



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Экономики и управления на предприятии природопользования»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(бакалаврская работа)
по направлению подготовки 09.03.03 Прикладная информатика
(квалификация – бакалавр)

На тему «Разработка информационного приложения виртуальной лаборатории»

Исполнитель Чакаев Евгений Анатольевич

Руководитель к.т.н., доцент по кафедре Информатики и прикладной математики
Полупанов Владимир Николаевич

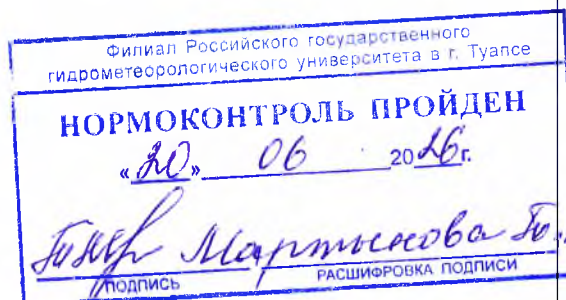
«К защите допускаю»

Руководитель кафедры _____

кандидат экономических наук

Майборода Евгений Викторович

«22» 06 2026 г.



Туапсе
2026

ОГЛАВЛЕНИЕ

Введение	3
1. Аналитический раздел	6
1.1 Анализ предметной области	6
1.2 Обоснование выбора задачи	9
1.3 Педагогическая и экономико–информационная сущность задачи.....	12
2. Проектный раздел	16
2.1 Информационное обеспечение задачи.....	16
2.1.1 Архитектура данных и модель предметной области.....	16
2.1.2 Взаимосвязи сущностей и целостность данных	19
2.1.3 Доступ к данным	20
2.2 Технологическое обеспечение.....	22
2.2.1 Обоснование проектных решений по технологическому обеспечению	22
2.2.2 Технологические операции	24
2.3 Техническое обеспечение.....	26
2.4 Архитектура и дизайн программного обеспечения.....	29
2.4.1 Backend–архитектура.....	32
2.4.2 Frontend–архитектура	34
2.5 Реализация основных модулей приложения	37
2.6 Руководство пользователя.....	46
2.6.1 Описание интерфейса	46
2.6.2 Порядок работы с приложением	47
3. Обоснование экономической эффективности результатов ВКР	50
3.1 Выбор и обоснование методики расчета экономической эффективности.....	50
3.2 Расчет показателей экономической эффективности	51
Заключение	53
Список литературы	56

Введение

В современном мире цифровизация образования является одним из ключевых факторов повышения качества и доступности учебного процесса в высших учебных заведениях[3;4;21]. Особенно остро этот процесс проявляется в преподавании естественных наук, где традиционные лабораторные работы требуют значительных материальных и временных затрат, а также связаны с определёнными рисками для здоровья студентов.

Традиционные лабораторные работы по химии, в частности по разделу качественного анализа (определение катионов и анионов по характерным реакциям), имеют ряд существенных недостатков. К ним относятся высокая стоимость реактивов и лабораторного оборудования, необходимость строгого соблюдения техники безопасности при работе с токсичными, едкими и пожароопасными веществами, ограниченное время на проведение и повторение экспериментов, а также сложность организации лабораторных занятий в условиях дистанционного или смешанного обучения[5;12;22]. Кроме того, в школах и при большом потоке учеников часто возникают проблемы с обеспечением каждого учащегося необходимыми реактивами и приборами.

Виртуальная лаборатория позволяет существенно снизить указанные проблемы. Она предоставляет студентам возможность проводить химические эксперименты в интерактивной цифровой среде, наблюдать образование осадков, изменение окраски растворов, газовыделение и другие характерные явления без реальных химических веществ.

Подобные приложения помогают учебным заведениям обезопасить процесс обучения и снижают материальные затраты на оборудование, позволяя многократно проводить эксперименты, свободно меняя условия и список реагентов.

Актуальность темы продиктована потребностью учебных заведений в современных цифровых инструментах для проведения лабораторных работ по химии. Важность разработки подобных решений также может быть

обусловлена переходом на гибридные формы обучения и ограниченностью бюджетов учебных заведений.

Объектом исследования является процесс проведения лабораторных работ по качественному анализу в курсе неорганической химии.

Предметом исследования выступает информационное приложение виртуальной лаборатории для автоматизированной симуляции качественных химических реакций.

Цель выпускной квалификационной работы – разработка браузерного приложения виртуальной лаборатории по качественному анализу, обеспечивающего интерактивную симуляцию химических реакций, удобный пользовательский интерфейс для студентов и преподавателей, а также хранение результатов выполненных работ.

Для достижения поставленной цели необходимо решить следующие задачи:

Провести анализ предметной области виртуальных лабораторий по химии и существующих решений в данной сфере.

Обосновать выбор задачи и описать педагогическую и экономико–информационную сущность разрабатываемого приложения.

Разработать информационное, технологическое, техническое и программное обеспечение виртуальной лаборатории.

Создать удобный интерфейс приложения и руководство пользователя.

Рассчитать экономическую эффективность внедрения разработанного приложения в учебный процесс.

Методы исследования включают анализ научно–методической литературы, сравнительный анализ существующих виртуальных лабораторий, системный анализ, имитационное моделирование химических процессов, объектно–ориентированное программирование, а также методы экономического анализа эффективности.

Теоретическая основа работы опирается на исследования в области цифровых образовательных технологий, виртуального моделирования

химических экспериментов и разработки информационных систем в образовании.

Практическая значимость результатов заключается в возможности использования разработанного браузерного приложения для проведения лабораторных работ по качественному анализу в очном, дистанционном и смешанном форматах обучения. Приложение может быть внедрено в образовательный процесс школ и колледжей, что позволит снизить затраты на проведение лабораторных работ и повысить качество подготовки студентов.

Структура выпускной квалификационной работы соответствует требованиям методических рекомендаций и включает введение, аналитический раздел, проектный раздел, раздел обоснования экономической эффективности, заключение и список литературы.

1. Аналитический раздел

1.1 Анализ предметной области

Предметная область исследования – проведение лабораторных работ по качественному анализу в курсе неорганической химии с использованием современных информационных технологий.

Важнейшим разделом неорганической химии в начале обучения является качественный анализ. Он подразумевает определение катионов и анионов в растворах по определенным видимым признакам: изменению цвета раствора, выделению газа, выпадению осадка и другим признакам.

В школах и других учебных заведениях проведение лабораторных работ по химии связано со значительными затратами: закупкой реактивов, обеспечением и соблюдением мер безопасности, закупкой оборудования и уборкой рабочего места с утилизацией реактивов. Также в качестве недостатка стоит выделить ограниченное количество попыток выполнения эксперимента для каждого студента, это может снижать качество обучения.

Благодаря цифровизации появился новый класс учебных инструментов. Виртуальные лаборатории позволяют имитировать реальные химические реакции, не имея затрат на оборудование и реактивы.

Студент может добавлять реагенты в пробирки, наблюдать протекание реакций в реальном времени, фиксировать наблюдения и делать выводы о составе неизвестного вещества.

На сегодняшний день на рынке образовательных технологий существует достаточно большое количество виртуальных лабораторий по химии. Среди наиболее известных зарубежных решений можно выделить:

- PhET Interactive Simulations (University of Colorado) – набор бесплатных интерактивных симуляций, включая некоторые химические опыты[6];
- Labster– коммерческая платформа с высококачественной 3D–графикой и виртуальной реальностью[7];

- ChemCollective (CarnegieMellon University) – виртуальные лаборатории с акцентом на качественный и количественный анализ[8];
- BeyondLabz– полноценная виртуальная лаборатория с широким набором органических и неорганических экспериментов[25].

В России и странах СНГ также активно разрабатываются и используются виртуальные лаборатории[10;11]:

- Платформа «Виртуальная лаборатория» на базе образовательных порталов[13];
- Решения на основе Unity и WebGL, разрабатываемые отдельными вузами;
- Различные веб–приложения и мобильные приложения для школьного и вузовского уровней.

Более глубокий анализ существующих на рынке решений выявил следующие проблемы:

- Большинство приложений имеют упрощённый уровень детализации реакций;
- Коммерческие платформы имеют ограниченную доступность;
- Определенное количество решений имеют слабую реализацию визуальной части (статичные картинки вместо анимаций);
- Во многих приложениях отсутствует возможность сохранить результаты экспериментов студентов, также отсутствует привязка к пользователю;
- Некоторые платформы также требуют установки дополнительного ПО для своей работы, что делает их менее адаптивными для разных видов устройств и операционных систем

Важным недостатком также является слабая проработка качественных реакций. Большая часть существующих решений позволяет лишь выбрать готовый результат реакции без самостоятельного эксперимента.

Это существенно снижает развивающий эффект лабораторной работы и не позволяет в полной мере сформировать практические навыки аналитической

химии.

Таблица 1.1 – Сравнение существующих виртуальных лабораторий по химии

Система / Платформ а	Уровень детализа ции реакций	Визуализ ация (анимаци и)	Сохране ние результата тов	Адаптивн ость (браузер)	Доступн ость для вузов и школ	Поддержк а качествен ного анализа
PhET Interactive	Средний	Хорошая	Нет	Высокая	Бесплатн о	Частичная
Labster	Высокий	Отличная (3D)	Да	Средняя	Платная	Полная
ChemColle ctive	Высокий	Хорошая	Частичн о	Высокая	Бесплатн о	Хорошая
BeyondLab z	Высокий	Отличная	Да	Средняя	Платная	Полная
Российски е разработки	Средний	Средняя	Частичн о	Разная	Разная	Частичная

Анализ таблицы показывает, что на рынке практически отсутствует бесплатное или условно–бесплатное браузерное приложение, специально ориентированное на качественный анализ неорганических веществ с реалистичной визуализацией реакций и системой сохранения результатов для преподавателя[24].

Разработка собственного браузерного приложения виртуальной лаборатории позволит устранить указанные недостатки[5]. Приложение будет работать непосредственно в веб–браузере без установки дополнительного ПО, обладать достаточной интерактивностью для моделирования последовательного добавления реагентов, обеспечивать качественную анимацию наблюдаемых эффектов (выпадение осадка, изменение цвета,

выделение газа) и сохранять историю экспериментов каждого студента[12;15].

Таким образом, анализ предметной области подтверждает актуальность и практическую необходимость разработки специализированного браузерного приложения виртуальной лаборатории по качественному анализу в химии[30].

1.2 Обоснование выбора задачи

Выбор темы выпускной квалификационной работы «Разработка информационного приложения виртуальной лаборатории» (на примере качественного анализа в химии) обусловлен сочетанием высокой практической актуальности, наличием нерешённых проблем в предметной области и возможностью достижения значимого педагогического и экономического эффекта при относительно умеренных затратах на разработку.

В настоящее время при проведении лабораторных работ по качественному анализу сохраняется ряд существенных проблем[1]:

- Высокая стоимость проведения лабораторных работ. Закупка реактивов, лабораторной посуды и расходных материалов требует значительных ежегодных затрат со стороны образовательного учреждения[5].
- Риски для здоровья студентов и преподавателей. Работа с концентрированными кислотами, щелочами, солями тяжёлых металлов и другими опасными веществами требует строгого соблюдения техники безопасности и постоянного контроля со стороны преподавателя[12].
- Ограниченные возможности повторения экспериментов. Каждый студент имеет ограниченное количество попыток выполнения работы, что снижает качество усвоения материала и не позволяет в полной мере отработать навыки аналитического мышления.
- Сложности организации лабораторных занятий в школах и при дистанционном обучении. Географическая распределённость студентов и отсутствие оборудованных лабораторий делают традиционную форму проведения работ практически невозможной.

- Отсутствие удобной системы фиксации и проверки результатов. Преподаватель вынужден проверять бумажные журналы или отчёты студентов, что увеличивает административную нагрузку и снижает оперативность обратной связи.

Разрабатываемое приложение направлено на решение именно этих проблем за счёт создания современного браузерного инструмента, который:

- Позволяет студентам проводить виртуальные эксперименты по качественному анализу в любое время и с любого устройства, имеющего доступ в интернет;
- Обеспечивает реалистичную визуализацию химических реакций (образование осадков различной окраски, изменение цвета растворов, газовыделение);
- Даёт возможность многократно повторять эксперимент, самостоятельно выбирая последовательность добавления реагентов;
- Автоматически сохраняет результаты каждой лабораторной работы с привязкой к конкретному студенту;
- Позволяет преподавателю просматривать историю выполненных работ и оценивать прогресс студентов, проводить контроль посещаемости и активности, а также формировать лабораторные работы с методическими указаниями;
- Существенно снижает материальные затраты учебных заведений на проведение лабораторных работ и повышает безопасность учебного процесса.

Критерии выбора именно этой задачи:

1. Соответствие профилю подготовки. Тема полностью соответствует направлению подготовки, связанному с разработкой информационных систем и приложений, и включает все основные этапы: анализ, проектирование, реализацию и оценку эффективности.

2. Высокая практическая значимость. Виртуальная лаборатория решает реальную проблему образовательного процесса в химии и может быть использована как в очном, так и в дистанционном формате обучения.

3. Реализуемость в рамках ВКР. Задача имеет чёткие границы: не требует создания сложной физической симуляции, но позволяет продемонстрировать навыки разработки полноценного веб–приложения с интерактивной графикой, базой данных и удобным интерфейсом.

4. Перспектива внедрения. Разработанное приложение может быть внедрено в учебный процесс школ и других учебных заведений без значительных дополнительных затрат.

5. Экономическая целесообразность. Предварительная оценка показывает, что затраты на разработку и внедрение окупаются за счёт существенного сокращения расходов на реактивы, оборудование и организацию лабораторных занятий. Подробный расчёт будет представлен в разделе 3 выпускной квалификационной работы.

Для более глубокого обоснования выбора задачи был проведён SWOT–анализ

Таблица 1.2 – SWOT–анализ внедрения виртуальной лаборатории по химии

Strengths (Сильные стороны)	Weaknesses (Слабые стороны)	Opportunities (Возможности)	Threats (Угрозы)
Высокий уровень безопасности экспериментов	Необходимость разработки качественной визуализации	Расширение на другие разделы химии	Сопротивление традиционно мыслящих преподавателей
Снижение затрат на реактивы и оборудование	Зависимость от качества интернет–соединения	Интеграция с системами дистанционного обучения	Возможные технические проблемы у студентов с устаревшими устройствами

Продолжение таблицы 1.2

Возможность многократного повторения опытов	Ограниченная точность моделирования некоторых реакций	Создание банка лабораторных работ для всего курса	—
Удобство проверки результатов преподавателем	Требуется время на освоение приложения	Использование в школах и колледжах	—

SWOT–анализ показывает, что сильные стороны и возможности значительно перевешивают слабые стороны и угрозы. При грамотной реализации приложение способно существенно улучшить качество проведения лабораторных работ по химии.

Таким образом, выбранная задача является оптимальной: она решает актуальную проблему образовательного процесса, соответствует требованиям образовательных стандартов и позволяет в полной мере продемонстрировать профессиональные компетенции в области разработки информационных приложений.

1.3 Педагогическая и экономико–информационная сущность задачи

Экономико–информационная сущность задачи заключается в автоматизации и цифровизации процесса проведения лабораторных работ по качественному анализу в химии путём создания интерактивного браузерного приложения, которое позволяет студентам безопасно и эффективно осваивать практические навыки без использования реальных химических реактивов.

Педагогическая сущность задачи

Качественный анализ является фундаментальным разделом

неорганической химии, направленным на формирование у студентов навыков аналитического мышления, наблюдательности и умения делать правильные выводы на основе визуальных признаков химических реакций (образование осадка, изменение окраски, выделение газа и др.)[15].

Виртуальная лаборатория решает следующие педагогические задачи[24]:

- Обеспечение безопасного проведения экспериментов с веществами, работа с которыми в реальной лаборатории требует повышенных мер предосторожности.
- Создание возможности многократного повторения одного и того же эксперимента с варьированием условий и последовательности добавления реагентов.
- Развитие навыков самостоятельной работы: студент сам выбирает реагенты, наблюдает реакцию и фиксирует результаты в электронном журнале.
- Формирование понимания систематического хода качественного анализа (групповое разделение катионов и анионов).
- Повышение мотивации студентов за счёт современного интерактивного формата обучения и мгновенной визуальной обратной связи[22].
- Возможность индивидуализации обучения: каждый студент может работать в своём темпе и повторять сложные моменты.

Приложение позволяет реализовать деятельностный подход в обучении, когда студент не просто наблюдает за готовой демонстрацией, а активно взаимодействует с виртуальной средой, моделируя реальный лабораторный процесс.

Информационная сущность задачи

Задача относится к классу информационных систем оперативного и аналитического уровня, обеспечивающих моделирование, хранение и обработку данных о проведённых виртуальных экспериментах.

Основные информационные потоки включают:

Входные данные:

- Данные о пользователе (студент / преподаватель);
- Список доступных лабораторных работ;
- Набор реагентов и ионов для каждой лабораторной работы;
- Параметры эксперимента (выбранный неизвестный раствор, последовательность добавления реагентов).

Выходные данные:

- Визуализация протекания химической реакции в реальном времени;
- Электронный журнал наблюдений студента;
- Сохранённый результат лабораторной работы;
- Отчёты для преподавателя (список выполненных работ, качество выполнения, частые ошибки студентов).

Внутренние процессы обработки:

- Симуляция химических процессов во время реакций происходят на стороне бэкенда с помощью библиотеки Chem.js
- Расчёт и отображение визуальных эффектов (анимация выпадения осадка, изменение цвета раствора, появление пузырьков газа);
- Автоматическая фиксация последовательности действий студента;
- Хранение результатов в базе данных с привязкой к пользователю и лабораторной работе.

Основой системы является база данных, содержащая информацию о пользователях, лабораторных работах, реагентах, возможных реакциях и сохранённых экспериментах студентов.

Экономическая сущность задачи

Информация, получаемая в результате использования виртуальной лаборатории, обладает высокой экономической ценностью, поскольку напрямую влияет на эффективность учебного процесса и финансовые затраты образовательного учреждения.

Основные экономические эффекты[4;14;21]:

- Существенное снижение затрат на закупку реактивов, лабораторной посуды и расходных материалов;

- Снижение расходов на обеспечение техники безопасности и утилизацию химических отходов;
- Уменьшение временных затрат преподавателей на подготовку и проведение лабораторных занятий;
- Возможность проведения лабораторных работ в больших потоках студентов без увеличения количества аудиторий и оборудования;
- Повышение качества подготовки студентов, что косвенно влияет на репутацию учебных заведений и результаты аккредитации.

Таблица 1.3 – Оценка экономической значимости основных информационных объектов

Информационный объект	Экономический эффект от автоматизации	Примерная количественная оценка
Виртуальные реакции и визуализация	Снижение затрат на реактивы и оборудование	Экономия до 70–90 % от бюджета на лабораторные работы
Сохранённые результаты экспериментов	Снижение трудозатрат преподавателя на проверку работ	Сокращение времени проверки на 40–60 %
История выполненных работ	Возможность анализа типичных ошибок студентов	Повышение качества обучения, снижение пересдач
Электронный журнал наблюдений	Упрощение документооборота	Снижение бумажного документооборота

Таким образом, экономико–информационная сущность задачи состоит в преобразовании традиционного лабораторного практикума по качественному анализу в современный автоматизированный процесс, обеспечивающий высокую безопасность, доступность, наглядность и экономическую эффективность учебного процесса.

2. Проектный раздел

2.1 Информационное обеспечение задачи

2.1.1 Архитектура данных и модель предметной области

Информационное обеспечение приложения построено на базе NoSQL-СУБД MongoDB. Это позволяет сохранить единый язык во всем стеке ПО, что существенно снижает стоимость разработки. Модель спроектирована с учётом требований к хранению информации о пользователях, учебных заданиях, химических реакциях, реагентах, ионах и истории выполнения экспериментов.

Основные сущности системы:

1. User (Пользователь) Центральная сущность, представляющая всех участников системы. Содержит следующие атрибуты:

- id– уникальный идентификатор (CUID);
- email и username– уникальные поля;
- password–хэшированный пароль с использованием bcryptjs;
- role –рольпользователя (student, teacher, admin);
- createdAt, updatedAt– временные метки создания и обновления.

Ролевая модель необходима для реализации разграничения прав доступа (RBAC).

2. LabWork (Лабораторная работа) Учебное задание или шаблон лабораторной работы, создаваемый преподавателем. Включает:

- title– название работы;
- description– подробное описание цели и методики;
- difficulty– уровень сложности (easy | medium | hard);
- category– категория лабораторной работы (необязательное поле);
- methodicalGuidelines – методические указания к работе;
- sourceLinks – список ссылок на источники;
- userId– идентификатор автора.

Лабораторная работа выступает контейнером для группы химических реакций.

3. Reaction (Химическая реакция) Одна из ключевых сущностей приложения. Описывает конкретную химическую реакцию в рамках лабораторной работы и содержит:

- name– название реакции;
- description– описание (необязательно);
- type –типреакции (acid–base, redox, precipitation, other);
- equation– молекулярное уравнение;
- ionicEquation– ионное уравнение;
- isBalanced– признак сбалансированности уравнения;
- labWorkId и userId– связи с лабораторной работой и автором.

4. Reagent (Реагент) Химическое вещество, участвующее в реакции. Хранит следующие свойства:

- name, formula;
- concentration– концентрация раствора (моль/л);
- volume– объём раствора (мл);
- color– цвет вещества в HEX–формате (используется для визуализации);
- molarMass– молярная масса.

5. Ion (Ион) Ионная частица, необходимая для моделирования качественного анализа. Включает:

- name, formula (например, Na^+ , SO_4^{2-});
- charge– заряд иона;
- color– цвет для визуализации;
- molarMass– молярная масса.

6. ExperimentResult (Результат эксперимента) Сущность, логирующая действия пользователя. Включает:

- userId – идентификатор студента;
- labWorkId – идентификатор лабораторной работы;
- actions – список действий пользователя;
- observations – заметки;

- conclusion – итог лабораторной работы;
- grade – оценка преподавателя;
- createdAt – дата выполнения.

7. Attendance (Посещаемость) Вспомогательная сущность для отслеживания посещаемости. Содержит:

- userId – ссылка на студента;
- labWorkId – ссылка на лабораторную работу;
- status – статус (present | absent);
- date – дата занятия.

Взаимосвязи сущностей:

- Один пользователь может создавать множество лабораторных работ и реакций, а также множество результатов экспериментов и записей посещаемости (отношение 1:N).
- Одна лабораторная работа содержит множество реакций, а также связана с множеством результатов экспериментов и записей посещаемости (отношение 1:N).
- Одна реакция может включать произвольное количество реагентов и ионов (отношение 1:N).

Все связи в базе данных реализованы через ссылки и вложенные документы.

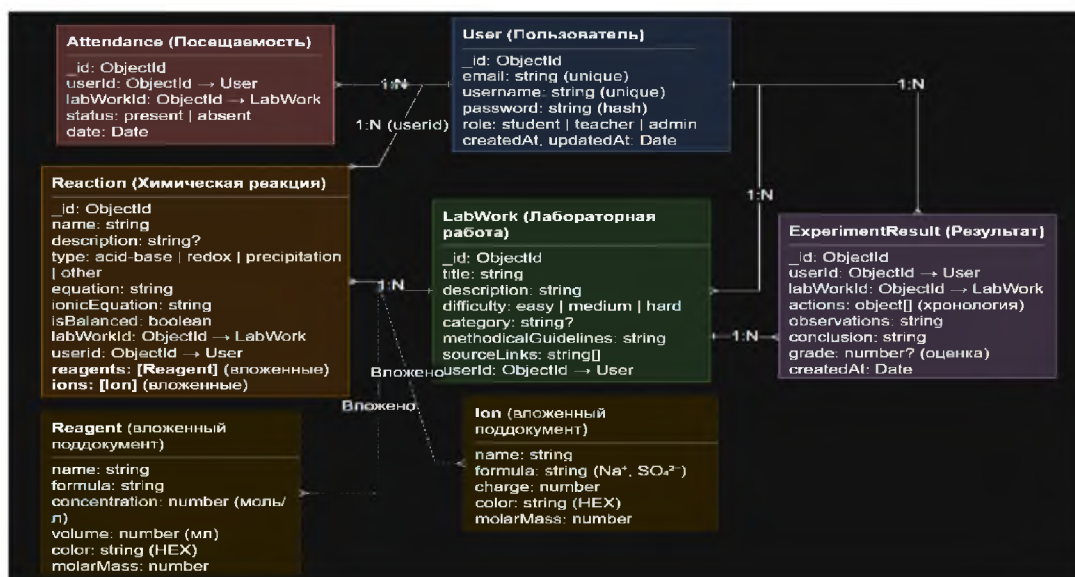


Рисунок 2.1 – Схема связей данных

Подобное устройство позволяет реализовать весь функционал системы, а также эффективно хранить данные.

2.1.2 Взаимосвязи сущностей и целостность данных

Модель данных была спроектирована с учетом реальных бизнес-процессов предметной области и отвечает всем требованиям надежности хранения данных. Архитектура связей в системе обеспечивает атомарность операций и целостность данных.

Основные связи:

- USER – LABWORK

Каждый пользователь может иметь множество лабораторных работ

- USER – REACTION

Каждый пользователь может создавать множество реакций

- USER – EXPERIMENTRESULT

Каждый пользователь может иметь множество результатов экспериментов

- USER – ATTENDANCE

Каждый пользователь имеет записи о посещаемости

- LABWORK – REACTION

Каждая лабораторная работа может содержать множество реакций

- LABWORK – EXPERIMENTRESULT

Каждая лабораторная работа связана с множеством результатов

- REACTION – REAGENT

Каждая химическая реакция может содержать множество реагентов

- REACTION – ION

Каждая химическая реакция может содержать множество ионов

В проекте использована MongooseODM. Все отношения сущностей реализованы через ссылки и вложенные документы. Для обеспечения целостности данных в системе были применены несколько механизмов:

1. Все внешние ключи имеют ограничения на уровне БД, это

запрещает создание записей со ссылками на не существующие объекты.

2. Поля с данными пользователей имеют уникальные индексы, что исключает дублирование учетных записей в системе.

3. Благодаря механизмам ORM, все сложные операции выполняются атомарно, что позволяет откатить всю операцию в случае ошибки и база останется в согласованном состоянии.

4. В часто используемые поля добавлены индексы для ускорения работы и отклика системы.

5. История создания и изменения записей постоянно отслеживается с помощью полей `createdAt` и `updatedAt`, что автоматически управляется ORM.

Такая организация обеспечивает высокую надёжность хранения результатов экспериментов, безопасность данных и возможность безопасного расширения модели в будущем.

2.1.3 Доступ к данным

Доступ к данным в приложении «Виртуальная химическая лаборатория» организован через `ODMMongoose`, что позволяет работать с ними в виде типизированных документов.

Основные принципы доступа к данным:

- Типобезопасность: Типы документов описываются в интерфейсах `TypeScript` и схеме `Mongoose`. Это исключает ошибки несоответствия типов между кодом и структурой базы данных.
- Единый интерфейс: Все операции чтения, создания, обновления и удаления данных выполняются исключительно через модели `Mongoose`.
- Оптимизация запросов: Используются техники выборочного извлечения полей (`select`), загрузки связанных сущностей (`include`), пагинации и индексов для повышения производительности.

Ключевые операции доступа к данным:

- Получение текущего пользователя по JWT-токену (`findUnique`);

- Получение списка лабораторных работ с возможностью фильтрации по автору или сложности;
 - Получение одной лабораторной работы вместе со всеми связанными реакциями (findUnique с include: { reactions: { include: { reagents: true, ions: true } } });
 - Создание полной реакции (Reaction + Reagents + Ions) в рамках одной атомарной транзакции;
 - Получение реакции со всеми реагентами и ионами для последующей визуализации на 3D-сцене с помощью Three.js;
 - Каскадное удаление лабораторной работы или реакции.
- Обеспечение безопасности и целостности:
- Все операции, изменяющие данные (создание, обновление, удаление), защищены middleware аутентификации.
 - Проверка прав доступа осуществляется на уровне сервисного слоя: студент может работать только со своими лабораторными работами и реакциями, преподаватель – со своими шаблонами, администратор имеет полный доступ.
 - Защита от SQL-инъекций обеспечивается за счет внутренних механизмов Mongoose.
 - Конфиденциальные данные (хэшированные пароли) никогда не возвращаются в API-ответах.

Использование транзакций

Особое внимание уделяется операциям, затрагивающим несколько коллекций. Пример создания реакции с использованием транзакции MongoDB:

```
const session = await mongoose.startSession();
try {
  session.startTransaction();

  const [reaction] = await Reaction.create(
    [{ ...reactionData, userId: req.user.id }],
    { session }
  );

  if (reagents?.length) {
    await Reagent.insertMany(
```

```

    reagents.map(r => ({ ...r, reactionId: reaction._id })),
    { session }
  );
}

if (ions?.length) {
  await Ion.insertMany(
    ions.map(i => ({ ...i, reactionId: reaction._id })),
    { session }
  );
}

await session.commitTransaction();
} catch (error) {
  await session.abortTransaction();
  throw error;
} finally {
  session.endSession();
}

```

В случае любой ошибки вся транзакция откатывается, гарантируя атомарность данных.

Таким образом, механизм доступа к данным сочетает типобезопасность, высокую производительность, безопасность и удобство сопровождения, полностью соответствуя принципам Clean Architecture.

2.2 Технологическое обеспечение

2.2.1 Обоснование проектных решений по технологическому обеспечению

Выбор технологического стека для разработки приложения «Виртуальная химическая лаборатория» был обусловлен необходимостью создания современного, высокопроизводительного, полностью браузерного образовательного решения, которое должно работать стабильно как в очном, так и в дистанционном формате обучения. Основными критериями выбора стали полная типизация кода, высокая производительность клиентской визуализации, безопасность, простота разработки и возможность дальнейшего масштабирования.

Принятый технологический стек:

Backend:

- Node.js 18+ как runtime-окружение[17];
- Express.js как веб-фреймворк для построения REST API;
- TypeScript как основной язык программирования;
- MongoDB как NoSQL-СУБД;
- MongooseODM для описания схем и работы с данными[18];
- JWT (jsonwebtoken) и bcryptjs для реализации аутентификации и хэширования паролей.

• Chem.js для работы с химическими формулами и моделирования реакций.

Frontend:

- React 18 + TypeScript[16];
- Vite как инструмент сборки и разработки;
- Zustand как менеджер состояния;
- Tailwind CSS для стилизации интерфейса;
- Framer Motion для анимаций[20];
- Three.js для интерактивной визуализации химических реакций и рабочего пространства;
- CanvasAPI используется для визуализации вспомогательных элементов интерфейса[19];
- Axios как HTTP-клиент.

Дополнительные инструменты:

- MongoDB Compass – для визуальной работы с базой данных во время разработки;
- jsPDF + html2canvas – для генерации PDF-отчётов по лабораторным работам.

Обоснование выбора стека:

Использование TypeScript на всём стеке обеспечивает строгую типизацию, снижает количество ошибок и значительно упрощает рефакторинг и поддержку кода.

React + Vite имеет мгновенную горячую перезагрузку и оптимизацию сборки.

Для визуализации химического оборудования и экспериментов была выбрана библиотека Three.js, позволяющая строить 3D-модель рабочего пространства, улучшая пользовательский опыт работы с приложением. Трёхмерная сцена позволит реалистично отобразить лабораторную посуду, реактивы и химические реакции.

Zustand выбран как лёгкий и производительный менеджер состояния, значительно превосходящий по простоте Redux для данного типа приложения.

MongoDB с Mongoose ODM упрощает работу с данными лабораторных работ, не требуя жесткой схемы. Также благодаря этому выбору сохраняется единый стек разработки.

Express.js в сочетании с системой middleware обеспечивает чистую и хорошо структурированную архитектуру API.

Альтернативные варианты (Python + Django/FastAPI, C# + .NET + Blazor, PHP + Laravel) были рассмотрены, но отвергнуты, поскольку требовали использования разных языков программирования на клиентской и серверной частях, что существенно усложнило бы разработку и отладку в рамках одной выпускной квалификационной работы.

Выбранный технологический стек представляет собой оптимальный баланс между современными практиками разработки, производительностью, безопасностью и удобством сопровождения.

2.2.2 Технологические операции

Технологическое обеспечение приложения «Виртуальная химическая лаборатория» включает комплекс взаимосвязанных операций, обеспечивающих полный цикл работы системы – от аутентификации пользователя до визуализации химических реакций и сохранения результатов экспериментов.

Основные технологические операции:

1. Аутентификация и авторизация Включает регистрацию нового пользователя, вход в систему, генерацию и верификацию JWT–токена, а также получение информации о текущем пользователе (/auth/me). Пароли хэшируются с помощью bcryptjs (10+ раундов). Все защищённые маршруты проверяются через auth–middleware.

2. Управление лабораторными работами Полный CRUD–цикл для сущности LabWork: создание новой лабораторной работы (доступно преподавателям и администраторам), получение списка работ, просмотр детальной информации с включением связанных реакций, обновление и удаление (с каскадным удалением зависимых данных).

3. Работа с химическими реакциями Одна из наиболее сложных операций – создание реакции вместе с набором реагентов и ионов в рамках единой атомарной транзакции базы данных. Также поддерживаются операции получения реакции со всеми связанными сущностями, обновление и удаление. Реакция содержит молекулярное и ионное уравнения, тип реакции и признак сбалансированности.

4. Управление реагентами и ионами Добавление, просмотр и удаление реагентов и ионов, связанных с конкретной реакцией. Для визуализации сохраняются цветовые характеристики веществ.

5. Интерактивная визуализация химических реакций Динамическая отрисовка реакции на Three.js. Операция включает инициализацию симулятора, отрисовку реагентов, анимацию перемешивания, отображение характерных эффектов (выпадение осадка, изменение окраски раствора, выделение газа) в реальном времени.

6. Фиксация и сохранение результатов экспериментов Автоматическая или ручная фиксация последовательности действий студента, наблюдений и итогового вывода о составе неизвестного вещества. Результаты сохраняются с привязкой к пользователю и лабораторной работе.

7. Генерация отчётов Формирование PDF–отчёта по выполненной лабораторной работе с использованием библиотек jsPDF и html2canvas. Отчёт

включает описание работы, ход эксперимента, наблюдения и выводы студента.

Все операции выполняются асинхронно. Критически важные бизнес-операции (создание реакции, сохранение лабораторной работы) оборачиваются в `database.transactions` для обеспечения целостности данных. Доступ к операциям, изменяющим данные, контролируется через систему `middleware` (`auth`, `asyncHandler`, `errorHandler`).

Такая организация технологических операций позволяет реализовать полный цикл образовательного процесса: от подготовки лабораторной работы преподавателем до выполнения эксперимента студентом и проверки результатов.

2.3 Техническое обеспечение

Обоснование проектных решений по техническому обеспечению

Техническое обеспечение приложения «Виртуальная химическая лаборатория» спроектировано с акцентом на максимальную доступность, кросс-платформенность и минимальные требования к оборудованию конечных пользователей. Приложение разработано как полностью браузерное веб-приложение, что позволяет использовать его без установки дополнительного программного обеспечения на любом устройстве, имеющем современный веб-браузер.

Основные технические решения:

- Клиентская часть полностью реализована с использованием веб-технологий и работает в любом современном браузере (Google Chrome, Mozilla Firefox, Microsoft Edge, Safari), поддерживающем HTML5 Canvas и ES2022+.
- Серверная часть построена на Node.js и может быть развёрнута на любом хостинге, поддерживающем серверную среду Node.js.
- База данных: MongoDB используется на этапе разработки (локальный экземпляр базы данных), так и во время промышленной эксплуатации, обеспечивая масштабируемость и надежность работы системы.

- Архитектура развёртывания: Frontend может размещаться на статических хостингах (Vercel, Netlify), Backend – на PaaS-сервисах (Railway, Render) или собственном сервере.

Обоснование выбранных технических решений:

Браузерная архитектура обеспечивает нулевую установку для студентов и преподавателей – достаточно открыть ссылку в браузере. Это особенно важно для школ, дистанционного и смешанного обучения, где организация традиционных химических лабораторий затруднена.

Использование Canvas API в сочетании с Three.js на клиентской стороне позволяет добиться высокой производительности визуализации химических реакций даже на относительно слабых устройствах, без необходимости установки дополнительных плагинов.

Разделение приложения на статический frontend и отдельный API-backend даёт гибкость при масштабировании: frontend может эффективно раздаваться через CDN, снижая нагрузку на серверную часть.

Использование MangoDB на обоих этапах (разработка и эксплуатация) обеспечивает единообразие модели данных: простота на этапе создания проекта и необходимая производительность + надёжность в реальной многопользовательской среде.

Минимальные системные требования к рабочему месту пользователя:

- Современный веб-браузер с поддержкой HTML5 Canvas;
- Процессор: 2+ ядра (рекомендуется 4+);
- Оперативная память: от 4 ГБ (рекомендуется 8 ГБ);
- Стабильное подключение к интернету (для загрузки реакций и сохранения результатов экспериментов).

Комплекс технических средств

Комплекс технических средств виртуальной химической лаборатории включает аппаратные, программные и сетевые компоненты, обеспечивающие стабильную работу приложения в учебной среде.

1. Аппаратные средства

- Серверная инфраструктура: сервер (или облачный PaaS), на котором размещается backend–приложение и база данных MongoDB.
- Клиентские устройства пользователей: персональные компьютеры, ноутбуки, планшеты и другие устройства студентов и преподавателей.
- Сетевые компоненты: каналы связи с доступом в интернет для взаимодействия клиентской и серверной частей приложения.

2. Программные средства

2.1 Клиентская часть:

- Веб–браузер (Chrome, Firefox, Edge, Safari) с поддержкой HTML5 и Canvas API;
- React 18 приложение, собранное с помощью Vite;
- Zustand для управления состоянием;
- Three.js для визуализации химических реакций;
- Canvas API для визуализации вспомогательных элементов.

2.2 Серверная часть:

- Node.js 18+;
- Express.js – веб–фреймворк;
- MongoDB (разработка и эксплуатация)

2.3 Средства разработки и сопровождения:

- MongoDB Compass– визуальный инструмент для работы с базой данных;
- TypeScript Compiler;
- ESLint и Prettier– для контроля качества кода;
- Git– система контроля версий.

2.4 Дополнительные инструменты:

- jsPDF + html2canvas – генерация PDF–отчётов;
- Axios– клиент для HTTP–запросов.

3. Средства коммуникации и обмена данными

- Протокол HTTP/HTTPS;
- REST API как основной интерфейс взаимодействия между frontend

и backend;

- JSON как формат сериализации данных;
- JWT–токены для передачи информации об аутентификации.

4. Средства защиты информации

- HTTPS в продакшене;
- JWT–based аутентификация с ограниченным сроком жизни токена;
- bcryptjs для безопасного хранения паролей;
- CORS–политика с чётко определённым разрешённым происхождением (CORS_ORIGIN);
- Защита от SQL–инъекций на уровне MongooseODM;
- Валидация всех входных данных на уровне домена и сервисов.

Схема развёртывания

- На этапе разработки: локальная машина (frontend– порт 3000, backend– порт 3001, локальный экземпляр MongoDB).
- В промышленной эксплуатации возможны 2 варианта:
 1. Размещение на сервере образовательной организации, что минимизирует затраты при наличии собственной инфраструктуры.
 2. Размещение в облаке: frontend на статичном хостинге, backend на node.js-хостинге (Railway, Render), база данных MongoDBAtlas. Этот вариант не требует оборудования, но подразумевает дополнительные ежемесячные затраты на содержание системы. Такой комплекс технических средств обеспечивает высокую доступность приложения, безопасность данных, удобство разработки и эксплуатации, а также соответствует современным требованиям к образовательным информационным системам.

2.4 Архитектура и дизайн программного обеспечения

Общая архитектура приложения

Общая архитектура приложения «Виртуальная химическая лаборатория» построена по принципам Clean Architecture с чётким разделением

ответственности на четыре независимых слоя. Такое построение обеспечивает высокую поддерживаемость кода, удобство тестирования, независимость бизнес-логики от конкретных фреймворков и возможность дальнейшего расширения системы.

Архитектура приложения следует многослойной модели, где каждый слой имеет строго определённые обязанности и зависит только от слоёв, расположенных «ниже» него (DependencyRule). Это позволяет легко заменять реализации (например, базу данных или UI-фреймворк) без изменения бизнес-логики.

Четыре основных слоя архитектуры:

1. Presentation Layer (Слой представления) Отвечает за взаимодействие с пользователем и внешним миром.

- Backend: Express.js controllers и маршруты (AuthController, LabWorkController, ReactionController).

- Frontend: React-компоненты, страницы и пользовательский интерфейс. Этот слой принимает HTTP-запросы, валидирует входные данные и делегирует обработку сервисам.

2. Application Layer (Слой приложения / бизнес-логики) Содержит основные сценарии использования системы.

- Backend: Service-классы (AuthService, LabWorkService, ReactionService, ReagentService, IonService).

- Frontend: Zustand stores (AuthStore, LabWorkStore) и custom hooks (useChemistrySimulator, useAsync). Здесь реализуется оркестрация операций: создание реакции с реагентами и ионами, проверка прав доступа, валидация бизнес-правил.

3. Domain Layer (Предметный / доменный слой) Ядро приложения, независимое от внешних технологий. Содержит:

- Бизнес-сущности и типы (User, LabWork, Reaction, Reagent, Ion, ExperimentResult, Attendance);

- Кастомные ошибки (ValidationError, UnauthorizedError,

NotFoundError, AppError). Этот слой содержит все бизнес-правила и логику, которая не должна зависеть от фреймворков или баз данных.

4. Infrastructure Layer (Инфраструктурный слой) Реализует технические детали и внешние зависимости.

- Backend: Mongoose ODM, JWT-утилиты, middleware (auth, asyncHandler, errorHandler), конфигурация (.env).
- Frontend: Axios client, симулятор (ChemistrySimulator class).

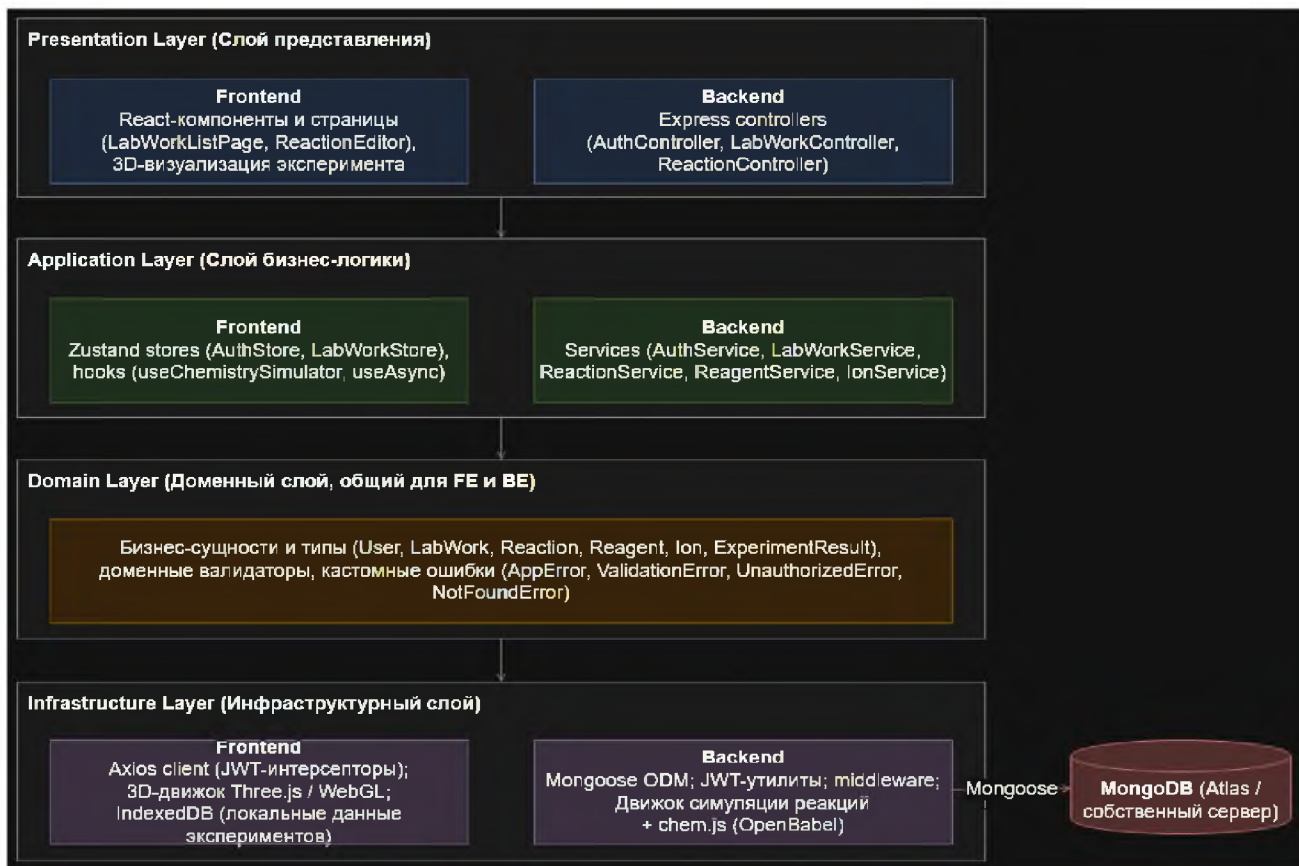


Рисунок 2.2 – Архитектура приложения

Преимущества выбранной архитектуры:

- Чёткое разделение ответственности (Single Responsibility Principle);
- Независимость доменной логики от деталей реализации (базы данных, UI, фреймворков);
- Высокая тестируемость – доменный и application слои легко тестируются в изоляции;
- Удобство расширения – добавление новых типов реакций или

интеграций не требует переписывания ядра приложения;

- Соответствие современным лучшим практикам разработки (Clean Architecture, Layered Architecture).

2.4.1 Backend–архитектура

Backend приложения «Виртуальная химическая лаборатория» построен по многослойной архитектуре с чётким разделением ответственности. Основной подход –Service Layer Pattern в сочетании с принципами Clean Architecture. Это позволяет отделить бизнес–логику от технических деталей и сделать код хорошо структурированным и поддерживаемым.

Структура backend–слоёв:

1. Presentation Layer (Слой представления) Состоит из контроллеров и маршрутов.

- AuthController, LabWorkController, ReactionController.
- Контроллеры принимают HTTP–запросы, извлекают данные из тела запроса, вызывают соответствующий сервис и формируют ответ в едином формате ({ success, message, data } или { success, error, details }).
- Все контроллеры оборачиваются в asyncHandler, чтобы избежать повторяющегося try/catch.

2. Application Layer (Слой приложения) Содержит бизнес–логику системы в виде сервисных классов:

- AuthService– регистрация, вход, генерация JWT;
- LabWorkService– CRUD операций над лабораторными работами;
- ReactionService– создание, обновление и получение реакций;
- ReagentService, IonService– управление связанными сущностями.

Сервисы выполняют валидацию бизнес–правил, проверку прав доступа и координируют работу с репозиториями (Mongoose-модели).

3. Domain Layer (Доменный слой) Ядро приложения, полностью независимое от внешних технологий. Содержит:

- Типы интерфейсы (User, LabWork, Reaction, Reagent, Ion, ExperimentResult, Attendance);
- Доменные валидаторы (validateEmail, validatePassword, validateReactionData и др.);
- Кастомные исключения (AppError, ValidationError, UnauthorizedError, NotFoundError).

Здесь сосредоточены все бизнес-правила (например, проверка сбалансированности уравнения, уникальность email/username и т.д.).

4. Infrastructure Layer (Инфраструктурный слой) Реализует технические детали:

- MongooseODM для работы с базой данных;
- JWT-утилиты (генерация и верификация токенов);
- Middleware-цепочка (authMiddleware, errorHandler, asyncHandler);
- Конфигурация приложения (.env, CORS, порт сервера).

Ключевые технические паттерны, применённые на backend:

- Service Layer Pattern– каждый домен имеет свой сервис, который инкапсулирует всю бизнес-логику.
- Database Transactions– создание реакции вместе с реагентами и ионами выполняется в одной атомарной транзакции с помощью механизмов Mongoose.
- CustomErrorHandling– все ошибки обрабатываются централизованно через errorHandlermiddleware и возвращаются в едином формате.
- AsyncHandlerMiddleware– устраняет дублирование try/catch в контроллерах.
- Multi-levelValidation– валидация происходит на трёх уровнях: доменная, сервисная и на уровне базы данных (валидаторы схем Mongoose).

Маршрутизация и безопасность

Все защищённые маршруты проходят через authMiddleware, который проверяет наличие и валидность JWT-токена. Роли пользователей учитываются

при выполнении операций (студент может работать только со своими данными, преподаватель – со своими шаблонами).

Такая backend–архитектура обеспечивает высокую степень разделения ответственности, лёгкость тестирования и удобство расширения функциональности.

2.4.2 Frontend–архитектура

Frontend–часть приложения «Виртуальная химическая лаборатория» построена по тем же принципам многослойной архитектуры (Clean Architecture), что и backend, но адаптирована под специфику React–приложения. Это обеспечивает чёткое разделение ответственности, эффективное управление состоянием и высокую производительность интерактивной визуализации.

Структура frontend–слоёв:

1. Presentation Layer (Слой представления) Отвечает за пользовательский интерфейс и взаимодействие с пользователем.

- React–компоненты и страницы (LabWorkListPage, LabWorkDetailPage, ReactionEditor, ChemistrySimulator и др.).
- Переиспользуемые UI–компоненты (Button, Card, TestTube, ReagentShelf, ObservationJournal).
- Используются Framer Motion для плавных анимаций и Tailwind CSS для стилизации. Этот слой полностью декларативный и отвечает только за рендеринг и обработку событий пользователя.

2. Application Layer (Слой приложения) Содержит бизнес–логику и оркестрацию операций на клиентской стороне.

- Zustand stores: AuthStore, LabWorkStore– глобальное состояние приложения с поддержкой подписок и persist (localStorage).
- Custom hooks: useAsync, useChemistrySimulator, useAuth и др.
- Здесь реализуются асинхронные действия: загрузка лабораторных

работ, создание реакции, управление состоянием симуляции.

3. Domain Layer (Доменный слой) Содержит чистые бизнес–типы и правила, независимые от React и других библиотек:

- TypeScript–интерфейсы и типы (User, LabWork, Reaction, Reagent, Ion, ExperimentResult, Attendance);
- Доменные валидаторы;
- Бизнес–сущности и константы. Этот слой может использоваться как на frontend, так и на backend (через sharedtypes).

4. Infrastructure Layer (Инфраструктурный слой) Реализует технические детали и внешние зависимости:

- Axiosclient с интерсепторами (автоматическое добавление JWT–токена);
- ChemistrySimulatorclass– класс для работы с визуализацией (инициализация, анимационный цикл, отрисовка реагентов и эффектов);
- Локальное хранилище (localStorage для токена и состояния);
- Браузерная NoSQL-СУБД IndexedDB для накопления данных об эксперименте на используемом устройстве;
- Утилиты и константы.

Ключевые технические решения на frontend:

- Zustand используется вместо Redux благодаря своей минималистичности и высокой производительности. Stores содержат как состояние, так и асинхронные действия (fetchLabWorks, createReaction и т.д.).
- CustomHookPattern– особенно useChemistrySimulator, который инкапсулирует всю логику работы с визуализацией реакций и процессов: инициализацию контекста, анимационный цикл (requestAnimationFrame), добавление реагентов, паузу и сброс симуляции.
- Code Splitting и мемоизация (React.memo, useCallback, useMemo) для оптимизации производительности.

Преимущества frontend–архитектуры:

- Чёткое разделение UI от бизнес–логики;

- Легкость тестирования компонентов и хуков;
- Высокая производительность визуализации благодаря нативному Canvas API и оптимизированному анимационному циклу;
- Возможность повторного использования доменных типов между frontend и backend;
- Удобная масштабируемость – добавление новых лабораторных работ или типов визуализации требует минимальных изменений.

Ключевые технические паттерны и решения

1. ServiceLayerPattern На backend каждый домен имеет dedicated сервис (AuthService, LabWorkService, ReactionService и др.). Сервисы инкапсулируют всю бизнес-логику, выполняют валидацию, проверку прав доступа и координируют работу с базой данных. Это позволяет контроллерам оставаться тонкими и сосредоточиться только на обработке HTTP-запросов.

2. Multi-level Validation Валидация данных выполняется на трёх уровнях:

- Доменный уровень (validateEmail, validatePassword, validateReactionData);
- Сервисный уровень (бизнес-правила, проверка уникальности);
- Уровень базы данных (уникальные индексы Mongoose). Такой подход обеспечивает раннее выявление ошибок и информативные сообщения для пользователя.

3. Custom Error Handling Используется иерархия кастомных ошибок (AppError, ValidationError, UnauthorizedError, NotFoundError). Все ошибки централизованно обрабатываются через errorHandler middleware и возвращаются клиенту в едином формате с полем timestamp.

4. AsyncHandlerMiddleware Специальный middleware устраняет дублирование try/catch в контроллерах, автоматически перехватывая ошибки и передавая их в errorHandler.

5. Database Transactions Все сложные операции (создание реакции вместе с реагентами и ионами) выполняются внутри сессии Mongoose. Это гарантирует атомарность и согласованность данных даже при возникновении ошибок на

любом этапе.

6. На frontend реализован класс ChemistrySimulator, который управляет Canvas-контекстом, классами библиотеки Three.js, анимационным циклом (requestAnimationFrame) и визуальными эффектами. Логика визуализации инкапсулирована в customhookuseChemistrySimulator, что позволяет легко подключать симуляцию к любому компоненту.

7. State Management с Zustand Глобальное состояние приложения (аутентификация, лабораторные работы, текущая реакция) управляется через Zustandstores с поддержкой подписок и persist в localStorage. Это обеспечивает эффективные обновления UI без лишних ререндеров.

8. Repository / Mongoose-модели выступает в роли абстракции над базой данных. Все запросы выполняются через сервисы, а не напрямую из компонентов или контроллеров.

9. Middleware Chain Backend использует цепочку middleware:

- CORS → Security headers → Async Handler → Auth Middleware → Error Handler.

10. Code Splitting и Performance Patterns На frontend применяются React.lazy, React.memo, use Callbackи use Memo для оптимизации производительности.

Применение этих паттернов делает архитектуру приложения чистой, тестируемой и масштабируемой. Они позволяют легко добавлять новые лабораторные работы, типы реакций или интеграции без значительного рефакторинга существующего кода.

2.5 Реализация основных модулей приложения

Модуль аутентификации и авторизации является базовым компонентом системы и обеспечивает безопасный доступ пользователей к функционалу приложения. Он реализован в соответствии с лучшими практиками безопасности и полностью интегрирован с многослойной архитектурой.

Функциональные возможности модуля:

- Регистрация нового пользователя (с проверкой уникальности email и username);
- Авторизация пользователя (логин по email и паролю);
- Получение информации о текущем пользователе (/auth/me);
- Выход из системы (удаление токена);
- Ролевая модель доступа (student, teacher, admin).

Техническая реализация:

- Backend:
- AuthService содержит логику регистрации, входа, хэширования паролей (bcryptjs с 10+ раундами) и генерации JWT–токенов.
- AuthController обрабатывает HTTP–запросы (POST /auth/register, POST /auth/login, GET /auth/me).
- authMiddleware проверяет наличие и валидность JWT–токена в заголовке Authorization: Bearer <token>.
- Кастомные ошибки (UnauthorizedError) используются для унифицированной обработки случаев неверных учётных данных.
- Frontend:
- AuthStore (Zustand) хранит информацию о текущем пользователе, токене и статусе аутентификации с автоматическим сохранением в localStorage.
- Формы регистрации и входа построены с использованием доменных валидаторов (validateEmail, validatePassword).
- Axiosinterceptor автоматически добавляет JWT–токен во все защищённые запросы и обрабатывает 401 ошибки (очистка токена и редирект на страницу входа).

Безопасность модуля:

- Пароли никогда не хранятся в открытом виде;
- JWT–токены имеют ограниченное время жизни (24 часа);
- Защищённые маршруты недоступны без валидного токена;
- Все входные данные проходят многоуровневую валидацию.

Модуль полностью готов к использованию и служит фундаментом для всех остальных частей приложения, обеспечивая безопасное разграничение прав доступа между студентами, преподавателями и администраторами.

Модуль управления лабораторными работами предназначен для создания, хранения и организации учебных заданий (шаблонов лабораторных работ). Он является одним из центральных модулей системы и обеспечивает полный цикл работы преподавателя с учебным материалом.

Основные функции модуля:

- Создание новой лабораторной работы с методическими указаниями и ссылками на источники (доступно пользователям с ролью teacher и admin);
- Проверка уже выполненных лабораторных работ и их оценка;
- Просмотр списка всех лабораторных работ (доступно всем авторизованным пользователям);
- Получение детальной информации о конкретной лабораторной работе вместе со всеми связанными реакциями;
- Редактирование и удаление лабораторных работ (только автором или администратором);
- Фильтрация и сортировка работ по сложности (easy, medium, hard), категории и автору.

Техническая реализация:

Backend:

- LabWorkService содержит всю бизнес-логику: проверку прав доступа, валидацию входных данных и работу с базой данных.
- LabWorkController обрабатывает следующие эндпоинты:
 - POST /lab-works– создание новой работы;
 - GET /lab-works– получение списка работ;
 - GET /lab-works/:id– получение работы с включением реакций;
 - PUT /lab-works/:id и DELETE /lab-works/:id– обновление и удаление.
- При удалении лабораторной работы автоматически удаляются все

связанные реакции, реагенты и ионы благодаря каскадному удалению через хуки `Mongoose`.

Frontend:

- `LabWorkStore` (`Zustand`) управляет состоянием списка лабораторных работ и текущей выбранной работой.
- Компоненты: `LabWorkListPage`, `LabWorkCard`, `LabWorkDetailPage`.
- Форма создания/редактирования лабораторной работы реализована с использованием контролируемых полей и доменной валидации.
- Асинхронные действия (`fetchLabWorks`, `createLabWork`, `deleteLabWork`) обрабатываются через `customhookuseAsync`.

Ключевые особенности реализации:

- При создании лабораторной работы автоматически устанавливается связь с текущим пользователем (`userId = req.user.id`).
- Получение лабораторной работы выполняется с `eagerloading` связанных реакций (`include: { reactions: { include: { reagents: true, ions: true } } }`), что позволяет сразу отобразить полную структуру работы на frontend.
- Реализована защита: студент может просматривать все работы, но редактировать и удалять только свои собственные (если это предусмотрено бизнес-правилами).

Модуль тесно интегрирован с модулем реакций и служит основой для проведения виртуальных лабораторных работ. Он полностью соответствует требованиям и обеспечивает удобный интерфейс как для преподавателей (создание шаблонов), так и для студентов (выбор и выполнение заданий).

Модуль работы с химическими реакциями является одним из наиболее сложных и важных в приложении. Он отвечает за создание, хранение, редактирование и подготовку данных для визуализации химических реакций в рамках лабораторных работ.

Основные функции модуля:

- Создание новой химической реакции внутри лабораторной работы;
- Добавление реагентов и ионов к реакции;

- Получение реакции со всеми связанными данными (реагенты + ионы);
- Редактирование и удаление реакции;
- Проверка сбалансированности уравнения (молекулярного и ионного).

Техническая реализация:

Backend:

- ReactionService– центральный сервис, реализующий всю бизнес–логику.
- Ключевая операция – создание реакции с реагентами и ионами в рамках одной атомарной транзакции:

```
async create(data: CreateReactionDTO): Promise<ReactionDocument> {
  // Реагенты и ионы хранятся как вложенные поддокументы реакции,
  // поэтому вся реакция создаётся одним атомарным документом.

  const reaction = await Reaction.create({
    labWorkId: data.labWorkId,
    userId: data.userId,
    name: data.name,
    equation: data.equation,
    ionicEquation: data.ionicEquation,
    type: data.type,
    isBalanced: data.isBalanced,
    reagents: data.reagents ?? [],
    ions: data.ions ?? [],
  });

  return reaction;
}
```

- ReactionController обрабатывает эндпоинты:
 - POST /reactions
 - GET /lab-works/:labWorkId/reactions
 - GET /reactions/:id
 - PUT /reactions/:id
 - DELETE /reactions/:id
- Добавление реагентов и ионов также поддерживается отдельными эндпоинтами (POST /reagents, POST /ions).

Frontend:

- LabWorkStore и ReactionEditor компонент управляют формой создания/редактирования реакции.
- После успешного создания реакции данные автоматически добавляются в Zustandstore (addReaction).
- ChemistrySimulator получает полные данные реакции (reagents + ions) для визуализации.

Особенности реализации модуля

- Атомарность каждой транзакции.
- Проверка уравнения и типа реакции. А также соответствия реагентов на сервисном уровне. Для работы с химическими формулами и определения продуктов реакции используется библиотека Chem.js.
- Прямая связь с визуализацией осуществляется за счет мгновенной загрузки реагентов и их характеристик в момент получения реакции.

Описываемый модуль имеет прямую связь с модулем визуализации и является связующим звеном для всех частей проекта.

Модуль интерактивной визуализации является ключевой отличительной особенностью приложения «Виртуальная химическая лаборатория». Он обеспечивает реалистичную динамическую симуляцию химических реакций в браузере с использованием Three.js (WEBGL) для создания 3D-сцен.

Основные функции модуля:

- Визуализация пробирки / колбы как основной рабочей области;
- Добавление реагентов в реакционную среду (drag&drop или кнопками);
- Реалистичная анимация химических процессов:
 - Появление и перемешивание реагентов;
 - Изменение цвета раствора;
 - Выпадение осадка различной окраски и плотности;
 - Выделение газа (пузырьки);
- Управление симуляцией (Play, Pause, Reset);

- Синхронизация визуализации с данными реакции (реагенты + ионы + тип реакции).

Техническая реализация:

Frontend (основная часть модуля):

- ChemistrySimulator– отдельный класс в инфраструктурном слое (infrastructure/canvas/ChemistrySimulator.ts), который инкапсулирует всю логику работы с Canvas:

- Инициализация canvas–контекста;
- Управление анимационным циклом через requestAnimationFrame (цель – 60 FPS);
- Расчёт позиций частиц, скоростей и состояний;
- Отрисовка визуальных эффектов (градиенты, частицы, текстуры осадка).

- useChemistrySimulator–customhook, который связывает React–компонент с классом симулятора:

- Принимает массив реакций/реагентов;
- Управляет состоянием isRunning;
- Предоставляет методы animate(), pause(), reset(), addReagent().
- ReactionViewer / ChemistrySimulator Component– React–компонент, содержащий интерфейс симулятора и панель управления (кнопки Play/Pause/Reset).

Ключевые технические решения:

- Использование Three.js для построения 3D-объектов (рабочий стол, лабораторная посуда).
- Анимационный цикл оптимизирован: пересчёт физики частиц и отрисовка происходят в одном кадре.
- Цвета реагентов и ионов берутся напрямую из базы данных (color поле в Reagent и Ion) и используются для динамической отрисовки.
- Поддержка разных типов реакций: кислотно–основные (изменение цвета), осадочные (выпадение твёрдой фазы), газовыделяющие (анимация

пузырьков).

- Реакция на действия пользователя в реальном времени (добавление нового реагента мгновенно отражается в симуляции).

Интеграция с другими модулями:

- Получает данные из LabWorkStore и Reaction сущности (список реагентов и ионов).
- Синхронизируется с ReactionEditor– при добавлении реагента в форму он сразу появляется в визуализации.
- Результаты наблюдений (цвет осадка, изменение раствора и т.д.) студент может фиксировать в журнале наблюдений.

Модуль обеспечивает высокий уровень вовлечённости студентов за счёт интерактивности и приближенности к реальному лабораторному опыту, при этом полностью безопасен и не требует реальных химических веществ.

Модуль фиксации результатов и генерации отчётов завершает цикл выполнения лабораторной работы, позволяя студенту сохранять ход эксперимента и формировать итоговый документ для проверки преподавателем.

Основные функции модуля:

- Автоматическая фиксация последовательности действий студента (добавление реагентов, порядок их внесения);
- Ручная запись наблюдений в электронный журнал (цвет осадка, изменение окраски раствора, выделение газа и другие визуальные эффекты);
- Формирование итогового вывода о составе неизвестного вещества;
- Сохранение полного результата лабораторной работы в базе данных;
- Генерация PDF–отчёта по завершённой работе.

Техническая реализация:

Backend:

- Расширение LabWorkService и ReactionService для сохранения истории выполнения.
- Отдельная сущность ExperimentResult содержит результаты

выполнения работы (лог действий, наблюдения, вывод и оценку преподавателя).

- Эндпоинт для сохранения результата эксперимента с привязкой к пользователю и лабораторной работе.

Frontend:

- ObservationJournal– компонент правой панели, где студент может в реальном времени записывать наблюдения.
- Автоматическая запись действий: каждое добавление реагента логируется с временной меткой.
- LabWorkStore сохраняет текущее состояние эксперимента (список выполненных действий, наблюдения, вывод).
- Кнопка «Завершить работу» вызывает сохранение всех данных на сервер.

Генерация PDF–отчёта:

- Используются библиотеки jsPDF и html2canvas.
- Отчёт включает:
 - Название лабораторной работы и цель;
 - Список добавленных реагентов с концентрациями и объёмами;
 - Хронологию действий студента;
 - Зафиксированные наблюдения;
 - Итоговый вывод студента;
 - Дата выполнения и имя студента.

Ключевые особенности реализации:

- Сохранение происходит как автоматически (по ключевым действиям), так и вручную по кнопке «Сохранить результат».
- Все данные сохраняются с привязкой к конкретному пользователю (userId), что позволяет преподавателю видеть историю выполнения каждого студента.
- PDF–отчёт формируется на клиентской стороне, что снижает нагрузку на сервер и позволяет работать даже при слабом соединении (отчёт

можно скачать сразу).

- Журнал наблюдений поддерживает форматированный текст и возможность добавления фотографий/скриншотов canvas (в перспективе).

Модуль обеспечивает полный переход от традиционного бумажного журнала к электронной системе фиксации результатов, значительно упрощая проверку работ преподавателем и давая студенту возможность многократно анализировать свой эксперимент.

2.6 Руководство пользователя

2.6.1 Описание интерфейса

Интерфейс приложения «Виртуальная химическая лаборатория» разработан в современном минималистичном стиле с акцентом на удобство использования в учебном процессе. Он полностью адаптивен и комфортно работает как на десктопных компьютерах, так и на планшетах.

Общая структура интерфейса:

- Боковая навигационная панель (Sidebar) Содержит основные разделы:
 - Дашборд
 - Лабораторные работы
 - Мои эксперименты
 - Отчёты (для преподавателей) На мобильных устройствах панель сворачивается в гамбургер–меню.
- Верхняя панель (AppBar) Отображает название текущей лабораторной работы, информацию о текущем пользователе (имя и роль), уведомления и кнопку выхода из системы.
- Главная рабочая область делится на три основные зоны при выполнении лабораторной работы:
 - Левая панель – «Шкаф реагентов» Список доступных реагентов в виде карточек. Поддерживается drag&drop для добавления реагента в пробирку,

а также добавление по кнопке.

- Центральная область – Зона симуляции Основное рабочее пространство с виртуальной пробиркой (Three.js). Здесь происходит вся визуализация химической реакции в реальном времени: добавление веществ, перемешивание, изменение цвета раствора, выпадение осадка, выделение газа.

- Правая панель – «Журнал наблюдений» Электронный журнал, где студент фиксирует свои наблюдения. Автоматически записываются действия (какой реагент был добавлен и когда). Есть возможность ручного ввода текста.

- Нижняя панель управления Содержит кнопки: «Перемешать», «Нагреть», «Сбросить пробирку», «Завершить работу», «Сохранить результат» и «Скачать отчёт».

Дополнительные особенности интерфейса:

- Подсказки (tooltips) у всех интерактивных элементов;
- Поддержка светлой и тёмной темы;
- Адаптивный дизайн (на планшетах зоны автоматически перестраиваются);
- Цветовая схема соответствует лабораторной тематике (спокойные тона, акцентные цвета для реакций);
- Интуитивно понятная навигация, приближенная к реальной лабораторной работе.

Интерфейс построен с использованием Tailwind CSS и Framer Motion, что обеспечивает плавные анимации и высокую отзывчивость.

2.6.2 Порядок работы с приложением

Приложение «Виртуальная химическая лаборатория» имеет интуитивно понятный интерфейс, однако порядок работы различается в зависимости от роли пользователя (студент или преподаватель).

Порядок работы для студента:

1. Авторизация Пользователь переходит на страницу входа, вводит

email и пароль (или регистрируется, если аккаунт отсутствует). После успешной авторизации открывается дашборд.

2. Выбор лабораторной работы На дашборде или в разделе «Лабораторные работы» студент просматривает доступные задания. Выбирает нужную лабораторную работу и нажимает кнопку «Начать работу».

3. Выполнение эксперимента

- Изучает описание лабораторной работы, цель и теоретический материал.

- Переходит в рабочую область.

- Из «Шкафа реагентов» (левая панель) добавляет реагенты в виртуальную пробирку (перетаскиванием или кнопкой).

- Наблюдает за реакцией в центральной области (Three.js) в реальном времени.

- Фиксирует наблюдения в правом журнале (цвет осадка, изменение окраски, выделение газа и др.).

- Делает вывод о составе неизвестного вещества на основе наблюдаемых эффектов.

4. Завершение работы После завершения эксперимента студент нажимает кнопку «Завершить работу». Результат автоматически сохраняется в системе с привязкой к его аккаунту.

Общие рекомендации по работе:

- Все действия сопровождаются понятными уведомлениями (success/error).

- При неправильных действиях (несовместимые реагенты) система выводит подсказку с объяснением.

- Рекомендуется выполнять работу в полноэкранном режиме для лучшего восприятия визуализации.

- Перед завершением работы рекомендуется проверить журнал наблюдений и итоговый вывод.

Интерфейс максимально приближен к реальной лабораторной работе, что

помогает студентам быстрее адаптироваться и лучше усваивать материал. Порядок работы построен таким образом, чтобы минимизировать технические сложности и сосредоточить внимание пользователя на химическом эксперименте.

3. Обоснование экономической эффективности результатов ВКР

3.1 Выбор и обоснование методики расчета экономической эффективности

Расчёт экономической эффективности внедрения разработанного приложения виртуальной лаборатории по качественному анализу в химии проводится на основе сравнения затрат на традиционное проведение лабораторных работ и затрат после внедрения цифрового решения.

Методика расчёта основана на типовых подходах оценки эффективности информационных систем в образовании и учитывает следующие виды экономического эффекта[4;14]:

- Прямую экономию на закупке реактивов и расходных материалов;
- Снижение затрат на лабораторное оборудование и его обслуживание;
- Уменьшение трудозатрат преподавателей и лаборантов;
- Снижение расходов на обеспечение техники безопасности и утилизацию отходов;
- Косвенный эффект от повышения качества обучения и сокращения времени на передачи лабораторных работ.

Для расчёта используются нормативные показатели филиала/кафедры на 2025–2026 учебный год, а также данные о количестве студентов, проходящих курс неорганической химии. Коэффициент снижения затрат на реактивы принят равным 0,85 (85 %), так как виртуальная лаборатория полностью заменяет реальные реактивы для качественного анализа.

Расчёт включает определение:

- Единовременных затрат на разработку и внедрение;
- Годового экономического эффекта;
- Срока окупаемости;
- Чистой приведённой стоимости (NPV).

Методика соответствует требованиям методических рекомендаций по

выполнению выпускных квалификационных работ и позволяет объективно оценить целесообразность внедрения приложения.

3.2 Расчет показателей экономической эффективности

Расчёт проведён для условий типичного потока: 240 студентов в год (4 группы по 60 человек), проходящих 6 лабораторных работ по качественному анализу.

1. Затраты на разработку приложения

- Затраты на оплату труда разработчика (3 месяцев): 180 000 руб.
- Ежегодные эксплуатационные затраты в случае облачного размещения системы: 36 000 руб./год.
- Затраты на тестирование и отладку: 20 000 руб.
- Накладные расходы (20 %): 40 000 руб.
- Итого единовременные затраты: 240 000 руб.

2. Годовой экономический эффект от внедрения

Текущие ежегодные затраты на проведение лабораторных работ по качественному анализу:

- Закупка реактивов и расходных материалов: 380 000 руб./год
- Амортизация и обслуживание оборудования: 85 000 руб./год
- Трудозатраты лаборантов и преподавателей на подготовку и проведение работ: 420 000 руб./год

- Утилизация химических отходов: 45 000 руб./год

Итого текущие затраты: 930 000 руб./год

После внедрения виртуальной лаборатории:

- Затраты на реактивы снижаются на 85 % → экономия 323 000 руб.
- Затраты на оборудование снижаются на 70 % → экономия 59 500 руб.
- Трудозатраты преподавателей и лаборантов снижаются на 45 % → экономия 189 000 руб.

- Затраты на утилизацию отходов практически исчезают → экономия 45 000 руб.

Итого годовой экономический эффект: 616500 руб, однако с учетом размещения в облаке годовой эффект снижается до 580 500 (36000 рублей в год на аренду инфраструктуры)[21].

3. Срок окупаемости и NPV

- Срок окупаемости = Затраты на разработку / Годовой эффект = 240 000 / 616 500 $\approx$ 0,38 года (около 5 месяцев).

Расчёт чистой приведённой стоимости (NPV) при норме дисконта 12 % за 5 лет эксплуатации показывает положительное значение уже на втором году использования.

Таблица 3.1 – Расчёт годового экономического эффекта

Статья затрат	Затраты до внедрения, руб./год	Затраты после внедрения, руб./год	Экономия, руб./год
Закупка реактивов	380 000	57 000	323 000
Обслуживание оборудования	85 000	25 500	59 500
Трудозатраты персонала	420 000	231 000	189 000
Утилизация отходов	45 000	0	45 000
Итого	930 000	313 500	616 500

Таким образом, внедрение разработанного приложения виртуальной лаборатории является экономически целесообразным. Срок окупаемости составляет менее полугода, а ежегодная экономия превышает 616 тысяч рублей при хостинге на собственной инфраструктуре. Кроме прямого экономического эффекта, внедрение приложения позволит повысить безопасность учебного процесса, увеличить доступность лабораторных работ и улучшить качество подготовки студентов.

Заключение

В рамках выпускной квалификационной работы была успешно достигнута поставленная цель – разработано современное браузерное информационное приложение виртуальной лаборатории по качественному анализу в курсе неорганической химии.

Реализация проекта позволила решить все поставленные задачи. В аналитическом разделе проведён подробный анализ предметной области виртуальных лабораторий по химии. Изучены как отечественные, так и зарубежные решения (PhET, Labster, ChemCollective, BeyondLabz и другие). Выявлены их основные недостатки: недостаточная интерактивность при моделировании качественных реакций, слабая визуализация наблюдаемых эффектов (осадки, изменение окраски, газовыделение), отсутствие удобной системы сохранения и проверки результатов студентов, а также высокая стоимость большинства коммерческих платформ. На основе проведённого анализа обоснована необходимость создания специализированного приложения, ориентированного именно на раздел качественного анализа.

В проектном разделе разработано полное обеспечение приложения. Сформирована модель данных, включающая сущности пользователей, лабораторных работ, ионов, реагентов, реакций и истории экспериментов. Обоснован и реализован технологический стек на базе React.js (с TypeScript), Node.js, Express.js, MongoDB и MongooseODM. Особое внимание уделено реализации интерактивной визуализации химических реакций с использованием Three.js и CanvasAPI. Разработана многослойная архитектура программного обеспечения, обеспечивающая высокую поддерживаемость и возможность дальнейшего развития системы. Создано удобное адаптивное пользовательское приложение и подробное руководство пользователя для студентов и преподавателей.

В третьем разделе работы был выполнен расчет основных экономических характеристик. По результатам оценки затраты на разработку и внедрение

составляют около 240 тысяч рублей (с учетом хостинга на сервере образовательной организации), а годовая экономия бюджета для учебных заведений была оценена в 616 тысяч рублей, это обосновано значительным сокращением расходов на закупку оборудования и его обслуживания. С учетом этих данных срок окупаемости проекта составляет около 5 месяцев. Эти показатели наглядно демонстрируют целесообразность проекта.

В качестве основных преимуществ разрабатываемого приложения по сравнению с традиционными методами обучения можно выделить: безопасность учебного процесса и легкость логирования действий во время проведения лабораторных работ, это позволяет преподавателю легче и быстрее оценивать качество работы учеников.

Полностью браузерная реализация обеспечивает удобный доступ к лаборатории с любого устройства без установки дополнительного программного обеспечения.

Практическая значимость выполненной работы заключается в создании реального инструмента, который может быть эффективно использован в учебном процессе высших и средних профессиональных образовательных организаций. Приложение особенно полезно для школ, где организация традиционных химических лабораторий затруднена, а также при реализации дистанционных и смешанных форм обучения.

Полученные результаты подтверждают, что разработка виртуальных лабораторий является перспективным направлением цифровизации химического образования. Созданное приложение не только решает текущие проблемы проведения лабораторных работ, но и закладывает основу для дальнейшего развития цифровых образовательных технологий в естественных науках.

Перспективы дальнейшего совершенствования приложения включают:

Расширение набора лабораторных работ на другие разделы химии (количественный анализ, органическая химия, физико–химические методы)[3].

Реализацию алгоритма систематического качественного анализа с

автоматическими подсказками и проверкой хода решения.

Интеграцию с существующими системами электронного обучения вуза (Moodle, ЭИОС и др.).

Добавление элементов геймификации и системы рейтинга студентов.

Разработку версии приложения с поддержкой виртуальной и дополненной реальности[29].

Таким образом, выполненная выпускная квалификационная работа демонстрирует полный цикл проектирования и разработки современного образовательного информационного приложения – от анализа потребностей и проектирования до реализации и экономического обоснования. Результаты работы имеют как теоретическую, так и ярко выраженную практическую ценность и могут быть использованы для реального внедрения в образовательный процесс.

Список литературы

1. Федеральный закон от 29.12.2012 № 273–ФЗ (ред. от 2025 г.) «Об образовании в Российской Федерации» [Электронный ресурс]. URL: https://www.consultant.ru/document/cons_doc_LAW_140174/ (дата обращения: 30.03.2026).
2. Постановление Правительства РФ от 20.04.2024 № 509 «О проведении эксперимента по формированию в электронном виде сведений о студенческих билетах и зачетных книжках» URL: https://storage.consultant.ru/site20/202404/22/pr_220424-509.pdf. (дата обращения: 30.03.2026).
3. Национальный проект «Образование». Федеральный проект «Цифровая образовательная среда» // Министерство просвещения РФ. 2025 [Электронный ресурс]. URL: <https://edu.gov.ru/national-project/projects/cos/> (дата обращения: 30.03.2026).
4. Рынок EdTech в России: итоги 2024–2025 годов // Smart Ranking. 2025 [Электронный ресурс]. URL: <https://edtechs.ru/analitika-i-intervyu/edtech-gynok-vygos-v-2024-godu-na-19/> (дата обращения: 30.03.2026).
5. Виртуальные химические лаборатории как средство повышения эффективности учебного процесса в школе // Мир педагогики и психологии. 2025. № 1 (102) [Электронный ресурс]. URL: <https://scipress.ru/pedagogy/articles/virtualnye-khimicheskie-laboratorii-kak-sredstvo-povysheniya-effektivnosti-uchebnogo-protsesssa-v-shkole.html> (дата обращения: 30.03.2026).
6. PhET Interactive Simulations: Chemistry [Электронный ресурс]. University of Colorado Boulder. URL: <https://phet.colorado.edu/> (дата обращения: 30.03.2026).
7. Labster: Virtual Labs for Chemistry [Электронный ресурс]. URL: <https://www.labster.com/> (дата обращения: 30.03.2026).
8. ChemCollective: Virtual Labs [Электронный ресурс]. Carnegie Mellon University. URL: <https://chemcollective.org/vlabs> (дата обращения: 30.03.2026).

9. Цифровая лаборатория по химии 2025–2026 учебный год [Электронный ресурс]. URL: https://wiki.i-edu.ru/mediawiki/index.php/Цифровая_лаборатория_по_химии_2021 (дата обращения: 30.03.2026).
10. VR Chemistry Lab – виртуальная химическая лаборатория [Электронный ресурс]. URL: <https://vrchemlab.ru/> (дата обращения: 30.03.2026).
11. Виртуальная Химическая Лаборатория VRLine [Электронный ресурс]. URL: <https://www.vr-line.ru/> (дата обращения: 30.03.2026).
12. Обзор виртуальных лабораторий по химии // Chemical42.ru. 2025 [Электронный ресурс]. URL: <https://chemical42.ru/tutorials/informatika-v-khimii/virtualnye-laboratorii-po-khimii/> (дата обращения: 30.03.2026).
13. Пилотное внедрение школьной химической лаборатории в виртуальной реальности // АСИ. 2024 [Электронный ресурс]. URL: <https://asi.ru/news/202490/> (дата обращения: 30.03.2026).
14. Тренды развития EdTech–рынка в России // Вектор экономики. 2024. № 8 [Электронный ресурс]. URL: <http://www.vectoreconomy.ru/images/publications/2024/8/economicsmanagement/Travkina2.pdf> (дата обращения: 30.03.2026).
15. Цифровизация на уроках химии: достижения, проблемы и перспективы. 2026 [Электронный ресурс]. URL: https://www.researchgate.net/publication/392432677_Cifrovizacia_na_urokah_himii_dostizhenia_problemy_i_perspektivy (дата обращения: 30.03.2026).
16. React.js Documentation. 2025 [Электронный ресурс]. URL: <https://react.dev/> (дата обращения: 30.03.2026).
17. Node.js и Express.js: разработка веб–приложений // Официальная документация. 2025.[Электронный ресурс]. URL:<https://nodejs.org/en/docs/>(дата обращения: 30.03.2026).
18. Mongoose [Электронныйресурс]. URL: <https://mongoosejs.com/docs> (дата обращения: 30.03.2026).
19. Canvas API – Mozilla Developer Network (MDN). 2025

- [Электронный ресурс]. URL:
https://developer.mozilla.org/ru/docs/Web/API/Canvas_API (дата обращения: 30.03.2026).
20. Framer Motion: Animation library for React [Электронный ресурс]. URL: <https://www.framer.com/motion/> (дата обращения: 30.03.2026).
21. Рынок образовательных технологий в России: итоги 2025 года // RBC Trends. 2025 [Электронный ресурс]. URL: <https://trends.rbc.ru/trends/education/6952218e9a79473ffc6f6b06> (дата обращения: 30.03.2026).
22. Виртуальные лаборатории в школьном образовании: обзор и перспективы // Педагогика. 2025. [Электронный ресурс]. URL: <https://istina.msu.ru/publications/article/775435756/> (дата обращения: 30.03.2026).
23. Холохонова Л.В. Виртуальные лабораторные работы по физической и коллоидной химии: электронный лабораторный практикум. 2024. [Электронный ресурс]. URL: <https://catalog.inforeg.ru/Inet/GetEzineByID/283512> (дата обращения: 30.03.2026).
24. Интерактивное обучение химии через виртуальные лаборатории. 2024 [Электронный ресурс]. URL: <https://elibrary.ru/item.asp?id=74493966> (дата обращения: 30.03.2026).
25. Beyond Labz: Virtual Chemistry Labs [Электронный ресурс]. URL: <https://beyondlabz.com/> (дата обращения: 30.03.2026).
26. EduWebLabs: Virtual Labs for Chemistry [Электронный ресурс]. URL: <http://eduweblabs.com/> (дата обращения: 30.03.2026).
27. Цифровые лаборатории в школе 2025: лидеры рынка // IT–Concept. 2025 [Электронный ресурс]. URL: <https://it-concept16.ru/blog/articles/kakie-tsifrovye-laboratorii-vybirali-shkoly-v-2025-godu-i-pochemu/> (дата обращения: 30.03.2026).
28. Федеральный закон от 28.02.2025 № 29–ФЗ «О внесении изменений

в Федеральный закон «Об образовании в Российской Федерации»».[Электронный ресурс]. URL: <http://kremlin.ru/acts/bank/51692>(дата обращения: 30.03.2026).

29. Технологии виртуальной реальности в образовании: российский опыт. 2025.[Электронный ресурс]. URL: <https://www.fa.ru/>(дата обращения: 30.03.2026).

30. Review to Analyze and Compare Virtual Chemistry Laboratories // Journal of Chemical Education. 2020.[Электронный ресурс]. URL:<https://eric.ed.gov/?id=EJ1276934>(дата обращения: 30.03.2026).