

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа информационных технологий и робототехники
Направление подготовки 09.03.04 «Программная инженерия»
ООП/ОПОП Разработка программно-информационных систем
Отделение информационных технологий

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

Тема работы
Разработка социальной сети в виде одностраничного веб-приложения с использованием JavaScript-фреймворков

УДК 004.774:316.472.4

Обучающийся

Группа	ФИО	Подпись	Дата
8К93	Криницкий Кирилл Алексеевич		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Мыцко Евгений Алексеевич	к.т.н.		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ОСГН	Гасанов Магеррам Али оглы	д.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД	Мезенцева Ирина Леонидовна			

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Чердынцев Евгений Сергеевич	к.т.н.		

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП/ОПОП

Код компетенции	Наименование компетенции
Универсальные компетенции	
УК(У)-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.
УК(У)-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений.
УК(У)-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде.
УК(У)-4	Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(-ых) языке(-ах).
УК(У)-5	Способен воспринимать межкультурное разнообразие общества в социально-историческом, этическом и философском контекстах
УК(У)-6	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни.
УК(У)-7	Способен поддерживать должный уровень физической подготовленности для обеспечения полноценной социальной и профессиональной деятельности.
УК(У)-8	Способен создавать и поддерживать безопасные условия жизнедеятельности, в том числе при возникновении чрезвычайных ситуаций.
УК(У)-9	Способен проявлять предприимчивость в практической деятельности, в т.ч. в рамках разработки коммерчески перспективного продукта на основе научно-технической идеи.
Общепрофессиональные компетенции	
ОПК(У)-1	Способен применять естественнонаучные и общепрофессиональные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности.
ОПК(У)-2	Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.
ОПК(У)-3	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.
ОПК(У)-4	Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью.
ОПК(У)-5	Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем.
ОПК(У)-6	Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов.

ОПК(У)-7	Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой.
ОПК(У)-8	Способен осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий.
Профессиональные компетенции	
ПК(У)-1	Способен выполнять интеграцию программных модулей и компонент.
ПК(У)-2	Владение навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения.
ПК(У)-3	Способен создавать техническую документацию на продукцию в сфере информационных технологий, управлять технической информацией.
ПК(У)-4	Владение навыками использования операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных.
ПК(У)-5	Способен проводить, оценивать и следить за выполнением концептуального, функционального и логического проектирования систем малого и среднего масштаба и сложности.

Консультанты по разделам выпускной квалификационной работы <i>(с указанием разделов)</i>	
Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Гасанов Магеррам Али оглы, д.э.н., профессор ОСГН
Социальная ответственность	Мезенцева Ирина Леонидовна, старший преподаватель ООД
Названия разделов, которые должны быть написаны на русском и иностранном языке:	
-	

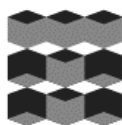
Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	
---	--

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Мыцко Евгений Алексеевич	к.т.н.		

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8К93	Криницкий Кирилл Алексеевич		



Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа информационных технологий и робототехники
Направление подготовки 09.03.04 Программная инженерия
Уровень образования бакалавриат
Отделение информационных технологий
Период выполнения весенний семестр 2022/2023 учебного года

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы**

Обучающийся:

Группа	ФИО
8К93	Криницкий Кирилл Алексеевич

Тема работы:

Разработка социальной сети в виде одностраничного веб-приложения с использованием JavaScript фреймворков
--

Срок сдачи обучающимся выполненной работы:	29.05.2023
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
10.02.2023	Обзор решений и технологий	15
05.03.2023	Проектирование веб-приложения	30
15.04.2023	Программная реализация веб-приложения	30
10.05.2023	Финансовый менеджмент, ресурсоэффективность, ресурсосбережение	15
20.05.2023	Социальная ответственность	10

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Мыцко Евгений Алексеевич	к.т.н.		

СОГЛАСОВАНО:

Руководитель ООП

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Чердынцев Евгений Сергеевич	к.т.н.		

Обучающийся

Группа	ФИО	Подпись	Дата
8К93	Криницкий Кирилл Алексеевич		

РЕФЕРАТ

Выпускная квалификационная работа содержит 83 с., 15 рис., 27 табл., 13 источников.

Ключевые слова: веб-приложение, социальная сеть, фуллстек, фронтенд, бэкенд.

Объектом исследования является веб-приложение в виде социальной сети для корпоративного взаимодействия.

Цель данной работы – разработка веб-приложения при помощи JavaScript фреймворков.

В ходе работы проводились исследования, направленные на изучение требований пользователей, анализ функциональности и особенностей существующих социальных сетей, а также разработку и тестирование прототипов социальной сети.

Результаты исследований и разработок была реализована социальная сеть с необходимыми функциональными возможностями.

Основные конструктивные, технологические и технико-эксплуатационные характеристики разработанной социальной сети – это использование MERN-стека (MongoDB, Express.js, React.js, Node.js) для разработки полнофункционального веб-приложения.

Степень внедрения разработанной социальной сети зависит от планируемой стратегии распространения и маркетинговых усилий. Сеть может быть развернута как внутренняя платформа для ограниченного круга пользователей (например, внутри предприятия), так и как открытая платформа для широкого использования.

Область применения разработанной социальной сети включает: создание онлайн-сообществ для обмена информацией и взаимодействия между пользователями с общими интересами и целями. Построение корпоративных социальных сетей для улучшения внутренней коммуникации и сотрудничества внутри организации.

Экономическая эффективность и значимость работы связаны с потенциалом привлечения пользователей, созданием ценных взаимодействий и коммуникаций, а также возможностью монетизации через рекламу, платные функциональности или премиум-подписки.

В будущем планируется дальнейшее развитие социальной сети, включая:

Расширение функциональности, добавление новых возможностей и улучшение пользовательского опыта.

Реализация дополнительных механизмов безопасности и защиты данных пользователей.

Проведение масштабирования и оптимизации системы для обеспечения стабильной работы при росте числа пользователей.

Исследование и внедрение инновационных технологий, таких как искусственный интеллект и аналитика данных, для улучшения персонализации и рекомендаций.

СОДЕРЖАНИЕ

	С.
Введение	11
Определения, обозначения, сокращения	12
Обзор литературы	13
1 Обзор существующих решений	15
1.1 Анализ социальных сетей	15
1.2 Вывод по разделу	16
2 Обзор технологий разработки веб-приложений	17
2.1 HTML	17
2.2 CSS	17
2.3 JavaScript	17
2.4 Современные наборы технологий для веб-разработки	18
2.4.1 MERN	18
2.4.2 MEAN	19
2.4.3 LAMP	20
2.4.4 MEVN	20
2.5 Набор технологий MERN	21
2.5.1 MongoDB	21
2.5.2 Express.js	21
2.5.3 React.js	22
2.5.4 Node.js	23
2.6 Вывод по разделу	23
3 Проектирование веб-приложения	24
3.1 Требования к проекту	24
3.2 Целевая аудитория	25
3.3 Диаграмма вариантов использования	25
3.4 Архитектура веб-приложения	27
3.5 Структура базы данных	28
3.6 Организация серверной части	30
3.7 Макеты страниц	31
4 Реализация веб-приложения	34
4.1 Разработка базы данных и настройка соединения с сервером	34
4.2 Создание серверного API	35
4.3 Разработка клиентской части приложения	37
5 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	44
5.1 Введение	44
5.2 Потенциальные потребители результатов исследования	44
5.3 Анализ конкурентных технических решений	45
5.4 SWOT-анализ	46
5.5 Планирование работ по научно-техническому исследованию	49
5.5.1 Структура работ в рамках научного исследования	49
5.5.2 Структура работ в рамках научного исследования	50

5.5.3	Разработка графика проведения научного исследования	51
5.6	Бюджет научно-технического исследования (НТИ)	54
5.6.1	Расчет материальных затрат	55
5.6.2	Расчет затрат на специальное оборудование для научных работ	55
5.6.3	Основная заработная плата исполнителя темы	56
5.6.4	Расчет дополнительной заработной платы	58
5.6.5	Отчисления во внебюджетные фонды	59
5.6.6	Накладные расходы	60
5.6.7	Формирование бюджета затрат научно-исследовательского проекта	60
5.7	Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	61
5.8	Вывод по разделу	63
6	Социальная ответственность	67
6.1	Введение	67
6.2	Правовые и организационные вопросы обеспечения безопасности	67
6.2.1	Специальные правовые нормы трудового законодательства	67
6.2.2	Основные эргономические требования к правильному расположению и компоновке рабочей зоны	68
6.3	Производственная безопасность	69
6.3.1	Статические физические перегрузки	70
6.3.2	Психологические нагрузки, вызванные монотонной работой	70
6.3.3	Воздействие электромагнитного поля, излучения, связанного с работой компьютеров и других устройств	71
6.3.4	Отсутствие или недостаток необходимого искусственного освещения	71
6.3.5	Повышенный уровень шума	72
6.3.6	Производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего	72
6.3.7	Неправильное использование электрооборудования и инженерных систем	73
6.4	Экологическая безопасность	74
6.5	Безопасность в чрезвычайных ситуациях	75
6.6	Вывод по разделу	76
	Заключение	77
	Список использованных источников	78
	Приложение А	80
	Приложение Б	81

ВВЕДЕНИЕ

В настоящее время социальные сети стали неотъемлемой частью нашей повседневной жизни. Они предоставляют возможность людям со всего мира общаться, делиться информацией, находить новых друзей и наставников. Среди многообразия существующих социальных платформ особую значимость и актуальность приобретает создание закрытой социальной сети специально для учебных заведений.

Актуальность: в первую очередь, закрытая социальная сеть предоставляет возможность создания образовательной среды, которая сочетает в себе коммуникацию, совместное обучение и совместную работу. Закрытая платформа способствует укреплению связей между студентами, преподавателями и администрацией, создавая единое пространство для обмена знаниями, опытом и идеями. Одним из основных преимуществ закрытой социальной сети является повышение уровня безопасности и конфиденциальности. В отличие от публичных социальных платформ, где информация доступна широкой аудитории, закрытая сеть предоставляет возможность контролировать доступ и защищать личные данные студентов и преподавателей. Это особенно важно в учебных заведениях, где конфиденциальность играет важную роль.

Цель работы: разработка социальной сети в виде веб-приложения для корпоративного взаимодействия.

Задачи для реализации проекта: изучение необходимой литературы; проектирование архитектуры приложения; выбор стека разработки; разработка базы данных и настройка соединения с сервером; создание серверного API; разработка клиентской части приложения.

Объектом исследования является социальная сеть в виде веб-приложения.

Методы исследования: анализ, классификация, проектирование, разработка.

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ

В данной работе применены следующие термины с соответствующими определениями:

Фреймворк – это комплексная структура, предоставляющая основу для разработки приложений или программного обеспечения. Он включает в себя набор инструментов, библиотек и правил, которые помогают разработчикам создавать приложения более эффективно.

Библиотека – это набор предопределенных функций, классов или методов, которые могут быть использованы разработчиками для решения конкретных задач или обеспечения определенной функциональности.

Социальная сеть – это веб-платформа или приложение, которое позволяет пользователям взаимодействовать, обмениваться информацией, создавать и делиться контентом, устанавливать связи и поддерживать социальные отношения онлайн.

MERN – это стек технологий, который объединяет MongoDB, Express.js, React и Node.js для разработки полноценных веб-приложений.

MongoDB – это гибкая и масштабируемая база данных, основанная на документах, которая позволяет хранить и обрабатывать большие объемы данных.

Express.js – это минималистичный и гибкий веб-фреймворк для Node.js, который позволяет разрабатывать веб-приложения и API.

React – это JavaScript библиотека, используемая для создания пользовательских интерфейсов. Она позволяет разрабатывать компоненты, которые обновляются только при изменении данных, что обеспечивает быстрое и эффективное отображение интерфейса.

Node.js – это среда выполнения JavaScript, построенная на движке V8, который используется в браузере Chrome. Она позволяет запускать JavaScript на сервере и разрабатывать серверные приложения.

ОБЗОР ЛИТЕРАТУРЫ

В ходе разработки социальной сети были использованы технологии MERN (MongoDB, Express.js, React.js, Node.js). Также для реализации решения были использованы другие библиотеки.

Документация ReactJS [1] является официальным источником информации о фреймворке React. Она содержит подробные объяснения концепций, инструкции по использованию компонентов и API, а также примеры кода. Документация ReactJS является надежным источником для понимания и использования React в разработке веб-приложений.

Документация Redux [2] предоставляет информацию о библиотеке управления состоянием Redux. Она охватывает основные концепции Redux, такие как хранилище (store), действия (actions), редукторы (reducers) и селекторы (selectors). Документация Redux предлагает подробные примеры использования и позволяет разработчикам эффективно управлять состоянием приложения.

Документация Axios [3] содержит информацию о библиотеке Axios, которая предоставляет простой и удобный API для выполнения HTTP-запросов из JavaScript. Она объясняет основные методы и опции Axios, а также предоставляет примеры использования для отправки запросов на сервер и обработки ответов. Документация Axios является полезным ресурсом при работе с взаимодействием клиента и сервера.

Документация Node.js Express [4] предоставляет информацию о фреймворке Express.js, используемом для разработки серверной части приложений на Node.js. Она содержит руководства, примеры и объяснения API, позволяющие разработчикам создавать мощные и масштабируемые веб-серверы. Документация Node.js Express является надежным ресурсом для понимания и использования Express.js в разработке приложений.

Документация MongoDB [5] предоставляет информацию о базе данных MongoDB, которая используется для хранения информации в приложении. Она охватывает различные аспекты использования MongoDB, включая модели

данных, запросы, индексы и агрегацию данных. Документация MongoDB является официальным источником информации о данной базе данных и предоставляет разработчикам все необходимые сведения для работы с MongoDB в приложениях.

1 ОБЗОР СУЩЕСТВУЮЩИХ РЕШЕНИЙ

1.1 Анализ социальных сетей

В данном разделе будет проведен анализ двух популярных социальных сетей, «ВКонтакте» и «Instagram», с целью выделения их преимуществ и недостатков.

Преимущества «ВКонтакте» состоят в следующем:

- большая аудитория: «ВКонтакте» имеет огромное количество зарегистрированных пользователей, что обеспечивает широкие возможности для поиска и взаимодействия с различными людьми;
- разнообразие функций: платформа предлагает различные функциональные возможности, включая обмен сообщениями, создание и присоединение к группам, загрузку медиа-файлов и т.д.;
- интеграция с другими сервисами: «ВКонтакте» позволяет пользователям интегрировать свои аккаунты в других сервисах, таких как музыкальные платформы или приложения для обмена фотографиями.

Недостатками «ВКонтакте» являются такие характеристики как:

- ограниченная приватность: «ВКонтакте» является открытой платформой, где пользовательская информация доступна широкой аудитории. Это может вызывать опасения с точки зрения конфиденциальности данных;
- низкое качество контента: из-за огромного объема контента, размещаемого пользователями, качество информации и релевантность контента могут быть низкими, что затрудняет поиск и фильтрацию нужной информации.

Платформа «Instagram» обладает следующими преимуществами:

- фокус на визуальном контенте: «Instagram» является платформой, где пользователи активно обмениваются фотографиями и видео. Это создает возможности для креативного самовыражения и визуального взаимодействия;
- сильное сообщество: «Instagram» имеет широкое сообщество пользователей, включая известных личностей, бренды и влиятельных людей, что делает его популярным местом для поиска вдохновения и общения;

– простота использования: «Instagram» имеет простой и интуитивно понятный интерфейс, что делает ее доступной для широкого круга пользователей.

Из недостатков «Instagram» можно отметить:

– ограниченные возможности коммуникации: в отличие от «ВКонтакте», платформа «Instagram» предоставляет ограниченные возможности для обмена сообщениями и комментариями, что может затруднять взаимодействие между пользователями;

– ограниченный функционал: «Instagram» сосредоточен на визуальном контенте и имеет ограниченные функциональные возможности, такие как отсутствие возможности создания групп и организации событий.

1.2 Вывод по разделу

Несмотря на преимущества и популярность «ВКонтакте» и «Instagram», закрытая социальная сеть представляет более предпочтительное решение для учебного заведения. Закрытая социальная сеть обеспечивает более высокий уровень конфиденциальности и безопасности, позволяет более тесное взаимодействие внутри сообщества учебного заведения и создает контролируемую среду для обмена информацией и совместной работы студентов и преподавателей. Кроме того, закрытая социальная сеть может быть специально адаптирована под потребности учебного заведения, предоставляя функциональные возможности, связанные с образовательными процессами и академическими задачами.

2 ОБЗОР ТЕХНОЛОГИЙ РАЗРАБОТКИ ВЕБ-ПРИЛОЖЕНИЙ

2.1 HTML

HTML (HyperText Markup Language) – язык разметки, используемый для создания структуры и представления веб-страниц. Основной задачей HTML является определение структуры и организации содержимого на веб-странице. С помощью тегов и атрибутов HTML, разработчики могут создавать заголовки, параграфы, списки, таблицы и другие элементы, которые обеспечивают семантическую структуру документа. HTML также позволяет встраивать изображения, видео, аудио и другие медиа-элементы на веб-страницы. Благодаря простоте и распространенности HTML, он является основным инструментом для создания содержимого веб-страниц и обеспечивает основу для последующего стилизации и манипулирования с помощью CSS.

2.2 CSS

CSS (Cascading Style Sheets) – язык таблиц стилей, используется для оформления и визуального представления веб-страниц. Задача CSS заключается в разделении структуры (HTML) и внешнего вида (стили) веб-страницы. С помощью CSS разработчики могут определять цвета, шрифты, размеры, отступы, расположение элементов и другие атрибуты, которые определяют визуальное представление контента на странице. CSS обеспечивает возможность создания согласованного и единообразного дизайна для всех страниц сайта, а также позволяет адаптировать представление страницы под различные устройства и размеры экранов (респонсивный дизайн). Благодаря возможностям CSS, веб-разработчики могут значительно повысить привлекательность и пользовательский опыт на своих веб-страницах, создавая уникальные и стильные дизайны.

2.3 JavaScript

JavaScript – это мощный и универсальный язык программирования, который применяется для разработки интерактивных веб-приложений. Он позволяет добавлять динамическое поведение на веб-страницах,

взаимодействовать с пользователями и обрабатывать различные события. Задачи, решаемые JavaScript, включают в себя манипуляцию DOM (Document Object Model) для изменения содержимого и структуры страницы, выполнение асинхронных операций с помощью AJAX (Asynchronous JavaScript and XML) для обмена данными с сервером без перезагрузки страницы, создание анимаций, валидацию форм и многое другое.

JavaScript является клиентским языком программирования, выполняющимся непосредственно в браузере пользователя. Благодаря этому, он обеспечивает богатый пользовательский опыт, позволяя создавать динамические и интерактивные веб-приложения, которые могут реагировать на действия пользователя в режиме реального времени.

Одним из ключевых преимуществ JavaScript является его широкая поддержка во всех современных веб-браузерах. Это позволяет разработчикам создавать кросс-браузерные приложения, которые могут работать на различных платформах и устройствах.

JavaScript также обладает обширной экосистемой, включающей множество библиотек и фреймворков, таких как React.js, Angular.js и Vue.js. Эти инструменты значительно упрощают разработку сложных приложений, предоставляя готовые компоненты, управление состоянием и другие полезные функциональности.

Благодаря своей универсальности, гибкости и возможности взаимодействия с другими технологиями, JavaScript является неотъемлемой частью современной веб-разработки и способствует созданию веб-приложений.

2.4 Современные наборы технологий для веб-разработки

2.4.1 MERN

MERN (MongoDB, Express.js, React.js, Node.js) – MERN является полным стеком JavaScript, включающим MongoDB в качестве базы данных, Express.js для создания серверной части, React.js для клиентской части и

интерфейса пользователя, а также Node.js в качестве среды выполнения на сервере.

Преимущества:

- гибкость и универсальность JavaScript;
- одноязычное решение на всех уровнях разработки;
- широкое сообщество разработчиков и обширная документация;
- масштабируемость и возможность создания высокопроизводительных

приложений.

Недостатки:

- большое количество JavaScript-кода может усложнить поддержку и отладку;
- если возникают проблемы или ограничения с JavaScript, это может отразиться на разработке и функциональности приложения.

2.4.2 MEAN

MEAN также является полным стеком JavaScript, использующим MongoDB в качестве базы данных, Express.js для создания серверной части, Angular.js для клиентской части и Node.js в качестве среды выполнения.

Преимущества:

- единообразие языка программирования на всех уровнях разработки;
- широкий выбор сторонних библиотек и модулей;
- высокая производительность и масштабируемость;
- расширенные возможности для создания сложных и интерактивных

пользовательских интерфейсов.

Недостатки:

- высокий порог вхождения для новых разработчиков из-за сложности Angular.js;
- значительное потребление памяти и ресурсов.

2.4.3 LAMP

LAMP – это популярный стек технологий для разработки веб-приложений, включающий операционную систему Linux, веб-сервер Apache, систему управления базами данных MySQL и язык программирования PHP.

Преимущества:

- широкая поддержка и стабильность;
- большое количество доступных библиотек и расширений для PHP;
- простота развертывания на различных серверах.

Недостатки:

- ограниченная масштабируемость и производительность при большой нагрузке;
- PHP имеет слабую типизацию и требует более внимательного управления безопасностью;
- ограниченные возможности для создания сложных пользовательских интерфейсов.

2.4.4 MEVN

MEVN (MongoDB, Express.js, Vue.js, Node.js) MEVN – это стек технологий, аналогичный MERN, но с использованием фреймворка Vue.js для клиентской части.

Преимущества:

- простота изучения и использования Vue.js;
- высокая производительность и масштабируемость всего стека;
- легкая интеграция и совместимость между компонентами стека;
- широкие возможности для создания интерактивных пользовательских интерфейсов.

Недостатки:

- относительная новизна Vue.js по сравнению с Angular.js и React.js;
- ограниченный выбор библиотек и модулей по сравнению с другими стеками. в виде таблиц, полей, индексов и других объектов, которые необходимы для хранения и обработки информации.

Каждый из этих стеков имеет свои преимущества и недостатки, и выбор наиболее подходящего зависит от требований проекта, опыта команды разработчиков и других факторов. MERN был выбран для данного проекта из-за широкой поддержки JavaScript, гибкости, высокой производительности и возможности создания масштабируемых и современных веб-приложений.

2.5 Набор технологий MERN

2.5.1 MongoDB

MongoDB является документоориентированной базой данных, которая позволяет хранить и управлять данными в формате JSON-подобных документов. Он предоставляет гибкую схему данных, позволяя добавлять и изменять поля в документах без необходимости предварительного определения схемы.

Некоторые преимущества MongoDB в составе MERN включают:

- гибкость данных: MongoDB позволяет разработчикам хранить и обрабатывать различные типы данных без жесткой схемы. Это особенно полезно при разработке веб-приложений, где требуется гибкость в структуре данных;
- масштабируемость: MongoDB обеспечивает горизонтальное масштабирование путем распределения данных по нескольким серверам. Это позволяет обрабатывать большие объемы данных и поддерживать высокую производительность при росте нагрузки на приложение;
- высокая доступность: MongoDB предоставляет механизмы репликации и шардирования для обеспечения высокой доступности данных и отказоустойчивости. Это позволяет удовлетворить требования к непрерывной работе веб-приложений.

2.5.2 Express.js

Express.js – это минималистичный и гибкий веб-фреймворк для разработки серверной части приложений. Он предоставляет простой и интуитивный интерфейс для обработки HTTP-запросов, управления маршрутами и создания обработчиков запросов.

Некоторые преимущества Express.js в составе MERN включают:

- быстрая разработка: Express.js упрощает процесс создания серверной части приложения благодаря его минималистичной структуре и простому синтаксису. Это позволяет разработчикам быстро создавать и настраивать веб-серверы для своих приложений;

- модульность: Express.js предоставляет множество готовых модулей и плагинов, которые облегчают добавление дополнительных функций и расширение возможностей сервера. Разработчики могут выбирать и интегрировать только те модули, которые необходимы для их приложения, что способствует модульной архитектуре и повторному использованию кода;

- Middleware: Express.js предлагает мощную систему промежуточного программного обеспечения (middleware), которая позволяет обрабатывать и изменять запросы и ответы на разных этапах обработки запроса. Это дает разработчикам большую гибкость в управлении потоком данных и добавлении дополнительной логики в приложение.

2.5.3 React.js

React.js – это JavaScript-библиотека для разработки пользовательского интерфейса. Он позволяет создавать компоненты с независимым состоянием и эффективно обновлять только необходимые части интерфейса при изменении данных.

Некоторые преимущества React.js в составе MERN включают:

- компонентный подход: React.js основан на компонентной архитектуре, которая позволяет разрабатывать интерфейс приложения в виде независимых и переиспользуемых компонентов. Это упрощает организацию кода, повторное использование компонентов и обеспечивает легкость поддержки и расширения приложения;

- виртуальный DOM: React.js использует виртуальный DOM для эффективного обновления только измененных частей интерфейса. Это позволяет уменьшить нагрузку на браузер и повысить производительность приложения;

Односторонний поток данных: React.js пропагирует односторонний поток данных, что означает, что данные всегда движутся в одном направлении от верхнего компонента к нижним компонентам. Это облегчает понимание и отслеживание потока данных в приложении.

2.5.4 Node.js

Node.js – это среда выполнения JavaScript, построенная на движке V8 от Google Chrome. Он позволяет выполнять JavaScript-код на сервере и обеспечивает высокую производительность и масштабируемость. Некоторые преимущества Node.js в составе MERN включают:

- высокая производительность: Node.js обеспечивает высокую производительность благодаря асинхронной модели ввода-вывода (non-blocking I/O), которая позволяет обрабатывать множество запросов одновременно без блокировки потоков. Это особенно полезно для обработки многопоточных запросов в реальном времени;

- единая кодовая база: Node.js позволяет использовать один и тот же язык (JavaScript) как на клиентской, так и на серверной стороне.

2.6 Вывод по разделу

В данном разделе был проведен обзор известных стеков технологий для веб-разработки, включая MERN, MEAN, LAMP, и MEVN. Каждый из этих стеков имеет свои особенности и преимущества, которые могут быть важными при выборе технологий для разработки веб-приложений.

MERN является одним из наиболее популярных и мощных стеков технологий. Он предлагает гибкость в хранении и обработке данных с помощью MongoDB, эффективную разработку серверной части с использованием Express.js, мощные возможности создания интерфейса с помощью React.js и высокую производительность на сервере с помощью Node.js. MERN также обладает преимуществами, такими как модульность, удобство разработки, возможность масштабирования и высокая производительность.

3 ПРОЕКТИРОВАНИЕ ВЕБ-ПРИЛОЖЕНИЯ

3.1 Требования к проекту

Для реализации решения, необходимо определить основные требования и функциональность, которую планируется включить в проект. Ниже приведены основные требования к функционалу.

Регистрация и авторизация пользователей:

- пользователи должны иметь возможность создать аккаунт, вводя необходимую информацию, такую как имя, электронная почта и пароль;
- система должна обеспечивать проверку уникальности электронной почты и обработку ошибок при регистрации;
- после регистрации пользователь должен иметь возможность входа в свой аккаунт с помощью своих учетных данных.

Лента новостей:

- система должна предоставлять пользователю ленту новостей, где он может видеть обновления и публикации от пользователей, на которых он подписан;
- публикации в ленте должны быть упорядочены по времени, с наиболее свежими показываються вверху;
- пользователи должны иметь возможность комментировать и ставить лайки под публикациями.

Подписка и отписка от пользователей:

- пользователи должны иметь возможность подписываться на других пользователей, чтобы видеть их обновления в своей ленте новостей;
- пользователи должны иметь возможность отписаться от пользователей, если им больше не интересны их публикации.

Отправка сообщений:

- пользователи должны иметь возможность отправлять личные сообщения другим пользователям;
- система должна обеспечивать отправку и получение сообщений в реальном времени.

Аудио и видеоконференции:

- система должна поддерживать возможность проведения аудио и видеоконференций между пользователями;
- пользователи должны иметь возможность инициировать и присоединяться к конференциям, обмениваться сообщениями и использовать медиафайлы во время конференции.

Редактирование профиля пользователя: пользователи должны иметь возможность редактировать свой профиль, включая информацию о себе, фотографии, контактные данные и настройки конфиденциальности.

Требования к проекту должны быть определены на этапе планирования и разработки, чтобы обеспечить успешную реализацию и удовлетворение потребностей пользователей в функциональности социальной сети.

3.2 Целевая аудитория

Целевая аудитория – студенты и сотрудники Томского политехнического университета.

3.3 Диаграмма вариантов использования

На рисунке 1 представлена диаграмма вариантов использования разрабатываемого приложения.

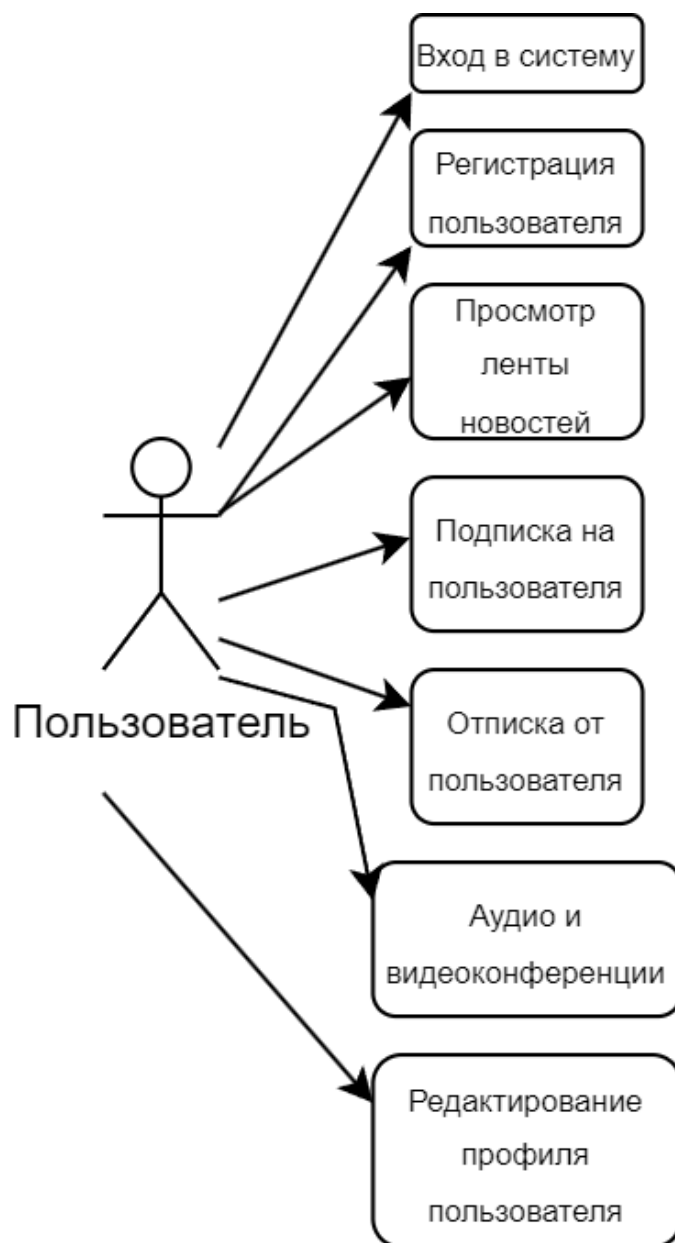


Рисунок 1 – Диаграмма вариантов использования

Данная диаграмма иллюстрирует основные варианты использования социальной сети, начиная с входа в систему и регистрации пользователя, затем позволяет просматривать ленту новостей, подписываться и отписываться от других пользователей, отправлять сообщения, участвовать в аудио и видеоконференциях, а также редактировать профиль пользователя. Эти функциональные возможности обеспечат удобство и взаимодействие пользователей.

3.4 Архитектура веб-приложения

Данное веб-приложение спроектировано на основе модели MVC (Model-View-Controller). Клиент-серверное взаимодействие в архитектуре MVC основано на разделении ответственностей между компонентами системы: моделью, представлением и контроллером.

Модель представляет собой компонент, ответственный за обработку данных и бизнес-логики приложения. Он взаимодействует с базой данных или другими источниками данных, выполняет операции чтения, записи и обновления данных. Когда клиентское приложение отправляет запрос на сервер, модель обрабатывает этот запрос, выполняет необходимые операции и возвращает результат обратно клиенту.

Представление отвечает за отображение данных пользователю и предоставляет пользовательский интерфейс. Оно получает данные от модели и формирует их в удобный для пользователя вид. Представление может быть в виде веб-страниц, графического интерфейса или любого другого способа представления информации пользователю. Когда пользователь взаимодействует с представлением, например, вводит данные или выполняет действия, представление отправляет соответствующие запросы контроллеру.

Контроллер является посредником между моделью и представлением. Он получает запросы от представления, интерпретирует их и вызывает соответствующие методы модели для обработки данных. Контроллер также обрабатывает данные, полученные от модели, и передает их в представление для отображения результатов пользователю. Контроллер обеспечивает координацию работы между моделью и представлением, а также обрабатывает различные события и действия пользователя.

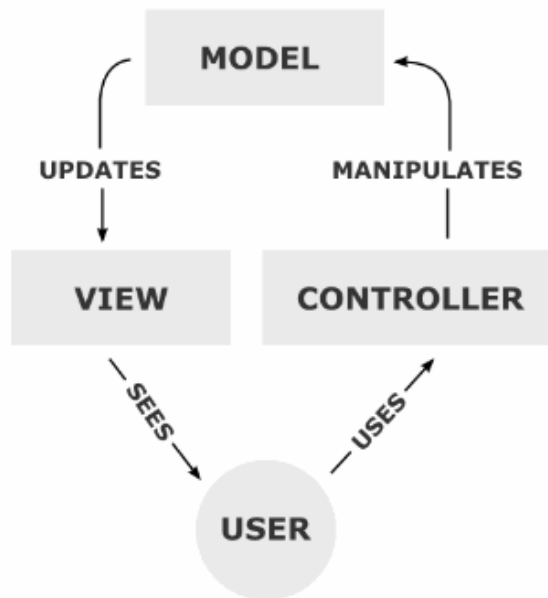


Рисунок 2 – Модель MVC

Клиент-серверное взаимодействие в архитектуре MVC происходит следующим образом: клиентское приложение отправляет запросы на сервер через сетевое соединение, сервер принимает эти запросы и передает их соответствующему контроллеру. Контроллер взаимодействует с моделью для обработки данных, выполняет необходимые операции и возвращает результаты обратно клиенту. Представление получает эти результаты и отображает их пользователю.

3.5 Структура базы данных

Структура базы данных в MongoDB основана на документах и коллекциях. В отличие от традиционных реляционных баз данных, MongoDB является документоориентированной NoSQL базой данных, где данные хранятся в документах формата BSON (бинарный JSON).

Коллекция – это группа документов, которые объединены похожей структурой и хранятся вместе. Коллекции аналогичны таблицам в реляционных базах данных. В MongoDB нет жестких схем для коллекций, поэтому каждая коллекция может содержать документы с разной структурой. Это позволяет гибко изменять схему данных без необходимости обновления всех документов.

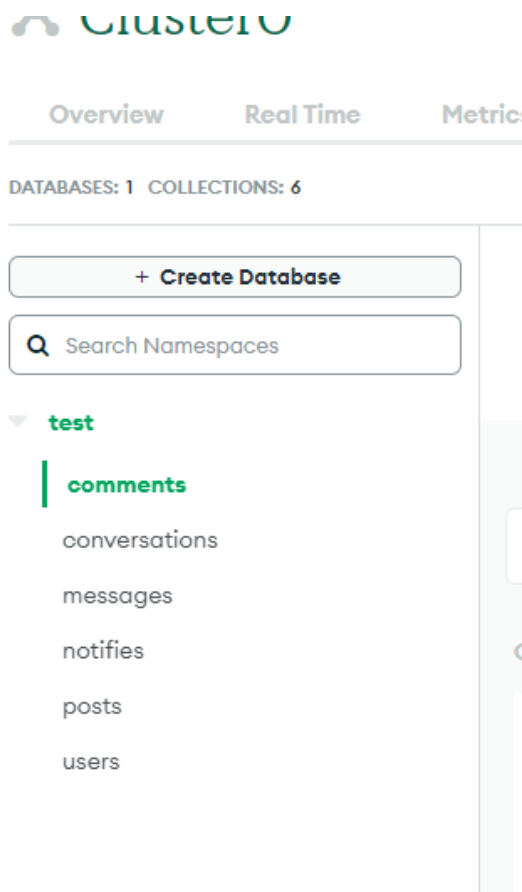


Рисунок 3 – Коллекции в MongoDB

Документ представляет собой набор полей, которые содержат данные в формате ключ-значение. В MongoDB, документы хранятся в BSON, что позволяет эффективно работать с различными типами данных, такими как строки, числа, массивы, вложенные объекты и другие. Документы в MongoDB могут иметь гибкую структуру, то есть поля не обязательно должны быть одинаковыми для всех документов в коллекции. Это позволяет хранить разнородные данные в одной коллекции.

```
_id: ObjectId('644912ccla73872478b3e339')
avatar: "https://res.cloudinary.com/devatchannel/image/upload/v1602752402/avata..."
role: "user"
gender: "male"
mobile: ""
address: ""
story: ""
website: ""
▶ followers: Array
▶ following: Array
▶ saved: Array
fullname: "Danil Zenakov"
username: "danilka1"
email: "danila228@mail.ru"
password: "$2b$12$vYjrVkQMt18KbxNz4KDt0e/ZsRcsmal3vJQSL/G4kRoP4304EHGcm"
createdAt: 2023-04-26T12:02:20.754+00:00
updatedAt: 2023-04-26T12:32:37.616+00:00
__v: 0
```

```
_id: ObjectId('644919e21a73872478b3e35f')
avatar: "https://res.cloudinary.com/devatchannel/image/upload/v1602752402/avata..."
role: "user"
gender: "male"
```

Рисунок 4 – Документ в MongoDB

3.6 Организация серверной части

Серверная часть веб-приложения представляет собой набор маршрутов и контроллеров.

Маршруты в Express позволяют определить пути (URL) и связать их с соответствующими обработчиками запросов. Каждый маршрут может быть связан с функцией-обработчиком, которая будет выполнена при обращении к этому маршруту. Маршруты могут быть определены для различных типов HTTP-запросов, таких как GET, POST, PUT и DELETE.

Контроллеры в Express являются функциями-обработчиками, которые выполняют определенные действия при обработке запросов. Они получают объект запроса (req), который содержит информацию о запросе, и объект ответа (res), с помощью которого можно отправить ответ клиенту.

Контроллеры могут включать логику обработки запроса, взаимодействие с базой данных, выполнение бизнес-логики и т.д.

Подключение к MongoDB осуществляется с помощью соответствующих модулей и библиотек, таких как mongoose. Mongoose предоставляет удобный способ взаимодействия с MongoDB, позволяя определить модели данных и выполнять операции чтения и записи в базу данных. При подключении к MongoDB важно указать правильные параметры подключения, такие как URL базы данных, порт, имя пользователя и пароль.

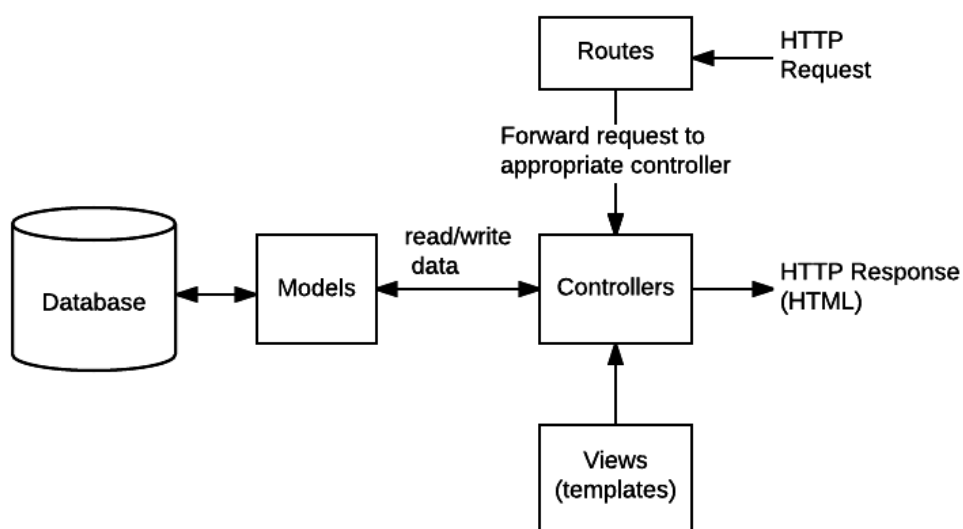


Рисунок 5 – Схема взаимодействия сервера, базы и клиента

3.7 Макеты страниц

Веб-приложение состоит из нескольких разделов, а именно: шапка с меню и поиском пользователей, страница новостей, страница пользователя, раздел мессенджера с выбором диалога и самими диалогами. Макеты данных разделов представлены на рисунках 6-8.

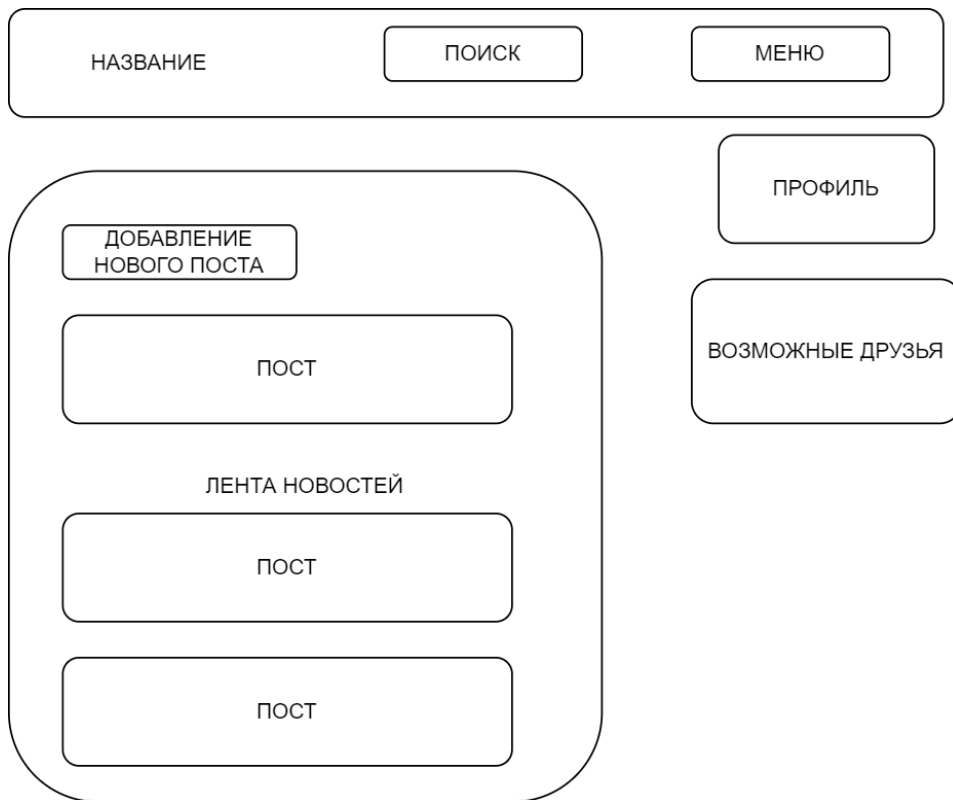


Рисунок 6 – Макет страницы новостей

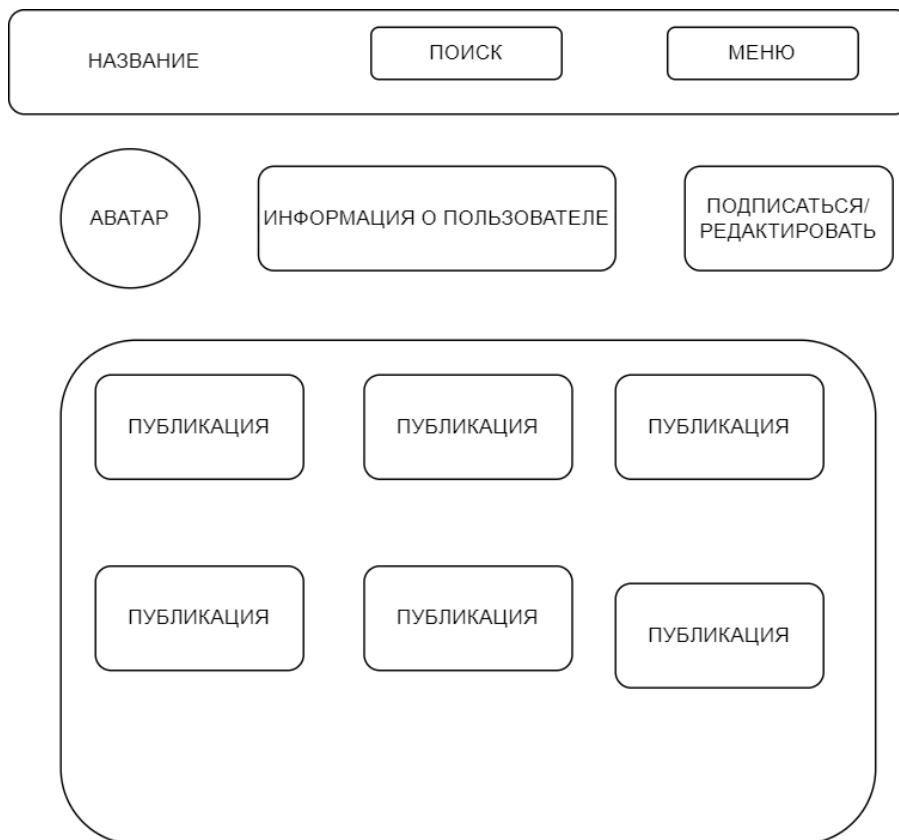


Рисунок 7 – Макет страницы пользователя

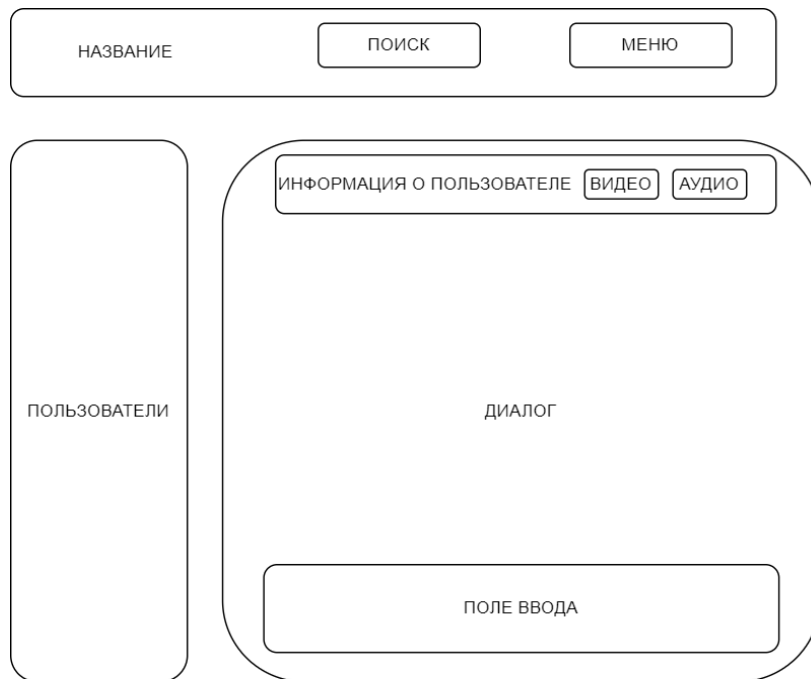


Рисунок 8 – Макет страницы мессенджера

4 РЕАЛИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЯ

4.1 Разработка базы данных и настройка соединения с сервером

Первым шагом для разработки базы данных социальной сети на базе MERN состоит в установке и настройке MongoDB на сервере. Необходимо заранее зарегистрировать собственный кластер на сайте MongoDB, затем можно приступить к подключению базы к серверу.

После установки и настройки MongoDB мы можем продолжить с созданием схемы базы данных. Схема базы данных определяет структуру данных хранящихся в базе данных. Все модели будут основаны на этой схеме. Можно использовать библиотеку mongoose для определения схемы MongoDB. Диаграмма модели пользователя представлена на рисунке 10, код данной модели представлен в приложении А.

User
fullName: String
username: String
emailAddress: String
userPassword: String
userAvatar: String
userRole: String
userGender: String
userMobile: String
userAddress: String
userStory: String
userWebsite: String
userFollowers: [User]
userFollowing: [User]
userSavedPosts: [User]
createdAt: Date
updatedAt: Date

Рисунок 9 – Диаграмма модели пользователя

После определения схемы базы данных мы можем продолжить с разработкой моделей и запросов к базе данных. Модели представляют классы, которые используются для выполнения операций CRUD (Create, Read, Update и Delete) над данными в базе данных. Запросы к базе данных используются для извлечения, изменения и удаления данных из базы данных. На рисунке 11 представлена диаграмма контроллера, который использует модель в предыдущем примере для работы, код фрагмента представлен в приложении Б.

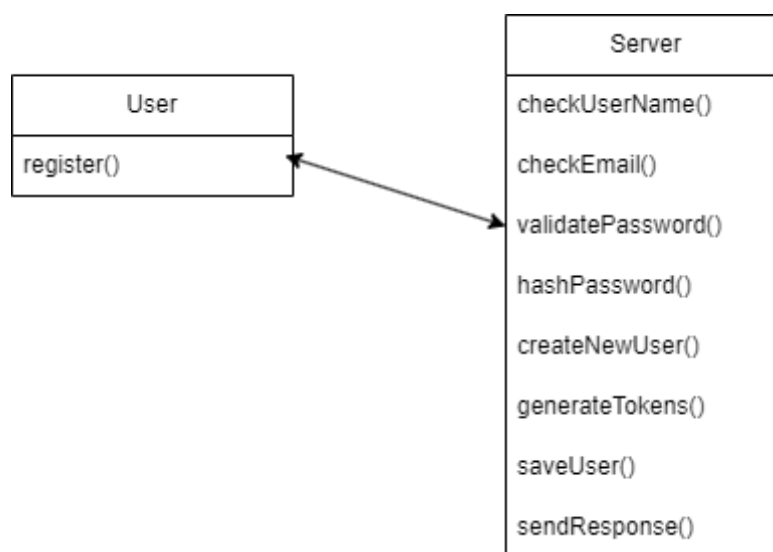


Рисунок 10 – Диаграмма контроллера для работы с моделью пользователя

4.2 Создание серверного API

REST API (Representational State Transfer Application Programming Interface) представляет собой набор HTTP-запросов и ответов, который позволяет клиентам взаимодействовать с серверами через Интернет. Для работы веб-приложения необходимо определить маршруты для запросов с клиента.

В таблицах 1-6 представлены маршруты, определенные при разработке серверной части.

Таблица 1 – Запросы для авторизации и регистрации

Метод	Маршрут	Описание
POST	/register	Регистрация пользователя
POST	/login	Авторизация пользователя
POST	/logout	Выход пользователя из системы
POST	/refresh_token	Генерация нового токена доступа

Таблица 2 – Запросы для работы с комментариями

Метод	Маршрут	Описание
POST	/comment	Создание комментария
PATCH	/comment/:id	Обновление комментария
PATCH	/comment/:id/like	Поставить лайк комментарию
PATCH	/comment/:id/unlike	Убрать лайк с комментария
DELETE	/comment/:id	Удаление комментария

Таблица 3 – Запросы для работы мессенджера

Метод	Маршрут	Описание
POST	/message	Создание сообщения
GET	/conversations	Получение списка бесед
GET	/message/:id	Получение сообщений из беседы
DELETE	/message/:id	Удаление сообщений из беседы
DELETE	/conversation/:id	Удаление беседы

Таблица 4 – Запросы для работы уведомлений

Метод	Маршрут	Описание
POST	/notify	Создание уведомления
DELETE	/notify/:id	Удаление уведомления
GET	/notifies	Получение списка уведомлений
PATCH	/isReadNotify/:id	Пометка уведомления как прочитанного
DELETE	/deleteAllNotify	Удаление всех уведомлений

Таблица 5 – Запросы для работы с постами

Метод	Маршрут	Описание
POST	/posts	Создание поста
GET	/posts	Получение списка постов
PATCH	/post/:id	Обновление поста
GET	/post/:id	Получение информации о посте
DELETE	/post/:id	Удаление поста
PATCH	/post/:id/like	Поставить лайк посту
PATCH	/post/:id/unlike	Убрать лайк с поста
GET	/user_posts/:id	Получение постов пользователя
GET	/post_discover	Получение рекомендуемых постов
PATCH	/savePost/:id	Сохранение поста
PATCH	/unSavePost/:id	Удаление сохраненного поста
GET	/getSavePosts	Получение сохраненных постов

Таблица 6 – Запросы для работы с пользователями

Метод	Маршрут	Описание
GET	/search	Поиск пользователей
GET	/user/:id	Получение информации о пользователе
PATCH	/user	Обновление информации о пользователе
PATCH	/user/:id/follow	Подписка на пользователя
PATCH	/user/:id/unfollow	Отписка от пользователя
GET	/suggestionsUser	Получение рекомендаций по пользователям

Веб-приложение предоставляет широкий набор функциональных возможностей, таких как регистрация и авторизация пользователей, работа с новостной лентой, обмен сообщениями, проведение аудио и видеоконференций, а также редактирование профиля пользователя. Для каждой функциональности имеются соответствующие маршруты, которые обрабатывают запросы клиента и выполняют соответствующие действия на сервере.

4.3 Разработка клиентской части приложения

Разработка интерфейса пользователя (UI) включает в себя создание макета, выбор цветовой схемы, шрифтов и других элементов дизайна. Для разработки UI используется библиотека React, которая позволяет создавать компоненты пользовательского интерфейса, которые могут быть повторно использованы.

Страница авторизации предоставляет пользователю возможность войти в систему, используя свой электронный адрес (почту) и пароль. На этой странице пользователь должен ввести свои учетные данные в соответствующие поля ввода и нажать кнопку "Войти".

После нажатия кнопки "Войти", введенные данные отправляются на сервер для проверки. Если предоставленные учетные данные верны и пользователь существует, система авторизует пользователя и предоставит ему доступ к защищенным функциям приложения. В случае неверных данных или отсутствия пользователя в системе будет выдано сообщение об ошибке.

СОЦИАЛЬНАЯ СЕТЬ

Электронный адрес

Ваши данные никогда не попадут в чужие руки.

Пароль

Показать

Войти

Не зарегистрированы? [Зарегистрироваться](#)

Рисунок 11 – Страница авторизации

Страница регистрации предоставляет пользователю возможность создать учетную запись в системе, используя свой электронный адрес (почту) и пароль. На этой странице пользователь должен заполнить регистрационную форму, вводя необходимые данные, такие как почта и пароль, а также возможно дополнительные данные, такие как имя, фамилия и другие.

После заполнения всех необходимых полей формы, пользователь должен нажать кнопку "Зарегистрироваться" для отправки данных на сервер. В процессе регистрации система проведет проверку введенных данных, включая уникальность почты и правильность формата пароля. Если введенные данные прошли проверку, система создаст новую учетную запись для пользователя и предоставит ему доступ к функциям приложения.

СОЦИАЛЬНАЯ СЕТЬ

Полное имя

Имя пользователя

Электронный адрес

Пароль

 Show

Подтвердите пароль

 Show

Мужской: Женский:

Регистрация

Уже зарегистрированы? [Войти](#)

Рисунок 12 – Страница регистрации

Лента новостей представляет собой динамическую страницу, которая отображает последние посты от пользователей, на которых подписан текущий пользователь. Эта страница предоставляет удобный способ получать актуальные обновления и взаимодействовать с контентом, размещенным другими пользователями.

При открытии ленты новостей пользователь увидит список постов, отсортированных по дате публикации. Каждый пост содержит информацию о его авторе, текстовое содержимое, изображения или видео, а также количество лайков и комментариев. Пользователь может просматривать посты, ставить лайки, оставлять комментарии и взаимодействовать с контентом, который его заинтересовал.

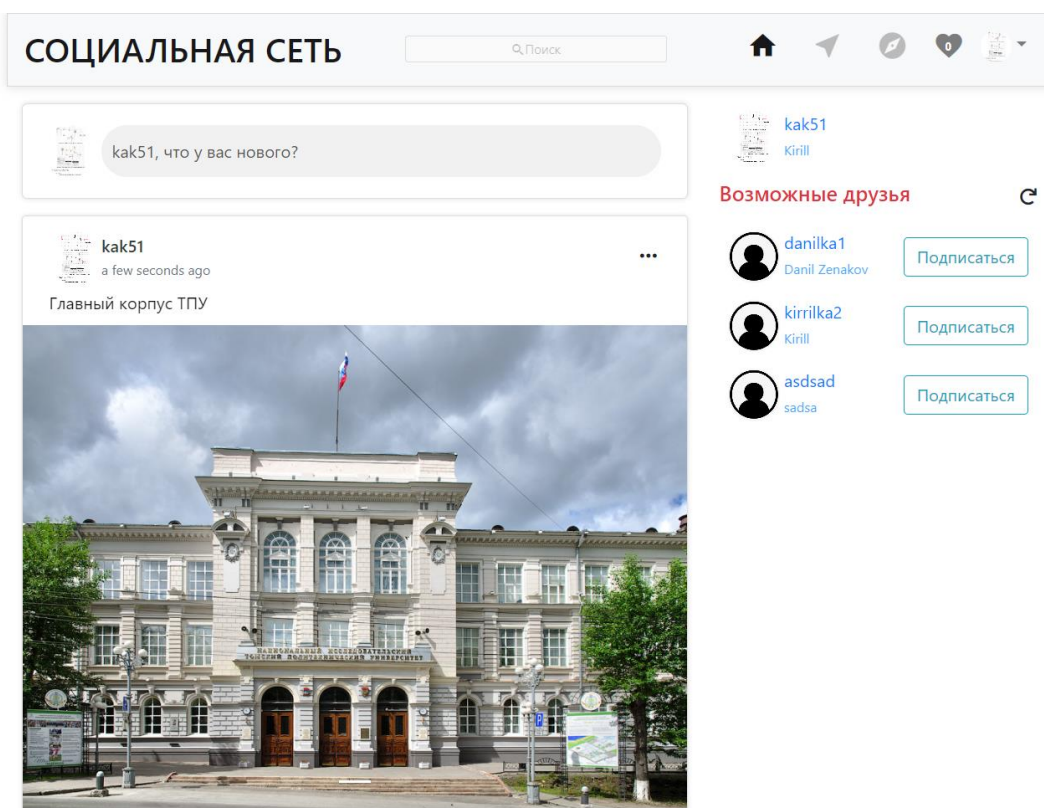


Рисунок 13 – Лента новостей

Страница пользователя представляет собой информационную страницу, которая содержит информацию о конкретном пользователе. На этой странице отображается профиль пользователя, его данные и активность на платформе.

На странице пользователя пользователь может увидеть основную информацию о профиле, такую как имя, фотографию, биографию и другие детали, которые пользователь решил поделиться. Эта информация позволяет другим пользователям получить представление о личности и интересах данного пользователя.

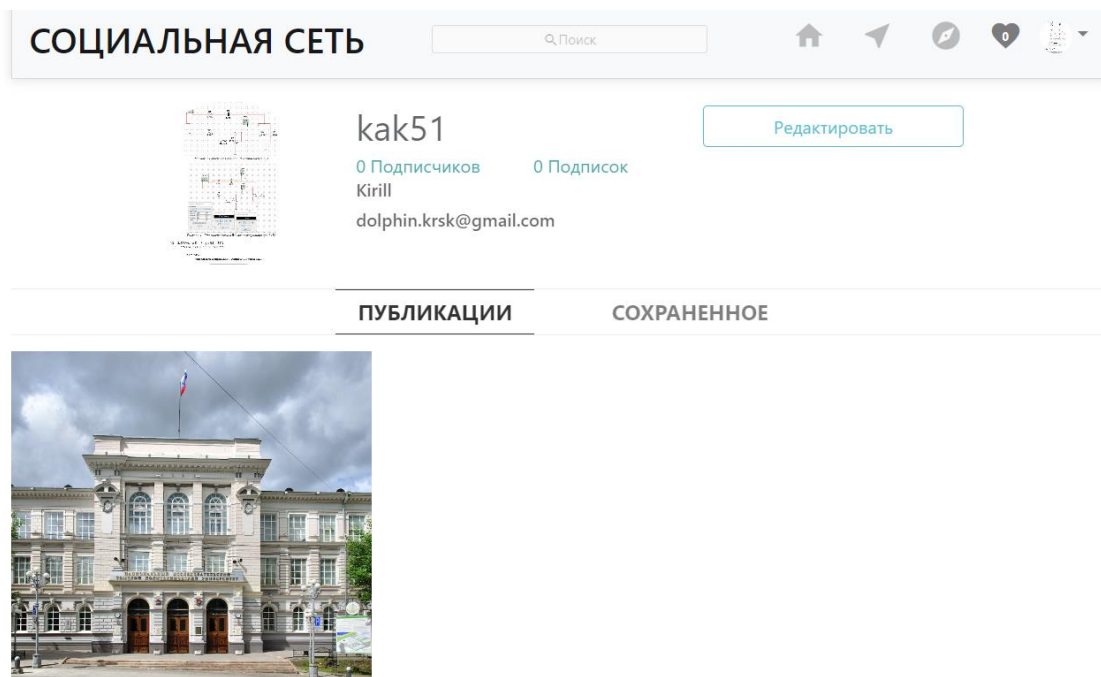


Рисунок 14 – Страница пользователя

Мессенджер – это инструмент коммуникации, который позволяет пользователям обмениваться сообщениями, изображениями, аудио- и видеофайлами. Он предоставляет удобную платформу для общения между пользователями и обмена различными типами контента.

В мессенджере пользователи могут отправлять текстовые сообщения друг другу. Они могут набирать и отправлять сообщения непосредственно в чате, что обеспечивает мгновенную доставку сообщений. Также пользователи могут создавать групповые чаты, где они могут общаться с несколькими людьми одновременно.

СОЦИАЛЬНАЯ СЕТЬ

Поиск

danilka1
Danil Zenakov

Поиск...

danilka1
Как дела?

asdsad
SSASDSAD

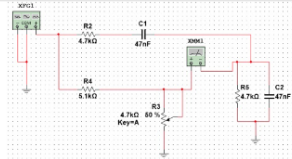


Рисунок 3 - Мостовая схема в RC-автогенераторе.

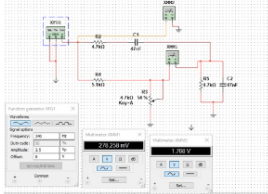


Рисунок 4 - Мостовая схема в RC-автогенераторе при 54%.

Так как $R_{42} = 2.55 \text{ Ом}$, то $K = 3$ при $R_3 = 54\%$
 Так как $R_{44} = 1.275 \text{ Ом}$, то $K = 5$ при $R_3 = 27\%$

$U_{\text{max}} = 278.2 \text{ мВ}$ при $R_3 = 54\%$
 Положение движка при минимальном напряжении совпало с начальным положением $R_3 = 54\%$.

19.05.2023, 01:39:42

kak51

Как дела?

19.05.2023, 18:52:36

Введите сообщение...

Рисунок 15 – Мессенджер

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И
РЕСУРСОСБЕРЕЖЕНИЕ»**

Обучающемуся:

Группа	ФИО
8К93	Криницкому Кириллу Алексеевичу

Школа	ИШИТР	Отделение школы (НОЦ)	ОИТ
Уровень образования	Бакалавриат	Направление/ООП/ОПОП	09.03.04 Программная инженерия

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Оклад руководителя – 30 000 руб. Оклад инженера – 15 000 руб.
2. Нормы и нормативы расходования ресурсов	Премимальный коэффициент руководителя 30%; Премимальный коэффициент инженера 20%; Доплаты и надбавки руководителя 30%; Доплаты и надбавки руководителя 30%; Дополнительной заработной платы 12%; Накладные расходы 16%; Районный коэффициент 1,3%.
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	Коэффициент отчислений на уплату во внебюджетные фонды 30,2 %

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения	Определение потенциального потребителя результатов исследования, SWOT-анализ разработанной стратегии
2. Планирование и формирование бюджета научных исследований	Определение структуры работы. Расчет трудоемкости выполнения работ. Подсчет бюджета исследования
3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	Рассчитать показатели финансовой эффективности, ресурсоэффективности и эффективности исполнения

Перечень графического материала (с точным указанием обязательных чертежей):

1. Оценка конкурентоспособности технических решений
2. Матрица SWOT
3. Альтернативы проведения НИ
4. График проведения и бюджет НИ
5. Оценка ресурсной, финансовой и экономической эффективности НИ

Дата выдачи задания к разделу в соответствии с календарным учебным графиком	
---	--

Задание выдал консультант по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ОСГН	Гасанов Магеррам Али оглы	д.э.н.		

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8К93	Криницкий Кирилл Алексеевич		

5 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

5.1 Введение

Разработкой НИ занимаются студент-инженер и научный руководитель.

Цель данной выпускной квалификационной работы заключается в проектировании и разработке социальной сети в виде веб-приложения.

Целью раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» является определение перспективности и успешности НИ, оценка его эффективности, уровня возможных рисков, разработка механизма управления и сопровождения конкретных проектных решений на этапе реализации.

Для достижения поставленной цели необходимо выполнить следующие задачи:










- оценить коммерческий потенциал и перспективность разработки НИ;
- осуществить планирование этапов выполнения исследования;
- рассчитать бюджет затрат на исследования;
- произвести оценку научно-технического уровня исследования и оценку рисков.

5.2 Потенциальные потребители результатов исследования

Целевой рынок – пользователи всех возрастных категорий, которые используют социальную сеть для общения с друзьями и близкими. Также, результаты работы могут быть полезны для организаций, ассоциаций, коммерческих и государственных учреждений, которые используют социальные сети для привлечения новых клиентов, информирования своей аудитории, проведения кампаний и рекламных акций.

Сегментировать рынок услуг по данной тематике можно по степени использования подобных разработок и виду заинтересованных лиц.

Таблица 7 – Карта сегментирования рынка социальных сетей

		Вид заинтересованного лица			
		Физ. лица, регулярно использующие социальные сети	Физ. лица, пользующиеся социальными сетями редко	Гос. учреждения	Коммерческие учреждения
Степень использования	Высокая	  			 
	Средняя		 		
	Низкая				

 «Facebook»  «Instagram»  Данная разработка

На карте сегментирования рынка социальных сетей обозначены продукты-конкуренты «Facebook» и «Instagram». На основе карты хорошо прослеживается, что государственным и некоторым коммерческим учреждениям удобнее использовать отечественный продукт для информирования целевой аудитории, поэтому в данных сегментах прослеживается наименьшая конкуренция. Именно на эти сегменты и стоит ориентироваться.

5.3 Анализ конкурентных технических решений

Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения позволяет провести оценку сравнительной эффективности научной разработки и определить направления для ее будущего повышения. Целесообразно проводить данный анализ с помощью оценочной карты. Оценочная карта для сравнения конкурентных технических решений представлена в таблице 8.

Таблица 8 – Оценочная карта для сравнения конкурентных технических решений

№	Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
			C ₁	C ₂	C ₃	K ₁	K ₂	K ₃
Технические критерии оценки ресурсоэффективности								
1	Простота использования	0,15	5	5	4	0,75	0,75	0,6
2	Производительность и стабильность работы	0,15	5	5	4	0,75	0,75	0,6
3	Дизайн	0,1	5	4	4	0,5	0,4	0,4
4	Безопасность данных	0,2	5	4	4	1	0,8	0,8
Экономические критерии оценки эффективности								
1	Конкурентоспособность продукта	0,1	3	5	4	0,3	0,5	0,3
2	Цена	0,1	3	5	4	0,5	0,5	0,4
Итого		1	26	28	24	3,8	3,7	3,1

Примечание: C₁ – программный продукт «Facebook»; C₂ – программный продукт «Instagram»; C₃ – данная разработка

Анализ конкурентных технических решений определяется по формуле:

$$K = \sum B_i \times C_i, \quad (1)$$

где K – конкурентоспособность вида;

B_i – вес критерия (в долях единицы);

C_i – балл i-го показателя.

По данным оценочной карты можно сделать вывод, что программный продукт «Instagram» имеет наибольшую конкурентоспособность среди своих конкурентов, но в то же время не лишен такого минуса как безопасность данных пользователя, поскольку у компании, разработавшей данное приложение, несколько раз случались утечки данных в сеть. При разработке и продвижении разрабатываемого программного продукта следует учесть сильные стороны конкурентов и повысить собственную конкурентоспособность за счет качественного дизайна и производительности работы.

5.4 SWOT-анализ

Произведем также в данном разделе SWOT – анализ НИ, позволяющий оценить факторы и явления, способствующие или препятствующие продвижению проекта на рынок.

Сильные стороны – это факторы, которые положительно сказываются на развитии проекта. Сюда обычно включают все, что превращает функционирование в успешную и конкурентную работу.

Слабые стороны – это недостаток, упущение или ограниченность научно-исследовательского проекта, которые препятствуют достижению его целей. Это то, что плохо получается в рамках проекта или где он располагает недостаточными возможностями или ресурсами по сравнению с конкурентами.

Возможности включают в себя любую предпочтительную ситуацию в настоящем или будущем, возникающую в условиях окружающей среды проекта: тенденцию, изменение или предполагаемую потребность, которая поддерживает спрос на результаты проекта и позволяет руководству проекта улучшить свою конкурентную позицию.

Угроза представляет собой любую нежелательную ситуацию, тенденцию или изменение в условиях окружающей среды проекта, которые имеют разрушительный или угрожающий характер для его конкурентоспособности в настоящем или будущем. В качестве угрозы может выступать барьер, ограничение или что-либо еще, что может повлечь за собой проблемы, разрушения, вред или ущерб, наносимый проекту.

На первом этапе SWOT-анализа в таблице 9 были описаны сильные и слабые стороны проекта, выявлены возможности и угрозы реализации НИ.

Таблица 9 – Матрица SWOT-анализа

Сильные стороны	Возможности во внешней среде
<p>С₁. Доступность для государственных и коммерческих учреждений</p> <p>С₂. Удобное, адаптивное веб-приложение</p> <p>С₃. Возможность внедрения решения в виде закрытой корпоративной сети для государственных и коммерческих учреждений.</p> <p>С₄. Востребованность среди пользователей и организаций</p>	<p>В₁. Потенциал для продвижения по всей России</p> <p>В₂. Возрастающая потребность в интернет-коммуникациях</p>
Слабые стороны	Угрозы внешней среды
<p>Сл₁. Зависимость от быстро меняющихся технологий</p> <p>Сл₂. Значительные временные и интеллектуальные затраты</p>	<p>У₁. Отсутствие спроса на данное решение</p> <p>У₂. Давление отечественных конкурентов</p>

Второй этап состоит в выявлении соответствия сильных и слабых сторон научно-исследовательского проекта внешним условиям окружающей среды. Это соответствие или несоответствие должны помочь выявить степень необходимости проведения стратегических изменений. В рамках данного этапа необходимо построить интерактивную матрицу проекта. Ее использование помогает разобраться с различными комбинациями взаимосвязей областей матрицы SWOT. Возможно использование этой матрицы в качестве одной из основ для оценки вариантов стратегического выбора. Каждый фактор помечается либо знаком «+» (означает сильное соответствие сильных сторон возможностям), либо знаком «-» (что означает слабое соответствие); «0» – если есть сомнения в том, что поставить «+» или «-». Интерактивная матрица проекта представлена в таблице 10.

Таблица 10 – Интерактивная матрица сильных и слабых сторон и возможностей

		Сильные стороны				Слабые стороны	
		С ₁	С ₂	С ₃	С ₄	Сл ₁	Сл ₂
Возможности проекта	В ₁	+	+	+	+	-	-
	В ₂	+	+	+	+	+	+

Таблица 11 – Интерактивная матрица сильных, слабых сторон и возможностей

		Сильные стороны				Слабые стороны	
		С ₁	С ₂	С ₃	С ₄	Сл ₁	Сл ₂
Угрозы проекта	У ₁	+	0	+	+	+	0
	У ₂	+	+	+	+	-	+

Самой большой угрозой для проекта является давление со стороны отечественных конкурентов. Что касается слабых сторон, то в будущем необходимо расширить группу разработчиков, привлечь квалифицированных специалистов и консультантов, а также делегировать избыточную работу новым членам команды.

В рамках третьего этапа составляется итоговая матрица SWOT-анализа, представленная в таблице 12.

Таблица 12 – SWOT-анализ

	<p>Сильные стороны научно-исследовательского проекта:</p> <p>С₁. Доступность для государственных и коммерческих учреждений</p> <p>С₂. Удобное, адаптивное веб-приложение</p> <p>С₃. Возможность внедрения решения в виде закрытой корпоративной сети для государственных и коммерческих учреждений</p> <p>С₄. Востребованность среди пользователей и организаций</p>	<p>Слабые стороны научно-исследовательского проекта:</p> <p>Сл₁. Зависимость от быстро меняющихся технологий</p> <p>Сл₂. Значительные временные и интеллектуальные затраты</p>
<p>Возможности:</p> <p>В₁. Потенциал для продвижения по всей России</p> <p>В₂. Возрастающая потребность в интернет-коммуникациях</p>	<p>Большой потенциал проекта заключается в заинтересованности государством, коммерческими организациями по всей стране, с возрастанием потребностей в интернет-коммуникациях это становится все более реализуемым.</p>	<p>Проект предполагает значительные временные затраты на разработку полноценной социальной сети, необходимо реализовать основные необходимые функции, привлечь потенциальных пользователей, параллельно расширять команду разработчиков и функционал приложения.</p>
<p>Угрозы:</p> <p>У₁. Отсутствие спроса на данное решение</p> <p>У₂. Давление отечественных конкурентов</p>	<p>Ставка на возможность внедрения данного решения как закрытая корпоративная сеть для каждой отдельной организации поможет получить преимущество перед конкурентами.</p>	<p>Самой большой угрозой для проекта является давление конкурентов среди крупных компаний в России, из-за ограниченности ресурсов и некоторых каналов продвижения будет непросто развивать проект.</p>

5.5 Планирование работ по научно-техническому исследованию

5.5.1 Структура работ в рамках научного исследования

Планирование комплекса предполагаемых работ осуществляется в следующем порядке:

- определение структуры работ в рамках научного исследования;
- определение участников каждой работы;
- установление продолжительности работ;
- построение графика проведения научных исследований.

Для выполнения научных исследований формируется рабочая группа, в состав которой могут входить научные сотрудники и преподаватели, инженеры, техники и лаборанты, численность групп может варьироваться. По каждому виду запланированных работ устанавливается соответствующая должность исполнителей. Перечень этапов и работ, распределение исполнителей по данным видам работ приведен в таблице 13.

Таблица 13 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№ раб	Содержание работ	Должность исполнителя
Выбор направления исследований	1	Составление и утверждение темы бакалаврской работы	Руководитель Бакалавр
Содержание проекта	2	Определение содержания проекта	Руководитель Бакалавр
Техническое задание	3	Постановка требований к программному обеспечению	Руководитель Бакалавр
	4	Разработка бюджета проекта	Бакалавр
	5	Создание календарного плана-графика	Руководитель Бакалавр
Проектирование программного обеспечения	6	Проектирование серверной части приложения	Бакалавр
	7	Проектирование базы данных	Бакалавр
	8	Проектирование веб-приложения	Бакалавр
Разработка программного обеспечения	9	Разработка веб-приложения	Бакалавр
Тестирование	10	Модульное тестирование	Бакалавр
	11	Интеграционное тестирование	Бакалавр
Документация	12	Подготовка документации	Бакалавр
Оформление отчета по НИР	13	Разработка плана оформления ВКР	Бакалавр
	14	Оформление ВКР	

5.5.2 Структура работ в рамках научного исследования

Трудовые затраты в большинстве случаев образуют основную часть стоимости разработки, поэтому важным моментом является определение трудоемкости работ каждого из участников научного исследования.

Трудоемкость выполнения научного исследования оценивается экспертным путем в человеко-днях и носит вероятностный характер, который зависит от множества трудно учитываемых факторов. Для определения, ожидаемого (среднего) значения трудоемкости $t_{ожі}$ используется следующая формула:

$$t_{ожі} = \frac{3t_{mini} + 2t_{maxi}}{5}, \quad (2)$$

где $t_{ожі}$ – ожидаемая трудоемкость выполнения i -ой работы чел.-дн.;

t_{mini} – минимально возможная трудоемкость выполнения заданной i -ой работы, чел.-дн.;

t_{maxi} – максимально возможная трудоемкость выполнения заданной i -ой работы, чел.-дн.

Исходя из ожидаемой трудоемкости работ, определяется продолжительность каждой работы в рабочих днях T_p , учитывающая параллельность выполнения работ по нескольким исполнителями:

$$T_{pi} = \frac{t_{ожi}}{Ч_i}, \quad (3)$$

где T_{pi} – продолжительность одной работы, раб.дн.;

$t_{ожi}$ – ожидаемая трудоемкость выполнения одной работы, чел.-дн.;

$Ч_i$ – численность исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел.

5.5.3 Разработка графика проведения научного исследования

Наиболее удобным и наглядным представлением проведения научных работ является построение ленточного графика в форме диаграммы Ганта.

Диаграмма Ганта – горизонтальный ленточный график, на котором работы по теме представляются протяженными во времени отрезками, характеризующимися датами начала и окончания выполнения данных работ.

Для удобства построение графика, длительность каждого из этапов работ из рабочих дней следует перевести в календарные дни. Для этого необходимо воспользоваться следующей формулой:

$$T_{ki} = T_{pi} \times k_{\text{кал}}, \quad (4)$$

где T_{ki} – продолжительность выполнения i -й работы в календарных днях;

T_{pi} – продолжительность выполнения i -й работы в рабочих днях;

$k_{\text{кал}}$ – коэффициент календарности.

Коэффициент календарности определяется по следующей формуле:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - (T_{\text{вых}} + T_{\text{пр}})}, \quad (5)$$

где $T_{\text{кал}}$ – общее количество календарных дней в году;

$T_{\text{вых}}$ – общее количество выходных дней в году;

$T_{\text{праз}}$ – общее количество праздничных дней в году.

Для построения таблицы временных показателей проведения НИ был рассчитан коэффициент календарности. С учетом того, что календарных дней в 2022 году 365, количество выходных и праздничных дней для пятидневной рабочей недели равно 118, то коэффициент календарности $k_{\text{кал}} = 1,48$.

Расчеты временных показателей проведения научного исследования приведены в таблице 14.

Таблица 14 – Временные показатели проведения научного исследования

Название работы	Трудоёмкость работ									Исполнители	Длительность работ					
	T _{min} , чел-дн			T _{max} , чел-дн			T _{ож} , чел-дн				в рабочих днях T _{рi}			в календарных днях T _{кi}		
	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3		Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
Составление и утверждение темы бакалаврской работы	1	1	1	2	2	2	1,4	1,4	1,4	Студент, научный руководитель	1	1	1	1	1	1
Определение содержания проекта	1	1	1	2	2	2	1,4	1,4	1,4	Студент, научный руководитель	1	1	1	1	1	1
Постановка требований к программному обеспечению	2	2	3	4	4	5	2,8	2,8	3,8	Студент, научный руководитель	1	1	1	1	1	1
Разработка бюджета проекта	5	6	7	8	8	9	6,2	6,8	7,8	Студент	6	6	7	8	8	10
Создание календарного плана-графика	2	2	2	3	3	4	2,4	2,4	2,8	Студент, научный руководитель	1	1	1	1	1	1
Проектирование серверной части приложения	10	12	14	16	18	20	12,4	14,4	16,4	Студент	12	14	16	17	20	23
Проектирование базы данных	7	9	9	10	12	12	8,2	10,2	10,2	Студент	8	10	10	11	14	14
Проектирование веб-приложения	14	14	18	19	19	20	16	16	18,8	Студент	16	16	18	23	23	26
Разработка вебприложения	36	38	40	44	46	48	39,2	41,2	43,2	Студент	39	41	43	57	60	63
Модульное тестирование	3	4	4	5	5	6	3,8	4,4	4,8	Студент	3	4	4	4	5	5
Интеграционное тестирование	2	3	3	4	5	5	3,2	3,8	3,8	Студент	3	3	3	4	4	4
Подготовка документации	5	7	7	8	8	10	6,2	7,4	8,2	Студент	6	7	8	8	10	11
Разработка плана оформления ВКР	1	2	2	3	3	4	1,8	2,4	2,8	Студент	1	2	2	1	2	2
Оформление ВКР	5	7	7	8	9	10	6,2	7,8	8,2	Студент	6	7	8	8	10	11

В рамках формирования плана научного исследования был разработан календарный план-график выполнения работ. Для иллюстрации календарного плана была использована диаграмма Ганта, указывающая на целесообразность проведения данного исследования (таблица 15).

Таблица 15 – Календарный план-график выполнения работ

№ раб	Название работы	Исполнители	Продолжительность выполнения работ								
			Сентябрь	Октябрь	Ноябрь	Декабрь	Январь	Февраль	Март	Апрель	Май
1	Составление и утверждение темы бакалаврской работы	Ст.									
		НР									
2	Определение содержания проекта	Ст.									
		НР									
3	Постановка требований к программному обеспечению	Ст.									
		НР									
4	Разработка бюджета проекта	Ст.									
5	Составление календарного плана-графика	Ст.									
		НР									
6	Проектирование серверной части	Ст.									
7	Проектирование базы данных	Ст.									
8	Проектирование вебприложения	Ст.									
9	Разработка веб-приложения	Ст.									
10	Модульное тестирование	Ст.									
11	Интеграционное тестирование	Ст.									
12	Подготовка документации	Ст.									
13	Разработка плана оформления ВКР	Ст.									
14	Оформление ВКР	Ст.									

5.6 Бюджет научно-технического исследования (НТИ)

При планировании бюджета разработки необходимо в полной мере отразить все виды расходов, сопутствующих работе над проектом. Бюджет НТИ включает следующие виды затрат:

- материальные затраты;
- затраты на специальное оборудование;

- основная и дополнительная заработная плата;
- социальные отчисления;
- прямые затраты;
- накладные расходы.

5.6.1 Расчет материальных затрат

Расчет материальных затрат осуществляется по формуле:

$$Z_M = (1 + k_T) \times \sum_{i=1}^m C_i \times N_{расхи}, \quad (6)$$

где m – количество видов материальных ресурсов, потребляемых при выполнении научного исследования;

$N_{расхи}$ – количество материальных ресурсов i -го вида, планируемых к использованию при выполнении научного исследования (шт., кг, м, м² и т.д.);

C_i – цена приобретения единицы i -го вида потребляемых материальных ресурсов (руб./шт., руб./кг, руб./м, руб./м² и т.д.);

k_T – коэффициент, учитывающий транспортно-заготовительные расходы.

В таблице 16 приведены результаты расчета материальных затрат.

Таблица 16 – Материальные затраты

Наименование	Единица измерения	Количество			Цена за ед., руб.	Затраты на материалы (Z _М), руб.		
		Исп.1	Исп.2	Исп.3		Исп.1	Исп.2	Исп.3
Интернет	Мбит/с	1	1	1	500	500	500	500
Лампа настольная	Шт	1	1	1	750	750	750	750
Коврик для мыши	Шт	1	1	1	450	450	450	450
Итого, руб.						1 700	1 700	1 700

Таким образом, общие материальные затраты составили 1700 рублей.

5.6.2 Расчет затрат на специальное оборудование для научных работ

В данную статью включают все затраты, связанные с приобретением специального оборудования (приборов, контрольно-измерительной аппаратуры, стендов, устройств и механизмов), необходимого для проведения работ по конкретной теме. Определение стоимости спецоборудования производится по действующим прейскурантам, а в ряде случаев по договорной

цене. При приобретении спецоборудования необходимо учесть затраты по его доставке и монтажу в размере 15% от его цены. Расчет затрат по данной статье представлен в таблице 17.

Таблица 17 – Затраты на приобретение спецоборудования для научных работ

Наименование	Единица измерения	Количество			Цена за ед., тыс. руб.	Затраты на материалы, (ЗМ), руб.		
		Исп. 1	Исп. 2	Исп. 3		Исп. 1	Исп. 2	Исп. 3
Ноутбук	Шт	1	1	1	25	25	25	25
Программное обеспечение WebStorm	Шт	1	1	1	1	1	1	1
Итого, руб.						26	26	26

5.6.3 Основная заработная плата исполнителя темы

В настоящую статью включается основная заработная плата научных и инженерно-технических работников, рабочих макетных мастерских и опытных производств, непосредственно участвующих в выполнении работ по данной теме. Величина расходов по заработной плате определяется исходя из трудоемкости выполняемых работ и действующей системы окладов и тарифных ставок. В состав основной заработной платы включается премия, выплачиваемая ежемесячно из фонда заработной платы в размере 20-30 % от тарифа или оклада. Расчет основной заработной платы приводится в таблице 18.

Таблица 18 – Расчет основной заработной платы

№ п/п	Наименование этапа	Исполнители по категориям	Трудоемкость, чел-дн.			Зарплата плата/ 1 чел-дн.			Всего ЗП по тарифу (окладам), т.руб.		
			Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
1	2	3	4	5	6	7			8	9	10
1	Составление и утверждение темы бакалаврской работы	Ст. НР	1	1	1	3,1			3,1	3,1	3,1
2	Определение содержания проекта	Ст. НР	1	1	1	3,1			3,1	3,1	3,1
3	Постановка требований к программному обеспечению	Ст. НР	2	2	3	2			4	4	6

Продолжение таблицы 18

1	2	3	4	5	6	7	8	9	10
4	Разработка бюджета проекта	Ст.	5	6	7	3,1	15,5	18,6	21,7
5	Создание календарного плана-графика	Ст. НР	2	2	2	2	4	4	4
6	Проектирование серверной части приложения	Ст.	10	12	14	3,1	31	37,2	43,4
7	Проектирование базы данных	Ст.	7	9	9	2	14	18	18
8	Проектирование веб-приложения	Ст.	14	14	18	3,1	43,4	43,4	55,8
9	Разработка веб-приложения	Ст.	36	38	40	3,1	111,6	117,8	124
10	Модульное тестирование	Ст.	3	4	4	2	6	8	8
11	Интеграционное тестирование	Ст.	2	3	3	2	4	6	6
12	Подготовка документации	Ст.	5	7	7	2	10	14	14
13	Разработка плана оформления ВКР	Ст.	1	2	2	2	2	4	4
14	Оформление ВКР	Ст.	5	7	7	2	10	14	14
Итого							261,7	295,2	325,1

Статья включает основную заработную плату работников, непосредственно занятых выполнением проекта, (включая премии, доплаты) и дополнительную заработную плату и рассчитывается по формуле:

$$Z_{ЗП} = Z_{осн} + Z_{доп} , \quad (7)$$

где $Z_{осн}$ – основная заработная плата;

$Z_{доп}$ – дополнительная заработная плата (12–20 % от $Z_{осн}$).

Основная заработная плата руководителя рассчитывается по следующей формуле:

$$Z_{осн} = Z_{дн} \times T_p , \quad (8)$$

где $Z_{осн}$ – основная заработная плата одного работника;

$Z_{осн}$ – продолжительность работ, выполняемых научно-техническим работником, раб. дн.;

$Z_{дн}$ – среднедневная заработная плата работника, руб.

Среднедневная заработная плата рассчитывается по формуле:

$$Z_{дн} = \frac{Z_m \times M}{F_d} , \quad (9)$$

где Z_m – месячный должностной оклад работника, руб.;

M – количество месяцев работы без отпуска в течение года:

при отпуске в 24 раб. дня $M = 11,2$ месяца, 5-дневная неделя;

при отпуске в 48 раб. дней $M = 10,4$ месяца, 6-дневная неделя;

F_d – действительный годовой фонд рабочего времени научно-технического персонала, раб. дн.

Таблица 19 – Баланс рабочего времени

Показатели рабочего времени	Руководитель	Студент
Календарное число дней	365	365
Количество нерабочих дней - выходные дни - праздничные дни	118	118
Потери рабочего времени - отпуск - невыходы по болезни	48 0	72 0
Действительный годовой фонд рабочего времени	199	175

Месячный должностной оклад работника (руководителя):

$$Z_m = Z_{tc} \times (1 + k_{пр} + k_d) \times k_p, \quad (10)$$

где Z_{tc} – заработная плата по тарифной ставке, руб.;

$k_{пр}$ – премиальный коэффициент, равный 0,3 (т.е. 30 процентов от Z_{tc});

k_d – коэффициент доплат и надбавок составляет примерно 0,2 – 0,5;

k_p – районный коэффициент, равный 1,3 (для Томска).

Тарифная заработная плата Z_{tc} находится из произведения тарифной ставки работника 1-го разряда $T_{ci} = 600$ руб. на тарифный коэффициент k_T и учитывается по единой для бюджетной организации тарифной сетке.

Тарифный коэффициент для НР = 1,866, для С = 1,407.

Расчет основной заработной платы представлен в таблице 20.

Таблица 20 – Расчет основной заработной платы

Исполнители	Разряд	k_T	Z_{tc} , руб.	$k_{пр}$	k_d	k_p	Z_m , руб.	$Z_{дн}$, руб.	T_p , раб. дн.	$Z_{осн}$, руб.
Научный руководитель	к.т.н. доцент	1,866	30 000	0,3	0,4	1,3	66 300	3 731,45	7	26 120,15
Студент	Инженер	1,407	15 000	0,3	0,3	1,3	31 200	1 996,50	94	187 671,00
Итого										213 791,15

5.6.4 Расчет дополнительной заработной платы

Дополнительная заработная плата учитывает величину предусмотренных Трудовым кодексом РФ доплат за отклонение от

нормальных условий труда, а также выплат, связанных с обеспечением гарантий и компенсаций (при исполнении государственных и общественных обязанностей, при совмещении работы с обучением, при предоставлении ежегодного оплачиваемого отпуска и т.д.).

Расчет дополнительной заработной платы рассчитывается по формуле:

$$Z_{\text{доп}} = k_{\text{доп}} \times Z_{\text{осн}}, \quad (11)$$

где $k_{\text{доп}}$ – коэффициент дополнительной заработной платы, принятый на стадии проектирования за 0,15.

5.6.5 Отчисления во внебюджетные фонды

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников.

Величина отчислений во внебюджетные фонды определяется исходя из формулы:

$$Z_{\text{внеб}} = k_{\text{внеб}} \times (Z_{\text{осн}} + Z_{\text{доп}}), \quad (12)$$

где $k_{\text{внеб}}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

В соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30,2%.

Результаты расчетов суммы отчислений во внебюджетные фонды представлены в таблице 21.

Таблица 21 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.			Дополнительная заработная плата, руб.		
	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
Руководитель проекта	26 120,15	29 851,60	33 583,05	3 918,00	4 477,74	5 037,45
Студент	187 671,00	215 622,00	235 587,00	28 150,65	32 343,30	35 338,05
$k_{\text{внеб}}$	0,302					
Итого						
Исполнение 1	74 249,66					
Исполнение 2	85 252,98					
Исполнение 3	93 482,75					

5.6.6 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов. Их величина определяется по формуле:

$$Z_{\text{накл}} = \left(\sum \text{статей} \right) \times k_{\text{нр}}, \quad (13)$$

где $k_{\text{нр}}$ – коэффициент, учитывающий накладные расходы.

Величину коэффициента накладных расходов можно взять в размере 16%.

Накладные расходы для исполнения 1 составили:

$$Z_{\text{накл}} = (1\,700 + 26\,000 + 26\,120,15 + 187\,671 + 3\,918 + 28\,150,65 + 74\,249,66) \times 0,16 = 55\,649,51 \text{ руб.}$$

Накладные расходы для исполнения 2 составили:

$$Z_{\text{накл}} = (1\,700 + 26\,000 + 29\,851,6 + 215\,622 + 4\,477,74 + 32\,343,3 + 85\,252,96) \times 0,16 = 63\,239,61 \text{ руб.}$$

Накладные расходы для исполнения 3 составили:

$$Z_{\text{накл}} = (1\,700 + 26\,000 + 33\,583,05 + 235\,587 + 5\,037,45 + 35\,338,05 + 93\,482,75) \times 0,16 = 68\,915,56 \text{ руб.}$$

5.6.7 Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта. Определение бюджета затрат на научно-исследовательский проект приведено в таблице 22.

Таблица 22 – Расчет бюджета затрат НИИ

Наименование статьи	Сумма, руб.			Примечание
	2	3	4	
1. Материальные затраты НИИ	1 700	1 700	1 700	Пункт 4.5.1
2. Затраты на специальное оборудование для научных (экспериментальных) работ	26 000	26 000	26 000	Пункт 4.5.2
3. Затраты по основной заработной плате исполнителей темы	213 791,15	245 473,6	269 170,05	Пункт 4.5.3
4. Затраты по дополнительной заработной плате исполнителей темы	32 068,65	36 821,04	40 375,5	Пункт 4.5.4

Продолжение таблицы 22

1	2	3	4	5
5. Отчисления во внебюджетные фонды	74 249,66	85 252,98	93 482,75	Пункт 4.5.5
6. Затраты на научные и производственные командировки	-	-	-	Отсутствуют
7. Контрагентские расходы	-	-	-	Отсутствуют
8. Накладные расходы	55 649,51	63 239,61	68 915,56	Пункт 4.5.6
9. Бюджет затрат НИИ	403 458,97	458 487,23	499 643,86	

5.7 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

Интегральный показатель финансовой эффективности научного исследования определяется как:

$$I_{\text{фин.р}}^{\text{исп } i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}}, \quad (14)$$

где $I_{\text{фин.р}}^{\text{исп } i}$ – интегральный финансовый показатель разработки;

Φ_{pi} – стоимость i -го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта.

Далее приведен расчет интегральных показателей финансовой устойчивости:

$$I_{\text{фин.р}}^{\text{исп } 1} = \frac{403\,458,97}{499\,643,86} = 0,80;$$

$$I_{\text{фин.р}}^{\text{исп } 2} = \frac{458\,211,12}{499\,643,86} = 0,91;$$

$$I_{\text{фин.р}}^{\text{исп } 3} = \frac{499\,643,86}{499\,643,86} = 1.$$

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить следующим образом:

$$I_{pi} = \sum_{i=1}^n a_i \times b_i, \quad (15)$$

где I_{pi} – интегральный показатель ресурс эффективности для i -го варианта исполнения разработки;

a_i – весовой коэффициент i -го варианта исполнения разработки;

b_i – бальная оценка i -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

n – число параметров сравнения.

Таблица 23 – сравнительная оценка характеристик вариантов исполнения проекта

Объект исследования Критерии	Весовой коэффициент параметра	Исп.1	Исп.2	Исп.3
1. Способствует снижению стоимости разработки	0,1	5	4	3
2. Удобство в эксплуатации	0,2	5	5	5
3. Порог входа	0,15	5	5	5
4. Безопасность	0,15	5	5	5
5. Надежность	0,1	5	5	5
6. Производительность	0,2	5	5	4
7. Доступность	0,1	4	4	4
Итого	1	4,9	4,8	4,5

Далее приведен расчет интегральных показателей финансовой устойчивости:

$$I_{p-исп1} = 0,1 \cdot 5 + 0,2 \cdot 5 + 0,15 \cdot 5 + 0,15 \cdot 5 + 0,1 \cdot 5 + 0,2 \cdot 5 + 0,1 \cdot 4 = 4,9;$$

$$I_{p-исп2} = 0,1 \cdot 4 + 0,2 \cdot 5 + 0,15 \cdot 5 + 0,15 \cdot 5 + 0,1 \cdot 5 + 0,2 \cdot 5 + 0,1 \cdot 4 = 4,8;$$

$$I_{p-исп3} = 0,1 \cdot 3 + 0,2 \cdot 5 + 0,15 \cdot 5 + 0,15 \cdot 5 + 0,1 \cdot 5 + 0,2 \cdot 4 + 0,1 \cdot 4 = 4,5.$$

Интегральный показатель эффективности вариантов исполнения разработки ($I_{испи}$) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{\text{исп1}} = \frac{I_{p-\text{исп1}}}{I_{\text{фин.р}}^{\text{исп1}}} = \frac{4,9}{0,80} = 6,1;$$

$$I_{\text{исп2}} = \frac{I_{p-\text{исп1}}}{I_{\text{фин.р}}^{\text{исп1}}} = \frac{4,8}{0,91} = 5,27;$$

$$I_{\text{исп3}} = \frac{I_{p-\text{исп1}}}{I_{\text{фин.р}}^{\text{исп1}}} = \frac{4,5}{1} = 4,5.$$

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных.

Сравнительная эффективность проекта ($\mathcal{E}_{\text{ср}}$) рассчитывается как:

$$\mathcal{E}_{\text{ср}} = \frac{I_{\text{исп2}}}{I_{\text{исп1}}} \quad (16)$$

Таблица 24 – Сравнительная эффективность разработки

№	Показатели	Исп.1	Исп.2	Исп.3
1	Интегральный финансовый показатель разработки	0,80	0,91	1
2	Интегральный показатель ресурсоэффективности разработки	4,9	4,8	4,15
3	Интегральный показатель эффективности	6,1	5,27	4,5
4	Сравнительная эффективность вариантов исполнения	1	0,86	0,72

После сравнения значений интегральных показателей эффективности можно сделать вывод, что реализация технологии в первом исполнении является более эффективным вариантом решения задачи, поставленной в данной работе с позиции финансовой и ресурсной эффективности. Эффективность первого решения заключается в рациональном распределении времени на разработку данного решения.

5.8 Вывод по разделу

В рамках раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» был оценен коммерческий потенциал разработки, потенциальные потребители результатов исследования, приведен анализ конкурентных решений. Основываясь на результатах проведенного в данном

разделе анализа, можно сделать вывод, что проект является конкурентоспособным и перспективным. Стоимость разработки НИ составила 403 458,97 руб.

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Обучающемуся:

Группа	ФИО
8К93	Криницкому Кириллу Алексеевичу

Школа	ИШИТР	Отделение школы (НОЦ)	ОИТ
Уровень образования	Бакалавриат	Направление/ООП/ОПОП	09.03.04 Программная инженерия

Исходные данные к разделу «Социальная ответственность»:

<p>Введение – Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика) и области его применения – Описание рабочей зоны (рабочего места) при разработке проектного решения/при эксплуатации</p>	<p><i>Объект исследования</i> – социальная сеть в виде одностраничного веб-приложения <i>Область применения</i> – физические лица, использующие социальную сеть для общения, бренды и компании для продвижения своих продуктов, организации и общественные личности для общения с публикой и информировании о своей деятельности, СМИ. <i>Рабочая зона:</i> офис <i>Размеры помещения</i> – 18 м² <i>Количество и наименование оборудования рабочей зоны:</i> ноутбук, стационарный компьютер <i>Рабочие процессы, связанные с объектом исследования, осуществляющиеся в рабочей зоне</i> – разработка и тестирование программного обеспечения</p>
--	---

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<p>1. Правовые и организационные вопросы обеспечения безопасности при разработке проектного решения: – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны.</p>	<p>– ГОСТ 12.2.032-78 устанавливает стандарты для регулирования сидячих рабочих мест – Требования Трудового Кодекса РФ (ФЗ-197) гарантируют рациональную организацию труда в рабочее время</p>
<p>2. Производственная безопасность при разработке проектного решения: – Анализ выявленных вредных и опасных производственных факторов – Обоснование мероприятий по снижению воздействия</p>	<p>Вредные факторы: – Статические физические перегрузки – Психологические нагрузки, вызванные монотонной работой – Воздействие электромагнитного поля, излучения, связанного с работой компьютеров и других устройств – Отсутствие или недостаток необходимого искусственного освещения; – Повышенный уровень шума – Производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего</p> <p>Опасные факторы:</p>

	<p>– Неправильное использование электрооборудования и инженерных систем, что может быть причиной пожара, поражения электрическим током или других аварийных ситуаций.</p> <p>Требуемые средства:</p> <p>– Регулярные проверки оборудования и инженерных систем для обеспечения их надежной и безопасной работы.</p> <p>– Использование гигиенических средств защиты для предотвращения физических нагрузок, таких как эргономические кресла и столы, мониторы с защитой от блеска, клавиатуры и мыши с подставками для запястий и т.д.</p>
3. Экологическая безопасность при разработке проектного решения	<p>Воздействие на селитебную зону, литосферу, гидросферу - утилизация компьютерной, офисной техники; утилизация макулатуры и бытовых отходов</p> <p>Воздействие на атмосферу - выбросы опасных химических соединений при сжигании деталей и комплектующих электронно-вычислительных машин.</p>
4. Безопасность в чрезвычайных ситуациях при разработке проектного решения	<p>Возможные ЧС:</p> <p>– Пожар</p> <p>– Землетрясения</p> <p>Наиболее типичная ЧС:</p> <p>– Пожар</p>

Дата выдачи задания к разделу в соответствии с календарным учебным графиком

Задание выдал консультант по разделу «Социальная ответственность»:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Мезенцева Ирина Леонидовна			

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8К93	Криницкий Кирилл Алексеевич		

6 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

6.1 Введение

Выпускная квалификационная работа представляет собой разработку социальной сети в виде одностраничного веб-приложения. Социальные сети составляют важную часть нашей повседневной жизни, они используются физическими лицами для общения, брендами и компаниями для продвижения своих продуктов, организациями и общественными деятелями для информирования о своей деятельности и коммуникации с публикой, а также СМИ. В данном исследовании будут рассмотрены процессы, связанные с разработкой программного обеспечения для социальной сети, а именно, создание новых функций, тестирование и оптимизация приложения. Результаты данной работы могут быть полезны широкому кругу пользователей социальной сети всех возрастных категорий, которые используют социальную сеть для общения с друзьями и близкими. Также, результаты работы могут быть полезны для организаций, ассоциаций, коммерческих и государственных учреждений, которые используют социальные сети для привлечения новых клиентов, информирования своей аудитории, проведения кампаний и рекламных акций.

Разработка и тестирование программного обеспечения в рамках данной работы осуществляется в рабочей зоне, которая представляет собой офис размером 18 м², оборудованный ноутбуком и стационарным компьютером.

6.2 Правовые и организационные вопросы обеспечения безопасности

6.2.1 Специальные правовые нормы трудового законодательства

При разработке проектного решения необходимо учитывать специальные правовые и организационные вопросы обеспечения безопасности. В частности, при проектировании рабочей зоны необходимо соблюдение правил и стандартов, которые регулируются с помощью законодательного акта "Трудовой кодекс Российской Федерации" от 30.12.2001 N 197-ФЗ.

Далее приведены наиболее важные для соблюдения фрагменты:

- продолжительность рабочего времени не должна превышать 40 часов в неделю;
- в течение рабочего дня должен предоставлен перерыв для отдыха и питания продолжительностью не более двух часов и не менее 30 минут;
- всем работникам предоставляются выходные дни (еженедельный непрерывный отдых);
- список обязанностей, режим работы и размер заработной платы должны быть зафиксированы в трудовом договоре;
- работодатель обязан возмещать вред, причиненный работникам в связи с исполнением ими трудовых обязанностей.

6.2.2 Основные эргономические требования к правильному расположению и компоновке рабочей зоны

Рабочее место должно быть организовано с учетом требований ГОСТ 12.2.032–78 ССБТ, устанавливающим общие эргономические требования к рабочим местам при выполнении работ в положении сидя, а именно:

1. Экран видеомонитора должен находиться от глаз пользователя на расстоянии (600–700) мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов.

2. Конструкция рабочей мебели должна обеспечивать возможность индивидуальной регулировки соответственно росту пользователя и создавать удобную позу для работы.

3. Конструкцией рабочего места должно быть обеспечено выполнение трудовых операций в пределах зоны досягаемости моторного поля.

4. Рабочие столы следует размещать таким образом, чтобы видеодисплейные терминалы были ориентированы боковой стороной к световым проемам, чтобы естественный свет падал преимущественно слева.

5. Часто используемые средства отображения информации, требующие менее точного и быстрого считывания показаний, допускается располагать в вертикальной плоскости под углом $\pm 30^\circ$ от нормальной линии взгляда и в горизонтальной плоскости под углом $\pm 30^\circ$ от сагиттальной плоскости.

При выполнении выпускной квалификационной работы правовых и организационных нарушений по указанным требованиям не было выявлено, рабочее место было оборудовано согласно всем нормам и правилам.

6.3 Производственная безопасность

ГОСТ 12.0.003–2015 устанавливает вредные и опасные факторы, которые могут воздействовать на сотрудника. В таблице 25 перечислены факторы, которые могут возникнуть при работах по проектированию, разработке и тестированию веб-приложения.

Таблица 25 – Возможные опасные и вредные производственные факторы на рабочем месте инженера-программиста

Факторы	Нормативные документы
Статические физические перегрузки	Приказ Минтруда России от 29 октября 2021 г. № 774н «Об утверждении общих требований к организации безопасного рабочего места»
Психологические нагрузки, вызванные монотонной работой Воздействие электромагнитного поля, излучения, связанного с работой компьютеров и других устройств	СанПиН 1.2.3685–21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания
Отсутствие или недостаток необходимого искусственного освещения;	СП 2.2.3670–20 Санитарно-эпидемиологические требования к условиям труда СанПиН 1.2.3685–21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания
Повышенный уровень шума	ГОСТ 12.1.003–2014 ССБТ. Шум. Общие требования безопасности СанПиН 1.2.3685–21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания
Производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего	СанПиН 1.2.3685–21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания
Неправильное использование электрооборудования и инженерных систем, что может быть причиной пожара, поражения электрическим током или других аварийных ситуаций	ГОСТ 12.1.019–2017 Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты (с Поправкой)

В таблице 25 описаны опасные и вредные факторы на рабочем месте программиста, указанные в задании, и соответствующие нормативные документы с требованиями к безопасности и условиям труда.

6.3.1 Статические физические перегрузки

Статические физические перегрузки возникают из-за неправильного оснащения рабочего места, например, стул на рабочем месте не позволяет регулировать высоту и наклон спинки.

Такого рода перегрузки отражаются, преимущественно, на опорно-двигательный аппарат и функциональные системы организма (сердечно-сосудистую, дыхательную и др.). Статические перегрузки могут приводить к таким заболеваниям, как: остеохондроз, сколиоз, варикозное расширение вен, а также общее снижение иммунитета.

Чтобы компенсировать этот фактор, необходимо руководствоваться приказом Минтруда России об организации безопасного рабочего места. Рабочее место должно быть оборудовано креслом с возможностью регулировки его высоты и наклона спинки.

6.3.2 Психологические нагрузки, вызванные монотонной работой

При работе с персональным компьютером негативное влияние на состояние разработчика безусловно оказывает очень большой по объему поток информации, который необходимо воспринимать в единицу времени.

Монотонная работа приводит к существенному снижению тонической активности симпатической нервной системы и повышению активности парасимпатической нервной системы.

Согласно СанПиН 1.2.3685–21, длительность сосредоточенного наблюдения должна составлять от 26 до 50% от времени смены, то есть не более 4-х часов при 8-часовом рабочем дне.

Для снижения вредного воздействия фактора монотонности работы необходимо предусматривать регулярные перерывы работы, а также выполнять физические упражнения.

6.3.3 Воздействие электромагнитного поля, излучения, связанного с работой компьютеров и других устройств

Персональный компьютер имеет свойство подвергать работника вредному электромагнитному излучению, вызывающая рак, потерю памяти, болезни Альцгеймера и Паркинсона, импотенцию, разрушение хрусталика глаза, уменьшение количества красных кровяных телец.

Норма допустимых уровней напряженности полей и излучений регламентируются СанПиН 1.2.3685. Согласно установленным нормам, при уровне напряженности электрических полей не выше 4 кВ/м разрешенное время пребывания в рабочей зоне может составлять до 10,5 часов. Время пребывания в рабочей зоне составляет примерно 8 часов в день, следовательно, уровень электромагнитных излучений на рабочем месте в норме.

6.3.4 Отсутствие или недостаток необходимого искусственного освещения

Недостаточная освещенность приводит к понижению работоспособности, а также может вызвать проблемы со здоровьем, а именно может повлиять на качество зрения работников.

Согласно СП 2.2.3670–20, компьютеры должны быть размещены таким образом, чтобы показатели освещенности не превышали установленных в СанПиН 1.2.3685–21 гигиенических нормативов, которые представлены в таблице 26.

Таблица 26 – Требования к освещению на рабочих местах

Показатель	Норма
Освещенность поверхности рабочего стола	500 лк
Освещенность экрана монитора	Не более 200 лк
Коэффициент пульсации освещенности	Не более 5%

Для снижения влияния фактора недостаточной освещенности необходимо, чтобы уровень естественного освещения рабочего места и яркость дисплея персонального компьютера были приблизительно одинаковыми, так как яркий свет в зоне периферийного зрения заметно

увеличивает глазное напряжение, что приводит к их быстрой утомляемости. При недостаточной освещенности помещения может помочь расширение оконного проема помещения и установка качественных источников искусственного освещения.

6.3.5 Повышенный уровень шума

Охлаждающие вентиляторы, принтеры и кондиционеры на рабочем месте программиста создают шум, который может негативно влияет на его здоровье, работоспособность и производительность. Влияние шума на организм сотрудника могут вызвать усталость, раздражительность, головные боли, нарушение физиологических функций и другие проблемы. Уровень шума не должен превышать 80 дБ(А) в течение 8-часового рабочего дня, как определено СанПиН 1.2.3685–21 и ГОСТ 12.1.003-2014.

Для снижения уровня шума стены и потолок помещений, где установлены компьютеры, могут быть облицованы звукопоглощающими материалами.

6.3.6 Производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего

Микроклимат – это метеорологические условия внутренней среды, определяемые действующими на организм человека сочетаниями температуры, относительно влажности и скорости движения воздуха. Причиной отклонения показателей от установленных норм зачастую является некорректная работа системы вентиляции офисного помещения, которая одновременно влияет и на температуру окружающего воздуха в помещении, на влажность, и на скорость его движения.

Высокий уровень влажности воздушных потоков внутри жилого или производственного помещения способствует развитию заболеваний дыхательной и мочевыделительной системы. Если он находится в условиях повышенной или пониженной температуры воздуха, это вредит организму, поскольку происходит его перегрев или переохлаждение.

Работа программиста относится к категории Ia тяжести труда, поскольку работы выполняются сидя и сопровождаются незначительным физическим напряжением с интенсивностью энергозатрат до 139 Вт/час. На рабочих местах пользователей персональных компьютеров должны обеспечиваться оптимальные параметры микроклимата в соответствии с СанПиНом 1.2.3685–21 для категории тяжести работ I_a (таблица 27).

Таблица 27 – Допустимые величины параметров микроклимата

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность, %	Скорость движения воздуха, м/с
Холодный	22–24	21–25	15–75	0,1
Тёплый	24,1–25	19–26		0,2

Для обеспечения перечисленных параметров требуется в зависимости от погодных условий обеспечить либо кондиционирование воздуха, либо отопление, либо естественную вентиляцию. Фактические значения параметров микроклимата на рабочем месте соответствуют допустимым значениям.

6.3.7 Неправильное использование электрооборудования и инженерных систем

Среди распространенных опасностей при работе с компьютером является поражение электрическим током. Опасность поражения определяется величиной тока проходящего через тело человека или напряжением прикосновения. Электрический ток, проходящий через тело человека, может привести к остановке сердца или даже летальному исходу. Помимо того, нарушения правил работы и эксплуатации электрооборудования может привести к пожару и к другим несчастным случаям. Требования к защите от поражения электрическим током определены в ГОСТ 12.1.019–2017.

Согласно ПУЭ, рабочее помещение инженера-программиста относится к помещениям без повышенной опасности.

Для защиты от поражения электрическим током необходимо:

– перед началом работы убедиться, что кабели не имеют оголённых токоведущих частей.

– убедиться, что все токоведущие части защищены от возможных прикосновений, а металлические корпуса заземлены.

– безопасное расположение токоведущих частей, размещение их вне зоны досягаемости частями тела, конечностями.

– обеспечение надёжными и быстродействующими автоматами и устройствами защитного отключения.

– обеспечение электрозащитными средствами и другими средствами индивидуальной защиты.

Программистам присваивается группа I по электробезопасности, так как они относятся к неэлектротехническому персоналу.

6.4 Экологическая безопасность

Разработка проектного решения может оказывать косвенное влияние на природную среду, в частности на селитебную зону, литосферу и гидросферу в связи с утилизацией компьютерной, офисной техники, а также макулатуры и бытовых отходов. При неправильной утилизации этих отходов может произойти загрязнение почвы, подземных вод и атмосферного воздуха. Например, при сжигании деталей и комплектующих электронно-вычислительных машин могут выделяться опасные химические соединения, которые являются источником загрязнения атмосферы.

Для того чтобы минимизировать негативное воздействие на природную среду, следует использовать методы переработки и утилизации отходов. Необходимо использовать специальные контейнеры для сбора макулатуры и других отходов, а также утилизировать компьютерную и офисную технику через специальные организации. Значительное влияние на экологическую безопасность проекта может оказывать использование материалов с низким содержанием опасных веществ.

Рабочее помещение инженера-программиста относится к IV категории объектов, оказывающих негативное воздействие на окружающую среду.

6.5 Безопасность в чрезвычайных ситуациях

Возможными чрезвычайными ситуациями на рабочем месте инженера-программиста являются природные катаклизмы. С учетом специфики работы и наличием вычислительной техники на рабочем месте инженера-программиста наиболее типичной чрезвычайной ситуацией является пожар.

Причинами возгорания при работе с компьютером могут быть:

- небрежность пользователя ПК;
- неисправность компьютера или электрических сетей;
- воспламенение из-за перегрузки;
- короткое замыкание.

Согласно ГОСТ 12.1.004–91 «Пожарная безопасность. Общие требования», при работе с компьютером необходимо соблюдать следующие нормы пожарной безопасности:

- запрещается подключать к сети количество потребителей, превышающих допустимую нагрузку;
- работы за компьютером проводить только при исправном состоянии оборудования, электропроводки.
- регулярно проверять техническое состояние оборудования, в особенности кабелей. Плохая эксплуатация или загрязнение офисной техники может привести к пожару;
- соблюдать чистоту на рабочем месте. Это поможет потушить пожар на ранней стадии или предотвратит быстрое распространение пожара;
- курить в специально отведенных для этого местах. Перед выбрасыванием окурки нужно тщательно затушить.

Рабочее помещение является пожароопасным и относится к категории В, так как содержит твердые горючие и трудногорючие вещества и материалы. Согласно классификации пожаров по виду горючего материала, установленной статьей 8 Федерального закона от 22.07.2008 N 123-ФЗ, возможный пожар относится к классу Е.

Этаж и рабочее помещение оборудовано следующими средствами пожаротушения: переносные огнетушители, пожарные краны и средства обеспечения их использования.

При появлении пожара, любой, увидевший пожар должен: незамедлительно заявить о данном в пожарную службу по телефонному номеру 01 или 112, заявить о происшествии и соблюдать покой. В случае возникновения пожара в здании автоматически срабатывают датчики пожаротушения, и звуковая система оповещает всех сотрудников о немедленной эвакуации из здания, после чего сотрудники направляются на выход в соответствии с планом эвакуации при пожарах и других ЧС.

6.6 Вывод по разделу

В результате работы над данным разделом были выявлены опасные и вредные факторы, являющиеся потенциально опасными для программиста, работающего над созданием веб-приложения. Возможные вредные и опасные факторы соответствуют нормативным значениям.

Согласно ПУЭ, рабочее помещение инженера-программиста относится к помещениям без повышенной опасности поражения электрическим током и возникновения пожаров.

Рабочее помещение оборудовано в соответствии с требованиями электро- и пожарной безопасности. Работа программиста относится к категории тяжести труда Ia и требует 1 группы по электробезопасности.

Рабочее помещение является пожароопасным и относится к категории В, а возможный пожар относится к классу Е.

В отношении негативного воздействия на окружающую среду, рабочее помещение инженера-программиста относится к IV категории объектов.

Используемое помещение для разработки проекта выпускной квалификационной работы удовлетворяет всем требованиям безопасности и охраны труда.

ЗАКЛЮЧЕНИЕ

В данной выпускной квалификационной работе был проведен обзор различных технологий, необходимых для разработки социальной сети. После анализа и сравнения различных вариантов, был выбран стек технологий MERN (MongoDB, Express.js, React и Node.js) в качестве основы для реализации проекта.

С использованием выбранных технологий была успешно разработана социальная сеть, включающая все необходимые модули и функциональность. Социальная сеть включает в себя функции регистрации и авторизации пользователей, ленту новостей, возможность подписываться на других пользователей, отправлять сообщения, проводить аудио и видеоконференции, а также редактировать профиль пользователя.

Разработка приложения прошла успешно. MongoDB обеспечила гибкое хранение данных и быстрый доступ к ним, Express.js позволил легко организовать серверную часть приложения, а React и Node.js обеспечили отзывчивый и мощный клиентский интерфейс.

В результате работы была создана полноценная социальная сеть, соответствующая заданным требованиям и обеспечивающая удобство использования, безопасность и взаимодействие между пользователями. Такой проект является полезным и актуальным в настоящее время, поскольку социальные сети становятся все более популярными и востребованными в обществе.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 ReactJS документация [Электронный ресурс]. – URL: <https://ru.reactjs.org/> (дата обращения: 15.05.2023).
- 2 Redux документация [Электронный ресурс]. – URL: <https://redux.js.org/> (дата обращения: 24.04.2023).
- 3 Axios документация [Электронный ресурс]. – URL: <https://axios-http.com/ru/> (дата обращения: 25.03.2023).
- 4 Node.js Express документация [Электронный ресурс]. – URL: <https://nodejsdev.ru/doc/express/> (дата обращения: 20.03.2023).
- 5 MongoDB документация [Электронный ресурс]. – URL: <https://www.mongodb.com/docs/> (дата обращения: 31.05.2023).
- 6 Swagger [Электронный ресурс]. – URL: <https://swagger.io/> (дата обращения: 19.03.2023).
- 7 Трудовой кодекс Российской Федерации от 30.12.2001 № 197-ФЗ (ред. от 01.03.2023) // Собрание законодательства РФ. - 07.01.2002. - № 1 (ч. 1). – Ст. 3.
- 8 ГОСТ 12.2.032–78 Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования [Электронный ресурс] // Электронный фонд правовых и нормативно-технических документов. – Режим доступа: <https://docs.cntd.ru/document/1200003913>, свободный. – Загл. с экрана (Дата обращения: 07.05.2023).
- 9 ГОСТ 12.0.003–2015 Система стандартов безопасности труда (ССБТ). Опасные и вредные производственные факторы. Классификация (с Поправками) ГОСТ от 09 июня 2016 г. № 12.0.003-2015 – Текст: непосредственный
- 10 СП 52.13330.2016. Естественное и искусственное освещение: утвержден Приказом Министерства строительства и жилищно-коммунального хозяйства Российской Федерации от 7 ноября 2016 г. N 777/ введен с 8 мая 2017 г. – Текст: непосредственный.

11 ГОСТ 12.1.019–2017 Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты (с Поправкой) [Электронный ресурс] // Электронный фонд правовых и нормативно-технических документов. – Режим доступа: <https://docs.cntd.ru/document/1200161238>, свободный. – Загл. с экрана (Дата обращения: 07.05.2023).

12 Об утверждении критериев отнесения объектов, оказывающих негативное воздействие на окружающую среду, к объектам I, II, III и IV категорий (с изменениями на 7 октября 2021 года) [Электронный ресурс] // Электронный фонд правовых и нормативно-технических документов. – Режим доступа: <https://docs.cntd.ru/document/573292854>, свободный. – Загл. с экрана (Дата обращения: 07.05.2023).

13 Федеральный закон от 22.07.2008 N 123-ФЗ (ред. От 30.04.2021) «Технический регламент о требованиях пожарной безопасности» – Текст: непосредственный.

ПРИЛОЖЕНИЕ А

(обязательное)

Модель пользователя

```
const mongoose = require('mongoose');

const userSchema = new mongoose.Schema({
  fullName: {
    type: String,
    required: true,
    trim: true,
    maxlength: 25
  },
  username: {
    type: String,
    required: true,
    trim: true,
    maxlength: 25,
    unique: true
  },
  emailAddress: {
    type: String,
    required: true,
    trim: true,
    unique: true
  },
  userPassword: {
    type: String,
    required: true
  },
  userAvatar: {
    type: String,
    default:
'https://res.cloudinary.com/devatchannel/image/upload/v1602752402/avatar/avatar_cugq40.png'
  },
  userRole: {type: String, default: 'user'},
  userGender: {type: String, default: 'male'},
  userMobile: {type: String, default: ''},
  userAddress: {type: String, default: ''},
  userStory: {
    type: String,
    default: '',
    maxlength: 200
  },
  userWebsite: {type: String, default: ''},
  userFollowers: [{type: mongoose.Types.ObjectId, ref: 'user'}],
  userFollowing: [{type: mongoose.Types.ObjectId, ref: 'user'}],
  userSavedPosts: [{type: mongoose.Types.ObjectId, ref: 'user'}]
}, {
  timestamps: true
});

module.exports = mongoose.model('User', userSchema);
```


ПРИЛОЖЕНИЕ Б

(обязательное)

Контроллер для работы с пользователями

```
const Users = require('../models/userModel')
const bcrypt = require('bcrypt')
const jwt = require('jsonwebtoken')

const authCtrl = {
  register: async (req, res) => {
    try {
      const { fullName, userName, email, password, gender } = req.body
      let newUserName = userName.toLowerCase().replace(/ /g, '')

      const userByName = await Users.findOne({ username: newUserName })
      if (userByName) return res.status(400).json({ msg: "Имя
пользователя занято." })

      const userByEmail = await Users.findOne({ email })
      if (userByEmail) return res.status(400).json({ msg: "Электронный
адрес занят." })

      if (password.length < 6)
        return res.status(400).json({ msg: "Длина пароля должна быть
не менее 6 символов." })

      const passwordHash = await bcrypt.hash(password, 12)

      const newUser = new Users({
        fullName, userName: newUserName, email, password:
passwordHash, gender
      })

      const accessToken = createAccessToken({ id: newUser._id })
      const refreshToken = createRefreshToken({ id: newUser._id })

      res.cookie('refreshToken', refreshToken, {
        httpOnly: true,
        path: '/api/refresh_token',
        maxAge: 30 * 24 * 60 * 60 * 1000 // 30 days
      })

      await newUser.save()

      res.json({
        msg: 'Пользователь зарегистрирован!',
        accessToken,
        user: {
          ...newUser._doc,
          password: ''
        }
      })
    } catch (err) {
      return res.status(500).json({ msg: err.message })
    }
  },
  login: async (req, res) => {
    try {
      const { email, password } = req.body
```

```

        const user = await Users.findOne({ email })
        .populate("followers following", "avatar userName fullName
followers following")

        if (!user) return res.status(400).json({ msg: "Неверный email или
пароль." })

        const isMatch = await bcrypt.compare(password, user.password)
        if (!isMatch) return res.status(400).json({ msg: "Неверный email
или пароль." })

        const accessToken = createAccessToken({ id: user._id })
        const refreshToken = createRefreshToken({ id: user._id })

        res.cookie('refreshToken', refreshToken, {
            httpOnly: true,
            path: '/api/refresh_token',
            maxAge: 30 * 24 * 60 * 60 * 1000 // 30 days
        })

        res.json({
            msg: 'Вход выполнен успешно!',
            accessToken,
            user: {
                ...user._doc,
                password: ''
            }
        })
    } catch (err) {
        return res.status(500).json({ msg: err.message })
    }
},
logout: async (req, res) => {
    try {
        res.clearCookie('refreshToken', { path: '/api/refresh_token' })
        return res.json({ msg: "Logged out!" })
    } catch (err) {
        return res.status(500).json({ msg: err.message })
    }
},
generateAccessToken: async (req, res) => {
    try {
        const refreshToken = req.cookies.refreshToken
        if (!refreshToken) return res.status(400).json({ msg: "Пожалуйста
войдите." })

        jwt.verify(refreshToken, process.env.REFRESH_TOKEN_SECRET, async
(err, result) => {
            if (err) return res.status(400).json({ msg: "Пожалуйста
войдите." })

            const user = await Users.findById(result.id).select("-
password")
                .populate('followers following', 'avatar userName
fullName followers following')

            if (!user) return res.status(400).json({ msg: "Пользователя
не существует." })

            const accessToken = createAccessToken({ id: result.id })

            res.json({
                accessToken,
                user

```

```
        })
      })
    } catch (err) {
      return res.status(500).json({ msg: err.message })
    }
  }
}

const createAccessToken = (payload) => {
  return jwt.sign(payload, process.env.ACCESS_TOKEN_SECRET, { expiresIn:
'1d' })
}

const createRefreshToken = (payload) => {
  return jwt.sign(payload, process.env.REFRESH_TOKEN_SECRET, { expiresIn:
'30d' })
}

module.exports = auth
```