

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Институт компьютерных и инженерных наук
Кафедра информационных и управляющих систем
Направление подготовки 09.03.02 – Информационные системы и технологии
Направленность (профиль) образовательной программы Информационные системы и технологии

ДОПУСТИТЬ К ЗАЩИТЕ
Зав. кафедрой

_____ А.В. Бушманов

« _____ » _____ 2025 г.

БАКАЛАВРСКАЯ РАБОТА

на тему: Разработка автоматизированного рабочего места менеджера для координации проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ»

Исполнитель
студент группы 1104-об

(подпись, дата)

Р.А. Грицаенко

Руководитель
доцент, канд. физ-мат.
наук

(подпись, дата)

А.В. Павельчук

Консультант:
по безопасности и
экологичности
доцент, канд. техн. наук

(подпись, дата)

А.Б. Булгаков

Нормоконтроль
инженер кафедры

(подпись, дата)

В.Н. Адаменко

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Институт компьютерных и инженерных наук
Кафедра информационных и управляющих систем

УТВЕРЖДАЮ

Зав. кафедрой

_____ А.В. Бушманов

« _____ » _____ 2024 г.

З А Д А Н И Е

К выпускной квалификационной работе студента 1104-об группы Грицаенко Романа Андреевича

1. Тема выпускной квалификационной работы: Разработка автоматизированного рабочего места менеджера для координации проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ»

(утверждена приказом от 14.04.2025 № 980-уч)

2. Срок сдачи студентом законченной работы (проекта) 10.06.2025

3. Исходные данные к выпускной квалификационной работе: формулировка темы ВКР, литературные источники,

4. Содержание выпускной квалификационной работы (перечень подлежащих разработке вопросов): Анализ предметной области и организационная структура предприятия, проектирование автоматизированного рабочего места менеджера, описание разработанного программного обеспечения, Безопасность и экологичность

6. Консультанты по выпускной квалификационной работе (с указанием относящихся к ним разделов) Булгаков Андрей Борисович (Безопасность и экологичность)

7. Дата выдачи задания 02.10.2024

Руководитель выпускной квалификационной работы: Павельчук Анна Владимировна, доцент, канд. физ-мат. наук

(фамилия, имя, отчество, должность, ученая степень, ученое звание)

Задание принял к исполнению (дата): _____

(подпись студента)

РЕФЕРАТ

Работа содержит 80 страниц, 22 рисунка, 4 приложения, 32 источника

HTML, CSS, JAVASCRIPT, JQUERY, BOOTSTRAP, PHP, MODX, REDBEANPHP, MySQL, ИНФОРМАЦИОННАЯ СИСТЕМА, АРМ МЕНЕДЖЕРА, БАЗА ДАННЫХ, ПРОЕКТИРОВАНИЕ, АВТОМАТИЗАЦИЯ УПРАВЛЕНИЯ ПРОЕКТАМИ.

В рамках выпускной квалификационной работы была разработана информационная система, представляющая собой автоматизированное рабочее место менеджера ООО «ВОСТОК-ТЕХНОЛОГИЯ». Целью работы являлось создание программного обеспечения, позволяющего повысить эффективность координации проектов за счёт централизации данных, систематизации задач и упрощения коммуникации между участниками команды.

Были изучены современные подходы к проектированию информационных систем; выбраны и обоснованы технологии реализации; спроектированы архитектура системы, пользовательский интерфейс и модель данных; выполнено тестирование системы, включая модульное, функциональное и нагрузочное тестирование.

Разработанная система предназначена для автоматизации управленческой деятельности и может быть внедрена в ООО «ВОСТОК-ТЕХНОЛОГИЯ» как инструмент повышения производительности труда и снижения ошибок при распределении задач.

НОРМАТИВНЫЕ ССЫЛКИ

В настоящей практической работе использованы ссылки на следующие стандарты и нормативные документы:

СТО СМК 4.2.3.21-2018. Оформление выпускных квалификационных и курсовых работ (проектов);

ГОСТ 19.001-77. Единая система программной документации (ЕСПД). Общие положения;

ГОСТ 19.002-80. ЕСПД. Схемы алгоритмов и программ. Правила выполнения;

ГОСТ 19.003-80. ЕСПД. Схемы алгоритмов и программ. Обозначения условные графические;

ГОСТ 19.102-77. Стадии разработки;

ГОСТ 19.201-78. Техническое задание. Требования к содержанию и оформлению;

ГОСТ 19.404-79. Пояснительная записка. Требования к содержанию и оформлению;

ГОСТ Р 50958-2001. Средства отображения информации индивидуального пользования. Общие эргономические требования и требования безопасности;

ГОСТ Р ИСО 9241-161-2016. Эргономика взаимодействия человек-система. Часть 161. Элементы графического пользовательского интерфейса;

ГОСТ Р 52872-2019. Интернет-ресурсы и другая информация, представленная в электронно-цифровой форме. Приложение для стационарных и мобильных устройств, иные пользовательские интерфейсы;

ГОСТ Р 50923-96. Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения.

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

CSS3	Cascading Style Sheets (каскадная таблица стилей) версии 3
HTML5	HyperText Markup Language (язык гипертекстовой разметки) версии 5
HTTPS	Hyper Text Transfer Protocol Secure (безопасный протокол передачи гипертекста)
IT	Information Technology
JS	JavaScript
JSON	JavaScript Object Notation (объектная нотация JavaScript)
ORM	Object Relational Mapping (объектно-реляционное сопоставление)
PHP	Hypertext Preprocessor (препроцессор гипертекста)
REST	Representational State Transfer (передача состояния представления)
SQL	Structured Query Language (структурированный язык запросов)
UML	Unified Modeling Language (унифицированный язык моделирования)
URL	Uniform Resource Locator (единообразный указатель местонахождения ресурса)
XSS	Cross-Site Scripting (межсайтовый скриптинг)
ГОСТ	Государственный стандарт Российской Федерации
ООО	Общество с ограниченной ответственностью
СУБД	Система управления базами данных

СОДЕРЖАНИЕ

Введение	8
1 Анализ предметной области и организационная структура предприятия	10
1.1 Обоснование актуальности темы выпускной квалификационной работы	10
1.2 Предметная характеристика объекта профессионального исследования	13
1.3 Организационная структура предприятия	15
1.4 Обзор существующих методов решения аналогичных типовых задач	17
1.5 Формулировка задач исследования и общей методики их решения	21
2 Проектирование автоматизированного рабочего места менеджера	26
2.1 Архитектура информационной системы	26
2.2 Выбор технологий реализации	28
2.3 Проектирование пользовательского интерфейса	31
2.3.1 Разграничение прав пользователей в интерфейсе	35
2.4 Реализация функциональных модулей	39
2.5 Проектирование базы данных	43
3 Описание разработанного программного обеспечения	50
3.1 Организация процесса разработки	50
3.2 Реализация клиентской части	51
3.3 Реализация серверной логики	53
3.4 Интеграция с базой данных	56
3.5 Реализация мер информационной безопасности	59
4 Безопасность и экологичность	56
4.1 Безопасность	61
4.2 Экологичность	61
4.3 Чрезвычайные ситуации	63
Заключение	68

Библиографические ссылки	70
Библиографический список	71
Приложение А – Техническое задание	74
Приложение Б – Диаграмма последовательности	78
Приложение В – Диаграмма прецедентов	79
Приложение Г – Диаграмма классов	80

ВВЕДЕНИЕ

Современные условия ведения бизнеса, характеризующиеся высокой конкуренцией, динамичностью рынка и растущей сложностью управленческих процессов, делают автоматизацию ключевым фактором успеха компаний. Особую актуальность приобретают технологии, направленные на оптимизацию работы менеджерского состава, который отвечает за планирование, координацию и контроль выполнения задач. Внедрение систем автоматизации позволяет не только повысить эффективность внутренних процессов, но и обеспечить гибкость управления в условиях постоянно меняющейся внешней среды.

Объектом исследования в данной работе выступает процесс контроля и управления задачами менеджеров ООО «ВОСТОК-ТЕХНОЛОГИЯ» – компании, чья деятельность связана с необходимостью оперативного реагирования на рыночные изменения и координации множества параллельных проектов. Предмет исследования – разработка информационной системы, способной автоматизировать управленческие процессы, минимизировать рутинные операции и обеспечить аналитическую поддержку принятия решений.

Актуальность темы обусловлена рядом факторов. Во-первых, рост объемов данных и усложнение бизнес-процессов требуют перехода от ручного управления к цифровым решениям. Во-вторых, существующие инструменты (например, YouGile, Trello, Jira) часто не соответствуют специфическим требованиям компании, ограничены в аналитических возможностях или создают риски информационной безопасности. В-третьих, интеграция системы автоматизации с корпоративными платформами, такими как Enterprise Resource Planning (далее – ERP) – планирование ресурсов предприятия и Customer Relationship Management (далее – CRM) – система управления взаимоотношениями с клиентами, становится необходимым условием для создания единой информационной среды.

Цель работы – разработать и внедрить автоматизированное рабочее место (АРМ) менеджера, обеспечивающее повышение эффективности рабочих процессов

и снижение ошибок. Для её достижения поставлены следующие задачи:

- провести анализ текущих бизнес-процессов компании и выявить ключевые проблемы в управлении задачами;
- исследовать существующие методы и инструменты автоматизации, определить их недостатки;
- спроектировать архитектуру системы, включая интерфейс, базу данных и интеграционные модули;
- реализовать прототип системы и провести его тестирование;
- оценить эффективность внедрения на основе пилотного проекта и обратной связи сотрудников.

Методологическую основу исследования составляют системный анализ, методы проектирования баз данных и принципы информационной безопасности. Практическая значимость работы заключается в создании решения, адаптированного под уникальные бизнес-процессы ООО «ВОСТОК-ТЕХНОЛОГИЯ», что позволит компании повысить прозрачность управления, сократить сроки выполнения задач и укрепить конкурентные позиции.

1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ОРГАНИЗАЦИОННАЯ СТРУКТУРА ПРЕДПРИЯТИЯ

1.1 Обоснование актуальности темы выпускной квалификационной работы

Актуальность темы «Разработка автоматизированного рабочего места менеджера для координации проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ»» определяется множеством факторов, связанных с изменениями в деловой среде и возрастающими требованиями к эффективности управленческой деятельности. В современных условиях растущей конкуренции и усложнения бизнес-процессов компаниям необходимо внедрять современные инструменты автоматизации для оптимизации работы менеджерского состава.

Разработка системы автоматизации задач менеджера представляет собой создание комплексного программного решения, позволяющего структурировать рабочий процесс, контролировать выполнение задач и повышать общую эффективность управленческой деятельности. Внедрение данной системы в ООО «ВОСТОК-ТЕХНОЛОГИЯ» позволит более эффективно планировать рабочее время менеджеров, отслеживать прогресс по проектам и оперативно реагировать на изменения в рабочих процессах. Это особенно важно в условиях, когда менеджеры сталкиваются с необходимостью одновременного управления несколькими проектами и координацией взаимодействия между различными отделами компании.

Современный менеджер сталкивается с необходимостью обработки большого объема информации, координации множества параллельных задач и соблюдения сроков выполнения проектов. Существующие методы организации работы часто не справляются с возрастающей нагрузкой и не позволяют в полной мере использовать потенциал цифровых технологий. Разработка системы автоматизации позволит оптимизировать процессы управления задачами, минимизировать ошибки и повысить производительность труда менеджеров. Это особенно важно для компаний, работающих в сфере высоких технологий, где своевременное выполнение задач напрямую влияет на конкурентоспособность бизнеса.

Актуальность темы подтверждается значительными изменениями в подходах к организации управленческой деятельности, где требуется внедрение современных технологий для повышения эффективности работы. В условиях роста объемов информации и увеличения сложности координируемых процессов, компании, такие как ООО «ВОСТОК-ТЕХНОЛОГИЯ», должны стремиться к внедрению инструментов, обеспечивающих прозрачность и управляемость бизнес-процессов. Автоматизация задач менеджера позволяет не только повысить эффективность внутренних процессов, но и обеспечить гибкость в управлении, что особенно важно в условиях динамично меняющегося рынка.

Важным аспектом является возможность интеграции системы автоматизации с другими корпоративными системами, используемыми в компании. Это может включать системы документооборота, ERP-системы, корпоративные порталы и другие программные решения. Такая интеграция позволяет создавать единую информационную среду, где все данные о задачах и проектах синхронизируются и доступны для анализа в режиме реального времени. Это особенно важно для компаний, которые стремятся к масштабируемости и гибкости в управлении своими ресурсами, так как позволяет быстро адаптироваться к изменениям внешней среды.

Не менее важно учитывать аспекты информационной безопасности при работе с корпоративными данными. Современные системы автоматизации предлагают надежные механизмы защиты информации, что особенно актуально в условиях ужесточения законодательства о защите персональных данных и роста киберугроз. Разрабатываемая система будет предусматривать современные методы защиты, включая шифрование данных и многофакторную аутентификацию. Это не только снижает риски утечки информации, но и укрепляет доверие сотрудников к использованию новой системы, что способствует более быстрому внедрению и адаптации.

Аналитические возможности системы автоматизации задач играют ключевую роль в повышении эффективности управленческой деятельности. Система позволит

отслеживать загрузку менеджеров, анализировать сроки выполнения задач, выявлять узкие места в рабочих процессах и прогнозировать потребности в ресурсах. Это даст возможность не только оптимизировать текущие процессы, но и разрабатывать стратегические решения по совершенствованию организационной структуры. Например, на основе аналитических данных можно выявить типовые задержки в выполнении задач или неэффективные этапы работы, что позволит предложить конкретные улучшения.

Глобализация и цифровизация бизнеса требуют от компаний адаптации к новым условиям работы. Внедрение современных инструментов автоматизации становится неотъемлемой частью стратегии развития любой прогрессивной компании. Разработка системы автоматизации задач менеджера позволит ООО «ВОСТОК-ТЕХНОЛОГИЯ» повысить эффективность управленческой деятельности и обеспечить соответствие современным требованиям делового сообщества. Это особенно важно для технологических компаний, где скорость принятия решений и точность выполнения задач напрямую влияют на успех бизнеса.

Наконец, внедрение системы автоматизации задач менеджера способствует повышению общей организационной культуры компании. Когда сотрудники видят, что процессы становятся более прозрачными, а их работа лучше организована, это повышает уровень их вовлеченности и мотивации. Это также помогает новым сотрудникам быстрее адаптироваться, так как они получают четкое представление о своих задачах и ответственности.

Таким образом, разработка системы автоматизации контроля и управления задачами менеджера для ООО «ВОСТОК-ТЕХНОЛОГИЯ» является актуальной и необходимой задачей, способствующей повышению эффективности управленческой деятельности, оптимизации бизнес-процессов и успешному развитию компании в современных рыночных условиях. Это создаст основу для дальнейшего совершенствования организационной структуры и устойчивого развития бизнеса, что является важной целью для любой компании, стремящейся к

успеху. Система позволит не только улучшить текущие показатели работы, но и заложить фундамент для долгосрочного роста и масштабирования бизнеса.

1.2 Предметная характеристика объекта профессионального исследования

Объектом профессионального исследования в рамках рассматриваемой темы является весь процесс организации, планирования и контроля выполнения задач менеджерами компании. Это включает начальные этапы постановки задач, их распределение между сотрудниками, отслеживание прогресса выполнения, а также анализ результатов для дальнейшего совершенствования рабочих процессов. Основным объектом исследования включает разработку информационной системы, ориентированной на автоматизацию управления задачами, с акцентом на повышение эффективности работы менеджерского состава.

На этапе разработки системы особое внимание уделяется работе с данными о задачах и проектах. Это ключевой элемент, от которого зависит вся работа системы. Данные о задачах включают в себя такие сведения, как описание задачи, сроки выполнения, ответственные лица, приоритетность, статус выполнения и связанные ресурсы. Основная задача системы в этом аспекте заключается в том, чтобы эффективно собирать, хранить и предоставлять эти данные в удобном для анализа формате, обеспечивая их доступность для всех участников процесса.

Другим важным аспектом исследования является оптимизация бизнес-процессов, связанных с управлением задачами. В ООО «ВОСТОК-ТЕХНОЛОГИЯ» процессы управления задачами включают не только их постановку и контроль выполнения, но и координацию между различными отделами, настройку уведомлений о сроках, автоматизацию отчетности и перераспределение ресурсов при необходимости. Объект исследования предполагает разработку механизма, который автоматизирует и упрощает эти процессы, что в свою очередь позволит сократить время на выполнение рутинных задач и снизить вероятность ошибок, возникающих при ручной обработке данных.

Аналитика и прогнозирование также играют важную роль в рамках предмета

исследования. Система автоматизации задач должна обладать возможностями для детального анализа данных о выполнении задач. Более того, аналитические функции системы будут позволять компании прогнозировать загрузку менеджеров, выявлять узкие места в рабочих процессах и предлагать оптимальные решения для повышения эффективности. Это поможет разрабатывать более точные стратегии управления ресурсами, адаптировать рабочие процессы под текущие потребности и, таким образом, повышать общую производительность труда.

Неотъемлемой частью предмета исследования является также интеграция новой системы с уже существующими в компании программными решениями. На момент разработки в компании могут быть задействованы системы документооборота, ERP-системы, корпоративные порталы и другие информационные платформы, с которыми необходимо обеспечить эффективную работу в единой среде. Интеграция позволит создать единую информационную экосистему, где все данные о задачах и проектах будут синхронизироваться и доступны для анализа в режиме реального времени.

Также стоит отметить важность вопросов информационной безопасности в рамках исследования. Внедрение новой системы подразумевает обработку большого объема корпоративных данных, что требует внедрения современных механизмов защиты информации. Система должна обеспечивать высокий уровень безопасности за счет использования шифрования, защиты от несанкционированного доступа и других инструментов, которые соответствуют современным требованиям законодательства в сфере защиты персональных данных и корпоративной информации.

В итоге, предмет профессионального исследования охватывает целый комплекс задач, связанных с разработкой и внедрением системы автоматизации контроля и управления задачами менеджера. Основная цель исследования – это создание эффективной и безопасной системы, которая будет интегрирована в существующие бизнес-процессы компании и позволит ООО «ВОСТОК-ТЕХНОЛОГИЯ» повысить эффективность работы менеджеров, оптимизировать

использование ресурсов и обеспечить устойчивое развитие бизнеса в условиях растущей конкуренции.

1.3 Организационная структура предприятия

Организационная структура компании ООО «ВОСТОК-ТЕХНОЛОГИЯ» построена таким образом, чтобы обеспечить эффективное управление всеми подразделениями и скоординированную работу различных отделов для достижения стратегических целей. На вершине иерархии находится генеральный директор, который несет ответственность за общее руководство компанией, определение ее стратегии и принятие ключевых решений. Он координирует работу всех подразделений и взаимодействует с партнерами, акционерами и клиентами.

Одним из ключевых подразделений компании является финансовый отдел, который отвечает за финансовую стабильность и устойчивость компании. В его задачи входит управление финансовыми потоками, планирование бюджета, составление финансовой отчетности и взаимодействие с налоговыми органами. В составе отдела работает бухгалтерия, занимающаяся повседневным учетом и финансовым планированием.

Отдел продаж и маркетинга играет важную роль в продвижении продуктов и услуг компании на рынке, а также в работе с клиентами и увеличении продаж. В этом отделе разрабатываются маркетинговые стратегии, проводятся рекламные кампании, осуществляются исследования рынка.

Отдел по работе с клиентами отвечает за управление взаимоотношениями с клиентами, предоставление качественного обслуживания и решения всех возникающих вопросов. Руководитель отдела по работе с клиентами контролирует все процессы, связанные с поддержкой клиентов, в то время как специалисты отдела занимаются обработкой запросов, урегулированием претензий и обеспечением высокого уровня удовлетворенности клиентов.

Особое внимание в компании уделяется IT-направлению, которое курируется отделом разработки и информационных технологий. IT-директор координирует работу программистов и инженеров, которые занимаются созданием,

обслуживанием и развитием IT-инфраструктуры компании. Системные администраторы поддерживают работоспособность всей технической базы компании.

Юридический отдел обеспечивает правовое сопровождение деятельности компании, консультируя руководство по вопросам правоприменения и контроля за соблюдением правовых норм. Юристы отдела занимаются подготовкой и проверкой договоров, а также решают все правовые вопросы, связанные с деятельностью компании. Организационная структура представлена на рисунке 1.



Рисунок 1 – Организационная структура компании ООО «ВОСТОК-ТЕХНОЛОГИЯ»

Таким образом, организационная структура компании ООО «ВОСТОК-ТЕХНОЛОГИЯ» обеспечивает слаженную работу всех подразделений, что способствует достижению стратегических целей компании и поддержанию высокой эффективности внутренних бизнес-процессов. Четкое распределение обязанностей между отделами, координация их деятельности и внедрение современных инструментов автоматизации позволяют оптимизировать рабочие процессы, повышать производительность труда менеджеров и обеспечивать своевременное выполнение задач. Это создает прочную основу для устойчивого развития компании, укрепления ее позиций на рынке и успешной реализации долгосрочных планов.

1.4 Обзор существующих методов решения аналогичных типовых задач

Обзор существующих методов решения задач автоматизации контроля и управления задачами менеджера показывает, что на рынке представлено множество подходов, каждый из которых имеет свои особенности, преимущества и недостатки. В зависимости от потребностей компании и специфики ее деятельности выбираются различные инструменты, которые помогают оптимизировать процессы планирования, распределения и контроля выполнения задач. Рассмотрим наиболее популярные решения, включая современные системы управления задачами, такие как YouGile, Trello, Jira, Asana и другие.

YouGile представляет собой популярную российскую систему управления задачами, который предлагает удобный интерфейс и широкий функционал для управления задачами и проектами. Система позволяет создавать доски задач, назначать ответственных, устанавливать сроки и приоритеты, а также отслеживать прогресс выполнения. Простота использования и интуитивно понятный интерфейс делают YouGile доступным для пользователей любого уровня подготовки. Возможность работы с досками задач (Kanban), списками и календарями обеспечивает гибкость в организации рабочих процессов. Интеграция с другими сервисами, такими как Google Drive, Excel и CRM-системами, расширяет функциональность платформы. Поддержка совместной работы в режиме реального времени позволяет командам эффективно взаимодействовать. Доступная стоимость, включая бесплатную версию для небольших команд, делает YouGile привлекательным решением для малого и среднего бизнеса. Однако ограниченные возможности аналитики и отчетности в базовых версиях могут стать препятствием для компаний, требующих детального анализа данных. Меньшая известность по сравнению с международными аналогами также может быть проблемой для организаций, предпочитающих проверенные бренды. Пример работы программы YouGile представлен на рисунке 2.

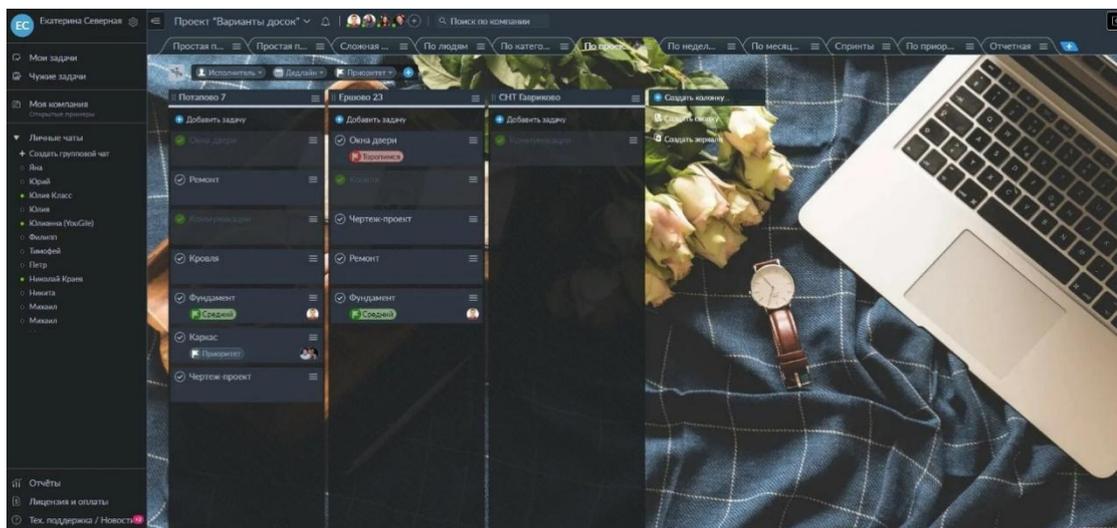


Рисунок 2 – Пример работы YouGile

Trello является одной из самых популярных систем управления задачами в мире и активно используется в России. Он основан на системе досок Kanban, где задачи представляются в виде карточек, которые можно перемещать между колонками, например, «В планах», «В работе» и «Завершено». Простота и доступность для пользователей любого уровня подготовки делают Trello универсальным инструментом для небольших команд. Бесплатная версия с достаточным функционалом для базовых задач привлекает компании, стремящиеся минимизировать затраты. Возможность добавления комментариев, файлов и чек-листов к задачам повышает удобство использования. Интеграция с множеством сторонних сервисов через Power-Ups расширяет возможности платформы. Однако ограниченные возможности для крупных проектов и сложных рабочих процессов могут стать препятствием для масштабирующихся компаний. Отсутствие встроенных инструментов для глубокой аналитики и отчетности также ограничивает применение Trello в организациях, где требуется детальный анализ данных. Платные дополнения могут значительно увеличить стоимость использования. Пример работы программы Trello представлен на рисунке 3.

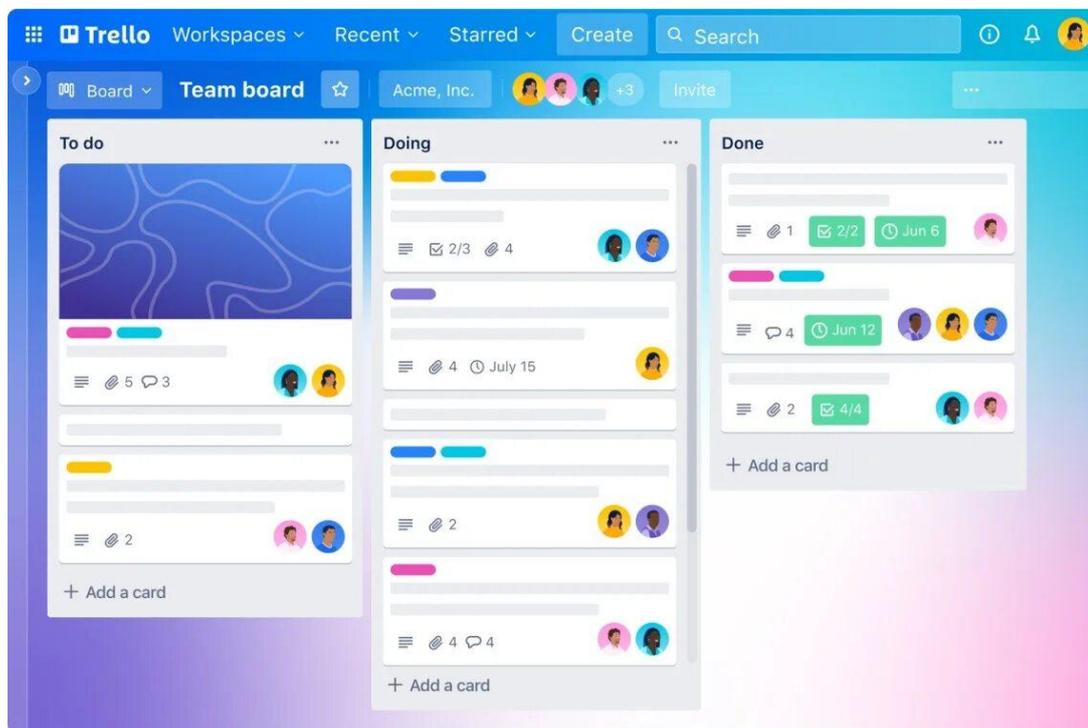


Рисунок 3 – Работа программы Trello

Jira представляет собой мощный инструмент для управления задачами, особенно популярный среди IT-компаний и разработчиков. Он ориентирован на работу с проектами, требующими детального контроля над процессами и задачами. Высокая гибкость настройки под конкретные бизнес-процессы делает Jira незаменимым для технических команд. Расширенные возможности для трекинга ошибок и управления проектами позволяют эффективно контролировать выполнение задач. Интеграция с различными инструментами разработки и DevOps обеспечивает бесшовное взаимодействие между отделами. Подробная аналитика и отчетность помогают принимать обоснованные решения. Однако сложность внедрения и обучения сотрудников может стать препятствием для компаний, не имеющих технической экспертизы. Высокая стоимость лицензий для крупных команд также ограничивает использование Jira в небольших организациях. Избыточный функционал для простых задач может усложнять работу пользователей. Пример работы программы Jira представлен на рисунке 4.

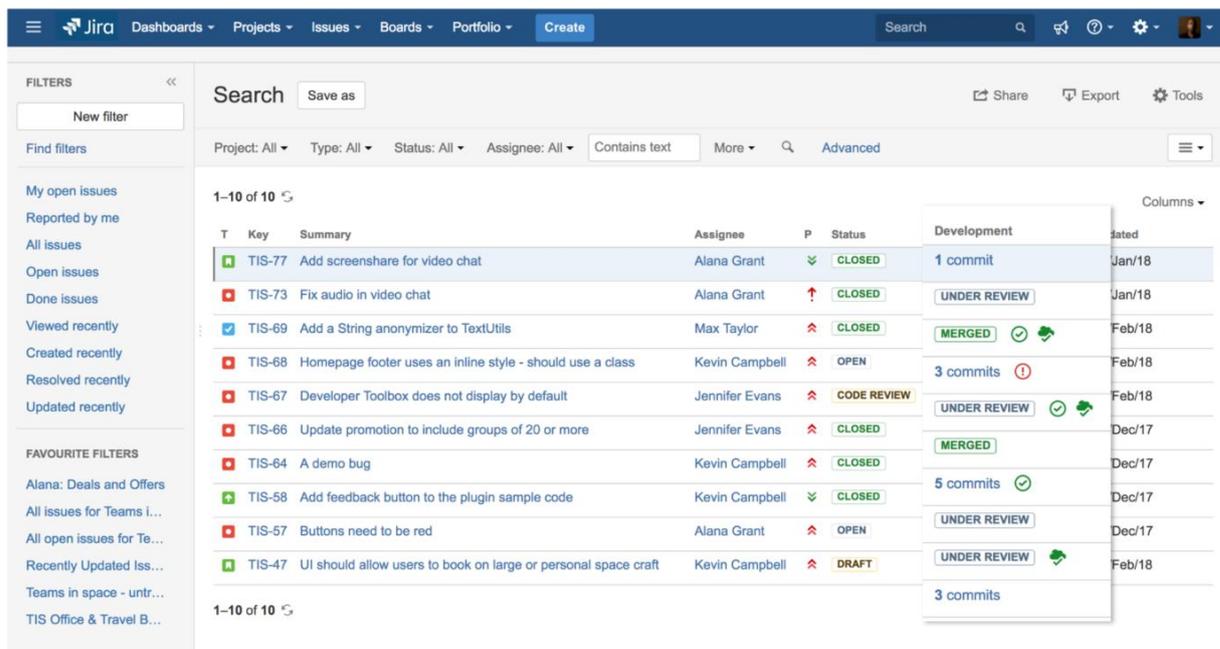


Рисунок 4 – Работа программы Jira

Asana является универсальной системой управления задачами, которая подходит для управления задачами в компаниях различного масштаба. Удобный интерфейс с возможностью переключения между видами представления задач (списки, доски, календарь) делает Asana гибким инструментом для организации работы. Хорошие возможности для совместной работы и общения внутри команды способствуют повышению эффективности взаимодействия. Поддержка создания подзадач и установки зависимостей между задачами позволяет структурировать сложные проекты. Интеграция с популярными сервисами, такими как Slack, Google Drive и Microsoft Teams, расширяет функциональность платформы. Однако ограниченные возможности аналитики в базовых версиях могут стать препятствием для компаний, требующих детального анализа данных. Высокая стоимость для больших команд также ограничивает использование Asana в крупных организациях. Некоторые функции могут быть избыточными для небольших проектов. Пример работы программы Asana представлен на рисунке 5.

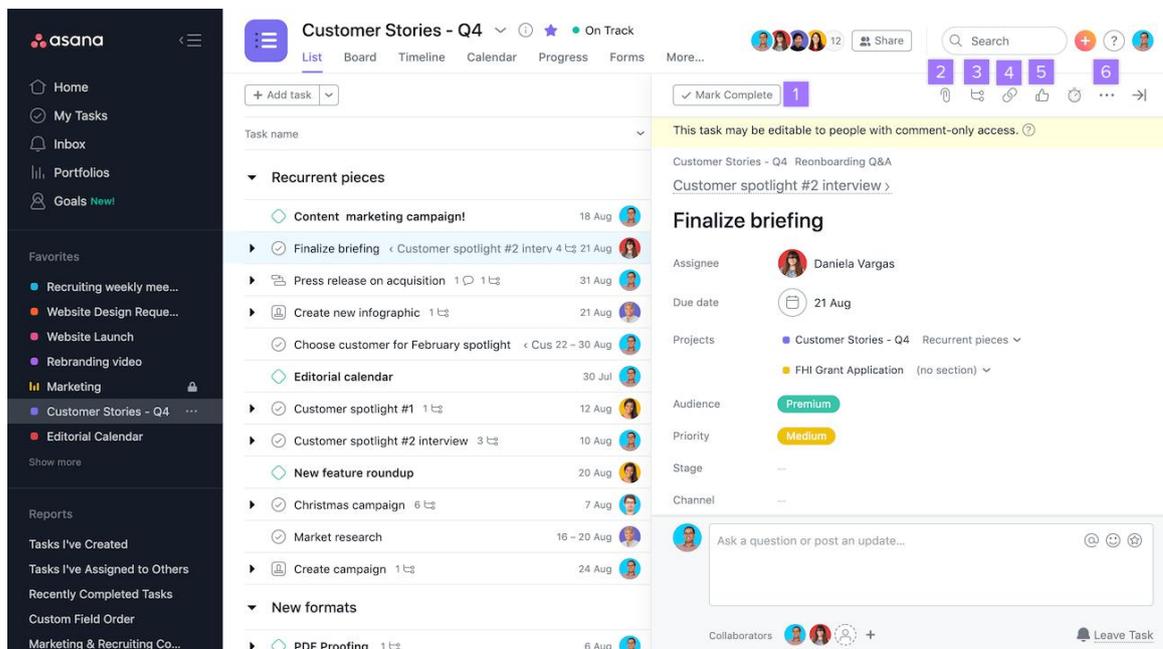


Рисунок 5 – Работа программы Asana

Таким образом, рассмотренные системы управления задачами, несмотря на их многочисленные преимущества, включая простоту использования, доступность и минимальные затраты на внедрение, имеют свои недостатки. Ограниченные возможности настройки, высокая стоимость для крупных команд и потенциальные риски, связанные с безопасностью данных, могут стать серьезными проблемами для компаний с особыми требованиями и высокими стандартами безопасности. Для компаний, стремящихся к максимальной эффективности, выгоде и безопасности, собственные разработки могут стать оптимальным выбором. Собственные системы позволяют полностью адаптировать функционал под уникальные бизнес-процессы, обеспечивают полный контроль над данными и их защитой, а также возможность гибкой интеграции с существующими системами. Для ООО «ВОСТОК-ТЕХНОЛОГИЯ» разработка собственной системы автоматизации задач менеджера может стать лучшим решением для достижения стратегических целей.

1.5 Формулировка задач исследования и общей методики их решения

Формулировка задачи исследования и общей методики ее решения требует четкого понимания целей и этапов разработки информационной системы. В рамках данной работы основной задачей является создание и внедрение системы, которая позволит оптимизировать процессы планирования, распределения и контроля

выполнения задач, повысить эффективность работы менеджеров и обеспечить прозрачность бизнес-процессов. Чтобы достичь этой цели, необходимо последовательно выполнить ряд шагов, включающих аналитическую, проектировочную и практическую части работы.

Первая задача исследования заключается в проведении детального анализа текущих процессов компании, связанных с управлением задачами и проектами. Это включает в себя исследование существующих бизнес-процессов, выявление их слабых сторон и определение ключевых требований к новой системе. Для этого этапа используются такие методы, как интервьюирование сотрудников, анализ документов и проведение обзоров существующих систем управления задачами, применяемых в компании и на рынке. Особое внимание уделяется выявлению узких мест в текущих процессах, таких как неэффективное распределение задач, отсутствие единой системы отслеживания прогресса и недостаток аналитических инструментов.

Вторая задача состоит в выборе и обосновании методик и инструментов, которые будут использоваться для разработки системы. Сюда входит выбор подходящей архитектуры программного обеспечения, определение технических спецификаций и инструментов для реализации. Важно выбрать такие технологии, которые обеспечат гибкость, масштабируемость и безопасность будущей системы. На этом этапе могут быть проведены сравнительные исследования различных языков программирования, баз данных и фреймворков для выявления оптимальных решений. Также рассматриваются возможности интеграции с существующими корпоративными системами.

Третья задача заключается в разработке прототипа системы и его тестировании. Это важный этап, который позволяет определить, насколько эффективно предложенные решения соответствуют поставленным требованиям. В ходе тестирования выявляются ошибки и недочеты, которые затем корректируются для улучшения функциональности системы. Используются методики юнит-тестирования, функционального тестирования и интеграционного тестирования для

обеспечения высокого качества разрабатываемого продукта. Особое внимание уделяется удобству пользовательского интерфейса и скорости работы системы.

Четвертая задача – внедрение готовой системы в работу компании и обучение сотрудников. На этом этапе проводится подготовка пользователей к использованию новой системы через тренинги и семинары. Также осуществляется мониторинг работы системы в реальных условиях, что помогает выявить возможные проблемы и внести корректировки. Особое внимание уделяется обратной связи от сотрудников компании, что позволяет оценить степень удобства использования системы и ее влияние на рабочие процессы.

Общая методика решения задачи строится на использовании комбинированного подхода, включающего элементы анализа, проектирования, программирования и тестирования. На первом этапе осуществляется детальный анализ бизнес-процессов и сбор требований. Это позволяет выявить ключевые элементы и аспекты, которые должна охватывать система, и определить ожидания конечных пользователей. Используются методы системного анализа, которые помогают структурировать и формализовать собранные данные.

Далее следует этап проектирования, где разрабатывается архитектура системы, определяются логические и физические структуры базы данных, создаются интерфейсы и прорабатывается пользовательский опыт (UX). Важной частью этого этапа является создание технического задания, которое станет основой для последующей разработки. Проектирование проводится с учетом гибких методологий, таких как Agile или Scrum, которые позволяют адаптировать процесс разработки к изменяющимся условиям и быстро реагировать на новые требования.

Этап программирования включает непосредственную реализацию системы на основе разработанных спецификаций и требований. При этом используются современные языки программирования и фреймворки, которые обеспечивают надежность, безопасность и масштабируемость. Важно применять передовые практики разработки, такие как использование системы контроля версий (например,

Git) и автоматизация процесса сборки и развертывания приложения. Это позволяет минимизировать риски ошибок и ускорить процесс внедрения.

После написания кода следует этап тестирования, на котором система проверяется на предмет соответствия требованиям, выявления ошибок и их исправления. Проводятся различные виды тестирования: юнит-тестирование, функциональное, нагрузочное и интеграционное. Это позволяет гарантировать, что все компоненты системы работают корректно и взаимодействуют между собой без сбоев. Особое внимание уделяется тестированию безопасности системы, чтобы исключить возможность несанкционированного доступа к данным.

Завершающий этап – внедрение системы и обучение пользователей. Проводятся пилотные запуски, которые помогают оценить, как система работает в реальных условиях и насколько она удобна для пользователей. Полученные данные используются для внесения окончательных корректировок перед полноценным внедрением. На этом этапе также организуются обучающие сессии для сотрудников, чтобы они могли эффективно использовать новую систему в своей работе. После внедрения осуществляется сопровождение и поддержка системы, что включает регулярное обновление и улучшение ее функционала на основе обратной связи от пользователей и новых бизнес-требований.

Таким образом, общая методика решения задачи опирается на итеративный и адаптивный процесс, который сочетает тщательное планирование и анализ с гибким подходом к разработке и внедрению. Это помогает минимизировать риски и обеспечивать высокое качество конечного продукта, соответствующего требованиям бизнеса и потребностям пользователей.

2 ПРОЕКТИРОВАНИЕ АВТОМАТИЗИРОВАННОГО РАБОЧЕГО МЕСТА МЕНЕДЖЕРА

2.1 Архитектура информационной системы

При разработке автоматизированного рабочего места менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ» особое внимание было уделено архитектуре системы, поскольку от её структуры напрямую зависят такие характеристики, как надёжность, расширяемость, производительность и удобство сопровождения программного продукта. В качестве основной архитектурной модели была выбрана клиент-серверная архитектура, позволяющая эффективно распределять логику между клиентской и серверной частями приложения.

Система построена на трёхуровневой схеме, которая включает в себя представление (интерфейс пользователя), бизнес-логику (обработка данных) и уровень хранения данных (база данных). Такое разделение позволяет модифицировать каждый из уровней независимо от других, что особенно важно при дальнейшем развитии системы.

На уровне представления реализован пользовательский интерфейс, выполненный с использованием технологий HTML5, CSS3 и JS. Благодаря применению фреймворка Bootstrap обеспечивается адаптивность и единообразие элементов управления. Взаимодействие с пользователем организовано через браузер, что делает систему доступной с любого устройства с поддержкой веб-технологий. Система сеток Bootstrap использует серию контейнеров, строк и колонок для компоновки и выравнивания содержимого. Он построен с помощью flexbox и полностью адаптивен [1].

Бизнес-логика реализована на стороне сервера с помощью языка PHP. Для обработки запросов, маршрутизации и формирования ответов используется CMS MODX, которая предоставляет гибкие возможности для управления контентом и шаблонами. Сниппеты MODX позволяют интегрировать динамические данные прямо в веб-страницы, обеспечивая связь между интерфейсом и сервером.

Для взаимодействия с данными применяется ORM RedBeanPHP, которая

абстрагирует работу с базой данных и позволяет оперировать объектами вместо SQL-запросов. Это повышает читаемость кода, снижает риск ошибок и обеспечивает защиту от SQL-инъекций. RedBeanPHP – это инструмент, устраняющий несоответствие между объектными моделями и реляционными базами данных. Он создаёт таблицы и столбцы динамически, по мере работы с объектами. Нет необходимости вручную писать SQL-запросы или поддерживать XML-файлы структуры. Достаточно создавать, сохранять, извлекать и манипулировать доменными объектами с помощью PHP, а всю работу с базой данных (далее – БД) выполнит RedBeanPHP [2].

Уровень хранения данных представлен реляционной базой MySQL. Все таблицы спроектированы с учётом принципов нормализации, что исключает дублирование информации и обеспечивает целостность данных. При этом сохранён баланс между нормализацией и производительностью, чтобы система могла эффективно обрабатывать увеличивающийся объём информации без потери скорости работы.

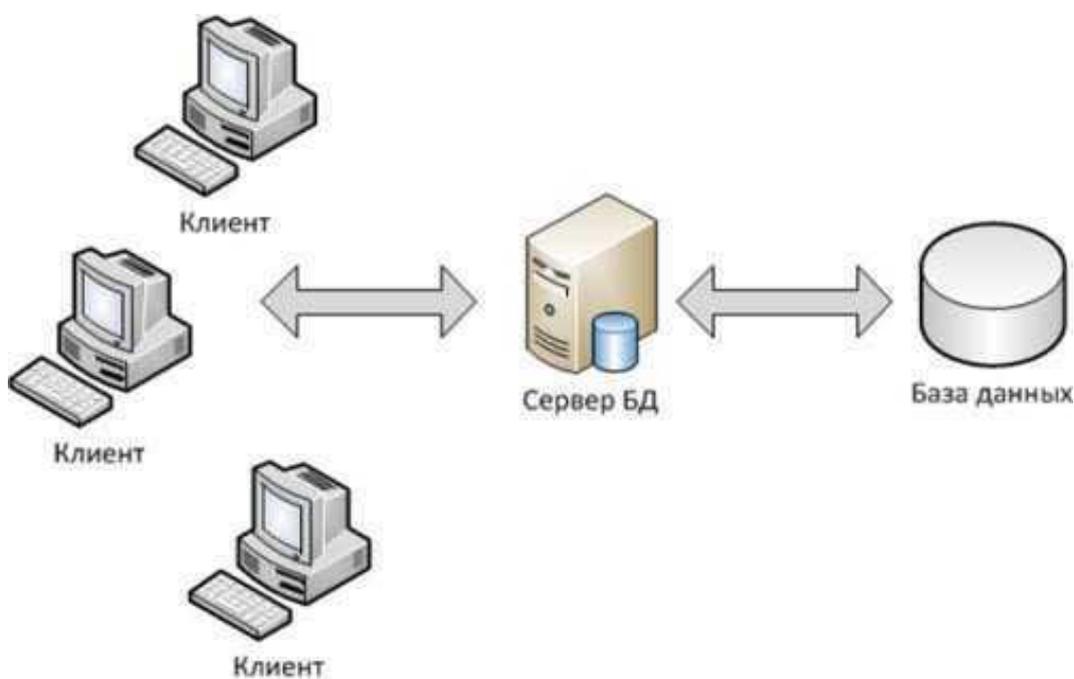


Рисунок 6 – Архитектура информационной системы

Такая архитектура обеспечивает высокую степень масштабируемости. Например, в случае увеличения нагрузки возможно выделение отдельных

компонентов (например, уровня баз данных или серверной логики) на отдельные серверы или использование облачных решений. Также предусмотрена возможность добавления новых функциональных модулей без значительной перестройки существующих компонентов.

Ещё одним важным аспектом является безопасность. Обмен данными между клиентом и сервером осуществляется по протоколу HTTPS, что защищает информацию от прослушивания и несанкционированного изменения. Проверка прав доступа реализована на уровне пользовательских ролей, что позволяет ограничивать действия в зависимости от принадлежности к той или иной группе.

Таким образом, выбранная архитектура позволила создать устойчивую, легко расширяемую и защищённую систему, полностью соответствующую требованиям, предъявляемым к автоматизированному рабочему месту менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ».

2.2 Выбор технологий реализации

При разработке автоматизированного рабочего места менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ» особое внимание было уделено выбору программных технологий и инструментов, поскольку от этого напрямую зависят такие характеристики системы, как производительность, простота сопровождения, масштабируемость и удобство использования.

Для реализации клиентской части была выбрана комбинация HTML5, CSS3 и JS, что обусловлено их широким распространением, кроссплатформенностью и поддержкой современных браузеров. Эти технологии позволяют создавать адаптивный и семантически корректный интерфейс, который одинаково хорошо работает на различных устройствах – от десктопов до мобильных телефонов.

Для повышения интерактивности и упрощения работы с DOM-деревом использовалась библиотека jQuery. Её применение позволило значительно ускорить разработку за счёт готовых функций анимации, обработки событий и AJAX-запросов. jQuery – это быстрая, компактная и богатая по возможностям JS-библиотека. Она упрощает такие задачи, как навигация по документу HTML,

манипуляции с элементами, обработка событий, анимация и использование AJAX, при этом предоставляя простой в использовании интерфейс, работающий во всех современных браузерах. Модель Ajax предполагает использование веб-API в качестве прокси для более разумного запроса данных, а не просто для того, чтобы браузер перезагружал всю страницу [3].

Серверная часть реализована на языке PHP, который остаётся одним из самых популярных языков для создания веб-приложений. PHP (рекурсивный акроним словосочетания PHP: Hypertext Preprocessor) – распространённый язык программирования общего назначения с открытым исходным кодом, который сконструировали специально для веб-разработки и который встраивают в HTML-разметку. [4]. Его использование оправдано следующими преимуществами:

- простота интеграции с базой данных;
- поддержка большинства хостинг-провайдеров;
- быстрая разработка прототипов;
- широкая экосистема и сообщество разработчиков.

Для управления контентом и шаблонами использовалась CMS MODX, которая предоставляет гибкую систему управления ресурсами и позволяет легко внедрять динамические данные в веб-страницы. Для взаимодействия с базой данных был применён ORM-фреймворк RedBeanPHP, который обеспечивает простоту работы с данными через объектно-ориентированный подход. Преимущества его использования включают:

- отсутствие необходимости писать SQL-запросы вручную;
- защиту от SQL-инъекций;
- удобство миграции структуры базы данных;
- минимальные требования к конфигурации.

Таблица 1 – Сравнение выбранных технологий по ключевым критериям

№	Технология	Производительность	Поддержка сообществом	Безопасность	Масштабируемость
1	HTML5 / CSS3	Высокая	Отличная	Умеренная	Высокая
2	JS	Средняя	Отличная	Средняя	Средняя
3	jQuery	Средняя	Хорошая	Средняя	Средняя
4	PHP	Средняя	Отличная	Средняя	Хорошая
5	RedBeanPHP (ORM)	Средняя	Хорошая	Высокая	Средняя
6	MODX	Средняя	Хорошая	Средняя	Хорошая
7	MySQL	Хорошая	Отличная	Хорошая	Хорошая

В качестве СУБД была выбрана MySQL, которая является одной из наиболее распространённых реляционных баз данных в мире. Её выбор обусловлен такими факторами, как:

- высокая производительность при работе с относительно небольшими объёмами данных;
- поддержка транзакций и механизмов безопасности;
- интеграция с PHP и другими компонентами стека;
- большое количество инструментов администрирования (например, phpMyAdmin).

Таким образом, выбранный стек технологий представляет собой устойчивый, экономически выгодный и достаточно простой для реализации набор инструментов, соответствующий современным требованиям к разработке веб-приложений. Использование open-source решений позволило минимизировать лицензионные затраты и обеспечить долгосрочное сопровождение продукта.

2.3 Проектирование пользовательского интерфейса

При разработке автоматизированного рабочего места менеджера проектов особое внимание было уделено проектированию пользовательского интерфейса. Целью стало создание системы, которая будет удобной для работы, интуитивно понятной и соответствующей современным стандартам юзабилити. Интерфейс был спроектирован таким образом, чтобы максимально облегчить выполнение повседневных задач, повысить эффективность управления проектами и обеспечить простоту освоения системы новыми пользователями.

Все элементы интерфейса были реализованы с использованием технологий HTML5, CSS3 и JS, что позволило добиться высокой адаптивности и отзывчивости при работе как на десктопах, так и на мобильных устройствах. В качестве основного фреймворка для верстки применялся Bootstrap, что обеспечило единый стиль и согласованность всех компонентов. Все страницы имеют чёткую структуру: шапка, основная часть и боковой блок.

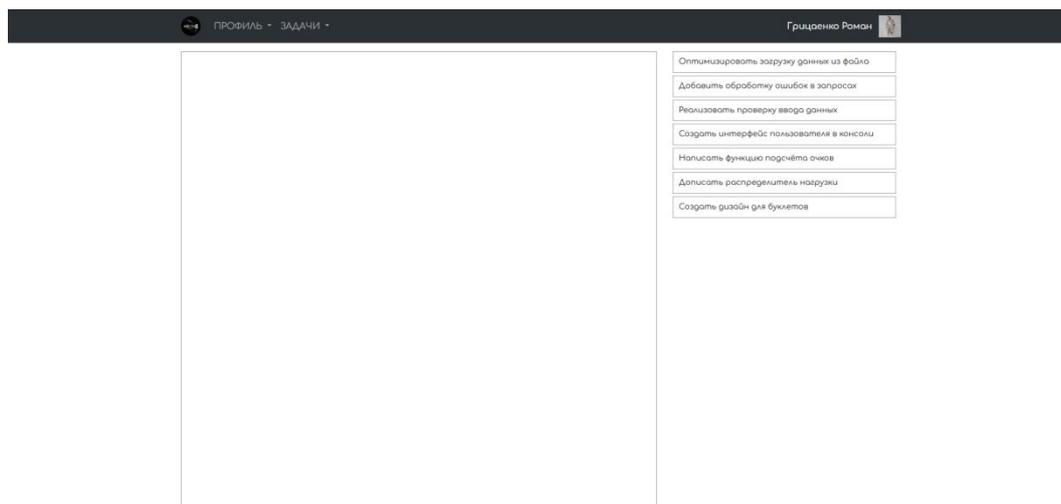


Рисунок 7 – Главная страница системы

На главной странице расположен список задач. Пользователь может быстро выбрать нужную задачу и перейти к её детальному просмотру. В шапке расположены выпадающие меню «Профиль» и «Задачи», которые позволяют управлять личными данными, создавать новые задачи и просматривать список существующих.

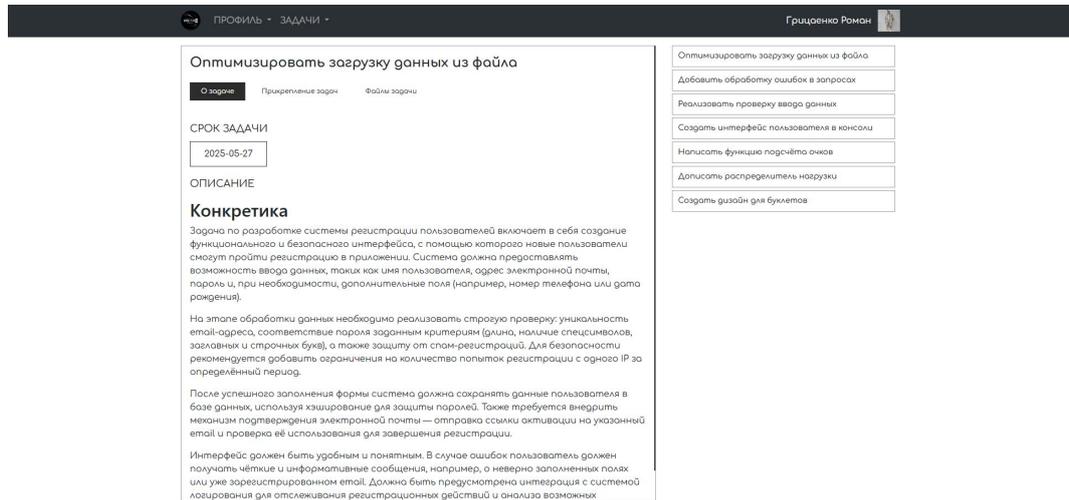


Рисунок 8 – Страница просмотра задачи

Страница просмотра задачи содержит информацию о названии задачи, сроке выполнения, исполнителе и контролирующем лице, а также описание задачи. Также здесь доступны три вкладки: «О задаче», «Прикрепление задач» и «Файлы задач». Это позволяет не покидать страницу при переходе между разделами и сохраняет контекст текущей задачи.

Каждый файл, прикрепленный к задаче, кликабелен и открывает документ в новом окне. Это упрощает работу с приложениями, избавляя от необходимости скачивания файла до его просмотра. Доступ к редактированию задачи осуществляется через кнопку «Изменить задачу», которая открывает модальное окно с формой, содержащей все поля, доступные при создании задачи.

Оптимизировать загрузку данных из файла Вернуться

О задаче

Даты

Описание

<h2>Конкретика</h2> <p>Задача по разработке системы регистрации пользователей включает в себя создание функционального и безопасного интерфейса, с помощью которого новые пользователи смогут пройти регистрацию в приложении. Система должна предоставлять возможность ввода данных, таких как имя пользователя, адрес электронной почты, пароль и, при необходимости, дополнительные поля (например, номер телефона или дата рождения).</p> <p>На этапе обработки данных необходимо реализовать строгую проверку: уникальность email-адреса, соответствие пароля заданным критериям (длина, наличие спецсимволов, заглавных и строчных букв), а также защиту от спам-регистраций. Для безопасности рекомендуется добавить ограничения на количество попыток регистрации с одного IP за определённый период.</p> <p>После успешного заполнения формы система должна сохранять данные пользователя в базе данных, используя хэширование для защиты паролей. Также требуется внедрить механизм подтверждения электронной почты —

Сотрудники

Задача гана:

Контролирующие исполнение:

Документация

Рисунок 9 – Форма редактирования задачи

Форма редактирования задачи была спроектирована с учётом удобства заполнения и проверки корректности данных. Поля ввода имеют валидацию, что исключает ошибки при внесении информации. Здесь можно изменить название задачи, её описание, срок выполнения, назначить новых исполнителей и загрузить дополнительные файлы. Такой подход к оформлению форм позволил снизить количество ошибок при работе с системой и повысить общую производительность.

Авторизация и регистрация реализованы на отдельных страницах, где используются простые и понятные формы. При регистрации требуется указать email, логин, пароль и подтверждение пароля. После успешной регистрации

пользователь получает письмо с подтверждением электронной почты. Авторизация происходит через стандартную форму входа, где нужно ввести электронную почту и пароль.

АВТОРИЗАЦИЯ

Почта

Пароль

Войти

РЕГИСТРАЦИЯ

Почта

Логин

Пароль

Подтверждение пароля

Зарегистрироваться

Рисунок 10 – Страницы авторизации и регистрации

Раздел профиля пользователя предоставляет возможность просмотра и изменения личных данных. На странице отображается аватар, ФИО, email, телефон и дата рождения.

Моя страница

 Грицаенко Роман [Редактировать](#)

Пользователь:
Полное ФИО: Грицаенко Роман Андреевич
Email: rmngr1caenk0@mail.ru
Телефон: +79143839016
День рождения: 25.12.2000

Рисунок 11 – Страница профиля

Кнопка «Редактировать» открывает боковое меню, где можно изменить любую из указанных выше характеристик. Также можно добавить или изменить список проектов, в которых участвует пользователь.

Настройки аккаунта

[Перейти в аккаунт](#)

Фото



Выберите файл | Файл не выбран

Фамилия

Грицаенко Изменено 01.01.2025

Имя

Роман Изменено 01.01.2025

Отчество

Андреевич Изменено 01.01.2025

Email

rmngr1caenk0@mail.ru Изменено 01.01.2025

Телефон

+79143839016 Изменено 01.01.2025

День рождения

25.12.2000 Изменено 01.01.2025

Должность

Разработчик Изменено 01.01.2025

Рисунок 12 – Страница профиля и редактирования данных

Интерфейс системы был разработан с учётом принципов минимального дизайна и единообразия. Все элементы управления имеют одинаковый стиль и поведение, что упрощает обучение и снижает риск ошибок. Поддерживается два режима отображения – светлая и тёмная темы, что положительно влияет на зрительное восприятие информации в разных условиях освещения.

2.3.1 Разграничение прав пользователей в интерфейсе

Для обеспечения гибкости управления задачами и проектами, а также безопасности информации, в разработанной системе реализовано разграничение прав пользователей на основе ролевой модели. Система поддерживает следующие роли: директор, менеджер, исполнитель, наблюдатель. Каждая роль имеет определённый уровень доступа к функционалу системы, что отражается как в логике работы, так и во внешнем виде пользовательского интерфейса.

Роль пользователя влияет не только на набор доступных действий, но и на то, какие данные он может видеть и как с ними взаимодействовать. Например,

директор имеет полный доступ ко всем задачам и проектам, аналогично уровню доступа менеджера. Он может просматривать, редактировать, удалять любые задачи, назначать исполнителей и наблюдателей, изменять срок выполнения и прикреплять файлы.

Менеджер может управлять только теми задачами, за которыми закреплён. Его возможности включают изменение описания, сроков, состава исполнителей и наблюдателей, удаление задачи, прикрепление файлов.

Исполнитель может просматривать задачу и отметить её как выполненную. Все остальные поля заблокированы, чтобы исключить несанкционированные изменения.

Наблюдатель имеет только право просмотра задачи. Это может быть, например, заказчик или сотрудник, которому требуется только отслеживать прогресс без возможности вмешательства.

Особое внимание было уделено удобству использования системы. Задачу может создать любой пользователь, и по умолчанию он становится менеджером этой задачи. Такое решение обусловлено особенностями малого бизнеса, где сотрудники часто совмещают несколько ролей.

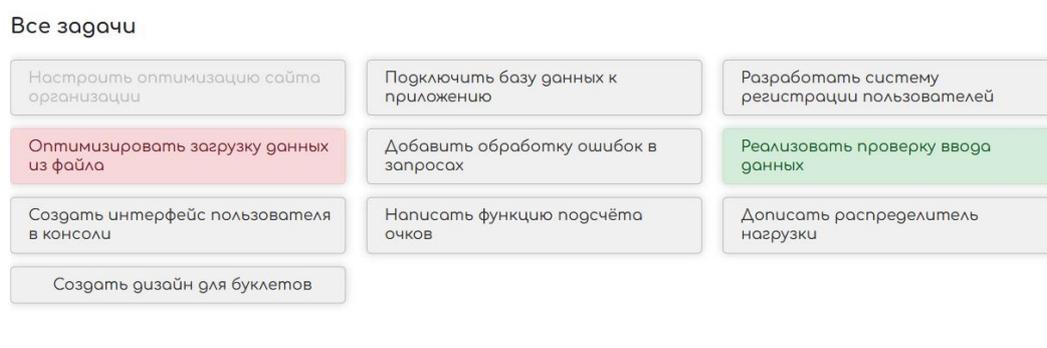


Рисунок 13 – Пример списка задач

Список всех задач отображается с учётом принадлежности пользователя к той или иной задаче. Для упрощения восприятия информация окрашена в разные цвета:

- просроченные задачи выделены красным цветом;

- выполненные задачи – зелёным;
- задачи, к которым пользователь не причастен – серые и недоступны для клика.

Такой подход позволяет пользователю сразу понять, какие задачи ему доступны, а какие нет, без лишних объяснений.

Кроме того, пользователи, не являющиеся менеджерами или директорами, видят только те задачи, которые им назначены. Остальные остаются скрытыми, что усиливает защиту данных и снижает вероятность случайного доступа.

The screenshot shows a task management interface. At the top, the task title is "Оптимизировать загрузку данных из файла" (Optimize data loading from file). To the right of the title is a button labeled "Изменить задачу" (Edit task). Below the title are three tabs: "О задаче" (About task), "Прикрепление задач" (Attach tasks), and "Файлы задачи" (Task files). The "О задаче" tab is active. Below the tabs, the task's due date is displayed as "СРОК ЗАДАЧИ" (Task deadline) with a value of "2025-05-27". The main section is titled "ОПИСАНИЕ" (Description) and contains the following text:

Конкретика

Задача по разработке системы регистрации пользователей включает в себя создание функционального и безопасного интерфейса, с помощью которого новые пользователи смогут пройти регистрацию в приложении. Система должна предоставлять возможность ввода данных, таких как имя пользователя, адрес электронной почты, пароль и, при необходимости, дополнительные поля (например, номер телефона или дата рождения).

На этапе обработки данных необходимо реализовать строгую проверку: уникальность email-адреса, соответствие пароля заданным критериям (длина, наличие спецсимволов, заглавных и строчных букв), а также защиту от спам-регистраций. Для безопасности рекомендуется добавить ограничения на количество попыток регистрации с одного IP за определённый период.

После успешного заполнения формы система должна сохранять данные пользователя в базе данных, используя хэширование для защиты паролей. Также требуется внедрить механизм подтверждения электронной почты — отправка ссылки активации на указанный email и проверка её использования для завершения регистрации.

Интерфейс должен быть удобным и понятным. В случае ошибок пользователь должен получать чёткие и информативные сообщения, например, о неверно заполненных полях или уже зарегистрированном email. Должна быть предусмотрена интеграция с системой логирования для отслеживания регистрационных действий и анализа возможных аномалий.

Рисунок 14 – Детальный просмотр задачи у менеджера

Доступ к странице детального просмотра задачи предоставляется только тем

пользователям, которые имеют к ней отношение. На этой странице отображаются:

- название задачи;
- описание;
- срок выполнения;
- статус;
- исполнители и наблюдатели;
- прикрепленные файлы.

В зависимости от роли пользователя, на этой странице становятся доступны или недоступны определённые элементы интерфейса. Например, только директор и менеджер могут редактировать задачу, удалять её или прикреплять файлы. Исполнитель видит кнопку «Отметить как выполненную», а наблюдатель не видит никаких действий вообще.

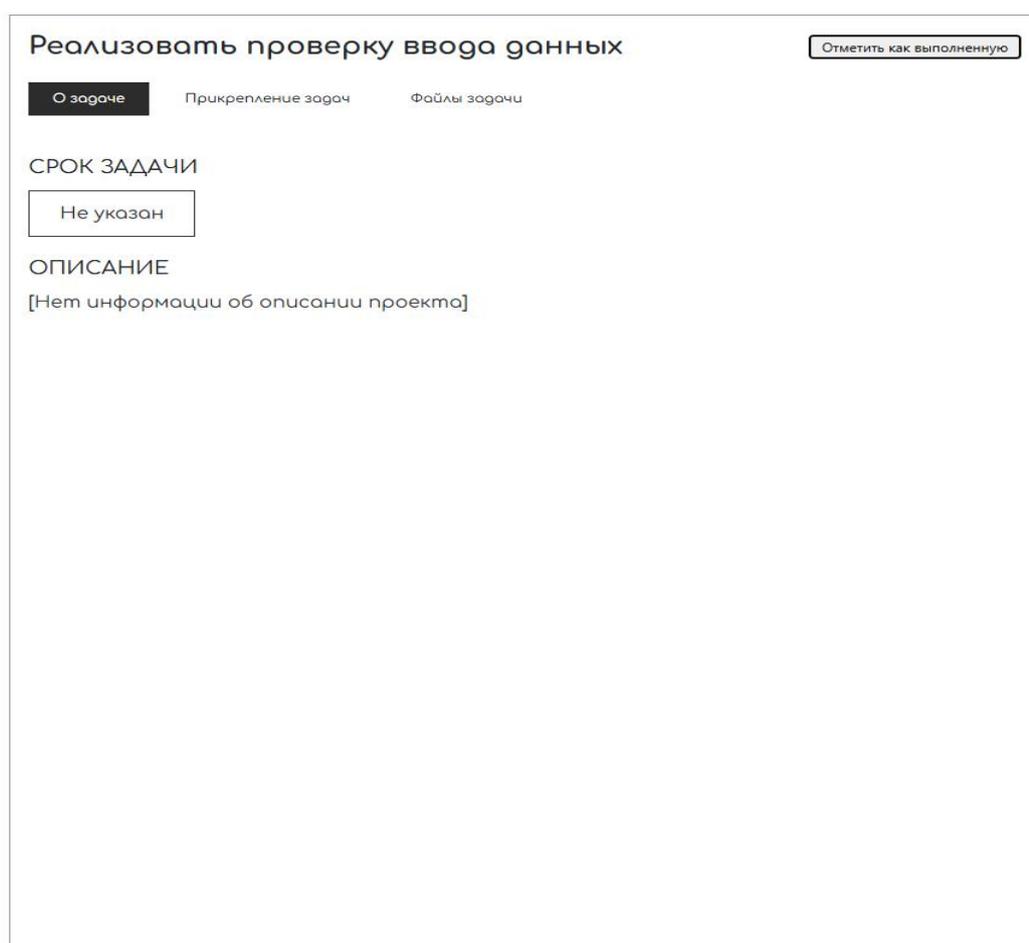


Рисунок 15 – Детальный просмотр задачи у исполнителя

В списке всех проектов также реализовано разграничение прав. Директор и

менеджер могут кликнуть на любой проект и перейти к его детальному просмотру. Исполнитель и наблюдатель видят только те проекты, в которых они участвуют, и могут переходить только по ним.

На странице детального просмотра задачи интерфейс также меняется в зависимости от роли. Менеджер может использовать все доступные инструменты, включая редактирование, удаление и загрузку файлов. Исполнитель получает ограниченный набор возможностей, а наблюдатель видит только информацию о задаче без возможности внесения изменений.

Это разделение позволило сделать систему более предсказуемой и безопасной. Пользователь всегда видит только ту информацию, которая относится к нему, и может выполнять действия строго в рамках своей роли.

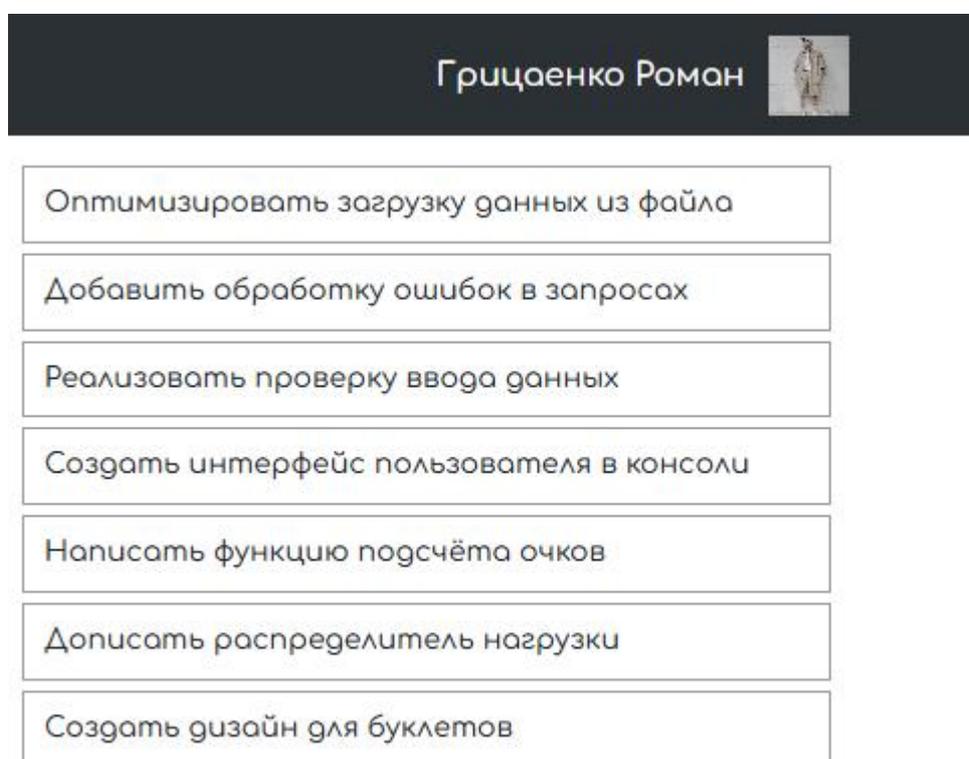


Рисунок 16 – Список проектов у пользователя

2.4 Реализация функциональных модулей

Реализация функциональных модулей автоматизированного рабочего места менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ» осуществлялась с учётом

требований к интерфейсу, безопасности и взаимодействию между пользователями. Основные компоненты системы были разработаны с использованием клиент-серверной архитектуры, что позволило разделить логику представления и обработки данных. В качестве технологического стека применялись HTML5, CSS3, JS и jQuery для реализации клиентской части; PHP и CMS MODX использовались на серверной стороне. Для работы с базой данных была внедрена ORM RedBeanPHP, обеспечившая безопасное и удобное взаимодействие с реляционной СУБД MySQL.

Модуль задач стал центральным элементом системы, поскольку именно через него организуется основная работа пользователей. Была реализована возможность создания новой задачи с указанием её названия, описания, срока выполнения, приоритета и назначения исполнителя. Также можно выбрать контролирующего лицо, отвечающее за результат выполнения задачи. После отправки формы данные сохраняются в таблице tasks, связанной с таблицей users. Это позволяет устанавливать чёткую связь между задачами и ответственными сотрудниками.

Просмотр задач реализован в виде списка или карточек, где каждая запись содержит название, статус, срок выполнения и имя исполнителя. Пользователь может кликнуть на любую задачу, чтобы перейти к её детальному просмотру. Редактирование задачи выполняется через модальное окно, где доступны изменения всех полей, включая описание, срок выполнения, список исполнителей и контролирующих лиц, а также загрузка новых файлов. Удаление задачи доступно только администратору или автору, что снижает риск случайного удаления важной информации.

Создать задачу

Логотип

Выберите файл | Файл не выбран

Название

Впишите сюда название задачи

Создать

Рисунок 17 – Страница создания задачи

Форма редактирования задачи построена таким образом, чтобы максимально упростить внесение изменений. Все поля предварительно заполняются из базы данных, что исключает необходимость повторного ввода информации. Такой подход повышает эффективность управления задачами и снижает вероятность ошибок при редактировании.

Модуль пользователей был реализован с учётом необходимости регистрации, входа, просмотра профиля и редактирования данных. Новый пользователь может зарегистрироваться, указав email, логин и пароль. После регистрации данные сохраняются в таблице users, а также отправляется письмо с подтверждением электронной почты. Авторизация осуществляется через страницу входа, где требуется указать email и пароль. После успешного входа система перенаправляет пользователя на главную страницу.

На странице профиля отображаются такие данные, как фамилия, имя, отчество, email, телефон, дата рождения и аватар. Эти данные можно редактировать через специальную форму. Предусмотрена возможность загрузки нового аватара и изменения других параметров аккаунта. Также администратор имеет доступ к списку всех зарегистрированных пользователей, что позволяет управлять учетными записями и назначать роли.

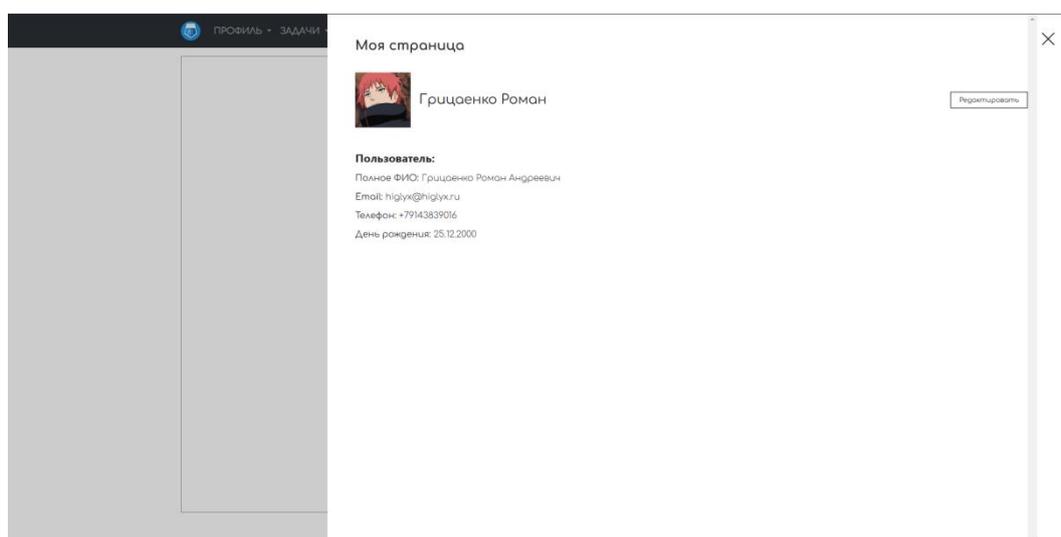


Рисунок 18 – Страница профиля пользователя

Модуль документов был создан для хранения и обмена информацией,

связанной с задачами. Пользователь может прикреплять файлы к задачам, просматривать их список и удалять ненужные документы. Файлы хранятся в таблице files, связанной с задачами через внешний ключ. Каждый файл имеет уникальный идентификатор, имя, дату загрузки и ссылку на соответствующую задачу.

Закреть

Загрузите файл

Выберите файл Файл не выбран

Сохранить

Рисунок 19 – Прикрепление файлов к задаче

CMS MODX использовалась для управления контентом и шаблонами системы. Через сниппеты MODX была реализована динамическая подгрузка данных без перезагрузки страниц. Например, сниппет `getTaskDetails` отвечает за вывод информации о задаче, а сниппет `getAllUsers` формирует список пользователей при назначении задач. Также были созданы шаблоны для основных страниц, что позволило сохранить единый стиль и логику отображения данных.

Для упрощения работы с базой данных была внедрена ORM RedBeanPHP. Её использование позволило отказаться от ручного написания SQL-запросов, обеспечить защиту от SQL-инъекций и ускорить разработку благодаря простому API. Все операции с данными выполняются через объекты, что делает код более читаемым и безопасным.

2.5 Проектирование базы данных

Проектирование базы данных является одним из ключевых этапов разработки автоматизированного рабочего места менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ». От её структуры зависит эффективность хранения информации, скорость выполнения запросов и простота сопровождения системы. При проектировании учитывались принципы нормализации, направленные на исключение дублирования данных и обеспечение целостности информации. База данных была реализована с использованием СУБД MySQL, что обеспечило надёжное хранение информации и поддержку реляционной модели.

Для взаимодействия с базой данных применялась ORM RedBeanPHP, которая позволила работать с данными в объектно-ориентированном стиле и обеспечила защиту от SQL-инъекций. Это повысило читаемость кода, облегчило его тестирование и последующее сопровождение. Все операции выполнялись через вызовы методов, а не напрямую через SQL-запросы.

Таблица users предназначена для хранения данных о зарегистрированных пользователях системы. Она содержит информацию, необходимую как для аутентификации, так и для назначения задач. В ней хранятся данные о менеджерах и других сотрудниках. Также там может храниться информация о заказчиках, если у них имеется доступ к системе.

Таблица 2 – Сущность «Пользователи»

Поле	Тип данных	Описание
id	INT (PK)	Уникальный идентификатор пользователя
email	VARCHAR(55)	Адрес электронной почты
password	TEXT	Зашифрованный пароль
name	VARCHAR(30)	Имя
surname	VARCHAR(30)	Фамилия

patronymic	VARCHAR(30)	Отчество
phone	VARCHAR(20)	Контактный телефон
avatar	TEXT	Ссылка на аватар
date_register	TIMESTAMP	Дата регистрации пользователя
position	VARCHAR(100)	Должность пользователя
Birth_date	DATETIME	Дата рождения пользователя

Эта таблица служит основой системы, поскольку именно через неё организуются связи с задачами и документами. Все пользователи имеют уникальный идентификатор, который используется как внешний ключ в других таблицах, таких как tasks, files и tasks_users.

После определения структуры пользователей была спроектирована таблица tasks, предназначенная для хранения информации о задачах. Каждая запись содержит данные, характеризующие содержание задачи, сроки её выполнения и текущий статус.

Таблица 3 – Сущность «Задачи»

Поле	Тип данных	Описание
id	INT (PK)	Уникальный идентификатор задачи
title	VARCHAR(100)	Название задачи
description	TEXT	Описание задачи
deadline	DATE	Срок выполнения задачи
status	VARCHAR(10)	Текущий статус задачи

id_author	INT (FK)	Идентификатор автора задачи
date_creation	TIMESTAMP	Дата создания задачи

Задачи являются центральным элементом системы, поэтому к этой таблице предъявлялись высокие требования по скорости чтения и записи. Для повышения производительности были добавлены индексы для полей id_author и status. Такой подход позволил ускорить фильтрацию и выборку данных при работе с интерфейсом.

Следующей важной таблицей стала таблица files, используемая для хранения файлов, прикрепленных к задачам. Эта таблица позволяет загружать, просматривать и удалять документы, относящиеся к конкретным задачам.

Таблица 4 – Сущность «Файлы»

Поле	Тип данных	Описание
id	INT (PK)	Уникальный идентификатор файла
file_name	VARCHAR(255)	Имя файла
url	TEXT	Путь или ссылка на файл
date_upload	TIMESTAMP	Дата загрузки файла
id_task	INT (FK)	Идентификатор связанной задачи
id_user	INT (FK)	Идентификатор пользователя, загрузившего файл

Для работы с изображениями, включая аватары пользователей и другие графические материалы, была создана таблица photos. Эта таблица аналогична таблице files, но используется для хранения ссылок на изображения.

Таблица 5 – Сущность «Фотографии»

Поле	Тип данных	Описание
id	INT (PK)	Уникальный идентификатор фотографии
id_task	INT (FK)	Идентификатор связанного проекта
id_user	INT (FK)	Идентификатор пользователя, загрузившего фото
url	TEXT	Ссылка на изображение

Для реализации гибкой системы управления участниками задач была внедрена промежуточная таблица tasks_users, обеспечивающая связь «многие-ко-многим» между пользователями и проектами.

Таблица 6 – Сущность «Пользователи в задаче»

Поле	Тип данных	Описание
id	INT (PK)	Уникальный идентификатор связи
id_task	INT (FK)	Идентификатор проекта
id_user	INT (FK)	Идентификатор пользователя
type	VARCHAR(50)	Роль пользователя в проекте

Структура базы данных была спроектирована таким образом, чтобы максимально соответствовать предметной области. Все таблицы связаны между собой через внешние ключи, что обеспечивает целостность данных и упрощает формирование выборок при помощи JOIN-запросов. Связь один-ко-многим реализована между задачами и пользователями, а отношение многие-ко-многим – через таблицу tasks_users.

В результате была спроектирована логическая модель базы данных,

содержащая 4 сущности, их атрибуты и связи между ними. Схема логической модели представлена на рисунке 20.

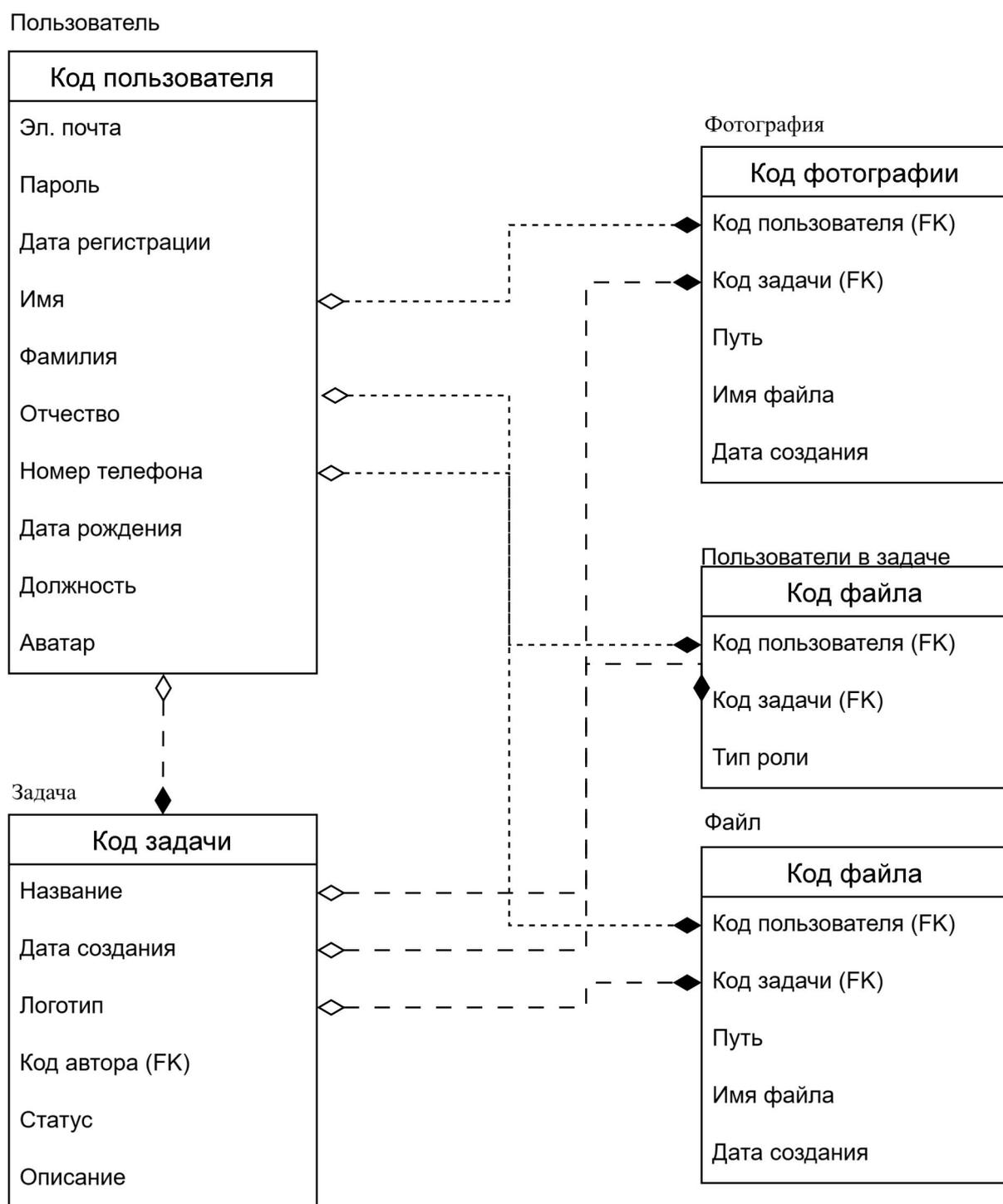


Рисунок 20 – Логическая модель базы данных

Затем на основании данных из таблиц 2-6 и логической модели данных была спроектирована физическая модель, представленная на рисунке 21.

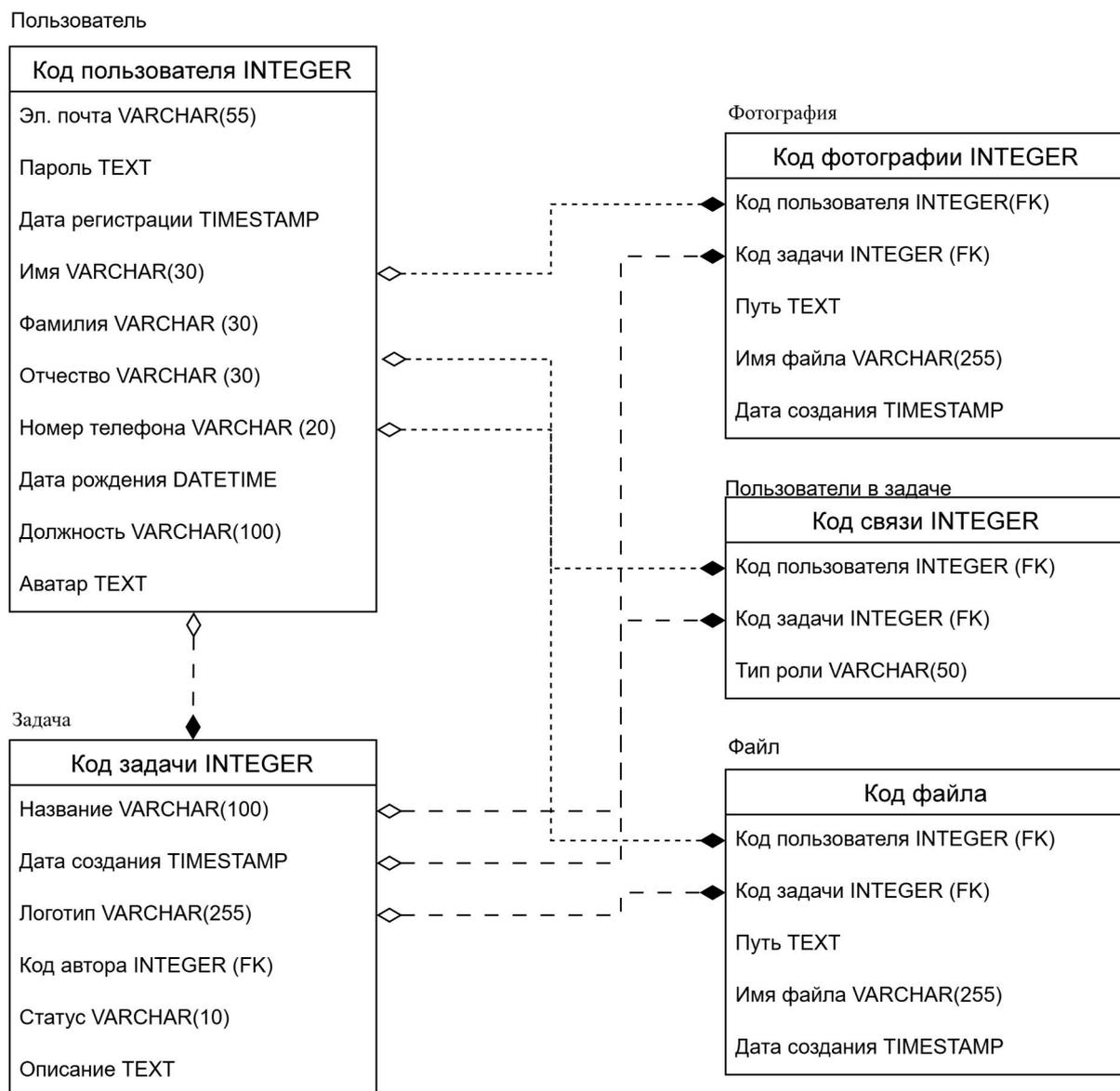


Рисунок 21 – Физическая модель базы данных

Применение ORM RedBeanPHP позволило значительно упростить работу с базой данных. Все операции выполняются через вызовы методов, что делает код более понятным и упрощает его тестирование и последующее сопровождение.

Итоговые отношения были отражены в ER-диаграмме базы данных, которая наглядно демонстрирует связи между сущностями, типы ключей и уровень взаимодействия таблиц.

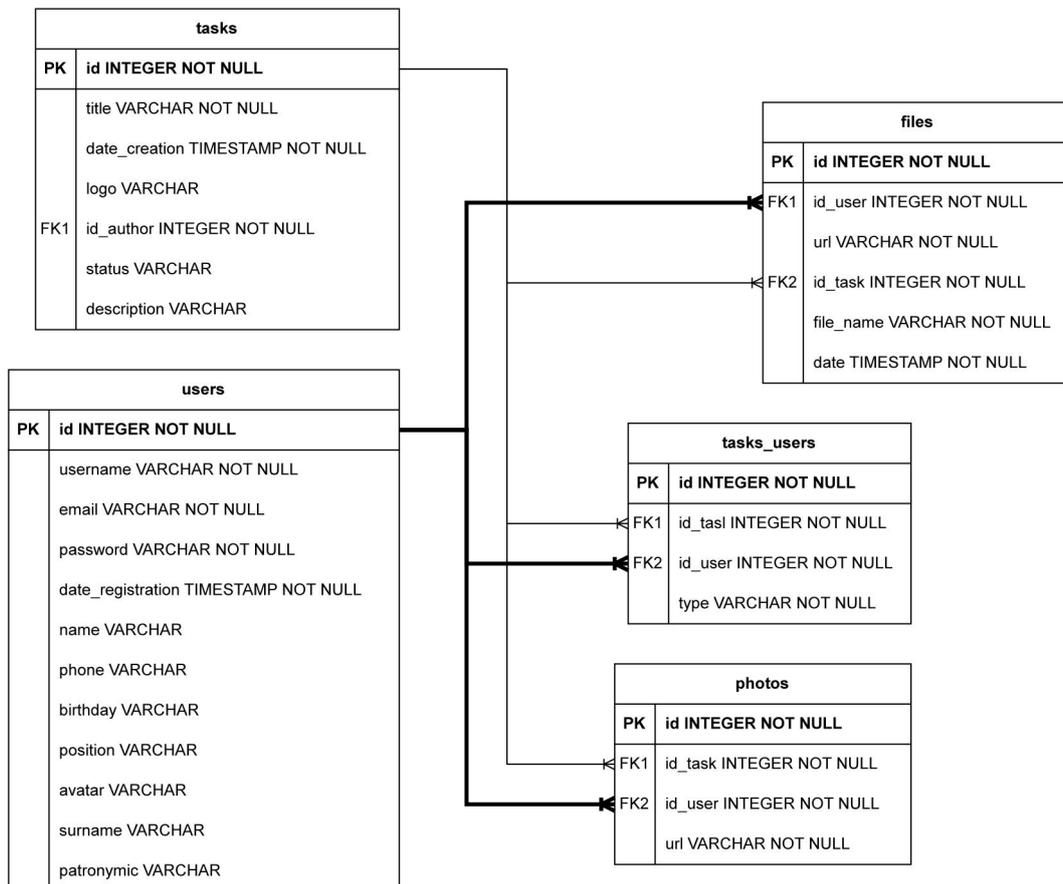


Рисунок 22 – ER-диаграмма базы данных

Нормализация данных проводилась до третьей нормальной формы, что исключило избыточность информации и обеспечило согласованность данных. Например, информация о пользователях вынесена в отдельную таблицу, а связи между задачами и пользователями организованы через промежуточные таблицы. Это позволило избежать повторяющихся строк и снизить риск возникновения аномалий при модификации данных.

Однако были учтены возможные компромиссы между нормализацией и производительностью. Например, в некоторых случаях допускается денормализовать данные ради ускорения часто используемых запросов, таких как вывод всех задач пользователя с деталями исполнителей и контролирующих лиц.

Для защиты данных были применены следующие меры:

- пароли хранятся в зашифрованном виде с использованием алгоритма bcrypt;

- все SQL-запросы выполняются через ORM, что обеспечивает защиту от SQL-инъекций;
- реализована система ролей, ограничивающая действия в зависимости от принадлежности пользователя к определённой группе (например, руководитель, менеджер, исполнитель);
- перед выполнением операций чтения и изменения данных осуществляется проверка прав доступа.

Эти меры позволили минимизировать риски несанкционированного доступа и обеспечить безопасное хранение конфиденциальной информации. Также были созданы UML-диаграммы, представленные в приложениях Б-Г.

3 ОПИСАНИЕ РАЗРАБОТАННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

3.1 Организация процесса разработки

Разработка автоматизированного рабочего места менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ» проводилась с использованием современных инструментов, обеспечивающих высокую скорость работы, надёжность и простоту внедрения. Для локальной разработки был выбран серверный пакет OpenServer, который предоставил гибкую среду для запуска PHP-приложения и взаимодействия с базой данных. Open Server Panel – это портативная программная среда, созданная специально для веб-разработчиков с учётом их рекомендаций и пожеланий. На сегодняшний день OSPanel широко используется веб-разработчиками по всему миру с целью разработки, отладки и тестирования веб-проектов, а также для предоставления веб-сервисов в локальных сетях. Наибольшую популярность проект завоевал у начинающих веб-мастеров, поскольку позволяет быстро создать рабочее окружение и сразу начать изучение веб-технологий без сложных манипуляций по установке и настройке большого количества незнакомых программных компонентов.[5].

Для администрирования базы данных применялась графическая оболочка phpMyAdmin, встроенная в OpenServer. Она позволила оперативно создавать таблицы, управлять данными, выполнять SQL-запросы, а также экспортировать и импортировать данные при необходимости. Интерфейс phpMyAdmin обеспечил наглядное представление структуры базы данных и упростил отладку запросов, особенно на этапе проектирования данных и реализации связей между таблицами.

В качестве основного редактора кода использовался Visual Studio Code, выбранный за его высокую производительность, расширяемость и наличие мощных инструментов для веб-разработки. Благодаря встроенному терминалу, поддержке автодополнения, подсветке синтаксиса и возможности отладки, редактор стал центральным элементом всей разработки. Он позволял эффективно работать с HTML, CSS, JS, PHP и сниппетами MODX, что положительно сказалось на

скорости написания и тестирования кода.

Управление исходным кодом осуществлялось с помощью Git, который обеспечил контроль версий, безопасное хранение истории изменений и возможность отката к предыдущим состояниям системы. Работа велась с двумя основными ветками: main – для стабильной версии, и dev – для активной разработки. Это позволило избежать случайного повреждения рабочей версии и сохранить прозрачность процесса доработки.

Проект хранился в локальном репозитории, где были организованы фичевые ветки для реализации отдельных модулей, таких как формы задач, интерфейс пользователя и система безопасности. Такая структура управления версиями способствовала чёткой координации изменений и облегчила совместную работу с другими разработчиками или руководителем проекта.

Все инструменты были выбраны с учётом открытости, доступности и низкого порога входа, что делает подход экономически выгодным и легко воспроизводимым в других организациях. Использование open-source решений исключило лицензионные затраты и сделало систему доступной даже для небольших команд или индивидуальных разработчиков.

3.2 Реализация клиентской части

Реализация клиентской части автоматизированного рабочего места менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ» осуществлялась с использованием современных веб-технологий, обеспечивающих кроссбраузерную совместимость, адаптивность и высокую степень интерактивности. Основными инструментами стали HTML5, CSS3 и JS, дополненные библиотекой jQuery и фреймворком Bootstrap, что позволило создать функциональный и эстетически привлекательный интерфейс.

Клиентская часть системы отвечает за взаимодействие с пользователем, отображение данных и отправку запросов на сервер. Для обеспечения гибкости и масштабируемости были применены следующие подходы:

- разделение логики представления и данных;

- использование компонентного подхода при верстке интерфейса;
- применение AJAX для динамической подгрузки информации без перезагрузки страницы;
- поддержка мобильных устройств через медиа-запросы и адаптивные элементы управления.

Для верстки интерфейса применялись HTML5 и CSS3, которые позволили структурировать содержимое и задать внешний вид всех элементов. Фреймворк Bootstrap был выбран как основной инструмент для построения сетки, оформления форм и кнопок, что дало возможность добиться единого стиля и корректного отображения на разных устройствах. Также он предоставил готовые компоненты, такие как модальные окна, выпадающие меню и боковые панели, что значительно ускорило разработку.

Для реализации динамического поведения страниц и обработки событий использовался язык программирования JS в сочетании с библиотекой jQuery. Эта комбинация позволила:

- обрабатывать действия пользователя, такие как клики, наведение курсора, изменение полей формы;
- выполнять AJAX-запросы для получения и отправки данных без перезагрузки страницы;
- добавлять анимацию и эффекты взаимодействия;
- повышать отзывчивость интерфейса и общее качество пользовательского опыта.

Одним из ключевых элементов клиентской части стали формы, через которые пользователь взаимодействует с системой. Были разработаны формы:

- регистрации и авторизации;
- создания и редактирования задач;
- изменения профиля пользователя;
- загрузки файлов.

Формы были протестированы на предмет корректности ввода данных,

предусмотрена валидация полей как на стороне клиента (через JS), так и на стороне сервера (через PHP). Такой подход позволил предотвратить попадание некорректных данных в базу и повысить надежность всей системы.

Адаптивность интерфейса была достигнута за счет применения медиа-запросов в CSS, что позволило точно настраивать отображение элементов на различных разрешениях экрана. Также были реализованы скрытые панели и выпадающие меню, что сделало систему одинаково удобной для использования как на десктопах, так и на мобильных устройствах. Упрощённые формы и элементы управления были адаптированы для малых экранов, что повысило удобство работы вне офиса.

Все элементы интерфейса были сверстаны в соответствии с макетами и требованиями к дизайну. Шапка сайта содержит логотип системы, навигационное меню и кнопки входа или аватар пользователя, если он авторизован. Основная часть состоит из центрального блока с информацией о задаче и бокового меню со списком задач. Настройки интерфейса позволяют переключаться между темами, что положительно влияет на зрительное восприятие информации в разных условиях освещения.

В ходе реализации клиентской части были учтены такие аспекты, как:

- масштабируемость – стили и компоненты спроектированы так, чтобы их можно было использовать повторно в других разделах системы;
- чистый и понятный код – все файлы HTML и CSS были структурированы и прокомментированы, что облегчило последующее сопровождение;
- безопасность – данные экранировались перед выводом в интерфейс, чтобы исключить возможность XSS-атак.

После доработки и тестирования интерфейс полностью соответствовал заявленным требованиям и обеспечивал быстрое выполнение операций, высокую степень удовлетворённости пользователей и простоту освоения.

3.3 Реализация серверной логики

Реализация серверной части автоматизированного рабочего места менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ» проводилась с использованием языка программирования PHP, что обусловлено его широкой поддержкой, простотой интеграции с базами данных и гибкостью при разработке веб-приложений. Для упрощения взаимодействия с данными была применена ORM RedBeanPHP, которая позволила работать с таблицами базы данных в объектно-ориентированном виде и обеспечила защиту от SQL-инъекций за счёт автоматического экранирования входящих данных.

Организация серверной логики предусматривала обработку HTTPS-запросов, формирование ответов клиентской части и обеспечение безопасности передачи данных между браузером и сервером. Все операции выполнялись через стандартные методы GET и POST, что дало возможность использовать единый подход к обработке запросов и упростило интеграцию с клиентским интерфейсом. Такая реализация способствовала созданию единой системы, где клиентская часть отвечает за представление, а серверная – за бизнес-логику и работу с данными.

Для управления контентом и маршрутизацией применялась система управления контентом MODX, выбранная за её гибкость и удобство администрирования. MODX предоставила возможность создания шаблонов страниц, организации динамической подгрузки информации через сниппеты и настройки ЧПУ (человекопонятных URL). Это позволило упростить навигацию по системе и сделать адреса более понятными как для пользователей, так и для последующего сопровождения кода.

Обмен данными между клиентом и сервером осуществлялся через AJAX-запросы, что исключило необходимость перезагрузки страницы и повысило отзывчивость интерфейса. Такой подход сделал возможным моментальное выполнение действий, таких как изменение статуса задачи или добавление нового исполнителя, без потери контекста.

CMS MODX сыграла важную роль в организации серверной логики. Она позволила легко управлять шаблонами, внедрять сниппеты и формировать

динамическое содержание страниц. Например, snippet `getTaskDetails` обеспечивал выборку данных о задаче из базы и формирование JSON-ответа, который затем использовался клиентской частью для отображения информации. Такой способ отделения логики от представления упростил поддержку и расширяемость системы.

Все данные хранились в реляционной СУБД MySQL версии 5.7, выбранной за её производительность и надёжность при работе с относительно небольшим объёмом данных. База данных включала таблицы `users`, `tasks`, `files`, `photos` и `tasks_users`, которые связаны между собой внешними ключами и обеспечивают целостность информации.

Для работы с данными была внедрена ORM RedBeanPHP, которая позволила абстрагироваться от написания SQL-запросов и сосредоточиться на логике приложения. Вместо традиционных команд `SELECT` и `INSERT` использовались методы вроде `R::load()` и `R::store()`, что значительно ускорило разработку и сделало код более читаемым и безопасным. Также эта ORM обеспечила простую миграцию структуры базы данных, позволяя адаптировать её под новые требования без значительных трудозатрат.

Защита данных стала важной частью реализации серверной логики. Все входящие данные проверялись и фильтровались перед дальнейшей обработкой. Это позволило минимизировать риск ошибок и повысить уровень защиты от XSS и CSRF-атак. Пароли пользователей хранятся в зашифрованном виде с использованием функции `password_hash()`. Проверка прав доступа проводится до выполнения любой операции изменения или удаления данных. Это ограничило действия пользователей в зависимости от их роли: администратор, менеджер или исполнитель.

Для повышения уровня безопасности также применялись токены при отправке форм, что помогло защитить систему от подделки межсайтовых запросов. Все выходные данные, выводимые на страницу, подвергались экранированию, что исключило возможность запуска вредоносного кода в браузере клиента.

Процесс обработки запросов был организован следующим образом. Клиент

отправляет данные, например, создаёт новую задачу. Сервер получает запрос и проверяет:

- авторизацию пользователя;
- корректность введённых данных;
- наличие прав на выполнение действия.

Если все условия соблюдены, данные передаются в ORM RedBeanPHP для сохранения, обновления или удаления. После этого формируется ответ, который отправляется обратно клиенту в формате JSON или HTML.

Такой подход обеспечил чёткое разделение уровней логики и представления, а также позволил избежать дублирования кода и упростил его последующее сопровождение. Система получилась модульной, и каждая операция могла быть доработана или расширена без влияния на остальные компоненты.

3.4 Интеграция с базой данных

При разработке автоматизированного рабочего места менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ» особое внимание было уделено интеграции с базой данных, поскольку от эффективности взаимодействия с данными зависит надёжность всей системы. Для хранения информации была выбрана реляционная СУБД MySQL 5.7, которая обеспечивает высокую производительность, поддержку современных стандартов SQL и широкую совместимость с PHP. Такой выбор позволил организовать данные в виде таблиц, установить чёткие связи между сущностями и обеспечить целостность информации.

Интеграция с базой данных осуществлялась через ORM RedBeanPHP, которая стала ключевым инструментом для работы с данными. Использование ORM позволило исключить необходимость ручного написания SQL-запросов, повысить читаемость кода и упростить его сопровождение. Все операции добавления, изменения и удаления записей выполнялись через объектно-ориентированный интерфейс, что положительно сказалось на скорости разработки и безопасности системы.

Этот подход сделал работу с базой данных более понятной и устойчивой к

ошибкам, особенно при динамическом изменении структуры данных или расширении функционала.

Одним из основных преимуществ RedBeanPHP стало её поведение при обращении к несуществующим сущностям. При вызове методов, связанных с новыми таблицами или полями, ORM автоматически создавала недостающие элементы в базе данных. Это дало возможность быстро прототипировать систему и адаптировать её под изменяющиеся требования. Например, если в системе появлялось новое поле `priority`, то ORM позволяла работать с ним сразу после указания в коде, без необходимости предварительного создания столбца в MySQL.

Также была реализована система миграций, предназначенная для сохранения истории изменений структуры базы данных. Каждое изменение (добавление нового поля, изменение типа данных, создание новых таблиц) фиксировалось в виде отдельного SQL-скрипта. Это позволило:

- восстанавливать актуальную структуру базы данных при переносе на другой сервер;
- избежать потери данных при обновлении;
- контролировать изменения в БД на всех этапах жизненного цикла проекта.

Миграции хранились в отдельной директории и применялись вручную или через сниппеты MODX, что делало процесс управления структурой данных гибким и управляемым.

Все таблицы были спроектированы с учётом принципов нормализации, чтобы исключить дублирование данных и обеспечить их согласованность. Основные таблицы включают:

- `users` – пользователи системы;
- `tasks` – задачи, которыми оперирует система;
- `files` – файлы, прикрепленные к задачам;
- `photos` – фотографии, связанные с пользователями;
- `tasks_users` – промежуточная таблица для связи «многие-ко-многим».

Связь между таблицами организована через внешние ключи, что обеспечило логическую целостность данных. Например, каждая задача имеет идентификатор автора (`id_author`), который ссылается на таблицу `users`. Аналогично, файл связан с задачей через `id_task`.

Такой подход к организации данных позволил строить сложные выборки с использованием JOIN-запросов, а также упростил анализ информации при работе с системой.

Для оптимизации производительности были применены следующие меры:

- использование индексов для часто используемых полей,
- ограничение объема выбираемых данных,
- применение кэширования на уровне PHP.
- Эти действия повысили скорость выполнения запросов и снизили нагрузку на сервер при увеличении объёма данных.

Защита от SQL-инъекций была обеспечена за счёт использования ORM, которая автоматически экранирует входящие данные. Таким образом, все операции с базой данных проводились безопасным способом, исключая возможность внедрения вредоносного SQL-кода через поля форм.

Интеграция с CMS MODX позволила использовать сниппеты для получения данных из базы и передачи их клиентской части. Например, сниппет `getTaskDetails` отвечал за выборку информации о задаче, а сниппет `getAllUsers` – за формирование списка пользователей для назначения исполнителей. Такая организация обеспечила чёткое разделение логики и представления, а также упростила тестирование и последующее сопровождение.

Интеграция с базой данных была проведена с учётом требований к масштабируемости и безопасности. Система позволяет легко расширять функционал за счёт добавления новых таблиц или полей, а также обеспечивает защиту конфиденциальной информации. Применение ORM `RedBeanPHP` сыграло важную роль в ускорении разработки и повышении надёжности взаимодействия с данными.

3.5 Реализация мер информационной безопасности

При разработке автоматизированного рабочего места менеджера проектов в ООО «ВОСТОК-ТЕХНОЛОГИЯ» особое внимание было уделено вопросам информационной безопасности, поскольку система оперирует чувствительными данными: учетные записи сотрудников, задачи, документы и другие материалы. В связи с этим были внедрены меры по защите информации как на уровне клиента, так и на сервере и базе данных.

Одной из ключевых мер стало использование протокола HTTPS, что позволило шифровать данные при передаче между клиентом и сервером. Это исключило возможность прослушивания трафика и перехвата конфиденциальной информации, такой как пароли, сессии и содержание задач.

Для обеспечения защиты от SQL-инъекций применялась ORM RedBeanPHP, которая автоматически экранирует входящие данные. Все операции с базой данных выполнялись через объектно-ориентированный интерфейс, что сделало работу с информацией более безопасной и предотвратило возможность внедрения вредоносных SQL-запросов.

Дополнительно была организована проверка типов данных и ограничения на уровне PHP. Например, все входящие данные проходили фильтрацию:

- электронная почта проверялась с помощью функции `filter_var()` с флагом `FILTER_VALIDATE_EMAIL`;
- пароли хранились в зашифрованном виде с использованием `password_hash()` и `password_verify()`;
- пользовательский ввод экранировался перед выводом на страницу для предотвращения XSS-атак;

Для хранения сессий применялись стандартные механизмы PHP, дополненные механизмом истечения времени жизни и автоматического выхода при длительном простое. Также был реализован контроль над количеством попыток входа, что помогло минимизировать риски подбора пароля.

4 БЕЗОПАСНОСТЬ И ЭКОЛОГИЧНОСТЬ

Обеспечение безопасности человека в условиях чрезвычайных ситуаций является одной из ключевых задач современного общества. Безопасность жизнедеятельности как научная дисциплина направлена на изучение факторов окружающей среды, способных нанести вред здоровью человека или окружающей природной среде, а также на разработку мер по предупреждению и минимизации последствий возникающих угроз. Особое значение эта область знаний имеет в профессиональной деятельности, связанной с использованием информационных технологий и длительной работой за компьютером, поскольку именно такие условия могут стать источником как физических, так и психоэмоциональных рисков.

В рамках настоящей главы рассматриваются основные аспекты безопасности, экологичности и готовности к чрезвычайным ситуациям, актуальные для использования разработанной информационной системы в офисных условиях ООО «ВОСТОК-ТЕХНОЛОГИЯ». Анализ охватывает организацию безопасного рабочего места, влияние проекта на окружающую среду, а также возможные внештатные ситуации, которые могут возникнуть в ходе эксплуатации технических средств. Представленные рекомендации направлены на обеспечение устойчивого функционирования системы, сохранения здоровья сотрудников и соблюдения требований экологической безопасности.

4.1 Безопасность

При разработке и эксплуатации информационной системы особое внимание уделялось обеспечению безопасных условий труда пользователей, работающих за персональными компьютерами. Рабочее место пользователя должно соответствовать эргономическим и санитарно-гигиеническим требованиям, установленным действующими нормативными документами, такими как СанПиН 1.2.3685-21, ГОСТ Р 50948-2001, ГОСТ Р 50923-96. Основные параметры организации рабочего места указаны в таблице 7.

Таблица 7 – Параметры организации рабочего места

Параметр	Нормативное значение
Высота стола	72–76 см
Расстояние от глаз до монитора	Не менее 50 см
Угол наклона монитора	10–15°
Освещенность помещения	Не менее 300 лк
Уровень шума	Не более 50 дБА
Температура воздуха	20–24 °С
Относительная влажность	40–60 %

Для снижения утомляемости и профилактики зрительного напряжения рекомендуется соблюдение режима труда и отдыха: каждые 50 минут работы необходимо делать 5-10 минут перерыва.

Эргономика графического интерфейса разрабатывалась с учётом требований ГОСТ Р ИСО 9241-161-2016 и ГОСТ Р 52872-2019. Интерфейс системы имеет чёткую структуру, интуитивно понятную навигацию и достаточный контраст между текстом и фоном. Размер шрифта не меньше 14 пунктов, что делает информацию легко воспринимаемой. Также реализована поддержка светлой и тёмной тем оформления, что позволяет пользователю выбрать наиболее комфортный вариант. Дизайн адаптирован под мобильные устройства, что расширяет доступность системы.

Информационная безопасность системы обеспечивается на нескольких уровнях. Все данные перед выводом экранируются для предотвращения XSS-атак.

Защита от SQL-инъекций реализована через использование ORM RedBeanPHP, исключающей возможность выполнения вредоносных запросов. Пароли хранятся в зашифрованном виде с использованием алгоритма bcrypt. Обмен данными между клиентом и сервером осуществляется по протоколу HTTPS, что гарантирует конфиденциальность и целостность информации.

Система использует ролевую модель доступа, которая определяет права пользователей: администратор, менеджер, исполнитель. Каждому уровню доступа соответствует свой набор возможностей, что позволяет избежать несанкционированного изменения или удаления данных. Перед выполнением любой операции проверяется авторизация пользователя и его права на выполнение действия.

Электробезопасность обеспечивается через использование заземлённого оборудования, сетевых фильтров и источников бесперебойного питания. Проводка соответствует правилам устройства электроустановок (ПУЭ), а техническое состояние оборудования регулярно проверяется ответственными лицами.

В процессе разработки были учтены основные аспекты безопасности: организация рабочего места, эргономика пользовательского интерфейса, защита информации и электробезопасность. Предложенные меры позволяют использовать информационную систему в соответствии с современными требованиями охраны труда и информационной безопасности.

4.2 Экологичность

Разработанная информационная система, как цифровое решение, не оказывает прямого негативного воздействия на окружающую среду. Однако эксплуатация сопутствующих технических средств – таких как персональные компьютеры, серверное оборудование, оргтехника и мебель – связана с экологическими аспектами, которые необходимо учитывать при использовании системы в офисной среде.

Особое внимание уделяется вопросам обращения с отходами, образующимися в процессе эксплуатации оборудования, а также соблюдению

современных требований к энергоэффективности и рациональному использованию ресурсов. Это связано с необходимостью соответствовать нормативным требованиям, включая Федеральный закон № 89-ФЗ «Об отходах производства и потребления», а также СанПиН 1.2.3685-21 и СанПиН 2.1.3684-21.

Наибольшее экологическое значение имеют такие категории отходов, как:

- вышедшая из строя оргтехника (мониторы, системные блоки, принтеры);
- картриджи и тонер;
- люминесцентные лампы;
- бумажные отходы;
- упаковочные материалы;
- мебель и прочие предметы интерьера.

Все эти отходы требуют специального подхода к сбору, хранению и утилизации, особенно если они содержат потенциально опасные вещества, такие как тяжёлые металлы, ртуть или полимеры. Неправильная утилизация может привести к загрязнению почвы, воды и атмосферы, что противоречит принципам устойчивого развития и ответственного управления ресурсами.

Для минимизации экологического влияния в ООО «ВОСТОК-ТЕХНОЛОГИЯ» внедрены следующие меры:

- использование бумаги ограничено за счёт перевода большинства операций в электронный вид;
- организовано отдельное накопление отходов с последующей передачей их специализированным организациям, имеющим лицензию на утилизацию;
- приобретается офисная техника с высоким классом энергоэффективности, что снижает потребление электроэнергии;
- предусмотрена возможность вторичного использования или ремонта техники перед её списанием.

Списание техники проводится в соответствии с установленными

процедурами: составляется акт о выходе из строя, определяется возможность восстановления устройства, после чего заключается договор с лицензированной организацией по утилизации. Все этапы документально фиксируются, что обеспечивает прозрачность и соответствие требованиям законодательства.

Кроме того, в компании проводится разъяснительная работа среди сотрудников по вопросам экологической безопасности, включая правильное обращение с отходами и бережное отношение к ресурсам. Это способствует формированию экологической культуры и повышает уровень осведомлённости персонала.

Таким образом, деятельность организации направлена на снижение антропогенной нагрузки на окружающую среду через внедрение рациональных практик обращения с отходами, использование энергосберегающих технологий и формирование экологической ответственности среди сотрудников. Предложенные меры позволяют использовать разработанную информационную систему в рамках концепции устойчивого развития и соответствовать современным требованиям экологической безопасности.

4.3 Чрезвычайные ситуации

При эксплуатации информационной системы и связанного с ней оборудования в офисной среде необходимо учитывать возможные чрезвычайные ситуации, которые могут повлиять как на здоровье сотрудников, так и на сохранность данных и технических средств. Анализ потенциальных рисков позволяет выявить наиболее вероятные виды аварийных ситуаций и разработать меры по их предупреждению и ликвидации.

Наиболее распространёнными чрезвычайными ситуациями в офисных помещениях являются пожары, аварии с электрическим оборудованием, затопления из-за прорыва труб отопления или водоснабжения, а также внезапные перебои с электроснабжением. Эти события могут привести к остановке работы системы, потере данных, порче имущества и даже угрозе жизни и здоровью персонала.

Для минимизации последствий подобных ситуаций в ООО «ВОСТОК-ТЕХНОЛОГИЯ» реализованы следующие меры профилактики:

- система пожаротушения и оповещения: в помещениях установлены датчики дыма и тепла, автоматические системы пожаротушения, а также звуковые и световые оповещатели. Регулярно проводятся проверки работоспособности систем;

- наличие первичных средств пожаротушения: огнетушители размещены в доступных местах, соответствующих нормам пожарной безопасности;

- электробезопасность: вся используемая техника заземлена, применяются сетевые фильтры и источники бесперебойного питания. Проводка регулярно проверяется на соответствие требованиям ПУЭ;

- план эвакуации: на каждом этаже размещены схемы эвакуации, указывающие пути выхода и места сбора сотрудников. Персонал проходит инструктажи по действиям при возникновении чрезвычайной ситуации;

- резервное копирование данных: база данных системы резервируется ежедневно, что позволяет восстановить данные в случае аппаратного сбоя или других непредвиденных обстоятельств;

- автономное питание серверов: используются ИБП (источники бесперебойного питания), обеспечивающие временную работу серверов при отключении электроэнергии и позволяющие корректно завершить процессы.

Особое внимание уделяется подготовленности персонала к действиям в условиях чрезвычайной ситуации. Все сотрудники проходят обязательный инструктаж по охране труда и пожарной безопасности. Руководящий состав обучен действиям в режиме повышенной готовности, включая организацию эвакуации и взаимодействие с экстренными службами.

В случае возникновения чрезвычайной ситуации предусмотрена система оповещения через внутреннюю телефонную связь, SMS-рассылку и тревожные кнопки, расположенные в ключевых точках офиса. Это позволяет оперативно реагировать на происшествия и минимизировать время на принятие решений.

Анализ существующих мер показал, что организация в целом готова к действию в условиях чрезвычайных ситуаций. Однако рекомендуется дополнительно:

- проводить учения по эвакуации не реже одного раза в год;
- обновлять средства защиты и оборудование в соответствии с техническими регламентами;
- расширить возможности резервного хранения данных за пределами офиса (например, в облачном хранилище).

Таким образом, в ООО «ВОСТОК-ТЕХНОЛОГИЯ» созданы условия для эффективного реагирования на чрезвычайные ситуации. Предложенные мероприятия позволят повысить уровень готовности организации и обеспечить безопасность сотрудников, оборудования и данных при возникновении внештатных ситуаций.

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы было разработано автоматизированное рабочее место менеджера проектов для ООО «ВОСТОК-ТЕХНОЛОГИЯ», позволяющее повысить эффективность управления задачами за счёт централизации данных, систематизации рабочих процессов и упрощения коммуникации между участниками. Реализованная система представляет собой веб-приложение, которое обеспечивает создание, редактирование, назначение и отслеживание задач, а также прикрепление документов, управление пользователями и контроль сроков выполнения.

Цель исследования – разработка программного обеспечения, позволяющего оптимизировать процессы координации задач – была достигнута. Для этого были решены следующие задачи: проведён анализ предметной области; определены функциональные требования; выбраны и обоснованы технологии реализации; спроектирована архитектура системы и пользовательский интерфейс; реализованы модули создания задач, назначения исполнителей и контролирующих лиц; выполнено тестирование системы и пилотное внедрение в условиях реального предприятия.

Созданное решение позволяет значительно ускорить выполнение операций по управлению задачами, исключает дублирование информации и делает данные доступными в одном месте. Это положительно влияет на прозрачность рабочих процессов и снижает количество ошибок при распределении задач. Система работает стабильно под нагрузкой до 50 активных пользователей, что соответствует требованиям компании и типичным условиям эксплуатации.

Тестирование показало высокую степень надёжности системы. Были выявлены и устранены технические и интерфейсные ошибки, после чего система прошла повторную проверку и была передана в пилотное использование. По результатам анкетирования пользователи отметили простоту освоения, интуитивность навигации и удобство работы с задачами.

Пилотное внедрение в ООО «ВОСТОК-ТЕХНОЛОГИЯ» продемонстрировало практическую значимость разработанного решения. Система позволила улучшить взаимодействие внутри команды, упростить отслеживание прогресса задач и повысить уровень контроля над сроками их выполнения. Также была отмечена возможность дальнейшего развития системы, включая интеграцию с корпоративными инструментами, добавление мобильного клиента и реализацию механизма автоматических отчётов.

Программный продукт может быть рекомендован к использованию не только в ООО «ВОСТОК-ТЕХНОЛОГИЯ», но и в других организациях, где требуется гибкое и удобное средство управления задачами без избыточной сложности и лицензионных затрат. Его преимуществом является открытость технологий, экономическая выгода и простота сопровождения, что делает его перспективным решением для малого и среднего бизнеса.

Таким образом, работа показала, что разработка информационной системы управления задачами является актуальной и востребованной задачей. Полученный опыт подтвердил применимость выбранных технологий и методов проектирования в реальных условиях, а также позволил получить навыки, необходимые для участия в разработке аналогичных программных продуктов в будущем.

БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ

- 1 Bootstrap 5 Documentation // getbootstrap.com: офиц. сайт. 16.04.2025.
URL:<https://getbootstrap.ru/docs/5.1> (дата обращения: 15.05.2025).
- 2 RedBeanPHP Official Site // redbeanphp.com: офиц. сайт. 18.03.2023.
URL:<https://redbeanphp.com> (дата обращения: 16.05.2025).
- 3 JavaScript Guide // developer.mozilla.org: офиц. сайт. 25.03.2025.
URL:<https://developer.mozilla.org/ru/docs/Web/JavaScript> (дата обращения: 17.05.2025).
- 4 PHP Manual // php.net: офиц. сайт. 06.04.2025.
URL:<https://www.php.net/manual/ru> (дата обращения: 18.05.2025).
- 5 Документация OSPanel // ospanel.io: офиц. сайт. URL:<http://ospanel.io> (дата обращения: 19.05.2025).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 Буч, Градди. Объектно-ориентированный анализ и проектирование с примерами приложений / Градди Буч. – СПб.: Питер, 2021. – 512 с.
- 2 Власов, В. К. Проектирование информационных систем / В. К. Власов. – Москва: Академия, 2022. – 368 с.
- 3 Дунаев, Владимир. Bootstrap 5: создание современных сайтов / В. Дунаев. – Москва: Бином, 2022. – 288 с.
- 4 Захаров, А. А. Разработка динамических веб-сайтов на PHP и MySQL / А. А. Захаров. – Новосибирск: СибГУТИ, 2020. – 192 с.
- 5 Зенкин, А. А. Управление проектами разработки ПО / А. А. Зенкин. – Санкт-Петербург: БХВ-Петербург, 2021. – 416 с.
- 6 Кардаш, Т. А. Эргономика рабочих мест служащих и инженерно-технических работников, оснащенных ПЭВМ [Текст]: учеб. пособие / Т. А. Кардаш ; АмГУ, ИФФ. - Благовещенск: Изд-во Амур. гос. ун-та, 2002. - 60 с.
- 7 Карпов, Ю. Г. Теория и практика верификации программ / Ю. Г. Карпов. – СПб.: БХВ-Петербург, 2021. – 384 с.
- 8 Красиков, И. В. Разработка требований к программному обеспечению / И. В. Красиков. – Москва: ДМК Пресс, 2020. – 256 с.
- 9 Крючкова, Е. Н. Объектно-ориентированное программирование: архитектурное проектирование и паттерны программирования: учебно-методическое пособие / Е. Н. Крючкова, С. М. Старолетов. – Барнаул: АлтГТУ, 2020. – 180 с.
- 10 Макконнелл, Стив. Совершенный код. Мастер-класс / Стив Макконнелл. – СПб.: Питер, 2020. – 820 с.
- 11 Мартин, Роберт К. Чистый код: создание, анализ и рефакторинг. Библия разработчика / Роберт К. Мартин. – СПб.: Питер, 2021. – 464 с.
- 12 Миронов, А. А. Введение в теорию баз данных / А. А. Миронов. – Москва: Бином, 2021. – 240 с.

- 13 MODX Revolution: Создание сайтов на CMS MODX / под ред. С.И. Петрова. – Москва: ДМК Пресс, 2021. – 272 с.
- 14 Николайко, Евгений. Разработка веб-приложений на PHP и MySQL / Е. Николайко. – Москва: ДМК Пресс, 2020. – 256 с.
- 15 Пособие по безопасной работе на персональных компьютерах [Текст] / разраб. В. К. Шумилин. - М.: НЦ ЭНАС, 2005. - 28 с.
- 16 Рубин, Харви. Управление жизненным циклом программного обеспечения / Харви Рубин. – Москва: ДМК Пресс, 2020. – 272 с.
- 17 Румбах, Джеймс. Введение в управление проектами программного обеспечения / Джеймс Румбах. – Москва: Диалектика, 2020. – 288 с.
- 18 Шабанов, Иван. Основы веб-программирования: HTML, CSS, JavaScript / И. Шабанов. – Москва: Лаборатория знаний, 2021. – 320 с.
- 19 Шумилин, В. К. ПЭВМ. Защита пользователя [Текст] / Шумилин В. К. – М.: Охрана труда и социальное страхование, 2001. - 214с.
- 20 Фаулер, Мартин. Шаблоны корпоративных приложений. Полный справочник / Мартин Фаулер. – СПб.: Питер, 2020. – 544 с.
- 21 Фленов, Михаил Евгеньевич. PHP: Полный справочник / М. Е. Фленов. – СПб.: Питер, 2022. – 608 с.
- 22 Фримен, Энди. Изучаем SQL программированию на практике / Энди Фримен, Эллен Принс. – СПб.: Питер, 2021. – 368 с.
- 23 Ульянов, М. В. Информационная безопасность. Основы теории / М. В. Ульянов. – Москва: ИНФРА-М, 2022. – 304 с.
- 24 Уэстерфельд, Тони. Разработка интерфейсов с использованием jQuery / Т. Уэстерфельд. – Москва: Диалектика, 2020. – 240 с.
- 25 jQuery API Documentation [Электронный ресурс] // jQuery Foundation. – Режим доступа: <https://api.jquery.com/> – 05.04.2025.
- 26 MDN Web Docs. JavaScript Guide [Электронный ресурс] // Mozilla Developer Network. – Режим доступа: <https://developer.mozilla.org/ru/docs/Web/JavaScript/Guide> – 05.04.2025.

27 MODX Revolution Documentation [Электронный ресурс] // MODX LLC. – Режим доступа: <https://docs.modx.com/revolution/latest/> – 05.04.2025.

28 MySQL Reference Manual [Электронный ресурс] // Oracle Corporation. – Режим доступа: <https://dev.mysql.com/doc/refman/8.0/en/> – 05.04.2025.

29 PHP: Hypertext Preprocessor. Official Documentation [Электронный ресурс] // The PHP Group. – Режим доступа: <https://www.php.net/manual/> – 05.04.2025.

30 RedBeanPHP Official Site [Электронный ресурс] // RedBeanPHP. – Режим доступа: <https://redbeanphp.com/> – 05.04.2025.

31 W3C CSS3 Specification [Электронный ресурс] // World Wide Web Consortium. – Режим доступа: <https://www.w3.org/Style/CSS/specs.en.html> – 05.04.2025.

32 W3C HTML5 Specification [Электронный ресурс] // World Wide Web Consortium. – Режим доступа: <https://html.spec.whatwg.org/multipage/> – 05.04.2025.

ПРИЛОЖЕНИЕ А

Техническое задание

1. ОБЩИЕ СВЕДЕНИЯ

1.1 Полное наименование системы

Автоматизированное рабочее место (АРМ) менеджера проектов для координации задач в ООО «ВОСТОК-ТЕХНОЛОГИЯ»

Условное обозначение: TaskManager

1.2 Цель создания программы

Целью работы является разработка программного обеспечения, предназначенного для повышения эффективности управления задачами в ООО «ВОСТОК-ТЕХНОЛОГИЯ». Система должна позволять:

- централизованно хранить информацию о задачах;
- назначать исполнителей и контролирующих лиц;
- прикреплять документы к задачам;
- отслеживать сроки выполнения задач;
- обеспечивать безопасность данных и контроль доступа.

1.3 Источники разработки

Задание на выпускную квалификационную работу.

Методичка по оформлению ВКР.

Анализ предметной области и аналогов.

Данные о бизнес-процессах ООО «ВОСТОК-ТЕХНОЛОГИЯ».

2. НАЗНАЧЕНИЕ РАЗРАБАТЫВАЕМОЙ СИСТЕМЫ

Созданная система предназначена для внутреннего использования в ООО «ВОСТОК-ТЕХНОЛОГИЯ», а также может быть внедрена в других организациях, где требуется централизованное управление задачами между сотрудниками. Она позволяет:

- упростить распределение задач,
- повысить прозрачность процессов,
- исключить дублирование информации,

Продолжение Приложения А

- организовать единую информационную среду,
- снизить количество ошибок при планировании и распределении задач.

3. ТРЕБОВАНИЯ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ

3.1 Функциональные требования

№	Модуль	Требования
1	Авторизация и регистрация	Возможность регистрации нового пользователя; вход в систему с проверкой прав; восстановление пароля через email.
2	Задачи	Создание, редактирование, удаление задач; изменение статуса задачи; назначение исполнителей и контролирующих лиц.
3	Профиль пользователя	Просмотр и редактирование личных данных; загрузка аватара; просмотр списка задач и документов.
4	Документы	Прикрепление файлов к задачам; просмотр и скачивание прикрепленных документов; ограничение форматов загружаемых файлов.
5	Безопасность	Защита от SQL-инъекций, XSS и CSRF; ролевая модель доступа; шифрование паролей; проверка прав перед операциями изменения и удаления.

3.2 Требования к интерфейсу

Интерфейс должен быть адаптивным под мобильные устройства.

Все формы должны быть интуитивно понятны без дополнительного обучения.

Навигация – простая и логически последовательная.

Поддержка тем оформления (светлая/тёмная).

Быстрая реакция на действия пользователя.

3.3 Требования к надёжности

Продолжение Приложения А

Система должна работать без сбоев при нагрузке до 50 пользователей одновременно.

Должна быть реализована система бэкапов базы данных.

Необходимо предусмотреть механизм логирования событий.

3.4 Требования к безопасности

Пароли должны храниться в зашифрованном виде (bcrypt).

Должна быть реализована система ролей (менеджер, исполнитель, администратор).

Все операции с данными должны проходить проверку прав доступа.

Обязательна защита от SQL-инъекций и XSS-атак.

Все данные должны передаваться по защищённому протоколу HTTPS.

3.5 Требования к видам обеспечения

Математическое обеспечение

Алгоритмы работы с задачами и документами.

Логика назначения исполнителей и контролирующих лиц.

Программное обеспечение

Сервер: PHP, MySQL, Apache/Nginx.

Клиент: HTML5, CSS3, JavaScript, jQuery, Bootstrap.

CMS: MODX Revolution.

ORM: RedBeanPHP.

Разработка с использованием веб-сервера OPanel OpenServer и ИСР Visual Studio Code.

Информационное обеспечение

Структура базы данных спроектирована с учётом нормализации.

Все таблицы связаны внешними ключами.

Предусмотрены механизмы миграции структуры базы данных БД.

Техническое обеспечение

Поддержка стандартных web-браузеров: Chrome, Firefox, Edge, Safari.

Продолжение Приложения А

Совместимость с Windows, macOS и Linux.

Доступность с мобильных устройств (iOS, Android).

Отсутствие требований к специальному оборудованию.

4. СОСТАВ И СОДЕРЖАНИЕ РАБОТ ПО СОЗДАНИЮ СИСТЕМЫ

4.1 Этапы разработки:

анализ предметной области и текущих процессов в компании;

выбор и обоснование технологий реализации;

проектирование архитектуры системы и модели данных;

разработка клиентской части (интерфейса);

реализация серверной логики и взаимодействия с БД;

тестирование системы (функциональное, нагрузочное, юзабилити);

пилотное внедрение и доработка на основе обратной связи;

подготовка технической документации и руководства пользователя.

4.2 Сроки выполнения:

начало работ: 16.12.2024 ;

окончание работ: 16.06.2025 .

5. ПОРЯДОК КОНТРОЛЯ И ПРИЁМКИ СИСТЕМЫ

5.1 Контроль качества

Проверка соответствия функционала техническому заданию.

Юзабилити-тестирование среди сотрудников ООО «ВОСТОК-ТЕХНОЛОГИЯ».

Функциональное тестирование всех модулей.

Нагрузочное тестирование на 50 активных пользователей.

5.2 Приёмка после внедрения

Сдача системы с актом внедрения.

Обучение пользователей работе с системой.

Передача материалов по сопровождению.

Продолжение Приложения А

6. СООТВЕТСТВИЕ СТАНДАРТАМ И НОРМАТИВНЫМ ДОКУМЕНТАМ

ГОСТ 34.2-2019 – Информационная технология. Сопровождение программного обеспечения. Терминология.

ГОСТ 19.102-77 – ЕСПД. Стадии разработки программного обеспечения.

ГОСТ 19.504-79 – ЕСПД. Методы программного обеспечения.

ГОСТ 19.501-78 – ЕСПД. Стандарты по программированию.

ГОСТ 19.101-77 – ЕСПД. Виды программной документации.

7. ЛИТЕРАТУРА

ГОСТ 19.101-77. ЕСПД. Виды программной документации [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1203411> (дата обращения: 10.04.2025).

ГОСТ 34.2-2019. Информационная технология. Сопровождение программного обеспечения. Терминология [Электронный ресурс]. – Режим доступа: https://www.gost.ru/wps/portal/gost/public/standards/search_standard/34_2_2019 (дата обращения: 10.04.2025).

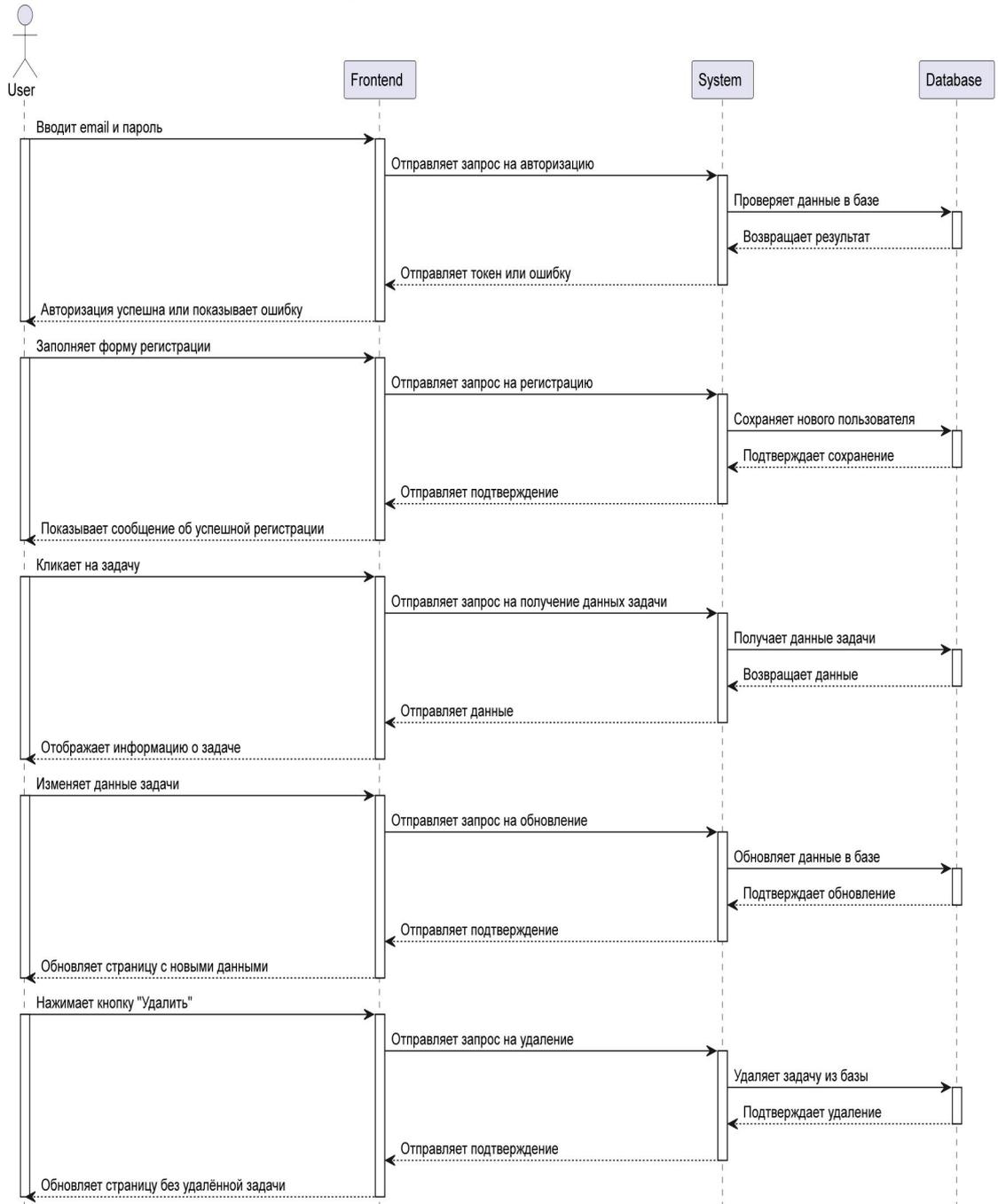
Документация по технологии Web NFC API [Электронный ресурс]. – Режим доступа: https://developer.mozilla.org/ru/docs/Web/API/Web_NFC_API (дата обращения: 10.04.2025).

Документация по MySQL 5.7 [Электронный ресурс]. – Режим доступа: <https://dev.mysql.com/doc/refman/5.7/en/> (дата обращения: 10.04.2025).

Документация по PHP [Электронный ресурс]. – Режим доступа: <https://www.php.net/manual> (дата обращения: 10.04.2025).

ПРИЛОЖЕНИЕ Б

Диаграмма последовательности



ПРИЛОЖЕНИЕ В

Диаграмма прецедентов

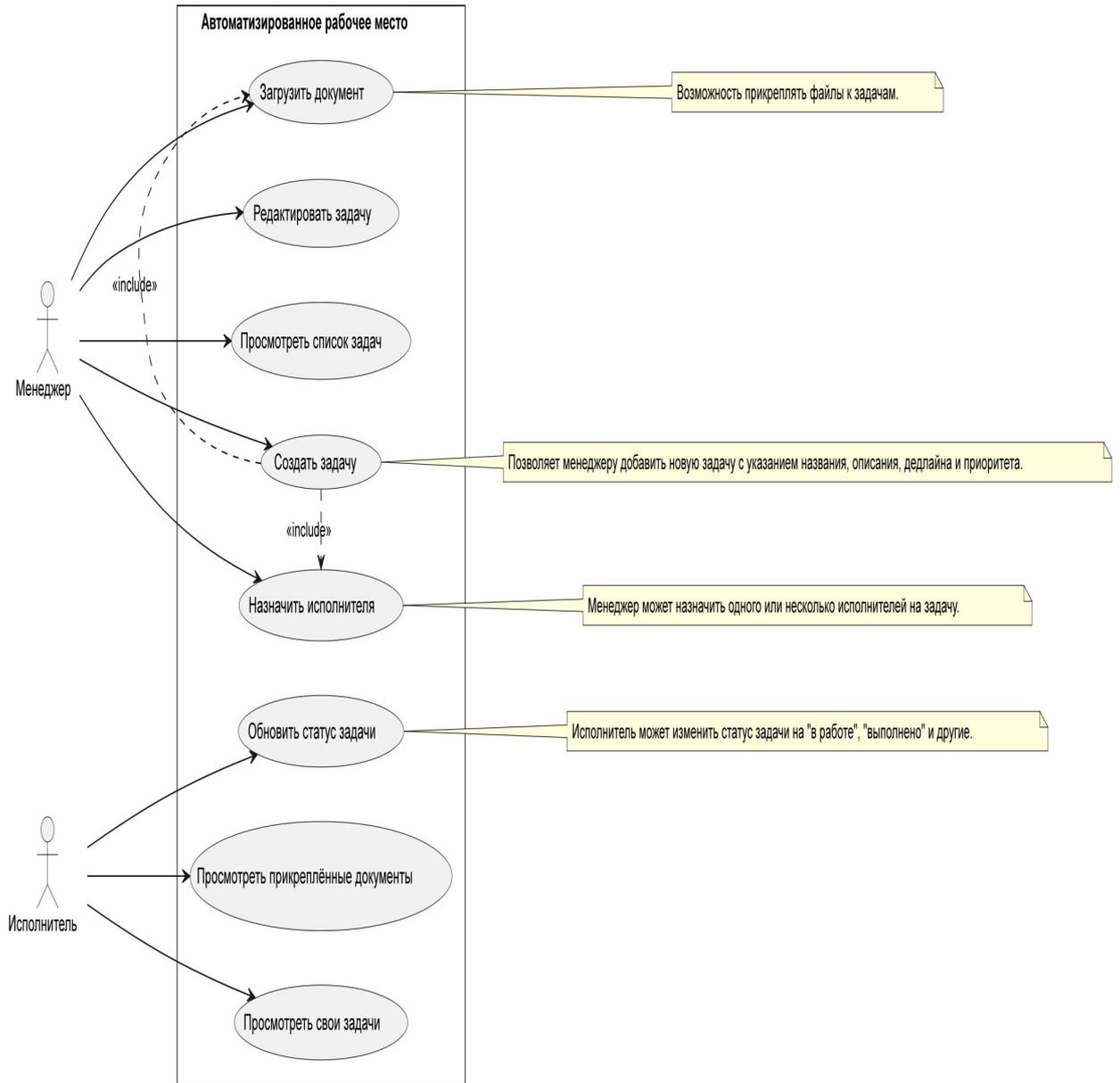


Диаграмма прецедентов автоматизированного рабочего места менеджера проектов

ПРИЛОЖЕНИЕ Г

Диаграмма классов

