

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение высшего образования  
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий  
(наименование института полностью)

Кафедра «Прикладная математика и информатика»  
(наименование)

09.03.03 Прикладная информатика  
(код и наименование направления подготовки)

Разработка социальных и экономических информационных систем  
(направленность (профиль))

## ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему Разработка информационной системы управления образовательными ресурсами школы

Обучающийся

Е.Б. Султанов  
(Инициалы Фамилия)

(личная подпись)

Руководитель

канд. пед. наук, О.Ю. Копша

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2023

## **Аннотация**

Тема бакалаврской работы: «Разработка информационной системы управления образовательными ресурсами школы».

Объектом исследования является система управления образовательными ресурсами школы.

Целью выпускной квалификационной работы является разработка информационной системы управления образовательными ресурсами школы.

Работа состоит из введения, трех глав, заключения и списка литературы.

Первая глава посвящена анализу предметной области автоматизации, проводится характеристика предметной области и рассматриваются бизнес-процессы. Во второй главе проводится анализ структуры системы, проводится функциональное моделирование и проектирование базы данных. В третьей главе выполняется полное описание работы и функционирования системы. В заключении сформулированы основные выводы и результаты выполненной работы.

Бакалаврская работа состоит из 52 страниц текста, 20 рисунков, 3 таблиц и 25 источников литературы.

## Содержание

Глава 1 Функциональное моделирование предметной области.....	6
1.1 Техничко-экономическая характеристика предметной области .....	6
1.2 Концептуальное моделирование предметной области .....	13
1.3 Анализ существующих разработок .....	16
Глава 2 Логическое проектирование информационной системы управления образовательными ресурсами школы .....	24
2.1 Выбор технологии логического моделирования .....	24
2.2 Логическая модель и ее описание .....	25
2.3 Проектирование базы данных .....	28
2.4 Требования к аппаратно-программному обеспечению.....	30
Глава 3 Физическое проектирование информационной системы управления образовательными ресурсами школы .....	32
3.1 Выбор архитектуры АИС .....	32
3.2 Выбор технологии разработки программного обеспечения АИС .....	33
3.3 Тестирование функций .....	43
Заключение .....	48
Список используемой литературы и используемых источников.....	49

## Введение

В современных условиях цифровизации образования всё более актуальным становится вопрос автоматизации управления образовательными ресурсами школы. Это связано с необходимостью повышения эффективности и качества образовательного процесса, а также с необходимостью оптимизации расходов на содержание и эксплуатацию образовательных ресурсов.

Актуальность темы состоит в том, что школа является сложной организацией, в которой задействовано большое количество ресурсов. Для эффективного управления этими ресурсами необходимо иметь чёткую систему учёта и контроля. Автоматизация управления образовательными ресурсами школы позволит повысить эффективность использования образовательных ресурсов. Сократить время на выполнение рутинных операций, связанных с учётом и контролем ресурсов, что позволит сотрудникам школы сосредоточиться на более важных задачах. А также оптимизация расходов на содержание и эксплуатацию образовательных ресурсов. Автоматизированная система позволит сделать информацию о ресурсах более доступной для сотрудников школы, что будет способствовать повышению эффективности принятия решений.

Целью ВКР является разработка информационной системы управления образовательными ресурсами школы.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Провести анализ существующих систем управления образовательными ресурсами школы.
2. Разработать функциональную модель информационной системы.
3. Разработать техническое решение информационной системы.
4. Провести испытания информационной системы.

Объектом исследования является система управления образовательными ресурсами школы.

Предметом исследования является автоматизация управления образовательными ресурсами школы. Для облегчения и автоматизации образовательного процесса.

Научная и практическая значимость работы обусловлена тем, что она содержит выводы и рекомендации, которые могут быть использованы для улучшения автоматизации управления образовательными ресурсами школ.

Курсовая работа состоит из введения, трёх глав, заключения, списка использованных источников. Введение является обоснование актуальности темы, цели, задач, предмета и объекта исследования. В первой главе идет анализ системы управления образовательными ресурсами школы – анализ существующих систем управления образовательными ресурсами школы, выявление проблем и недостатков; во второй главе представлена разработка и реализация проекта – разработка функциональной модели информационной системы, выбор программного и аппаратного обеспечения, разработка программного кода; в третьей главе выполняется тестирование и оценка эффективности информационной системы – проведение испытаний информационной системы, оценка эффективности информационной системы. Приложение содержит фрагменты программного кода.

## **Глава 1 Функциональное моделирование предметной области**

### **1.1 Техничко-экономическая характеристика предметной области**

#### **1.1.1 Характеристика школы**

Целью Муниципального бюджетного общеобразовательного учреждения «Средняя общеобразовательная школа № 15» заключается в обеспечении доступного и высококачественного образования для учащихся начальной, основной и средней школьных ступеней. Учреждение стремится обеспечить качественное обучение и развитие учащихся, подготовку их к успешной жизни и работе в современном обществе, а также создание условий для профессионального и личностного роста педагогических работников.

На веб-сайте учреждения предоставляется информация о образовательных программах, реализуемых в текущем учебном году:

Основная образовательная программа начального общего образования определяет содержание и организацию образовательного процесса на ступени начального общего образования. Она состоит из нескольких взаимосвязанных программ, каждая из которых является самостоятельным элементом, обеспечивающим конкретное направление деятельности школы. Эта программа обеспечивает функционирование и развитие школы в соответствии с основными принципами государственной политики Российской Федерации в сфере образования, изложенными в Федеральном законе «Об образовании в Российской Федерации» от 29 декабря 2012 года № 273-ФЗ.

Образовательная программа основного общего образования разработана в соответствии с требованиями Федерального государственного образовательного стандарта основного общего образования. Она определяет цели, задачи, планируемые результаты, содержание и организацию образовательного процесса на ступени основного общего образования. Главная цель этой программы – формирование общей культуры, духовно-нравственного, гражданского, социального, личностного и интеллектуального

развития учащихся, а также саморазвития и самосовершенствования, способствующих их успешности в обществе, развитию творческих способностей и сохранению здоровья.

Основная образовательная программа среднего общего образования разработана на основе Федерального государственного образовательного стандарта среднего общего образования. Она учитывает Конституцию Российской Федерации, Конвенцию ООН о правах ребенка и региональные, национальные, и этнокультурные потребности Российской Федерации. Программа направлена на достижение обучающимися образовательных результатов в соответствии с требованиями Федерального государственного образовательного стандарта, а также на формирование целей, задач, планируемых результатов, содержания и организации образовательной деятельности на уровне среднего общего образования. Она реализуется через урочную и внеурочную деятельность с соблюдением государственных санитарно-эпидемиологических норм и правил. Программа включает три раздела: целевой, содержательный и организационный. Кроме того, школа предоставляет возможность дистанционного обучения.

Задачи школы включают:

1. Предоставление образовательных программ. Школа реализует общеобразовательные программы начального общего, основного общего и среднего общего образования, которые соответствуют требованиям государственных стандартов. Это позволяет учащимся получить основные знания и навыки в различных предметных областях.

2. Развитие личности учащихся. Школа ставит перед собой задачу помочь каждому ученику раскрыть свой потенциал, развить свои таланты и достичь личностного и интеллектуального роста.

3. В сфере развития основных способностей школа стремится развивать жизненно важные компетенции своих учеников. Эти компетенции включают в себя такие элементы, как четкое общение, способность к критическому мышлению, умелое решение проблем, навык коммуникации и множество

дополнительных навыков. Так, эти навыки дадут учащимся способность ориентироваться и побеждать в испытаниях, предъявляемых современным обществом, а также окажутся ценными на их последующем жизненном пути.

4. Школа также берет на себя роль в обеспечении подготовки будущих образовательных начинаний и профессиональных устремлений учащихся. Это облегчает подготовку студентов к переходу на высшее образование или к изучению различных карьерных путей.

Подводя итог, следует подчеркнуть основные цели типичного образовательного учреждения. Они включают в себя обязанность предоставлять высококачественное образование, содействие росту учащихся, развитие устойчивых знаний и разнообразного спектра способностей в различных предметах, развитие критического мышления и творческого потенциала, а также воспитание эмоционального потенциала учащихся и социальный интеллект.

Понимание многогранной, сложной организационной архитектуры, присущей учебному заведению, становится жизненно важным условием для опознавания способов организации различных образовательных усилий, отличительных ролей, выполняемых различными отделами и персоналом, а также метода, с помощью которого взаимодействие между ними систематически координируется. Такое понимание дает учащимся более глубокое представление о процедурных тонкостях, что позволяет им более эффективно использовать доступные ресурсы. Преподаватели и другие педагогические сотрудники также извлекают пользу из такого понимания, поскольку оно дает им возможность более конструктивно взаимодействовать с коллегами и администрацией школы, а также выполнять свои профессиональные обязанности в строгом соответствии с заранее установленными руководящими принципами и строгими стандартами качества.

Если говорить далее о структуре, то муниципальное бюджетное образовательное учреждение, именуемое «Средняя школа № 15», состоит из нескольких иерархических уровней и подразделений, каждое из которых



наделено конкретными обязанностями, полномочиями и сетью взаимоотношений между персоналом.

Руководящему совету, играющему ключевую роль в управлении школой, поручена разработка тактических планов работы. Кроме того, на Совет возложена обязанность налаживать отношения с местными органами управления и другими важными заинтересованными сторонами. Эта совместная организация объединяет добровольных сотрудников и выступает в качестве консультативной организации по вопросам, касающимся преподавательской деятельности в школьной среде.

Обращаясь к Методическому совету, его обязанности имеют первостепенное значение и включают в себя:

- Контроль за координацией работы методических объединений и других структурных подразделений методической службы школы, основной задачей которой является методическое обеспечение учебного процесса.

- Разработка основных направлений методической работы школы.

- Установление целей и задач методической службы школы.

- Оказание методической помощи рабочим учебным планам и программам элективных курсов, а также разработке учебно-методических и дидактических материалов.

- Организация проектных и исследовательских инициатив, направленных, среди прочего, на внедрение новых педагогических технологий, разработку собственных программ, апробацию учебно-методических комплексов.

- Консультирование сотрудников по вопросам повышения профессиональных навыков, методов проведения занятий, их учебно-методическое и материально-техническое обеспечение.

- Создание мероприятий по обобщению и распространению педагогического опыта сотрудников школы.

1. Административный персонал: Административный персонал включает секретарей, бухгалтеров и других сотрудников, ответственных за

организационные и финансовые аспекты работы школы. Они занимаются ведением документации, учетом финансовых операций и общим организационным обеспечением школы.

2. Педагогический персонал состоит из учителей, которые ответственны за учебный процесс учащихся. Педагогический состав является коллегиальным органом управления, задачей которого является повышение качества образовательного процесса, его условий и результатов. Педагогический совет имеет целью обеспечить педагогическую целесообразность деятельности школьного совета и администрации.

При принятии решений Педагогический совет руководствуется Конвенцией ООН о правах ребенка, федеральным, региональным и местным законодательством в области образования и социальной защиты, Уставом школы и настоящим Положением. Решения Педагогического совета являются обязательными для всех участников образовательного процесса в МБОУ СОШ №15 и вводятся в действие приказом директора.

Основные задачи Педагогического совета включают:

- Определение основных направлений образовательной деятельности школы, дифференциации учебно-воспитательного процесса, условий и сроков аттестации обучающихся по индивидуальным учебным планам, а также способов улучшения воспитательной работы.

- Осуществление опережающей информационно-аналитической работы на основе достижений психолого-педагогической науки и практики образования, контроля за выполнением Устава и других локальных актов школы, регламентирующих образовательную деятельность, и защиты социальных прав обучающихся.

- Рассмотрение организации государственной итоговой аттестации и выпуска обучающихся, отчетов педагогических работников, докладов представителей организаций и учреждений, сотрудничающих со школой в образовательных вопросах, а также кандидатур педагогических работников, обладающих доверием коллектива.

– Утверждение годовых планов работы МБОУ СОШ №15, образовательных программ школы и её компонентов, а также кандидатур педагогических работников для представления к награждению отраслевыми, государственными и другими наградами.

**Поддерживающий персонал:** Поддерживающий персонал включает сотрудников, которые обеспечивают работу школы в техническом и оперативном плане. Это могут быть уборщики, охранники, обслуживающий персонал, медицинский персонал и другие специалисты, которые поддерживают безопасность, чистоту и комфорт в школьной среде.

**Родительский совет:** совет является органом самоуправления Школы и создаётся в целях развития и укрепления взаимодействия между участниками образовательных отношений: обучающимися, родителями (законными представителями) и педагогическими работниками Школы по различным вопросам обучения и воспитания, организации труда и отдыха детей. Руководство школы координирует работу всех подразделений, обеспечивает коммуникацию, распределение задач и контроль за выполнением поставленных целей. Педагогический и поддерживающий персонал работают в команде, сотрудничая и взаимодействуя для обеспечения эффективного обучения и функционирования школы. Организационная структура школы помогает оптимизировать работу, обеспечивать эффективное использование ресурсов и достижение поставленных образовательных целей. На рисунке 1 представлена структурная схема общеобразовательного учреждения.

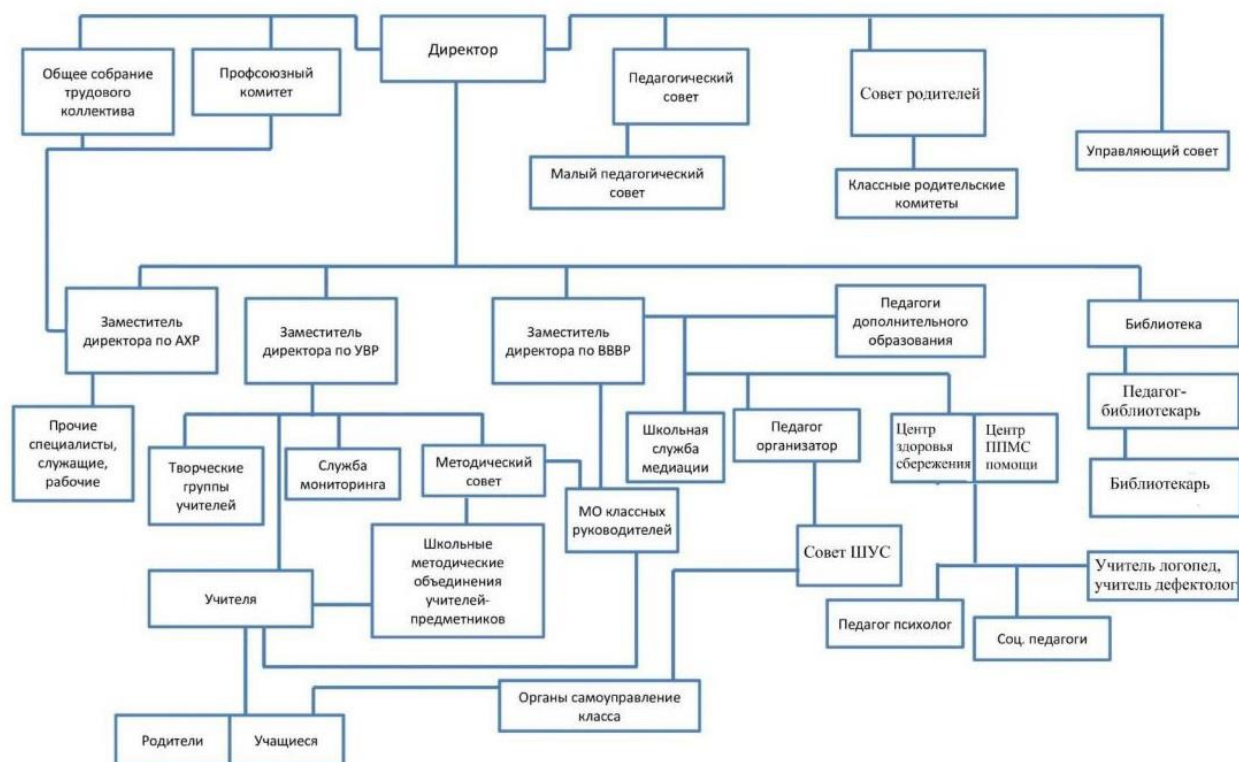


Рисунок 1 – Организационная структура предприятия СОШ №15

### 1.1.2 Сущность задачи автоматизации

Сущность задачи автоматизации «Разработка информационной системы управления образовательными ресурсами школы» заключается в создании программного обеспечения, которое позволит автоматизировать следующие процессы:

1. Планирование и распределение образовательных ресурсов. Система должна позволять сотрудникам школы составлять планы использования образовательных ресурсов, а также распределять ресурсы между различными подразделениями и пользователями.
2. Учёт и контроль использования образовательных ресурсов. Система должна позволять отслеживать состояние образовательных ресурсов, а также контролировать их использование.

3. Анализ эффективности использования образовательных ресурсов. Система должна позволять проводить анализ эффективности использования образовательных ресурсов.

## **1.2 Концептуальное моделирование предметной области**

Для того, чтобы описать и проанализировать текущую ситуацию, выявить проблемы и недостатки в управлении организации, либо в существующей работе системы часто прибегают к построению функциональной модели «как есть» и «как должно быть». Строить модель «как должно быть» помогает определить желаемое состояние системы, установить цели и задачи, а также разработать стратегию для их достижения. Это позволяет улучшить работу системы, повысить ее эффективность и эффективность бизнес-процессов. Кроме того, построение модели «как должно быть» позволяет провести сравнительный анализ текущего состояния системы с желаемым, выявить расхождения и разработать план действий для их устранения.

Функциональная модель будет построена в нотации IDEF0 – это язык моделирования, предназначенный для анализа и проектирования бизнес-процессов. IDEF0 предоставляет специальные символы и правила, чтобы помочь описать функциональные аспекты системы, включая входы, выходы, управление и механизмы обработки данных. IDEF0 часто используется для разработки блок-схем процессов, которые являются графическим представлением последовательности действий, которые должен выполнить бизнес-процесс. Он позволяет создавать наглядные модели, которые помогают более полно понять, как функциональные компоненты системы взаимодействуют друг с другом.

На рисунке 2 представлена модель «КАК ЕСТЬ» («AS-IS»). Первый уровень контекстной модели, использующий нотацию IDEF0, представляет блок «Учебный процесс», включающий входные стрелки, такие как «Задание», который служит источником информации и ресурсов для работы. Механизмом,

исполняющим данный процесс является «Учитель», который предоставляет необходимое обучение и направляет процесс обучения и «Ученик». Выходные стрелки системы включают «Усвоенный материал» и «Оценку», которые являются результатом учебного процесса и отображают степень усвоения материала учеником. Управление системой осуществляется «Учебной программой», которая определяет цели и задачи обучения, а также осуществляет контроль и оценку процесса обучения.

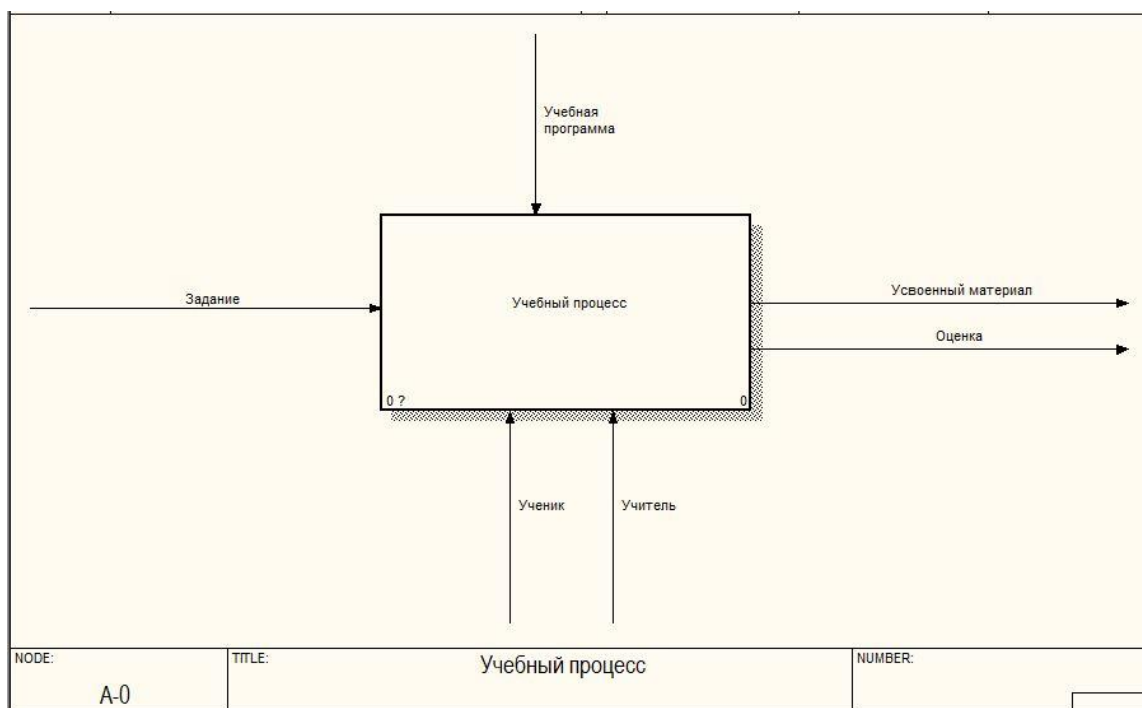


Рисунок 2 – Контекстная модель AS-IS

На декомпозируемой диаграмме (рисунок 3) представлены 5 блоков, которые также можно декомпонировать. Представленные блоки описывают процесс организации обучения начиная с составления задания, заканчивая проверкой работы учителем. В данном процессе недостатком является задержка во времени, как предоставленного материала, так и сдачи работы, возможно и проверке работы, если тип работы относится к тестам. Такой процесс можно оптимизировать при помощи информационной системы по управлению образовательными ресурсами, где учитель сможет создавать отдельные задания

и прикреплять к ним соответствующие материалы, а ученик, в свою очередь, в удаленном режиме сможет прикреплять готовую работу. На рисунке 4 представлен процесс «КАК ДОЛЖНО БЫТЬ», который решает описанную выше проблему.

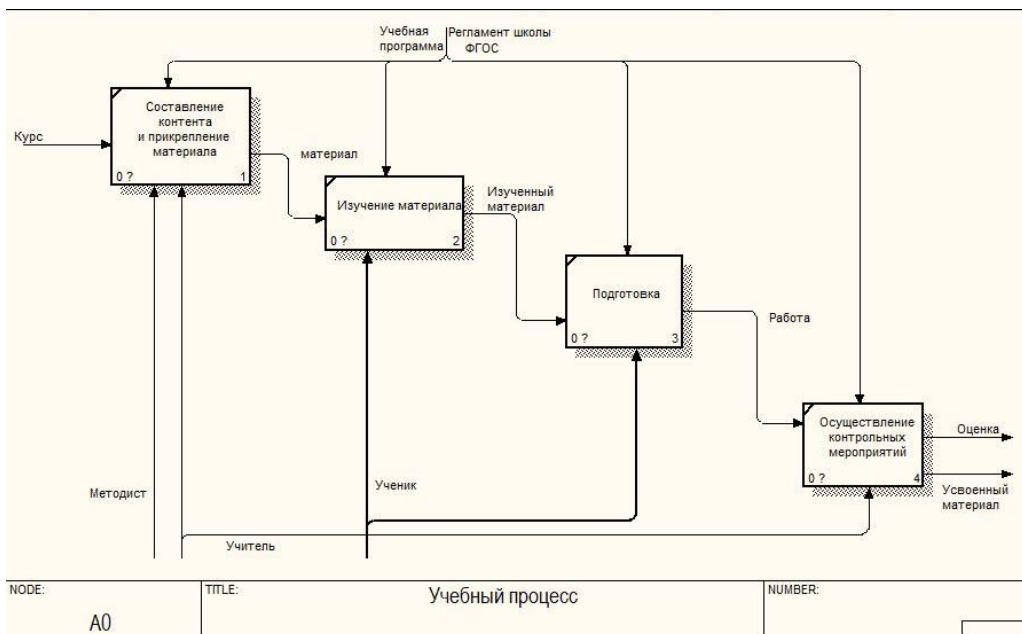


Рисунок 3 – Декомпозиция контекстной модели AS-IS

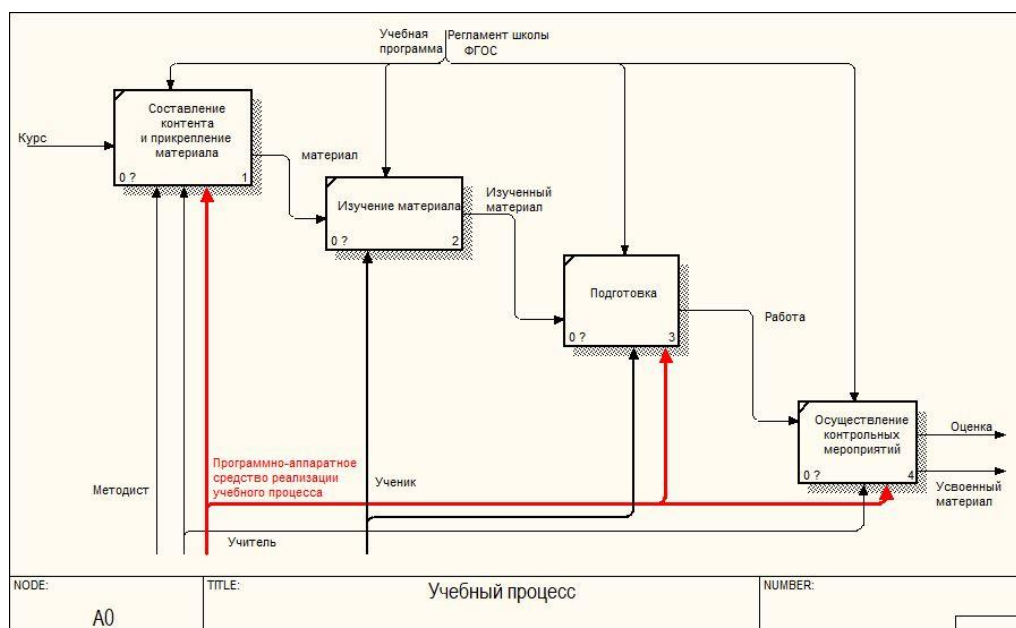


Рисунок 4 – Диаграмма «Как должно быть»

На основании данного замечания и на основании существующих технологий в области оптимизации в различных сферах, можно отметить, что при помощи информационных систем можно ускорить выполнение некоторых процессов, в данном случае – это предоставление материала и дальнейший анализ успеваемости.

### **1.3 Анализ существующих разработок**

Существует ряд разработок для онлайн обучения и тестирования учеников. Эти разработки различаются по следующим критериям:

- Целевая аудитория. Ориентированы на школьников всех возрастов, другие – на учеников определенных классов или предметов.
- Объекты обучения. Некоторые разработки позволяют обучать по всем предметам, другие – по отдельным предметам или темам.
- Формат обучения. Позволяют использовать различные форматы обучения, такие как видеолекции, интерактивные задания, игры.
- Механизм оценивания. Используют автоматическую систему оценивания, другие – требуют участия учителя.
- Доступность: некоторые разработки доступны бесплатно, другие – требуют оплаты.

К числу наиболее известных разработок для онлайн обучения и тестирования учеников в школе России относятся:

1. УМНАЯ ШКОЛА – ориентирована на школьников всех возрастов, позволяет обучать по всем предметам, использует различные форматы обучения, имеет автоматическую систему оценивания и доступна бесплатно.
2. ЯНДЕКС.УЧЕБНИК – подходит для школьников всех возрастов, позволяет обучать по всем предметам, использует различные форматы обучения, имеет автоматическую систему оценивания и доступна бесплатно.



3. РЕШУ ЕГЭ – создана для школьников старших классов, позволяет обучать по предметам ЕГЭ, использует различные форматы обучения, имеет автоматическую систему оценивания и доступна бесплатно.

4. УЧИ.РУ – предлагает курсы для всех возрастов, позволяет обучать по всем предметам, использует различные форматы обучения, имеет автоматическую систему оценивания и доступна по подписке.

На основе перечисленных сервисов проведем сравнительный анализ. В таблице 1 представлен сравнительный анализ сервисов обучения и проведения тестирования для учеников.

Таблица 1 – Сравнительный анализ сервисов обучения и тестирования

Критерий	Умная школа	Яндекс.Учебник	Решу ЕГЭ	Учи.ру
Целевая аудитория	Школьники всех возрастов	Школьники всех возрастов	Школьники старших классов	Школьники всех возрастов
Объекты обучения	Все предметы	Все предметы	Предметы ЕГЭ	Все предметы
Формат обучения	Видеолекции, интерактивные задания	Видеолекции, интерактивные задания	Тесты	Видеолекции, интерактивные задания, игры
Механизм оценивания	Автоматическая система оценивания	Автоматическая система оценивания	Автоматическая система оценивания	Автоматическая система оценивания
Доступность	Бесплатно	Бесплатно	Бесплатно	Бесплатно по подписке

Оцениваемые сервисы действительно имеют ряд общих преимуществ, включая, помимо прочего, их удобство для пользователя, возможность настройки в соответствии с индивидуальными потребностями и способность предоставлять комплексную аналитику. Тем не менее, они демонстрируют определенные различия, которые, в зависимости от конкретных требований пользователя, могут иметь существенные последствия. Например, «Умная школа» и «Яндекс.Учебник» ориентированы на широкую аудиторию учащихся,

в отличие от «Решения ЕГЭ», которая сужает фокус на подготовке к ЕГЭ. Uchi.ru щедро предлагает бесплатные материалы по всем темам, что не отражено в других сервисах и ограничивает свободный доступ к более узкому набору ресурсов. В свете оцениваемых услуг возникает необходимость в разработке индивидуальной системы, способной широко охватить все предметы и темы, включенные в школьную программу.

Для реализации этой цели необходимо решить ряд задач: во-первых, создание технического решения предлагаемой информационной системы. Это предполагает выбор подходящего программного и аппаратного обеспечения в сочетании с разработкой системного программного обеспечения. Полученная в результате система должна дать возможность школьному персоналу разрабатывать планы использования образовательных ресурсов и эффективно распределять эти ресурсы между различными отделами и пользователями. Во-вторых, система должна облегчить управление и регулирование образовательных ресурсов. Это влечет за собой возможность мониторинга состояния этих ресурсов, а также контроля за их использованием. В-третьих, система должна позволять анализировать эффективность использования образовательных ресурсов.

Предполагаемая система образования принимает форму электронной платформы, служащей каналом для получения образования и расширения спектра знаний посредством доступа к цифровым учебным материалам. Эта система дает пользователям возможность углубляться в множество тем, участвовать в курсах и сдавать экзамены. Это представляет собой эффективный метод приобретения знаний, предлагающий гибкость и удобство, одновременно устраняя необходимость физического посещения курсов или колледжей, как того требуют традиционные образовательные пути. К основным целям и задачам разработки обучающей системы относятся:

– В сфере приобретения знаний наша основная цель заключается в улучшении организации и закреплении полученных знаний. Цель системы обучения должна выходить за рамки простой доставки информации; он должен

помочь пользователю создать консолидированное, хорошо организованное понимание выбранного предмета. В результате понимание пользователем конкретной изучаемой темы заметно улучшится;

– Обязанность по обучению и обучению пользователей полностью ложится на плечи системы образования. Роль системы заключается в предоставлении всеобъемлющего и подробного плана обучения, до краев наполненного соответствующей информацией. Формирование структуры должно быть направлено на содействие устойчивому и эффективному обучению пользователя;

– Оценка и подтверждение полученных знаний – еще одна важная функция системы обучения. Он несет ответственность за предоставление инструментов, которые помогают оценить понимание пользователем материала курса. Такие механизмы могут включать в себя викторины или практические упражнения, предназначенные для эффективной оценки понимания пользователем;

– Взаимодействие с системой обучения должно вселить в вас чувство уверенности в вашем понимании. После взаимодействия с системой пользователь должен иметь непоколебимую уверенность в своем понимании предмета, изучению которого он приложил значительные усилия.

Ожидаемые результаты от использования системы обучения, следующие:

– Ключевыми ожиданиями являются эффективное использование времени и сокращение финансовых затрат. Система образования должна предоставить пользователю возможность экономить как время, так и денежные ресурсы. Пользователи не должны регистрироваться на курсы или покупать дорогостоящие учебники, вместо этого доверяя системе обучения;

– Улучшение и расширение понимания пользователей является фундаментальной целью развития системы обучения. Развертывание системы обучения может служить для пользователя средством расширения объема своих знаний в конкретной области обучения;

– Регистрация пользователя и проверка личности — обязательная функция. Система должна обладать возможностью регистрации широкого круга пользователей, включая студентов, преподавателей и административный персонал. Кроме того, оно должно предоставить надежные методы проверки личности тех, кто пытается войти в систему;

– Создание курсов и управление ими является еще одним важным элементом. Система должна упростить процесс создания и управления различными курсами. Эта функция должна охватывать добавление описаний курсов, загрузку образовательного контента и определение предварительных требований и заданий для учащихся;

– Еще одной важной задачей является составление и поддержание расписания занятий. Система должна обеспечивать функциональные возможности для создания и контроля расписания занятий, включая определение времени и места проведения занятий, а также назначение учителей;

– Постановка целей и оценка эффективности — еще одна ключевая функция информационной системы. Учителя должны иметь возможность разрабатывать, назначать учебные задачи, оценивать успеваемость учащихся по этим заданиям, обеспечивать конструктивную обратную связь и сообщать об успехах учащихся;

– Возможности наблюдения и анализа являются важными особенностями информационной системы. Он должен предлагать функции, позволяющие отслеживать активность учащихся и анализировать данные об успеваемости. Эти возможности должны способствовать всестороннему пониманию процесса обучения учителям и администраторам;

– Что касается управления данными и безопасности, информационная система должна обеспечивать безопасное хранение и обработку данных. Он также должен соблюдать требования законодательства о защите персональных данных.

Продолжая процесс определения функциональных требований, следует отметить о существовании бизнес-целей, относящихся к рассматриваемому проекту. Установление бизнес-целей в значительной степени способствует определению основных задач и стратегических траекторий создания системы управления информацией, предназначенной для контроля образовательных ресурсов. Эти цели одновременно служат эталоном, по которому можно измерить эффективность достижения цели. При отсутствии четко сформулированных бизнес-целей может оказаться весьма затруднительным определить жизненно важные функциональные требования и соответствующий им порядок важности. Более того, бизнес-цели служат основой, на которой могут быть построены критерии успеха продукта, и они могут потребовать изменения функциональных требований для облегчения их реализации. Ниже приведены бизнес-цели и спецификации информационной системы, предназначенной для управления образовательными ресурсами:

- Повышение эффективности процесса постановки педагогами учебных задач, а также их оценки;
- Повышение эффективности взаимного общения между педагогами и учениками;
- Проведение углубленного анализа показателей успеваемости учащихся с целью содействия улучшению общего качества образовательного процесса;
- Гарантия надежной безопасности, связанной с сохранением индивидуальных данных студентов;
- Соблюдение законов и правил, касающихся защиты персональных данных;
- Усиление контроля за вовлеченностью студентов в образовательный процесс;
- Централизованное управление учебными материалами. Целью этой инициативы является создание интегрированной системы, отвечающей за централизованное управление различными образовательными ресурсами. Сюда

могут входить академические материалы, расписания учебных занятий, инструменты оценки и другие соответствующие данные, и все это с целью обеспечения их эффективного использования и широкой доступности;

– Улучшение образовательного прогресса: Одной из потенциальных целей проекта может быть создание системы, которая будет служить инструментом оптимизации и улучшения текущего образовательного прогресса. Сюда входит задача разработки функциональных возможностей для создания и администрирования академических курсов, структурирования заданий и экзаменов, облегчения обмена информацией между преподавателями и учащимися, а также отслеживания прогресса студентов;

– Усиление коммуникации и взаимодействия: Миссия проекта заключается в разработке системы, которая будет действовать как катализатор, способствуя и облегчая коммуникацию и взаимодействие, происходящие между преподавателями, учащимися и административным персоналом;

– Обеспечение безопасности данных. Основная цель проекта — предоставить гарантированно безопасную среду для хранения и обработки данных, касающихся учащихся, преподавателей и других пользователей, взаимодействующих с системой. Это требует разработки механизмов аутентификации и авторизации пользователей, методов шифрования данных и решений резервного копирования для защиты от любой потери или несанкционированного доступа;

– Интеграция с уже существующими системами. Потенциально проект может быть направлен на создание бесшовного интерфейса с уже существующими системами управления образовательными ресурсами. Это могут быть электронные журналы, электронная почта и другие связанные системы с целью обеспечения унифицированного и последовательного потока информации.

Эти выделенные моменты служат основными бизнес-целями и требованиями для ИТ-проекта, целью которого является разработка системы управления информацией для управления образовательными ресурсами.

Вывод по первой главе

Предприятие МБОУ СОШ №15 проанализировано с точки зрения технических и экономических характеристик, структуры и организационной структуры.

Произведя анализ существующей модели бизнес-процесса управления образовательными ресурсами можно сделать вывод что, что для автоматизации этого процесса необходимо интегрировать автоматизированную систему, которая в будущем будет иметь более широкий охват, а также позволит учащимся получать образование по всем предметам и темам, которые необходимы для успешной учебы.

Также анализ ИТ-решений показал, что необходимо разработать систему для упрощения процесса предоставления доступа к образовательным ресурсам предприятия.

## **Глава 2 Логическое проектирование информационной системы управления образовательными ресурсами школы**

### **2.1 Выбор технологии логического моделирования**

Методология объектно-ориентированного программирования, известная как UML (унифицированный язык моделирования), является известным и весьма востребованным подходом в современном мире программирования. Он предлагает возможность создавать сложные и масштабируемые системы, которыми легко управлять и которые можно развивать.

Основные причины, по которым отдается предпочтение методологии UML:

- UML присущ атрибут простоты моделирования. Он включает в себя обширный набор диаграмм и обозначений, обеспечивающих точное и полное представление структуры системы и ее поведения. Этот атрибут существенно упрощает процесс освоения и разработки многогранных программных продуктов;
- С точки зрения полезности UML демонстрирует универсальную природу. Он применим для моделирования широкого спектра систем, от миниатюрных приложений до крупных корпоративных платформ, тем самым позиционируя его как гибкий и масштабируемый инструмент;
- Аспект поддержки стандартов является неотъемлемой частью UML. Будучи признанным стандартом ISO, он пользуется широким признанием и поддерживается в различных средах разработки, что обеспечивает его универсальную привлекательность;
- UML улучшает понимание системы. Это дает разработчикам возможность лучше понять структуру и поведение системы, что приводит к созданию превосходного программного продукта. Это также способствует эффективности проектирования. UML помогает разработчикам создавать системы, которые не только более эффективны, но и легко масштабируются;



– С точки зрения упрощения разработки и обслуживания UML является ценным инструментом. Это открывает путь к упрощению процессов разработки и сопровождения, присущих сложным программным системам. Однако UML представляет определенный уровень сложности. Это довольно сложная методология, требующая от разработчиков наличия определенных навыков и знаний. Это также может свидетельствовать об отсутствии гибкости. UML может не обеспечивать достаточную гибкость для моделирования определенных систем.

## **2.2 Логическая модель и ее описание**

В контексте проектируемой информационной системы основными действующими лицами являются учитель и ученик. Было выявлено несколько функциональных требований, в том числе:

– Регистрация и аутентификация пользователей: Система должна обладать возможностью регистрации студентов, преподавателей и административного персонала. Более того, он должен предлагать механизмы безопасной аутентификации для облегчения доступа к системе;

– Создание курсов и управление ими. Система должна предоставлять функции, позволяющие создавать курсы и управлять ими. Сюда входит возможность добавлять описания курсов, загружать учебные материалы, определять предварительные требования и задания для студентов;

– Расписание занятий: Система должна предоставлять функциональные возможности для создания расписаний занятий и управления ими. Это включает в себя определение времени и места проведения занятий, а также информацию, касающуюся учителей;

– Управление заданиями и оценкой: Система должна предлагать факультетам возможность разрабатывать и назначать задачи, а также оценивать их выполнение студентами. Кроме того, он должен поддерживать

предоставление обратной связи и составление отчетов об успеваемости учащихся.

На рисунке 5 представлена диаграмма Use-case, характеризующая функциональные требования к системе.

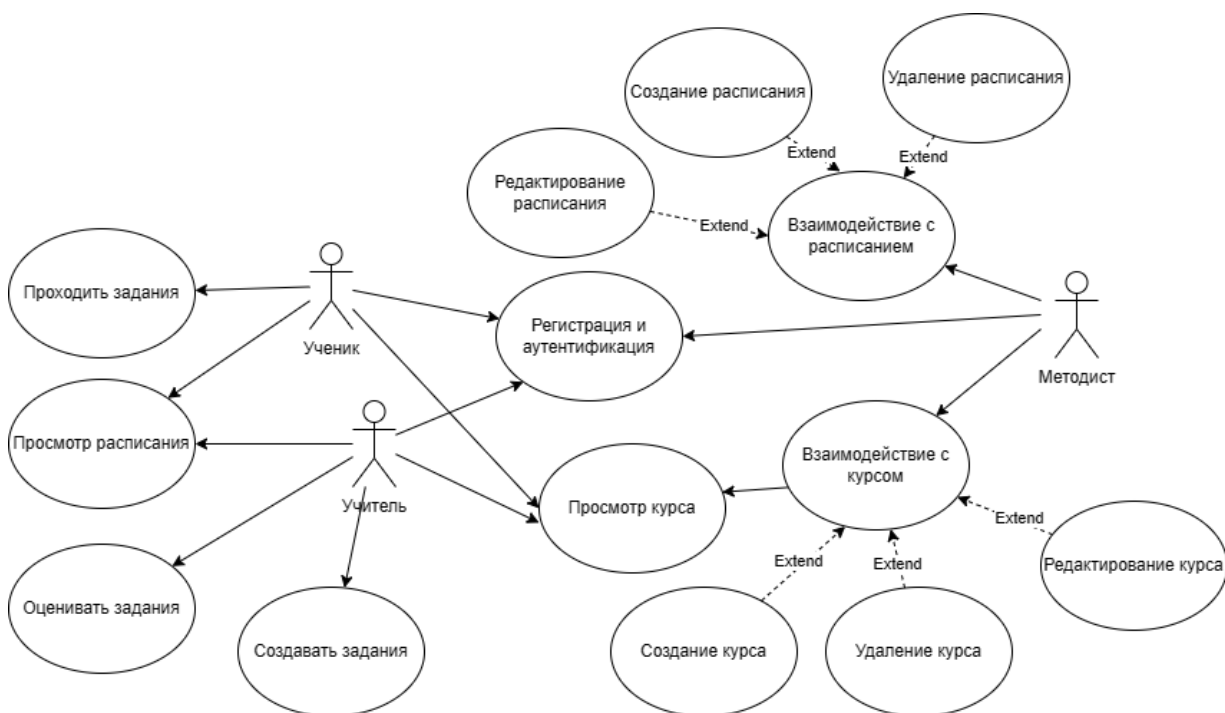


Рисунок 5 – Use-case диаграмма

После создания диаграммы вариантов использования следующим важным этапом является формулирование диаграммы последовательности. Это играет ключевую роль в демонстрации с повышенным уровнем детализации взаимодействий, которые происходят между различными компонентами системы, включая участников (которые обычно являются пользователями), в контексте каждого отдельного варианта использования. Диаграмма последовательности предлагает графический метод для изображения временной шкалы сообщений и взаимодействий, происходящих между объектами в данной системе. В нем описывается способ взаимодействия объектов посредством обмена сообщениями и их взаимных взаимодействий для выполнения определенных действий. В качестве иллюстративного примера

рассмотрим диаграмму последовательности, которая изображает взаимодействие внутри системы управления образовательными ресурсами с участием учащегося, учителя, администратора и базы данных.

- Учащийся имеет возможность просмотреть полный список уроков, предоставленных учителем. Это достигается за счет получения учеником информации по уроку из базы данных.

- После просмотра доступных уроков студент имеет возможность зарегистрироваться на выбранный урок. После этой регистрации в базу данных записываются регистрационные данные.

- После выполнения и выполнения задания преподаватель подтверждает, что работа учащегося принята. Кроме того, информация, относящаяся к выполнению задачи, должным образом отмечается и сохраняется в базе данных.

- Учитель также обладает способностью создавать задания. После создания задания учащийся узнает об этой новой доступной задаче. При этом в базе данных сохраняется информация о создании задачи.

В контексте выполненного учеником задания преподаватель имеет право оценить работу. Результаты этой оценки затем безопасно сохраняются в базе данных.

Методолог, еще один жизненно важный участник системы, обладает способностью устранять существующие уроки и вводить новые. Эти изменения также представлены на диаграмме последовательности.

База данных служит центральным хранилищем, в котором регистрируется вся информация, связанная с действиями и транзакциями пользователей. Это обеспечивает централизованное хранение и доступность данных. Диаграмма последовательности представляет собой наглядное пособие, отображающее последовательность операций в системе управления образовательными ресурсами. Он графически демонстрирует взаимодействие между различными участниками и базой данных.

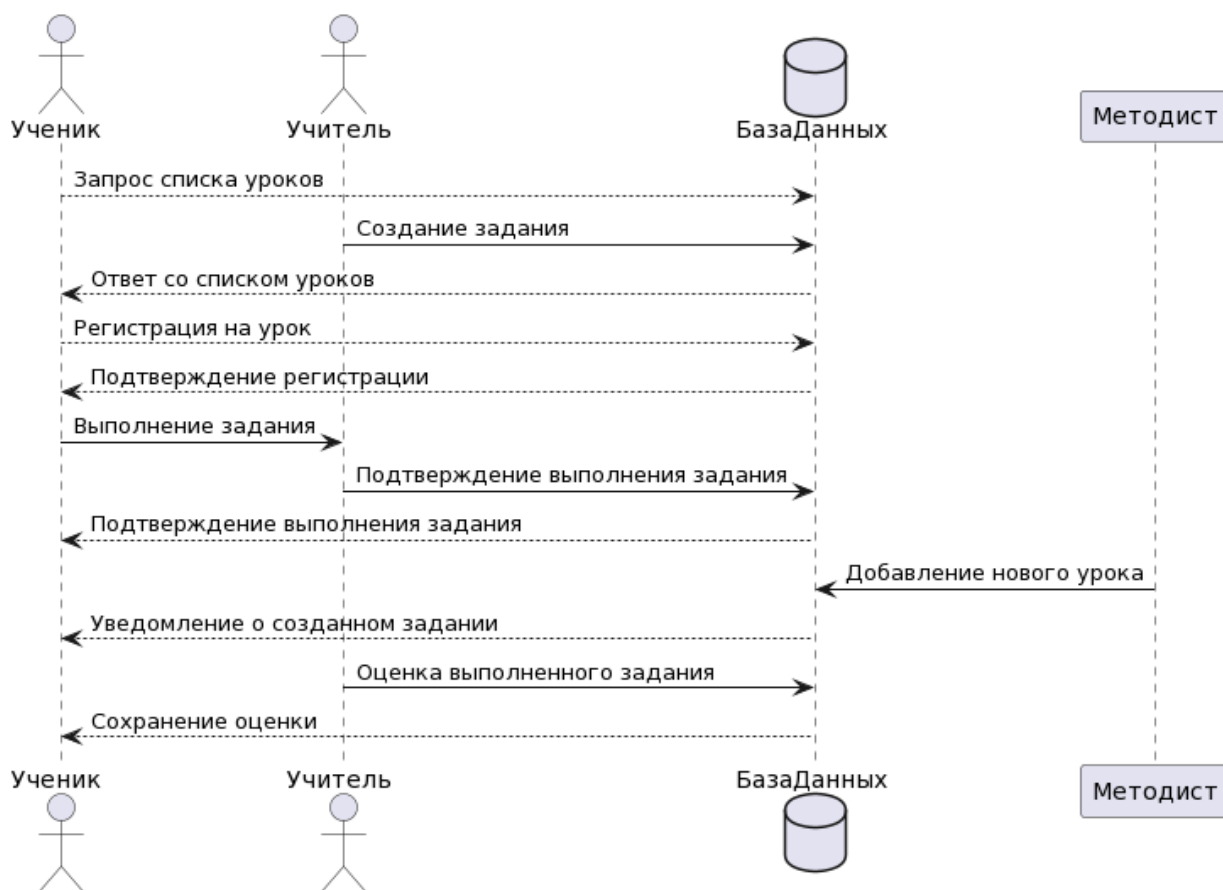


Рисунок 6 – Диаграмма последовательности

Каждое взаимодействие пользователя с системой работает с обращением в базу данных, к таким взаимодействиям можно отнести вход в систему, регистрация, добавление курсов, оценивание курсов и так далее.

### 2.3 Проектирование базы данных

Процессом создания предварительной версии базы данных, которая позволяет проверить и протестировать ее функциональность и соответствие требованиям проекта. Прототипирование базы данных позволяет выявить ошибки и недочеты в структуре базы данных на ранней стадии разработки, что упрощает последующие этапы проекта.

В прототипе базы данных должны обязательно присутствовать такие сущности как пользователь, расписание, дисциплины, класс, материалы, тесты,

курсы. Рассмотрим сущности в рамках ограничений и связей между другими сущностями представленной в таблице 2.

Таблица 2 – Прототип базы данных

Users	id	Идентификатор	PK (int)
	name	Имя	Varchar (50)
	email	Е-маил	Varchar (100)
	password	Пароль	Varchar (100)
	role	Роль: 1 – ученик, 2 – учитель, 3 – администратор	Varchar (20)
	class	Номер класса	FK (int)
Class	id_class	Идентификатор	PK (int)
	name	Наименование	Varchar (20)
Discipline	id_dis	Идентификатор	PK (int)
	name	Наименование	Varchar (255)
Schedule	Id_sch	Идентификатор	PK (int)
	day	День недели	Varchar (20)
	number	Аудитория	Varchar (10)
	time	Время	Varchar (10)
	class	Номер класса	FK (int)
	discipline	Номер дисциплины	FK (int)
Answers	id	Идентификатор	PK (int)
	Question id	Номер вопроса	FK (int)
	Answer_text	Текст вопроса	Varchar (255)
	letter	Да/нет	Bool
Questions	id	Идентификатор	PK (int)
	Test_id	Номер теста	FK (int)
	Questions_text	Вопрос	Varchar (255)
Tests	id	Идентификатор	PK (int)
	title	Заголовок	Varchar (255)
	description	Описание	Varchar (255)
	Section_id	Номер раздела	FK (int)
results	id	Идентификатор	PK (int)
	User_id	Пользователь	FK (int)
	Test_id	Тест	FK (int)
	Correct_answers	Результат	int
Materials	id	Идентификатор	PK (int)
	Sub_section_id	Номер подраздела	FK (int)
	Material_name	Наименование	Varchar (100)
	Material_description	Описание	text
	Material_content	Контент	Varchar (100)
Sub_course_Sections	id	Идентификатор	PK (int)
	Section_id	Номер раздела	FK (int)
	Sub_section_name	Наименование	Varchar (100)

## Продолжение таблицы 2

	Sub_section_description	Описание	Text
	img	Изображение	Varchar (100)
Course Sections	id	Идентификатор	PK (int)
	Section_name	Наименование	Varchar (100)
	Section_description	Описание	Text
	img	Изображение	Varchar (100)

На рисунке 7 представлена разработанная схема базы данных.

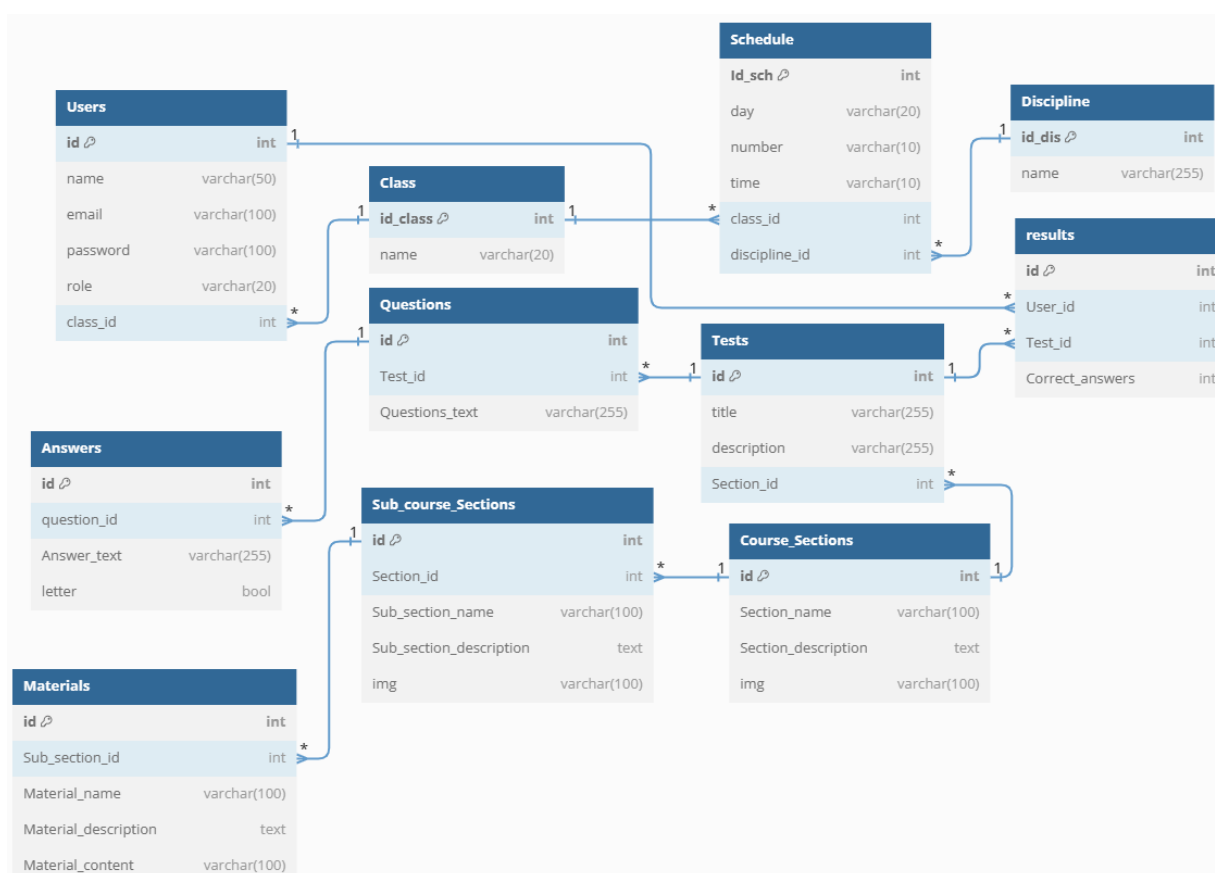


Рисунок 7 – Схема базы данных

## 2.4 Требования к аппаратно-программному обеспечению

Приведем следующие требования к аппаратно-программному обеспечению разрабатываемой информационной системы.

Аппаратное обеспечение серверной инфраструктуры:

- Количество серверов: от 2 до 4, в зависимости от масштаба системы
- Тип серверов: x86-серверы с процессорами Intel Core i7 или аналогичными
- Оперативная память: от 16 до 32 Гб на сервер
- Жесткие диски: от 1 Тб на сервер
- Сетевое оборудование: коммутаторы и маршрутизаторы с пропускной способностью не менее 1 Гбит/с

Программное обеспечение:

- Операционная система: Windows Server 2022 или Ubuntu Server 22.04
- База данных: PostgreSQL

В качестве хостинга будем использовать размещение системы на серверах провайдера облачных услуг. Это позволяет снизить затраты на оборудование и обслуживание системы. Требования к аппаратно-программному обеспечению АИС информационной системы управления образовательными ресурсами школы определены с учетом масштаба системы и предполагаемого количества пользователей.

Вывод по второй главе

В данной главе были рассмотрены следующие вопросы:

Выбор технологии логического моделирования. В качестве технологии логического моделирования была выбрана методология UML. На ее основе была разработана логическая модель данных. Были определены требования к аппаратному и программному обеспечению, необходимые для работы информационной системы.

В результате разработки главы курсовой работы была получена логическая модель данных, которая соответствует предметной области информационной системы управления образовательными ресурсами школы. На основе логической модели была создана база данных.

## Глава 3 Физическое проектирование информационной системы управления образовательными ресурсами школы

### 3.1 Выбор архитектуры АИС

Для реализации информационной системы управления образовательными ресурсами школы может быть использована одна из следующих архитектур:

- Файл-серверная архитектура. В этой архитектуре все данные и логика приложения хранятся на одном сервере. Это простая и недорогая архитектура, которая подходит для небольших систем.
- Двухзвенная архитектура клиент-сервер. В этой архитектуре данные хранятся на сервере, а логика приложения выполняется на клиентских компьютерах. Это более масштабируемая архитектура, чем файл-серверная, и она подходит для систем с большим количеством пользователей.
- Трехзвенная архитектура клиент-сервер. В этой архитектуре данные хранятся на сервере, логика приложения выполняется на сервере приложений, а пользовательский интерфейс выполняется на клиентских компьютерах. Это самая гибкая и масштабируемая архитектура, но она также является самой сложной и дорогостоящей.

Сравнительный анализ архитектур приведен в таблице 3.

Таблица 3 – Сравнительный анализ архитектур

Архитектура	Достоинства	Недостатки
Файл-серверная	Простота, низкая стоимость	Ограниченная масштабируемость, сложность администрирования
Двухзвенная	Масштабируемость, простота администрирования	Сложность разработки клиентского приложения
Трехзвенная	Гибкость, масштабируемость, простота администрирования	Сложность разработки и администрирования



Для реализации информационной системы управления образовательными ресурсами школы целесообразно использовать показан на рисунке 8. Эта архитектура обеспечивает достаточную масштабируемость для системы, которая будет использоваться в небольшой школе. Кроме того, двухзвенная архитектура относительно проста в разработке и администрировании.

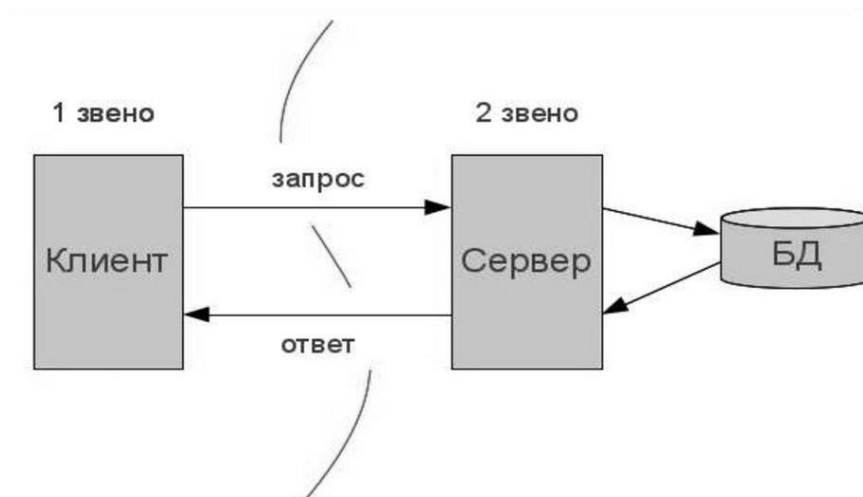


Рисунок 8 – Двухзвенная архитектура БД

### 3.2 Выбор технологии разработки программного обеспечения АИС

Код системы для обучающей системы можно написать в различных средах разработки, но IntelliJ IDEA Ultimate – это одна из популярных сред разработки, которая используется для написания кода на языках программирования Java, Kotlin, Groovy и т.д. IntelliJ IDEA Ultimate предоставляет множество инструментов для облегчения разработки, таких как автодополнение кода, проверка синтаксиса, отладка приложения и тестирование. Она также может интегрироваться с системами контроля версий, что делает совместную работу над проектом более удобной. Кроме того, IntelliJ IDEA Ultimate предоставляет возможность разработчикам работать с базами данных, используя различные плагины, такие как Database Tools and SQL,

которые позволяют подключаться к базе данных, выполнять запросы на получение данных и вносить изменения в базу данных. Это помогает разработчикам максимально использовать возможности базы данных для обучающей системы. Таким образом, при разработке обучающей системы на языке программирования, таком как Java, Kotlin или Groovy, использование среды разработки IntelliJ IDEA Ultimate может ускорить процесс разработки и обеспечить более эффективную работу с базой данных.

В качестве фреймворка будет использован Spring Boot – это фреймворк для Java, который позволяет быстро создавать, конфигурировать и запускать приложения на основе Spring, которые легко масштабируются и могут работать в любом окружении (cloud-native).

Spring Boot построен на основе фреймворка Spring, который содержит множество модулей и компонентов для разработки приложений на Java, таких как Spring Core, Spring MVC, Spring Data и т.д. Однако, Spring Boot упрощает и ускоряет разработку приложений, позволяя создавать каркас приложения, интегрируя конфигурацию, зависимости и выбранные технологии в одном компактном пакете. Spring Boot также обладает рядом преимуществ, таких как автоматическая конфигурация, которая позволяет уменьшить количество кода, необходимого для начала работы, встроенный сервер приложений, облегчающий процесс разработки и тестирования, удобный механизм управления зависимостями и поддержка множества баз данных и других технологий. Spring Boot подходит для разработки различных типов приложений, таких как веб-приложения, сервисы, микросервисы, приложения для обработки данных и т.д. Он также является основой для различных проектов в экосистеме Spring, таких как Spring Cloud и Spring Data.

Spring Boot работает на основе паттерна MVC (Model-View-Controller) — это один из наиболее популярных и часто используемых паттернов при проектировании веб-приложений. MVC представляет собой архитектурный паттерн, разделяющий приложение на три компонента: Model (модель), View (представление) и Controller (контроллер).

Model (модель) — ответственен за представление данных. Этот компонент обрабатывает данные и содержит бизнес-логику приложения. Он предоставляет интерфейс для работы с данными, но не знает о том, как эти данные будут отображаться в представлении. Например, модель может содержать данные из базы данных или файловой системы.

View (представление) — представление данных. Этот компонент отвечает за отображение данных на экране или их вывод на другое устройство, такое как принтер или мобильный телефон. Представление не знает о модели или контроллере и не обрабатывает данные. Оно просто отображает данные, полученные из модели.

Controller (контроллер) — связующее звено между моделью и представлением. Этот компонент обрабатывает запросы от пользователя, вызывает соответствующие методы модели для получения нужных данных и передает их в представление. Контроллер также управляет состоянием приложения. Он реагирует на действия пользователя и отправляет соответствующие данные в модель для их обработки.

В качестве СУБД будет использована база данных PostgreSQL. PostgreSQL позволяет управлять гигантскими базами данных, она очень надёжна, соответствует всем стандартам SQL и отлично справляется с выполнением сложных запросов.

Описанные компоненты работают в совместной работе и позволяют разделить каждую часть приложения на отдельные компоненты. Это позволяет ускорить разработку и облегчить поддержку приложения, так как при изменении одного из компонентов не придется переписывать всю систему. Паттерн позволяет более эффективно управлять данными и взаимодействием с пользователем. Spring Boot предоставляет реализацию MVC через Spring MVC, которая поддерживает множество фреймворков представления, таких как Thymeleaf, JSP, Velocity и т.д. Spring MVC также позволяет использовать аннотации для управления запросами и ответами.

Перед началом написания кода в Spring Boot необходимо выполнить следующие шаги:

1. Java Development Kit (JDK) – Spring Boot работает на языке Java, поэтому необходимо установить JDK, чтобы использовать его в разработке.

2. Установить среду разработки (IDE) – для разработки приложения в Spring Boot рекомендуется использовать интегрированную среду разработки, такую как IntelliJ IDEA, Eclipse или NetBeans. Установите IDE, настройте проект и установите необходимые плагины.

3. Создать новый проект – после настройки IDE нужно создать новый проект Spring Boot. Этот шаг можно выполнить, используя стартовые шаблоны, доступные в IDE, или использовать официальный сайт Spring Initializr.

4. Настроить зависимости – для работы приложения в Spring Boot необходимы некоторые зависимости, такие как Spring Web, Spring Security, Spring Data и другие. Необходимо добавить их в файл конфигурации проекта (например, pom.xml в Maven).

5. Добавить зависимость – для подключения к базе данных нужно добавить соответствующую зависимость в файл конфигурации проекта. Например, для работы с базой данных PostgreSQL, нужно добавить зависимость «postgresql» в файл pom.xml.

6. Настроить соединение с базой данных – после того, как добавили зависимость необходимо настроить соединение с базой данных, используя параметры, такие как url, username и password. Для этого нужно добавить соответствующие параметры в файл application.properties (или application.yml), который находится в ресурсах проекта.

Так как в системе будет происходить ролевое управление пользователями, то следует начать разработку приложения именно с прототипов авторизации и регистрации. На рисунке 9 представлен прототип входа в систему, что обеспечивает безопасность системы, которая проверяет валидность данных. В случае успешного входа, пользователь попадает на главную страницу, в противном случае получает соответствующее сообщение.

Также пользователь может сам зарегистрироваться в системе, вводя имя пользователя, электронную почту и пароль представлено на рисунке 10. Тестирование данных будет представлено в следующем разделе.

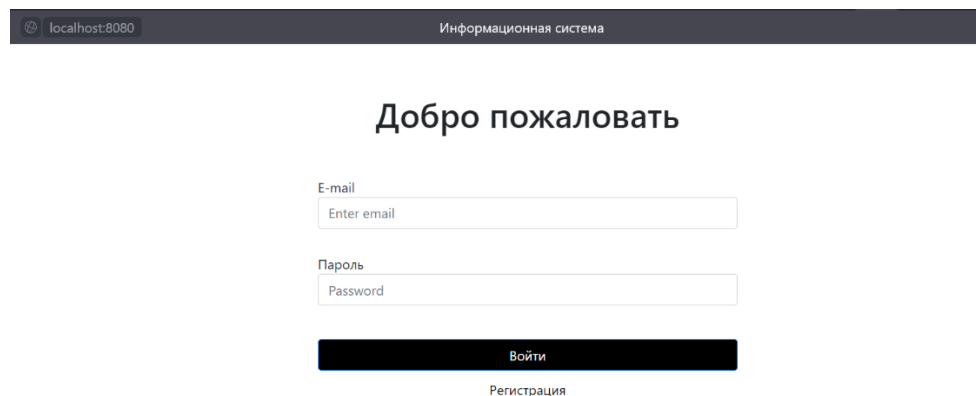


Рисунок 9 – Вход в систему

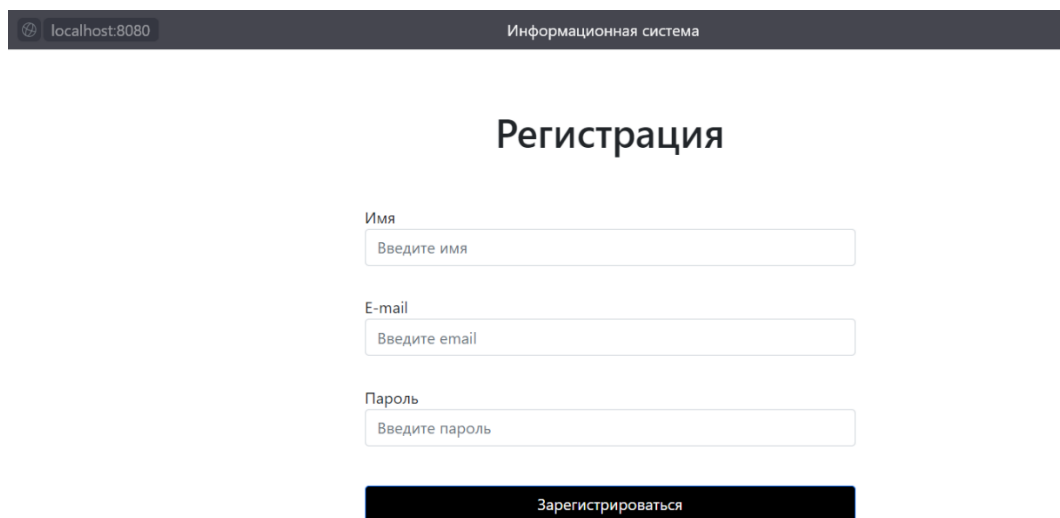


Рисунок 10 – Регистрация пользователя

Поясним некоторые моменты работы логики приложения. В контроллере должен быть указан следующий код:

```
@RequestMapping("/login")
public String home(Model model, HttpSession session) {
    session.invalidate();
    return "Autorization";
}
```

Контроллер, указанный аннотацией `@RequestMapping("/login")`, отвечает за обработку GET запросов по адресу `/login`. Этот контроллер обычно используется для отображения страницы веб-приложения.

Аргументы метода `home`:

`Model model` – Этот параметр представляет модель данных, которые будут переданы в вид. Модель содержит атрибуты, которые могут быть использованы для отображения данных в виде.

`HttpSession session` – Этот параметр представляет текущую сессию пользователя, которая может быть использована для хранения данных на сервере между запросами. В этом случае, вызов `session.invalidate()` удаляет все данные из текущей сессии.

Метод `home` возвращает имя представления, которое должно быть использовано для отображения страницы приложения. В данном случае, возвращается имя `"Authorization"`, которое указывает на шаблон для отображения страницы авторизации.

`RequestMapping` – это аннотация, которая связывает URL-адрес с методом контроллера. Она может использоваться с различными параметрами, такими как метод запроса (GET, POST, PUT, DELETE), заголовки запроса и т.д. В приведенном контроллере, аннотация `@RequestMapping("/login")` указывает на то, что метод `home` вызывается при обращении по URL `/login`.

Шаблон страницы авторизации написан на языке разметки `html` и имеет следующий вид:

```
<body>
<section class="main-contents">
  <div class="content">
    <h1 style="text-align: center; margin-top: 50px;">Добро
пожаловать</h1>
    <div id="login-form">
      <form action="/login" method="post">
        <div class="form-groups">
          <label for="exampleInputEmail1">E-mail</label>
          <input type="email" class="form-control"
id="exampleInputEmail1" aria-describedby="emailHelp" placeholder="Enter email"
name="email">
        </div>
        <div class="form-groups">
          <label for="exampleInputPassword1">Пароль</label>
          <input type="password" class="form-control"
id="exampleInputPassword1" placeholder="Password" name="password">
      </form>
    </div>
  </div>
</section>
</body>
```

```

        </div>
        <button type="submit" class="btn btn-primary">Войти</button>
        <button type="submit" class="btn btn-primary">Войти как
гость</button>
        <a class="regis" href="/regis">Регистрация </a>
    </form>
</div>
</div>
</section>
</body>

```

Form action "/login" означает, что вводимые пользователем данные из этой формы будут отправлены по адресу /login на сервер. При отправке формы происходит запрос на указанный адрес, и на сервер отсылается все данные, указанные в форме, для обработки. В контроллере, который отвечает за этот запрос (в данном случае, это контроллер, связанный с URL-адресом /login), обрабатываются эти данные и выполняются необходимые действия, например, проверка аутентификации пользователя. В данном примере метод запроса указан явно как "POST", что означает, что данные формы отправляются на сервер в теле запроса, а не в URL-адресе.

Таким образом, данный код отображает только шаблон HTML-страницы, чтобы обработать логику авторизации, необходим другой запрос:

```

    @RequestMapping("/index")
    public String homeWithSession(Model model, HttpSession session) {
        Product product = new Product();
        ArrayList<Product> products = product.getProduct();
        model.addAttribute("products", products);
        Integer userRole = User.getUserRole((String)
session.getAttribute("userEmail"));
        model.addAttribute("isManager", userRole == 1);
        return "index";
    }

    @PostMapping("/login")
    public String login(@RequestParam String email, @RequestParam String
password, HttpSession session) {
        if (User.checkUser(email, password)) {
            // если пользователь найден, сохранить информацию о пользователе
в сессии
            session.setAttribute("userEmail", email);
            // перенаправить на главную страницу
            return "redirect:/index";
        } else {
            // если пользователь не найден, перенаправить обратно на страницу
авторизации с сообщением об ошибке
            return "Autorization";
        }
    }
}

```

@PostMapping("/login") отвечает за обработку POST запросов по адресу /login, которые отправляются при входе в систему. В контроллере проверяется, есть ли пользователь с таким электронным адресом и паролем в базе данных. Если пользователь найден, информация о нем сохраняется в сессии. Затем пользователь перенаправляется на главную страницу приложения, которая отображается с использованием контроллера @RequestMapping("/index"). Если пользователь не найден, он перенаправляется обратно на страницу авторизации, где отображается сообщение об ошибке.


Для каждой сущности в базе данных должен быть создан свой класс в модели, с параметрами, которые относятся к полям базы данных, с геттерами и сеттерами, например, класс «Пользователи», который является сущностью таблицы Users. Этот класс описывает модель пользователя в приложении. В классе определены поля, которые соответствуют свойствам пользователя: имя, электронный адрес, пароль, роль (администратор, менеджер, гость) и идентификатор пользователя. Также в классе определены методы для получения пользователя по его идентификатору, проверки наличия пользователя с заданным электронным адресом и паролем, получения роли пользователя по его электронному адресу. В контроллере только создается объект модели и вызывается необходимый метод.

Главная страница системы с точки зрения интерфейса учителя представлена на рисунке 11. На главной странице каждый из пользователей видит список курсов, который он может пройти. учитель имеет возможность редактирования, добавления и удаления курса. На рисунке 12 представлен прототип формы редактирования курса. Формы и страницы для подразделов и материалов идентичны формам создания курса.






	ГЛАВНАЯ	РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ	РАСПИСАНИЕ	ВЫЙТИ
--	---------	-------------------------	------------	-------

**Добавить раздел**






**Линейное программирование**  
Учебное пособие

**Пройти курс**  



**Математика. Задачи**  
Вводный курс

**Пройти курс**  



**Неделя Русского языка**

Неделя Русского языка



**Пройти курс**  

Рисунок 11 – Главная страница

	ГЛАВНАЯ	РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ	РАСПИСАНИЕ	ВЫЙТИ
--	---------	-------------------------	------------	-------

Название раздела

Программирование на Python

Описание раздела

|

Изображение

Выбор файла Не выбран ни один файл

**Добавить раздел**

Рисунок 12 – Добавление раздела

Помимо курсов и подразделов, в системе должны быть предусмотрены задания, учитель может их прикрепить как отдельным файлом, так и создать тесты для учеников. Прототип интерфейса теста представлен на рисунке 13. Тесты создаются внутри раздела, т.к. в одном курсе может быть несколько разделов или тем.

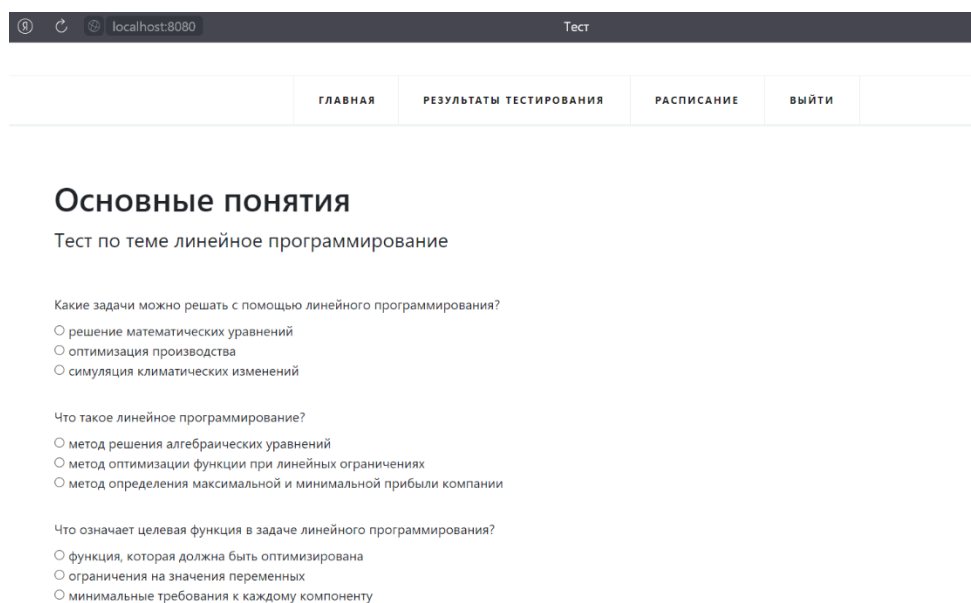


Рисунок 13 – Прототип теста

В функционал учителя должна быть добавлена также возможность просмотра выполненных тестов и заданий с возможностью редактирования показано на рисунке 14. Учитель видит все результаты учеников, когда сам ученик видит только свои результаты. Аналогичный функционал имеет страница с расписанием представлен на рисунке 15.

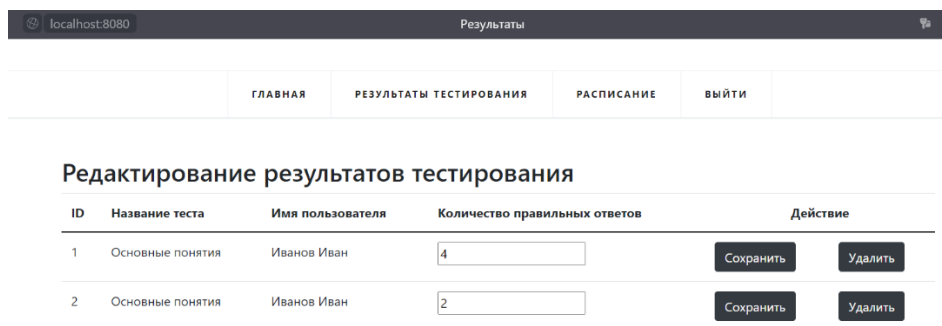


Рисунок 14 – Редактирование результатов тестирования

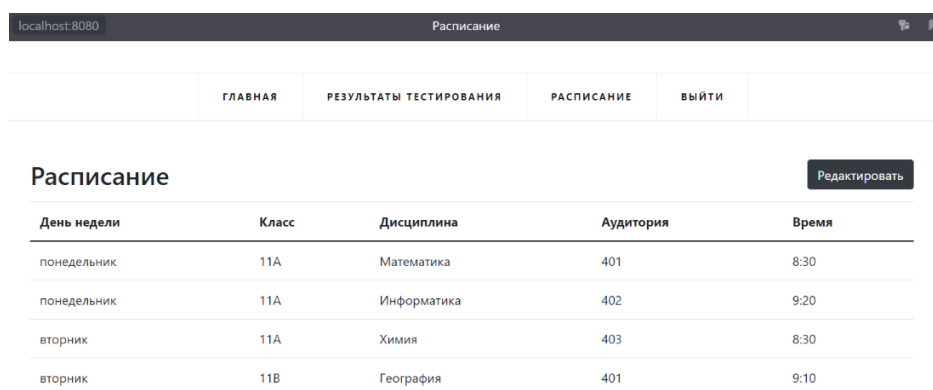


Рисунок 15 – Страница расписания

### 3.3 Тестирование функций

1. Добавление курса (Рис. 16 – 18). В данном случае срабатывает инструкция INSERT в таблицу Course Sections.

Название раздела

Описание раздела

Изображение

Рисунок 16 – Добавление курса

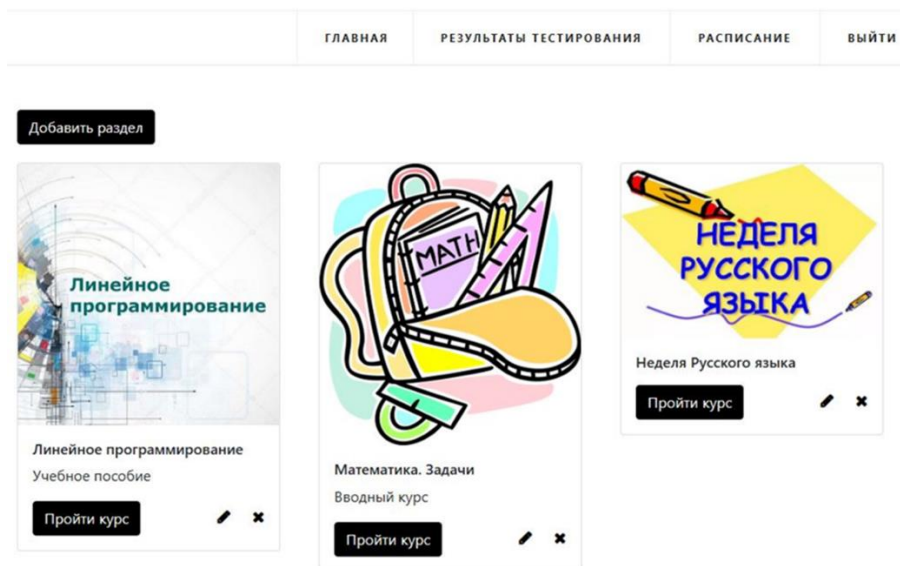


Рисунок 17 – Результат добавления курса в интерфейсе

```

1 SELECT * FROM public."Course Sections"
2 ORDER BY id ASC

```

	id [PK] integer	section_name character varying (100)	section_description text	img character varying
1	1	Линейное программирование	Учебное пособие	img/one.png
2	2	Теория игр	Учебное пособие	img/two.png
3	3	Специальные задачи и методы	Учебное пособие	img/three.png
4	4	Пустой раздел	Пустой раздел	img/plus.png
5	5	Тестовый курс	Тестовый курс	img/one.png

Рисунок 18 – Результат добавления курса в БД

## 2. Валидация данных

На рисунках 19 – 20 представлена таблица пользователей с логинами и паролями, зная эту информацию можно войти в систему.

```

1 SELECT * FROM public."Users"
2 ORDER BY id ASC

```

id	name	email	password	role	class
1	Иванов Иван	ivan2000@mail.ru	12345	2	1
2	Администратор	admin@mail.ru	12345	1	[null]
3	Дмитрий Петров	dmitr@mail.ru	323223	2	2

Рисунок 19 – Таблица пользователей

## Добро пожаловать

E-mail

Пароль

[Войти](#)

[Регистрация](#)

Рисунок 20 – Ввод данных в поля авторизации

В результате нескольких тестирований можно сделать вывод о том, что система находится в работоспособном состоянии и готова к дальнейшей эксплуатации.

### 3.4 Инструкция пользователям

- Зарегистрируйтесь в системе, если вы еще не зарегистрированы. Для этого, нажмите на кнопку «Зарегистрироваться» и введите свое имя, почту и e-mail. После этого, администратор системы сделает вам роль учителя.
- После регистрации, войдите в систему, введя свой e-mail и пароль.

– На главной странице вы увидите список курсов, рядом с которыми есть кнопки «Редактировать» и «Удалить». Чтобы отредактировать курс, нажмите на кнопку «Редактировать» и заполните форму с названием курса, описанием и изображением.

– После создания курса, нажмите на кнопку «Пройти курс» и вы увидите список тем для соответствующего курса. Здесь также есть кнопки «Редактировать» и «Удалить». Чтобы отредактировать тему, нажмите на кнопку «Редактировать» и заполните форму с названием темы и прикрепленными материалами.

– У тем также есть тестирования, которые проходят ученики. Результаты тестирования можно увидеть на странице «Результаты тестов». Здесь вы можете просмотреть результаты всех учеников и изменить или удалить их.

– На странице «Расписание» можно просмотреть все расписания учеников и изменить любое поле в таблице.

– Чтобы выйти из системы, нажмите на кнопку «Выход»

– Как учащийся, вы должны пройти процедуру регистрации для доступа к системе. Для этого на основном интерфейсе вам необходимо найти и нажать кнопку с надписью «Регистрация». Впоследствии вам будет предложено ввести такие данные, как ваше полное имя, предпочтительный почтовый адрес и действующий адрес электронной почты.

– После завершения регистрации вы получаете возможность подтвердить свою личность. Это делается путем ввода адреса электронной почты и пароля, которые вы ранее установили, непосредственно в главном интерфейсе.

– Главный интерфейс служит доступом к множеству курсов, отображаемых в виде подробного списка. Если вас интересует конкретный курс, вы можете записаться, найдя и нажав кнопку «Пройти курс» рядом с предполагаемым курсом.

- Некоторые темы могут содержать тесты, предназначенные для оценки вашего понимания. Чтобы попробовать эти испытания, найдите и нажмите кнопку с надписью «Пройти тест».
- Ваши результаты в этих тестах можно просмотреть на отдельной странице под названием «Результаты тестов». На этой странице показаны ваши баллы по каждой теме, по которой вас оценивали.
- Существует специальная страница под названием «Расписание», где вы можете просмотреть расписания занятий, которые касаются исключительно вас.
- Для выхода из системы найдите и нажмите кнопку «Выход», расположенную на главном интерфейсе.
- Чтобы оптимизировать взаимодействие с системой, настоятельно рекомендуется ознакомиться с инструкциями и тщательно использовать каждую функцию, имеющуюся в вашем распоряжении.

#### Выводы по третьей главе

В современную эпоху образования создание современной информационной системы для контроля и управления образовательными ресурсами в школе имеет первостепенное значение. Этот этап образовательного пути является основополагающим и влечет за собой всестороннее описание технологических рабочих процессов, лежащих в основе работы информационной системы, тщательное планирование физической структуры базы данных, а также определение ее функциональных модулей.

Включение такой информационной системы управления ресурсами в деятельность школы может значительно повысить уровень производительности учебного заведения. Эта система призвана значительно упростить учебный процесс, тем самым оптимизируя всю работу.

## Заключение

Принятие необходимых мер по созданию информационной системы, специально предназначенной для управления образовательными ресурсами школы, является неизбежным шагом в современном образовательном ландшафте. Эта система призвана экономить время и денежные ресурсы. Школы стремятся улучшить качество предлагаемого ими образования и повысить свою общую эффективность с помощью этой системы.

Учебная платформа, созданная с использованием среды Spring Boot, служит высокоэффективным инструментом администрирования образовательных ресурсов. В этой системе полный список материалов предлагает пользователям всеобъемлющий набор ресурсов, необходимых для приобретения новых навыков и знаний. Состав системы отличается гибкой ролевой структурой, предоставляющей администраторам и пользователям различную степень доступа к системе. Этот уникальный атрибут помогает упростить системное администрирование, обеспечивая при этом безопасность системы. Учебная платформа также предоставляет серию тестов, предназначенных для оценки знаний пользователей, включающих ряд вопросов и возможных ответов. После успешного завершения этих тестов пользователям отображаются соответствующие результаты, отображаемые непосредственно на их пользовательских страницах. Использование передовых технологий и инновационных подходов при разработке этой системы привело к повышению эффективности. Это обеспечивает быстрое и легкое обучение пользователей, а администрирование системы требует минимальных усилий со стороны администраторов. Подводя итог, можно сказать, что эта система дает пользователям свободу учиться в удобном для них темпе, одновременно предоставляя преподавателям и администраторам возможность управлять системой с максимальной легкостью и эффективностью.



## Список используемой литературы и используемых источников

1. Хомоненко А.Д. Базы данных: Учебник для высших учебных заведений СПб.: КОРОНА принт, 2009.— 672 с.
2. Редько В. Н. Базы данных и информационные системы / В.Н. Редько, И.А. Басараб. - М.: Знание, 2017. - 341 с.
3. Казаченок Н.Н. Производственная практика: учебно-методическое пособие / Н. Н. Казаченок, О. П. Михеева. — Тольятти: ТГУ, 2018. — 50 с. — ISBN 978-5-8259-1389-6.
4. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем: учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — М.: Издательство Юрайт, 2019. — 432 с.
5. Панюкова, Т.А. Проектирование программных средств. / Т.А. Панюкова. – М.: КД Либроком, 2012. 362 с.
6. Васильев, А. Java. Объектно-ориентированное программирование: Учебное пособие Стандарт третьего поколения / А. Васильев. – СПб.: Питер, 2013. 400 с.
7. Суркова Н. Е. Методология структурного проектирования информационных систем. Красноярск: Научно-инновационный центр, 2014. 190 с.
8. Фельдман Я.А. Создаем информационную систему / Я.А. Фельдман. – М.: Солон-Пресс, 2007. – 120 с.
9. Криницкий Н.А., Миронов, Г.А., Фролов, Г.Д. Автоматизированные информационные системы. – М.: Наука, 2017. – 382 с.
10. Коваленко, В.В. Проектирование информационных систем: учеб. пособие / В.В. Коваленко. – 2-е изд. – Москва: ИНФРА-М, 2021. – 357 с.

11. Беляева И. В. Архитектура информационных систем. УлГТУ, 2019. 192 с.
12. Мкртычев С.В., Гущина О.М., Очеповский А.В. Прикладная информатика. Бакалаврская работа [Электронный ресурс] : электрон. Учеб-метод. пособие. Тольятти. ТГУ: Изд-во ТГУ, 2019. URL: <https://dspace.tltsu.ru/handle/123456789/8868> (дата обращения: 10.09.2023).
13. Чистов, К.В. Проектирование информационных систем. Учебник и практикум для академического бакалавриата / К.В. Чистов. – М: Юрайт, 2016. – 258 с.
14. Фуфаев, Д.Э., Разработка и эксплуатация автоматизированных информационных систем. – Академия, 2016. – 304 с.
15. Ерохин, В.В., Прогонышева Д. А. Безопасность информационных систем: учебное пособие/В.В. Ерохин, Д.А. Погонышева. – Флинта, 2015. – 184 с.
16. Пирогов, В.Ю. Информационные системы и базы данных: организация и проектирование. / В.Ю. Пирогов – СПб: БХВ-Петербург, 2009. – 528 с.
17. Грабер, М. Введение в SQL. / М. Грабер – Москва: Лори, 1996. – 375 с.
18. Голицина О.Л. Базы данных: Учебное пособие / О.Л. Голицина. – М.: ФОРУМ: ИНФРА-М, 2015. – 352 с.
19. Маркин А. В. Web-программирование: учебник / А. В. Маркин. — Москва: Ай Пи Ар Медиа, 2021. — 286 с.
20. Илюшечкин В. М. Основы использования и проектирования баз данных. Учебник / В.М. Илюшечкин. - М.: Юрайт, 2015. - 214 с.
21. Грекул В.И. Проектирование информационных систем: курс лекций/ В.И. Грекул. – М.: Интернет-Ун-т Информ технологий, 2016. – 304 с.

22. Золотов С.Ю. Проектирование информационных систем: учебное пособие / Золотов С.Ю.. — Томск: Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2013. – 88 с.
23. Дейт К. Дж. Введение в системы баз данных – Вильямс, 2005. – 1316 с.
24. Database Design and Relational Theory: Normal Forms and All That Jazz Paperback, O'Reilly Media, 2019. 472 p.
25. Sasaki B., Chao J., Howard R.. Graph Databases For Beginners: Self-publishing, 2018.