Эта технология оптимизирует работу искусственного интеллекта путем внедрения этапа предварительного поиска информации в локальных доверенных источниках перед процессом формирования итогового аналитического заключения. В отличие от традиционных моделей, функционирующих на базе общих знаний, полученных в ходе обучения, система с архитектурой *RAG* задействует динамическую базу знаний организации, включающую профили компетенций, архивы аттестаций, отчеты о производительности и траектории карьерного роста.

Функциональная структура такой информационной системы базируется на трех последовательных этапах обработки данных о людях. На стадии индексации вся входящая информация, включая резюме, оценки 360 градусов и индивидуальные планы развития, разбивается на смысловые фрагменты, которые преобразуются в

математические векторы и сохраняются в специализированном векторном хранилище [3]. Это позволяет алгоритмам осуществлять поиск специалистов не по простым ключевым словам, а по семантическому сходству их опыта поставленным задачам. В момент поступления запроса слой извлечения обращается к накопленным данным и отбирает наиболее релевантные профили или фрагменты характеристик. На финальном этапе языковая модель синтезирует полученный контекст и формулирует комплексный портрет сотрудника или аналитическую записку по команде, строго опираясь на найденные факты и предоставляя ссылки на подтверждающие документы.

### Литература

1 Льюис, П. Генерация с дополненной выборкой для задач обработки знаний в системах обработки естественного языка / П. Льюис, Э. Перес, А. Пиктус А. [и др.]. – Нью-Йорк: Академик Пресс, 2020. – 250 с.

2 Паньков, А. В. Архитектура информационных систем. – М.: Инфра-М, 2021. – 320 с.

**Е. Г. Матеюк**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ВЛИЯНИЕ ОБЯЗАТЕЛЬНОЙ ПЕРИОДИЧЕСКОЙ СМЕНЫ ПАРОЛЕЙ НА СТОЙКОСТЬ АУТЕНТИФИКАЦИИ**

В современных информационных системах базовый контроль доступа основан на аутентификации по паролю. Доказано, что данный вариант смены паролей неэффективен. Частые запросы на смену паролей заставляют пользователей изменять свои предыдущие пароли предсказуемым образом или использовать слабые, легко запоминающиеся комбинации. Результаты анализа утечек показывают, что многие пользователи (около 40 %) незначительно меняют свои пароли, а большинство сложных паролей (около 60 %) имеют структуру: заглавная буква + слово + цифра или символ в конце. Для оценки последствий сравним два подхода.

Вариант 1 (классический, с регулярной сменой): Пользователь вынужден менять пароль каждые 90–365 дней. Типичный размер составляет 10–12 символов, с обязательным использованием четырех классов символов. При скорости перебора в автономном режиме  $10^{10}$  попыток/с полная атака методом перебора занимает примерно 35–40 дней. Многочисленные смены увеличивают вероятность использования ненадежных вариантов, тем самым снижая показатель надежности на 20–40 % по сравнению с теоретическим значением.

Вариант 2 (основан на рисках, без стандартной смены): Смена требуется только при компрометации, смене роли или аномальном поведении входа. Рекомендуемая длина  $\geq 14$ –15 символов (кодовая фраза из 4–5 слов или 15 случайных символов). Время подбора  $\approx 3$ –300 лет при  $10^{10}$  попыток/с. Это делает ваш пароль чрезвычайно надежным.

### Литература

1 NIST Special Publication 800-63B. Digital Identity Guidelines: Authentication and Authenticator Management [site]. – 2025. – URL: <https://pages.nist.gov/800-63-4/sp800-63b.html>. (дата обращения: 06.02.2026).

2 The 2025 Hive Systems Password Table. Hive Systems [site] – 2025. – URL: <https://www.hivesystems.com/blog/are-your-passwords-in-the-green>. (дата обращения: 06.02.2026).

**А. В. Мезенцев, Т. В. Процкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ИНТЕРНЕТ-МАГАЗИНА КАНЦЕЛЯРСКИХ ТОВАРОВ С ИСПОЛЬЗОВАНИЕМ ФРЕЙМВОРКА ASP.NET CORE MVC**

Разработанный веб-сервис «Интернет-магазин канцтоваров» представляет собой цифровую платформу, предназначенную для полной автоматизации процессов онлайн-продаж и управления торговой деятельностью. Для корректной работы системы создаётся структурированная база данных, содержащая подробную информацию

о товарах, иерархическую структуру категорий для удобной навигации по каталогу, учётные записи зарегистрированных пользователей и полный архив всех сделанных покупок.

Основная цель проекта – создание универсального веб-приложения, автоматизирующего ключевые этапы функционирования интернет-магазина: от демонстрации ассортимента до формирования аналитических статистик. Платформа обеспечивает простое и удобное взаимодействие покупателей с каталогом благодаря возможностям поиска и фильтрации, позволяет динамично формировать корзину и оформлять заказы. Для администраторов предусмотрена панель управления, которая позволяет контролировать основные операции: обновление ассортимента, отслеживание остатков на складе и обработку заказов. Покупатели могут легко просмотреть детализированную информацию о товаре через всплывающие карточки, а сотрудники имеют доступ к эффективным инструментам для оперативного редактирования данных посредством интерактивных веб-форм.

Техническая реализация основана на современном технологическом стеке. Серверная часть написана на языке C# с использованием фреймворка ASP.NET Core MVC, что обеспечивает модульность и безопасную обработку данных. Клиентская часть построена с применением HTML5, CSS3, адаптивного фреймворка Bootstrap 5 и JavaScript для создания интерактивных элементов интерфейса. В качестве системы управления базами данных используется Microsoft SQL Server, взаимодействие с которым осуществляется через ORM Entity Framework Core.

Внедрение данного решения позволяет повысить эффективность работы интернет-магазина за счёт автоматизации операций.

**И. Г. Мельник, Л. К. Титова**  
(ГГТУ имени П. О. Сухого, Гомель)

## **ОСОБЕННОСТИ АВТОМАТИЗАЦИИ ИНСПЕКЦИИ КОДА СРЕДСТВАМИ НЕЙРОСЕТЕЙ**

В силу гибкости, глубокого понимания контекста и поддержки новейших технологических наборов генеративные нейросети весьма

Современные информационные технологии  
Системное и программное обеспечение информационных технологий

полезны для автоматической проверки кода. Ориентированный на контекст подход дает возможность проводить быстрый анализ для вставленных фрагментов программного кода.

Вместе с тем, система пригодна чтобы обрабатывать целые подключенные хранилища систем контроля версий.

Обширная база знаний внутри системы предоставляет доступ к лучшим мировым практикам программирования. Интеллектуальная система поддерживает много языков и архитектурных шаблонов, поэтому это делает ее гибким инструментом для разработчиков разного уровня подготовки, от новичков до опытных специалистов.

Глубокий анализ смысла является главной особенностью при проверке текста программ. Система предлагает также встроенные алгоритмы в дополнение к используемым статическим анализаторам. Вдобавок к синтаксическим ошибкам, они находят скрытые «симптомы плохого кода» в виде плохих или старых архитектурных решений.

Тем не менее, для выявления подобных недочетов требуется принятие во внимание относительно специфики конкретного проекта. В целях упрощения начала работы, сложные технические термины переводятся, причем нейросеть описывает выявленные проблемы ясным да к тому же доступным языком. Ради экономии времени руководителей команд это важно для обучения начинающих разработчиков.

Интеллектуальная система генерирует автоматически улучшенный, более читаемый, а также оптимизированный вариант, предлагая структурную переработку программного текста, адаптированную под стандарты конкретной команды.

Современные приложения подвергаются риску кибератак на постоянной основе, следовательно, крайне важна тщательная проверка кода. Это необходимо вплоть до попадания кода в промышленную эксплуатацию.

Интеллектуальная система предоставляет инструменты для глубокого анализа написанного текста с целью выявления потенциальных уязвимостей безопасности.

**И. Г. Мельник, Л. К. Титова**  
(ГГТУ имени П. О. Сухого, Гомель)

## **РАЗРАБОТКА ИНТЕРАКТИВНОГО МОДУЛЯ РЕЦЕНЗИРОВАНИЯ ПРОГРАММНОГО ТЕКСТА НА БАЗЕ ГЕНЕРАТИВНЫХ МОДЕЛЕЙ**

Контроль качества изменений является важным, но трудоемким этапом создания программного обеспечения. В командной работе выявление изъянов традиционно требует ручного чтения текста коллегами, что сильно увеличивает затраты времени. Предложенное решение на базе генеративных моделей автоматизирует этот процесс, так как проверка логических связей, поиск упущений и оценка производительности выполняются через специализированный интерактивный модуль. Это сводит к минимуму человеческий фактор и ускоряет согласование правок путем визуального отображения предложенных улучшений прямо в среде разработки.

Основной функционал инструмента включает построение карты зависимостей. Она позволяет оценивать влияние новых функций на всю систему и гарантирует безопасное внедрение дополнений без нарушения работы существующих механизмов. Кроме того, модуль выгружает результаты проверки в виде готовых пакетов исправлений, которые можно мгновенно применить в рабочей среде или отправить на согласование, что добавляет гибкости при совместной разработке.

Важной особенностью разработки является поддержка диалогового режима взаимодействия, в котором разработчик получает возможность задавать уточняющие вопросы касательно предложенных изменений. Система способна анализировать сложность алгоритмов и прогнозировать потенциальные проблемы с потреблением памяти еще на этапе написания. Интерактивная среда позволяет сравнивать исходный вариант с предложенным генеративной моделью решением в режиме реального времени, что помогает программисту понять суть оптимизации и повышает общую квалификацию специалиста. Предложенное решение радикально упрощает проверку программного текста. Его главное преимущество заключается в наглядной демонстрации логических ошибок с мгновенным предоставлением путей их устранения. Данный подход универсален

Современные информационные технологии  
Системное и программное обеспечение информационных технологий  
и легко адаптируется для смежных задач, таких как проектирование  
структуры баз данных или автоматическое создание сопроводитель-  
ной документации.

**Н. С. Михалев**  
(ГрГУ имени Янки Купалы, Гродно)

## **ПРИМЕНЕНИЕ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ПРОГНОЗИРОВАНИЯ СБОЕВ В ИТ-ИНФРАСТРУКТУРЕ**

Бесперебойная работа ИТ-инфраструктуры критически важна для предприятий. Традиционный мониторинг по пороговым значениям часто не выявляет скрытые аномалии до наступления сбоя. Целью работы является разработка программной модели прогнозирования отказов серверного оборудования с использованием машинного обучения.

Для обучения использовались логи системного мониторинга: загрузка процессора, использование оперативной памяти, температура компонентов и сетевые задержки. Был проведен сравнительный анализ алгоритмов: логистической регрессии, случайного леса (Random Forest) и рекуррентных сетей (LSTM). Программная реализация выполнена на языке Python с применением библиотек Scikit-learn и TensorFlow.

Результаты показали, что модель на основе случайного леса обеспечивает точность до 92 % на табличных данных. При этом архитектура LSTM оказалась наиболее эффективной для выявления временных зависимостей в потоковых данных. Разработанная система интегрирована в панель мониторинга и генерирует предупреждения об аппаратных или программных сбоях за 2–4 часа до их возникновения.

Таким образом, применение предиктивной аналитики значительно повышает отказоустойчивость систем. Внедрение решения позволяет сократить время простоя оборудования на 30 % и оптимизировать графики технического обслуживания. Практическая значимость работы заключается в возможности масштабирования предложенного алгоритма на крупные распределенные дата-центры. Дальнейшие исследования будут направлены на адаптацию модели для работы в облачных средах.

### Литература

1 Флах, П. Машинное обучение. Наука и искусство построения алгоритмов / П. Флах. – М.: ДМК Пресс, 2015. – 400 с.

2 Шолле, Ф. Глубокое обучение на Python / Ф. Шолле. – СПб.: Питер, 2018. – 400 с.

**В. П. Мишур, А. А. Атвиновский**  
(ГГУ имени Ф. Скорины, Гомель)

### ПРИМЕНЕНИЕ БИБЛИОТЕКИ REACT ДЛЯ РАЗРАБОТКИ WEB-ПРИЛОЖЕНИЯ «СТОМАТОЛОГИЧЕСКАЯ КЛИНИКА»

В работе представлен процесс разработки веб-сайта для современной стоматологической клиники с использованием технологий HTML, CSS, JavaScript и React. Сайт предназначен для информирования пациентов о предоставляемых услугах, повышения узнаваемости клиники и упрощения процесса записи на приём.

Разработанный веб-сайт выполняет несколько ключевых задач: информирует пользователей о предоставляемых услугах, акциях и специальных предложениях, способствует росту числа записей на приём и улучшает коммуникацию с пациентами. Также сайт служит важным инструментом интернет-продвижения, расширяя возможности клиники для привлечения новых клиентов и партнёров.

На сайте представлена подробная информация о клинике, истории её создания, команде специалистов, спектре оказываемых услуг и режимах работы. Яркие и качественные фотографии врачей, кабинетов и оборудования помогают пациентам лучше ориентироваться и формируют доверие к клинике. Каждая услуга сопровождается детальным описанием, что позволяет пользователям легко понимать, какие процедуры доступны и как они проводятся (рисунок 1).

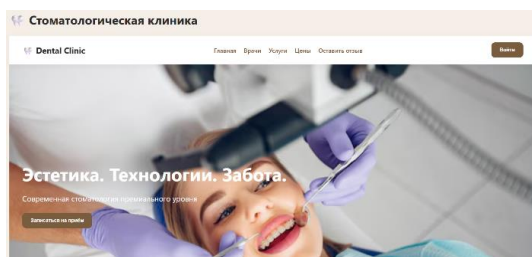


Рисунок 1 – Главная страница web-приложения

В процессе разработки использовались современные технологии веб-разработки: HTML для структурирования страниц, CSS для оформления и стилизации, JavaScript и React для интерактивных блоков и динамического отображения данных, React Router для многостраничной структуры и удобной навигации, Axios и JSON-Server для работы с данными и имитации базы данных.

**Д. А. Мошкович**  
(БГУ, Минск)

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ АВТОМАТИЗИРОВАННОГО АНАЛИЗА БАНКОВСКИХ ТРАНЗАКЦИЙ НА ОСНОВЕ ВЫПИСОК**

В условиях роста безналичных платежей контроль личных финансов становится важной задачей. Существующие решения (CoinKeeper, Monefy, ZenMoney) либо требуют передачи учётных данных от интернет-банка, либо основаны на ручном вводе операций. Использование банковских выписок в качестве источника данных является безопасной альтернативой, не требующей предоставления доступа к счёту.

Целью работы является разработка веб-приложения для автоматизированного анализа банковских транзакций на основе загружаемых выписок белорусских банков. Были исследованы и поддержаны форматы выписок трёх банков: Альфа-Банка (PDF), Приорбанка (CSV) и Белинвестбанка (CSV).

Ключевым компонентом системы является трёхуровневый алгоритм классификации транзакций по категориям. Первый уровень использует категории, предоставленные банком. Второй – MCC-коды, представляющие международную классификацию торговых точек. Третий – шаблоны названий операций.

Серверная часть реализована на языке Python с использованием фреймворка Django [1] и СУБД PostgreSQL. Визуализация финансовой аналитики включает графики динамики баланса, доходов и расходов по дням, а также кольцевую диаграмму расходов по категориям. Развёртывание выполнено в облачной инфраструктуре Yandex Cloud с применением подхода Infrastructure as Code (Terraform).

Разработанная система обеспечивает автоматизацию анализа личных финансов без необходимости передачи конфиденциальных банковских данных третьим лицам.

### **Литература**

1 Chen, S. Django Web Development Framework: Powering the Modern Web / S. Chen, S. Ahmmed, K. Lal, C. Deming // American Journal of Trade and Policy. – 2020. – Vol. 7, № 3. – P. 99–106.

**С. А. Мячина**

*(ГрГУ имени Янки Купалы, Гродно)*

## **КОЛИЧЕСТВЕННАЯ ОЦЕНКА УСТОЙЧИВОСТИ МЕТОДОВ БРАУЗЕРНОГО ФИНГЕРПРИНТИНГА К СРЕДСТВАМ ЗАЩИТЫ ПРИВАТНОСТИ**

Современные сайты всё чаще используют фингерпринтинг для скрытого отслеживания пользователей. В отличие от cookie, такие методы работают даже в режиме инкогнито и после очистки истории.

Целью работы – оценить, насколько стандартные средства защиты влияют на основные методы фингерпринтинга.

Эксперимент проводился в Google Chrome и Mozilla Firefox в трёх режимах: стандартном, приватном, а также с расширениями uBlock Origin и Privacy Badger. Для анализа использовались сервисы AmlUnique, BrowserLeaks и разработанный HTML-скрипт для генерации Canvas-хэшей.

Результаты показали, что в Chrome все исследуемые параметры (Canvas, WebGL, шрифты, разрешение экрана, User-Agent) оставались неизменёнными во всех режимах. Единственное отличие – состояние AudioContext в режимах с расширениями переходило в suspended, что не влияло на остальные характеристики. Firefox также продемонстрировал стабильность основных параметров, за исключением количества доступных шрифтов в приватном режиме (снижение со 188 до 76). Canvas-хэши в обоих браузерах оставались идентичными внутри каждой среды, но различались между Chrome и Firefox. расширения uBlock Origin и Privacy Badger блокировали рекламные трекеры, однако не изменяли значения Canvas- и WebGL-фингерпринтов.

Таким образом, стандартные средства защиты оказывают минимальное влияние на формирование браузерного отпечатка. Наиболее устойчивым оказался Canvas-фингерпринтинг. Полученные результаты могут быть использованы при разработке специализированных антифингерпринт-решений и в образовательных целях для оценки эффективности средств защиты приватности.

**Н. В. Нагорнюк, А. А. Атвиновский**  
(ГГУ имени Ф. Скорины, Гомель)

## **ПРИМЕНЕНИЕ ВЕБ-ТЕХНОЛОГИЙ ДЛЯ СОЗДАНИЯ ИНТЕРАКТИВНЫХ ТЕСТИРУЮЩИХ СИСТЕМ ПО МАТЕМАТИЧЕСКОМУ АНАЛИЗУ**

В контексте модернизации высшего образования наблюдается устойчивый тренд к интеграции информационно-коммуникационных технологий в учебный процесс, что обусловлено необходимостью повышения эффективности дидактических методов. Актуальность данного направления подтверждается потребностью в разработке инструментов, способных обеспечить не только трансляцию теоретического материала, но и объективную, оперативную диагностику уровня сформированности профессиональных компетенций обучающихся.

Авторами предлагается веб-ориентированное программное решение для оценки знаний по математическому анализу (раздел «Производная функций»), реализованное на базе языков HTML, CSS и JavaScript. Приложение функционирует как автономная гипертекстовая среда, предоставляющая пользователю возможность последовательного выполнения тестовых заданий с фиксацией промежуточных результатов.

Фундаментальным преимуществом разработанного приложения является реализация принципа немедленной обратной связи. По завершении процедуры тестирования генерируется детализированный отчет, содержащий не только количественные показатели успешности, но и качественную составляющую – референсные значения правильных решений для каждого задания.

Логика обработки пользовательских данных базируется на алгоритме последовательной валидации. В структуру скрипта интегрирован эталонный массив (объект `correctAnswers`), содержащий нормативные ответы по всем пунктам теста. При активации триггера верификации инициируется циклическая процедура перебора вопросов. Результаты сверки по каждой позиции визуализируются в интерфейсе, индицируя соответствие или несоответствие выбора. По завершении итерационного цикла вычисляется итоговый показатель успешности, после чего предусмотрена автоматическая реинициализация формы для обеспечения возможности повторного тестирования.

**Н. В. Нагорнюк, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **СТРАТЕГИЯ МИГРАЦИИ ВЕБ-ПРИЛОЖЕНИЯ «БИБЛИОТЕКА» ОТ МОНОЛИТНОЙ К МИКРОСЕРВИСНОЙ АРХИТЕКТУРЕ**

В условиях цифровизации образовательной среды традиционные библиотечные системы сталкиваются с необходимостью быстрой адаптации к меняющимся требованиям, независимого масштабирования компонентов и интеграции разнородных клиентских приложений. Монолитная архитектура, обеспечивая надежность базового функционала, демонстрирует ограничения при необходимости внедрения инновационных сервисов.

Целью работы является исследование, проектирование и практическая реализация стратегии перехода веб-приложения «Библиотека» от монолитной к микросервисной архитектуре с сохранением существующей функциональности. В процессе разработки были использованы: JavaScript, ASP.NET Core, Entity Framework для работы с данными и Swagger для документирования API.

Ключевым практическим результатом стала реализация сервиса QR-кодов как автономного компонента системы, интегрированного как с веб-интерфейсом, так и с существующим WPF-агентом библиотекаря QRScannerAgent. Сервис генерирует структурированные JSON-данные, встраиваемые в QR-код, которые содержат номер брони, ISBN, название книги и срок действия. При сканировании агентом выполняется локальное декодирование с последующей

верификацией через API, что обеспечивает полный цикл: от бронирования книги пользователем до её выдачи библиотекарем. Разработанная архитектура межсервисного взаимодействия основана на комбинации синхронных REST API и асинхронной событийной модели. Для агрегации данных и маршрутизации запросов применяется паттерн API Gateway, что позволило адаптировать существующий веб-интерфейс на Razor Pages для работы с распределенной архитектурой без существенных изменений клиентского кода. Стратегия разделения данных предполагает переход от единой базы данных EFDbContext к специализированным хранилищам для каждого сервиса с применением паттерна Saga для обеспечения согласованности данных в распределенных транзакциях. Разработанные решения могут быть применены для модернизации аналогичных библиотечных и образовательных систем.

**М. С. Немкевич**

*(ГрГУ имени Янки Купалы, Гродно)*

## **СИСТЕМА ОТСЛЕЖИВАНИЯ ЦЕНОВОЙ ДИНАМИКИ ТОВАРОВ НА МНОГОПОЛЬЗОВАТЕЛЬСКИХ ТОРГОВЫХ ОНЛАЙН-ПЛАТФОРМАХ**

В условиях цифровизации экономики и стремительного роста электронной коммерции наблюдается значительный ценовой разброс на идентичные товары, представленные на различных торговых онлайн-платформах. Это усложняет процесс выбора оптимального предложения как для конечных потребителей, так и для предпринимателей, осуществляющих мониторинг конкурентной среды. В связи с этим возрастает потребность в автоматизированных инструментах анализа и визуализации ценовой динамики.

Целью работы является проектирование и реализация мобильного приложения для отслеживания динамики цен товаров на многопользовательских торговых онлайн-платформах.

В ходе исследования проведён анализ предметной области и существующих мобильных решений в сфере мониторинга цен. Проектирование клиентской части приложения выполнено на основе принципов «Чистой архитектуры» и архитектурного паттерна, что

Материалы XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 23–25 марта 2026 г.

---

обеспечивает чёткое разделение ответственности между слоями представления, бизнес-логики и доступа к данным, а также повышает масштабируемость, тестируемость и сопровождаемость программного продукта.

Особенностью разработки является использование подхода «пустых данных», что позволило создать полнофункциональный клиентский прототип независимо от готовности серверной части и заранее определить контракты взаимодействия с серверной частью.

Практическая значимость работы заключается в разработке работоспособного мобильного прототипа системы мониторинга цен, реализующего полный цикл пользовательского взаимодействия – от управления списком товаров до визуализации ценовой динамики.

### **Литература**

- 1 Android Developers: [site]. – URL: <https://developer.android.com/>.
- 2 Kotlin Documentation: [site]. – URL: <https://kotlinlang.org/docs/home.html>.

**Д. С. Нечай, Е. Ю. Кузьменкова**  
(ГТУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ ПРИЛОЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ ASP.NET CORE**

В современном мире все большую популярность приобретает использование искусственного интеллекта и чат-ботов, способных вести диалог с пользователями. Разработка веб-приложения с возможностью создания уникальных AI-чатов с индивидуальными характеристиками и памятью контекста диалога является актуальной и вместе с тем интересной задачей.

Разработанное веб-приложение предоставляет пользователям возможность регистрации, создания и управления чатами с различными AI-персонажами. Каждый персонаж обладает уникальным описанием, системным промптом и аватаром, что позволяет создавать разнообразные сценарии взаимодействия.

Данное приложение поддерживает создание множественных чатов, сохранение истории сообщений и потоковую передачу ответов от AI в режиме реального времени.

Для разработки приложения был выбран фреймворк ASP.NET Core, обеспечивающий высокую производительность и кроссплатформенность. Приложение построено на основе RESTful API с использованием минимального API подхода. В качестве базы данных используется SQLite с применением Entity Framework Core для работы с данными через Code First подход.

Серверная часть написана на C# с использованием асинхронного программирования для обработки HTTP-запросов. Для интеграции с AI используются внешние API сервисов Mistral и Zrok с поддержкой потоковой передачи данных.

Основной функционал приложения включает регистрацию и аутентификацию пользователей по email и паролю, просмотр доступных AI-персонажей и создание новых, управление чатами с возможностью создания новых диалогов или продолжения существующих, отправку сообщений с получением ответов от AI в режиме потоковой передачи, сохранение истории диалогов с ограничением контекста последними 20 сообщениями для оптимизации производительности. Все взаимодействие с клиентской частью происходит через JSON-формат обмена данными, что обеспечивает легкую интеграцию с любыми frontend-решениями.

**П. Е. Новоженцев**

*(ГГУ имени Ф. Скорины, Гомель)*

## **СОВРЕМЕННЫЕ ИНСТРУМЕНТЫ ВИЗУАЛИЗАЦИИ ДАННЫХ**

Визуализация данных – это процесс преобразования числовой информации в графическую форму (диаграммы, графики, карты), что облегчает анализ и интерпретацию данных. Она помогает выявлять тенденции, закономерности и аномалии, а также ускоряет принятие управленческих решений.

Качественная визуализация основывается на принципах графической целостности и минимизации лишних элементов. Эдвард Тафти предложил концепцию «data-ink ratio», согласно которой график должен содержать максимум информации и минимум декоративных деталей (Tufte E. The Visual Display of Quantitative Information. 1983).

Современные инструменты визуализации можно разделить на три основные категории.

BI-платформы (Microsoft Power BI, Tableau) обеспечивают полный цикл работы с данными: подключение к источникам, трансформацию, моделирование и создание интерактивных дашбордов. Они широко применяются в корпоративной аналитике.

Облачные решения (Looker Studio, Яндекс DataLens) отличаются простотой использования и интеграцией с веб-сервисами. Подходят для маркетинговой и веб-аналитики.

Программные библиотеки (Matplotlib, Seaborn, Plotly, D3.js) используются в научной и инженерной среде. Они предоставляют высокую гибкость и возможность интеграции с аналитическими моделями, но требуют навыков программирования.

Таким образом, выбор инструмента зависит от целей анализа, аудитории и технических требований. Визуализация данных является важным элементом современной аналитики и инструментом поддержки принятия решений.

### **Литература**

1 Андерсон, К. Аналитическая культура. От сбора данных до бизнес-результатов / К. Андерсон; пер. с англ. Ю. Константиновой; [науч. ред. Р. Салахиев]. – М.: Манн, Иванов и Фербер, 2017. – 366 с.

**Д. В. Параев, А. В. Лубочкин**  
(ГГУ имени Ф. Скорины, Гомель)

### **РАЗРАБОТКА 2D-ИГРЫ В ЖАНРЕ RPG С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА C# И СРЕДЫ РАЗРАБОТКИ UNITY**

Жанр ролевых игр (RPG) на протяжении многих лет остаётся востребованным направлением в индустрии компьютерных игр. Особый интерес представляют двухмерные RPG-проекты, которые сочетают сравнительно невысокие требования к ресурсам с широкими возможностями реализации игровых механик, сюжета и системы развития персонажа.

Современные инструменты разработки значительно упрощают процесс создания игр и позволяют сосредоточиться на расширении их функциональных возможностей. Одной из наиболее популярных сред разработки является игровой движок Unity, который активно применяется для создания 2D-игр благодаря встроенным инструментам визуального редактирования, поддержке спрайтовой графики и удобной системе компонентов [1].

В представляемой работе для реализации игровой логики и механик использовался язык программирования C#, являющийся основным языком разработки в Unity. Его объектно-ориентированная модель позволяет эффективно организовать структуру проекта, реализовать взаимодействие игровых объектов, а также расширять функционал игры за счёт добавления новых систем и модулей [2].

В рамках разработки 2D-RPG были реализованы и расширены ключевые элементы жанра, такие как система характеристик персонажа, инвентарь, взаимодействие с неигровыми персонажами, а также боевая система. Использование скриптов на языке C# позволило гибко настраивать игровые механики и обеспечило возможность их дальнейшего масштабирования.

### Литература

1 Unity Documentation [site]. – URL: <https://docs.unity3d.com>. (date of access: 10.12.2024).

2 Грегори, Дж. Программирование игр на C# с использованием Unity / Дж. Грегори. – СПб.: Питер, 2022. – 480 с.

**И. А. Парфеев, Д. Л. Карагодин**  
(БрГТУ, Брест)

### МОДИФИКАЦИЯ КОДА КЛАССИЧЕСКОГО ШАБЛОНА ПРОЕКТА WIN API СРЕДСТВАМИ C#

Объектом рассмотрения являются оконные приложения, управляемые событиями, и их проекты в части построения моделей – прототипов проектов [1] по результатам анализа требований к ним [2], представленных в терминах языка UML [3]. Цель работы – разработка подхода и программных средств (приложения) автоматизации построения шаблонов проектов приложений в соответствии с проектными решениями разработчиков.

Указанные средства созданы в среде Visual Studio 2022 на языке программирования C#. Они позволяют пользователю модифицировать классический шаблон Windows API проектов (в процедурной парадигме), добавляя элементы интерфейсов, обработчиков сопутствующих событий.

Обеспечивается:

- автоматическое нахождение, просмотр, импорт-экспорт кодов файлов из папки проекта, указанного пользователем;
- поддержка обработки базовых сообщений ввода-вывода с добавлением комментариев, окон сообщений;
- поддержка функционирования иерархического пользовательского меню с неограниченным количеством пунктов и подпунктов;
- добавление ресурса типа диалоговое окно, поддержка иерархической системы окон и др.

Результаты могут использоваться для автоматизации процессов генерации прототипов проектов в целях повышения результативности этапов анализа и проектирования программ.

### **Литература**

1 Орлов, С. А. Программная инженерия: технологии разработки программного обеспечения: учебник для вузов / С. А. Орлов. – СПб.: Питер, 2018. – 640 с.

2 Муравьев, Г. Л. О прототипировании приложений с использованием диаграмм UML / Г. Л. Муравьев, В. И. Хвещук, С. В. Мухов // Инновационные технологии обучения физико-математическим и профессионально-техническим дисциплинам: материалы 14-й междунар. научно-практ. конф., Мозырь, 29 марта 2022. – С. 266–268.

3 Буч, Г. UML / Г. Буч, А. Якобсон, Дж. Рамбо. – СПб.: Питер, 2006. – 736 с.

**А. А. Пацков, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

### **МНОГОЯЗЫЧНОСТЬ ИНФОРМАЦИОННОЙ СИСТЕМЫ «ОНЛАЙН-ТАБЛО АЭРОПОРТА»**

Web-приложения занимают важное место в повседневной жизни пользователей. Одним из востребованных направлений является

разработка ресурсов, предоставляющих оперативный доступ к актуальным данным, например, онлайн-табло аэропортов, позволяющие пассажирам отслеживать статус рейсов и планировать поездки.

С учетом того, что услугами аэропорта пользуются пассажиры из разных стран, клиентская часть приложения реализована с поддержкой нескольких языков. Пользователь может выбрать удобный язык интерфейса с помощью специального переключателя в шапке сайта. При изменении языка автоматически переводятся все элементы навигации, информационные сообщения и статусы рейсов, что делает систему доступной для международной аудитории.

Реализация многоязычности выполнена с использованием файлов языковых констант, что упрощает добавление новых языков в будущем. Текстовый контент, включая названия авиакомпаний, городов и статусы рейсов, адаптируется под выбранную локализацию. Данный подход позволяет существенно расширить аудиторию пользователей и повысить комфорт использования сервиса для иностранных граждан. В системе реализована поддержка русского и английского языков, однако архитектура приложения позволяет легко подключать дополнительные языковые пакеты без изменения основного кода. Для корректного отображения языков с различными системами письменности (кириллица, латиница) применяются соответствующие кодировки и шрифты, что гарантирует правильное воспроизведение текста независимо от выбранного языка. Переключение между языками происходит динамически без перезагрузки страницы, что обеспечивает плавность взаимодействия и не требует повторного ввода данных или поисковых запросов. В приложении также реализована возможность автоматического определения языка пользователя на основе его IP-адреса или настроек браузера, что делает взаимодействие с системой еще более удобным и интуитивно понятным.

**В. А. Переплетчикова, Е. М. Березовская**  
*(ГГУ имени Ф. Скорины, Гомель)*

## **ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА КНИЖНОГО ИНТЕРНЕТ-МАГАЗИНА “BOOKSTORE”**

В современном мире, где ценится каждая минута, важно иметь быстрый и удобный доступ к качественным и увлекательным

Материалы XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 23–25 марта 2026 г.

книгам. Именно поэтому сегодня книжный магазин должен быть не только на улице, но и в сети, чтобы у каждого была возможность найти и купить нужную книгу, не выходя из дома.

Цель исследования – разработка функционального прототипа книжного интернет-магазина с использованием фундаментальных веб-технологий. В ходе работы создан веб-интерфейс интернет-магазина “BookStore” с использованием современного технологического стека HTML5, CSS3 и JavaScript (ES6+) (рисунок 1).

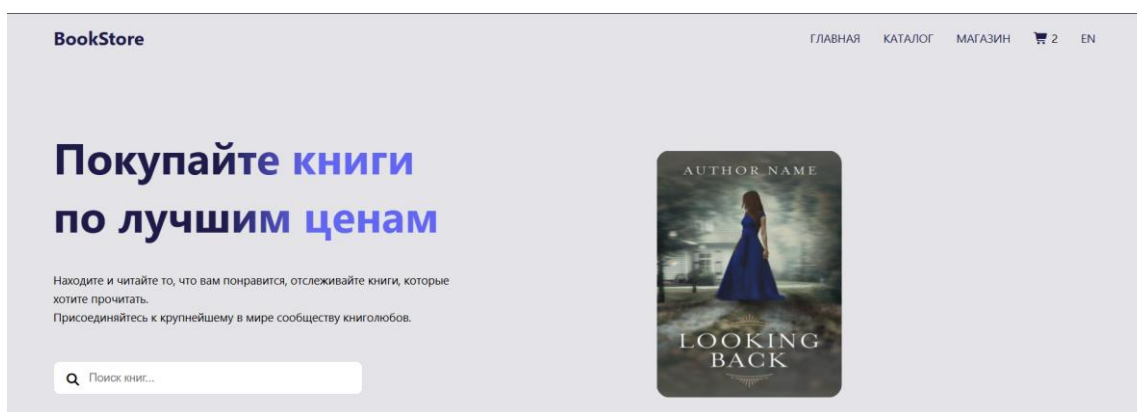


Рисунок 1 – Баннер главной страницы интернет-магазина

Были реализованы динамические каталоги товаров с различными разделами. Например, раздел «Бестселлер» – один из ключевых категорий на главной странице книжного интернет-магазина, выполняет функцию витрины наиболее популярных книг, стимулируя пользовательский интерес и повышая конверсию в покупки. Также была разработана и реализована логика поиска, «живая» корзина покупок с возможностью управлять их количеством и увидеть итоговую стоимость заказа перед подтверждением, пошаговая система оформления заказа с возможностью выбора даты и времени доставки. Для имитации работы с сервером было организовано взаимодействие через REST API с помощью MockAPI, что позволило отработать асинхронные операции получения данных и отправки заказов. Пользовательский интерфейс приложения “BookStore” является интуитивно понятным, современным и адаптивным. Разработка функционирует корректно и доступна для комфортного использования пользователями.

**Д. И. Пивунов**  
(ГрГУ имени Янки Купалы, Гродно)

## **РАЗРАБОТКА МОДУЛЯ МАРКЕТПЛЕЙСА ДЛЯ ОРГАНИЗАЦИИ ТОРГОВЛИ В ФОРМАТЕ АДВЕНТ-КАЛЕНДАРЯ**

Современный маркетплейс характеризуется высокой конкуренцией и необходимостью постоянного удержания внимания пользователей. Внедрение современных цифровых инструментов является необходимым условием для повышения эффективности продаж, увеличения вовлеченности пользователей и оптимизации сезонных маркетинговых кампаний. Целью данной работы является представление концепции модуля маркетплейса, ориентированного на формат адвент-календаря.

Модуль должен предоставлять следующие функциональные возможности: конструктор ежедневных товарных слотов, механизм динамического ценообразования для акционных товаров, систему персонализации предложений на основе истории покупок и просмотров, интеграция с бонусным счетом пользователя, а также аналитическая панель для мониторинга конверсии каждого отдельного «окна» календаря.

При проектировании модуля учтены следующие технические аспекты: отказоустойчивость – обеспечение стабильной работы в моменты пиковых нагрузок при обновлении ежедневных предложений; защита от фрода – предотвращение повторного получения бонусов и злоупотребления скидками; масштабируемость – архитектура, позволяющая адаптировать модуль под различные инфоповоды и длительность акций.

Разработка модуля маркетплейса для организации торговли в формате адвент-календаря является перспективным направлением в области E-commerce, позволяющим повысить экономическую эффективность торговой площадки и улучшить пользовательский опыт. Реализация такого функционала требует тщательного проектирования бизнес-логики, использования современных технологий, кэширования данных и обеспечения бесшовной интеграции с существующими системами управления заказами и складом.

**А. В. Понедькова, Г. Л. Карасёва**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ ПОИСКА АПТЕК И ЛЕКАРСТВЕННЫХ СРЕДСТВ В ГОРОДАХ РЕСПУБЛИКИ БЕЛАРУСЬ**

В современном мире люди привыкли получать информацию онлайн. Однако поиск лекарств до сих пор остается неудобным. Актуальность данного проекта обусловлена потребностью в единой платформе, которая структурирует данные о наличии лекарств и работе аптек. Такой сервис не только ускоряет поиск нужных препаратов, но и повышает доверие к аптечным сетям за счет актуальной информации: сведений о наличии товаров и реальном режиме работы.

Приложение для поиска аптек и лекарственных средств в городах Республики Беларусь – это веб-приложение, созданное для жителей всех городов. С его помощью достаточно одного запроса, чтобы узнать, где находится ближайшая аптека и есть ли в наличии нужное лекарство.

Основная цель разработки – создать веб-приложение, которое обеспечит быстрый доступ к актуальным данным обо всех аптеках и лекарствах в стране. Приложение предлагает доступ к контактным данным, местоположению и часами работы множества аптек. Особое внимание уделено возможности поиска конкретных препаратов. Для удобства реализована фильтрация результатов, а также поиск по интерактивной карте. Пользователи могут просматривать график работы, фотографии аптек, читать и оставлять отзывы о них. Так же в приложении предусмотрен личный кабинет пользователя. В нем сохраняется история комментариев и оценок, которые посетитель оставлял на страницах аптек. Приложение разработано с использованием HTML, CSS, JavaScript и актуальных фреймворков, что обеспечивает высокую скорость его работы. Оно отличается интуитивно понятным дизайном, доступным пользователям любого возраста, и продуманной системой навигации.

Разработанный мной «фармацевтический справочник» существенно упрощает процесс поиска медикаментов и аптек, объединяя данные по всей Беларуси в одном месте. Удобный интерфейс, полезный функционал и наличие личного кабинета делают его незаменимым помощником, экономящим время и силы пользователей.

**Т. А. Поть, А. М. Лисай**  
(ГрГУ имени Янки Купалы, Гродно)

## **НЕЙРОСЕТЕВЫЕ МОДЕЛИ ДЛЯ СЕГМЕНТАЦИИ МРТ-ИЗОБРАЖЕНИЙ В ЗАДАЧАХ БЫСТРОЙ ДИАГНОСТИКИ ЗАБОЛЕВАНИЙ**

Рост объема данных лучевой диагностики делает автоматизацию анализа МРТ актуальной задачей нейрорадиологии. Поскольку ручная сегментация трудоемка и вариабельна, применение глубокого обучения направлено на повышение точности и ускорение диагностики.

Цель работы – разработка и исследование нейросетевой модели для автоматической сегментации патологических областей на МРТ головного мозга. В качестве базовой архитектуры использована U-Net++ с энкодером EfficientNet-B4, что обеспечило высокую точность локализации границ благодаря многоуровневым пропускным связям и трансферному обучению [1].

Предметная область характеризуется многоканальными данными (T1, T2, FLAIR), выраженным дисбалансом классов и ограниченным объемом выборки, что потребовало использования комбинированной функции потерь, аугментации и ранней остановки для предотвращения переобучения.

В ходе экспериментов достигнут коэффициент Dice 0.8706 на валидации и 0.8629 на финальной эпохе при средней абсолютной ошибке 0.0028, что подтверждает эффективность модели. Подход обеспечивает быстрый автоматический анализ МРТ и воспроизводимую оценку патологий, снижая нагрузку на врача.

Разработанная модель может применяться как инструмент поддержки диагностических решений. В дальнейшем планируется создание пользовательского интерфейса для интеграции в клиническую практику с возможностью загрузки МРТ, визуализации сегментации и получения количественных показателей в реальном времени.

### **Литература**

1 Wiki.cloudfactory.com: UNet++ – URL: <https://wiki.cloudfactory.com/docs/mp-wiki/model-architectures/u-net-plus-plus> (дата обращения: 20.02.2026).

**В. Д. Приходько, Г. Л. Карасёва**  
(ГГУ имени Ф. Скорины, Гомель)

## **МАТЕМАТИЧЕСКАЯ МОДЕЛЬ АДАПТИВНОГО УПРАВЛЕНИЯ ВЫЧИСЛИТЕЛЬНЫМИ РЕСУРСАМИ**

Распределённые информационные системы функционируют при стохастической нагрузке, что делает статические механизмы масштабирования неэффективными. Предлагается формализованная модель адаптивного управления ресурсами, ориентированная на минимизацию задержек и эксплуатационных затрат в составе системного программного обеспечения облачной инфраструктуры.

Пусть входной поток заявок характеризуется интенсивностью  $\lambda(t)$ , а система содержит  $n(t)$  однородных узлов производительности  $\mu$ . При допущении экспоненциального обслуживания используется модель M/M/n(t). Коэффициент загрузки определяется как  $\rho(t) = \lambda(t) / (n(t)\mu)$ . Среднее время ожидания  $Wq(t)$  зависит от вероятности ожидания и разности  $n(t)\mu - \lambda(t)$ .

Задача управления формулируется как минимизация функционала затрат

$$J = \int_0^T (c_1 n(t) + c_2 Wq(t)) dt$$

при ограничении  $\rho(t) < 1$ , где  $c_1$  – стоимость ресурса,  $c_2$  – штраф за задержку.

Для прогнозирования  $\lambda(t)$  применяется модель временных рядов  $\lambda(t+1) = f\theta(xt)$ , параметры  $\theta$  определяются из условия минимизации квадратичной ошибки. Реализация выполнена с использованием TensorFlow. Оптимальное значение  $n(t)$  вычисляется методом стохастического градиентного спуска с адаптивным шагом.

Подсистема реализована как микросервис, взаимодействующий с оркестратором контейнеров Kubernetes. При превышении допустимой ошибки прогноза активируется резервная пороговая стратегия масштабирования.

Экспериментальная апробация показала снижение среднего времени отклика на 16–18 % и уменьшение интегральных затрат на ресурсы до 10 % по сравнению с реактивным управлением. Полученные результаты подтверждают эффективность интеграции методов теории массового обслуживания и численной оптимизации в системное и программное обеспечение распределённых ИТ-систем.

**Т. В. Пыршина, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА И КЛИЕНТСКОЙ ЛОГИКИ ИНТЕРНЕТ-МАГАЗИНА «ЦВЕТОЧНЫЙ МАГАЗИН»**

Реализована клиентская часть web-приложения «Цветочный магазин», обеспечивающая взаимодействие пользователя с серверной логикой и базой данных. При разработке использованы HTML5, CSS3, JavaScript и фреймворк Bootstrap, который позволил стандартизировать элементы интерфейса, ускорить процесс верстки и обеспечить адаптивность системы на различных устройствах.

В рамках клиентской реализации выделены следующие функциональные модули: «Каталог», «Карточка товара», «Корзина», «Оформление заказа», «Личный кабинет».

«Каталог» реализует динамическое отображение товаров с возможностью фильтрации по категориям и диапазону цен. Фильтрация осуществляется с использованием JavaScript и асинхронных запросов к серверу (AJAX), что исключает необходимость полной перезагрузки страницы.

«Карточка товара» обеспечивает отображение детальной информации о товаре и передачу данных в модуль корзины. Реализована обработка событий добавления товара с передачей параметров (id товара, количество) на серверную часть.

«Корзина» функционирует в интерактивном режиме. При изменении количества товаров автоматически пересчитывается итоговая стоимость заказа. Для обновления данных используется асинхронный обмен с сервером.

«Оформление заказа» включает валидацию пользовательских данных на стороне клиента. Проверка корректности ввода осуществляется с использованием встроенных инструментов Bootstrap Validation и JavaScript.

В приложении реализованы: асинхронная передача данных с использованием fetch / AJAX; обработка пользовательских событий (click, change, submit); динамическое обновление DOM-элементов; модальные окна Bootstrap; всплывающие уведомления о результатах операций; индикаторы загрузки при отправке запросов. Для повышения устойчивости интерфейса предусмотрена обработка ошибок при получении некорректных ответов от сервера.

**В. А. Радюк, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА КЛИЕНТСКОЙ ЧАСТИ ИНТЕРНЕТ-МАГАЗИНА “AUTOMARKET”**

Спроектировано и реализовано web-приложение “AutoMarket” – интернет-магазин по продаже автомобилей. Для реализации клиентского интерфейса использовался HTML5, CSS3 и фреймворк Bootstrap. Основной целью разработки интерфейса являлось создание системы, обеспечивающей удобное взаимодействие пользователя с каталогом товаров, корзиной и оформлением заказов. В приложении реализованы:

1 *Каталог товаров*, который представляет собой сетку карточек (grid), где каждый элемент содержит изображение автомобиля, его основные характеристики (двигатель, привод, пробег) и цену. В интерфейс интегрированы интерактивные элементы: кнопки «Добавить в избранное» и «В корзину», состояние которых меняется в зависимости от действий пользователя (например, изменение цвета иконки при добавлении в избранное).

2 *Навигация и обратная связь*. В шапке сайта размещен динамический счетчик корзины, который обновляется при каждом добавлении товара, информируя пользователя о текущем состоянии покупок.

3 *Личный кабинет и история заказов*. Страница профиля пользователя агрегирует персональные данные и историю активности.

Реализован вывод списка всех совершенных заказов. При клике на конкретный заказ открывается модальное окно, в котором генерируется детализированный чек с перечнем товаров, итоговой суммой и датой покупки.

4 *Формы и валидация.* Все формы ввода оснащены двухуровневой валидацией, как на стороне клиента, так и на стороне сервера, предотвращая ввод некорректной информации или попытки обхода клиентских ограничений.

Благодаря использованию современных инструментов верстки и интерактивных компонентов Bootstrap, удалось создать интуитивно понятный и отзывчивый интерфейс. Применение модальных окон и динамических счетчиков значительно повышает удобство взаимодействия пользователя с магазином (UX), обеспечивая при этом корректное отображение на любых типах устройств.

**И. А. Рогальский**  
(БрГУ имени А. С. Пушкина, Брест)

## **МНОГОУРОВНЕВАЯ ЗАЩИТА СЕССИЙ И JWT-ТОКЕНОВ С УЧЁТОМ РОЛЕЙ**

Функциональное разделение токенов на пару Access Token и Refresh Token является базовым принципом построения защищенной системы аутентификации. Access Token, обладающий коротким сроком действия, предъявляется при каждом запросе к защищенным ресурсам, что минимизирует риски его использования злоумышленником в случае перехвата. Refresh Token служит исключительно для получения новой пары токенов и позволяет централизованно управлять жизненным циклом пользовательской сессии на серверной стороне [1].

Привязка токена к контексту устройства обеспечивает защиту от использования перехваченных токенов на иных устройствах. Включение в полезную нагрузку хешированных характеристик окружения клиента – параметров браузера, сетевых идентификаторов и криптостойкой случайной последовательности – позволяет серверу при верификации сопоставлять предъявленный контекст с эталонным, что делает невозможным применение скомпрометированного токена вне легитимного окружения.

Встраивание ролей и прав доступа непосредственно в тело токена соответствует парадигме самодостаточности и позволяет децентрализовать процесс авторизации. Включение в состав Access Token компактного представления ролевой модели и конкретных разрешений дает возможность промежуточным слоям программного обеспечения принимать решения о предоставлении доступа в реальном времени без обращений к централизованным сервисам управления идентификацией, что существенно повышает производительность и отказоустойчивость системы [2].

### Литература

1 Обеспечение безопасности токенов аутентификации: [сайт]. – URL: <https://tproger.ru/translations/user-session-security> (дата обращения: 13.02.2026).

2 RBAC на практике: [сайт]. – URL: <https://blog.logto.io/ru/rbac-in-practice> (дата обращения: 13.02.2026).

**А. А. Романовский**

*(ГрГУ имени Янки Купалы, Гродно)*

### **ОБНАРУЖЕНИЕ АНОМАЛЬНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ДЕЙСТВИЙ В ЛОГАХ ОБЛАЧНЫХ СРЕД (AWS CLOUDTRAIL) С ПОМОЩЬЮ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ**

С массовой миграцией корпоративных инфраструктур в облачные среды (AWS) критическую важность приобретает аудит действий пользователей. Основным источником данных для аудита выступает сервис AWS CloudTrail, регистрирующий вызовы API. Однако традиционные средства защиты (SIEM, AWS GuardDuty) демонстрируют низкую эффективность против сложных целевых атак (APT) и инсайдеров, чьи действия по отдельности выглядят легитимно, но в совокупности представляют угрозу.

Целью работы является повышение защищенности облачной инфраструктуры за счет внедрения методов глубокого обучения для анализа последовательностей событий. В качестве базового алгоритма выбрана рекуррентная нейронная сеть архитектуры LSTM (Long Short-Term Memory), способная решать проблему затухающего градиента и учитывать долгосрочный контекст действий пользователя.

Предложенный подход включает предобработку логов в формате JSON: парсинг полей, токенизацию событий и формирование векторов фиксированной длины методом скользящего окна. Обучение модели производится в режиме Self-Supervised Learning, где сеть предсказывает вероятность следующего действия в цепочке. Если реальное событие имеет вероятность ниже установленного порога, оно помечается как аномалия.

Разработанная архитектура системы, реализованная в виде Docker-контейнера с использованием библиотек TensorFlow/Keras, позволяет интегрировать модуль детекции в существующие процессы DevSecOps и выявлять инциденты в режиме, близком к реальному времени.

### **Литература**

1 Du, M. DeepLog: Anomaly Detection and Diagnosis from System Logs through Deep Learning / M. Du, F. Li // CCS '17. – 2017. – P. 1285-1298.

**К. С. Руть**

*(ГрГУ имени Янки Купалы, Гродно)*

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ АВТОМАТИЗАЦИИ ДОКУМЕНТООБОРОТА В ОРГАНИЗАЦИИ**

Эффективное управление компанией в сфере грузоперевозок требует непрерывного контроля над ключевыми операционными и финансовыми показателями. Традиционные методы учёта, основанные на бумажных носителях и разрозненных электронных таблицах, зачастую приводят к ошибкам, потере данных и отсутствию целостной картины по бизнесу. Это напрямую влияет на прибыльность, затрудняет расчёт налогов, планирование расходов и корректную оплату труда сотрудников.

В работе рассматривается подход к разработке веб-приложения, предназначенного для комплексной автоматизации учёта и контроля всех основных аспектов деятельности транспортной компании. Ядром системы является единая база данных, которая интегрирует управление автопарком, документооборот и финансовые

операции. Приложение позволит вести детальный учёт по каждому транспортному средству: фиксировать пробег, расход топлива, затраты на обслуживание и ремонт, а также закреплять автомобили за водителями с учётом их рабочего времени и рейсов.

Особое внимание уделяется автоматизации формирования первичной документации (таких как путевые листы, акты выполненных работ, товарно-транспортные накладные) на основе данных о выполненных перевозках. Это обеспечит юридическую корректность и своевременность оформления. Все поступающие данные автоматически агрегируются в аналитические отчёты, дающие владельцу бизнеса полный контроль над финансовыми потоками: себестоимость рейсов, расчёт заработной платы водителей, учёт доходов и расходов, формирование данных для налогового учёта.

Таким образом, разрабатываемое веб-приложение представляет собой инструмент оперативного и стратегического управления. Оно не только заменяет рутинный бумажный документооборот, но и предоставляет владельцу бизнеса актуальную аналитику в реальном времени. Это позволяет принимать обоснованные управленческие решения, оптимизировать затраты, минимизировать финансовые и налоговые риски, что в конечном итоге повышает прозрачность, управляемость и рентабельность бизнеса в сфере грузоперевозок.

**И. И. Савинский**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ИНТЕЛЛЕКТУАЛЬНЫЕ МЕТОДЫ ПЕРСОНАЛИЗАЦИИ КОНТЕНТА В СИСТЕМАХ ОНЛАЙН-КИНОТЕАТРОВ**

В условиях активного развития цифровых медиаплатформ онлайн-кинотеатры становятся одним из ключевых способов потребления аудиовизуального контента. В связи с этим особую актуальность приобретает использование методов искусственного интеллекта для анализа поведения пользователей и формирования рекомендаций киноконтента.

Основной задачей работы является создание программной архитектуры, способной анализировать историю просмотров, учитывать индивидуальные интересы зрителя и адаптировать предлагаемый контент в режиме реального времени. Предлагаемая система

строится на многоуровневой архитектуре, включающей клиентский, серверный и аналитический уровни. Клиентская часть обеспечивает интерактивное взаимодействие с пользователем и адаптацию интерфейса под различные устройства. Серверный уровень отвечает за обработку запросов, управление пользовательскими данными и доступ к медиаресурсам. Аналитический уровень реализует интеллектуальную обработку информации и включает модуль рекомендаций, основанный на методах машинного обучения.

В рамках рекомендательной подсистемы используются гибридные алгоритмы, сочетающие коллаборативную фильтрацию и контентно-ориентированный анализ. Коллаборативный подход позволяет выявлять схожесть между пользователями на основе их действий в системе, тогда как контентный анализ учитывает характеристики фильмов, такие как жанр, тематика и ключевые элементы описания. Дополнительно применяются методы обработки естественного языка для анализа пользовательских отзывов и аннотаций, что расширяет возможности персонализации.

Результатом работы является онлайн-видеосервис с интеллектуальной системой персонализации, которая может быть использована при разработке коммерческих и корпоративных платформ потокового видео. Применение методов искусственного интеллекта в подобных системах способствует повышению качества пользовательского опыта и эффективности использования цифровых медиаресурсов.

**А. С. Савицкий, Ф. С. Боярин**  
(ГрГУ имени Янки Купалы, Гродно)

## **ПРИМЕНЕНИЕ ХРАНИЛИЩА ДЛЯ ОРГАНИЗАЦИИ ФАЙЛОВОГО ОБМЕНА В МНОГОФУНКЦИОНАЛЬНОМ МЕССЕНДЖЕРЕ**

Хранение файлов непосредственно в базе данных мессенджера ведёт к росту объёма СУБД, снижению производительности и усложнению масштабирования. Альтернативой является, например, использование объектного хранилища Minio, который благодаря своей нативной совместимости с S3-API, обеспечивает высокую скорость загрузки медиафайлов.

Целью работы является описание такой архитектуры, при которой сервер управляет доступом, а передача файлов происходит напрямую между клиентом и сторонним хранилищем.

Ключевой особенностью подхода является разделение ответственности между бэкендом и фронтендом. Сервер отвечает за аутентификацию и генерацию временных ссылок (presigned URL), привязанных к конкретной операции. Сами файлы через бэкенд не проходят – он лишь выдаёт клиенту ссылки для работы с Minio. Это позволяет избежать хранения бинарных данных в базе: в БД сохраняются только метаданные и ссылки на объекты.

Фронтенд, получив от сервера presigned URL, напрямую взаимодействует с хранилищем Minio, отправляя HTTP-запросы на загрузку или скачивание. Такой подход полностью исключает прохождение файлового трафика через прикладной слой сервера, что критически важно для снижения нагрузки. Пользователь получает отзывчивый интерфейс с индикацией прогресса, а бэкенд в это время занимается только бизнес-логикой.

Использование Minio решает проблему хранения файлов в БД, которая усложняет её резервное копирование и снижает производительность в целом. Объектное хранилище оптимизировано для неструктурированных данных, горизонтально масштабируется и обеспечивает отказоустойчивость. Безопасность достигается за счёт кратковременных ссылок.

Таким образом, интеграция Minio с прямой передачей файлов разгружает бэкенд, ускоряет работу с контентом и позволяет базе данных заниматься только текстовыми данными и метаданной, что критически важно для производительности мессенджера.

**Е. В. Сазоненко, В. В. Орлов**  
(ГГУ имени Ф. Скорины, Гомель)

## **СОЗДАНИЕ ИНТЕРНЕТ-МАГАЗИНА ПО ПРОДАЖЕ АВТОМОБИЛЕЙ**

С развитием и без того продвинутых технологий интернет-продажи стали неотъемлемы в современном мире и продолжают наращивать своё влияние. Сайты сегодня являются визитной карточкой и, по совместительству, каталогом товаров любой организации и бизнеса.

Создание такой платформы для продажи товаров и услуг становится необходимостью для многих компаний и предпринимателей. В этой работе, создание интернет-магазина для продажи автомобилей является актуальной задачей, требующей глубокого понимания принципов построения вёрстки, наладки серверной части интернет-ресурса, дизайна и непосредственно предметной области.

Интернет-магазин позволяет расширять аудиторию, значительно повысить объем продаж, улучшить конкурентоспособности и повысить лояльность к бренду. Запуск такого ресурса позволит эффективно управлять складом, проводить рекламные кампании, создавать и анализировать отчеты о количестве проданных автомобилей. Появляется возможность оперативно менять стратегию продаж в зависимости от состояния рынка и подстраиваться под настроение его аудитории. Целью данной работы является создания интернет-магазина по продаже автомобилей разных марок, как новых, так и с пробегом. Были выбраны технологии Microsoft Visual Studio Code, PHP и MySQL. Visual Studio Code используется для написания кода на разных языках программирования и разметки таких как: PHP – для серверной части, HTML/CSS – для разметки и стилизации страниц, MySQL – для работы с базой данных. Этот стек технологий позволит создать любой сайт для продажи товаров. Пользователи смогут просматривать и заказывать обратный звонок специалиста, ознакомиться с фото, видео, ценами, наличием автомобилей и их подробными характеристиками. Проект выделяется среди других интернет-магазинов своей простой и интуитивной структурой для быстрого выбора товара по его критериям, и премиальным визуальным оформлением. Обеспечивается удобство работы с каталогом, а также интеграцию с системами учета и отслеживания заказов на каждом этапе.

**А. В. Санец, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ ИНТЕРНЕТ-МАГАЗИНА «КОМПЬЮТЕРЫ И КОМПЛЕКТУЮЩИЕ» НА ЯЗЫКЕ PYTHON С ПОМОЩЬЮ ФРЕЙМВОРКА FASTAPI**

В условиях динамичного развития сферы электронной коммерции к программным решениям для интернет-магазинов предъявляются

повышенные требования по скорости обработки запросов и отказоустойчивости. Разработанное серверное приложение реализовано на языке Python с применением фреймворка FastAPI, работающего под управлением высокопроизводительного ASGI-сервера Uvicorn. Выбор данного стека объясняется асинхронной моделью, гарантирующей высокую эффективность при работе с множественными одновременными запросами.

Для обеспечения надежного хранения сведений о товарах, заказах и пользователях выбрана реляционная СУБД PostgreSQL. Взаимодействие с базой данных реализовано посредством использования ORM-системы SQLAlchemy, а за миграцию баз данных отвечает Alembic, что сохраняет целостность данных при изменениях структуры проекта.

Одной из главных характеристик платформы является разделение прав доступа и механизмы безопасности на базе протокола OAuth2 с использованием JWT-токенов.

Обычный пользователь имеет возможность просматривать каталог с использованием фильтрации по различным характеристикам, управлять содержимым корзины и оформлять заказы.

Администратор обладает расширенными правами, включающими управление ассортиментом через операции добавления, редактирования и удаления позиций, а также доступ к системе мониторинга заказов и управлению учетными записями.

Для повышения производительности и обработки тяжелых задач внедрен механизм фоновых задач. В качестве брокера сообщений используется Redis, обеспечивающий хранение очереди задач для фреймворка Celery.

Валидация входящих данных производится библиотекой Pydantic, что предотвращает попадание некорректной информации в базу данных и обеспечивает автоматическую генерацию документации API (Swagger UI). Надежность системы обеспечивается тестированием эндпоинтов через библиотеку Pytest, в то время как удобство развертывания достигается благодаря применению Docker. Построенная архитектура гарантирует высокую скорость работы.

**А. В. Сердюков, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ВСПОМОГАТЕЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ УПРАВЛЕНИЯ КАДРОВЫМИ ДАННЫМИ В СОСТАВЕ ВЕБ-СИСТЕМЫ «БИБЛИОТЕКА» С ИСПОЛЬЗОВАНИЕМ JAVA**

В условиях цифровизации учреждений культуры и образования эффективное управление кадровыми ресурсами становится неотъемлемой частью комплексной автоматизации библиотечных процессов. Современные библиотечные системы требуют не только учёта книжного фонда, но и инструментов для работы с персоналом: отделами, должностями, сотрудниками. Целью работы является изучение основ языка Java и применение полученных знаний для разработки вспомогательного приложения, интегрируемого с существующей веб-системой управления библиотекой. В процессе разработки были использованы: Spring Boot для создания backend-части, Spring Data JPA для взаимодействия с базой данных, а также Jakarta Persistence API (JPA) для объектно-реляционного отображения.

Архитектура разработанного приложения построена по принципам многослойной (n-tier) архитектуры с чётким разделением ответственности между компонентами. Слой контроллеров выступает точкой входа для HTTP-запросов и реализует REST API. Слой сервисов инкапсулирует бизнес-логику, включая криптографическое хеширование паролей с использованием библиотеки `BCrypt`, что соответствует современным стандартам безопасности. Слой репозитория построен на основе Spring Data JPA, что позволило минимизировать шаблонный код и обеспечить типобезопасный доступ к данным. Сущности отображаются на таблицы реляционной базы данных SQL Server. Ключевым результатом работы стала реализация полнофункционального Java-модуля, обеспечивающего управление кадровыми данными библиотеки. Модуль включает: безопасную аутентификацию и регистрацию пользователей с разделением ролей; полный цикл CRUD-операций для отделов, должностей и сотрудников; поиск сотрудников по фамилии и должностям по названию; интеграцию с Swing-клиентом для сотрудников библиотеки, предоставляющим удобный графический интерфейс. Была реализована загрузка и

отображение фотографий сотрудников, а также генерация детализированных отчётов в формате DOCX с использованием Apache POI, содержащих личные данные и фотографию сотрудника.

**Е. С. Соловей**

*(ГрГУ имени Янки Купалы, Гродно)*

## **РАЗРАБОТКА ЗАЩИЩЕННОГО КЛИЕНТСКОГО ИНТЕРФЕЙСА СИСТЕМЫ ВЕБ-ТЕСТИРОВАНИЯ С ПРИМЕНЕНИЕМ СТРОГОЙ ТИПИЗАЦИИ И ИНСТРУМЕНТОВ КОНТРОЛЯ ВЕРСИЙ GITLAB**

В эпоху цифровизации образовательного процесса системы веб-тестирования являются ключевым методом проверки знаний. Клиентский интерфейс такой системы уже давно не является всего лишь инструментом визуализации; это один из основных элементов входа злоумышленника в ту часть сервера, где хранятся нужные ему данные. Уязвимость фронтенда повышает вероятность получения правильных ответов или формирования фальсифицированных результатов тестирования. Поэтому важно минимизировать данный риск в начальной стадии разработки ПО.

Новый подход строится на концепции «безопасности по проектированию», или Security by Design. Для этого используются два инструмента синергии: строгая статическая типизация на стороне программного кода и автоматизированный контроль жизненного цикла разработки в экосистеме GitLab.

При проектировании использован комплексный подход к обеспечению целостности и безопасности интерфейса, включающий следующие мероприятия:

1 Использование TypeScript. Строгое определение типов данных позволяет избавиться от передачи некорректных параметров, которые могут привести к обходу проверки прав доступа, а также от возможных сбоев в алгоритмах оценки.

2 Механизмы защиты интерфейса: политика безопасности контента, позволяющая защититься от XSS-атак, санитизация все варианты пользовательского ввода, а также использование защищённых хранилищ для сохранения состояния сессии.

3 Дополнительно для интеграции безопасности и разработки использовались DevSecOps-инструменты из экосистемы GitLab.

В ходе работы была успешно спроектирована и реализована архитектура клиентской части системы веб-тестирования, обладающая высокой степенью отказоустойчивости. Введение строгой типизации позволило сократить количество ошибок на этапе компиляции на 30–40 % из-за неправильного обращения к данным API. CSP и санитизация обеспечивают надежную защиту от инъекций вредоносного контента.

**К. В. Сорока**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ПРИМЕНЕНИЕ ГИБРИДНЫХ АРХИТЕКТУР ПРИ ПРОЕКТИРОВАНИИ РЕКОМЕНДАТЕЛЬНЫХ СИСТЕМ ДЛЯ МАРКЕТПЛЕЙСОВ РАЗЛИЧНОГО МАСШТАБА**

Гибридные архитектуры в рекомендательных системах становятся краеугольным камнем для современных маркетплейсов, поскольку они объединяют сильные стороны различных методов рекомендаций. Сочетая коллаборативную фильтрацию, основанную на поведении пользователей, и контентную фильтрацию, ориентированную на характеристиках товаров, гибридные модели обеспечивают более точные и полные рекомендации. В исследованиях отмечается, что такие подходы позволяют повысить качество рекомендаций даже при высокой разреженности данных [1; 2].

Это особенно важно в ситуациях, когда для новых товаров или пользователей недостаточно информации для формирования корректных рекомендаций. Гибридные системы эффективно решают проблему холодного старта, используя контентные характеристики для рекомендаций новинок, что делает их более устойчивыми к изменениям ассортимента и поведения пользователей [2].

Гибридные архитектуры дают маркетплейсам гибкость и устойчивость: они повышают релевантность и разнообразие рекомендаций, позволяют быстро реагировать на изменения спроса и обеспечивают персонализированный пользовательский опыт.

Такой подход напрямую влияет на удержание клиентов и коммерческие показатели, что подтверждается аналитическими материалами о влиянии рекомендательных систем на маркетинг и клиентский опыт.

### Литература

1 Правила применения рекомендательных технологий. Алекс Онлайн: [сайт]. – URL: <https://alexonline.ru/pravila-primeneniya-rekomendatelnih-tehnologij> (дата обращения: 28.02.2026).

2 Смешивание знаний: Гибридные рекомендательные системы с пояснениями. IT Researches: [сайт]. – URL: <https://itresearches.com/ru/blending-insights-hybrid-recommender-systems-explained/> (дата обращения: 28.02.2026).

**И. А. Сороколетов, Е. Ю. Кузьменкова**  
(ГГУ имени Ф. Скорины, Гомель)

## РАЗРАБОТКА ИГРОВОГО ПРИЛОЖЕНИЯ “DEFENCE” С ИСПОЛЬЗОВАНИЕМ UNITY

Современную игровую индустрию невозможно представить без удобных и мощных инструментов разработки, которые позволяют даже небольшим командам или отдельным авторам создавать качественные и интересные игры. Одним из таких инструментов и является кроссплатформенный движок Unity, который сочетает в себе гибкость, производительность и доступность.

Разработанная 2D-игра в жанре top-down shooter под названием “Defence” в полной мере демонстрирует возможности Unity и языка C# для реализации полноценного игрового приложения.

Разработанная игра представляет собой динамичный шутер с видом сверху, в котором игрок управляет персонажем, сражающимся с разнообразными противниками. Важнейшей особенностью проекта является система различных типов врагов, каждый из которых обладает определенным уникальным поведением. Также предусмотрена механика смены оружия, несколько уровней с уникальным дизайном, а также интуитивно понятный пользовательский интерфейс.

Для реализации проекта были выбраны игровой движок Unity и язык программирования C#. Это обеспечило возможность реализации сложных игровых механик, высокую производительность, модульность кода. Графический дизайн реализован с использованием встроенных инструментов Unity для работы со спрайтами и анимациями. Игровая логика использует компонентно-ориентированную архитектуру, характерную для Unity, в которой каждый объект на сцене является контейнером для набора независимых компонентов.

При запуске игрового приложения пользователь попадает в главное меню, где может начать новую игру или выйти из приложения. После начала игры игрок получает контроль над персонажем, который может перемещаться по уровню и атаковать врагов. Игроку доступны несколько видов оружия, между которыми можно переключаться в реальном времени. При прохождении уровней игрок сталкивается с различными врагами и боссами. Пользовательский интерфейс включает главное меню, внутриигровой HUD, отображающий здоровье, выносливость и количество зелий, а также экраны смерти и завершения уровня.

**Е. А. Станулевич, Е. А. Левчук**  
(БГУ, Минск)

## **РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ПО ПОИСКУ ВАКАНСИЙ**

В современных условиях развития гиг-экономики временная занятость становится неотъемлемой частью трудового рынка. Студенты и специалисты часто ищут гибкие проекты. Однако существующие платформы зачастую ориентированы на долгосрочный найм, имеют перегруженный интерфейс и сложный процесс регистрации. Целью данной работы является проектирование и разработка кросс-платформенного мобильного приложения, оптимизированного для оперативного поиска и управления подработками.

В ходе анализа предметной области были выделены ключевые проблемы: фрагментированность предложений, низкая скорость связи между сторонами и отсутствие прозрачных механизмов оценки надежности. Разработанное решение предлагает концепцию быстрой

регистрации (за 2–3 минуты) с использованием OCR-технологий для автоматического распознавания документов и поэтапного формирования профиля пользователя.

Технологический стек включает фреймворк Flutter (язык Dart) для клиентской части, обеспечивающий стабильную работу на iOS и Android. Серверная часть реализована на базе Laravel 11 с использованием СУБД PostgreSQL и Redis для кэширования. Для обеспечения масштабируемости (до 10000+ одновременных пользователей) выбрана микросервисная архитектура, состоящая из 8 независимых сервисов, включая службы аутентификации, поиска, уведомлений и скоринга. Особое внимание уделено алгоритмам бизнес-логики. Реализована система многоуровневого скоринга, где статус пользователя (от «Новичка» до «Эксперта») автоматически рассчитывается на основе своевременности выполнения заданий и отзывов работодателей. Безопасность данных обеспечивается применением протокола TLS 1.3, использованием JWT-токенов с механизмом обновления и защитой от типовых веб-атак. Визуальная составляющая выполнена в минималистичном стиле с использованием акцентной фиолетовой цветовой схемы, адаптированной под требования мобильного удобства использования.

**С. Д. Степанчиков, Е. А. Ружицкая**  
(ГГУ имени Ф. Скорины, Гомель)

## **ФРЕЙМВОРК FASTAPI КАК ЭФФЕКТИВНЫЙ ИНСТРУМЕНТ ДЛЯ СОЗДАНИЯ REST-СЕРВИСОВ**

В условиях активного развития цифровых технологий возрастает потребность в быстрых и надежных инструментах разработки серверной части web-приложений. Одним из наиболее перспективных решений является фреймворк FastAPI, предназначенный для построения RESTful-сервисов на языке Python.

FastAPI ориентирован на создание API с высокой производительностью и минимальными затратами времени на разработку. В основе фреймворка лежит асинхронная модель обработки запросов (ASGI), что позволяет эффективно обслуживать большое количество клиентов одновременно. Благодаря использованию современных

возможностей Python, таких как аннотации типов, достигается автоматическая валидация данных и строгая проверка входных параметров. Особенностью FastAPI является тесная интеграция с библиотекой Pydantic, обеспечивающей проверку структуры данных и их преобразование к необходимым форматам. Это значительно снижает вероятность логических ошибок и упрощает сопровождение проекта. Кроме того, фреймворк автоматически формирует интерактивную документацию API через Swagger UI и ReDoc, что облегчает тестирование и взаимодействие между разработчиками и заказчиками.

Отдельного внимания заслуживает производительность FastAPI. По результатам независимых тестирований, он демонстрирует скорость обработки запросов, сопоставимую с такими технологиями, как Node.js, что делает его конкурентоспособным инструментом для создания масштабируемых сервисов.

Фреймворк предоставляет встроенные средства реализации аутентификации и авторизации, включая поддержку OAuth2 и JWT. Это позволяет создавать безопасные механизмы доступа к ресурсам системы. Для хранения и обработки данных FastAPI легко интегрируется с ORM, например, SQLAlchemy, и поддерживает работу с различными СУБД, такими как PostgreSQL.

Таким образом, FastAPI – современный инструмент разработки серверных приложений, сочетающий высокую производительность, безопасность и позволяющий создавать надежные и масштабируемые web-сервисы, соответствующие требованиям электронной среды.

**Д. Ч. Стецкий**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ GRAPHQL API: УЯЗВИМОСТИ И РЕШЕНИЯ**

Современная разработка веб-приложений характеризуется стремительным переходом от традиционной REST-архитектуры к использованию гибкого языка запросов GraphQL. Однако архитектурная специфика GraphQL, основанная на графовой модели и единой точке входа, порождает уникальный класс угроз. В отличие от классических интерфейсов, здесь возникают риски, связанные с семантической сложностью запросов и нелинейными связями между

объектами. Вследствие этого специфические векторы атак, такие как злонамеренное исчерпание ресурсов через глубокую вложенность или манипуляция логикой через фрагменты, часто не нивелируются стандартными средствами защиты периметра [1].

Одной из критических уязвимостей является возможность проведения DoS-атак через эксплуатацию алгоритмической сложности вложенных запросов. Поскольку GraphQL позволяет запрашивать связанные данные неограниченной вложенности, злоумышленник может сформировать циклический запрос, приводящий к экспоненциальному росту нагрузки на сервер и исчерпанию ресурсов базы данных (проблема N+1).

Синтезированная методология защиты GraphQL API базируется на эшелонированном подходе к анализу входящих данных. Первый уровень включает статический анализ AST-дерева для превентивного расчета стоимости и вложенности запросов, что нейтрализует DoS-атаки до начала их исполнения. Второй эшелон направлен на предотвращение логических уязвимостей (BOLA) через внедрение декларативной авторизации в слой Middleware, обеспечивающей проверку прав доступа на уровне отдельных резолверов. Заключительный этап предполагает использование персистентных запросов и «белых списков», что полностью исключает выполнение произвольных инструкций на стороне сервера.

### **Литература**

1 Порселло, Е. GraphQL. Язык запросов для современных веб-приложений / Е. Порселло. – 2019. – 240 с.

**Д. С. Сухаревич**  
(БрГТУ, Брест)

## **ПОСТРОЕНИЕ КАРКАСОВ ПРИЛОЖЕНИЙ С УЧЕТОМ ИЕРАРХИИ ОКОН ИНТЕРФЕЙСА**

Макетирование оконной программы – важный этап в ее разработке [1, 2]. Для случая автоматизации построения иерархии окон пользовательского интерфейса это позволяет быстро получить версию программы с упрощенной функциональностью для дальнейшего наращивания ее возможностей. Здесь предполагается, что разработчик

оперирует текстовым описанием схемы иерархии окон и ее наполнением элементами управления. Для этого задаются типы и атрибуты окон, их соподчиненность по управлению, способы запуска, состав элементов и т. д.

Соответственно предложен вариант проекта средств для автоматизации оснащения программ оконными интерфейсами, базирующийся на использовании типовых каркасов, библиотеки компонентов интерфейсов, правил и алгоритмов модификации кодов приложения, отображающих вносимые изменения. Проект реализован в среде Visual Studio на языке C++. Разработана иерархия классов для работы с пользовательскими спецификациями окон. Поддержаны:

- типовая функциональность для макетирования оконных приложений, работы с пользовательскими спецификациями интерфейсов;
- возможность макетирования программы в «ручном» режиме;
- автоматическое построение интерфейсных компонентов, их элементов на основе выбранных шаблонов.

Возможно дальнейшее расширение системы для обеспечения ее использования через мобильные устройства.

### **Литература**

1 Эванс, Э. Предметно-ориентированное проектирование: структуризация сложных программных систем / Э. Эванс. – М.: ООО «И.Д. Вильямс», 2016. – 448 с.

2 Карагодин, Д. Л. Интеграция графических ресурсных элементов в шаблон проектов на visual C++ / Д. Л. Карагодин // Современные проблемы математики и вычислительной техники: материалы 13 республиканской науч. конф. молодых ученых и студентов, Брест, 2023 г. / БрГТУ. – Бр., 2023. – С. 72-74.

**А. В. Тарканов**

*(ГГТУ имени П. О. Сухого, Гомель)*

## **ИНТЕЛЛЕКТУАЛЬНЫЙ ВЕБ-АССИСТЕНТ С ИСПОЛЬЗОВАНИЕМ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ ОБРАБОТКИ ЕСТЕСТВЕННОГО ЯЗЫКА**

Современные веб-приложения всё чаще интегрируют интеллектуальные решения для взаимодействия с конечными пользователями.

Чат-боты и диалоговые интерфейсы находят широкое применение в техподдержке, электронном обучении и различных информационных сервисах.

Создание интеллектуального веб-ассистента позволяет автоматизировать анализ запросов пользователей и оперативно предоставлять точные ответы благодаря методам обработки естественного языка.

Архитектура разрабатываемого приложения строится по принципу клиент-сервер. Фронтенд реализован через веб-интерфейс на библиотеке React, обеспечивающей удобное взаимодействие пользователя с системой [1].

Серверная часть отвечает за передачу запросов к нейросетевой модели обработки текста, расположенной на удалённом сервере, и получение ответов

Коммуникация между компонентами осуществляется через запросы. Обработка текстовых сообщений пользователя осуществляется с применением нейросетевых моделей обработки естественного языка, предназначенных для анализа и генерации текста. Модель принимает входной запрос, интерпретирует его смысловое содержание и формирует корректный текстовый ответ.

Основная логика обработки естественного языка реализуется на стороне модели, а веб-приложение выполняет функции передачи данных, хранения истории диалога и управления пользовательскими сессиями.

Полученный от модели результат дополнительно приводится к форме, удобной для отображения в интерфейсе. Такой подход обеспечивает устойчивую работу диалоговой системы пользователя и модели.

Система проектируется с учётом масштабируемости – допускает подключение новых модулей, интеграцию с внешними сервисами и адаптацию под специализированные области применения.

## Литература

1 React документация: [сайт]. – URL: <https://react.dev> – (дата обращения 20.02.2025).

**К. М. Терещенко**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА АРМ ДИСПЕТЧЕРА ПО ФОРМИРОВАНИЮ ПАССАЖИРСКИХ ПОЕЗДОВ**

Современная эффективность организации пассажирских перевозок на железной дороге во многом определяется степенью автоматизации операционных процедур. Одним из наиболее ответственных этапов является формирование пассажирских поездов, которое требует высокой точности, быстроты реакции и комплексного учета различных факторов – состава, маршрута, расписания движения, наличия вагонов и локомотивов, а также норм безопасности. На сегодняшний день большинство систем планирования и диспетчерского управления основано на устаревших ИТ-решениях либо полностью зависят от ручной обработки информации, что способно вызывать простои, ошибки и неэффективное расходование ресурсов.

Цель данной работы – создание автоматизированного рабочего места (АРМ) для диспетчера, отвечающего за формирование составов пассажирских поездов. Проект реализуется на языке программирования C# с использованием платформы .NET для построения пользовательского интерфейса и базы данных MS SQL Server. Ключевые функции включают хранение и обновление информации о подвижном составе и маршрутах, формирование составов поездов с возможностью выбора предложенных вариантов на основе имеющихся ресурсов и правил формирования составов поездов, а также мониторинг технического состояния вагонов.

Практическая значимость разработанного АРМ заключается в объединении фрагментированных источников данных в единой удобной среде, что повышает прозрачность процессов и ускоряет выработку управленческих решений. Система существенно сократит трудозатраты на ручную работу, снизит вероятность человеческих ошибок при планировании и гарантированно поддержит достоверность и оперативность информации.

Таким образом, внедрение автоматизированного рабочего места диспетчера по формированию пассажирских поездов позволит повысить эффективность, надёжность и прозрачность процесса формирования составов, обеспечив современный уровень управления операционными процессами.

**И. Ю. Тимощенко, Н. А. Аксёнова**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА СИСТЕМЫ МОНИТОРИНГА СЕТЕВЫХ УСТРОЙСТВ НА ЯЗЫКЕ С**

Современные компьютерные сети представляют собой сложные инфраструктуры, требующие постоянного контроля и обслуживания. Эффективное администрирование невозможно без использования специализированных средств мониторинга, позволяющих своевременно обнаруживать сбои и отклонения в работе оборудования. В связи с этим разработка систем автоматизированного сбора и анализа данных о состоянии сетевых устройств является актуальной задачей.

Разработано приложение, представляющее собой систему мониторинга сетевых устройств, реализованную на языке С. Приложение предназначено для сбора и отображения технических параметров компьютеров и серверов, находящихся в локальной сети. Программа выполняет регулярный опрос устройств, отслеживая доступность узлов, время отклика, а также собирает системные метрики.

База данных контролируемых устройств содержит перечень IP-адресов, а также набор метрик, подлежащих отслеживанию. При обнаружении недоступности устройства или превышении пороговых значений нагрузки программа фиксирует событие и выводит предупреждение.

Для удобства анализа предусмотрен веб-интерфейс отображения статистики: по окончании цикла опроса формируется отчёт с указанием доступных и недоступных узлов. Это позволяет администратору выявлять тенденции и прогнозировать возможные сбои.

Приложение разработано с использованием среды разработки Visual Studio Code и компилятора GCC. Выбор языка С обусловлен высокой производительностью, возможностью прямого взаимодействия с сетевыми интерфейсами. Сбор данных об использовании ресурсов удалённых устройств реализован через SNMP, а для хранения метрик используется файловая система либо лёгкая встраиваемая база данных SQLite. Приложение работает в операционных системах семейства Windows и Linux, что позволяет использовать его в гетерогенных сетевых средах.

**И. Н. Тонко**

*(ГГУ имени Ф. Скорины, Гомель)*

## **РАЗРАБОТКА ИГРОВОЙ СИСТЕМЫ «КТО ХОЧЕТ СТАТЬ МИЛЛИОНЕРОМ?»**

Интерактивные викторины занимают значимое место среди развлекательных и обучающих программных продуктов, сочетая элементы проверки знаний с возможностью состязания.

В докладе излагаются вопросы, связанные с программной реализацией игровой системы викторины «Кто хочет стать миллионером?» в виде desktop-приложения для операционных систем Windows 10/11. В качестве основной платформы разработки выбран игровой движок Unity. Вся игровая логика приложения реализована на языке программирования C# с использованием .NET.

Основные этапы разработки включали проектирование игровой механики и структуры интерфейса, реализацию системы управления вопросами, интеграцию искусственного интеллекта и голосового ввода.

Игровые вопросы и варианты ответов хранятся в формате JSON, что упрощает их загрузку, обновление и обработку. Данные профиля игрока, включая возраст, страну проживания, предпочитаемые темы и уровень сложности, сохраняются локально с использованием встроенного механизма PlayerPrefs. Это позволяет реализовать персонализацию игрового процесса без необходимости использования внешних серверов хранения данных.

Ключевой особенностью разработанной системы является интеграция искусственного интеллекта через сервис Le Chat от Mistral AI. Используемая языковая модель осуществляет генерацию вопросов в реальном времени с учётом параметров профиля игрока: возрастной категории, культурного контекста, выбранных тематических направлений и требуемого уровня сложности. Такой подход обеспечивает практически неограниченное разнообразие уникальных вопросов и повышает реиграбельность приложения без поддержки статической базы данных.

Голосовое управление реализовано с использованием встроенных средств Unity и платформы Windows, а именно модуля Windows.Speech. Данный механизм повышает доступность приложения и демонстрирует применение технологий распознавания речи в игровых системах.

**Д. В. Федотова, Г. Л. Карасёва**  
(ГГУ имени Ф. Скорины)

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ ПУБЛИКАЦИИ ВИДЕОКОНТЕНТА**

Современные технологии значительно изменили способы получения и потребления информации. В условиях стремительного развития цифровых медиа возникает потребность в эффективных инструментах для размещения и распространения контента. В частности, веб-приложения становятся незаменимыми платформами для публикации журналистских статей, видеоматериалов и других форматов медиаконтента, которые позволяют осуществлять обмен информацией в реальном времени с высоким уровнем взаимодействия с аудиторией.

Основной целью работы является реализация веб-приложения для размещения и просмотра видеоконтента, поэтому видеоматериалы становятся основным форматом потребления контента. Такое приложение может предоставить пользователям простые и удобные инструменты для редактирования и публикации видео. Например, такая платформа будет полезна и удобна для размещения личного портфолио начинающих видеоблогеров, режиссеров, моушн-дизайнеров. В качестве инструментов разработки используются язык программирования Java, СУБД PostgreSQL, фреймворк SpringBoot. Так же в веб-приложение будет реализована система разграничения прав, при которой каждой из ролей будет доступен свой функционал.

Функциональные возможности веб-приложения должны включать: возможность создания личного блога и добавления в него видеоматериалов; возможность добавления описания к размещенным материалам; возможность администрирования своего блога; возможность дальнейшего расширения приложения.

Разрабатываемое приложение должно иметь интуитивно понятный интерфейс, который будет доступен для пользователей любого уровня навыков владения компьютером. Важно, чтобы интерфейс сайта не вызывал затруднений в использовании.

Актуальность работы заключается в необходимости создания удобного ресурса для журналистов и читателей, который будет способствовать развитию процессов информирования общества и поддержке авторов в их творчестве.

**И. Ю. Филон**  
(БГУ, Минск)

## **РАЗРАБОТКА ИНТЕРАКТИВНОЙ ПЛАТФОРМЫ ДЛЯ МОНИТОРИНГА И АНАЛИЗА СПОРТИВНЫХ МЕРОПРИЯТИЙ**

Целью разработки интерактивной платформы по мониторингу спортивных мероприятий является не просто предоставление пользователям актуальных данных о результатах спортивных мероприятий, о командах, спортсменах, но и предоставление аналитики, интерактивное вовлечение болельщиков и создание персонализированного контента. Для достижения этой цели в приложении используется интеграция с внешним API посредством публичного ключа доступа, поддерживающего взаимодействие через протокол HTTPS для получения актуальных данных в формате JSON. Такое соединение будет использоваться для получения списка команд, соревнований, игроков, положения в турнирной таблице. Также используется We-Socket-соединение для отслеживания событий в реальном времени.

Для повышения вовлеченности пользователей в приложении разработаны игровые модули для проведения викторин, связанных с конкретной командой или спортивным мероприятием. Вопросы и ответы для викторин составляются из списка подготовленных шаблонов, параметризуемых реальными данными. Модуль аналитики реализован для оценки формы команды и игроков. Анализ данных проводится за определенный пользователем временной период.

Практическая значимость работы заключается в создании готового API и мобильного приложения для спортивных порталов, фан-сообществ, клубов. Платформа разрабатывается на базе многослойной архитектуры, в которой выделяются три слоя: слой базы данных или доступа к ней, сервисный слой, который берет на себя бизнес-логику по обработке данных, и слой отображения данных. Современный стек технологий позволяет эффективно разрабатывать проект. Для серверной части приложения выбраны фреймворки Spring Boot, Spring Security, для доступа к базе данных MySQL из приложения используется Spring Data и ORM Hibernate для маппинга таблиц БД в java-объекты, Flyway отвечает за управления миграциями, а для кэширования часто повторяющихся запросов применяется Redis. Мобильный клиент реализуется с использованием фреймворка Flutter.

**К. В. Фомин, Г. Л. Карасёва**  
(ГГУ имени Ф. Скорины, Гомель)

## **ИСПОЛЬЗОВАНИЕ СОВРЕМЕННЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ В ОБЛАСТИ МАРКЕТИНГА ДЛЯ ПРОДВИЖЕНИЯ МЕБЕЛЬНОЙ ПРОДУКЦИИ**

В условиях цифровой трансформации экономики наблюдается активное внедрение современных информационных технологий в сферу маркетинга. Рынок мебельной продукции характеризуется высокой конкуренцией, широким ассортиментом и необходимостью формирования устойчивых коммуникаций с целевой аудиторией. Развитие цифровых каналов продвижения, и технологий автоматизации маркетинга создаёт новые возможности для повышения эффективности продаж и укрепления позиций компаний на рынке. В этой связи особую актуальность приобретает применение комплексных информационных решений для продвижения мебельной продукции.

Целью данной работы является анализ и разработка подходов к использованию современных информационных технологий в маркетинговой деятельности предприятий мебельной отрасли. В рамках исследования рассматриваются инструменты цифрового продвижения, платформ автоматизации маркетинга мебельной отрасли и инструментов веб-аналитики. Для реализации цифровой маркетинговой системы продвижения мебельной продукции выбран современный технологический стек. В качестве основы клиентской части используется библиотека React, обеспечивающая создание динамичного и интерактивного пользовательского интерфейса. Визуальное оформление и адаптивность веб-приложения реализованы с применением CSS, что позволяет обеспечить корректное отображение контента на различных устройствах. Для хранения и обработки данных используется база данных, обеспечивающая централизованное управление информацией.

Разработанное веб-приложение для продвижения мебельной продукции позволяет автоматизировать процессы взаимодействия с клиентами, управлять ассортиментом, анализировать поведение пользователей и оценивать эффективность рекламных кампаний. Использование цифровых инструментов в рамках разработанного решения позволяет повысить эффективность маркетинговой деятельности предприятия, оптимизировать процессы обработки заявок и укрепить конкурентные позиции на рынке мебельной продукции.

**П. М. Храпунова, В. А. Короткевич**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ АВТОМАТИЗАЦИИ РАБОТЫ ДИСПЕТЧЕРСКОЙ ВУЗА**

Основной функцией диспетчерской вуза является предоставление сведений о занятости аудиторий деканатам факультетов как в процессе составления расписания занятий, так и на протяжении учебных семестров. Предоставление такой информации требует ведения объемной базы данных со сведениями о факультетах, учебных группах, аудиторном фонде (с указанием вместимости, наличия мультимедийного оборудования, компьютеров и т.п.), сроков занятости аудиторий в текущем семестре.

Целью работы являлась реализация специализированного desktop-приложения «Диспетчерская вуза» для автоматизации ведения, хранения и обработки указанных данных, с визуализацией сведений о свободных аудиториях по запросам пользователей. Данные могут быть как импортированы из баз данных системы «ІС8-Университет», так и внесены в базу данных в ручном режиме. Для работников диспетчерской в приложении разработана функциональность, охватывающая все основные аспекты работы с аудиторным фондом вуза. Пользователь может осуществлять выбор аудиторий на основе заданных параметров (размер группы, вместимость и оснащение аудитории, желательное время занятий), есть возможность гибкой настройки параметров выбора (например, можно указать приоритетные аудитории для отдельных дисциплин), реализован интерактивный интерфейс для удобного просмотра информации о занятости аудиторий.

Также в рамках проекта было разработано web-приложение, позволяющее преподавателям оперативно получить информацию о свободных аудиториях без обращения в диспетчерскую вуза.

Разработанное desktop-приложение реализовано на языке C# с использованием библиотеки Windows Forms. При разработке web-интерфейса использовались HTML5, CSS3, JavaScript. В качестве сервера баз данных использована СУБД Microsoft SQL Server. Часть функциональности приложения реализована на языке Transact-SQL на сервере баз данных.

Разработанное приложение позволяет повысить эффективность использования аудиторного фонда вуза и снизить временные затраты на составление расписания.

**В. А. Цилюлько**

*(ГрГУ имени Янки Купалы, Гродно)*

## **О РАЗРАБОТКЕ КОМПЬЮТЕРНОЙ ИГРЫ «ПОДЗЕМЕЛЬЕ БЕГАРИТТА» НА UNITY**

Современная игровая индустрия активно развивается в направлении инди-проектов, ориентированных на глубокий сюжет, метанарратив и высокую реиграбельность. Одним из перспективных жанров в данной области является roguelike RPG, сочетающий процедурную генерацию, элементы ролевой системы и нелинейное прохождение. В данной работе рассматривается процесс разработки компьютерной игры «Подземелье Бегаритта», создаваемой с использованием игрового движка Unity.

Разрабатываемый проект представляет собой инди roguelike RPG. Основными особенностями предлагаемой разработки являются сюжетное повествование, метанарратив и вариативность игрового опыта. Структура прохождения уровней сохраняется неизменной, однако каждый уровень подземелья формируется на основе уникального seed-значения, что обеспечивает различие в расположении врагов, игровых структур и внутренних объектов при каждом новом прохождении.

Особое внимание уделяется игровым механикам. В проекте реализована система сохранений, доступная только в определённых внутриигровых структурах, что повышает уровень сложности и стратегического планирования. Отсутствие стандартного отображения показателей здоровья способствует усилению погружения игрока в игровой процесс. Боевая система ориентирована на ближний бой с непосредственным управлением персонажем, а также на активное взаимодействие с объектами и структурами, расположенными на уровне.

Ролевая составляющая игры реализуется через систему развития персонажа, основанную на получении новых способностей,

влияющих на стиль прохождения и тактические возможности игрока. Итоговая цель игры заключается в достижении финального исхода, который напрямую зависит от решений игрока, его старательности и характера взаимодействия с внешним игровым миром.

Таким образом, разработка игры «Подземелье Бегаритта» демонстрирует комплексный подход к созданию инди-проекта, объединяющего элементы roguelike и RPG, нелинейный сюжет и высокую степень реиграбельности.

**М. А. Цитриков, В. В. Орлов**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ИНТЕРАКТИВНОГО САЙТА СПА-САЛОНА**

Архитектура сайта строиться как совокупность разных сервисов, каждый из которых выполняет строго определённую функцию в соответствии с принципом ответственности. Это повышает не только стабильность работы сайта, но и упрощает работу по масштабированию, сопровождению. Я решил выделить два основных компонента: интерфейс, планировщик.

Интерфейс по своей сути является веб-приложением обеспечивающую пользователем возможность взаимодействовать с системой. Благодаря нему пользователь может находить нужную ему информацию о спа-салоне, например, услугах, мастерах, акциях, ценах, времени работы и т.д. Он отвечает за отображение данных, обработку действий пользователя и передачу запросов во внутренние сервисы. Чаще всего именно по нему клиенты получают первое впечатление об организации и плохо выглядящий интерфейс, пусть и правильно работающий, может негативно сказаться на желании человека продолжать находиться на сайте, или вообще пользоваться услугами организации.

Планировщик выполняет роль главного координирующего компонента, который принимает запросы от интерфейса и передаёт их другим внутренним сервисам. Он отвечает за последовательность обработки данных, очереди запросов, согласованность операций.

Взаимосвязь между компонентами происходит благодаря REST архитектуре, что не только обеспечивает большую гибкость благодаря

модульности, но и благодаря той же модульности позволяет независимое развитие для каждого сервиса. Интерфейс передаёт запросы планировщику, планировщик распределяет между сервисами, а потом возвращает результат через интерфейс пользователю. Такая архитектура позволяет создать сайт, который не только будет стабильным даже при высоких нагрузках, но и в будущем модернизировать его или изменять в других целях.

Для выполнения задачи создания сайта поддерживающего подобную архитектуру были выбраны JavaScript, Sublime Text, PHP. Это поможет мне создать сайт, подходящий по всем требованиям для успешного продвижения бизнеса. Пользователи смогут просматривать различные услуги, узнавать об акциях и новостях спа-салона.

**Д. С. Цуранков**

*(ГГУ имени Ф. Скорины, Гомель)*

## **РАЗРАБОТКА ВЕБ-СИСТЕМЫ ИНТЕРНЕТ-МАГАЗИНА С ЧАТОМ ПОДДЕРЖКИ И СИСТЕМОЙ ОЦЕНКИ КАЧЕСТВА ОБСЛУЖИВАНИЯ**

В современных условиях электронной коммерции важную роль играет оперативное взаимодействие между клиентом и службой поддержки. Онлайн-чаты позволяют быстро решать возникающие вопросы пользователей, однако для повышения качества обслуживания необходимо не только обеспечить коммуникацию, но и реализовать механизмы оценки эффективности работы операторов.

В рамках работы разработана веб-система интернет-магазина, включающая каталог товаров, систему обработки заказов и модуль онлайн-чата для общения клиентов с оператором. Особое внимание уделено реализации механизма оценки качества обслуживания, который позволяет пользователям после завершения диалога оставить оценку работы поддержки [1]. Полученные данные сохраняются в системе и используются для формирования статистики качества обслуживания.

Архитектура разработанной системы включает модули управления пользователями, обработки заказов, хранения сообщений чата и анализа пользовательских оценок. Реализован механизм хранения

данных и формирования статистических показателей, позволяющий автоматически вычислять среднюю оценку обслуживания и учитывать все оценки, включая повторные обращения клиентов. Это обеспечивает более объективную оценку качества работы операторов.

Практическая значимость работы заключается в создании веб-решения, которое может быть использовано в интернет-магазинах малого и среднего бизнеса для организации поддержки клиентов и анализа эффективности взаимодействия с пользователями. Реализованная система позволяет повысить качество обслуживания и улучшить пользовательский опыт при работе с интернет-магазином.

### **Литература**

1 Назаров, С. В. Проектирование информационных систем: учебник для вузов / С. В. Назаров. – М.: Юрайт, 2021. – 351 с.

**А. Н. Черкас, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

### **РАЗРАБОТКА ИГРОВОГО ПРИЛОЖЕНИЯ “MADMAN” В СРЕДЕ UNITY**

В современных интерактивных проектах тема безумия часто используется как эстетический прием, но редко рассматривается как результат осознанного морального выбора. Проект «Madman» предлагает переосмысление архетипа героя: через синтез динамичного геймплея и глубокого нарратива игрок проходит путь персонажа, который добровольно жертвует рассудком ради достижения высшей цели. В работе исследуется возможность интеграции психологической трансформации героя в технические системы кроссплатформенного мультиплеерного приложения.

Игра разработана на движке Unity [1] и C# с использованием стека DOTween и системы ECS. В отличие от объектно-ориентированной модели, ECS разделяет данные и логику, что позволяет эффективно обрабатывать тысячи активных сущностей, минимизируя нагрузку на CPU. Это критически важно для реализации масштабных сражений на мобильных устройствах при сохранении стабильной частоты кадров. Применение Burst Compiler обеспечивает оптимизацию кода до уровня нативных инструкций процессора.

Для синхронного взаимодействия до 4 игроков выбран фреймворк Photon Fusion, реализующий механизмы предсказания на стороне клиента и компенсации задержек.

Игровой процесс включает две фазы:

1 Активная фаза (Dread Waves): отражение волн противников с накоплением «Делюзий» – временных навыков, представляющих собой опасные запретные знания.

2 Мета-прогрессия (Цитадель): развитие базового города – физического пространства, выступающего оплотом реальности и человечности.

Разработанный проект демонстрирует, как архитектурные решения Unity, позволяют объединить сложную этическую дилемму о цене подвига с высокопроизводительным игровым процессом

### **Литература**

1 Хокинг, Д. Unity в действии. Мультиплатформенная разработка на C#. / Д. Хокинг. – СПб.: Питер, 2016. – 336 с.

**Ю. А. Чуйко**

*(ГрГУ имени Янки Купалы, Гродно)*

### **РАЗРАБОТКА КЛИЕНТСКОГО ПРИЛОЖЕНИЯ ДЛЯ УПРАВЛЕНИЯ УСЛУГАМИ ВЕБ-ХОСТИНГА**

Цифровизация экономики и увеличение числа веб-ресурсов в Республике Беларусь обуславливают необходимость совершенствования инструментов управления услугами веб-хостинга, обеспечивающих централизованное администрирование хостинговых ресурсов, доменов, виртуальных серверов и сопутствующих сервисов.

Разработка клиентского приложения основывается на интеграции с API хостинг-провайдеров, автоматизации обработки запросов и использовании современных фреймворков для построения пользовательского интерфейса. Архитектурный подход предполагает модульность и масштабируемость, что позволяет адаптировать систему под различные модели обслуживания (shared-hosting, VPS, VDS, облачные решения). Автоматизированный мониторинг состояния серверов (нагрузка CPU, использование памяти, сетевой трафик) обеспечивает своевременное выявление отклонений и минимизацию простоев.

Применение технологий асинхронного взаимодействия и механизмов кэширования повышает производительность клиентского приложения. Использование протоколов безопасной передачи данных и многофакторной аутентификации усиливает защиту учетных записей пользователей. Встроенные аналитические модули позволяют прогнозировать потребление ресурсов и оптимизировать тарифные планы на основе поведенческих данных.

Особое внимание уделяется удобству интерфейса и снижению порога входа для пользователей различного уровня подготовки. Интеграция уведомлений, систем биллинга и технической поддержки формирует единую цифровую экосистему управления услугами. Внедрение такого приложения способствует повышению прозрачности процессов администрирования, сокращению временных затрат и повышению качества обслуживания клиентов.

Таким образом, разработка клиентского приложения для управления услугами веб-хостинга представляет собой актуальное направление в сфере информационных технологий, направленное на повышение эффективности цифровой инфраструктуры и обеспечение устойчивого функционирования интернет-ресурсов в условиях растущей нагрузки и требований к надежности.

**Д. С. Чёрная, Д. С. Кузьменков**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ВЕБ-САЙТА МАГАЗИНА ОДЕЖДЫ С ИСПОЛЬЗОВАНИЕМ HTML5, CSS3, JAVASCRIPT, NODE.JS**

Современный белорусский рынок онлайн-продаж одежды переживает достаточно сильные изменения. Многие годы малые предприниматели вели успешную торговлю через соцсети. Однако с февраля 2026 года такая модель попала под запрет. Это, в свою очередь, создает спрос на переход в легальное поле, которым послужат сайты с доменом «.by». В отличие от торговли на маркетплейсах, где продавец покрывает высокие комиссии и не имеет прямого доступа к клиентской базе, а также вынужден конкурировать с десятками аналогов своего товара, собственный сайт предоставит предпринимателю полный контроль над своим бизнесом.

В процессе проектирования веб-сайта для магазина одежды ключевой задачей при разработке клиентской части стало создание адаптивного и интуитивного интерфейса, который обеспечивает комфортный процесс выбора и покупки товаров. Основной упор сделан на реализацию логики взаимодействия с пользователем, направленной на удержание внимания. Использование семантической верстки HTML5 обеспечило структуру контента, важную для поисковой оптимизации, в то время как CSS3 позволил реализовать гибкую систему стилей. С помощью языка JavaScript реализована сложная логика фильтрации товаров в каталоге. Это реализовано через манипуляции с DOM (объектной моделью документа) и асинхронные запросы к серверу для получения отфильтрованных данных. Также на JavaScript построена работа корзины покупок, где данные временно хранятся в памяти браузера (localStorage), это необходимо для сохранения выбранных товаров даже при случайном закрытии вкладки с магазином.

Серверной платформой для проекта выступила среда Node.js, которая позволила унифицировать язык разработки как на клиенте, так и на сервере, используя JavaScript. Это упростило архитектуру и позволило легко управлять потоками данных. На сервере реализована вся критически важная логика. Во время обработки запросов каталога, сервер принимает GET-запросы с параметрами фильтрации, формирует соответствующие SQL-запросы к базе данных и возвращает клиенту JSON-массив с товарами, соответствующими критериям поиска.

**К. С. Шамко**

*(ГрГУ имени Янки Купалы, Гродно)*

## **РЕАЛИЗАЦИЯ ПРИНЦИПОВ БЕЗОПАСНОЙ FRONTEND-РАЗРАБОТКИ В СОВРЕМЕННЫХ ФРЕЙМВОРКАХ**

В условиях цифровизации безопасность веб-приложений становится критически важной. По данным OWASP, 43 % атак в 2024 году связаны с недостаточной защитой frontend-компонентов. Данная работа рассматривает комплексный подход к обеспечению безопасности с использованием React и TypeScript [1].

Рассмотрим множество уязвимостей  $V = \{v_1, v_2, \dots, v_n\}$ , где каждая характеризуется вероятностью  $P(v_i)$  и ущербом  $D(v_i)$ . Риск определяется как:

$$R = \sum P(v_i) \times D(v_i), i = 1..n$$

К критичным уязвимостям относятся: XSS ( $P(\text{XSS}) \approx 0.73$ ), CSRF, утечки данных через localStorage, компрометация зависимостей (рост на 742 % за 3 года).

Методология защиты основана на многоуровневой обороне [2]:  
– строгая типизация TypeScript снижает runtime-ошибки на 38 %;  
– санитизация данных через DOMPurify с эффективностью  $E = 99.7\%$ ; (3) CSP-заголовки снижают XSS-риски на 87 %;  
– аудит зависимостей с метрикой  $S = 1 - (V_{\text{critical}} \times W_{\text{c}} + V_{\text{high}} \times W_{\text{h}}) / V_{\text{total}}$ .

Применение методологии безопасной frontend-разработки является эффективным инструментом снижения рисков. Интеграция принципов типизации, санитизации, CSP и контроля зависимостей способствует созданию устойчивых веб-приложений.

### Литература

- 1 Cherny, B. Programming TypeScript / B. Cherny. – O'Reilly Media, 2023. – 324 с.
- 2 Hoffman, A. Web Application Security / A. Hoffman. – O'Reilly Media, 2024. – 338 с.

**А. В. Шацкая, Е. Ю. Кузьменкова**  
(ГГУ имени Ф. Скорины, Гомель)

### РАЗРАБОТКА ИНТЕРНЕТ-МАГАЗИНА ПО ПРОДАЖЕ МЕБЕЛИ С ИСПОЛЬЗОВАНИЕМ ФРЕЙМВОРКА DJANGO

Разработка web-приложений является одним из наиболее популярных направлений в настоящее время, позволяя создавать различные продукты: от простых счётчиков количества посещений до полноценных приложений, таких как интернет-магазин, электронная регистрация в больнице и т.д. Именно поэтому было разработано web-приложение по продаже мебели. Оно представляет собой

стандартную основу интернет-магазина, где хранится информация о самом магазине, о пользователях, о товарах при помощи подключения баз данных. Web-приложение разработано с использованием HTML 5, CSS 3, JavaScript, а также Django.

HTML – стандартный язык разметки, который используется для создания различных web-сред. Документы описываются тегами, так называемыми командами, которые сообщают браузеру, каким образом отобразить тот или иной объект. CSS – набор параметров форматирования, который применяется к элементам документа, обычно используются для стилизации элементов web-сред.

JavaScript – язык программирования, разработанный для более сложных структур сайта. JS используется для программирования событий web-страниц.

Django – фреймворк Python, который использует набор инструментов, библиотек, а также шаблонов для создания полноценных web-приложений.

С использованием описанных технологий разработка интернет-магазина обеспечивает современный подход к созданию адаптивного интерфейса. Применение HTML 5 и CSS 3 позволяет создать удобный для пользователя сайт. С помощью JavaScript реализуется динамическая работа сайта (мгновенное обновление корзины, фильтрация каталога, проверка данных в формах), что повышает комфорт взаимодействия с магазином. На основе Django строится логика работы магазина: организовано хранение данных о товарах, пользователях и заказах, реализованы процессы оформления покупок и защищённое управление контентом через административную панель.

**Д. А. Шульга**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ОПТИМИЗАЦИЯ ТЕСТИРОВАНИЯ БЕЗОПАСНОСТИ ВЕБ-ПРИЛОЖЕНИЙ: СРАВНЕНИЕ ПОДХОДОВ НА ОСНОВЕ PYTHON И ДЕКЛАРАТИВНЫХ JSON-КОНФИГУРАЦИЙ**

В связи с внедрением DevOps и Agile безопасность веб-приложений становится обязательным условием разработки. Автоматизация тестирования безопасности – критический фактор своевременного выявления уязвимостей [1].

Цель работы – сравнительный анализ двух подходов: императивного программирования на Python и декларативных JSON/YAML-конфигураций.

Первый подход основан на написании кода с использованием фреймворков (PyTest, Selenium), позволяя реализовывать сложные сценарии атак. Второй использует декларативные конфигурации и DSL (Karate DSL), где тесты описываются структурированными данными [2].

Проведено сравнение по критериям гибкости, читаемости, скорости внедрения и интеграции с CI/CD. Python обеспечивает гибкость для сложных сценариев, но требует высокой квалификации и усложняет аудит. Декларативные конфигурации обеспечивают низкий порог входа, читаемость и простую интеграцию с Git, что оптимально для регрессионного тестирования и проверок OWASP Top 10, однако ограничены возможностями DSL.

Результаты показывают взаимодополняемость подходов. Планируется разработка гибридной методологии: декларативные конфигурации для базовых проверок в CI/CD и Python-скрипты для углубленного анализа уязвимостей, а также создание прототипа фреймворка.

### **Литература**

1 Автоматизированное тестирование: основы и принципы. Selectel: [сайт]. – URL: <https://selectel.ru/blog/test-automation/> (дата обращения: 06.02.2026).

2 Лучшие open-source инструменты для тестирования API в 2025 году. Habr: [сайт]. – URL: <https://habr.com/ru/companies/otus/articles/911768/> (дата обращения: 06.02.2026).

**Е. Ю. Шумский**

*(ГрГУ имени Янки Купалы, Гродно)*

### **АЛГОРИТМ ГИБРИДНОГО ШИФРОВАНИЯ ДЛЯ ОБЕСПЕЧЕНИЯ ПРИВАТНОСТИ ДАННЫХ**

В данном исследовании предлагается алгоритм ограничения доступа со стороны сервера к приватным данным пользователей. То есть, в случае полного взлома сервера или базы данных, никто не должен получить доступ к конфиденциальной информации.

На этапе регистрации с помощью ассиметричного шифрования **ЕСС Р-256** генерируется пара ключей. Закрытый ключ никогда не передается на сервер, шифруется на устройстве с помощью алгоритма **AES-GCM**, где используется ключ шифрования, полученный из пароля с помощью функции деривации ключей **PBKDF2-HMAC-SHA256**. Открытый ключ с остальными данными пользователя отправляется на сервер. То есть, сервер хранит информацию о пользователе для авторизации и данные для шифрования, но не имеет инструмента для расшифровки данных.

При отправке приватных данных на сервер генерируется случайный симметричный ключ (DEK) для шифрования данных. Данные шифруются с помощью алгоритма **AES-GCM**. Далее DEK шифруется открытым ключом владельца и для каждого пользователя, которому владелец дал доступ, создается отдельная копия DEK, зашифрованная открытым ключом этого пользователя. На сервер передаются зашифрованные данные и список, состоящий из пары: идентификатор пользователя и соответствующий зашифрованный DEK.

При чтении приватных данных пользователь загружает зашифрованные данные со списком зашифрованных ключей. Клиентское приложение находит подходящий данному пользователю зашифрованный DEK и расшифровывает его с помощью своего закрытого ключа. После приложение расшифровывает данные с помощью алгоритма **AES-GCM** прямо на устройстве, чтобы сервер не получил доступ к закрытым данным пользователя.

Такой подход помогает создать истинную защищенность данных. Даже в случае взлома сервера или базы данных, взломщик получит лишь список зашифрованных данных. Этот подход гарантирует, что приватные данные останутся конфиденциальными, а информация не может быть использована против пользователей.

**А. П. Шутова, Е. Ю. Кузьменкова**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА КЛИЕНТСКОЙ ЧАСТИ ПРИЛОЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ ASP.NET CORE**

Активное развитие технологий искусственного интеллекта и растущий интерес к персонализированным чат-ботам определяют

актуальность создания удобных платформ для общения с AI-персонажами. Для реализации клиентской части приложения был выбран фреймворк Blazor WebAssembly, позволяющий разрабатывать интерактивные веб-интерфейсы на языке C# без использования JavaScript.

Приложение выполняется непосредственно в браузере пользователя благодаря технологии WebAssembly.

Для построения пользовательского интерфейса применяется библиотека компонентов MudBlazor, предоставляющая готовые Material Design элементы. Управление состоянием приложения осуществляется через паттерн Dependency Injection и контейнер StateContainer.

Взаимодействие с внешними AI-сервисами реализовано через HTTP-клиент с поддержкой асинхронной потоковой передачи данных.

Архитектура приложения включает страницы регистрации и аутентификации с валидацией данных, страницу каталога персонажей с функциями поиска и фильтрации, страницу создания персонажей с предварительным просмотром, страницу чата с отображением истории сообщений и потоковым выводом ответов AI, боковую навигационную панель с отображением списка активных диалогов.

Все компоненты интерфейса стилизованы с применением CSS-анимаций и градиентов, обеспечивая современный визуальный дизайн и плавные переходы между состояниями.

Разработанное веб-приложение позволяет пользователям регистрироваться в системе, просматривать каталог доступных AI-персонажей с описаниями и аватарами, создавать собственных персонажей с настраиваемыми параметрами личности, вести множественные диалоги с различными AI-персонажами, а также сохранять историю переписки.

Данное приложение реализует потоковую передачу ответов от искусственного интеллекта, что обеспечивает плавное отображение генерируемого текста в режиме реального времени и улучшает пользовательский опыт взаимодействия.

**А. А. Щербак, А. В. Лубочкин**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА МИКРОСЕРВИСА ДЛЯ МОНИТОРИНГА АВИАПЕРЕЛЕТОВ В РЕАЛЬНОМ ВРЕМЕНИ**

В современных условиях информационные системы все чаще требуют обработки данных в реальном времени и высокой степени отказоустойчивости. Одной из критически важных областей, где эти требования проявляются в полной мере, является логистика и, в частности, мониторинг авиаперелетов.

Данная работа посвящена исследованию архитектурных подходов и технологий для создания отказоустойчивой и масштабируемой системы отслеживания авиарейсов. В качестве технологической основы для разработки используется язык программирования Java и фреймворк Spring [1], предоставляющие хорошую экосистему для создания корпоративного приложения. Выбран архитектурный стиль Event Driven Architecture [2] для обеспечения гибкой и слабосвязанной коммуникации между микросервисами.

Для организации обмена событий в реальном времени задействована платформа для потоковой обработки данных Apache Kafka [3], которая обеспечивает высокую отказоустойчивость и масштабируемость.

Для хранения данных используются две различные СУБД. Структурированные данные хранятся в реляционной базе данных MySQL. Для работы с большими объемами данных, которые часто меняются, выбрана документоориентированная NoSQL база данных MongoDB.

В результате данной работы с помощью указанных технологий разработан микросервис, обеспечивающий мониторинг авиаперелетов в реальном времени с высокой степенью надежности.

### **Литература**

- 1 Уоллс, К. Spring в действии / К. Уоллс. – 6-е изд. – СПб.: Питер, 2022. – 592 с.
- 2 Ричардсон, К. Микросервисы. Паттерны разработки и рефакторинга / К. Ричардсон. – М.: Вильямс, 2020. – 554 с.
- 3 Шапиро, Г. Kafka: потоковая обработка в реальном времени / Г. Шапиро, Т. Палино. – СПб.: Питер, 2021. – 400 с.

**К. Kurachka, Huanhai Ren, Xuemei Wang**  
(*Sukhoi State Technical University of Gomel, Gomel*)

## **DISCUSSION OF KEY TECHNOLOGIES FOR AUXILIARY DIAGNOSIS OF DEGENERATIVE LUMBAR-SPINE DISORDERS BASED ON MRI ANALYSIS**

Предложен компактный подход к МРТ-диагностике дегенеративных заболеваний поясничного отдела позвоночника с автоматической сегментацией и количественным анализом. Пилотная проверка показала расхождение с экспертной оценкой менее 5 % при более высокой скорости обработки.

Degenerative lumbar spine disorders are a major cause of chronic low back pain and functional limitation in middle-aged and older adults [1]. Lumbar spine MRI is widely used because it can clearly visualize vertebral bodies, intervertebral discs, and the spinal canal in one examination. However, routine interpretation and manual morphometric measurement are time-consuming and depend on reader experience, especially in complex degenerative cases [3]. To improve efficiency and consistency, this study proposes a compact framework for MRI-based auxiliary diagnosis based on automatic segmentation and quantitative feature extraction [2]. The framework includes pre-processing, anatomical segmentation, and quantitative analysis. Pre-processing applies denoising, intensity normalization, and lumbar region-of-interest localization. Anatomical segmentation is performed using a modified U-Net architecture [2]. The segmentation masks of vertebral bodies, intervertebral discs, and the spinal canal are then used to calculate vertebral body height, intervertebral disc thickness, and spinal canal cross-sectional area as objective indicators for MRI-based auxiliary diagnosis.

Primary validation was conducted on a compact cohort composed of public lumbar spine MRI data and anonymized clinical cases, so the current results should be regarded as pilot evidence [3]. The method maintained stable contour continuity for vertebral bodies and intervertebral discs even in cases with signal heterogeneity and degenerative morphology. For representative cases, automatic measurements of L4 vertebral body height, L4-L5 intervertebral disc thickness, and L5 spinal canal cross-sectional area differed from expert manual measurements by less than 5 %. With GPU acceleration,

extraction of major lumbar spine geometric parameters for one case was completed in under 10 s, whereas manual measurement typically required about 10–15 min [3]. These results support the feasibility of MRI-based auxiliary diagnosis using automated segmentation and quantitative analysis, while broader validation across scanners and institutions is still needed.

### **Literatura**

1 Hartvigsen, J What low back pain is and why we need to pay attention[J] / J. Hartvigsen, M. J. Hancock, A. Kongsted, et al. – The Lancet, 2018 – Vol. 10137, № 391. – P. 2356-2367.

2 Kurachka, K. Comparative Analysis of Deep Learning Models for Lumbar Vertebrae Segmentation in MRI Images [C] / K. Kurachka, Huanhai Ren, Xuemei Wang // Pattern Recognition And Information Processing – 2025.– P.176-179.

3 Ronneberger, O U-net: Convolutional networks for biomedical image segmentation[C] / O. Ronneberger, P. Fischer, T. Brox // International Conference on Medical image computing and computer-assisted intervention. Cham: Springer international publishing. – 2015. – P. 234-241.

**A. D. Dolnikov, E. Y. Kuzmenkova**

*(F. Skaryna GSU, Gomel)*

## **DEVELOPMENT OF A MOBILE APPLICATION FOR CREATING AND SEARCHING FOR EVENTS BY INTERESTS USING REACT NATIVE**

В данной работе рассматривается разработанное мобильное приложение для создания и поиска мероприятий по интересам, которое позволяет пользователям легко находить подходящие события и организовывать собственные встречи. Описываются используемые технологии, их основные преимущества и краткие характеристики. Также указаны основные функции приложения.

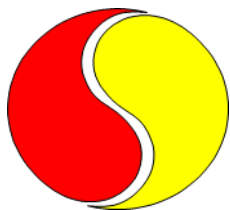
In the modern fast-paced world, socializing, finding like-minded individuals, and engaging in hobbies are essential components of a fulfilling lifestyle. However, people often face difficulties in discovering

local events that align with their personal interests or struggle with the lack of simple tools to organize and promote their own gatherings. The development of a dedicated mobile application for creating and searching for events by interests aims to solve these problems by providing an intuitive, accessible, and user-friendly platform.

During the development of the application, React Native was used as the primary framework. This framework is highly efficient because it translates React Native code of the developed application for both iOS and Android operating systems using a single unified codebase. By utilizing native components, React Native ensures high performance, smooth animations, and a seamless native-like user experience, which are crucial for dynamic mobile applications.

The developed mobile application provides a comprehensive set of features for users to manage their social activities. First of all, the application includes a secure registration and authorization system. Users can easily manage their accounts and personalize their profiles by editing their username, account description, and uploading an avatar photo.

The core functionality of the application allows users to view a customized feed of available events, search for specific gatherings by name, and apply date filters to find exactly what they are looking for. Users can seamlessly sign up to attend events organized by others. The platform empowers users to act as organizers by creating their own events, where they can specify all necessary details, including a cover photo, date, location, and a comprehensive description. The application includes dedicated sections, where users can track the list of events they have successfully registered for, as well as the events they have personally created.



## ПЕРВЫЕ ШАГИ В IT-СФЕРЕ

**К. Е. Ашаев**

*(ГрГУ имени Янки Купалы, Гродно)*

### РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ СИНХРОНИЗАЦИИ РАСПИСАНИЯ ЗАНЯТИЙ С СЕРВИСОМ GOOGLE CALENDAR

В современном образовательном процессе актуальной задачей является оперативное управление временными ресурсами. Ручной перенос расписания из университетской базы данных в личные календари трудоемок и сопряжен с ошибками. Целью работы является создание кроссплатформенного приложения для автоматизации этого процесса. Разработанное программное средство реализовано на языке C# с использованием платформы .NET 8 и фреймворка Avalonia UI, что обеспечивает работу в Windows, macOS и Linux. Архитектура приложения построена на паттерне MVVM, разделяющем логику и интерфейс. Ключевой функционал включает взаимодействие с API университета для получения JSON-данных о занятиях и интеграцию с Google Calendar API. Для безопасности данных реализована авторизация через протокол OAuth 2.0. Пользовательский интерфейс (рисунок 1) реализует поиск преподавателя с автодополнением и выбор временного диапазона для загрузки данных.

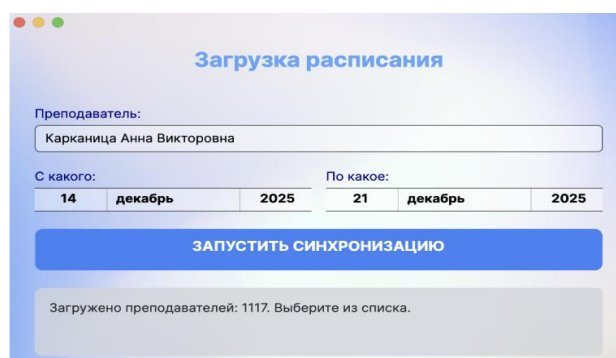


Рисунок 1 – Пользовательский интерфейс

Алгоритм синхронизации выполняет сравнение локальных данных с календарными событиями. Реализована логика полной синхронизации: создание новых событий при появлении их в расписании, обновление параметров (аудитория, время) при их изменении, удаление занятий из календаря Google, если они отменены.

Результатом работы является готовый программный продукт, повышающий эффективность планирования рабочего времени преподавателя.

**М. А. Беликеев, Л. К. Титова**  
(ГГТУ имени П. О. Сухого, Гомель)

## **РАЗРАБОТКА СИСТЕМЫ АВТОНОМНОГО УПРАВЛЕНИЯ И ПРОЦЕДУРНОЙ АНИМАЦИИ МНОГОЗВЕННОГО ОБЪЕКТА В СРЕДЕ UNITY3D**

Реализация естественного поведения сегментированных структур в виртуальном пространстве требует баланса между визуальной достоверностью и вычислительной эффективностью. Традиционные методы физической связки звеньев часто приводят к нестабильности вычислений. Предложенное решение базируется на модели автономного агента, использующей принцип целеориентированной навигации внутри ограниченной зоны (в UNITY, например, VoxCollider). Это позволяет интегрировать объект в сцену как самостоятельную единицу, действующую в заданных пространственных границах без ручного управления.

Процесс навигации строится на циклическом поиске целей: при вхождении головы объекта в радиус чувствительности текущей точки происходит перестроение маршрута к новым случайным координатам. Управление курсом реализуется через расчет углового рассогласования между вектором направления и вектором на цель с применением плавной интерполяции вращения. Для устранения «механистичности» в расчеты внедрен градиентный шум, который добавляет процедурные микроколебания курса, создавая эффект естественного виляния, свойственного биологическим прототипам.

Механика движения хвоста основана на принципе отложенного следования по траектории лидера. Координаты ведущего сегмента

записываются в буфер истории, откуда каждое последующее звено извлекает позицию с фиксированным шагом. Данный подход гарантирует, что все сегменты будут проходить строго по пути следования головы. Это обеспечивает сохранение дистанции между звеньями и полностью исключает визуальные артефакты, такие как наложение элементов друг на друга или разрыв сочленений при резких маневрах.

Предложенная модель отличается высокой производительностью и масштабируемостью за счет эффективного управления памятью и возможности выбора цикла обновления данных. Гибкость настройки динамических характеристик – скорости перемещения, интенсивности рыскания и плотности расположения звеньев – делает данный алгоритм универсальным инструментом для разработки ИИ-противников, моделирования биологических систем или создания визуальных эффектов текучих сред.

**Е. А. Борисёнок, Н. Б. Осипенко**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ИГРОВОГО ПРИЛОЖЕНИЯ “EMERGE” В ЖАНРЕ ИЗОМЕТРИЧЕСКОГО ШУТЕРА НА ЯЗЫКЕ C#**

В условиях стремительного развития индустрии компьютерных игр и постоянного роста требований к качеству интерактивного контента особую актуальность приобретает разработка игровых приложений с динамичным геймплеем и продуманной архитектурой. Жанр изометрических шутеров остаётся востребованным благодаря удачному сочетанию тактического обзора, динамики боя и удобства управления.

Работа посвящена разработке игрового приложения “Emerge” на языке программирования C# с использованием фреймворка Unity, что обусловлено их широкими возможностями для разработки 3D-игр. Проект ориентирован на создание динамичной игровой среды с видом сверху под углом (изометрическая проекция), когда игрок управляет персонажем и взаимодействует с окружением. Использование компонентно-ориентированной архитектуры Unity позволяет разделить систему на независимые модули: управление персонажем, система стрельбы, искусственный интеллект противников,

обработка столкновений, интерфейс и управление состояниями игры. Такой подход обеспечивает гибкость структуры проекта, удобство сопровождения и возможность масштабирования функционала.

Особое внимание в работе уделено реализации системы управления персонажем и механике стрельбы. Разработана система перемещения с учётом изометрической перспективы, реализована обработка пользовательского ввода, логика прицеливания и стрельбы, а также взаимодействие с игровыми объектами. Дополнительно реализованы механики появления врагов, система нанесения урона, обработка коллизий и базовый искусственный интеллект противников.

В результате был создан прототип игрового приложения “Emerge”, представляющий собой функциональный изометрический шутер с базовой игровой логикой, системой управления и боевой механикой. Разработанное решение может служить основой для дальнейшего расширения проекта: добавления новых уровней, улучшения системы искусственного интеллекта, внедрения прогрессии персонажа, расширения арсенала оружия, а также оптимизации производительности и доработки визуальных эффектов.

**Е. В. Говенько**

*(ГГТУ имени П. О. Сухого, Гомель)*

## **НАЧАЛО ПУТИ В IT-СФЕРЕ: ОСОЗНАННЫЙ СТАРТ**

Работа посвящена анализу основных шагов, которые необходимо предпринять абитуриента и начинающему специалисту для успешного старта в IT-сфере. Сегодня информационные технологии – одна из самых привлекательных и перспективных отраслей для построения карьеры, однако, как показывает практика, отсутствие системного подхода на начальном этапе становится главной причиной разочарований и неудач. Осознанный старт – это не выбор «самого легкого языка программирования», а выстраивание фундаментального понимания профессии и её базовых принципов [1].

Первый и самый важный шаг – освоение базы Computer Science: алгоритмы, структуры данных, архитектура ПК. Эти знания вне времени [2]. А. Бхаргава помогает преодолеть страх перед алгоритмами через визуализацию, В. Феррейра Фило дает практико-ориентированный минимум. И. Кырчумару отмечает: для первого

Материалы XXIX Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 23–25 марта 2026 г.

---

проекта достаточно 50 % знаний, остальное доучивается в процессе. Второй шаг – развитие soft skills. Р. Мартин формирует кодекс профессионала: отказ от «геройства», ответственность за код [3]. Дж. Сонмез учит стратегии развития в турбуентной среде [1]. Кырчумару подчеркивает: умение объяснять сложное ценнее знания синтаксиса.

Техстек – лишь инструмент. Главная проблема новичка – «паралич выбора». Выход – переход от потребления контента к созданию продукта. Ф. Доглио дает план роста: от кода до управления проектами [1].

Успех старта в IT определяет не количество купленных курсов или выученных фреймворков, а исключительно качество базы: фундаментальные знания Computer Science, развитые коммуникативные навыки и психологическая готовность к перманентному обучению. Осознанный старт – это марафон, построенный на фундаменте, а не спринт за модной технологией.

### **Литература**

1 Доглио, Ф. Путь инженера-программиста / Ф. Доглио. – М., 2025. – 238 с.

2 Альтхофф, К. Computer Science для программиста-самоучки / К. Альтхофф. – СПб., 2025. – 237 с.

3 Мартин, Р. Идеальный программист / Р. Мартин. – СПб., 2022. – 224 с.

**Н. А. Гордейчик**

*(ГрГУ имени Янки Купалы, Гродно)*

## **КЛАССИФИКАЦИЯ СОВРЕМЕННЫХ ПРИКЛАДНЫХ ПРОГРАММНО-АППАРАТНЫХ СИСТЕМ**

Современный этап развития информационных технологий характеризуется переходом от чисто программных решений к интеграции цифрового кода с физическими устройствами. Прикладные программно-аппаратные системы (ПАС) представляют собой единство вычислительной платформы и специализированного программного обеспечения, ориентированного на управление объектами реального мира.

Анализ источников [1–4] выявил четыре типа ПАС. Первый – системы с готовым управляющим ПО без операционной системы [1]. Второй – системы реального времени RTOS (FreeRTOS, QNX) для задач с жесткими требованиями к таймингу [4]. Третий – встраиваемые Linux-системы (Yocto, Buildroot) для сетевых приложений [2]. Четвертый – промышленные контроллеры ЧПУ высокой надежности [3].

Классификация по уровню реального времени: мягкое (до 10 мс), жесткое (до 1 мс), твердое (микросекунды) [4]. По ресурсам: от 2 КБ памяти (микроконтроллеры) до сотен мегабайт (Linux-системы). Метод позволяет систематизировать ПАС для выбора платформы под задачу.

### **Литература**

1 Марусов, А. Разработка программно-аппаратного комплекса на основе систем реального времени / А. Марусов // Современные технологии автоматизации. – 2010. – № 2. – С. 64–74.

2 Симмондс, К. Встраиваемые системы на основе Linux / К. Симмондс. – М.: ДМК Пресс, 2020. – 560 с.

3 Дьяков, И. Ф. Особенности использования интерфейса программы на металлорежущем станке MAZAK VARIAXIS / И. Ф. Дьяков // Автоматизация. Современные технологии. – 2019. – Т. 73, № 8. – С. 348–355.

4 Абушаев, А. А. Системы реального времени / А. А. Абушаев. – М.: Горячая линия - Телеком, 2018. – 320 с.

**А. А. Жукевич**

*(ГрГУ имени Янки Купалы, Гродно)*

### **ПЛАНИРОВЩИК ЗАДАЧ**

«Планировщик задач» – это инструмент, созданный для организации личной и профессиональной деятельности. Он заменяет традиционные бумажные записи, позволяя систематизировать задачи, управлять приоритетами и получать своевременные напоминания, чтобы ни одно важное дело не осталось без внимания.

Разработка велась с использованием современной системы .NET для обеспечения высокой производительности и надёжности.

Само приложение написано на C#. Пользовательский интерфейс реализован с помощью кроссплатформенного фреймворка Avalonia UI, что обеспечивает единый, адаптивный и современный внешний вид в различных операционных системах. Все данные хранятся локально в компактном бинарном формате BSON через сериализацию. Такой подход делает приложение автономным, быстрым и не требующим установки дополнительного серверного программного обеспечения.

Архитектура приложения построена по модульному принципу с чётким разделением ответственности для упрощения поддержки и развития. Все функциональные обязанности распределены между тремя компонентами. Основу системы составляет клиентское приложение AppUI, разработанное по шаблону Model-View-ViewModel (MVVM) и полностью ответственное за отзывчивый графический интерфейс и взаимодействие с пользователем. Работу с данными инкапсулирует общий уровень DataLayer, который обеспечивает целостность информации через сериализацию в BSON и предоставляет репозитории для работы с задачами и категориями. Фоновый мониторинг и операции с уведомлениями вынесены в две автономные службы: MonitorService отслеживает время выполнения задач, а Notificator отвечает за показ всплывающих системных уведомлений через API операционной системы.

Итогом работы стало создание целостного, отзывчивого и ресурсоэффективного кроссплатформенного приложения с функциональностью планировщика задач.

**Е. А. Зайко**

*(БрГУ имени А. С. Пушкина, Брест)*

## **РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ ВЕБ-ПРИЛОЖЕНИЯ НА REACT С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИЙ NODE.JS И EXPRESS**

Современная веб-разработка характеризуется углублением интеграции между клиентской и серверной частями приложений, что открывает широкие возможности для создания полнофункциональных систем на основе единого языка программирования. Библиотека React отвечает только за интерфейс.

Она не может самостоятельно загрузиться в браузере – для этого нужен сервер, который раздает файлы. Обычно в роли такого сервера выступает Node.js вместе с Express.

Express – фреймворк, написанный на языке JavaScript, который применяют в разработке бизнес-логики мобильных приложений и сайтов. Express работает в Node.js – среде выполнения кода для языка JavaScript [1].

Целью настоящей работы является проектирование и реализация серверного компонента веб-приложения на React с применением технологий Node.js, Express и MongoDB. Актуальность исследования подтверждается ростом числа full-stack проектов, требующих масштабируемой и поддерживаемой серверной архитектуры.

В процессе исследования был выполнен ряд этапов. Спроектирована логическая модель базы данных, определены основные коллекции: пользователи, проекты, задачи, комментарии, уведомления реализована серверная логика с использованием фреймворка Express. Для работы с MongoDB применена библиотека Mongoose, представляет собой специальную ODM-библиотеку (Object Data Modelling) для работы с MongoDB, которая позволяет сопоставлять объекты классов и документы коллекций из базы данных [2].

### Литература

1 Что такое Express: [сайт]. – URL: <https://workspace.ru/tools/framework/express/>. (дата обращения: 11.02.2026).

2 Mongoose: [сайт]. – URL: <https://metanit.com/web/nodejs/6.6.php>. (дата обращения: 11.02.2026).

**В. Э. Закарян, Е. В. Комракова**  
(ГГТУ имени П. О. Сухого, Гомель)

## **АРХИТЕКТУРА ИНТЕЛЛЕКТУАЛЬНОЙ WEB-ПЛАТФОРМЫ ЭЛЕКТРОННОЙ КОММЕРЦИИ ДЛЯ ПОДБОРА СОВМЕСТИМЫХ КОМПЬЮТЕРНЫХ КОМПЛЕКТУЮЩИХ**

Развитие электронной коммерции требует внедрения интеллектуальных инструментов поддержки принятия решений при выборе

технически сложных товаров. При формировании конфигурации персонального компьютера пользователь должен учитывать множество взаимосвязанных характеристик, что делает задачу проверки совместимости многопараметрической.

Современные методы искусственного интеллекта позволяют автоматизировать процессы анализа данных и выявления закономерностей в больших массивах технической информации. Это делает возможным переход от жёстко заданных алгоритмов проверки к адаптивным моделям классификации.

Целью работы является проектирование архитектуры *web*-платформы электронной коммерции с интеграцией нейросетевого модуля анализа совместимости комплектующих.

Архитектура системы включает клиентскую часть, серверную бизнес-логику, базу данных технических характеристик и модуль машинного обучения. Обучение модели осуществляется на размеченных данных конфигураций персональных компьютеров с использованием современных методов оптимизации нейронных сетей.

Обмен данными между компонентами реализуется посредством *REST API*, что обеспечивает гибкость развертывания и возможность дальнейшего масштабирования системы. Алгоритм работы платформы предусматривает формирование вектора параметров, выбранных комплектующих, передачу его в интеллектуальный модуль и получение вероятностной оценки совместимости.

Предложенная архитектура обеспечивает автоматизацию анализа конфигураций, снижение вероятности пользовательских ошибок и возможность дальнейшего расширения функционала за счёт внедрения рекомендательных механизмов. Реализация платформы ориентирована на применение в сфере электронной коммерции компьютерного оборудования и направлена на повышение качества цифровых сервисов.

**С. М. Заяц**

*(ГГТУ имени П. О. Сухого, Гомель)*

## **РЕКОМЕНДАЦИИ ДЛЯ НАЧИНАЮЩИХ В ИТ-СФЕРЕ**

ИТ – это не единая сфера, а множество направлений: разработка, тестирование, аналитика, DevOps, дизайн и т.д. При определении

Первые шаги в IT-сфере

цели выбор направления должен быть осознанным, основанным на личных интересах и склонностях, а не только на желании высокой зарплаты.

Ключевой этап программирования – создание собственного проекта. Проект в портфолио говорит работодателю о ваших навыках больше, чем любой сертификат. На первых порах новичок сталкивается с огромным объемом незнакомой информации и чувствует себя «глупым». Важно понять: это нормально.

IT-сфера требует постоянного обучения, и чувство дискомфорта – индикатор роста, а не профессиональной непригодности. Изоляция – враг новичка.

Участие в профильных чатах, конференциях (онлайн или офлайн), митапах и форумах позволяет быстрее находить ответы на вопросы, узнавать о трендах и заводить полезные знакомства, которые могут привести к первому офферту.

Маркетинговые обещания быстрых результатов часто приводят к разочарованию. Первые шаги в IT – это марафон, а не спринт.

Реальный срок входа в профессию с нуля до уровня, уверенного Junior-специалиста обычно занимает от 9 месяцев до 1.5 лет интенсивной работы.

Многие новички тратят месяцы на выбор «идеального» языка программирования или курса, вместо того чтобы начать. На начальном этапе выбор конкретного инструмента (Python, Java, JavaScript) менее важен, чем понимание логики программирования.

Лучше выбрать что-то одно и начать, чем искать идеал и не сделать ничего. Игнорирование английского языка сильно тормозит развитие. Вся актуальная документация, лучшие статьи, ответы на StackOverflow и современные библиотеки появляются на английском. Без него Junior быстро упирается в «стеклянный потолок» информации.

## Литература

1 Столяров, А. В. Программирование: введение в профессию / А. В. Столяров. – 2022. – 25-37 с.

**С. В. Зверко, С. В. Киргинцева**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ ОНЛАЙН-ПРОДАЖИ ЖЕЛЕЗНОДОРОЖНЫХ БИЛЕТОВ С ИСПОЛЬЗОВАНИЕМ PHP, MYSQL И BOOTSTRAP**

В современных условиях цифровизации транспортной сферы особую актуальность приобретает разработка веб-приложений, обеспечивающих автоматизацию процессов продажи и бронирования билетов. Разработано веб-приложение для онлайн-продажи железнодорожных билетов с использованием HTML, CSS, Bootstrap, PHP и СУБД MySQL [1–3]. Приложение предназначено для просмотра расписания поездов, поиска маршрутов по заданным параметрам, а также бронирования и покупки билетов в режиме реального времени. Пользователю предоставляется возможность регистрации и авторизации в системе, просмотра истории заказов и управления личными данными. Теоретической основой разработки является использование клиент-серверной архитектуры, принципов проектирования реляционных баз данных и технологий взаимодействия серверной и пользовательской частей веб-приложения. Серверная логика реализована на языке PHP с применением запросов к базе данных MySQL, что обеспечивает хранение и обработку информации о пользователях, поездах и бронированиях.

Для повышения удобства использования системы реализован адаптивный пользовательский интерфейс с применением фреймворка Bootstrap, обеспечивающий корректное отображение приложения на различных устройствах. Разработанное веб-приложение может быть использовано в качестве основы для создания полнофункциональных систем онлайн-продажи билетов и расширено за счёт добавления модулей оплаты и аналитической отчётности.

### **Литература**

- 1 Дакетт, Дж. HTML и CSS. Разработка и дизайн веб-сайтов / Дж. Дакетт. – М.: Эксмо, 2021. – 512 с.
- 2 Никсон, Р. PHP, MySQL и JavaScript. Исчерпывающее руководство / Р. Никсон. – СПб.: Питер, 2020. – 768 с.
- 3 Отвелл, М. Bootstrap. Адаптивная вёрстка для современных веб-приложений / М. Отвелл. – М.: ДМК Пресс, 2019. – 320 с.

**М. П. Касянович**  
(ГрГУ имени Янки Купалы, Гродно)

## **ПРИЛОЖЕНИЕ ДЛЯ ПЛАНИРОВАНИЯ ЗАДАЧ НА ЯЗЫКЕ JAVA**

Эффективное управление задачами в условиях современной цифровой среды требует специализированных инструментов, обеспечивающих структурированность и гибкость. В рамках данной работы реализовано full-stack веб-приложение “Taskcreator”, предназначенное для комплексного планирования и контроля выполнения задач с поддержкой полного набора CRUD-операций и развитой системы фильтрации. Система опирается на три основные сущности: базовую задачу, командную задачу как её специализированное расширение и связанное напоминание, при этом жёстко регламентирует жизненный цикл – от создания до завершения или отмены. Технологической основой серверной части выступили Java и фреймворк Spring Boot, обеспечивающие построение надёжного REST API, в то время как для хранения данных выбрана реляционная СУБД PostgreSQL, гарантирующая целостность и возможность выполнения сложных запросов. Клиентский интерфейс реализован как одностороннее приложение с использованием HTML, CSS и нативного JavaScript, что позволило создать отзывчивый и интуитивно понятный интерфейс с динамическими таблицами, пагинацией и модальными окнами. Архитектура проекта построена по многослойной модели, разделяющей ответственность между компонентами, что облегчает поддержку и развитие системы. Особое внимание уделено реализации сложной бизнес-логики, включающей контроль перегрузки исполнителей, валидацию временных параметров и автоматическое обслуживание данных. В результате разработано готовое к эксплуатации веб-приложение, которое демонстрирует сбалансированный подход к full-stack разработке и представляет собой основу для дальнейшего развития в сторону полноценного корпоративного решения. Его модульная конструкция и чистая архитектура позволяют без значительных усилий интегрировать новые функции, такие как расширенная аналитика, системы уведомлений или командная коллаборация.

**А. В. Кирчук, В. С. Закревская**  
(ГГУ имени Ф. Скорины, Гомель)

## **ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ИГРОВЫХ УРОВНЕЙ В 2D-ПЛАТФОРМЕРЕ**

Современная игровая индустрия характеризуется устойчивым интересом к жанру 2D-платформеров. Они сохраняют популярность благодаря доступности, выразительному визуальному стилю и глубоким механикам [1].

Цель работы – исследование процесса проектирования игровых уровней и его практическая реализация в среде Godot Engine. Для достижения цели решены задачи анализа теоретических основ геймдизайна, выбора технических средств, разработки структуры и визуального оформления трех уровней, а также создания систем интерфейса и сохранения прогресса.

Проектирование уровней выполнялось в три этапа: прототип, тестирование механик, графическая доработка. Первый уровень знакомит с передвижением, прыжком и рывком. Второй уровень вводит шипы и объект «Восстановитель рывка», требуя точного расчета и прыжков от стен. Третий уровень добавляет движущиеся платформы, повышая динамику.

Разработаны меню выбора уровней с отображением прогресса, механизм сохранения данных и меню паузы. Сохранение отслеживает пройденные уровни и собранные кристаллы.

Созданы три функциональных уровня, демонстрирующих принципы геймдизайна – постепенное усложнение, обучение через процесс и визуальное разнообразие (рисунок 1). Результаты могут быть использованы для расширения проекта, включая полноценную кампанию или процедурную генерацию.



Рисунок 1 – Результат проектирования уровней

Таким образом, работа подтвердила эффективность Godot Engine как инструмента для разработки 2D-платформеров, а предложенный подход к проектированию уровней может служить основой для создания качественных игровых продуктов.

### Литература

1 Роджерс, С. Level Up! Руководство по созданию классных видеоигр / С. Роджерс. – М.: ДМК Пресс, 2018. – 400 с.

**Д. С. Кособуцкая, Н. Б. Осипенко**  
(ГГУ имени Ф. Скорины, Гомель)

## **ИНТЕРАКТИВНАЯ СИСТЕМА ВИЗУАЛИЗАЦИИ ДАННЫХ В СРЕДЕ TOUCHDESIGNER**

Традиционные методы компьютерного зрения зачастую ограничиваются отслеживанием положения объектов, однако существует потребность в более художественных и эстетически выразительных подходах к интерпретации движений, где образ рождается не из контура, а из поведения множества мельчайших элементов.

Работа посвящена описанию реализации интерактивной системы визуализации, закрывающей потребность в создании уникального визуального языка для аудиовизуальных перформансов и интерактивных инсталляций, где человеческое движение превращается не в картинку, а в абстрактную композицию. Реализация выполнена в среде визуального программирования TouchDesigner с использованием встроенного языка Python для написания скриптов, управляющих логикой взаимодействия и поведением элементов. TouchDesigner предоставляет широкий спектр библиотек для компьютерного зрения, что является важным фактором для обеспечения быстрой разработки и отладки системы при работе с тысячами точек в реальном времени, его компонентно-ориентированная архитектура позволяет разделить разрабатываемую систему на независимые модули: захват изображения и выделение силуэта, расчет поведения точек и финальный рендеринг, что обеспечивает гибкость настройки и возможность дальнейшего расширения функционала. Созданный прототип функциональной интерактивной системы представляет собой интерактивное облако точек, трансформирующееся в реальном времени, следуя за движениями объекта и повторяя его контуры.

Разработку можно использовать в качестве художественного инструмента для создания генеративных визуализаций в реальном времени на концертах, выставках или в театральных постановках. В дальнейшем планируется расширение функционала: добавление возможностей по изменению характера поведения точек (например, придание им свойств текучести или упругости), а также интеграция с аудиоанализом для синхронизации визуальных изменений с музыкальным ритмом.

**Д. В. Курьянович, Е. В. Комракова**  
(ГГТУ имени П. О. Сухого, Гомель)

## **АВТОМАТИЗИРОВАННАЯ СИСТЕМА УЧЕТА УСЛУГ ВЕТЕРИНАРНОЙ КЛИНИКИ С САЛОНОМ ГРУМИНГА**

Деятельность ветеринарной клиники, совмещенной с салоном груминга, предполагает ведение учета по двум взаимосвязанным, но технологически различным направлениям. Разработанная автоматизированная система, построенная с использованием платформы «1С: Предприятие 8.3», решает задачи комплексного учета деятельности, обеспечивая накопление и обработку сведений о пациентах, оказанных услугах и товарно-материальных ценностях в едином информационном пространстве.

В качестве ключевого элемента конфигурации выступает регистр сведений «Карточка пациента», обеспечивающий ведение структурированных данных по каждому животному. Наряду с контактной информацией владельца и основными характеристиками питомца, в карточке предусмотрены специализированные медицинские разделы: данные о проведенных вакцинациях с функцией отслеживания очередных сроков, перечень перенесенных заболеваний, информация о хронических состояниях и аллергических реакциях, а также блок анамнеза для фиксации сведений в хронологическом порядке. Целостность информационной базы поддерживается за счет установления ссылочных отношений между карточкой животного и данными его владельца.

Для поддержки грумерского направления разработан объект «Карта ухода», предназначенный для регистрации технологических

особенностей обработки животных. Структура карты включает реквизиты для фиксации типа шерсти, рекомендуемой частоты проведения стрижки, а также текстовое поле для внесения служебных комментариев. Такой подход обеспечивает сохранение преемственности при работе различных специалистов и формирует массив данных об индивидуальных потребностях каждого питомца.

Учет товарно-материальных ценностей реализован в подсистеме складского учета на основе регистров накопления, поддерживающих партионный учет расходных материалов. В системе автоматизированы процессы фиксации поступлений товаров с привязкой к партиям и отслеживанием сроков годности, а также списание материалов непосредственно при регистрации оказанной услуги.

**В. В. Лапин**

*(ГрГУ имени Янки Купалы, Гродно)*

## **РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ ПЛАНИРОВАНИЯ ЗАДАЧ С ФУНКЦИЯМИ УВЕДОМЛЕНИЙ И КАТЕГОРИЗАЦИИ**

Характерной чертой развития современного общества является высокая степень занятости и многозадачности. В этой связи особую актуальность приобретает использование систем персонального планирования, позволяющих повысить эффективность личного и профессионального тайм-менеджмента.

Одним из таких приложений является планировщик задач, оснащенный функциями уведомлений и категоризации, позволяющий освободить ментальные ресурсы для непосредственной деятельности. Приложение-планировщик спроектировано с учетом основных функциональных требований: создание, редактирование и удаление задач; категоризация по приоритету; установка срока выполнения; настройка напоминаний; разделение задач на активные, просроченные и выполненные.

Для разработки приложения выбран язык программирования C#. Для реализации приложения использован современный технологический стек на основе .NET MAUI, что позволило создать приложение с единой кодовой базой для различных операционных систем.

С учетом требований к производительности реализована гибкая архитектура, сочетающая элементы паттерна Model-View-ViewModel (MVVM) с традиционным event-driven подходом.

Пользовательский интерфейс приложения (рисунок 1) с цветовым кодированием приоритетов и сроков выполнения задач обеспечивает понятную навигацию и удобное взаимодействие с задачами.

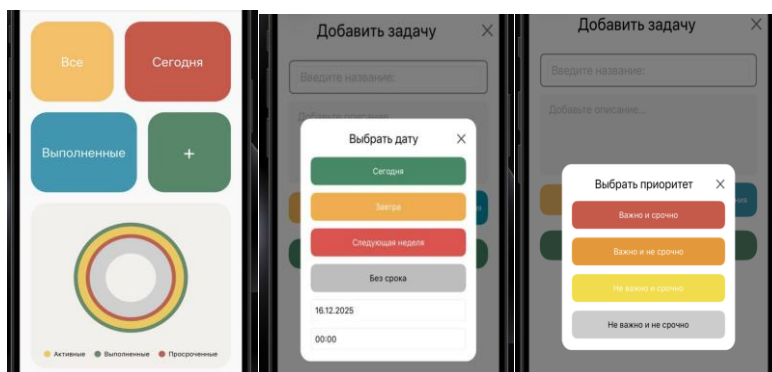


Рисунок 1 – Внешний вид приложения-планировщика

Разработанное приложение может рассматриваться как инструмент личной продуктивности.

**Ю. В. Матафонов, Н. Б. Осипенко**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА ПРОГРАММНОГО АГЕНТА ДЛЯ АВТОМАТИЗАЦИИ ТОРГОВЛИ ИНВЕНТАРЕМ В ИГРЕ PATH OF EXILE НА ЯЗЫКЕ PYTHON**

Индустрия многопользовательских онлайн-игр (MMORPG) в настоящее время представляет собой не только развлекательную сферу, но и сложные экономические симуляторы. Так, игра Path of Exile (PoE) выделяется среди конкурентов своей уникальной экономической моделью из-за отсутствия в ней единой валюты и осуществления торговли посредством бартера различными расходными материалами. Динамика рынка в Path of Exile сопоставима с реальными биржевыми торгами, цена игровых предметов зависит от множества факторов: текущей «меты» (популярности игровых стратегий), времени суток и стадии игрового сезона (лиги).

При этом объемы данных, генерируемых игроками в публичных вкладках сундуков, исчисляются терабайтами информации, что делает ручной анализ рынка неэффективным.

Эффективность персонажа напрямую зависит от экипировки, которую зачастую невозможно получить исключительно путем «фарма» (самостоятельной добычи), что вынуждает игроков обращаться к рынку. Объектом исследования является экономическая система игры Path of Exile, а предметом – методы автоматизированного сбора и анализа данных о виртуальных товарах.

Работа посвящена созданию десктопного приложения “PoE Trade Agent”, которое автоматизирует торговые операции и упрощает процесс анализа цен. Разработанное программное приложение представляет собой автоматизированного агента (торгового бота), предназначенного для оптимизации взаимодействия пользователя с торговой площадкой игры.

Основное назначение системы заключается в решении проблемы информационной перегрузки, предоставляя пользователю инструмент для алгоритмической торговли, основанной на математическом анализе данных и мгновенной реакции на изменения рынка за счет минимизации времени, затрачиваемого игроком на рутинные операции поиска, анализа стоимости и приобретения предметов.

Приложение ориентировано на опытных игроков и трейдеров, стремящихся получить конкурентное преимущество за счет скорости обработки информации и автоматизации механических действий.

Агент разработан на Python с использованием PyQt5 для графического интерфейса и SQLite для хранения данных. Асимметричная библиотека aiohttp обеспечивает быстрое взаимодействие с высоконагруженным API игры, позволяя обрабатывать большое количество данных. Также внедрены алгоритмы предиктивной аналитики на базе Scikit-learn, позволяющие определять ценовые тренды и предоставлять рекомендации по сделкам.

Созданный прототип обеспечивает мониторинг рынка в реальном времени и автоматизацию сообщений. В дальнейшем планируется внедрение более сложных моделей машинного обучения и переход на клиент-серверную архитектуру для обхода лимитов API.

**К. В. Мещанинова, В. И. Токочаков**  
(ГГТУ имени П. О. Сухого, Гомель)

## **РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ УЧЕТА ПОСЕЩАЕМОСТИ И УСПЕВАЕМОСТИ**

В представленной работе рассматриваются этапы проектирования и программной реализации информационной системы, предназначенной для автоматизации процессов учета контингента обучающихся, контроля посещаемости занятий и мониторинга успеваемости в учреждениях дополнительного образования.

Актуальность разработки обусловлена необходимостью перехода от бумажного документооборота к современным цифровым решениям. В качестве средств разработки выбрана интегрированная среда Microsoft Visual Studio и язык программирования C# на платформе .NET. Графический интерфейс пользователя реализован с использованием технологии Windows Forms. Хранение и обработка данных осуществляются с помощью Microsoft SQL Server. Взаимодействие с базой данных реализовано посредством технологии ADO.NET.

Важным этапом разработки стала реализация модуля идентификации и аутентификации пользователей. Администратор имеет полный доступ к управлению справочниками, а преподаватель получает доступ только к ведению электронного журнала закрепленных за ним групп. Это обеспечивает защиту персональных данных. База данных спроектирована с учетом требований нормализации. Выделены ключевые сущности: «Учащийся», «Преподаватель», «Курс», «Группа», «Расписание».

Основные функции системы:

- модуль «Контингент»: ведение личных карточек учащихся с указанием контактных данных родителей;
- модуль «Учебный процесс»: формирование учебных групп, закрепление их за преподавателями;
- электронный журнал: визуальная сетка расписания с возможностью быстрой отметки статуса присутствия и выставления баллов;
- модуль отчетности: автоматическая генерация сводной ведомости посещаемости за выбранный период.

Внедрение разрабатываемой информационной системы в деятельность учреждения дополнительного образования позволит сократить время на обработку отчетной документации, исключить ошибки ручного ввода данных и повысить прозрачность учета посещаемости и успеваемости обучающихся.

**А. Ю. Мусафиров, Е. В. Комракова**  
(ГГТУ имени П. О. Сухого, Гомель)

## **ПРИМЕНЕНИЕ ИЕРАРХИЧЕСКОГО ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ В СТОХАСТИЧЕСКИХ СРЕДАХ СТРАТЕГИЧЕСКИХ ИГР**

В стратегиях реального времени (*RTS*) сложность для ИИ (искусственного интеллекта) обусловлена большим количеством возможных ситуаций и действий ( $10^{26}$  и больше), а также неполной информацией из-за «Тумана войны». В данной работе рассматривается подход к созданию агента на основе использования иерархического обучения с подкреплением (*HRL*), чтобы обойти ограничения обычных архитектур.

Главная трудность применения стандартных алгоритмов, таких как *PPO*, в стратегиях – редкие поощрения при планировании на длительный срок. Для решения этой задачи предлагается двухуровневая структура агента. Верхний уровень (макро-менеджер) принимает стратегические решения (развитие экономики, выбор технологий) и обновляется редко. Нижний уровень (микромонтроллер) получает цели от менеджера и переводит их в отдельные действия (перемещение войск, использование умений) в реальном времени.

В технической части применяются сверточные нейронные сети (*ResNet*) для обработки пространственных данных, к примеру, карты высот и позиций войск, а также рекуррентные блоки (*LSTM*) для учета контекста в условиях частичной видимости (*POMDP – Partially Observable Markov Decision Process*). Чтобы быстрее обучить модель, используется метод *Action Masking*, который убирает явно невозможные действия, и обучение с самообучением (*Self-Play*) для автоматического создания учебного плана. Для повышения устойчивости

к шумам и неполным данным, в архитектуру интегрированы механизмы внимания (*Attention Mechanisms*), позволяющие модели динамически фокусироваться на наиболее релевантных участках входных данных.

Эксперименты показывают, что разделение задачи на части помогает агенту находить баланс между долгосрочным планированием и быстрым микроконтролем, и он работает лучше, чем скриптовые боты и обычные *RL*-агенты в условиях неопределенности.

**Р. А. Недбайло, В. И. Токочаков**  
(ГГТУ имени П. О. Сухого, Гомель)

## **ИНФОРМАЦИОННАЯ СИСТЕМА ВВЕДЕНИЯ И УЧЁТА ОСНОВНЫХ СРЕДСТВ В БЮДЖЕТНЫХ ОРГАНИЗАЦИЯХ**

Государственное учреждение «Центр по обеспечению деятельности бюджетных организаций Лоевского района» осуществляет обслуживание 35 бюджетных организаций района, включая учреждения образования, культуры и сельские исполнительные комитеты. Учёт основных средств – транспортных средств, вычислительной техники, мебели, оборудования – сопряжён с обработкой значительных массивов данных.

В настоящее время учёт ведётся с использованием бумажных инвентарных карточек формы ОС-6, накладных на внутреннее перемещение и актов на списание. Данный подход имеет существенные недостатки: высокая трудоёмкость поиска информации по инвентарным номерам, отсутствие централизованной базы данных, сложность формирования сводных отчётов по амортизации и переоценке, риск потери данных при ручном вводе.

Целью работы является разработка информационной системы для автоматизации учёта основных средств. Для реализации выбрана среда разработки Microsoft Visual Studio и язык программирования C# (платформа .NET, Windows Forms). В качестве СУБД используется SQLite – встраиваемая система, не требующая установки серверного программного обеспечения и обеспечивающая высокую скорость работы с локальными данными.

База данных информационной системы включает справочники подразделений (35 обслуживаемых организаций), материально-ответственных лиц, классификатор групп основных средств, а также основную таблицу реестра, содержащую инвентарный номер, наименование объекта, первоначальную стоимость, дату принятия к учёту, срок полезного использования, сумму начисленной амортизации, остаточную стоимость и текущий статус объекта. Структура базы данных спроектирована на основе анализа реальных инвентарных карточек формы ОС-6 и накладных на внутреннее перемещение, используемых в организации.

Информационная система реализует следующие функции: ведение реестра основных средств с учётом их поступления, внутреннего перемещения и выбытия; учёт закрепления оборудования за материально-ответственными лицами и подразделениями; автоматический расчёт амортизации и остаточной стоимости; учёт переоценки, ремонта и модернизации; формирование печатных форм документов; поиск и фильтрация данных по инвентарному номеру, группе, подразделению и материально-ответственному лицу.

Внедрение системы позволит сократить время на обработку учётной документации, исключить ошибки ручного ввода и повысить прозрачность учёта государственного имущества в бюджетных организациях Лоевского района.

**Д. Д. Новицкая**

*(ГрГУ имени Янки Купалы, Гродно)*

## **РАЗРАБОТКА 2D-ИГРЫ «ПУШИСТЫЙ СОН»**

На сегодняшний день очень популярны игры, в которых приятный и простой визуал сочетается с понятными механиками и интересным сюжетом. Это все прекрасно уместается в 2D-играх в жанре приключения. Особенно, если в игре присутствуют интересные задания и необычные повороты событий.

Жанр приключения – это один из основных жанров в игровом пространстве, в котором геймплей сосредоточен на исследовании локаций, взаимодействия с окружением и общении с персонажами.

Игра «Пушистый сон» является возможностью окунуться в мультяшный 2D-мир, где игрока ждут интересные локации и не самые простые задачи в виде загадок, квестов и головоломок.

Главный герой игры – обычный человек, который устал от рутинной работы и обязанностей. Перед сном к нему пришла кошка, которой он пожаловался на свою жизнь и отметил, что быть котом проще и он бы не отказался побыть им хотя бы сутки. Это и стало его ключевым желанием. Он проваливается в сон, где вместо ног и рук у него лапы, вырос длинный хвост, а на макушке заостренные ушки. Теперь, для пробуждения, ему необходимо пройти несколько испытаний, с которыми сталкиваются многие четверолапые друзья.

Игрок сможет выбрать имя героя, его пол и как будет выглядеть его образ в дальнейшем. Это необходимо для комфорта самого игрока.

Данная игра в жанре приключения и point-and-click. Лор игры разворачивается через такие игровые механизмы как: диалоговые сцены, поиск предметов, решения логических задач. Благодаря этому игра будет более занимательной, что заинтересует игрока на долгое время.

Для разработки игры проанализированы популярные инди-проекты, которые включают себе такие же жанры и механики. В частности, проанализирована игра «Fran Bow». Данная игра является хоррором, с ужасающей, но незабываемой атмосферой.

Игровой проект разрабатывается на кроссплатформенном игровом движке Unity, что облегчает создание 2D-игры для ПК с разными ОС. Основным языком программирования на этой платформе является C#.

**К. А. Петрович, Н. Б. Осипенко**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА VK MINI-ПРИЛОЖЕНИЯ «МЕЧТЫ» НА ОСНОВЕ JAVASCRIPT И REACT**

В условиях активного развития цифровых технологий и широкого распространения социальных сетей возрастает потребность в удобных онлайн-сервисах, позволяющих пользователям взаимодействовать

друг с другом, делиться мыслями и находить единомышленников. Социальная сеть «ВКонтакте» предоставляет широкие возможности для создания встроенных сервисов, что делает её удобной платформой для разработки интерактивных приложений.

Работа посвящена разработке мини-приложения «Мечты» для платформы VK Mini Apps, предназначенного для публикации и просмотра пользовательских мечтаний, а также взаимодействия между участниками сообщества. Приложение позволяет пользователям делиться своими целями и желаниями, публиковать их анонимно, получать отклики от других людей и находить поддержку.

Приложение реализовано с использованием языка программирования JavaScript и фреймворка React. Выбор данных технологий обусловлен их популярностью, возможностью создания динамического интерфейса и удобством реализации одностраничных веб-приложений.

Использование библиотеки VKUI позволило создать интерфейс, соответствующий дизайн-гайдам «ВКонтакте», а применение библиотеки Redux обеспечило централизованное управление состоянием приложения. Для взаимодействия с серверной частью использовалась библиотека Axios.

Особое внимание уделено архитектуре приложения. Она базируется на концепции Single Page Application с настроенной маршрутизацией. Реализованы ключевые разделы: лента желаний (мечт), форма создания записей и личный профиль пользователя. Предусмотрены функции анонимной публикации, отправки комментариев, взаимодействия с помощниками, а также интеграция с сервисами платформы VK через библиотеку VK Bridge.

В результате был разработан прототип функционального мини-приложения, обеспечивающий публикацию, хранение и просмотр пользовательских записей внутри социальной сети «ВКонтакте». Приложение может использоваться как социальный сервис для общения и поиска поддержки.

В дальнейшем планируется расширение функционала: внедрение системы рекомендаций, развитие механик взаимодействия между пользователями и добавление новых социальных возможностей.

**Д. В. Пигусова, Н. Б. Осипенко**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ 3D-ВИЗУАЛИЗАЦИИ ИНТЕРЬЕРОВ НА ЯЗЫКЕ C#**

В условиях активного развития цифровых технологий и широкого распространения мобильных устройств возрастает потребность в удобных инструментах для работы с трёхмерным контентом. 3D-визуализация активно применяется в архитектуре и дизайне, однако для демонстрации моделей конечным пользователям зачастую требуются простые и доступные средства просмотра без использования профессиональных редакторов.

Работа посвящена описанию реализации мобильного приложения 3DModelViewer для интерактивного просмотра 3D-моделей интерьера, которая закрывает данную потребность. Оно позволяет загружать и интерактивно просматривать 3D-модели на Android. А компонентно-ориентированная архитектура Unity позволяет разделить систему на независимые модули, что обеспечивает гибкость, удобство сопровождения и возможность дальнейшего расширения функционала. Мобильное приложение реализовано на платформе Android с использованием языка программирования C# и игрового движка Unity. Такой выбор был обусловлен стремлением использовать современные и мощные инструменты, упрощающие создание и поддержку приложений, повышающие качество конечного продукта, а также обеспечивающие доступ к широкому спектру библиотек и фреймворков, что является важным фактором для обеспечения долгосрочной поддержки и развития приложения.

Особое внимание уделено реализации мультитач-жестов. Поддерживается вращение, масштабирование и панорамирование объекта с использованием стандартных жестов, что обеспечивает интуитивное и плавное управление.

В результате был создан прототип функционального мобильного приложения, обеспечивающий загрузку, хранение и просмотр 3D-моделей на устройствах Android, который можно использовать в качестве мобильного просмотрщика 3D-контента для демонстрации архитектурных и дизайнерских проектов. В дальнейшем планируется расширение функционала: добавление просмотра списка анимаций модели, возможность запуска выбранной анимации и т.п.

**Н. В. Пикас, Н. Б. Осипенко**  
(ГГУ имени Ф. Скорины, Гомель)

## **ПРОТОТИП ПОЛНОФУНКЦИОНАЛЬНОГО ВЕБ-СЕРВИСА БРОНИРОВАНИЯ ГОСТИНИЧНЫХ УСЛУГ “МУНОТЕЛ”**

Сегодня, планируя собственное путешествие, необходимо продумать его детали, чтобы сделать поездку максимально комфортной и интересной, так всё шире распространяются онлайн-системы бронирования номеров в гостиницах (Booking.com, Airbnb и т. п.), являющиеся неотъемлемой частью туристической инфраструктуры.

Целью описываемого проекта является изучение и применение алгоритмов бронирования номеров; проектирование и разработка системы интернет-бронирования отелей; создание адаптивного, приятного и удобного интерфейса приложения, надежного, безопасного и быстрого сервера, а также использование технологий контейнеризации для кросс-платформенного запуска программного продукта и удобства его развертывания на облачных серверах. Обслуживание клиента начинается с предварительного заказа номера. Для этого клиенты обращаются в программный модуль управления бронированием, работающий в режиме «подтверждение/отказ» с привязкой по времени, функционируя как в рамках единой системы гостиничной сети, так и абсолютно автономно.

Программная реализация проекта базируется на современной клиент-серверной архитектуре с четким разделением ответственности между компонентами. Клиентская часть разработана с использованием библиотеки React, что позволяет создать SPA (Single Page Application) с высокой скоростью отклика, реактивностью и интуитивно понятным интерфейсом. Серверная логика реализована на базе мощного фреймворка Django, который гарантирует надежность обработки запросов, защиту данных пользователей и корректную работу с базой данных. Взаимодействие между клиентской и серверной частями осуществляется посредством REST API.

Для оптимизации работы при высокой загрузке в перспективе запланировано: подключение защищённых платёжных шлюзов (для оплаты напрямую на платформе для повышения безопасности и скорости оформления брони); интеллектуальной системы рекомендаций (для анализа истории поисков и предпочтений); личного кабинета (полноценного CRM-инструмента с разделами для управления номерным фондом, аналитики посещаемости и динамического ценообразования); мультиязычное оформление сервиса.

**П. В. Самулевич**

*(ГрГУ имени Янки Купалы, Гродно)*

## **ОБЕСПЕЧЕНИЕ СЕТЕВОЙ БЕЗОПАСНОСТИ LINUX-ИНФРАСТРУКТУРЫ С ИСПОЛЬЗОВАНИЕМ МЕЖСЕТЕВЫХ ЭКРАНОВ И IPTABLES**

Сетевые инфраструктуры в современных DevOps-проектах требуют надежной защиты от внешних и внутренних угроз. Linux-серверы широко используются благодаря стабильности и гибкости настройки сетевых компонентов. Одним из ключевых инструментов обеспечения безопасности является **iptables** – мощный пакетный фильтр и межсетевой экран (firewall), встроенный в ядро Linux.

**iptables** позволяет настраивать правила фильтрации входящего, исходящего и форвард-трафика, обеспечивая контроль доступа к сервисам и предотвращение несанкционированного взаимодействия. В DevOps-среде правила iptables интегрируются в процессы автоматизации с помощью скриптов, Ansible или Terraform, что обеспечивает повторяемость и масштабируемость конфигурации.

Типичная стратегия защиты включает несколько уровней: ограничение доступа к сервисам по IP и портам, защита от DoS-атак с помощью лимитов пакетов, логирование подозрительных соединений и автоматическое уведомление команды безопасности. Использование **stateful firewall** позволяет отслеживать состояния соединений, что особенно важно для динамических облачных сред, где сервисы и контейнеры создаются и уничтожаются автоматически.

В современных DevOps-пайплайнах критически важно интегрировать управление iptables с системами CI/CD. Это позволяет при развертывании автоматически применять актуальные правила безопасности, минимизируя человеческий фактор и снижая риски ошибок конфигурации.

Таким образом, сочетание межсетевых экранов, гибкой настройки iptables и практик DevOps позволяет создать надежную и масштабируемую инфраструктуру Linux с высоким уровнем сетевой безопасности.

**Е. А. Светловский**  
(ГрГУ имени Янки Купалы, Гродно)

## **ОТ МОНОЛИТА К МИКРОСЕРВИСАМ: ОПЫТ ПЕРЕПИСЫВАНИЯ УЧЕБНОГО TO-DO ЛИСТА НА EVENT-DRIVEN ARCHITECTURE**

Ранние пет-проекты часто строятся как монолиты, что приводит к жесткой связности и сложности расширения. Переход на событийную архитектуру (EDA) решает эту проблему, развивая инженерное мышление. Вместо традиционных брокеров (Kafka) был выбран Redis Pub/Sub с низким порогом входа, что минимизировало инфраструктурные издержки.

В переработанном To-Do приложении API Gateway на Fastify публикует событие “TaskCreated”, после чего асинхронно реагируют независимые сервисы «Уведомления» и «Аналитика». Эта модель хореографии, в отличие от централизованной оркестрации, обеспечивает автономность компонентов и изоляцию сбоев.

Ключевые технические решения включали:

1 Стандартизированный JSON-формат событий с обязательной временной меткой для обработки неупорядоченных сообщений.

2 Реализацию идемпотентности через поле idempotency-key с уникальным ограничением в БД.

3 Внедрение Dead Letter Queue (DLQ) для сохранения сообщений, обработка которых завершилась ошибкой.

Выводы: рефакторинг выявил преимущества EDA: изоляцию сбоев, освоение асинхронных паттернов и понимание децентрализованного управления. Пет-проект готов к переходу на EDA, если:

1 Имеет три и более независимых функциональных домена.

2 Требуется выполнения фоновых задач без блокировки интерфейса.

Опыт трансформирует подход разработчика, смещая акцент с написания кода на проектирование распределенных систем.

### **Литература**

1 Фаулер, М. Микросервисы: паттерны разработки и рефакторинга / М. Фаулер; пер. с англ. А. Киселёва. – СПб.: Питер, 2021. – 352 с.

**Д. В. Стракович**  
(ГГТУ имени П. О. Сухого, Гомель)

## **РОЛЬ САМООБРАЗОВАНИЯ В ИТ-СФЕРЕ**

ИТ-сфера в настоящее время является одной из наиболее динамично развивающихся областей мировой экономики. Информационные технологии охватывают широкий спектр направлений: программирование, веб-разработку, анализ данных, кибербезопасность, системное администрирование, разработку мобильных приложений и искусственный интеллект [1].

Первым шагом при выборе карьеры в ИТ является определение направления деятельности. На начальном этапе важно ознакомиться с основами различных областей: изучить принципы работы компьютера, сети Интернет, алгоритмы и структуры данных. Наиболее доступным стартом считается изучение языков программирования, таких как Python, JavaScript или C#. Эти языки обладают понятным синтаксисом и широкой областью применения.

Следующим этапом является получение практических навыков. Теоретические знания необходимо закреплять выполнением практических заданий, созданием собственных проектов и участием в учебных хакатонах. Начинающим специалистам рекомендуется формировать портфолио, размещая выполненные проекты на специализированных платформах. Важную роль играет самообразование. ИТ-сфера характеризуется быстрым обновлением технологий, поэтому специалист должен постоянно совершенствовать свои знания. Для этого используются онлайн-курсы, профессиональная литература, тематические форумы и сообщества разработчиков. Кроме того, большое значение имеет знание английского языка, поскольку большая часть технической документации публикуется именно на английском языке. Таким образом, первые шаги в ИТ-сфере предполагают осознанный выбор направления, освоение базовых знаний, активную практическую деятельность и постоянное саморазвитие. При систематическом подходе и высокой мотивации начать карьеру в информационных технологиях возможно даже без профильного образования.

### **Литература**

1 Таненбаум, Э. Современные операционные системы / Э. Таненбаум, Х. Бос. – СПб.: Питер, 2020. – 1120 с.

**Е. И. Стракович, Н. Б. Осипенко**  
(ГГУ имени Ф. Скорины, Гомель)

## **РАЗРАБОТКА 2D-ПЛАТФОРМЕРА НА ЯЗЫКЕ C# С ИСПОЛЬЗОВАНИЕМ ФРЕЙМВОРКА UNITY**

В работе рассматривается процесс создания 2D-платформера с использованием языка программирования C# и игрового движка Unity. Актуальность исследования обусловлена необходимостью освоения современных инструментов геймдева и демонстрации полного цикла разработки игрового продукта от концепции до реализации.

Выбор технологического стека обоснован преимуществами связки C# и Unity: компонентно-ориентированная архитектура, обеспечивающая модульность и гибкость системы; наличие встроенных инструментов для работы с физикой, анимациями и графикой; возможность быстрого прототипирования и отладки игровых механик.

Особое внимание уделено реализации ключевых игровых систем: создан отзывчивый контроллер персонажа с системой перемещения, прыжков и обработки коллизий; разработан сложный аниматор с конечным автоматом состояний, обеспечивающий плавные переходы между анимациями; внедрены интерактивные объекты окружения (движущиеся платформы, шипы, собираемые предметы).

В ходе работы были реализованы расширенные игровые механики: тактический рывок с системой расходования ресурса; динамическая прозрачность объектов окружения, решающая проблему видимости персонажа; разрушаемое окружение, открывающее новые пути и секреты; система сбора ресурсов (монеты и сердца), формирующая игровую экономику.

Практическим результатом работы стал полнофункциональный прототип 2D-платформера, демонстрирующий целостный и увлекательный геймплей.

Проект подтверждает эффективность выбранных технологий и может служить основой для дальнейшего развития в направлении добавления новых уровней, врагов и игровых возможностей.

**А. В. Тарканов**

*(ГГТУ имени П. О. Сухого, Гомель)*

## **РАЗРАБОТКА WEB-ПРИЛОЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ БИБЛИОТЕКИ REACT И МОДЕЛЕЙ NLP**

Современные веб-приложения всё чаще включают интеллектуальные компоненты, позволяющие обрабатывать и анализировать текстовые и файловые данные пользователей. Технологии обработки естественного языка NLP обеспечивают возможность интерпретации запросов, извлечения смысловых характеристик, генерации ответов и автоматизации взаимодействия с пользователем, что находит применение в чат-ботах, виртуальных помощниках и сервисах обслуживания [1].

Библиотека React широко применяется для разработки клиентской части веб-приложений, обеспечивая динамическое обновление интерфейса, управление состоянием и удобное отображение элементов. Использование библиотеки совместно с моделями обработки естественного языка позволяет создавать интерактивные веб-системы, способные обрабатывать текстовые и файловые запросы пользователей и обеспечивать адаптивный интерфейс.

Архитектура таких приложений обычно основана на разделении клиентской и серверной логики. Первая отвечает за визуальное взаимодействие и передачу данных, а серверная система выполняет обработку текстовых сообщений пользователей, анализ смыслового содержания и формирование ответов. В процессе работы с текстовыми данными могут применяться этапы предварительной обработки, включающие нормализацию текста и удаление служебных символов. Данная архитектура улучшает масштабируемость, расширяемость и позволяет интегрировать дополнительные сервисы.

Применение данных технологий обеспечивает создание адаптивных, масштабируемых web-приложений, которые улучшают взаимодействие с пользователем и расширяют возможности интеллектуальных систем во многих предметных областях.

### **Литература**

1 Hugging Face: Introduction to Natural Language Processing: [сайт]. – URL: <https://huggingface.co/learn/nlp-course> (дата обращения: 20.02.2025).

**С. А. Храменков, С. В. Киргинцева**  
(ГГУ имени Ф. Скорины, Гомель)

**РАЗРАБОТКА WEB-САЙТА  
«МАГАЗИН ЭЛЕКТРОННЫХ ГАДЖЕТОВ И АКСЕССУАРОВ»  
С ИСПОЛЬЗОВАНИЕМ HTML, CSS**

В современном обществе веб-технологии играют ключевую роль в сфере торговли и предоставления услуг. Интернет-магазины становятся одним из наиболее востребованных способов приобретения товаров, так как позволяют пользователям получать доступ к широкому ассортименту продукции в любое время и из любого места. Особенно актуальным является создание веб-сайтов, ориентированных на продажу электронных гаджетов и аксессуаров, поскольку данная категория товаров пользуется стабильным спросом у различных групп пользователей.

Разработан веб-сайт «Магазин электронных гаджетов и аксессуаров» с применением языков HTML и CSS, обеспечивающий наглядное представление товаров и удобство взаимодействия пользователя с сайтом [1–3].

Структура веб-сайта включает главную страницу, страницы каталога товаров, отдельные страницы с описанием электронных гаджетов и аксессуаров, а также элементы навигации и пользовательского интерфейса. Особое внимание было уделено удобству использования и визуальному оформлению. Практическая значимость работы заключается в возможности использования разработанного веб-сайта в учебных и практических целях, а также в качестве основы для создания полноценного интернет-магазина.

**Литература**

1 Дакетт, Джон. HTML и CSS. Разработка и дизайн веб-сайтов / Джон Дакетт. – М.: Эксмо, 2013. – 480 с.

2 Лазаро, Исси. Коэн Полный справочник по HTML, CSS и JavaScript / Лазаро Исси Коэн, Джозеф Исси Коэн. – М.: ЭКОМ Паблицерз, 2014. – 938 с.

3 Мержевич, В. HTML и CSS на примерах / В. Мержевич. – М.: «БХВ-Петербург», 2012. – 448 с.

**А. В. Шавель**

*(ГрГУ имени Янки Купалы, Гродно)*

## **КАТАЛОГ РАСТЕНИЙ С НАПОМИНАНИЯМИ О ПОЛИВЕ**

Приложение «Каталог растений с напоминаниями о поливе» представляет собой высокотехнологичное программное решение, спроектированное для комплексной автоматизации учета домашних ботанических коллекций и минимизирования рисков, обусловленных человеческим фактором.

Технический фундамент приложения построен на платформе .NET и языке C#. Такой выбор гарантирует высокую скорость работы и защиту от системных ошибок, что делает программу надежной при длительном использовании. Визуальная часть создана с помощью фреймворка Avalonia UI. Это современное решение обеспечивает приложению привычный для пользователя Windows вид и высокую скорость отклика интерфейса.

Система хранения реализована локально для упрощения развёртывания. Данные сериализуются в формат JSON с помощью System.Text.Json, что обеспечивает автономность приложения и портативность базы данных.

Архитектурный каркас приложения спроектирован в соответствии с принципами Clean Architecture, что выражается в жестком разделении ответственности между компонентами. Ядром системы выступает клиентское приложение Water\_Buddy, функционирующее на базе архитектурного паттерна MVVM для достижения высокой степени декомпозиции, что упрощает модульное тестирование и модернизацию программного кода. Слой DataLayer берет на себя все задачи по управлению информацией, гарантируя её сохранность и правильное отображение при записи в файл или чтении из него. Параллельно функционирует автономная служба NotificationManager, которая интегрируется с API операционной системы Windows для генерации системных push-уведомлений. Служба работает в асинхронном режиме, постоянно сопоставляя текущие временные метки с заданными графиками ухода.

Ключевой объект предметной области – класс **Plant**, моделирующий реальное растение в цифровом виде. Он содержит статические атрибуты (морфологические признаки, изображение) и

динамические показатели (регулярность полива, отметка времени последнего ухода). Логика приложения реагирует на изменение объекта: после каждого обновления данных автоматически пересчитывается график обслуживания, что сразу же отражается в плане работ планировщика. Итогом стала производительная и отзывчивая система. Её модульная основа позволяет не только эффективно работать сейчас, но и легко расширяться в будущем за счёт умных функций вроде нейросетевой диагностики или экспорта данных.

**Q. W. Dai**

*(Yanka Kupala State University of Grodno, Belarus)*

## **EDUCATIONAL ADMINISTRATION SYSTEM**

Представляемый проект представляет собой веб-сайт образовательной организации, разработанный на современных веб-технологиях и предназначенный для автоматизации и оптимизации управления академическими процессами в ВУЗах.

The developing system features a web-based interface accessible through standard browsers, eliminating the need for additional software installation. Its architecture consists of a presentation layer, an application logic layer, and a data layer built on a relational database management system, ensuring reliability, maintainability, and performance.

The platform includes several key functional modules. The student management module supports registration, editing, transfer, and expulsion operations. The curriculum management module enables administrators to define educational programs, disciplines, semesters, and types of assessment. The scheduling module allows the creation and modification of timetables with automatic conflict detection for teachers, groups, and classrooms. The grade and attendance tracking module provides electronic grade books, supports multiple grading scales, and automatically calculates final results.

The system implements role-based access control to ensure data security and confidentiality. Administrators, teachers, and students have differentiated permissions according to their responsibilities. Additional security measures include secure authentication, protection against common web vulnerabilities, and support for encrypted communication protocols.

Compared to traditional administrative approaches and existing complex systems, the developed solution offers several advantages. At the current moment the user interface has done. Next steps for the project are significantly improves operational efficiency by automating routine processes and reducing manual workload, then it enhances data accuracy and consistency through centralized database management and structured data validation, and finally, it provides analytical and reporting tools that support informed decision-making by university management.

By addressing the limitations of outdated administrative methods and overloaded commercial systems, this project aims to create a practical, efficient, and scalable educational administration platform. The developing application can serve as a prototype for real university information systems and can be further expanded with additional modules, including integration with e-learning platforms, mobile applications, and advanced analytical dashboards.

**Y. T. Jia**

*(Yanka Kupala State University of Grodno, Belarus)*

## **GLOBAL EARTHQUAKE MONITORING SYSTEME**

Описываемый проект представляет собой веб-приложение для мониторинга землетрясений в режиме реального времени по всему миру, созданное на основе современных веб-технологий и обеспечивающее интерактивную удобную платформу для визуализации сейсмической активности.

Existing earthquake information platforms often provide comprehensive data but suffer from complex interfaces and information overload, making it difficult for ordinary users to quickly locate relevant seismic events. While some third-party applications offer simplified views, they frequently include advertisements or lack seamless multi-device compatibility, forcing users to sacrifice either convenience or functionality. As demand for immediate and spatially aware information grows, users expect an intuitive, cross-device platform combining authoritative data with visual experience.

Some modern applications have introduced basic map features, yet they often lack real-time background updates, multi-language support, or

efficient backend systems, leading to slow content delivery and poor scalability during high seismic activity. This results in delayed information and reduced user trust.

The developed system addresses these limitations using ASP.NET Core MVC with C#, Bootstrap for responsive design, and Leaflet.js for interactive maps. It fetches real-time data from the USGS API, presenting earthquakes on a global map and sortable list with magnitude-coded markers. Users can click events to view detailed information including magnitude, location, coordinates, depth, and time. The system features automatic updates at configurable intervals, manual refresh, and bilingual interface support for Chinese and English speakers.

The new system offers faster content delivery through efficient background data fetching and caching, ensuring users receive the latest seismic information in real-time without manual reloads. Intuitive map visualization with magnitude-coded markers enhances spatial understanding of earthquake distribution worldwide. The clean, responsive interface enables seamless access across desktop, tablet, and mobile devices while maintaining focus on core monitoring tasks.

By addressing shortcomings of existing solutions, this project creates an interactive and efficient earthquake monitoring platform meeting needs of both general users and researchers in the modern digital age.

**Sh. T. Li**

*(Yanka Kupala State University of Grodno, Belarus)*

## **PRODUCT PORTAL API SYSTEM**

Представленный в статье проект представляет собой веб-приложение как «сайт-визитку» предприятия, реализующего свою продукцию, кроме того, запрограммированы функции его администрирования.

The project presents a full-stack web application for enterprise product management, built on a modern technical architecture that combines C# ASP.NET Core 8 for the backend API and Vue.js for the frontend interface. The system is designed to provide organized storage, access, and editing of products, promotion activities, career information, and file uploads through secure RESTful APIs. The proposed system

focused vertical application—specifically tailored for product portals. The application follows a layered architecture to ensure separation of concerns and maintainability. At the core are entity models for Product, Promotion, Career, and AdminUser, which are persisted in a MySQL database using Entity Framework Core. The service layer encapsulates business logic through dedicated classes (ProductService, PromotionService, CareerService, AuthService, FileService, and TokenBlacklistService), while the controller layer exposes RESTful endpoints. Data transfer objects guarantee clean communication between the backend and the Vue.js frontend. JWT authentication secures administrative operations, and a token blacklist mechanism enables secure logout. File uploads are managed by a dedicated file service, supporting image storage and retrieval for product and promotion visuals. The system implements comprehensive functionality: full CRUD operations for all entities, product categorization, carousel display management, live search and filtering, sorting by various criteria, promotion activity tracking with date-based status, and career opportunity postings with detailed requirements. Public API endpoints deliver product listings, search results, promotions, and career information to the frontend, while protected admin endpoints allow content managers to add, edit, or delete items. The Vue.js frontend provides a responsive user interface with pages for the product catalog, promotions, careers, admin dashboard, and login, all styled for seamless multi-device access. Docker containerization ensures consistent deployment across environments and simplifying scaling. By combining a robust .NET Core API with a dynamic Vue.js interface, the project delivers a modern product portal that meets the demands of today’s digital enterprises, offering faster content updates, personalized user experiences, and streamlined management capabilities.

**D. Y. Sun**

*(Yanka Kupala State University of Grodno, Belarus)*

## **BOOK MANAGEMENT SYSTEM**

В этом проекте представлена система управления книгами для локального пользования. Интерфейс разработан на языке C#, для хранения данных используется SQL Server. Цель системы – упростить рабочие процессы в библиотеке и повысить эффективность управления книжным фондом.

Traditional book management processes often rely on manual record-keeping or outdated systems, leading to cumbersome operations for adding, editing, and deleting books, slow data retrieval, and a lack of real-time visibility into inventory levels and collection statistics. As libraries and bookstores seek to optimize operational efficiency, there is a growing demand for an intuitive and user-friendly system that can streamline daily tasks while providing accurate, real-time information.

While some existing systems offer basic book tracking capabilities, they generally lack smooth CRUD operations, efficient search functionality, and clear visualization of collection data. Furthermore, many legacy systems feature clunky user interfaces, making it difficult for administrators to quickly complete tasks such as updating book details or monitoring inventory levels, which can lead to errors and inefficiencies in daily operations. The system features a user-friendly desktop interface, allowing authorized administrators to log in securely and perform core book management tasks with minimal effort. The front-end, presents detailed book information through a clear data table, including id, title, category, price, stock, and so on. The back-end relational database efficiently stores and manages all book data, ensuring data integrity and supporting fast retrieval of records.

The new book management system offers several key advantages over traditional approaches. Firstly, streamlined data operations and real-time updates. Secondly, efficient and precise book retrieval. Finally, enhanced inventory visibility and data-driven decision support.

This project provides a practical and user-friendly solution that meets the operational needs of modern libraries, bookstores, and other organizations managing book collections.

**J. W. Wang**

*(Yanka Kupala State University of Grodno, Belarus)*

## **SHOPPING EXPENSE TRACKER APP**

Рассматриваемое приложение представляет собой систему учета покупок, разработанную на языке программирования C# на платформе Console Window.

Since the dawn of humanity, people have been striving to acquire new knowledge and make their lives more comfortable. More and more

goods have entered people's lives to meet various human needs. Faced with the dazzling array of goods, humans have developed a stronger desire to purchase. Due to the excessive promotion by some merchants and the influence of online advertisements, many people made impulsive purchases and placed passionate orders, buying some goods that seemed useful but were actually useless.

The purpose of developing this program is to enable people to keep track of their shopping expenses at any time, to correctly understand their shopping desires, to restrain impulsive consumption, to manage their living expenses reasonably, and to avoid impulsive consumption and waste of money.

The application is a shopping information system, developed using the C# programming language and based on the Console window platform. This application is designed to provide users with the function of querying relevant information for shopping information statistics services. It supports users in performing operations such as deleting, editing and adding information.

The newly launched shopping record program has several advantages over the existing related software in the app store. Firstly, this program is free of charge and does not require viewing advertisements to use. Secondly, the operation is simple and only a laptop is needed. Users can modify the data in the TXT text file to obtain the desired results, which significantly improves the statistical efficiency. Finally, the program can save data, and users can compare the weekly or monthly expenditures to make reasonable allocations.

Compared to the shopping record software available in the software store, this program is dedicated to creating a simpler, more user-friendly program that is accessible to everyone, in order to meet the daily need of recording expenses at any time.

**Y. Wang**

*(Yanka Kupala State University of Grodno, Belarus)*

## **REPAIR SYSTEM DEVELOPMENT**

В статье описана система, оптимизирующая процесс подачи, управления и отслеживания заявок на ремонт бытовой техники, разработанная на языке программирования C#.

The web application enables users to submit repair orders and administrators to manage them through the close interaction between C# and the ASP.NET Core framework. At the same time, it combines HTML, CSS, Bootstrap and Razor Pages to build a complete and user-friendly front-end interface for both customers and administrators. The system implements core management functions such as viewing all orders, deleting orders, changing administrator passwords, and exporting order data.

Order submission module: a repair request form was developed using Razor Pages, through which users can easily enter their name, phone number, address, problem description, and mark the order as urgent. Form validation is used to ensure the legitimacy of input data. After submission, the data is serialized and stored in a JSON file using service-layer logic implemented in C#. Order management module provides administrators with tools to view, filter, and delete repair orders. Administrators can log in securely, view all orders in a dashboard, and perform operations such as deletion and password changes. C# controllers are used to query the list of orders, display order information on the page, and handle deletion requests. JSON is used as the storage format to ensure simplicity and maintainability. Order tracking module allows users to modify or cancel existing orders using their order ID, and receive clear feedback on the success or failure of their actions. ASP.NET Core MVC pattern is used to separate concerns and improve code maintainability.

However, from the perspective of practical application and future expansion, the system can be further improved, such as optimizing the aesthetics and interaction details of the interface, enhancing security and data validation mechanisms, and adding more practical management functions (such as batch operations, order prioritization, or status tracking) to better meet the diverse and complex needs of real-world repair service management.

**P. Xiang**

*(Yanka Kupala State University of Grodno, Belarus)*

**DESIGN AND IMPLEMENTATION OF A DORMITORY  
MANAGEMENT SYSTEM BASED ON C# AND WINDOWS  
FORMS**

В статье описывается разработанное на C# Windows Forms система учета студентов, проживающих в общежитии университета.

This thesis aims to design and implement a dormitory management system based on c# and windows forms to address inefficiencies and data update challenges in university dormitory management. by analyzing existing methods—such as paper records, excel spreadsheets, commercial software like starrez, and open-source solutions—the thesis identifies core required functions: student information management, multi-condition search and sorting, data export, and user authentication.

The system adopts a layered architecture, including the model layer (defining student entities and gender enumeration), service layer (data service class handling json read/write, search, and sorting via newtonsoft.json), presentation layer (login, main, and edit interfaces), and utility layer (validator class for email, phone, and id format validation). Json is used for lightweight, database-independent data storage. Functionally, the system supports administrator login (admin/admin123), complete CRUD operations, real-time multi-field search and sorting, and export to csv/txt files.

Input validation ensures data integrity. The interface is intuitive, with a main window featuring a search box, data table, detail panel, and operation buttons, while edit windows highlight required fields and provide real-time error feedback. Modular design facilitates maintenance and future expansion. Testing confirms stable performance in windows environments, handling hundreds of records efficiently.

This system effectively improves dormitory management accuracy and efficiency, reduces manual errors, and provides a foundation for future enhancements like multi-user permissions, data visualization, and automated reporting, making it suitable for small to medium-sized institutions as a digital management solution.

**H. Zhou**

*(Yanka Kupala State University of Grodno, Belarus)*

## **NCAA BASKETBALL DATA ANALYSIS SYSTEM**

Описываемая разработка представляет собой систему анализа данных для хронологических записей чемпионата NCAA по баскетболу среди мужчин с использованием платформы .NET. Она оптимизирует загрузку данных, их анализ, запросы и логирование для исследований спортивной статистики и обучения программированию.

This project develops a data analysis system for NCAA Men's Basketball Championship historical records using the .NET framework, optimizing data loading, parsing, querying, and logging for sports statistics research and programming education.

Implemented in C#, the system uses tab-separated text files for match records and command scripts. StreamWriter enables synchronized console and log output, a custom insertion sort arranges records chronologically, and relative paths ensure cross-platform flexibility. The data loading module reads records via File.ReadAllLines(), extracts year, teams, and scores, identifies champions and runners-up through score comparison, validates field integrity, and stores normalized data in a List<NCAAGame> collection. The sorting module applies insertion sort to arrange records in ascending year order, optimized for NCAA's historical dataset spanning 1939 to present.

The command processor supports three query types: "year" retrieves champion or runner-up for a specified year (type 1 or 2); "titles" counts cumulative championships and runner-up finishes for a team; "margin" calculates average point differential over a user-defined consecutive year range.

The logging module generates standardized headers with programmer identification. A centralized WriteOutput() method ensures synchronized output, zero-padded command numbers enhance readability, and visual separators structure the log. Interactive selection among three data file groups (A/B/C) includes automatic fallback for invalid input.

Potential improvements include replacing insertion sort with List<T>.Sort() for better performance, adding exception handling for I/O operations, implementing command syntax validation, extending analytics with features like streak detection, using configuration files to eliminate hardcoded paths, and introducing an interactive menu to improve usability.

## СОДЕРЖАНИЕ

### СОВРЕМЕННЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

#### *Системное и программное обеспечение информационных технологий*

<i>Абрамов А. Д., Кузьменков Д. С.</i> Разработка веб-сайта для back-end разработчиков “Road Map” с использованием Django.	3
<i>Аксамит Н. В.</i> Анализ применимости технологий workflow- автоматизации в системе анализа медицинских изображений.....	4
<i>Андросенко К. М., Кузьменкова Е. Ю.</i> Разработка сайта-визитки для СПА-салона.....	5
<i>Андрусенко К. В., Ружицкая Е. А.</i> Разработка Front-end части обучающей платформы с интеграцией функций искусственного интеллекта.....	6
<i>Базар Т. А.</i> О разработке мобильного приложения для кинотеатра	7
<i>Балабан В. В., Кузьменков Д. С.</i> Разработка визуальной части игрового приложения “Sky Flapper” в среде Unity.....	8
<i>Барабанов А. П., Березовская Е. М.</i> Создание чат-бота для управления сервером Discord с использованием языка программирования Python и API Discord.....	9
<i>Бельский М. Б., Кузьменков Д. С.</i> Разработка геймплея игрового приложения 2D-платформера в среде Unity.....	10
<i>Березин Г. А.</i> Разработка мобильного приложения-путеводителя по Республике Беларусь на платформе Unity.....	11
<i>Бондарева А. В.</i> Разработка Desktop-приложения для учёта личных финансов с использованием C# и SQLite.....	12
<i>Боярин Ф. С.</i> Реализация работы многофункционального мессенджера в режиме оффлайн.....	13
<i>Боярин Ф. С., Дудко Г. С.</i> Сравнительный анализ функционала веб-версии и мобильной версии мессенджера Telegram.....	15
<i>Булкина А. М., Соболев И. А.</i> Применение ASP.NET Core для разработки интернет-магазина товаров для домашних животных ...	16
<i>Бурикин А. С., Процкая Т. В.</i> Создание онлайн-магазина виниловых пластинок с использованием Spring Framework.....	17

<i>Ваденкова А. В., Кузьменкова Е. Ю.</i> Разработка хранилища данных для обработки Big Data с использованием DWH, Apache Spark и Airflow.....	18
<i>Василевский М. И., Стефановский И. Л.</i> Разработка информационной системы по планированию и автоматизации лабораторных и технических испытаний.....	19
<i>Вегера Е. П.</i> Разработка web-приложения для автоматизации материально-технического учета на кафедре с модулем подключения к 1С.....	20
<i>Вожейко А. В.</i> Нейросетевая классификация поведенческих паттернов для защиты от автоматизированных скриптов.....	21
<i>Войновская В. Э.</i> Автоматизированное тестирование безопасности веб-приложений для выявления уязвимостей.....	23
<i>Гаврик Д. Н.</i> Практическая оптимизация и ускорение инференса диффузионных моделей.....	24
<i>Гаврилова Д. А.</i> Оценка эффективности работы распределённых систем .....	25
<i>Гагалушко С. В.</i> Анализ уязвимостей и проектирование безопасной системы аутентификации для корпоративных сетей на основе сетевого протокола аутентификации Kerberos.....	26
<i>Галькевич В. В.</i> Разработка отказоустойчивого однорангового In-Memory хранилища данных.....	27
<i>Гапонова К. С., Ружицкая Е. А.</i> Разработка серверной части и структуры реляционной базы данных web-сайта ОДО «Амелия».	28
<i>Гарацкевич П. С.</i> Проектирование логики сражения в компьютерной игре .....	29
<i>Герасименко А. А., Карасёва Г. Л.</i> Разработка мобильной игры в жанре Sci-fi шутера в среде разработки Unity на C# .....	30
<i>Гребёнкин М. Д., Кузьменков Д. С.</i> Разработка front-end части приложения для создания и поиска мероприятий по интересам с использованием React, TypeScript .....	31
<i>Гришко А. А., Березовская Е. М.</i> Разработка онлайн-магазина с интеллектуальной системой рекомендаций товаров на основе машинного обучения.....	32
<i>Гришкова В. В.</i> Разработка интернет-магазина оборудования и бытовой техники «Robust» с использованием PHP .....	33
<i>Громыко Н. С., Лубочкин А. В.</i> Настольное приложение для автоматизации складского и бухгалтерского учёта кофейного бизнеса с использованием Python и PostgreSQL .....	34

<i>Гузов В. Д., Киргинцева С. В.</i> Разработка веб-приложения для банка “АBank” с использованием JavaScript .....	35
<i>Гулюк Н. С., Кузьменков Д. С.</i> Разработка инженерного калькулятора на языке программирования Java .....	36
<i>Гурбо Н. Д.</i> Применение модульно-монолитной и луковой архитектуры в системе обработки пользовательских обращений..	38
<i>Гусаков Д. А., Кузьменков Д. С.</i> Разработка front-end части веб-приложения «Маркетплейс» с использованием HTML5, CSS3, JavaScript, Figma .....	39
<i>Дашкевич Д. В., Аксёнова Н. А.</i> Разработка приложения веб-приложения «Персональный органайзер» .....	40
<i>Дебой А. Д.</i> Разработка мобильного приложения для скрининга фибрилляции предсердий .....	41
<i>Дедова П. В.</i> Разработка мобильного приложения для кафе «Поляна Сити».....	42
<i>Дидюля А. В.</i> Проектирование программного обеспечения для работы с графическим контентом при помощи нейросетевых моделей.....	43
<i>Дмитрук С. К.</i> Проектирование и реализация системы управления локальными базами данных в мобильном приложении-каталоге .....	44
<i>Довидович З. И.</i> Разработка безопасной системы мониторинга IT-инфраструктуры на основе Zabbix с интеграцией HashiCorp Vault .....	45
<i>Донец А. В., Березовская Е. М.</i> Разработка системы «1С: Логист» для управления автомобильными перевозками и сервера взаимодействия для расчета и оптимизации маршрутов.	46
<i>Дудинский А. В., Березовская Е. М.</i> Информационное Android-приложение “GSUAPP” на языке программирования Kotlin.....	47
<i>Екимова М. Д.</i> Автономное красное тестирование систем ИИ: применение LLM для симуляции атак и оценки защитных механизмов .....	48
<i>Ермак Д. С.</i> Проектирование расчетно-справочного модуля для приложения «Помощник врача-неонатолога».....	50
<i>Захаров А. В., Березовская Е. М.</i> Разработка интерактивного web-приложения «Ресторанный портал» с использованием React, Node.js и PostgreSQL.....	51

<i>Зеленков И. А., Ружицкая Е. А.</i> Разработка высокопроизводительного API-сервиса прогнозирования погоды с интеграцией нейросетевых моделей.....	52
<i>Зеленский К. А., Титова Л. К.</i> Разработка веб-приложения «Автоматизация технического обслуживания промышленного оборудования» с использованием ASP.NET Core, React Vite, PostgreSQL .....	53
<i>Землянский Е. Г., Соболев И. А.</i> Разработка мобильного приложения для формирования и трекинга персональных привычек на языке программирования Kotlin.....	54
<i>Зинович В. А.</i> Разработка функциональной основы веб-приложения «Онлайн-кинотеатр» на основе RESTful архитектуры и технологии контейнеризации Docker.....	55
<i>Зубрик Н. Д.</i> Применение систем предотвращения утечек данных для мониторинга и фильтрации сетевого трафика.....	56
<i>Иванова Е. В., Бычков П. В.</i> Разработка веб-сервис по доставке товара потребителю .....	57
<i>Исаченко А. С., Кузьменков Д. С.</i> Разработка веб-приложения для мониторинга и визуализации космического мусора с использованием Python, FastAPI, React .....	58
<i>Кавцевич П. П.</i> Проектирование модульной архитектуры локального программного комплекса для семантической обработки текстов .....	59
<i>Казека В. В., Ружицкая Е. А.</i> Клиентская часть web-приложения «Интернет-магазин музыкальных инструментов» .....	60
<i>Калинкова Е. О.</i> Моделирование и оценка защищённости сегментированной сетевой инфраструктуры систем видеонаблюдения на основе программных эмуляторов .....	62
<i>Каравацкий Г. Г.</i> Качественная оптимизация агентных систем с использованием технологии селф-промптинга.....	63
<i>Карасёва Г. Л., Тагангылыджова З. Р.</i> Создание приложения «Достопримечательности Республики Беларусь» в среде C++ Builder с использованием БД Access.....	64
<i>Караханова К. С., Кузьменкова Е. Ю.</i> Разработка веб-сайта социально-благотворительного учреждения с использованием JavaScript, React.....	65
<i>Карпович Е. Ю.</i> Разработка системы автоматизации управления складскими запасами .....	66

<i>Качура В. Д., Березовская Е. М.</i> Разработка приложений “Virtual piano” и “Musical calculator” с использованием интегрированной среды разработки Delphi .....	67
<i>Киптик Е. В., Кузьменков Д. С.</i> Разработка клиентской части веб-приложения “DigiHelper” с использованием фреймворка Blazor .....	68
<i>Киселев Н. Л.</i> Использование нейронных сетей при управлении автономным транспортным средством .....	69
<i>Киселёв И. К.</i> Методы и алгоритмы персонализации контента в электронной коммерции для роста ключевых бизнес-показателей .....	71
<i>Климов А. А.</i> Разработка 2D-шутера с элементами искусственного интеллекта на основе фреймворка MonoGame .....	72
<i>Коваленко В. С., Кузьменкова Е. Ю.</i> Разработка книжного интернет-магазина “Liberty” с использованием React, Node.js .....	73
<i>Кожмякин К. С., Ленденкова С. И.</i> Разработка плагина для IntelliJ IDEA .....	74
<i>Козликовская В. В., Березовская Е. М.</i> Подключение сервиса Mailjet для отправки кода подтверждения при регистрации в мобильном приложении “For All Into Life” .....	75
<i>Козловская Е. В., Кузьменков Д. С.</i> Разработка приложения для управления и хранения паролей с использованием C#, UWP....	76
<i>Коледа А. Р.</i> Генерация ресурсов для 2D-игр с применением условных генеративно-состязательных сетей.....	77
<i>Колеснева И. А., Ружицкая Е. А.</i> Клиентская часть блога «Моя кулинарная лаборатория» .....	78
<i>Кондратьев Н. И., Кузьменков Д. С.</i> Разработка автоматизированной платформы для продажи и проведения обучающих курсов с использованием искусственного интеллекта.....	79
<i>Кононенко Д. О., Короткевич В. А.</i> Приложение для визуализации данных о подаче заявлений в приемную комиссию вуза..	80
<i>Конопко Л. Л.</i> Генерация тест-кейсов для функционального тестирования программного обеспечения.....	81
<i>Копоть Д. А.</i> О разработке хостинг-платформы.....	82
<i>Король Д. А.</i> Разработка защищенного веб-сервиса для поиска сведений об организациях по УНП.....	83
<i>Кузьменков В. Д., Кузьменков Д. С.</i> Разработка веб-приложения «Картотека фильмов» с использованием HTML5, CSS3, JavaScript.	84

<i>Лашкин М. А., Кузьменков Д. С.</i> Разработка back-end части обучающей платформы с интеграцией функций искусственного интеллекта .....	85
<i>Лебедев Д. С., Войтешенко И. С.</i> Оптимизация постоянных соединений в service mesh.....	87
<i>Лебедев И. А., Демуськов А. Б.</i> Разработка Android-приложения “EasyGo” для определения маршрута на языке программирования Kotlin .....	88
<i>Лисай А. М., Поть Т. А.</i> Нейросетевые модели для диагностики гемангиом печени на МРТ-изображениях .....	89
<i>Лисичкин В. А., Короткевич В. А.</i> Разработка серверной части приложения для поиска и организации мероприятий.....	90
<i>Маркелов А. Д., Ружицкая Е. А.</i> Разработка модуля аутентификации и авторизации для web-приложения “DigiHelper” .....	91
<i>Марочкин В. Е.</i> Разработка сетевой социальной игры «Убежище» с интегрированной языковой моделью .....	92
<i>Марочкин В. Е.</i> Применение языковых моделей большого объёма в прикладных программных системах.....	93
<i>Марушкевич И. В.</i> Применение технологий .NET для создания инструмента обработки голосовых данных.....	94
<i>Маслак Е. И., Комраков В. В.</i> Технология RAG в информационно-аналитических системах для интеллектуального анализа. ....	95
<i>Матеюк Е. Г.</i> Влияние обязательной периодической смены паролей на стойкость аутентификации.....	96
<i>Мезенцев А. В., Процкая Т. В.</i> Разработка интернет-магазина канцелярских товаров с использованием фреймворка ASP.NET Core MVC .....	97
<i>Мельник И. Г., Титова Л. К.</i> Особенности автоматизации инспекции кода средствами нейросетей .....	98
<i>Мельник И. Г., Титова Л. К.</i> Разработка интерактивного модуля рецензирования программного текста на базе генеративных моделей.....	100
<i>Михалев Н. С.</i> Применение алгоритмов машинного обучения для прогнозирования сбоев в ИТ-инфраструктуре.....	101
<i>Мишур В. П., Атвиновский А. А.</i> Применение библиотеки React для разработки web-приложения «Стоматологическая клиника» .....	102
<i>Мошкович Д. А.</i> Разработка веб-приложения для автоматизированного анализа банковских транзакций на основе выписок..	103

<i>Мячина С. А.</i> Количественная оценка устойчивости методов браузерного фингерпринтинга к средствам защиты приватности.	104
<i>Нагорнюк Н. В., Атвиновский А. А.</i> Применение веб-технологий для создания интерактивных тестирующих систем по математическому анализу .....	105
<i>Нагорнюк Н. В., Кузьменков Д. С.</i> Стратегия миграции веб-приложения «Библиотека» от монолитной к микросервисной архитектуре .....	106
<i>Немкевич М. С.</i> Система отслеживания ценовой динамики товаров на многопользовательских торговых онлайн-платформах.	107
<i>Нечай Д. С., Кузьменкова Е. Ю.</i> Разработка серверной части приложения с использованием ASP.NET Core .....	108
<i>Новоженцев П. Е.</i> Современные инструменты визуализации данных.....	109
<i>Параев Д. В., Лубочкин А. В.</i> Разработка 2D-игры в жанре RPG с использованием языка С# и среды разработки Unity .....	110
<i>Парфеевец И. А., Карагодин Д. Л.</i> Модификация кода классического шаблона проекта Win API средствами С# .....	111
<i>Пацков А. А., Ружицкая Е. А.</i> Многоязычность информационной системы «Онлайн-табло аэропорта» .....	112
<i>Переплетчикова В. А., Березовская Е. М.</i> Проектирование и разработка книжного интернет-магазина “BookStore” .....	113
<i>Пивунов Д. И.</i> Разработка модуля маркетплейса для организации торговли в формате адвент-календаря .....	115
<i>Понедькова А. В., Карасёва Г. Л.</i> Разработка приложения для поиска аптек и лекарственных средств в городах Республики Беларусь .....	116
<i>Поть Т. А., Лисай А. М.</i> Нейросетевые модели для сегментации МРТ-изображений в задачах быстрой диагностики заболеваний..	117
<i>Приходько В. Д., Карасёва Г. Л.</i> Математическая модель адаптивного управления вычислительными ресурсами .....	118
<i>Пыршина Т. В., Ружицкая Е. А.</i> Разработка пользовательского интерфейса и клиентской логики интернет-магазина «Цветочный магазин» .....	119
<i>Радюк В. А., Ружицкая Е. А.</i> Разработка клиентской части интернет-магазина “AutoMarket” .....	120
<i>Рогальский И. А.</i> Многоуровневая защита сессий и JWT-токенов с учётом ролей .....	121

<i>Романовский А. А.</i> Обнаружение аномальных последовательностей действий в логах облачных сред (AWS CloudTrail) с помощью рекуррентных нейронных сетей .....	122
<i>Руть К. С.</i> Разработка веб-приложения для автоматизации документооборота в организации .....	123
<i>Савинский И. И.</i> Интеллектуальные методы персонализации контента в системах онлайн-кинотеатров .....	124
<i>Савинский А. С., Боярин Ф. С.</i> Применение хранилища для организации файлового обмена в многофункциональном мессенджере ...	125
<i>Сазоненко Е. В., Орлов В. В.</i> Создание интернет-магазина по продаже автомобилей .....	126
<i>Санец А. В., Ружицкая Е. А.</i> Разработка серверной части интернет-магазина «Компьютеры и комплектующие» на языке Python с помощью фреймворка FastAPI .....	127
<i>Сердюков А. В., Кузьменков Д. С.</i> Разработка вспомогательного приложения для управления кадровыми данными в составе веб-системы «Библиотека» с использованием Java .....	129
<i>Соловей Е. С.</i> Разработка защищенного клиентского интерфейса системы веб-тестирования с применением строгой типизации и инструментов контроля версий GitLab .....	130
<i>Сорока К. В.</i> Применение гибридных архитектур при проектировании рекомендательных систем для маркетплейсов различного масштаба .....	131
<i>Сороколетов И. А., Кузьменкова Е. Ю.</i> Разработка игрового приложения “Defence” с использованием Unity .....	132
<i>Станулевич Е. А., Левчук Е. А.</i> Разработка мобильного приложения по поиску вакансий .....	133
<i>Степанчиков С. Д., Ружицкая Е. А.</i> Фреймворк FastAPI как эффективный инструмент для создания Rest-сервисов .....	134
<i>Стецкий Д. Ч.</i> Обеспечение безопасности GraphQL API: уязвимости и решения .....	135
<i>Сухаревич Д. С.</i> Построение каркасов приложений с учетом иерархии окон интерфейса .....	136
<i>Тарканов А. В.</i> Интеллектуальный веб-ассистент с использованием нейросетевых моделей обработки естественного языка...	137
<i>Терещенко К. М.</i> Разработка АРМ диспетчера по формированию пассажирских поездов .....	139

<i>Тимощенко И. Ю., Аксёнова Н. А.</i> Разработка системы мониторинга сетевых устройств на языке С .....	140
<i>Тонко И. Н.</i> Разработка игровой системы «Кто хочет стать миллионером?» .....	141
<i>Федотова Д. В., Карасёва Г. Л.</i> Разработка веб-приложения для публикации видеоконтента .....	142
<i>Филон И. Ю.</i> Разработка интерактивной платформы для мониторинга и анализа спортивных мероприятий .....	143
<i>Фомин К. В., Карасёва Г. Л.</i> Использование современных информационных технологий в области маркетинга для продвижения мебельной продукции .....	144
<i>Храпунова П. М., Короткевич В. А.</i> Разработка приложения для автоматизации работы диспетчерской вуза.....	145
<i>Цилулько В. А.</i> О разработке компьютерной игры «Подземелье бегаритта» на Unity .....	146
<i>Цитриков М. А., Орлов В. В.</i> Разработка интерактивного сайта СПА-салона .....	147
<i>Цуранков Д. С.</i> Разработка веб-системы интернет-магазина с чатом поддержки и системой оценки качества обслуживания... ..	148
<i>Черкас А. Н., Кузьменков Д. С.</i> Разработка игрового приложения “Madman” в среде Unity .....	149
<i>Чуйко Ю. А.</i> Разработка клиентского приложения для управления услугами веб-хостинга .....	150
<i>Чёрная Д. С., Кузьменков Д. С.</i> Разработка веб-сайта магазина одежды с использованием HTML5, CSS3, JavaScript, Node.js ...	151
<i>Шамко К. С.</i> Реализация принципов безопасной frontend-разработки в современных фреймворках .....	152
<i>Шацкая А. В., Кузьменкова Е. Ю.</i> Разработка интернет-магазина по продаже мебели с использованием фреймворка Django .....	153
<i>Шульга Д. А.</i> Оптимизация тестирования безопасности веб-приложений: сравнение подходов на основе Python и декларативных JSON-конфигураций .....	154
<i>Шумский Е. Ю.</i> Алгоритм гибридного шифрования для обеспечения приватности данных .....	155
<i>Шутова А. П., Кузьменкова Е. Ю.</i> Разработка клиентской части приложения с использованием ASP.NET Core .....	156
<i>Щербак А. А., Лубочкин А. В.</i> Разработка микросервиса для мониторинга авиаперелетов в реальном времени .....	158

<i>Kurachka K., Huanhai Ren, Xuemei Wang</i> Discussion of key technologies for auxiliary diagnosis of degenerative lumbar-spine disorders based on mri analysis.....	159
<i>Dolnikov A. D., Kuzmenkova E. Y.</i> Development of a mobile application for creating and searching for events by interests using react native.....	160

## **ПЕРВЫЕ ШАГИ В ИТ-СФЕРЕ**

<i>Ашаев К. Е.</i> Разработка приложения для синхронизации расписания занятий с сервисом Google Calendar.....	162
<i>Беликеев М. А., Титова Л. К.</i> Разработка системы автономного управления и процедурной анимации многозвенного объекта в среде UNITY3D.....	163
<i>Борисёнок Е. А., Осипенко Н. Б.</i> Разработка игрового приложения “Emerge” в жанре изометрического шутера на языке C#. ..	164
<i>Говенько Е. В.</i> Начало пути в ИТ-сфере: осознанный старт.....	165
<i>Гордейчик Н. А.</i> Классификация современных прикладных программно-аппаратных систем.....	166
<i>Жукевич А. А.</i> Планировщик задач.....	167
<i>Зайко Е. А.</i> Разработка серверной части веб-приложения на React с использованием технологий Node.js и Express.....	168
<i>Закарян В. Э., Комракова Е. В.</i> Архитектура интеллектуальной web-платформы электронной коммерции для подбора совместимых компьютерных комплектующих .....	169
<i>Заяц С. М.</i> Рекомендации для начинающих в ИТ-сфере.....	170
<i>Зверко С. В., Киргинцева С. В.</i> Разработка веб-приложения для онлайн-продажи железнодорожных билетов с использованием PHP, MySQL и Bootstrap .....	172
<i>Касянович М. П.</i> Приложение для планирования задач на языке Java .....	173
<i>Кирчук А. В., Закревская В. С.</i> Проектирование и реализация игровых уровней в 2D-платформере.....	174
<i>Кособуцкая Д. С., Осипенко Н. Б.</i> Интерактивная система визуализации данных в среде TouchDesigner .....	175
<i>Курьянович Д. В., Комракова Е. В.</i> Автоматизированная система учета услуг ветеринарной клиники с салоном груминга .....	176

<i>Латин В. В.</i> Разработка приложения для планирования задач с функциями уведомлений и категоризации.....	177
<i>Матафонов Ю. В., Осипенко Н. Б.</i> Разработка программного агента для автоматизации торговли инвентарем в игре Path of Exile на языке Python.....	178
<i>Мещанинова К. В., Токочаков В. И.</i> Разработка информационной системы учета посещаемости и успеваемости .....	180
<i>Мусафиров А. Ю., Комракова Е. В.</i> Применение иерархического обучения с подкреплением в стохастических средах стратегических игр .....	181
<i>Недбайло Р. А., Токочаков В. И.</i> Информационная система введения и учёта основных средств в бюджетных организациях....	182
<i>Новицкая Д. Д.</i> Разработка 2D-игры «Пушистый сон» .....	183
<i>Петрович К. А., Осипенко Н. Б.</i> Разработка VK Mini-приложения «Мечты» на основе JavaScript и React .....	184
<i>Пигусова Д. В., Осипенко Н. Б.</i> Разработка мобильного приложения для 3D-визуализации интерьеров на языке C# .....	186
<i>Пикас Н. В., Осипенко Н. Б.</i> Прототип полнофункционального веб-сервиса бронирования гостиничных услуг “Myhotel”.....	187
<i>Самулевич П. В.</i> Обеспечение сетевой безопасности Linux-инфраструктуры с использованием межсетевых экранов и iptables .....	188
<i>Светловский Е. А.</i> От монолита к микросервисам: опыт переписывания учебного To-Do листа на Event-Driven Architecture..	189
<i>Стракович Д. В.</i> Роль самообразования в IT-сфере .....	190
<i>Стракович Е. И., Осипенко Н. Б.</i> Разработка 2D-платформера на языке C# с использованием фреймворка Unity .....	191
<i>Тарканов А. В.</i> Разработка web-приложений с использованием библиотеки React и моделей NLP .....	192
<i>Храменков С. А., Киргинцева С. В.</i> Разработка web-сайта «Магазин электронных гаджетов и аксессуаров» с использованием HTML, CSS .....	193
<i>Шавель А. В.</i> Каталог растений с напоминаниями о поливе.....	194
<i>Dai Q. W.</i> Educational administration system.....	195
<i>Jia Y. T.</i> Global earthquake monitoring systeme.....	196
<i>Li Sh. T.</i> Product portal API system.....	197
<i>Sun D. Y.</i> Book management system.....	198
<i>Wang J. W.</i> Shopping expense tracker app.....	199

<i>Wang Y.</i> Repair system development.....	200
<i>Xiang P.</i> Design and implementation of a dormitory management system based on C# and Windows Forms.....	201
<i>Zhou H.</i> NCAA basketball data analysis system.....	202

Научное издание

**Новые математические методы  
и компьютерные технологии  
в проектировании, производстве  
и научных исследованиях**

Материалы XXIX Республиканской научной конференции  
студентов и аспирантов  
(Гомель, 23–25 марта 2026 года)

В двух частях

Часть 2

Ответственный за выпуск *С. И. Ленденкова*

Подписано в печать 03.06.2026. Формат 60×84 1/16.  
Бумага офсетная. Ризография. Усл. печ. л. 12,56. Уч.-изд. л. 13,73.  
Тираж 10 экз. Заказ 336.

Издатель и полиграфическое исполнение:  
учреждение образования  
«Гомельский государственный университет имени Франциска Скорины».  
Специальное разрешение (лицензия) № 02330 / 450 от 18.12.2013 г.  
Свидетельство о государственной регистрации издателя, изготовителя,  
распространителя печатных изданий в качестве:  
издателя печатных изданий № 1/87 от 18.11.2013 г.;  
распространителя печатных изданий № 3/1452 от 17.04.2017 г.  
Ул. Советская, 104, 246028, Гомель.