

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»**

Кафедра математической теории упругости и биомеханики

Разработка системы учета успеваемости обучающихся ВУЗа

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 442 группы

направления 09.03.03 – Прикладная информатика

механико-математического факультета

Ханина Дениса Алексеевича

Научный руководитель
доцент, к.ф.-м.н.

А.М. Донник

Зав. кафедрой
зав. кафедрой, д.ф.-м.н., профессор

Л.Ю. Коссович

Саратов 2025

Введение. В современных образовательных учреждениях одним из ключевых аспектов организации учебного процесса является эффективное управление успеваемостью студентов. Учет и анализ данных об успеваемости, своевременная обработка и предоставление отчетности являются важнейшими элементами для оценки качества образования, принятия управленческих решений и планирования дальнейших шагов развития учебного процесса. В связи с этим возникает необходимость в разработке информационной системы, которая обеспечит автоматизацию процессов учета, обработки и анализа успеваемости студентов, а также упростит взаимодействие между преподавателями, студентами и администрацией.

Данная бакалаврская работа посвящена разработке информационной системы для учета успеваемости студентов. В работе рассматриваются основные требования, предъявляемые к такой системе, ее архитектура, функциональные возможности и технические характеристики. Ожидается, что предложенная система будет способствовать повышению оперативности обработки данных, улучшению качества предоставляемой информации и созданию удобного интерфейса для всех участников учебного процесса.

Целью бакалаврской работы является создание системы, которая обеспечит надежное и эффективное ведение учета успеваемости. В ходе работы будет проведен анализ существующих решений в данной области.

Разработка интерфейса — ключевой этап создания современных веб и мобильных приложений. Именно он обеспечивает комфортное взаимодействие пользователя с системой, отображение информации и выполнение базовой логики непосредственно в браузере или приложении. Стремительное развитие технологий и повышение требований к удобству использования делают проектирование и создание привлекательных, адаптивных и удобных интерфейсов критически важными. Все это определяет обоснованность выбора данной темы для бакалаврской работы.

К современным пользовательским интерфейсам предъявляются определённые требования, включая безопасность, доступность, адаптивность и

удобство. Разрабатываемый проект будет полностью соответствовать текущим стандартам веб-разработки, что гарантирует высокую производительность и стабильность. Это достигается применением нативных технологий языка программирования Python. Грамотно спроектированная структура клиентской части и адаптивная вёрстка предоставят возможность легко вносить изменения или дополнения в отдельные компоненты интерфейса, не требуя при этом масштабных переделок.

Для достижения целей проекта, были поставлены следующие **задачи**:

1. Анализ требований к информационной системе учета успеваемости.
2. Проектирование архитектуры системы и ее компонентов.
3. Разработка информационной системы.
4. Оценка эффективности предложенного решения.

Структура и объем работы. Работа состоит из введения, трех разделов, заключения, одного приложения и содержит 40 страниц. Список использованных источников включает 20 наименований.

Раздел 1. Теоретические основы учета успеваемости обучающихся.

Раздел 2. Проектирование системы учета успеваемости.

Раздел 3. Разработка системы учета успеваемости.

Основное содержание работы. Работа посвящена созданию веб-приложения для учета успеваемости в вузах.

Во введении описывается актуальность темы, ставятся цели и задачи в работе и приводятся примеры уже созданных подобных систем.

Первый раздел посвящен рассмотрению основных понятий, связанных с учетом успеваемости, значимость этого учета и современные методы для решения задач связанных с разработкой системы учета, а также обзор популярных информационных систем. Учет успеваемости включает в себя сбор и анализ данных о знаниях студентов, что позволяет выявлять слабые места в обучении и предлагать индивидуальные подходы.

Также в разделе обсуждаются преимущества автоматизации учета, такие как сокращение временных затрат на обработку информации и повышение точности данных. Автоматизированные системы позволяют создавать отчеты и анализировать успеваемость в реальном времени.

В рамках проведенного исследования был осуществлён детальный анализ ключевых характеристик конкурентных продуктов. Особое внимание уделялось изучению их функциональных возможностей, особенностей пользовательских интерфейсов, наличию и качеству персонализированных рекомендательных систем. Этот комплексный анализ позволил не только выявить существующие пробелы в предложениях конкурентов, но и определить перспективные направления для реализации в рамках разрабатываемого проекта. В частности, одним из наиболее значимых конкурентных преимуществ нового продукта стало решение создать именно приложение, а не сайт, для последующего удобства в использовании с мобильных устройств.

В условиях постоянной конкуренции и обилия аналогичных предложений на рынке задача выделения продукта среди множества конкурентов требует комплексного подхода.

Анализ существующих систем учета успеваемости в вузах включает в себя изучение различных подходов, технологий и инструментов, используемых для мониторинга и оценки успеваемости студентов. В России несколько информационных систем созданных для учета успеваемости. Например, ФГИС «Моя школа», «ЭлЖур», «UUSTud», «Система дистанционного обучения Ipsilon Uni», «Дневник.ру», «Баллов.Нет», «1С:Образование».

Например, "Дневник.ру" использует реляционную базу данных, которая хранит информацию о пользователях, предметах, оценках и расписании. Структура базы данных включает таблицы, такие как: Users (пользователи) — информация о студентах, преподавателях и родителях, Subjects (предметы) — данные о курсах и их характеристиках, Grades (оценки) — записи об успеваемости студентов по каждому предмету, Attendance (посещаемость) — информация о

посещаемости занятий. Эта структура позволяет эффективно управлять данными и быстро извлекать необходимую информацию.

«Система дистанционного обучения Ipsilon Uni» представляет собой комплексное решение для автоматизации учебного процесса в образовательных учреждениях, включая Саратовский государственный университет. Она включает в себя множество модулей, таких как: учет успеваемости студентов, ведение электронных журналов, формирование отчетов для администрации, интеграция с другими информационными системами. «Система дистанционного обучения Ipsilon Uni» также использует клиент-серверную архитектуру, где клиентская часть реализована как десктопное приложение, а серверная часть функционирует на базе мощного сервера с возможностью доступа через локальную сеть или интернет.

Обе системы имеют свои преимущества и недостатки. "Дневник.ру" отличается простотой использования и доступностью для родителей, что способствует вовлечению их в образовательный процесс. Однако она может быть ограничена в функциональности для высших учебных заведений.

«Система дистанционного обучения Ipsilon Uni», в свою очередь, предлагает более широкий функционал для университетов, включая возможность интеграции с другими системами и модульное построение, что позволяет настраивать систему под конкретные нужды учебного заведения.

Особое внимание в исследовании было уделено изучению целевой аудитории будущего продукта. Полученные данные показали, что потенциальными пользователями системы являются две основные группы: школы и вузы. В ходе исследования были тщательно проанализированы основные мотивации потенциальных пользователей, среди которых особо выделяются желание повысить простоту интерфейса, сделать систему интуитивно понятную, легкую в использовании и главное со всеми базовыми потребностями студентов и преподавателей.

На основе всестороннего анализа рыночной ситуации и потребностей целевой аудитории были сформулированы чёткие функциональные и

нефункциональные требования к разрабатываемому продукту. В перечень функциональных требований вошли: выставление оценок студентам, дистанционная выдача домашнего задания, возможность видеть свое расписание, что касается администрации – это возможность быстро редактировать расписание, назначать новых преподавателей и создавать группы студентов. Нefункциональные требования включают такие важные аспекты как адаптивность интерфейса для различных устройств, устойчивость системы к пиковым нагрузкам, высокая скорость отклика на пользовательские запросы, обеспечение безопасности персональных данных и возможность интеграции с внешними API-сервисами.

Система UUStud российская автоматизированная информационная система для учета образовательной деятельности в вузах и колледжах. Предназначена для автоматизации процессов управления учебным процессом, учета контингента и формирования отчетности.

Анализ существующих информационных систем для учета успеваемости обучающихся показал, что каждая из них имеет свои сильные и слабые стороны. Системы "Дневник.ру" и «Система дистанционного обучения Ipsilon Uni» обеспечивают учет успеваемости, но различаются по целевой аудитории и функционалу. Это создает возможность для разработки новой системы, которая будет учитывать лучшие практики существующих решений и устранять их недостатки.

Во втором разделе рассматриваются возможные методы решение проблемы по созданию информационной системы. Рассматриваются функциональные возможности, преимущества и недостатки. Например, микросервисная архитектура проекта способна внести существенные улучшения в систему оценки знаний в образовательных учреждениях, гарантируя гибкость, возможность масштабирования и удобство интеграции разнообразных функциональных модулей.

Монолитная архитектура представляет собой подход, при котором все компоненты системы объединены в одно целое. Это означает, что все функции,

модули и данные находятся в одной кодовой базе и развертываются как единое приложение.

Особое внимание уделено выбору типа базы данных. Выбор состоял из реляционной или нереляционной. Реляционная база данных (РБД) — это тип базы данных, в которой данные хранятся в виде таблиц, состоящих из строк и столбцов. Каждая таблица представляет собой отдельную сущность, а связи между сущностями реализуются через ключи. Основные принципы реляционных баз данных включают использование структурированных данных, нормализацию и возможность выполнения сложных запросов с помощью языка SQL (Structured Query Language).

Нереляционная база данных (NoSQL) — это тип базы данных, который не использует традиционную реляционную модель для хранения и управления данными. Нереляционные базы данных предназначены для работы с большими объемами данных, высокой скоростью обработки и гибкостью в структуре данных. Они могут быть особенно полезны в сценариях, где данные имеют неструктурированный или полуструктурированный формат.

Нереляционные базы данных позволяют хранить данные без строгой схемы, что дает возможность легко изменять структуру данных по мере необходимости.

Но также в работе рассматривались и другие типы баз данных, например, объектно-ориентированные и графовые.

Объектно-ориентированные системы управления базами данных (СУБД) представляют собой тип СУБД, которые используют объектно-ориентированную модель для хранения и управления данными. Эти системы объединяют концепции объектно-ориентированного программирования и управления базами данных, что позволяет более естественно моделировать сложные данные и их взаимосвязи.

Графовые системы управления базами данных (Графовые СУБД) — это специализированные базы данных, которые используют графовую модель для хранения и управления данными. В отличие от реляционных СУБД, где данные организованы в таблицы, графовые СУБД представляют данные в виде узлов

(вершин) и рёбер (связей) между ними. Это позволяет эффективно моделировать и обрабатывать сложные взаимосвязи между данными.

Рассматривались общие методы необходимые в создании полноценной информационной системы. Взаимодействие клиентской части с сервером через API (Application Programming Interface) является основным способом обмена данными между клиентом и сервером в современных веб-приложениях. Этот процесс обычно осуществляется с помощью HTTP-запросов, таких как GET, POST, PUT и DELETE.

API (Application Programming Interface) — это набор правил и протоколов, который позволяет различным программным компонентам взаимодействовать друг с другом. В контексте веб-разработки API часто используется для взаимодействия между клиентом (веб-приложением) и сервером.

Третий раздел посвящен реализации проекта. В нем описана логика работы приложения. Программирование системы учета успеваемости осуществлялось с использованием языка программирования Python.

В приложении имеются три категории пользователей. Каждый со своими возможностями.

Каждому из пользователей были выданы специальные права, на основе его функций.

Роль admin – выполняет человек который создает аккаунты преподавателей и студентов. Администратор заносит в базу данных пользователей и выдает им роли. Студент или преподаватель. Он создает им логин и пароль. Также он занимается расписанием. Студентам – он указывает номер группы. Преподавателям – их предмет. Также он заполняет расписание, указывая время пары, номер группы и предмет. На рисунке 1 приведена стартовая страница администратора

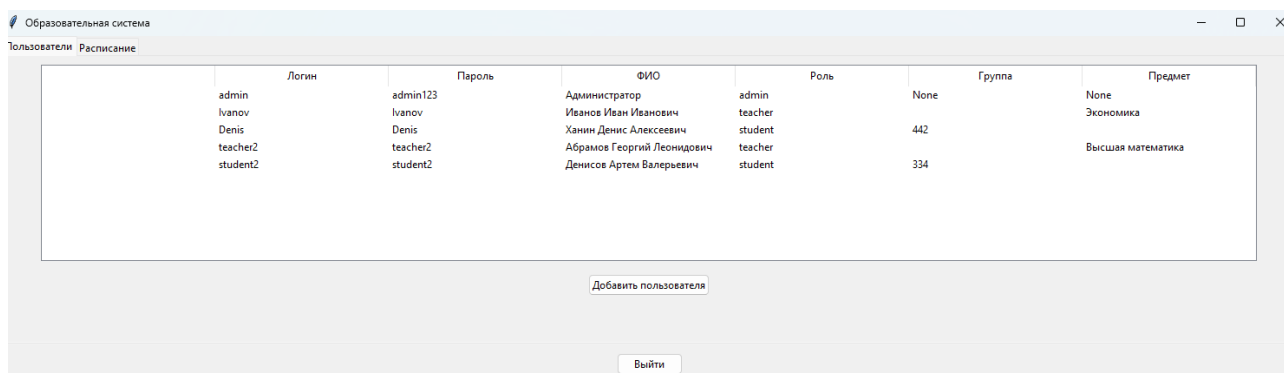


Рисунок 1 – Страница “Пользователи”

Роль teacher – выполняет пользователь, который ведет занятия и эту роль он получает от администратора. В его возможности входит выдача домашнего задания выбранной группе. Установка дедлайна для выполнения этого задания. Выставление оценки отдельно выбранному студенту группы и просмотр своего расписания. На рисунке 2 представлен пример страницы со студентами одного из преподавателей

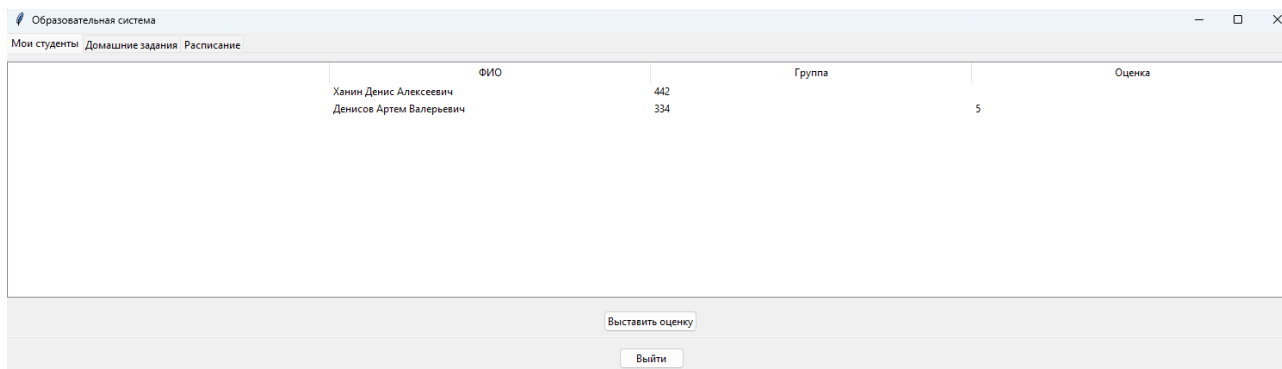


Рисунок 2 – Страница “Мои студенты”

Роль student – пользователю, который учится в учебном заведении и эту роль ему присваивает администратор. Он имеет свои индивидуальные возможности, а именно просмотр своей успеваемости по всем предметам. Возможность отправлять домашние задания, а также просматривать свое расписание. На рисунке 3 представлена страница успеваемости одного из студентов.

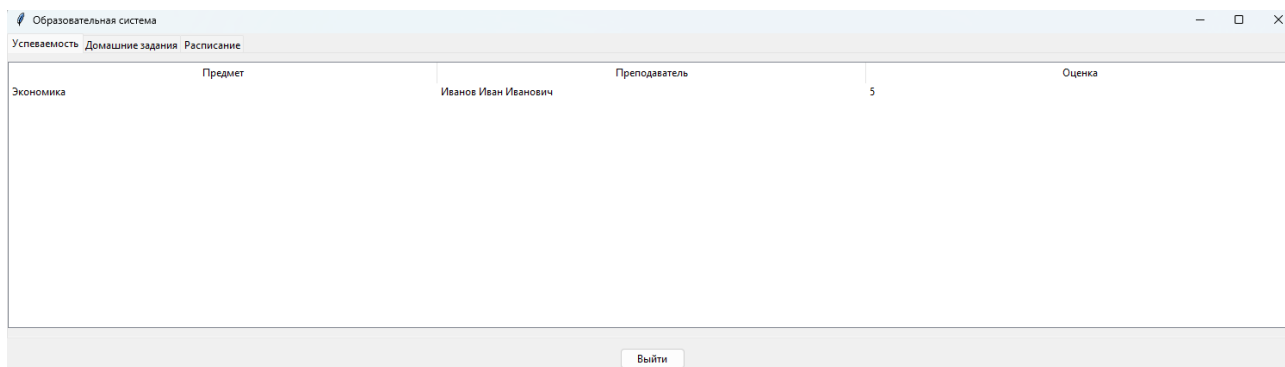


Рисунок 3 – Страница “Успеваемость”

Все приложение разбито на модули, позволяющие редактировать возможности каждой из трех ролей. Модуль ввода оценок – позволяет преподавателям вводить и редактировать оценки студентов. Пользовательский интерфейс – разработан с акцентом на удобство использования, что позволяет пользователям легко ориентироваться в системе.

Было произведено функциональное тестирование – проверены все функции на соответствие требованиям. Анализ производительности системы показал высокую производительность даже при высокой нагрузке.

Интеграция новой системы учета успеваемости с уже существующими информационными системами вуза является важным этапом реализации проекта. Это позволяет обеспечить единый доступ к данным и повысить эффективность работы.

Заключение. В результате выполнения бакалаврской работы разработано и реализовано приложение информационной системы для учета успеваемости студентов. Проект охватывает полный цикл интерфейсного проектирования — от анализа рынка до программной реализации. Сочетание эстетической выразительности и технической надёжности позволило создать решение, соответствующее современным требованиям в области цифровых сервисов.

Работа может рассматриваться как фундамент для дальнейшего масштабирования проекта, включая внедрение новых функций, а также выход на рынок мобильных приложений. Полученные результаты демонстрируют

возможности кроссдисциплинарного подхода, объединяющего программирование и исследования в области пользовательского опыта.