

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

Кафедра «Прикладная математика и информатика»
(наименование)

09.03.03 «Прикладная информатика»

(код и наименование направления подготовки / специальности)

Бизнес-информатика

(направленность (профиль) / специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему: «Разработка информационной системы управления проектами в
организации»

Обучающийся

Г.К. Байзаева

(Инициалы Фамилия)

(личная подпись)

Руководитель

к.т.н., О.Ю. Копша

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

Тольятти 2024

Аннотация

Выпускная квалификационная работа состоит из 70 страниц печатного текста, 31 рисунок, 5 таблиц, 26 источников информации и 1 приложения.

В работе описываются все этапы разработки информационной системы управления проектами в акционерном обществе «Гидропроект».

Цель работы – разработать информационную систему, которая позволит управлять проектами в организации.

Предметом исследования является процесс автоматизации управления проектами в группе управления проектами.

Объектом исследования является акционерное общество «Гидропроект» работающего в отрасли гидроэнергетики и занимающегося проектно-изыскательской работой.

Работа разделена на три главы каждая из которых направлена на изучение и представление результатов по главе.

В первой главе работы подробно описывается деятельность акционерного общества «Гидропроект», представляется структура управления в организации и выделяется отдел, который занимается процессом разработки проектов. На основе полученных данных производится моделирование процесса в нотации IDEF0, где показан процесс до автоматизации и после нее. При построении моделей выделяются места, которые необходимо улучшить за счет автоматизации и составляются требования к информационной системе управления проектами.

Во второй главе работы представлено описание как проводилось логическое проектирование информационной системы управления проектами, а также проектирование базы данных, где будет храниться информация о проектах и всем что с ними связано при реализации. Разработка информационной системы будет реализована на выбранной архитектуре.

Третья глава описывает физическую разработку информационной системы управления проектами, которая включает представление основных

форм информационной системы, алгоритма работы модулей информационной системы и действия сотрудников отдела, где осуществляется процесс по управлению проектами. Для разработки информационной системы были выбраны язык программирования C#, среда разработки Visual Studio 2022 и система управления базой данных SQL Server. Заключительным этапом в третьей главе было проведено тестирование работы системы, на проверку работы системы без ошибок и выполнение всех заложенных функций. Был выбран метод тестирования информационной системы, который и был применен для выявления корректности выполнения всех функций и ввод данных в систему конечным пользователем.

Заключение показывает какие задачи были получены при выполнении выпускной квалификационной работы.

Оглавление

Введение.....	5
Глава 1 Функциональное моделирование деятельности АО «Гидропроект» ...	7
1.1 Обзор деятельности АО «Гидропроект».....	7
1.2 Концептуальное проектирование процесса управления проектами.....	11
1.3 Анализ существующих разработок	17
1.4 Постановка задачи на разработку системы управления проектами	19
Глава 2 Логическое проектирование автоматизированной информационной системы управления проектами.....	22
2.1 Логическая модель информационной системы управления проектами.	22
2.2 Проектирование базы данных системы управления проектами	29
2.3 Выбор архитектуры информационной системы управления проектами	32
Глава 3 Физическое проектирование информационной системы управления проектами.....	35
3.1 Выбор инструментов разработки	35
3.2 Описание программных модулей системы управления проектами.....	36
3.3 Разработка информационной системы управления проектами.....	37
3.4 Тестирование информационной системы управления проектами.....	49
Заключение.....	52
Список используемой литературы и используемых источников.....	53
Приложение А Листинг программы.....	56

Введение

Выпускная квалификационная работа выполняется на тему «Разработка информационной системы управления проектами в АО «Гидропроект»».

Компания работает в отрасли гидроэнергетики и выполняет функции по разработке предпроектной и проектной документации по инвестиционным проектам.

Уверенность в том, что проекты будут выполнены в срок, без задержек добавляет процесс управления проектами, который возлагается на руководителя проектами. Руководитель проектов должен понимать на каком проекте происходит замедление и выяснить причины для их устранения.

Все данные о проектах, задачах и поручениях по проекту должны храниться в хранилище данных, что обеспечит руководителю проекта быстро получать информацию о том, кто задействован на проекте и какие поручения выполняет на данный момент.

Информационная система управления проектами автоматизирует процесс отслеживания процесса и поможет руководителю вести несколько проектов одновременно без потери информации о текущем исполнении каждого проекта и каждого поручения в нем.

Цель работы – разработать информационную систему, которая позволит управлять проектами в организации.

Предметом исследования является процесс автоматизации управления проектами в группе управления проектами.

Объектом исследования является акционерное общество «Гидропроект» работающего в отрасли гидроэнергетики и занимающегося проектно-изыскательской работой.

Достижение цели выпускной квалификационной работы позволит выполнение задач, таких как:

- изучить деятельность акционерного общества «Гидропроект»;
- провести моделирование бизнес-процесса управления проектами;

- провести проектирование системы и базы данных;
- реализовать информационную систему управления проектами;
- провести тестирование информационной системы.

Работа разделена на три главы каждая из которых направлена на изучение и представление результатов по главе.

В первой главе работы подробно описывается деятельность акционерного общества «Гидропроект», представляется структура управления в организации и выделяется отдел, который занимается процессом разработки проектов. На основе полученных данных производится моделирование процесса в нотации IDEF0, где показан процесс до автоматизации и после нее. При построении моделей выделяются места, которые необходимо улучшить за счет автоматизации и составляются требования к информационной системе управления проектами.

Во второй главе работы представлено описание как проводилось логическое проектирование информационной системы управления проектами, а также проектирование базы данных, где будет храниться информация о проектах и всем что с ними связано при реализации. Разработка информационной системы будет реализована на выбранной архитектуре.

Третья глава описывает физическую разработку информационной системы управления проектами, которая включает представление основных форм информационной системы, алгоритма работы модулей информационной системы и действия сотрудников отдела, где осуществляется процесс по управлению проектами. Перед реализацией системы были выбраны язык программирования, среда разработки и система управления базой данных. Заключительным этапом в третьей главе было проведение тестирования работы системы, на проверку работы системы без ошибок и выполнение всех заложенных функций.

Заключение показывает какие задачи были достигнуты при выполнении выпускной квалификационной работы.

Глава 1 Функциональное моделирование деятельности АО «Гидропроект»

1.1 Обзор деятельности АО «Гидропроект»

Полное и сокращенное наименование предприятия.

На государственном языке: «GIDROPROEKT».

На русском языке: Акционерное общество «ГИДРОПРОЕКТ»

На английском языке: Joint-Stock Company «GIDROPROEKT»

Юридический адрес: город Ташкент, Яккасарайский район, улица Бабура, дом 20 [1].

История образования АО «Гидропроект» берет свое начало с 22 декабря 1930 года и является уникальным по своему профилю предприятием в Центральной Азии, с 94-летним опытом проектирования гидроэлектростанций. По проектам АО «Гидропроект» построено 104 гидроэлектростанции общей мощностью 10,488 млн. кВт, которые ежегодно вырабатывают свыше 38,653 млрд. кВт-ч электроэнергии.

«АО «Гидропроект» входит в перечень ведущих проектно-изыскательских организаций Республики Узбекистан, утвержденный Постановлением Президента № ПП-2946 02 мая 2017 года, и выполняет комплексное проектирование гидроэнергетических и гидротехнических объектов с объектами производственного, жилищно-гражданского назначения и социальной инфраструктуры» [21].

Кроме того, указом Президента № УП-5044 от 18 мая 2017 года «Об образовании акционерного общества АО «Узбекгидроэнерго» на АО «Гидропроект» возложены функции генеральной проектной организации по разработке предпроектной и проектной документации по инвестиционным проектам.

Институт выполняет весь комплекс проектных, изыскательских и исследовательских работ на всех стадиях проектирования, строительства и

эксплуатации гидроэнергетических и гидротехнических объектов; рассматривает влияние объектов на окружающую среду, выполняет водохозяйственные и энергетические расчеты, оценку экономической эффективности и финансовой окупаемости проектов. Осуществляет подготовку технической части тендерной документации на поставку технологического оборудования и рассмотрение технических предложений участников торгов.

Проектные работы ведутся для объектов нового строительства:

- реконструкции и модернизации существующих больших, средних и малых гидроэлектростанций;
- комплексных гидроузлов водохозяйственного и энергетического назначения.

Для обоснования проектных работ выполняются комплексные инженерные изыскания (топографические, гидрологические, гидрогеологические, геологические и геофизические).

Разрабатываются схемы использования водотоков.

Осуществляется постоянный надзор за состоянием, безопасностью и надежностью гидротехнических сооружений [5].

Оказываются услуги по техническому содействию заказчикам в решении вопросов:

- подготовки тендерной документации;
- экспертизе технических предложений участников тендеров;
- экспертизе проектов на поставку оборудования в части его соответствия техническим требованиям;
- участие в переговорах заказчика и поставщика по вопросам поставки оборудования.

Численность работников АО «Гидропроект» составляет 390 человек.

Виды работ (услуг):

- проектно-сметная документация;

– этапы проектирования.

Для обоснования проекта проводятся проектно-изыскательские, опытно-лабораторные и полевые работы.

Основным сырьем для производства являются - бумага, горюче-смазочные материалы (бензин, дизельное топливо), компьютеры и буровые детали.

Экспортируемые услуги такие как проектно-изыскательские работы для Китая и Российской Федерации.

Целью проектирования АО «Гидропроект» является комплексное использование гидроэнергетического потенциала Средней Азии для выработки мощности и создания маневренного источника для покрытия пиковых нагрузок единой электроэнергетической системы Республик Средней Азии, выработка дешевой и экологически чистой электроэнергии в соответствии с потребностью национальной энергосистемы, создание новых рабочих мест, увеличение доли гидроэнергетики в структуре энергетического баланса страны и экономия топливных ресурсов за счет сокращения выбросов в атмосферу при сжигании топлива на теплоэлектростанциях (ТЭС)

Задачей проектирования является увеличение доли гидроэнергетики в общем объеме выработки электроэнергии, увеличение мощности и выработки электростанций, входящих в систему АО «Узбекгидроэнерго», создание современной высокотехнологичной ГЭС, разработка оптимального конструктивного и технологического решения по комплексному и эффективному освоению водотока с внедрением современных и всесторонне обоснованных научно-технических решений в области проектирования, инженерных изысканий и строительства, выбором передовых технологий и оборудования, отвечающих высоким международным стандартам и требованиям, новейшим ресурсосберегающим технологиям. Обеспечение максимальной эффективности и окупаемости проекта, надежности и безопасности работы сооружений, ввода мощностей в установленные Постановлениями Президента сроки.

Результатом реализации разработанных проектов и их преимуществами по сравнению с альтернативными вариантами являются:

- комплексное использование гидроэнергетических ресурсов Средней Азии, возобновляемого и экологически чистого энергоисточника;
- ввод новых мини-, малых, средних и больших гидроэлектростанций;
- высокая маневренность гидроэлектростанции, позволяющая работать в пиковом режиме покрытия графика нагрузок энергосистемы;
- развитие инфраструктуры района строительства ГЭС.

Согласно уставу АО «Гидропроект», органами управления являются общее собрание акционеров, наблюдательный совет и генеральный директор.

Наблюдательный совет состоит из пяти человек. Структура управления АО «Гидропроект» показана на рисунке 1.

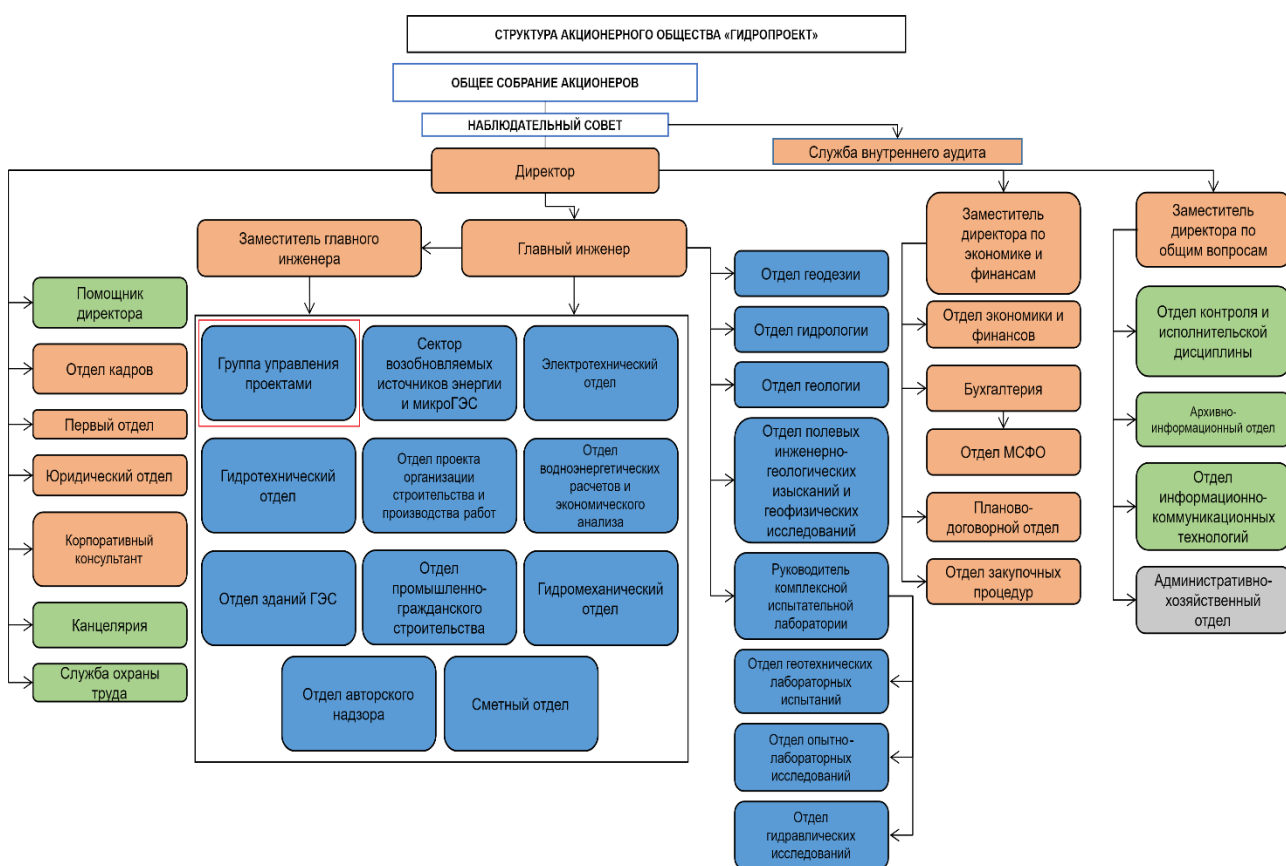


Рисунок 1 – Структура управления АО «Гидропроект»

На рисунке 1 показан структура управления компанией, где присутствуют различные отделы и структуры. Рассмотрим подробнее отдел управления проектами и основные процессы.

Отдел занимается проектной деятельностью, где осуществляется ведение проектно-сметной документации и реализация этапов проектирования.

Отдел работает под управлением начальника, который курирует работу всего отдела и ведет переговоры по вновь поступившим проектам в компании. Также начальник отдела может курировать проекты, которые ведет руководитель проекта и вести их управление.

Для того чтобы информация была актуальной необходимо обновлять ее и вести диалог с сотрудниками которые были назначены на проект и отдельные задачи. Чтобы проект исполнялся быстрее, руководитель проекта может назначить на одну задачу несколько сотрудников и отслеживать процент исполнения каждой задачи на проекте.

1.2 Концептуальное проектирование процесса управления проектами

Рассмотрев подробно процесс управления проектами в организации можно смоделировать процесс в одной из представленных нотаций.

Процесс управления проектами начинается с поступления проекта в организацию, который затем отдается группе управления проектами, где происходит поэтапная реализация проекта [1].

Моделирование процесса будет проведено в программе AllFusion ERwin Data Modeler 7.0.

Программный продукт обладает всеми необходимыми инструментами, которые необходимы для моделирования бизнес-процессов, также программа имеет возможность построения разных видов диаграмм таких как IDEF0, DFD, IDEF3.

Проведем моделирование процесса управления проектами в организации методом декомпозиции процесса на уровень ниже. Изначально будет построена диаграмма процесса первого уровня [5]. Процесс будет показан с абстрактной стороны, где указаны входящие и исходящие данные при выполнении процесса управления проектами. Также диаграмма будет иметь механизмы, которые управляют процессом или как-то в нем задействованы. После выполнения процесса будет получена исходящая информация или данные, которые будут получены по результату.

Диаграмма показана на рисунке 2.

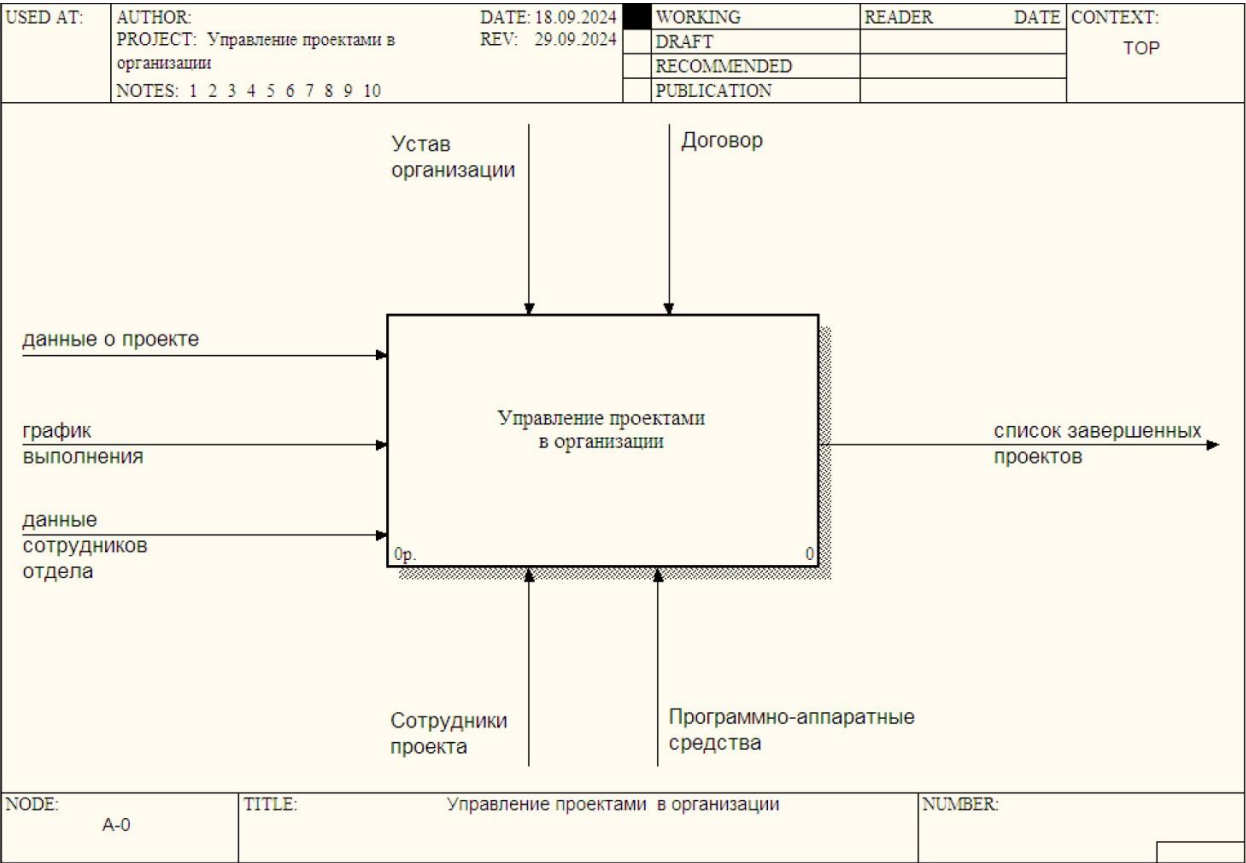


Рисунок 2 – Диаграмма процесса управления проектами в организации

Процесс управления проектами в организации происходит на основе устава организации и договора с заказчиком. Договор является управлением в процессе управления проектами, который регламентирует сроки выполнения

проекта. Условия, которые прописаны в договоре должны выполняться при реализации проекта [6].

При декомпозиции процесса будет подробнее показано как процесс протекает и какие ресурсы задействованы на разных этапах реализации проекта.

Декомпозированная диаграмма показана на рисунке 3.

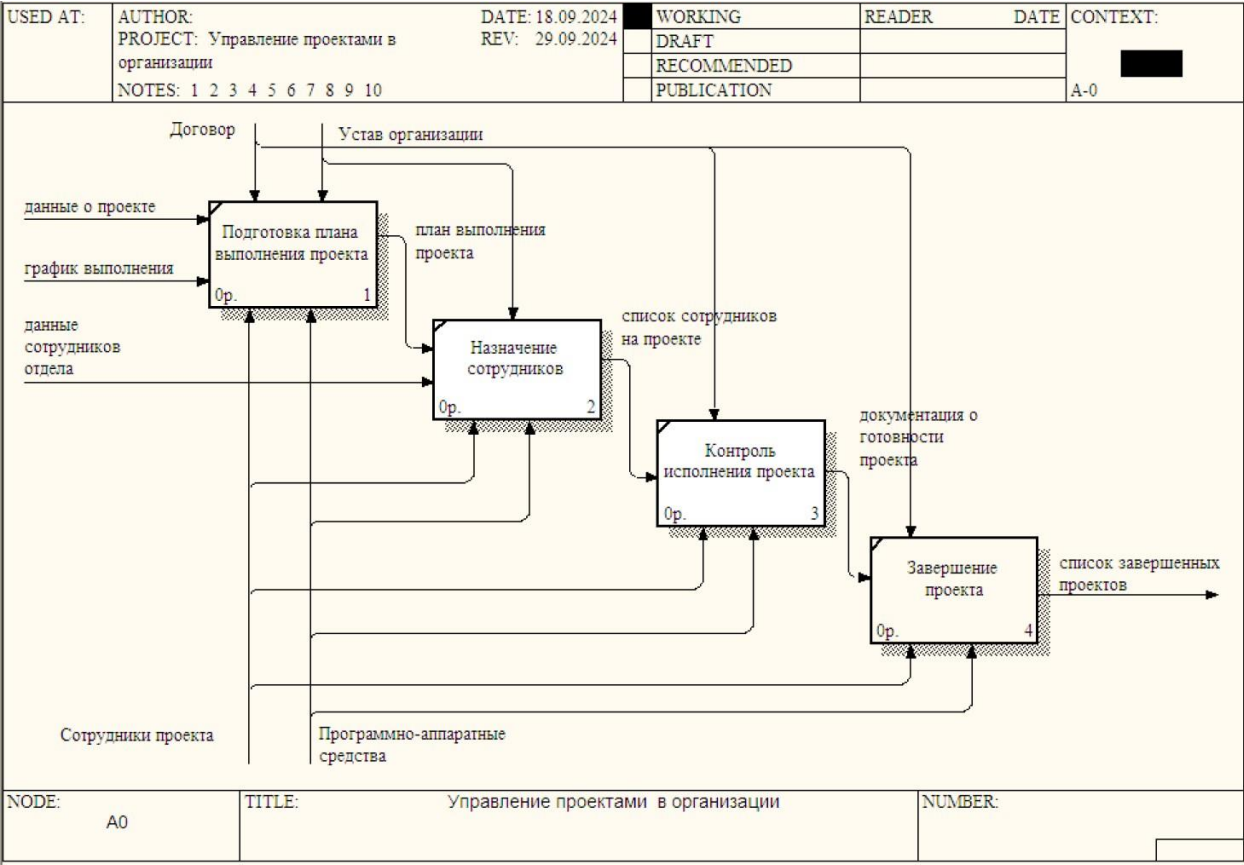


Рисунок 3 – Декомпозиция диаграммы первого уровня управления проектами в организации

Процесс управления проектами ведется с помощью программно-аппаратных средств, где указываются сотрудники, которые входят в группу выполнения проекта под управлением руководителя. Такой способ устарел и не дает прозрачности выполнения проекта на каждом из этапов. Управление проектами затрудняется и не дает возможности руководителю проекта предпринять меры если проект отстаёт от запланированного графика или

имеет перерасход бюджета. При реализации проекта также могут возникнуть риски, которые требуют внимания руководителя проекта и применения мер по их устранению.

Внедрение автоматизированной информационной системы позволит быстро и эффективно отслеживать ход работы проектов одному руководителю. Также система позволит один проект для быстрого выполнения разбивать на несколько задач или поручений, на которых будут назначены определенные сотрудники отдела.

Руководитель проекта сможет ежедневно отслеживать выполнение каждой задачи на проекте и процент выполнения целого проекта. При задержке какой-либо задачи или поручения на проекте руководитель сможет быстро отреагировать и предпринять меры для выравнивания сроков выполнения проектов.

Диаграмма, представленная на рисунке 4 смоделирована с учетом разработанной информационной системы и внедренной в процесс управления проектами в организации.

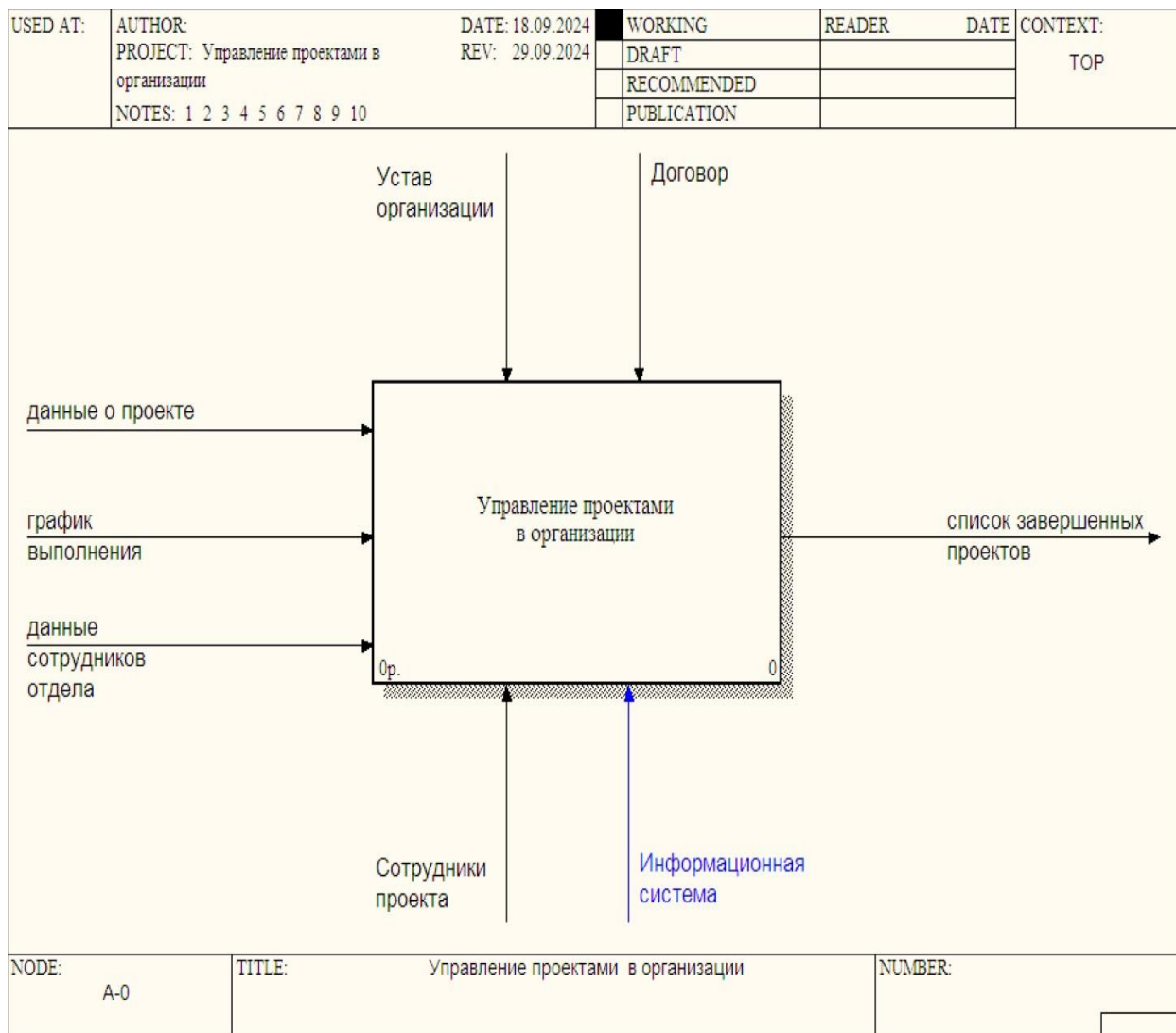


Рисунок 4 – Диаграмма первого уровня процесса управления проектами

Внедрение автоматизированной информационной системы позволит руководителю проектов вести все процессы и учет выполнения проектов гораздо быстрее и эффективнее. Также сократится количество ошибок допущенных при распределении сотрудников на проектах.

Покажем диаграмму декомпозиции первого уровня после автоматизации, рисунок 5.

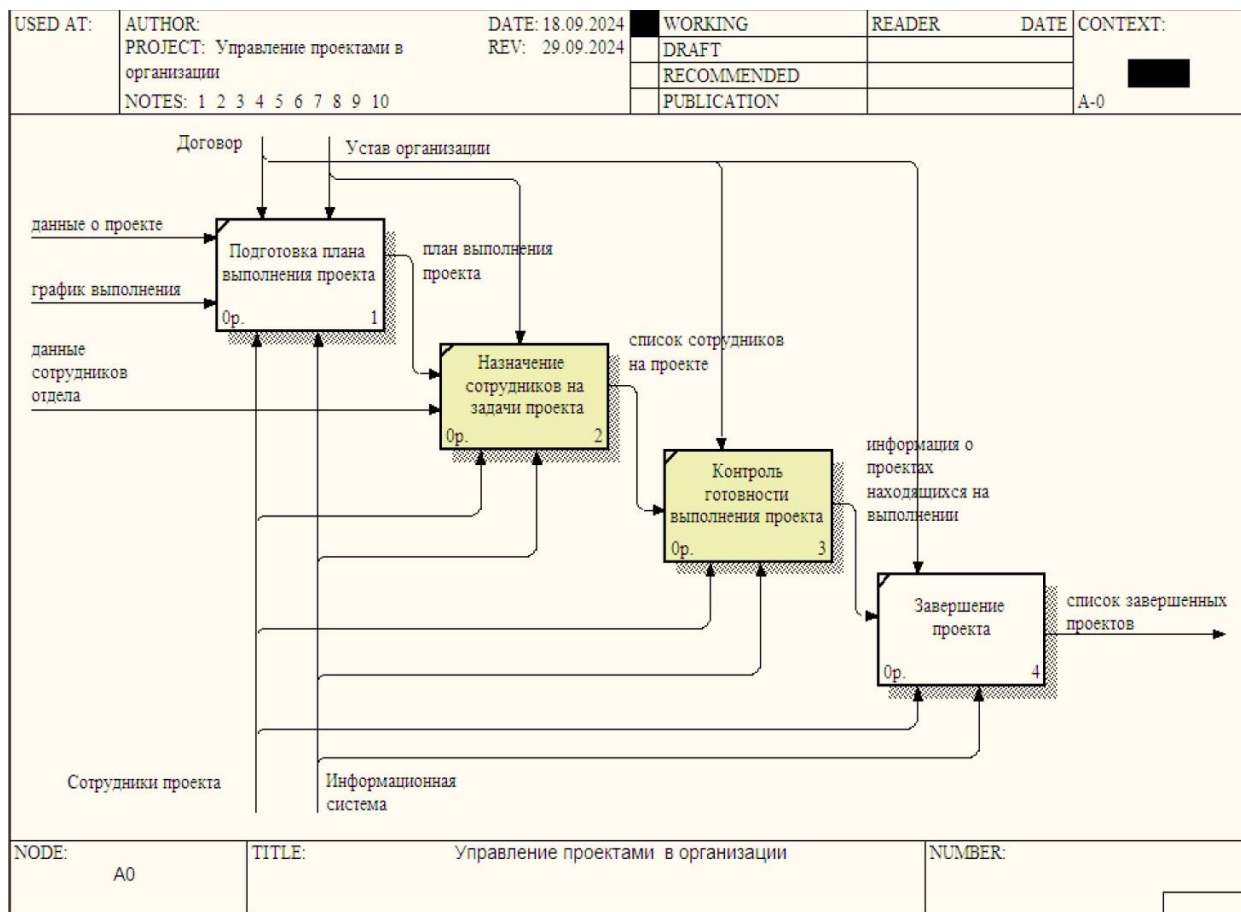


Рисунок 5 – Диаграмма декомпозиции управления проектами с информационной системой

Внедрение автоматизированной информационной системы изменит выполнение некоторых подпроцессов, которые были показаны на рисунке 3.

Блоки «Назначение сотрудников» и «Контроль исполнения проекта» после внедрения информационной системы будут изменены. Назначение сотрудников теперь будет проводиться непосредственно на задачи, на которые проект будет разделен. Контроль исполнения проекта будет осуществляться на основе данных о выполнении каждой задачи в процентном выражении и суммарное выполнение проекта по итогу рабочего дня.

Обновление этих процессов обеспечит прозрачность выполнения всего проекта, позволит быстро реагировать на отстающие задачи и назначать новых сотрудников при необходимости [7].

Сотрудник получит возможность вести отчет за свою деятельность на задачах по итогу их выполнения, даже если задача будет находиться на этапе исполнения.

Таким образом процесс управления проектами будет автоматизирован и повысит качество и скорость выполнения проектов под руководством одного руководителя.

1.3 Анализ существующих разработок

Информационная система позволит руководителю проекта эффективно управлять проектами на всем этапе выполнения и контролировать сотрудников, которые назначены на задачи проекта.

Для сравнения рассмотрим несколько аналогичных программ которые представлены русскими разработчиками на рынке.

«Мегаплан поможет наглядно контролировать прогресс проекта, собрать все данные проекта в одном месте, распределять задачи по ответственным, упростить работу с повторяющимися задачами, а также планировать и контролировать сроки.

Функции и преимущества:

- данные по проекту хранятся в единой карточке;
- наглядное планирование;
- общее пространство для работы;
- планирование сроков и трудозатрат;
- выполнение задачи по шагам;
- автоматизация однотипных задач;
- напоминания;
- проведение крупных собраний.

Недостатки программы:

- функции программы ограничены;
- отсутствие многих важных функций;

– ограниченный доступ» [11].

«Bitrix24 — целый комплекс инструментов для организации удаленной работы. Здесь есть буквально все: от канбан-доски до конструктора сайтов, плюс много инструментов для автоматизации рабочих процессов.

Функциональность программы:

- видеозвонки до 100 человек;
- мессенджер;
- учет рабочего времени и отчеты;
- управление проектами в разных режимах;
- роли и права доступа;
- шаблоны задач и работы» [1].

«Яндекс Трекер — система для организации работы в компании. Трекер используется в самом Яндексе «для всего — от разработки сервисов до закупки печенья на кухне».

Функции:

- управления проектами по Agile — с оценкой трудозатрат, спринтами, учетом времени, диаграммами;
- итеративная работа для дизайн-команд;
- ведения маркетинговых кампаний;
- ускорения документооборота, организации HR, работ по техподдержке, закупок и прочих процессов» [19].

«Strive — простой и удобный сервис без лишних опций. После регистрации можно сразу приступить к работе и не тратить много времени на обучение.

Функции:

- ведение канбан-досок;
- ведение регламентов;
- документация и вкладки;
- получение уведомлений;
- ведение задач проекта» [26].

Проведем сравнение программ в таблице 1.

Таблица 1 – Сравнительная таблица программ

Критерий	Мегаплан	Bitrix24	Яндекс Трекер	Strive
Работа в команде	+	+	+	+
Интеграция с мессенджерами	-	+	-	-
Ведение маркетинговых кампаний	-	-	+	+
Ведение задач проекта	+	+	+	+
Разграничение прав доступа	-	+	-	+
Получение уведомлений	+	-	-	+

Программы имеют похожие функции и достоинства, которые помогают вести проекты компании и назначать ресурсы на задачи. Все программы также имеют сложности в настройке и интеграции с другими системами. Ни одна из представленных программ в сравнительной таблице не позволяет полноценной отслеживать проект по каждой задаче и подзадаче проекта.

1.4 Постановка задачи на разработку системы управления проектами

Автоматизированная информационная система создается для автоматизации процессов, которые связаны с управлением проектов [2]. Целью создания системы является создание более эффективных процессов на всем

жизненном пути проекта. Система обеспечит руководителю проекта актуальную информацию о текущем состоянии проекта и выполнения задач.

Основными задачами, которые будет решать система будут формирование задач, подзадач и распределение их на сотрудников отдела, в котором осуществляется реализация проектов. Сотрудники отдела будут освобождены от рутины бумажного формирования отчетности о выполнении задач на проекте [3].

Система обеспечит беспрепятственную работу сотрудникам, назначенным на один проект, работать в команде и суммарно выполнять поставленные подзадачи. Система повышает эффективность работы команды путем оптимизации процессов управления задачами, улучшения взаимодействия между участниками, автоматизации повседневных задач [4].

В системе будут обрабатывать входящие данные, вводимые руководителем проекта.

При выполнении проекта на входе будут введены данные:

- данные о проекте, задачах, подзадачах;
- данные начала и окончания проекта;
- длительность каждой задачи.

После обработки входящих данных, будут получены:

- информация о стадии выполнения задачи, подзадачи, проекта.
- список выполненных проектов.
- информация о сотрудниках, которые назначены на задачах.

Информационная система позволит:

- вводить новые данные о проектах, задачах, сотрудниках;
- вести проект от начала и до завершения;
- вести контроль исполнения проекта;
- увеличить прозрачность занятости сотрудников на проекте.

Информационная система будет иметь набор функций:

- регистрация новых проектов;

- распределение сотрудников на задачах;
- ввод процента выполнения задачи;
- просмотр списка проектов и процент их выполнения.

Меню автоматизированной информационной системы должно быть понятным для сотрудников и руководителя проекта. Вход в систему должен осуществляться после прохождения аутентификации данных, введенных пользователем.

Система должна иметь разграничение на вход руководителя проекта или начальника отдела и сотрудника, который будет непосредственным исполнителем поручений и задач на проекте.

Меню автоматизированной информационной системы должно быть понятным при первом входе в систему и иметь навигацию с надписями. Личный кабинет сотрудника должен содержать всю необходимую информацию и иметь все инструменты для выполнения поставленных задач.

Вывод по первой главе.

Первая глава была направлена на исследование деятельности акционерного общества «Гидропроект» и подробного изучения процесса управления проектами.

Исследования показали, что в группе управления проектами реализуются все проекты, которые поступают в организацию. После проведения моделирования процесса управления проектами и более глубокого изучения были выделены недостатки в процессе которые замедляют процесс и не гарантируют прозрачности управления проектом [8].

На основе выделенной проблемы и предложений по автоматизации процесса управления проектами были рассмотрены аналогичные программы, которые присутствуют на рынке. Исходя из того, что программы являются зарубежными и в силу дороговизны и отсутствия дальнейшего обслуживания программ было принято решение о собственной разработке информационной системы.

Глава 2 Логическое проектирование автоматизированной информационной системы управления проектами

2.1 Логическая модель информационной системы управления проектами

В информационной системе предусмотрено разграничение прав доступа пользователей на выполнение различных операций над данными и ограничения доступа пользователей к задачам системы. Для этого в системе существуют понятия рабочего места и вида работы. Где рабочие места служат для объединения различных задач системы в группы, а в рамках каждого рабочего места для конкретного пользователя определяются доступные для него виды функций (добавление задач и назначение их сотрудникам).

Обслуживанием информационной системы будет заниматься администратор. В его задачи будет входить контроль работоспособности системы и реагирование на сбои. Администратор будет иметь возможность регистрировать сотрудников в системе присваивая им личные данные для входа в личный кабинет.

Ведение процесса по управлению проектами в отделе будет полностью возложена на руководителя проекта, который может вести сразу несколько проектов одновременно. Мониторинг по выполнению проектов будет полностью автоматизирован, что позволит снизить нагрузку на руководителя проекта.

Диаграмма вариантов использования позволит наглядно увидеть взаимосвязь пользователя и системы управления проектами. Диаграмма показана на рисунке 6.

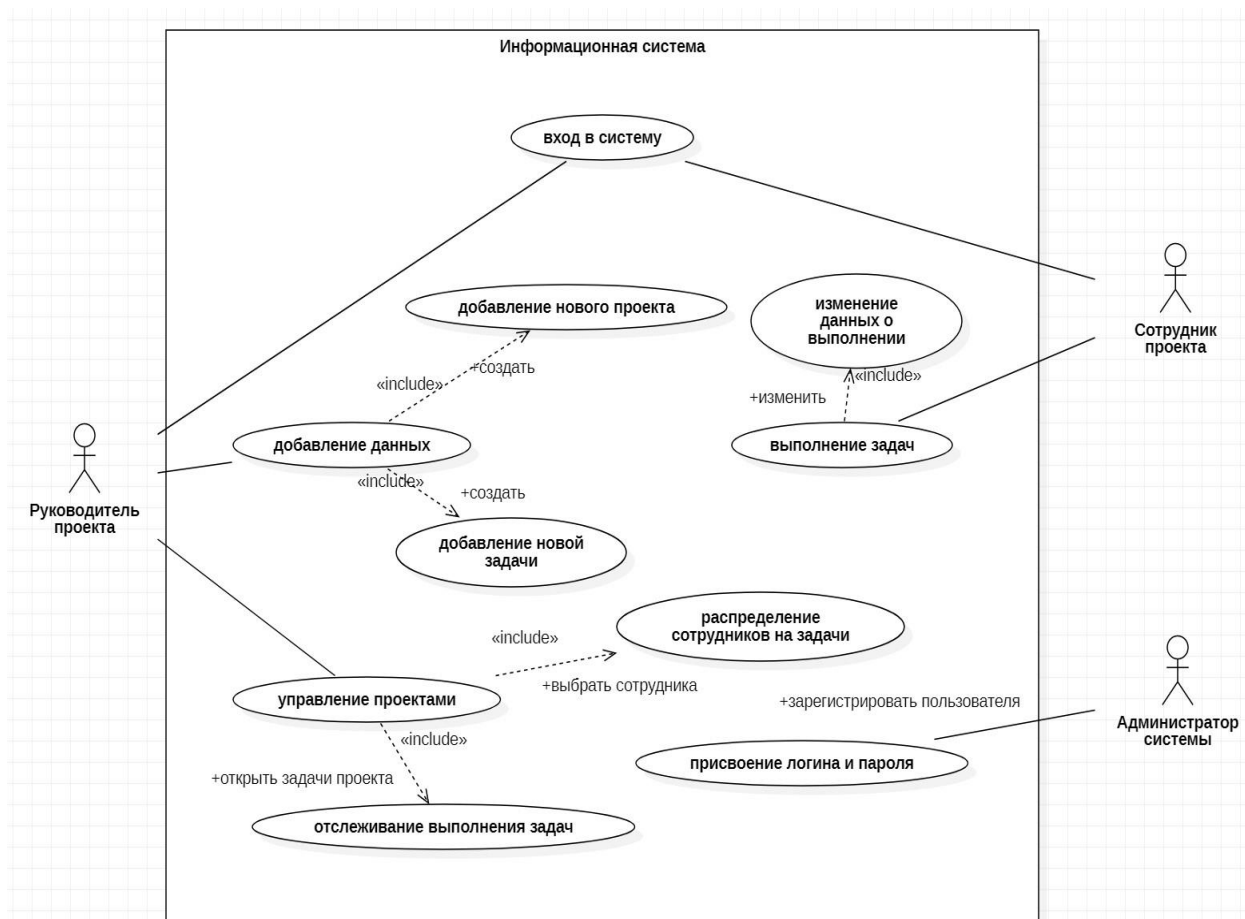


Рисунок 6 – Диаграмма вариантов использования системы управления проектами

Диаграмма вариантов использования демонстрирует трех акторов и функции, которые будут ими выполняться в системе. Руководитель проекта после входа в систему сможет добавлять новые данные о проекте и задаче, распределять сотрудников на задачи, вести управление проектами отслеживания выполнения отдельных задач.

Сотрудник проекта также после входа в систему под своими данными сможет выполнять задачи и изменять данные о выполнении.

Администратор помимо администрирования системы присваивает пользователям логин и пароль.

Вся работа с проектом начинается после получения нового проекта в отделе. Руководитель проекта добавляет проект в систему и разбивает его на несколько задач и подзадач если это необходимо. Каждый проект, задача и

подзадача(поручение) будут иметь свои сроки исполнения, которые должны соблюдаться сотрудниками при выполнении.

Руководитель проекта подбирает команду на проект путем назначения выбранных сотрудников из списка, представленного в системе. Все сотрудники по ходу выполнения проекта должны будут отмечать процент выполнения, процент может выставляться приблизительный, для того чтобы руководитель имел представление о ходе выполнения проекта [11].

Алгоритм управления проектами показан на рисунке 7.

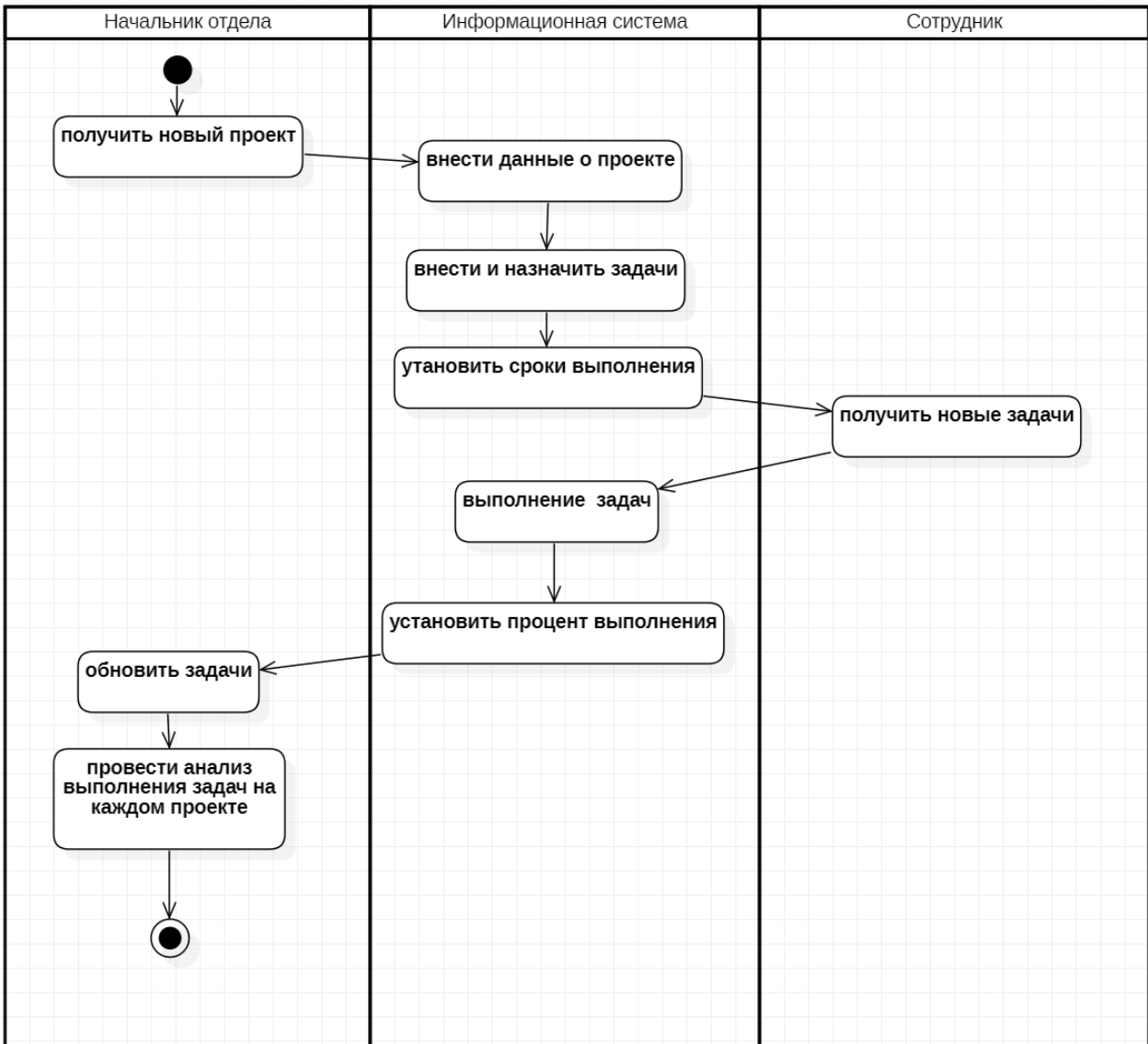


Рисунок 7 – Диаграмма деятельности пользователей при выполнении задач

Информация, которая будет обрабатываться на входе, все данные о проекте, который будет создан. Проект должен быть полностью создан и разбит на подзадачи, после чего можно будет формировать команду, которая займется выполнением задач на проекте. Вся информация о проекте и его выполнении будет сохраняться автоматически в базу данных и обновляться при запросе руководителя проекта.

Данные, с которыми будут работать сотрудники:

- сроки выполнения проекта и подзадач;
- информация на назначенных задачах.

Сотрудники отдела под руководством руководителя проекта будут выполнять задачи и отмечать готовность каждой задачи, которая будет добавлена в личный кабинет.

При формировании команды проекта руководитель должен будет выбрать сотрудников из списка, и назначить на задачу, согласно его квалификации. Задачи могут иметь разный уровень сложности и характер выполнения. Если задача представляет собой сложный алгоритм выполнения, руководитель может назначить на ее выполнение нескольких сотрудников и сократить срок выполнения [10].

Информация о задаче и подзадачах будет представлена в виде готовой формы с данными как показано на макете, рисунок 8.

Список задач, которые были созданы и назначены будут доступны для просмотра руководителю проекта и сотрудникам, назначенным на выполнение задач. Сотрудники будут видеть сроки необходимые для реализации задач, руководитель проекта будет видеть срок выполнения полной задачи с учетом подзадач.

Частичное выполнение задачи может также отмечаться ежедневно для того, чтобы руководитель проекта имел представление как идет проект и какие меры нужно предпринять на каких задачах, если проект начинает отставать от запланированного графика [16].

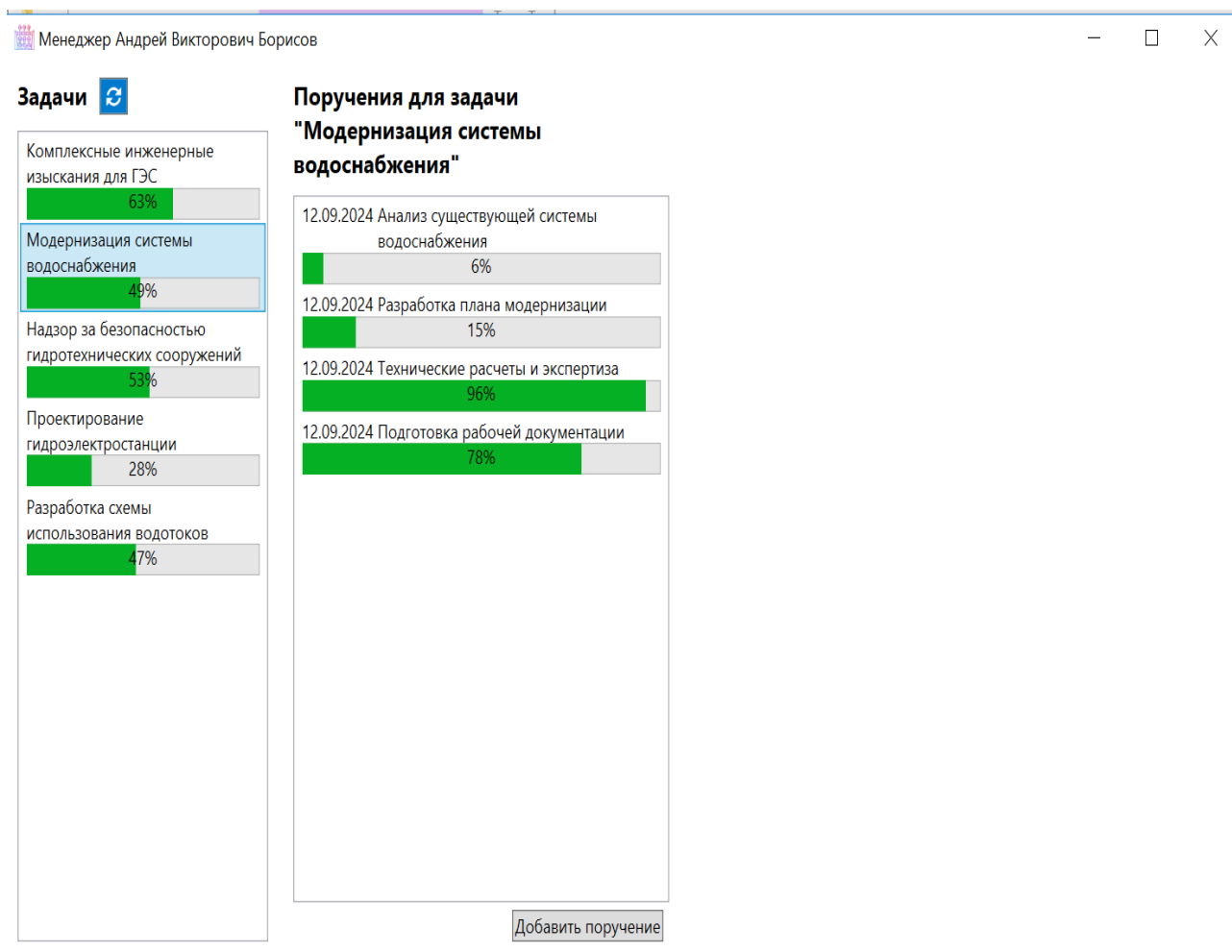


Рисунок 8 – Макет формы с данными проекта и задач по нему

На рисунке 9 показана логика информационной системы, содержащая классы обработки данных и методов системы.

Обработка информации и вывод значений мониторинга выполнения задач обрабатывается при помощи использования новых технологий и DLL, который представляют набор функций и процедур, которые вызваны программами для выполнения определенных задач [15].

В структуре содержатся такие модули как:

- TeamTrackWpfApplication.Services;
- TeamTrackWpfApplication.ViewModels;
- TeamTrackWpfApplication.Views.

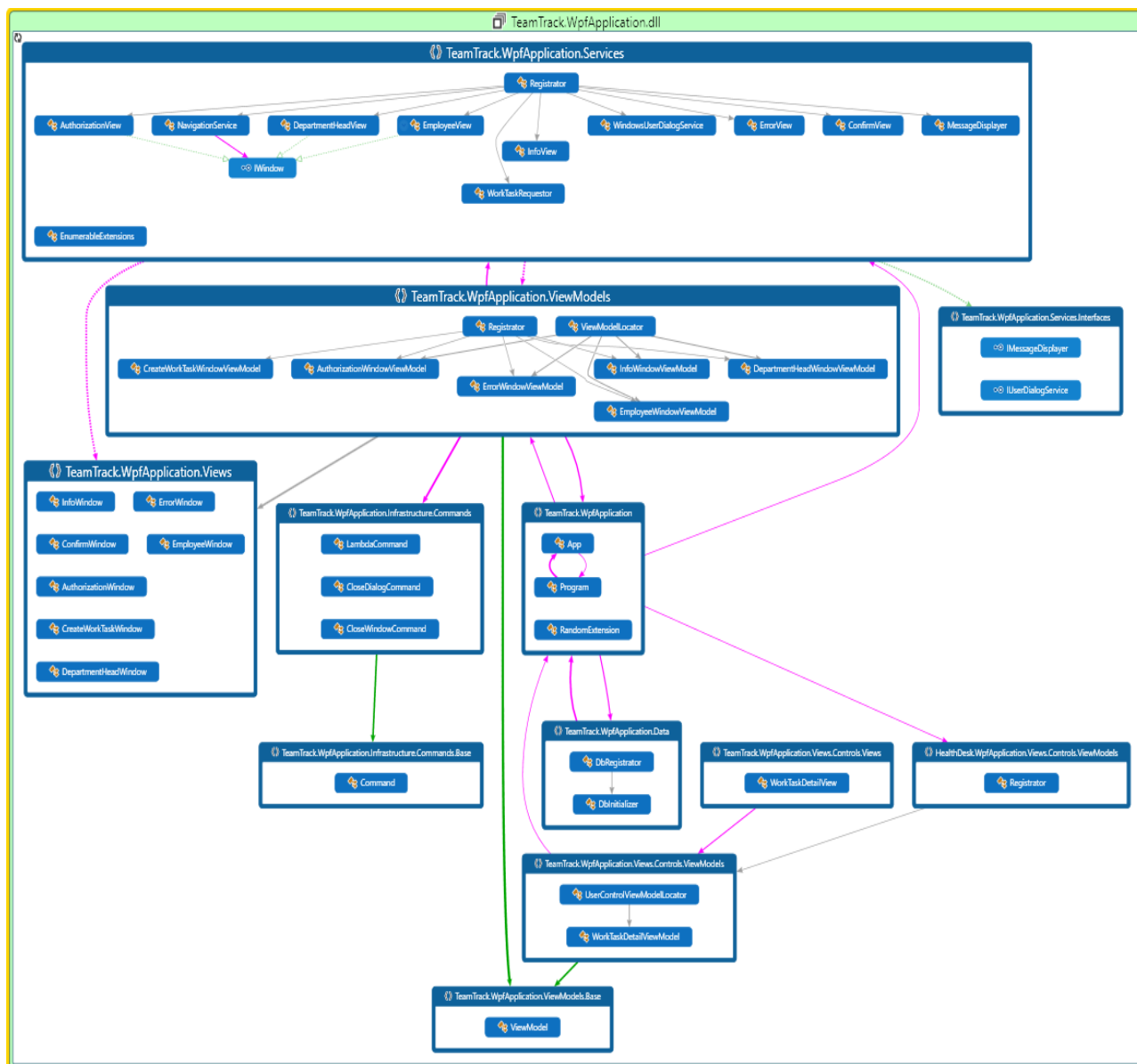


Рисунок 9 – Структура системы, сформированная в среде разработки Visual studio 2022

Каждый модуль отвечает за выполнение определенных функций в системе. `TeamTrackWpfApplication.Services` отвечает за бизнес-логику системы. `TeamTrackWpfApplication.ViewModels` представляет модуль обработки данных в системе. `TeamTrackWpfApplication.Views` отвечает за представление данных пользователю системы.

Структура файлов обработки и выполнения задач системы в расширенном виде показана на рисунке 10.

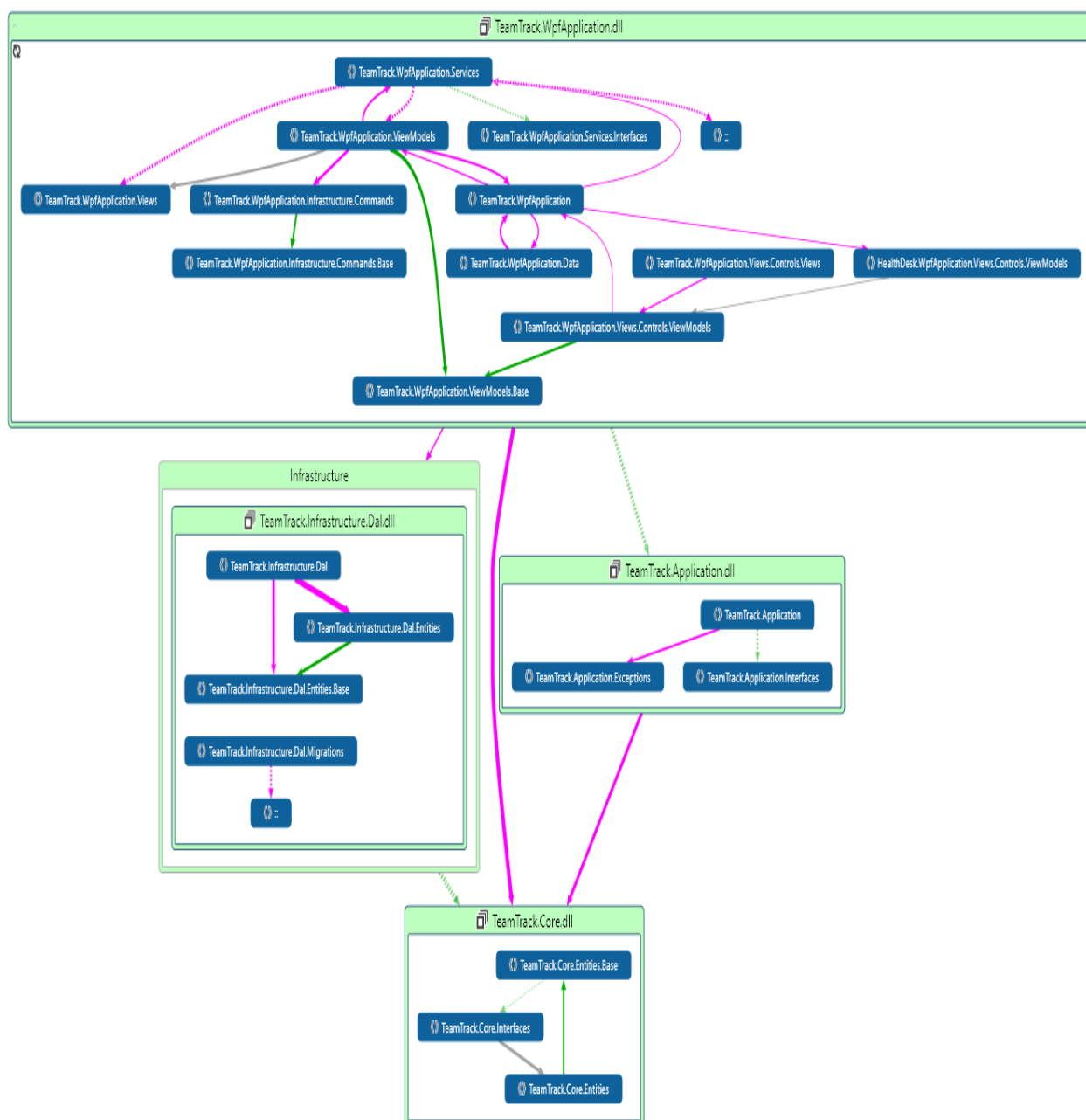


Рисунок 10 – Структура файлов DLL

DLL означает Dynamic-Link Library. Каждый вид DLL файла развит для выполнения некоторых задач, таких как представление инфраструктуры системы, файлы для запуска системы.

Методы и классы обеспечивают обработку всей поступающей информации при регистрации проекта и реализации его на всех этапах жизненного цикла [17].

2.2 Проектирование базы данных системы управления проектами

База данных, содержащая данные о сотрудниках, которые работают в организации и проектах, которые поступают в организацию строится по принципу индивидуальности.

Каждый проект будет вноситься в базу данных как новый объект, который будет содержать уникальный идентификатор проекта, который не будет повторяться, описание проекта, название и сроки его выполнения. Также проект будет иметь от одной до множества задач, которые также будут иметь уникальный идентификатор, так как задачи на разных проектах могут повторяться [9].

Защита информации в системе будет осуществляться путем разграничения прав доступа. Доступ к данным будут иметь сотрудники, которые будут принимать непосредственное участие в реализации проекта.

Информация будет храниться в таблицах, которые необходимо выделить для создания базы данных. Все данные должны храниться в реляционной базе данных.

База данных построена на данных представленных в таблицах 2 - 5. В таблицах указаны атрибуты и типы данных. Также показаны первичные ключи (PK), которые будут иметь связи с таблицами в таблицах отмечены как внешние ключи (FK).

Таблица 2 – Проекты

	Атрибут	Тип
PK	Идентификатор проекта	int
FK	Идентификатор отдела (внешний ключ)	int
	Срок выполнения	datetime
	Дата завершения	datetime
	Название	varchar (250)

Таблица 3 – Отделы

	Атрибут	Тип поля
PK	Идентификатор отдела	int
	Название	varchar(50)

Таблица 4 - Сотрудники

	Атрибут	Тип
PK	Идентификатор сотрудника	int
FK	Идентификатор отдела (внешний ключ)	int
	ФИО	varchar (50)
	Логин	varchar (30)
	Пароль	varchar (30)
	Роль пользователя	int

Таблица 5 –Задачи

	Атрибут	Тип
PK	Идентификатор задачи	int

Продолжение таблицы 5

	Атрибут	Тип
FK	Идентификатор сотрудника (внешний ключ)	Int
FK	Идентификатор проекта (внешний ключ)	Int
	Срок выполнения	datetime
	Описание	varchar (250)
	Дата завершения	datetime

Логическая структура базы данных представлена на рисунке 11.

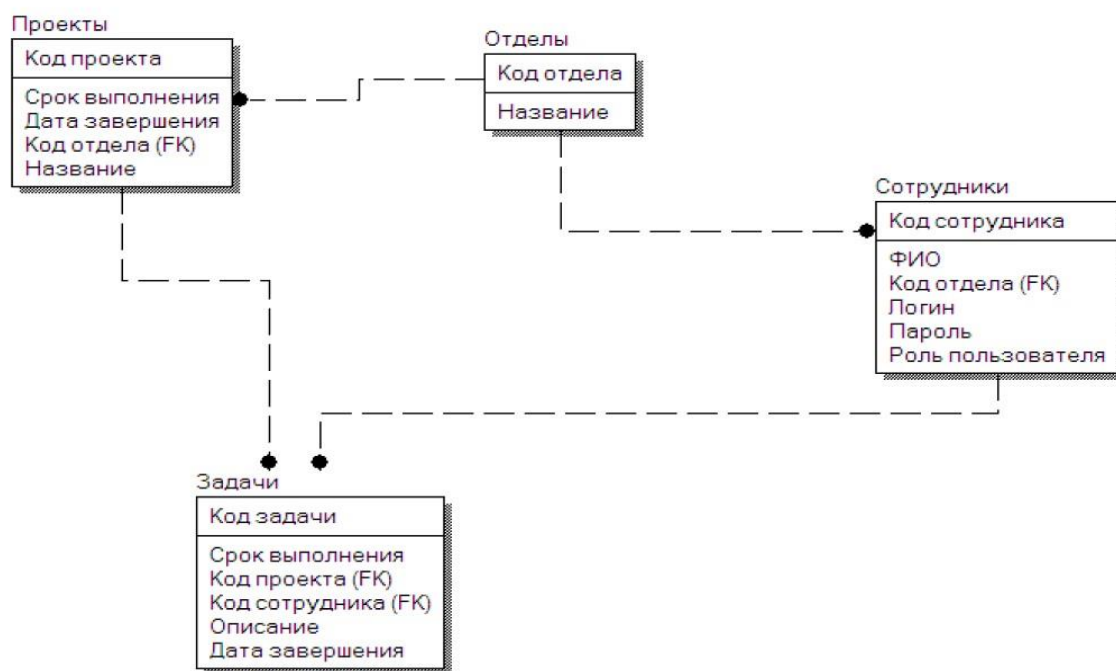


Рисунок 11 – Логическая структура базы данных

Схема базы данных показана на рисунке 12.

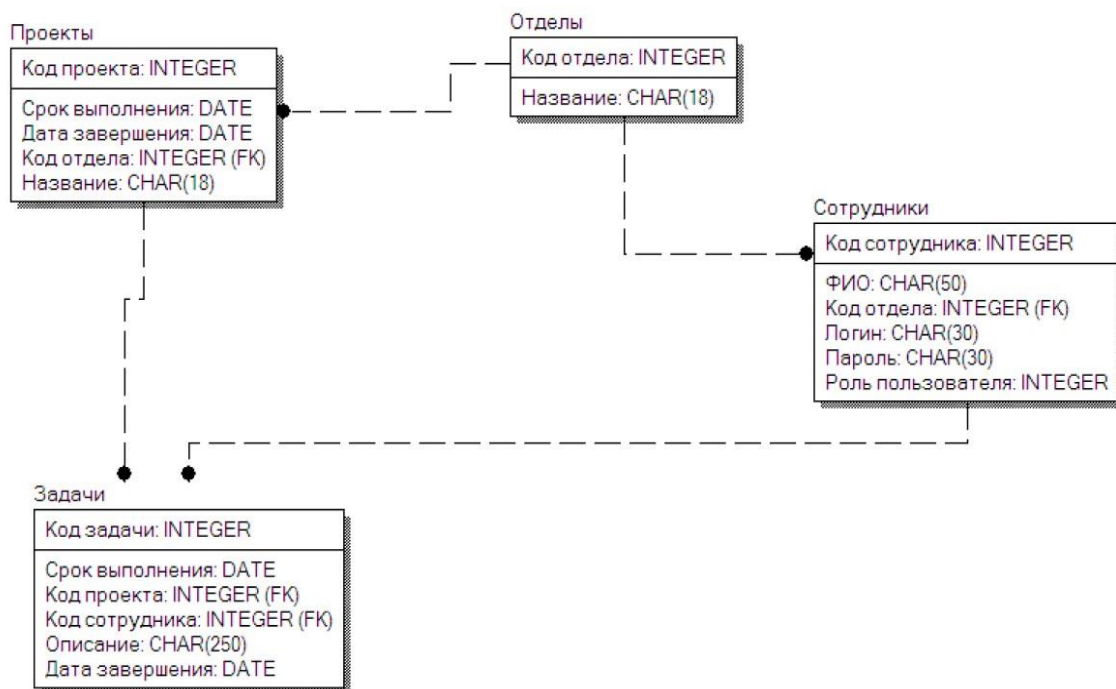


Рисунок 12 – Схема базы данных

Схема базы данных состоит из четырех таблиц, каждая из которых содержит информацию необходимую для процесса назначения и мониторинга выполнения задач проекта.

2.3 Выбор архитектуры информационной системы управления проектами

Система построена по трехзвенной клиент-серверной архитектуре. Трехзвенная архитектура построения информационной системы позволяет вести не только локальные базы данных в каждом учреждении, но и в дальнейшем без дополнительных доработок перейти к другому подразделению или филиалу организации [12].

Система должна быть стабильна в работе. Серверы должны быть связаны между собой сетью не ниже 1Gb. Все клиентские машины подключаются только к серверу приложений. Конфигурация клиентских машин на скорость работы в программе практически не влияет.

Дополнительно рекомендуется выделить в сети сервер хранения резервных копий баз данных около 5Тб дискового пространства.

Компьютеры должны быть укомплектованы мышью, клавиатурой, сетевыми шнурами [17].

Так как с системой будет работать специалист, то требуется монитор, который должен поддерживать вывод изображения с разрешением не менее 800х600. Рекомендовано 1024х768 точек и более [13].

Система выполняет действия над данными, хранящимися в базе данных, построенной по клиент-серверной технологии. Доступ к базе данных выполняется по локально-вычислительной сети.

Представление уровней представлено на рисунке 13.



Рисунок 13 – Распределение уровней системы

Клиент-серверная архитектура самая подходящая для реализации информационной системы.

Клиентский уровень не имеет прямых связей с базой данных, не нагружен основной бизнес-логикой и не хранит состояние приложения.

Выводы по второй главе.

Вторая глава содержит описание логического проектирования информационной системы управления проектами. Проектирование включает разработку диаграмм деятельности процесса, наглядно демонстрирующей взаимодействие пользователей с системой при управлении проектом.

Также было проведено проектирование базы данных, где были выделены основные объекты хранения информации о проектах, задачах и сотрудниках, которые будут выполнять задачи на проекте [18].

Для создания информационной системы были рассмотрены различные архитектуры и определена клиент-серверная архитектура, наиболее подходящая для работы группы управления проектами.

Глава 3 Физическое проектирование информационной системы управления проектами

3.1 Выбор инструментов разработки

Автоматизированная информационная система будет использоваться в отделе по управлению проектами, где выполняются основные проекты организации. Она добавит новый функционал в соответствующих формах, новое рабочее место и разграничение прав пользователей.

Информационная система должна быть объектно-ориентированной и гибкой при интеграции с другими системами. Для этого необходимо выбрать инструменты, которые обеспечат системе эти показатели.

Вся логика системы будет размещена на сервере компании. База данных будет располагаться на сервере базы данных.

Работа информационной системы будет осуществляться по средствам HTTP запросов, которые пользователь будет отправлять с рабочего места на сервер приложения. SQL запросы отправляются с сервера приложения на сервер базы данных.

При реализации информационной системы будут использоваться C#, MS SQL Server [14].

Файлы базы данных будут созданы при помощи Microsoft Visual C#.NET. Заполнение файлов базы данных производится при помощи прикладного программного обеспечения.

Система будет функционировать в среде Microsoft Windows. Для работы в системе необходимо установить локальную базу данных MS SQL Server. Система может запускаться из .exe файла, либо запускаться из исполняемого файла предварительно установив на ПК среду разработку Visual Studio 2022 версии.

Все необходимые компоненты системы должны быть установлены на одной машине. В этом случае система установлена и работает. Если в системе

предполагают работать несколько пользователей, тогда в сети организуется сервер, на который устанавливаются серверные компоненты – MS SQL Server, IIS + dotnet Framework и сама система.

Клиентский компонент - браузер – находится на каждом рабочем месте в сети, и с него происходит подключение к серверу [19].

Информационная система может быть установлена только, если на компьютере уже установлены вышеперечисленные компоненты. Далее запускается файл инсталлятора. Вводится имя SQL-сервера в соответствующее поле и указывается пароль для входа в систему.

3.2 Описание программных модулей системы управления проектами

Внедрение информационной системы не изменит процессы, выполняемые в организации. Для внедрения системы потребуются дополнительные ресурсы. Специалисты по внедрению подготовят всю необходимую инфраструктуру.

Внедрение системы в отдел позволит увеличить эффективность работы сотрудников, повысить количество выполняемых проектов за определенный период времени.

Настройка автоматизированной системы на рабочие места пользователей, определение их прав и паролей выполняются администратором системы, который будет назначен согласно штатному расписанию организации. Работать с системой будут только те сотрудники, которые будут иметь отношение к проектам в отделе.

Работа персонала должна соответствовать требованиям по безопасности труда.

Основным программным модулем информационной системы будет TeamTrack.WpfApplication, который будет содержать классы и методы системы необходимые для выполнения процесса управления проектами.

3.3 Разработка информационной системы управления проектами

Загрузка информационной системы потребует пройти проверку введенных данных, форма загрузки программы показана на рисунке 14. Пользователь должен ввести данные, которые будут ему присвоены администратором.

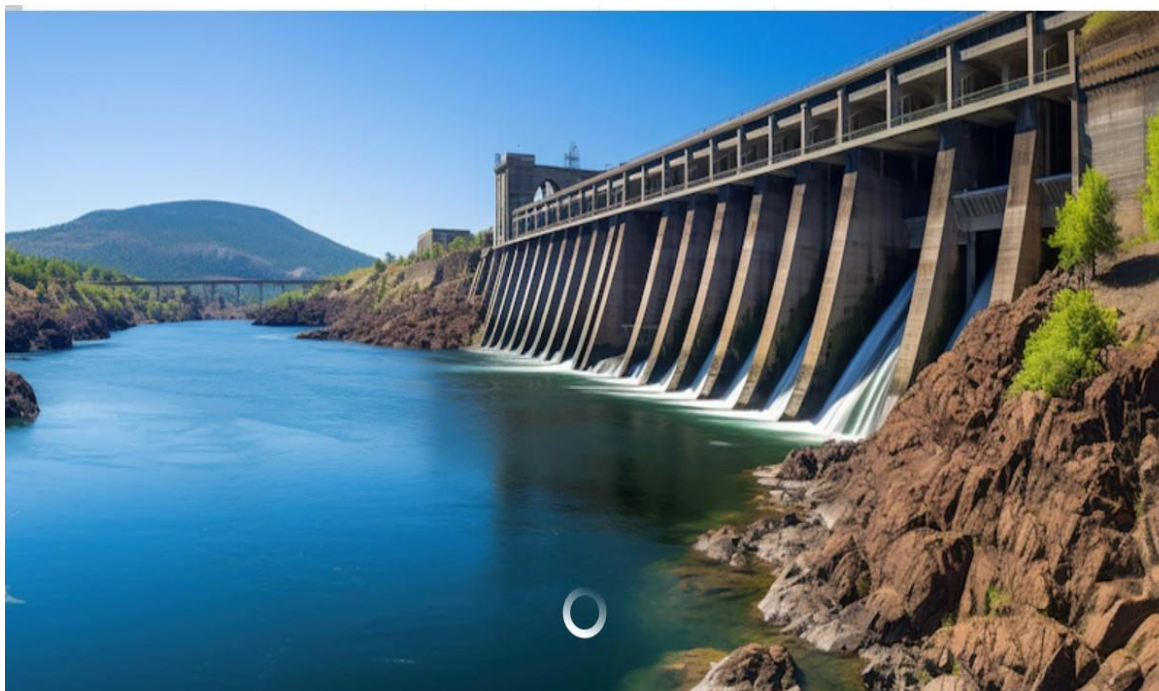


Рисунок 14 – Форма с запуском информационной системы

Матрица прав доступа к информации вводится администратором для каждого пользователя до начала работы. Таким образом, у каждого персонала свой набор возможностей работы с системой, основанный исключительно на его специализации. Описание контрольного примера будет происходить на правах начальника отдела, который занимается непосредственно назначением задач и мониторингом их выполнения, и сотрудника, который выполняет назначенные задачи. Форма авторизации показана на рисунке 15.

Авторизация

Логин:

BorisovAV

Пароль:

•••••

Войти

Рисунок 15 – Форма авторизации в системе

При входе в систему начальник отдела входит в свой личный кабинет, где отображаются проекты и задачи, которые уже находятся на выполнении, которые показаны на рисунке 16.

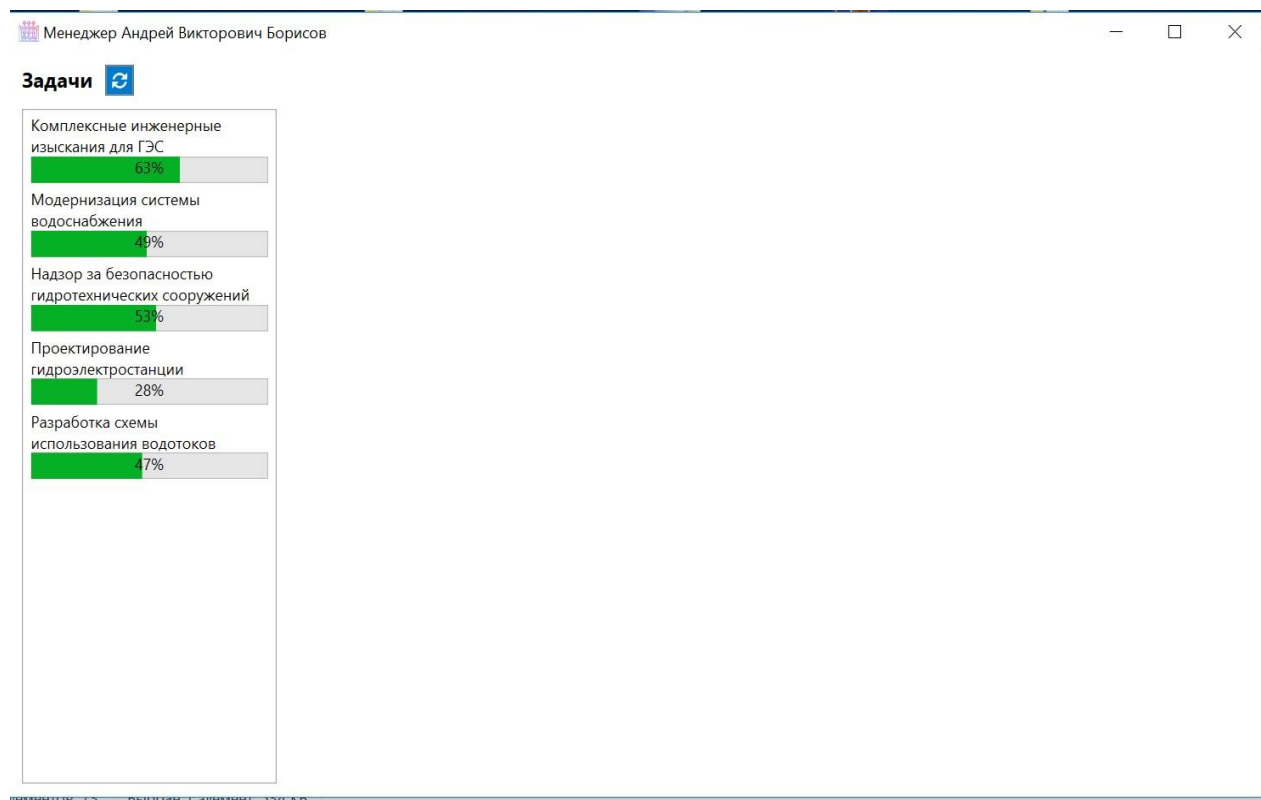


Рисунок 16 – Список проектов в работе

Проекты из списка имеют название и шкалу выполнения, отмеченную зеленым цветом. Список задач и проектов может обновляться несколько раз в день при необходимости. Список показывает только те проекты, которые еще не выполнены и находятся на выполнении. Для создания нового проекта, задачи или подзадачи в системе есть готовая форма, которая позволит руководителю проекта добавлять новые данные о проекте и задачах

Форма для добавления новых задач показана на рисунке 17.

Новое задача для проекта "Комплексные инженерные изыскания для ГЭС"

Срок выполнения задачи: 26.09.2024

Сотрудник: Виктория Андреевна Захарова

Описание задачи: Расчет сметно-проектной документации

Ok

Рисунок 17 – Добавление новой задачи

Добавление новой задачи или подзадачи проекта позволяет руководителю проекта выбирать и назначать сотрудников из списка. При создании новой задачи руководитель вводит описание задачи и устанавливает сроки выполнения [24].

После создания новой задачи и подтверждения на сохранение, задача будет сохранена и направлена в личный кабинет того сотрудника, который был назначен на эту задачу, рисунок 18.

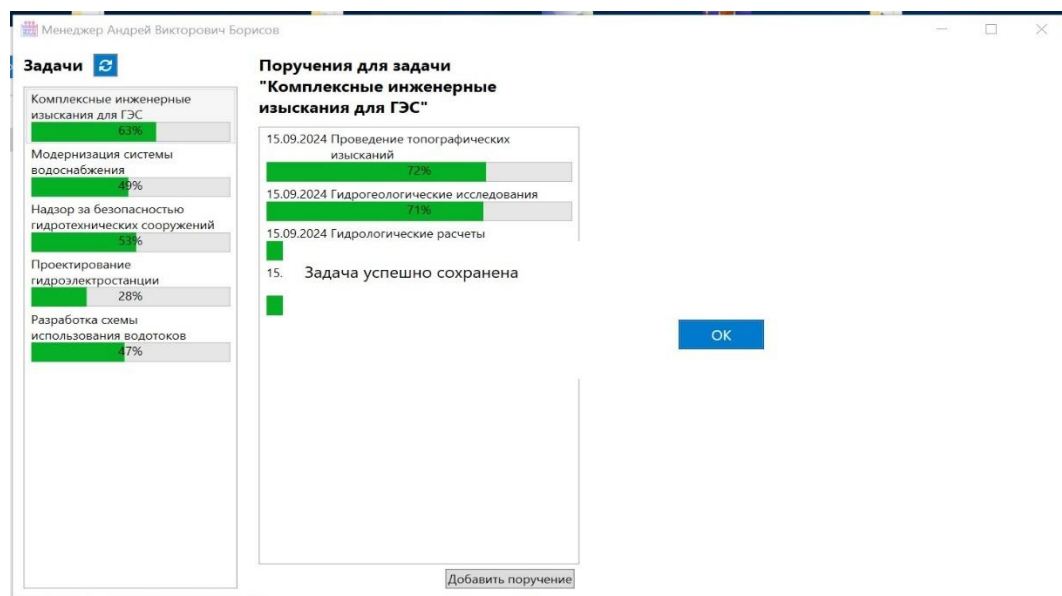


Рисунок 18 – Подтверждение создания задачи

После того как пользователь (начальник отдела) подтверждает, что задача сохранена она появляется в списке задач данного проекта, рисунок 19.

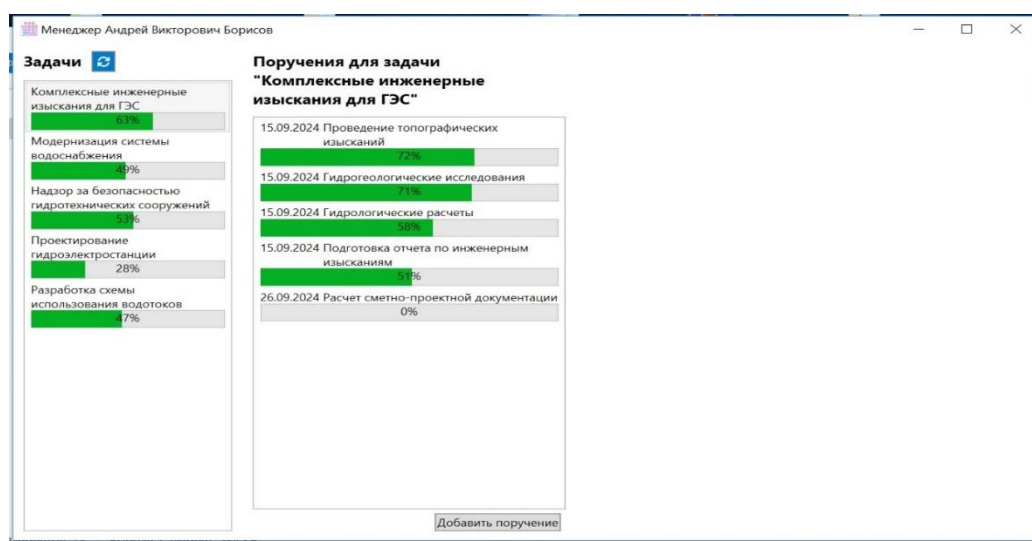


Рисунок 19 – Добавление новой задачи

Как видно новая задача не имеет еще процент выполнения, так как была только добавлена.

Рассмотрим функциональность системы со стороны сотрудника, которому назначаются задачи.

Сотрудники отдела работают в системе при помощи личного кабинета, в который также нужно войти под своим логином и паролем.

Сотрудник, осуществляющий выполнение задач проекта, которые ему были назначены, получает список задач, как только зайдет в свой личный кабинет и обновит задачи. Задачи показаны в виде списка с процентом выполнения если работа с ними уже ведется. Задачи показаны на рисунке 20.

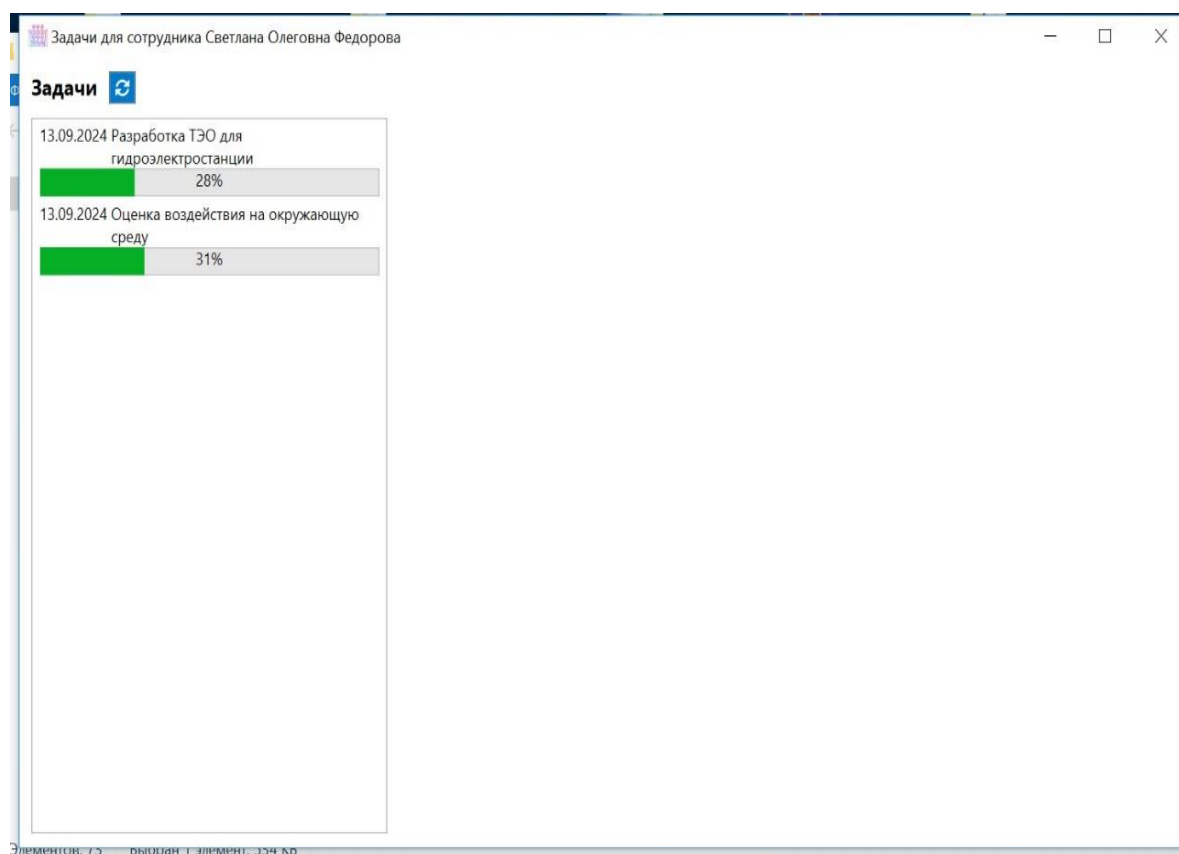


Рисунок 20 – Список задач у сотрудника

Список полученных задач пользователем (сотрудник отдела) происходит автоматически при входе в систему под своими данными [20]. Обновить задачи можно нажав на кнопку «Обновить» как показано на рисунке 21.

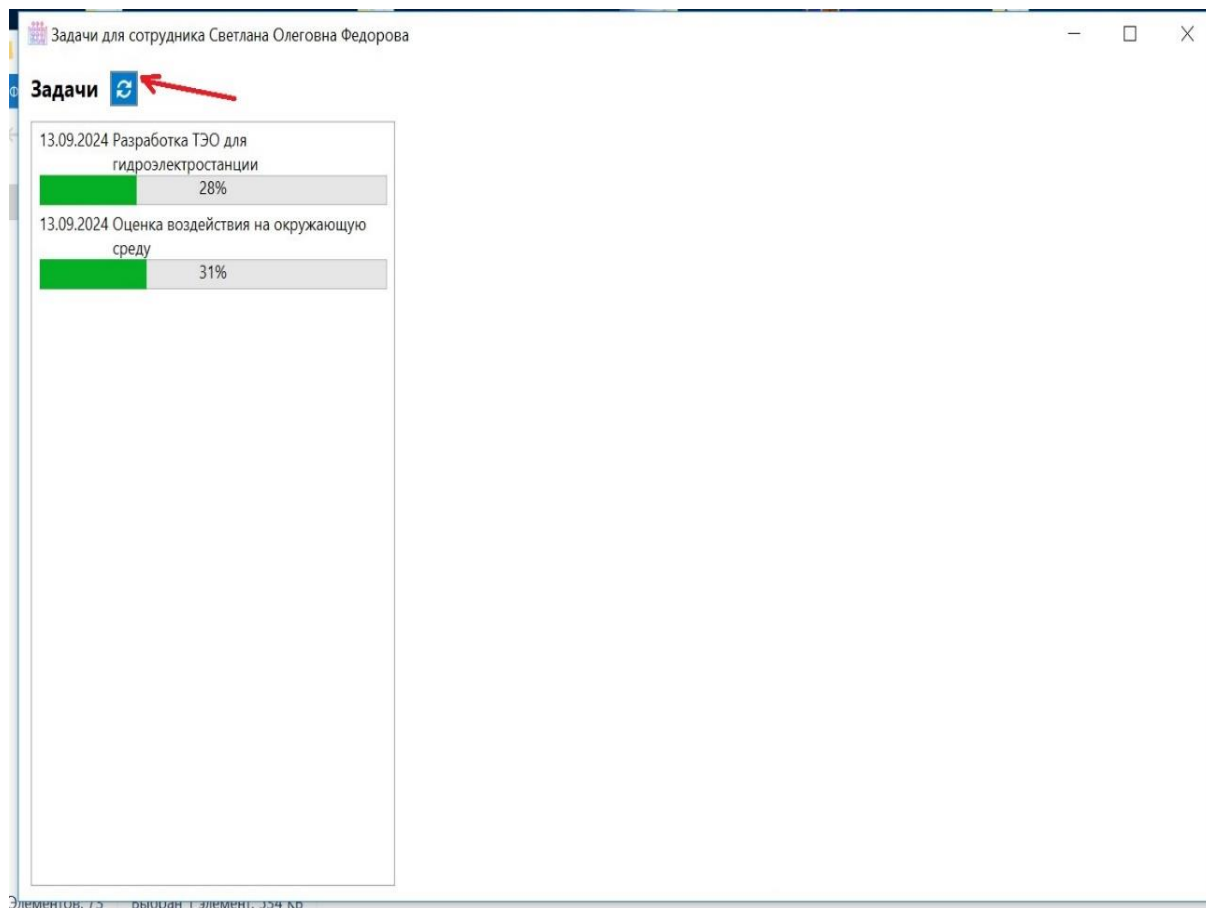


Рисунок 21 – Обновление списка задач

Обновление списка задач позволит получить новые задачи, которые были назначены в то время, когда сотрудника не было на рабочем месте и список не обновлялся [22]. Сотрудник видит в своем кабинете только те задачи, которые назначены именно ему.

Каждая задача имеет срок выполнения и описание, которое показано после того, когда пользователь нажмет на задачу, рисунок 22.

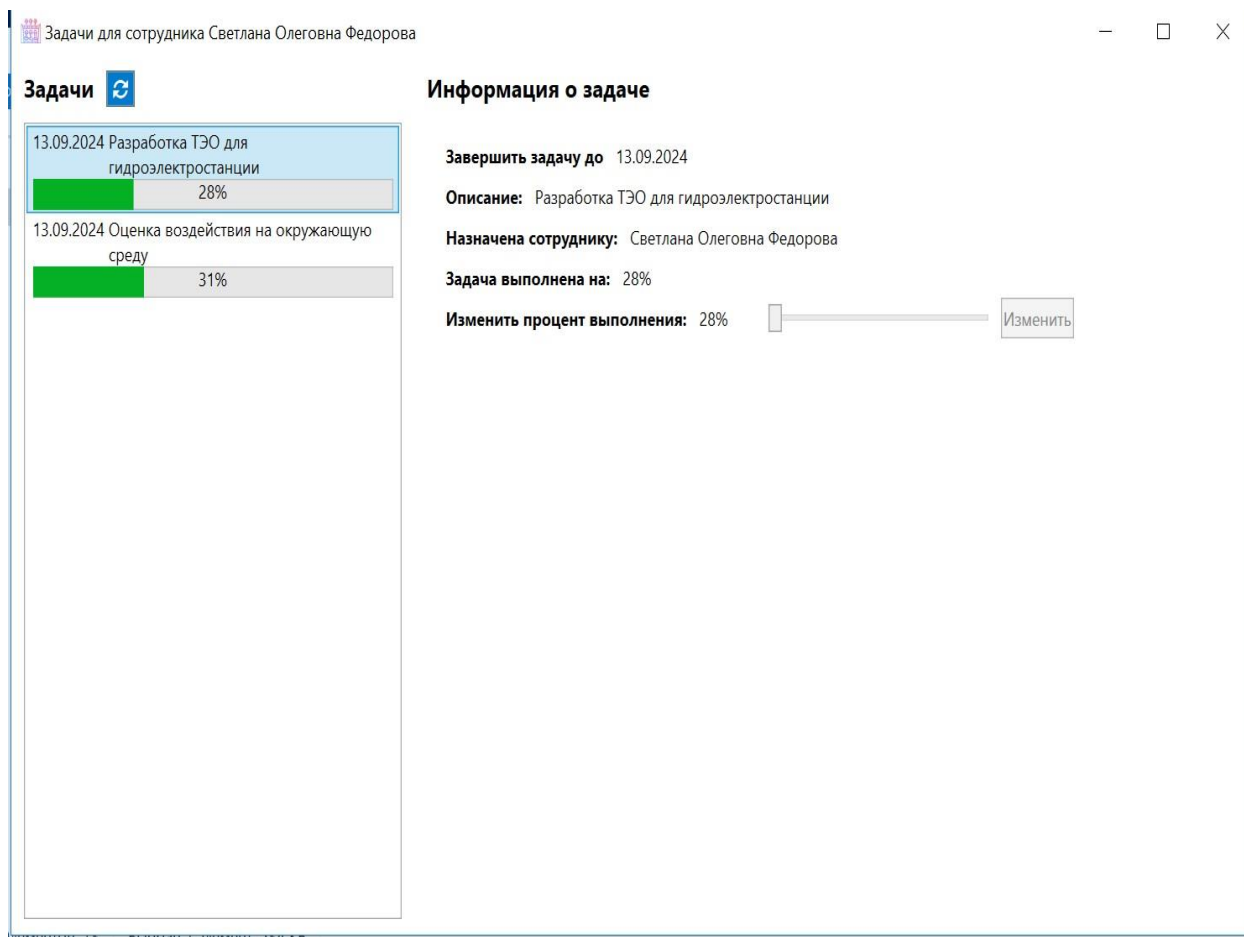


Рисунок 22 – Описание задачи

На форме со списком видно, что задачи назначены для сотрудника Федоровой Светлане Олеговне. Список задач имеет процент завершения каждой задачи. Если нажать на задачу открывается информация о задаче [23].

В информации представлена информация о сроке завершения задачи, описание задачи, информация о проценте завершения задачи и возможность изменить процент завершения задачи, после того как было что-то по ней сделано сотрудником. Изменение процента выполнения задачи можно выполнить как показано на рисунке 23.

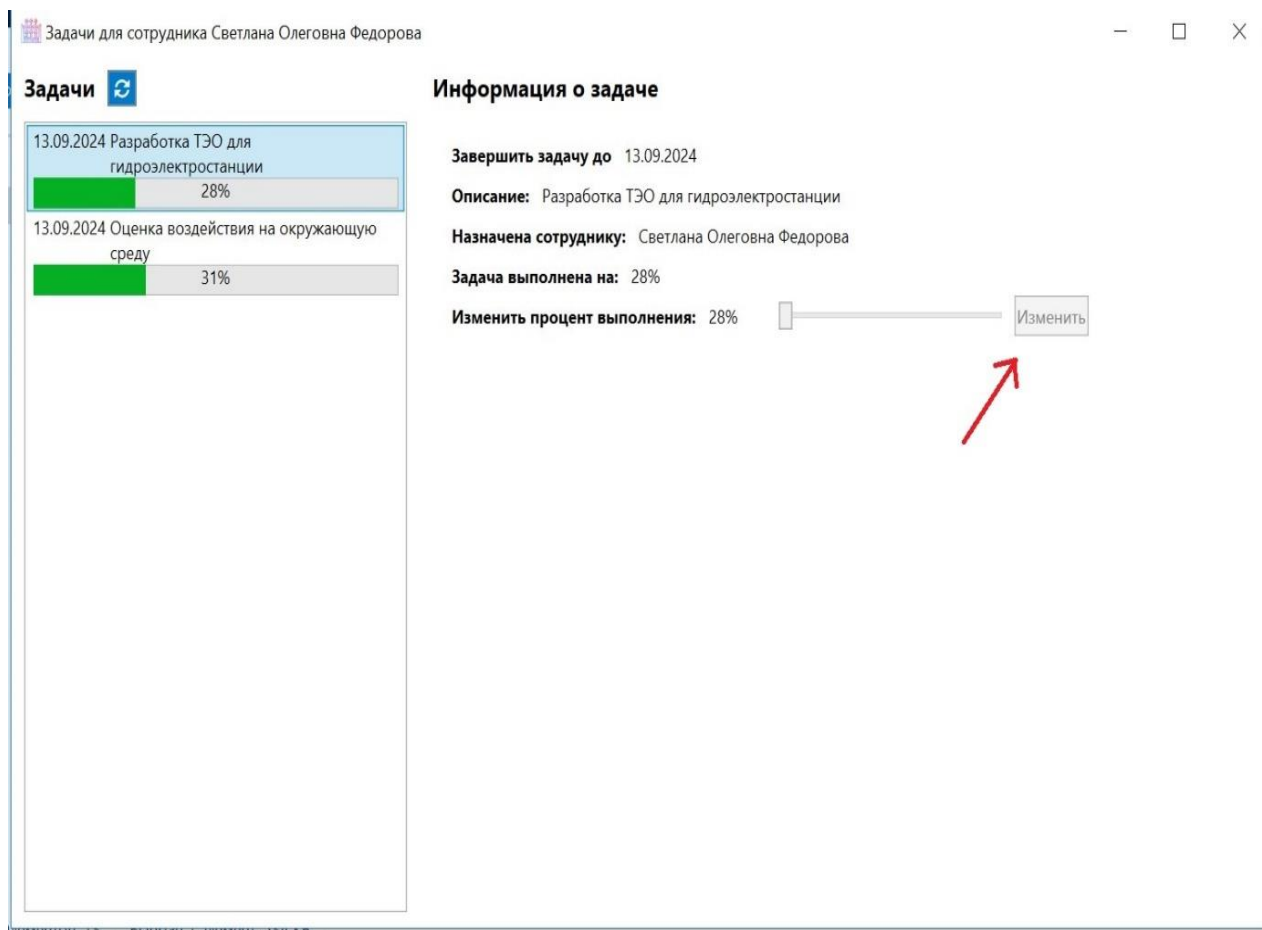


Рисунок 23 – Изменение процента выполнения задачи

После того как будет нажата кнопка «Изменить» процент выполнения задачи изменится как показано на рисунке 24.

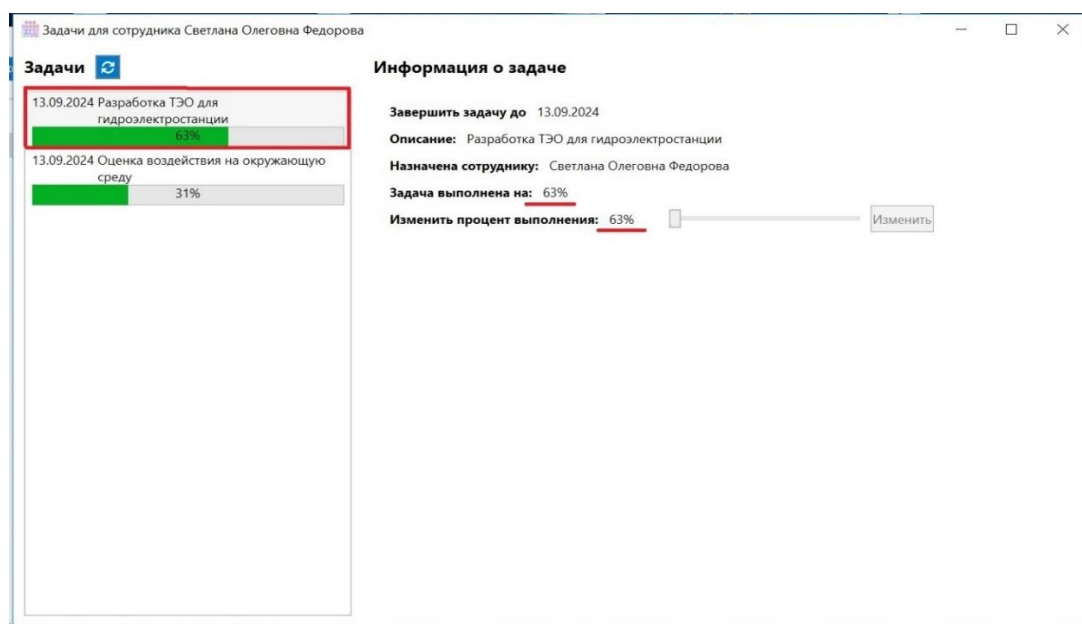


Рисунок 24 – Изменение процента выполнения задачи

Если задача выполняется на 100 % она исчезает из списка выполняемых задач у сотрудника. Изменим процент выполнения определенной задачи как показано на рисунке 25.

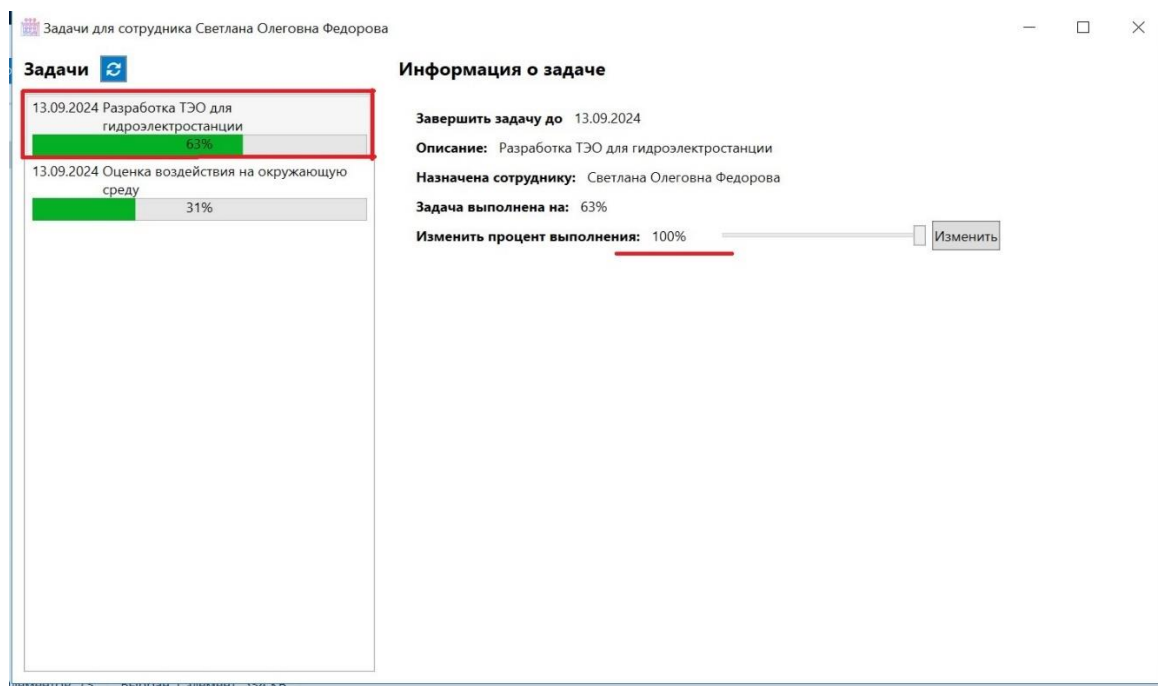


Рисунок 25 – Изменение процента выполнения задачи

После нажатия кнопки «Изменить», задача поменяет процент выполнения как 100% и исчезнет из списка выполняемых задач, рисунок 26.

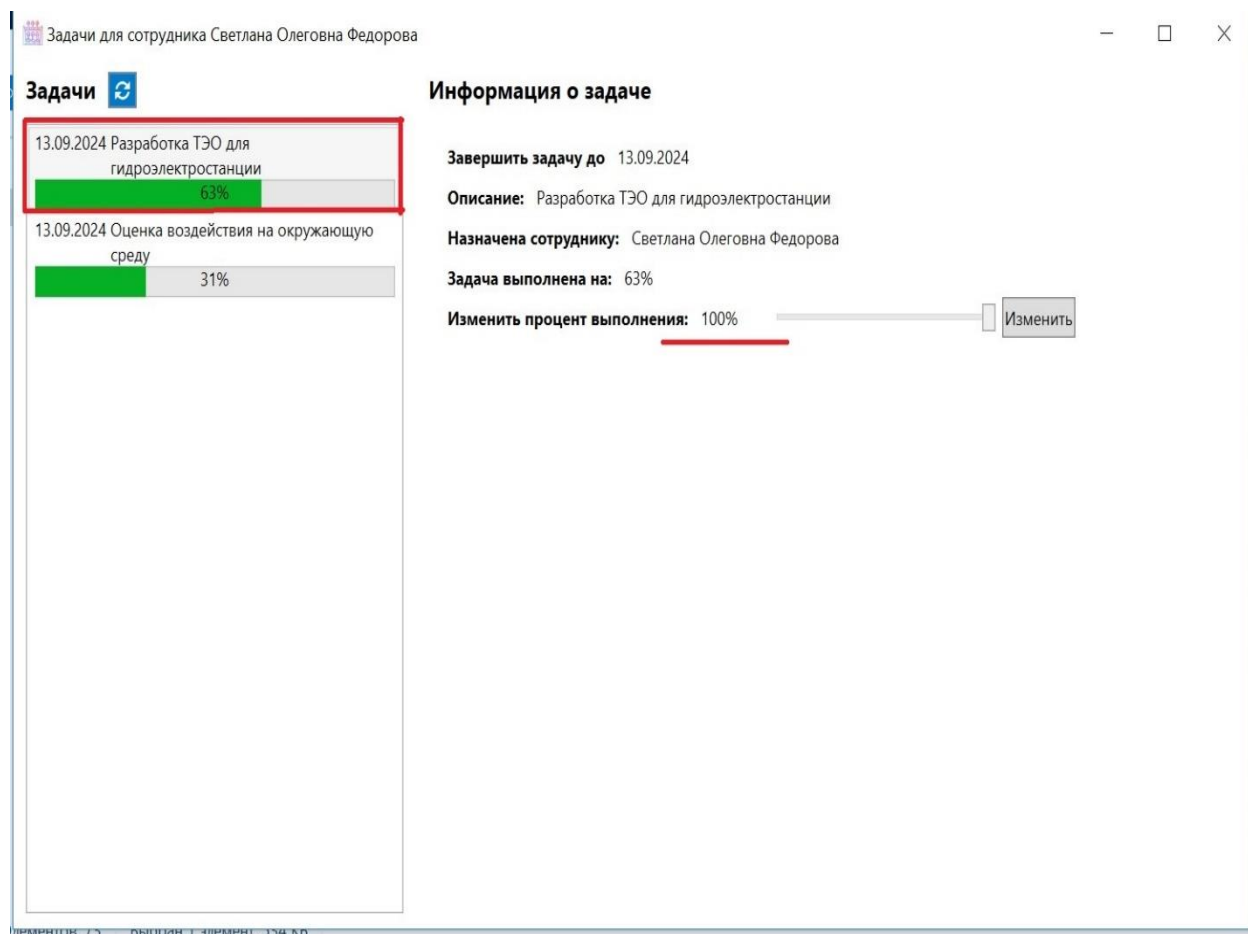


Рисунок 26 – Исчезновение задачи после выполнения

Однако задача исчезает из списка задач только у сотрудника, которому она была назначена, руководитель проекта (начальник отдела), который будет мониторить выполнение задач и проекта в целом будет видеть эту задачу как выполненную на 100%.

Зайдем под правами доступа начальника отдела, откроем этот же проект, мониторинг выполнения проектов будет показан в его личном кабинете, рисунок 27.

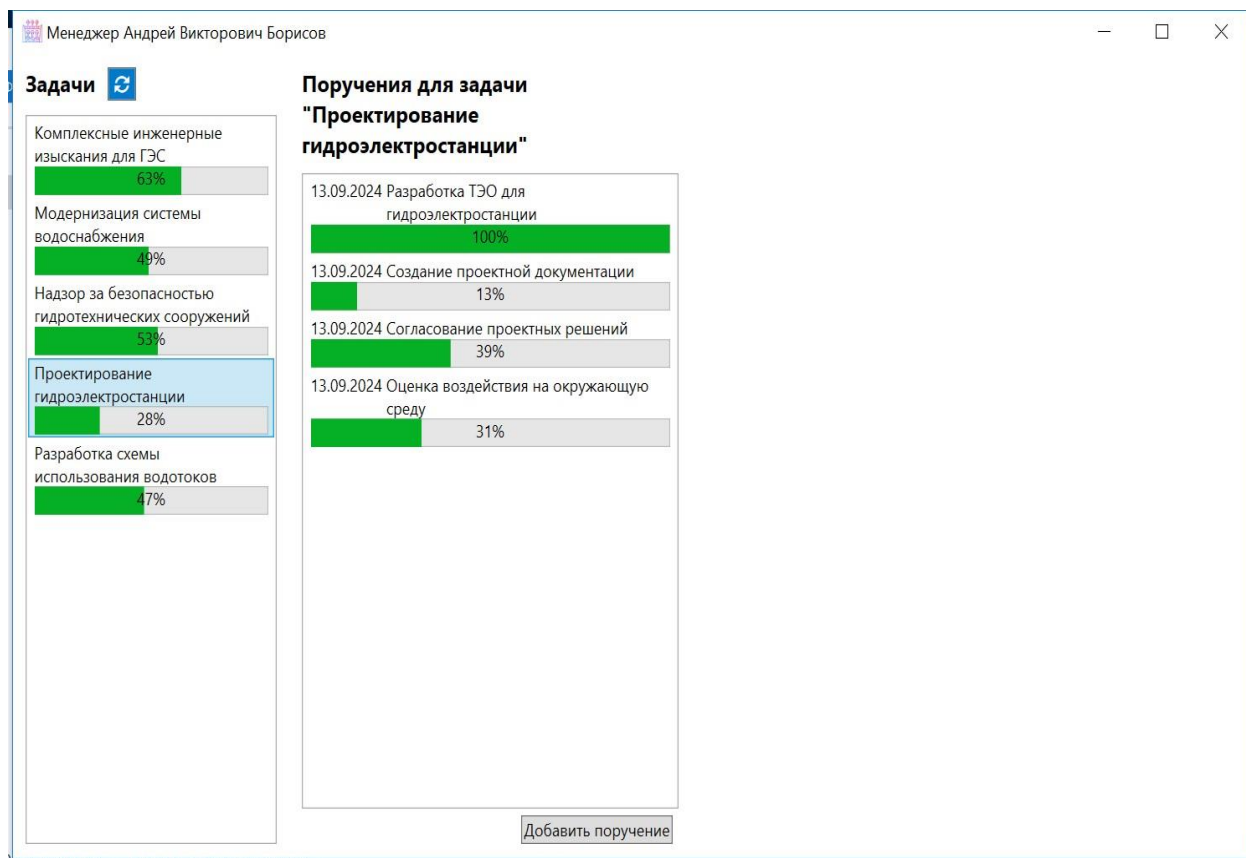


Рисунок 27 – Проекты в кабинете начальника отдела

Как видно в кабинете руководителя проекта задачи никуда не исчезают после того, как выполняются и та задача, которая была выполнена сотрудницей Федоровой С.О. показана здесь как 100% выполненная задача.

Руководитель проектов может вести несколько проектов одновременно и отслеживание выполнения задач проектов значительно упрощается с помощью внедрения системы для мониторинга и назначения задач сотрудникам.

Задачи создаются в каждом проекте и могут быть создано сколько угодно задач, на усмотрение руководителя, который планирует сроки выполнения проекта и контролирует их выполнение. Также задачи назначаются сотрудникам согласно их должности и уровня профессионализма.

Чтобы в любой момент выйти из системы необходимо на верхней панели нажать на кнопку «Выход», согласно рисунку 28.

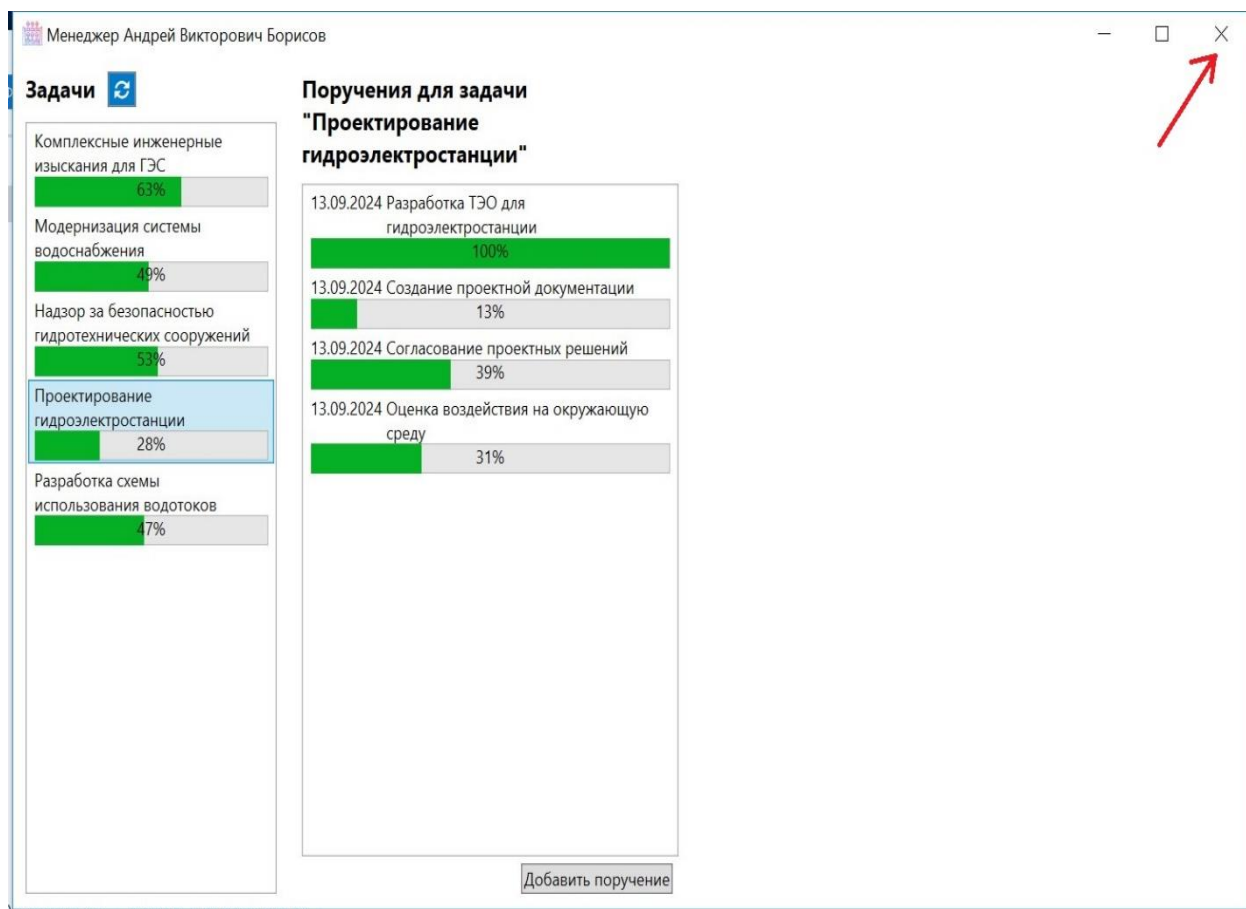


Рисунок 28 – Выход из системы

Система является сложным программным модулем, установка и настройка которого осуществляется администратором системы.

При внедрении системы могут возникнуть непредвиденные ситуации, которые увеличат срок внедрения системы в отделе управления проектами.

Администратор системы и специалисты по внедрению должны будут подготовить инфраструктуру отдела и рабочих мест сотрудников, что обеспечит беспрепятственную установку системы на рабочих местах.

3.4 Тестирование информационной системы управления проектами

Тестирование информационной системы можно провести разными способами, которые используются часто тестировщиками. Тестирование возможно провести с помощью ручного тестирования, такой способ подразумевает ручное тестирование системы тестировщиком без использования автоматизированных программ.

Автоматизированное тестирование проводится с использованием автоматизированных программ без участия человека.

Проверка корректности информационной системы управления проектами будет проводиться ручным способом, будут вводиться заведомо ложные данные, чтобы проверить проверку системы на обработку данных и вывод сообщений об ошибках.

Тестирование системы начинается с прохождения аутентификации пользователя при входе в систему. Введены неправильные данные логина или пароля, после чего система должна отреагировать и показать пользователю сообщение об ошибке. Сообщение показано на рисунке 29.

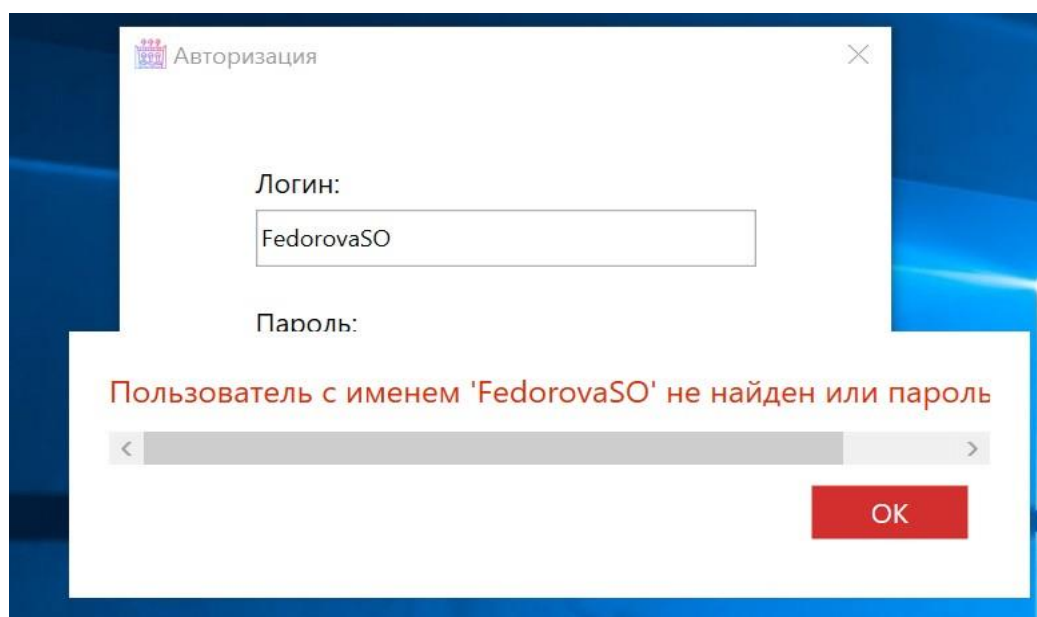


Рисунок 29 – Сообщение об ошибке

Видно, что система сама проверяет вводимые данные и выводит сообщение пользователю на основании чего, пользователь сможет изменить данные на правильные и войти в систему.

Проверка вводимых данных при создании новой задачи будет проводиться при отсутствии выбора данных сотрудника, который будет выполнять задачу при создании задачи. Ошибка при нажатии на создании показана на рисунке 30.

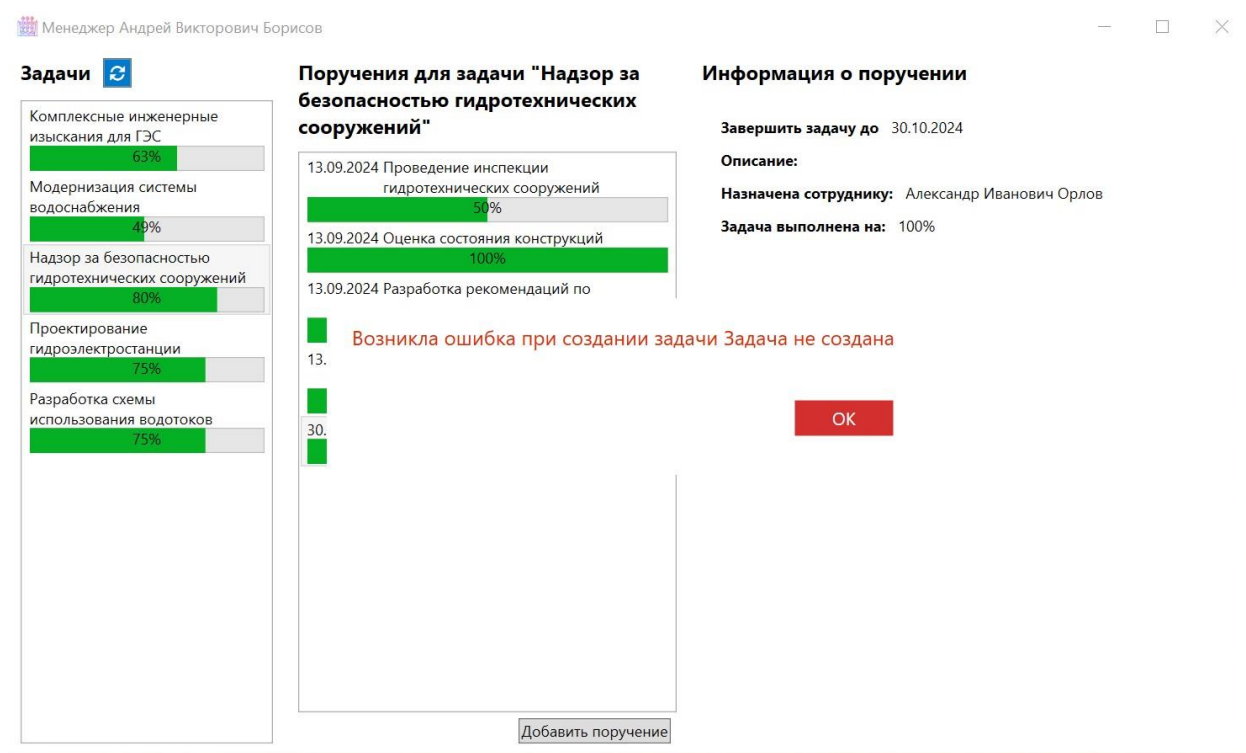


Рисунок 30 – Ошибка при создании задачи

Сообщение сигнализирует пользователю о том, что возникла ошибка при создании задачи, такая задача не создана в системе, тем самым подсказывая пользователю, что нужно создавать другую новую задачу.

При создании новой задачи в системе пользователь обязательно должен выбрать из календаря дату, которая будет определяющей период выполнения задачи. Для устранения допущения ошибок пользователем при выборе даты, система автоматически исключает даты, которые уже были пройдены, тем

самым пользователь сможет выбрать только даты, которые будут в будущем. Рисунок 31 показывает календарь выбора дат в системе.

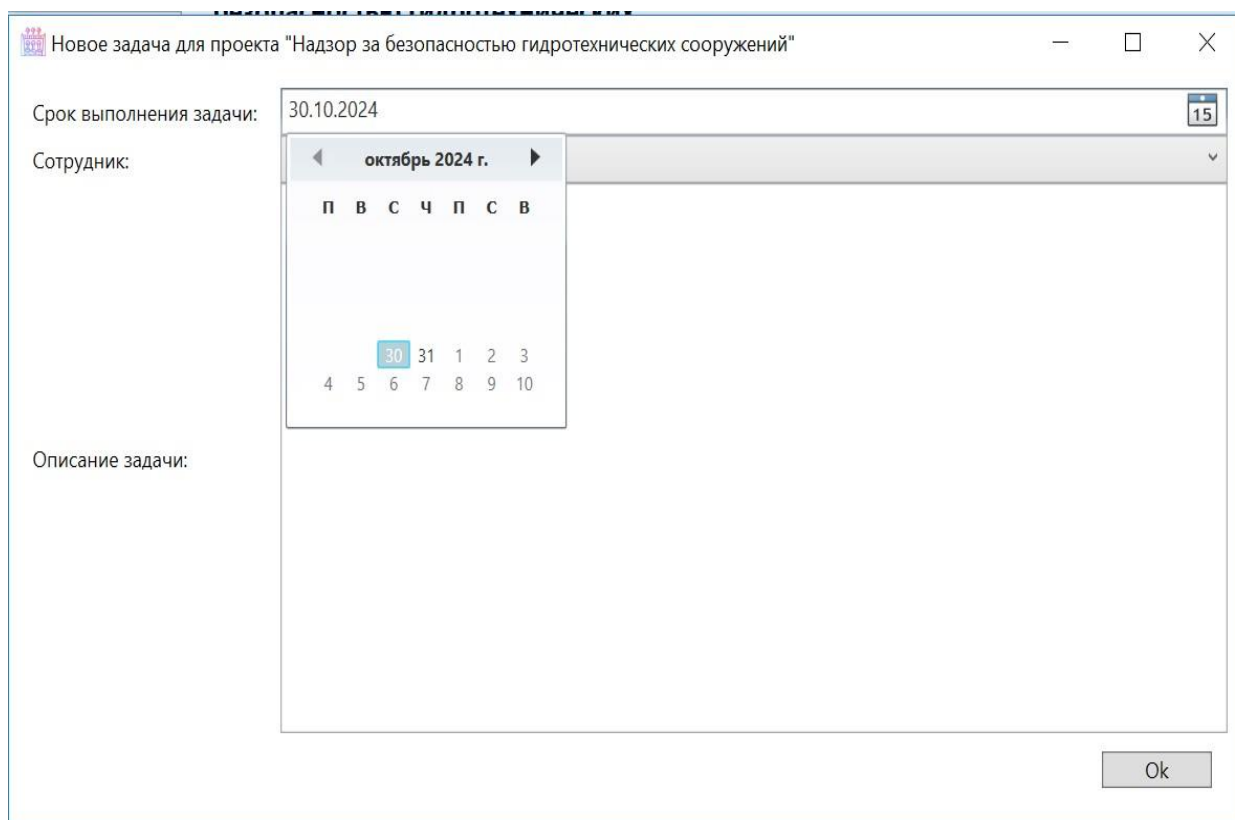


Рисунок 31 – Календарь выбора дат в системе

Рассмотрены основные функции в системе управления проектами, которые требуют внимания. Ручное тестирование продемонстрировало корректность проверки данных, вводимых пользователем и обработку в системе. Тем самым было проведено тестирование, которое показало корректную работу информационной системы управления проектами.

Заключение

Выполненная выпускная квалификационная работа включает завершение всех задач, которые были поставлены в работе для достижения цели по разработке автоматизированной информационной системы управления проектами в организации АО «Гидропроект».

Изначально было проведено исследование, в которое входило изучение деятельности предприятия и его структуры управления и отдельного подразделения, где осуществляется разработка проектов.

Полученные исследования дали представление о деятельности организации и помогли смоделировать процесс управления проектами в подразделении организации «как есть», где были выделены моменты необходимость автоматизации которых была обоснована. Основываясь на диаграммы процесса «как есть» смоделированы диаграммы процесса «как должно быть» и разработаны основные требования к реализации информационной системы.

Для реализации информационной системы были выбраны инструменты разработки C# и СУБД Microsoft SQL 2022, среда разработки Visual Studio 2022, как наиболее подходящие и эффективные для реализации системы управления проектами.

Разработанная информационная система была представлена в виде готовых форм для работы пользователю и описан алгоритм работы руководителя проекта и сотрудника которые будут выполнять основные функции в системе.

При внедрении автоматизированная информационная система повысит прозрачность управления проектами и обеспечит руководителя проектов дополнительным временем на проведение анализа точности реализации отдельно взятого проекта за счет статуса выполнения каждой задачи на проекте.

Список используемой литературы и используемых источников

1. Битрикс24. [Электронный ресурс]. – URL: [Электронный ресурс]. – URL: https://skillbox.ru/media/code/entity_framework/ (дата обращения: 15.09.2024)
2. ГОСТ 34.003-90 Автоматизированные системы. Термины и определения.
3. ГОСТ 24.702-85 Единая система стандартов автоматизированных систем управления. Эффективность автоматизированных систем управления. Основные положения.
4. Горбаченко В. И. Проектирование информационных систем с СА ERwin Modeling Suite 7.3: учебное пособие / В. И. Горбаченко, Г. Ф. Убиенных, Г. В. Бобрышева – Пенза: Изд-во ПГУ, 2012. – 154 с.
5. Гусов К. Н., Андреев А. А., Бондаренко Э. Н. Трудовое право России. Учебник для бакалавров. — М.: Проспект, 2021. 592 с.
6. Еременко, В.Т. Программно-аппаратные средства защиты информации: учебное пособие для высшего профессионального образования / В.Т. Еременко, А.П. Фисун. – Орел: ФГБОУ ВПО «Госуниверситет - УНПК», 2015. – 165 с.
7. Жмуров, Денис Борисович Программно-аппаратные средства защиты информации: учебное пособие / Д.Б. Жмуров, С.В. Жуков. – Самара: Издательство Самарского университета, 2022. – 80 с.
8. Инюшкина О.Г. Проектирование информационных систем (на примере методов структурного системного анализа): учебное пособие / О.Г. Инюшкина, Екатеринбург: «Форт-Диалог Исеть», 2014. 240 с.
9. Как использовать Entity Framework. [Электронный ресурс]. – URL: https://skillbox.ru/media/code/entity_framework/ (дата обращения: 15.09.2024)
10. Концептуальное логическое и физическое моделирование данных. [Электронный ресурс]. – URL: <https://frameworx.ru/SID/datamodelling.html> (дата обращения: 15.09.2024).

11. Мегаклан. [Электронный ресурс]. – URL: <https://megaplan.ru/crm-capabilities/projects-management/> (дата обращения: 15.09.2024)
12. Отличия, достоинства и недостатки базы данных PostgreSQL: что такое PostgreSQL. [Электронный ресурс]. – URL: <https://oracle-patches.com/common/3214-что-такое-postgresql> (дата обращения: 15.09.2024)
13. Семенова О.В. Информационные системы и программирование / О.В. Семенова. – М.: Инфра-М, 2020. – 214 с.
14. Си Шарп: описание и особенности языка [Электронный ресурс]. - URL: <https://otus.ru/journal/si-sharp-opisanie-i-osobennosti-yazyka/> (дата обращения: 15.09.2024)
15. Сухомлинов, А.И. Анализ и проектирование информационных систем: учебное пособие: для студентов, обучающихся по направлению 09.03.03. «Прикладная информатика» / А.И. Сухомлинов. – 2-е издание, испр. и доп. – Владивосток: Издательство Дальневосточного федерального университета, 2021. – 1 CD-ROM; [360 с.]. – Загл. с титул. экр. – ISBN 978-5-7444-5003-8.
16. Терехова, Н. Ю. Методология дизайн-проектирования: учебное пособие / Н.Ю. Терехова. – Москва: Издательство МГТУ им. Н.Э. Баумана, 2019.
17. Учебно-методическое пособие «Проектирование информационных систем» / Сост. Шамсутдинов Т.Ф. Казань: КГАСУ, 2018. – 110 с.
18. Худяков Г.А., Проектирование информационных систем / Г.А. Худяков. Учебное пособие, М.– Омега–Л, 2021. – 432 с.
19. Яндекс Tracker. [Электронный ресурс]. – URL: <https://tracker.yandex.ru/> (дата обращения: 15.09.2024)
20. Model-View-ViewModel (MVVM). [Электронный ресурс]. – URL: <https://learn.microsoft.com/en-us/dotnet/architecture/maui/mvvm> (дата обращения: 15.09.2024)

21. Современное положение энергетики. [Электронный ресурс]. – URL: <http://elibrary.ru> (дата обращения: 15.09.2024)
22. Cross-Platform Desktop GUI Application using .Net Core [Электронный ресурс]. - URL: <https://yasar-yy.medium.com/cross-platform-desktop-gui-application-using-net-core-5894b0ad89a8> (дата обращения: 15.09.2024)
23. Download Visual Studio 2022 Web Installer / ISO (Community / Professional / Enterprise). [Электронный ресурс]. – URL: <https://developerinsider.co/download-visual-studio-2022-web-installer-iso-community-professional-enterprise/> (дата обращения: 15.09.2024)
24. Download and Install Visual Studio 2022. [Электронный ресурс]. – URL: <https://www.itechtics.com/visual-studio-2022/> (дата обращения: 15.09.2024)
25. SQL Server 2022. [Электронный ресурс]. – URL: <https://download-original-windows.ru/msdn-sql-server/sql-server-2022/> (дата обращения: 15.09.2024)
26. Strive. [Электронный ресурс]. – URL: <https://striveapp.ru/> (дата обращения: 15.09.2024)

Приложение А

Листинг программы

```
namespace TeamTrack.Application
{
    public class NavigateBasedOnUserRoleUseCase:
    INavigateBasedOnUserRoleUseCase
    {
        private readonly IPasswordVerificationUseCase
        _passwordVerificationUseCase;

        private readonly INavigationService _navigationService;
        private readonly IEmployeeContext _employeeContext;
        private readonly IEmployeeRepository _employeeRepository;
        public NavigateBasedOnUserRoleUseCase(
            IPasswordVerificationUseCase passwordVerificationUseCase,
            INavigationService navigationService,
            IEmployeeContext employeeContext,
            IEmployeeRepository employeeRepository
        )
        {
            throw new Exception("Сотрудник с таким именем пользователя не
найден");
            _employeeContext.CurrentEmployee = employee;
            if (employee.UserRole == Core.Entities.UserRole.Manager)
                _navigationService.ShowView<IDepartmentHeaderView>();
            if (employee.UserRole == Core.Entities.UserRole.Employee)
                _navigationService.ShowView<IEmployeeView>();
            _navigationService.CloseView<IAuthorizationView>();
            return;
        }
    }
}
```


Продолжение Приложения А

```
}  
  
    catch (AuthenticationException e)  
    {  
        _navigationService.ShowErrorView(e.Message);  
    }  
  
    catch (Exception e)  
    {  
        _navigationService.ShowErrorView($"Что-то пошло не  
так...{e.Message}");  
    }  
}  
}
```

Продолжение Приложения А

```
namespace TeamTrack.Application  
{  
    public class PasswordVerificationUseCase : IPasswordVerificationUseCase  
    {  
        private readonly IEmployeeRepository _employeeRepository;  
        private readonly PasswordService _passwordService;  
  
        public PasswordVerificationUseCase(IEmployeeRepository  
employeeRepository, PasswordService passwordService)  
        {  
            _employeeRepository = employeeRepository ?? throw new  
ArgumentNullException(nameof(employeeRepository));  
        }  
    }  
}
```

Продолжение Приложения А

```
passwordService = passwordService ?? throw new
ArgumentNullException(nameof(passwordService));
}
public UserRole VerifyPassword(string username, string password)
{
    var user = _employeeRepository.FindByUsername(username);
    if (user == null)
        throw new AuthenticationException($"Пользователь с именем
'{username}' не найден или пароль неверен.");
    var isPasswordIncorrect = _passwordService.VerifyPassword(password,
user.PasswordHash);
    if (!isPasswordIncorrect)
        throw new AuthenticationException($"Пользователь с именем
'{username}' не найден или пароль неверен.");
    return user.UserRole;
}
public async Task<UserRole> VerifyPasswordAsync(string username, string
password, CancellationToken cancel = default)
{
    var user = await
_employeeRepository.FindByUsernameAsync(username.ToLower(), cancel);
    if (user == null)
        throw new AuthenticationException($"Пользователь с именем
'{username}' не найден или пароль неверен.");
    var isPasswordIncorrect = _passwordService.VerifyPassword(password,
user.PasswordHash);
    if (!isPasswordIncorrect)
```

Продолжение Приложения А

```
throw new AuthenticationException($"Пользователь с именем '{username}' не  
найден или пароль неверен.");  
    return user.UserRole;  
}  
}  
}
```

Продолжение Приложения А

```
namespace TeamTrack.ConsoleApp  
{  
    internal class DbInitializer  
    {  
        private static Random _random = new Random();  
        public async Task Initialize()  
        {  
            var initializedDepartments = await  
InitializeDepartmentsAsync(Departments);  
            var initializedDepartment = initializedDepartments.FirstOrDefault();  
            if (initializedDepartment is null)  
                throw new InvalidOperationException("Нет инициализированного  
отдела");  
            var employees =  
GenerateEmployeesForDepartment(initializedDepartment);  
            var initializedEmployees = await InitializeEmployeesAsync(employees);  
            var initializedAnotherDepartment =  
initializedDepartments.FirstOrDefault(e => e.Name == "Бухгалтерия");  
            if (initializedAnotherDepartment is null)
```

Продолжение Приложения А

```
throw new InvalidOperationException("Нет инициализированного отдела бухгалтерии");

var anotherEmployees =
GenerateEmployeesForAnotherDepartment(initializedAnotherDepartment);

var anotherInitializedEmployees = await
InitializeEmployeesAsync(anotherEmployees);

var projects = GenerateProjectsForDepartment(initializedDepartment);
var initializedProjects = await InitializeProjectsAsync(projects);
var employeesWithoutDepartmentHead = initializedEmployees
    .Where(e => e.UserRole != UserRole.Manager);
var workTasks = GenerateWorkTasksForProjects(
    employeesWithoutDepartmentHead,
    initializedProjects);
var initializedWorkTasks = await InitializeWorkTasksAsync(workTasks);
var updatedProjects = await UpdateCompletionPercentageProjectsAsync(
    initializedProjects,
    initializedWorkTasks);
}

private static IEnumerable<Department> Departments => new
List<Department>
{
    new Department(0, "ИТ"),
    new Department(0, "Бухгалтерия"),
};

private static async Task<IEnumerable<Department>>
InitializeDepartmentsAsync(
    IEnumerable<Department> departments)
{
```

Продолжение Приложения А

```
ArgumentNullException.ThrowIfNull(departments);

    using var scope = Program.Host.Services.CreateScope();

    var repository = scope.ServiceProvider.GetRequiredService<IDepartmentRepository>();

    await repository.AddRangeAsync(departments);

    var initializedItems = await repository.GetAllAsync();

    return initializedItems;
}

private static IEnumerable<Employee>
GenerateEmployeesForDepartment(Department department)
{
    List<Employee> employees = new List<Employee>
    {
        new Employee(0, "Иван Иванов", "ivanovi",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Manager),
        new Employee(0, "Алексей Смирнов", "smirnova",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Employee),
        new Employee(0, "Сергей Петров", "petrovs",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Employee),
        new Employee(0, "Марина Васильева", "vasilyevam",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Employee),
        new Employee(0, "Ольга Кузнецова", "kuznetsova",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Employee),
        new Employee(0, "Наталья Соколова", "sokolovan",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Employee),
        new Employee(0, "Дмитрий Романов", "romanovd",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Employee),
        new Employee(0, "Елена Попова", "popovae",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Employee),
    }
}
```

Продолжение Приложения А

```
new Employee(0, "Анна Лебедева", "lebedevaa",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Employee),

new Employee(0, "Виктор Николаев", "nikolaevv",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Employee)

};

return employees;

}

private static IEnumerable<Employee>
GenerateEmployeesForAnotherDepartment(Department department)
{
    List<Employee> employees = new List<Employee>
    {
        new Employee(0, "Петр Петров", "petrov",
"827ccb0eea8a706c4c34a16891f84e7b", department, UserRole.Manager)
    };

    return employees;
}

private static async Task<IEnumerable<Employee>>
InitializeEmployeesAsync(
    IEnumerable<Employee> employees)
{
    ArgumentNullException.ThrowIfNull(employees);

    using var scope = Program.Host.Services.CreateScope(); ;

    var repository =
scope.ServiceProvider.GetRequiredService<IEmployeeRepository>();

    await repository.AddRangeAsync(employees);

    var initializedItems = await repository.GetAllAsync();

    return initializedItems;
}
```

Продолжение Приложения А

```

    }

    private static IEnumerable<Project>
    GenerateProjectsForDepartment(Department department)
    {
        DateTime start = DateTime.Today;
        DateTime end = start.AddDays(_random.Next(1, 30));
        List<Project> projects = new List<Project>
        {
            new Project(0, "Разработка системы управления",
            _random.NextDate(start, end), department, 0),
            new Project(0, "Проект по обновлению интерфейса",
            _random.NextDate(start, end), department, 0),
            new Project(0, "Тестирование программного обеспечения",
            _random.NextDate(start, end), department, 0),
        };
        return projects;
    }

    private static async Task<IEnumerable<Project>> InitializeProjectsAsync(
        IEnumerable<Project> projects)
    {
        ArgumentNullException.ThrowIfNull(projects);
        using var scope = Program.Host.Services.CreateScope(); ;
        var repository =
scope.ServiceProvider.GetRequiredService<IProjectRepository>();
        await repository.AddRangeAsync(projects);
        var initializedItems = await repository.GetAllAsync();
        return initializedItems;
    }

```

Продолжение Приложения А

```
private static IEnumerable<WorkTask>
GenerateWorkTasksForProjects(IEnumerable<Employee> employees,
IEnumerable<Project> projects)
{
    var employeeList = employees.ToList();
    var projectList = projects.ToList();
    List<WorkTask> tasks = new List<WorkTask>
    {
        new WorkTask(0, "Разработка пользовательского интерфейса",
projectList[0].DueDate, 0, employeeList[0], projectList[0]),
        new WorkTask(0, "Интеграция базы данных", projectList[0].DueDate,
0, employeeList[1], projectList[0]),
        new WorkTask(0, "Настройка серверного окружения",
projectList[0].DueDate, 0, employeeList[2], projectList[0]),
        new WorkTask(0, "Прототипирование новых интерфейсов",
projectList[1].DueDate, 0, employeeList[0], projectList[1]),
        new WorkTask(0, "Тестирование взаимодействия элементов",
projectList[1].DueDate, 0, employeeList[1], projectList[1]),
        new WorkTask(0, "Разработка дизайн-системы",
projectList[1].DueDate, 0, employeeList[2], projectList[1]),
        new WorkTask(0, "Написание тестовых сценариев",
projectList[2].DueDate, 0, employeeList[0], projectList[2]),
        new WorkTask(0, "Проведение регрессионного тестирования",
projectList[2].DueDate, 0, employeeList[1], projectList[2]),
        new WorkTask(0, "Анализ результатов тестирования",
projectList[2].DueDate, 0, employeeList[2], projectList[2])
    };
    return tasks;
}

private static async Task<IEnumerable<WorkTask>> InitializeWorkTasksAsync(
```


Продолжение Приложения А

```

IEnumerable<WorkTask> workTasks)
{
    ArgumentNullException.ThrowIfNull(workTasks);
    using var scope = Program.Host.Services.CreateScope(); ;
    var repository =
scope.ServiceProvider.GetRequiredService<IWorkTaskRepository>();
    await repository.AddRangeAsync(workTasks);
    var initializedItems = await repository.GetAllAsync();
    return initializedItems;
}

private static async Task<IEnumerable<Project>>
UpdateCompletionPercentageProjectsAsync(
    IEnumerable<Project> projects,
    IEnumerable<WorkTask> workTasks)
{
    ArgumentNullException.ThrowIfNull(projects);
    ArgumentNullException.ThrowIfNull(workTasks);
    foreach (var project in projects)
    {
        var averageCompletionPercentage =
GetProjectAverageCompletionPercentage(workTasks, project);
        project.UpdateCompletionPercentage(averageCompletionPercentage);
    }
    using var scope = Program.Host.Services.CreateScope(); ;
    var repository =
scope.ServiceProvider.GetRequiredService<IProjectRepository>();
    await repository.UpdateRangeAsync(projects);
    var initializedItems = await repository.GetAllAsync();

```

Продолжение Приложения А

```
return initializedItems;

    }

    /// <summary>
    /// Возвращает среднее арифметическое завершения проекта,
    /// выбирая соответствующие задачи из списка задач
    /// </summary>
    /// <param name="workTasks">список, который содержит задачи,
    /// задачи могут быть назначены не только требуемому проекту,
    /// метод отфильтрует из списка только те задачи,
    /// которые назначены указанному проекту
    /// </param>
    /// <param name="project">проект, для которого требуется рассчитать
    /// среднее арифметическое завершения проекта исходя из задач</param>
    /// <returns></returns>

    private static int
    GetProjectAverageCompletionPercentage(IEnumerable<WorkTask> workTasks,
    Project project)
    {
        ArgumentNullException.ThrowIfNull(workTasks);
        ArgumentNullException.ThrowIfNull(project);
        if (!workTasks.Any())
            return 0;

        var onlyProjectWorkTasks = workTasks.Where(task => task.Project.Id ==
project.Id);
        if (onlyProjectWorkTasks is null)
            return 0;
        if (!onlyProjectWorkTasks.Any())
```

Продолжение Приложения А

```
return 0;

        return Convert.ToInt32(Math.Round(onlyProjectWorkTasks.Average(task
=> task.CompletionPercentage), 2));
    }
}
}
```

Продолжение Приложения А

```
namespace TeamTrack.ConsoleApp
{
    static class DbRegistrator
    {
        public static IServiceCollection AddDatabase(this IServiceCollection services,
        IConfiguration Configuration) => services
            .AddDbContext<TeamTrackDbContext>(opt =>
            {
                var type = Configuration["Type"];
                switch (type)
                {
                    case null: throw new InvalidOperationException("Не определён тип
БД");

                    default: throw new InvalidOperationException($"Тип подключения
{type} не поддерживается");

                    case "MSSQL":
                        opt.UseSqlServer(Configuration.GetConnectionString(type));
                        break;

                    case "MSSQL_Local":
```

Продолжение Приложения А

```
AppDomain.CurrentDomain.SetData("DataDirectory",
Directory.GetCurrentDirectory());

        opt.UseSqlServer(Configuration.GetConnectionString(type));
        break;
    case "SQLite":
        opt.UseSqlite(Configuration.GetConnectionString(type));
        break;
    }
})
.AddTransient<DbInitializer>()
.AddRepositoriesInDB()
;
}
}
```

Продолжение Приложения А

```
namespace TeamTrack.ConsoleApp
{
    internal class Program
    {
        #region Host
        private static object _singletonLocker = new Object();
        private static IHost? _host;
        public static IHost Host
        {
            get
            {

```

Продолжение Приложения А

```
if (_host == null)
{
    lock (_singletonLocker)
    {
        if (_host == null)
            _host =
CreateHostBuilder(Environment.GetCommandLineArgs()).Build();
    }
}
return _host;
}

public static IServiceProvider Services => Host.Services;
public static IHostBuilder CreateHostBuilder(string[] arg)
{
    var hostBuilder =
Microsoft.Extensions.Hosting.Host.CreateDefaultBuilder(arg);
    hostBuilder.UseContentRoot(Environment.CurrentDirectory);
    hostBuilder.ConfigureAppConfiguration((host, cfg) =>
    {
        cfg.SetBasePath(Environment.CurrentDirectory);
        cfg.AddJsonFile("appsettings.json", optional: true, reloadOnChange:
true);

    });
    hostBuilder.ConfigureServices(ConfigureMyServices);
    return hostBuilder;
}
```

Продолжение Приложения А

```
public static void ConfigureServices(HostBuilderContext host,
IServiceCollection services)
{
    services.RegisterServices();
    services.AddDatabase(host.Configuration.GetSection("Database"));
}
#endregion

static async Task Main(string[] args)
{
    using var scope = Host.Services.CreateScope();
    await Task.Delay(1000);
    await
scope.ServiceProvider.GetRequiredService<DbInitializer>().Initialize();
    await Console.Out.WriteLineAsync("Программа завершила свою
работу." +
    " Нажмите Enter для завершения.");
    await Console.In.ReadLineAsync();
}
}
```

Продолжение Приложения А

```
namespace TeamTrack.Core.Entities
{
    public class WorkTask : Entity
    {
```

Продолжение Приложения А

```
public WorkTask(  
    int id,  
    string description,  
    DateTime dueDate,  
    int completionPercentage,  
    Employee employee,  
    Project project) : base(id)  
{  
    Description = description;  
    DueDate = dueDate;  
    CompletionPercentage = completionPercentage;  
    Employee = employee;  
    Project = project;  
}  
  
public string Description { get; private set; }  
public DateTime DueDate { get; private set; }  
public int CompletionPercentage { get; private set; }  
public Employee Employee { get; private set; }  
public Project Project { get; private set; }  
public void UpdateCompletionPercentage(int newPercentage)  
{  
    CompletionPercentage = newPercentage;  
}  
}
```