

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

---

(наименование института полностью)

Кафедра «Прикладная математика и информатика»  
(наименование)

09.03.03 Прикладная информатика

---

(код и наименование направления подготовки, специальности)

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

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
(БАКАЛАВРСКАЯ РАБОТА)**

на тему «Разработка проекта информационной системы управления заявками  
информационного отдела организации»

Студент

И. А. Фуров

(И.О. Фамилия)

(личная подпись)

Руководитель

к.т.н, Н. В. Хрипунов

(ученая степень, звание, И.О. Фамилия)

Консультант

старший преподаватель, И. Ю. Усатова

(ученая степень, звание, И.О. Фамилия)

## Аннотация

Выпускная квалификационная работа посвящена разработке проекта информационной системы управления заявками информационного отдела организации. Цель работы заключается в создании эффективной системы, обеспечивающей автоматизацию процессов обработки, мониторинга и решения заявок, что способствует повышению производительности информационного отдела.

Работа состоит из трех глав. В первой главе проведен анализ предметной области, рассмотрены текущие процессы управления заявками, выявлены их недостатки, а также сформулированы требования к разрабатываемой системе. Во второй главе представлено проектирование логических и физических моделей системы, включая UML-диаграммы и IDEF0 модели, описывающие текущие и желаемые процессы. В третьей главе описаны этапы реализации проектных решений, выбор средств разработки и тестирование системы.

Практическая значимость работы заключается в разработке системы, которая позволяет улучшить управление заявками, оптимизировать рабочие процессы и ресурсы информационного отдела. Система включает функции регистрации заявок, назначения исполнителей, мониторинга выполнения и уведомления пользователей, что обеспечивает прозрачность и контроль над рабочим процессом.

Структура работы включает введение, три главы, заключение, а также список используемой литературы и используемых источников. Всего работа содержит 53 страниц текста, сопровождаемых 24 рисунками, 11 таблицами и 20 источниками.

В результате выполнения работы разработана информационная система, отвечающая современным требованиям и потребностям предприятия, способствующая повышению эффективности и качества работы информационного отдела.

## **Abstract**

The final qualification work is dedicated to the development of a project for an information request management system for the IT department of the enterprise JSC "Togliattiazot". The aim of the work is to create an efficient system that automates the processes of processing, monitoring, and resolving requests, thereby increasing the productivity of the IT department.

The work consists of three chapters. The first chapter analyzes the subject area, examines the current request management processes, identifies their shortcomings, and formulates the requirements for the developed system. The second chapter presents the design of logical and physical models of the system, including UML diagrams and IDEF0 models that describe the current and desired processes. The third chapter describes the stages of implementing design solutions, the selection of development tools, and the testing of the system.

The practical significance of the work lies in the development of a system that allows improving the management of applications, optimizing work processes and resources of the information department. The system includes functions for registering applications, assigning performers, monitoring progress and notifying users, which ensures transparency and control over the work process.

The structure of the work includes an introduction, three chapters, a conclusion, as well as a list of references and sources used. In total, the work contains 47 pages of text, accompanied by 24 figures, 11 tables and 18 sources.

As a result of the work, an information system was developed that meets modern requirements and needs of the enterprise, helping to improve the efficiency and quality of work of the information department.

## Содержание

Введение.....	5
Глава 1 Анализ предметной области и постановка задачи на разработку проекта информационной системы управления заявками информационного отдела организации .....	7
1.1 Идентификация предметной области .....	7
1.2 Выбор методологии и технологии концептуального моделирования....	10
1.3 Разработка и анализ модели бизнес-процесса «КАК ЕСТЬ» .....	12
1.4 Разработка модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ» .....	16
1.5 Разработка требований к ПО .....	17
1.6 Постановка задачи на разработку .....	20
Глава 2 Проектирование системы управления заявками информационного отдела.....	22
2.1 Разработка логической модели автоматизированной системы управления заявками. ....	22
2.2 Разработка логической модели данных автоматизированной системы управления заявками. ....	29
2.3 Разработка физической модели данных автоматизированной ИСУ заявками. ....	30
2.4 Требования к аппаратно-программному обеспечению автоматизированной системы управления заявками информационного отдела организации .....	31
Глава 3 Реализация проектных решений системы управления заявками информационного отдела организации и тестирование информационной системы управления.....	34
3.1 Выбор средств разработки .....	34
3.2 Функциональность автоматизированной системы управления заявками. ....	39
3.3 Тестирование информационной системы управления заявками АО «Тольяттиазот».....	47
Заключение .....	53
Список используемой литературы и используемых источников.....	55

## Введение

Современное развитие информационно-технологических решений существенно изменило их значение в организации бизнес-процессов в компаниях, а также на их внутреннее и внешнее взаимодействие. В этих условиях IT-отделы выступают не только как поддерживающие компьютерную инфраструктуру и программное обеспечение, но и как стратегические партнеры, предлагающие бизнесу инновационные решения, способствующие обретению конкурентных преимуществ на рынке. Такие тенденции приводят к существенным изменениям в подходе компаний к оценке эффективности своих IT-служб.

Объектом исследования является ИС АО «Тольяттиазот».

Предметом исследования является информационная система управления (ИСУ) заявками информационного отдела АО «Тольяттиазот».

Основной целью выпускной квалификационной работы является разработка проекта системы управления заявками информационного отдела АО «Тольяттиазот». Для достижения этой цели предусмотрены следующие задачи:

- анализ предметной области;
- разработка функциональных моделей текущего и желаемого состояний профессиональной деятельности информационного отдела, а также проектирование новых логических и физических моделей данных для системы управления заявками, отвечающих насущным потребностям отдела;
- реализация проектного решения, отвечающего потребностям информационного отдела и проведение экономической оценки эффективности от его внедрения.

Для проведения исследования используем методы методологически-структурного и объектно-ориентированного синтеза проекта информационных систем.

Практическая ценность работы состоит в создании информационной системы, которая позволит улучшить эффективность управления информационным отделом организации.

Введение, три главы, заключение и перечень используемой литературы и источников составляют структуру бакалаврской работы.

Первая глава содержит краткое ознакомление с деятельностью химического предприятия. Описание разработок функциональных моделей текущего и желаемого состояний работы информационного отдела. Формулировка направления и постановка требований к вопросу улучшения системы управления заявками информационного отдела.

Вторая глава работы включает UML-диаграммы, необходимые для описания предметной области, такие как диаграммы вариантов использования, последовательности и классов. Для точного построения физических моделей данных была также создана и разработана логическая модель данных информационной системы управления отдела.

Третья глава описывает разработку решения проекта, включая структурно-функциональную схему системы управления. Также выполнена экономическая оценка разработанного проекта.

В заключении представлены результаты выпускной квалификационной работы.

Бакалаврская работа содержит 53 страниц текста, 24 рисунка, 11 таблиц и 18 источников.

# **Глава 1 Анализ предметной области и постановка задачи на разработку проекта информационной системы управления заявками информационного отдела организации**

## **1.1 Идентификация предметной области**

Информационная система управления (ИСУ) заявками информационного отдела АО «Тольяттиазот» – это комплекс программных и аппаратных средств, предназначенных для эффективного управления и отслеживания заявок, поступающих в информационный отдел предприятия АО «Тольяттиазот». Она обеспечивает автоматизацию процессов обработки, мониторинга и решения заявок, поступающих из внутренних подразделений предприятия.

Основные функции и возможности информационной системы управления заявками включают:

- регистрацию заявок: система позволяет зарегистрировать заявки с указанием необходимых данных, таких как тип заявки, срок выполнения, приоритетность, описание проблемы и т.д;
- отслеживание статуса: пользователи могут отслеживать текущий статус выполнения заявки – от момента её поступления до завершения. Это позволяет контролировать процесс обработки и управлять приоритетами;
- автоматизация процессов: ИС позволяет автоматизировать ряд рутинных задач, связанных с обработкой заявок, таких как направление заявки на исполнение соответствующему сотруднику, уведомление о статусе заявки, формирование отчётов и т.д;
- мониторинг производительности: система предоставляет инструменты для мониторинга производительности

информационного отдела, такие как отчёты о количестве и времени обработки заявок, что позволяет оптимизировать рабочие процессы и ресурсы;

- безопасность данных: система обеспечивает защиту конфиденциальности и целостности данных, предотвращая несанкционированный доступ и сохраняя аудиторскую информацию о действиях пользователей.

Информационная система управления заявками информационного отдела АО «Тольяттиазот» является ключевым инструментом для повышения эффективности работы отдела, обеспечивая быстрое и качественное реагирование на поступающие запросы и обеспечивает прозрачность и контроль над рабочим процессом.

Схема организационной структуры АО «Тольяттиазот» представлена на рисунке 1 [9].

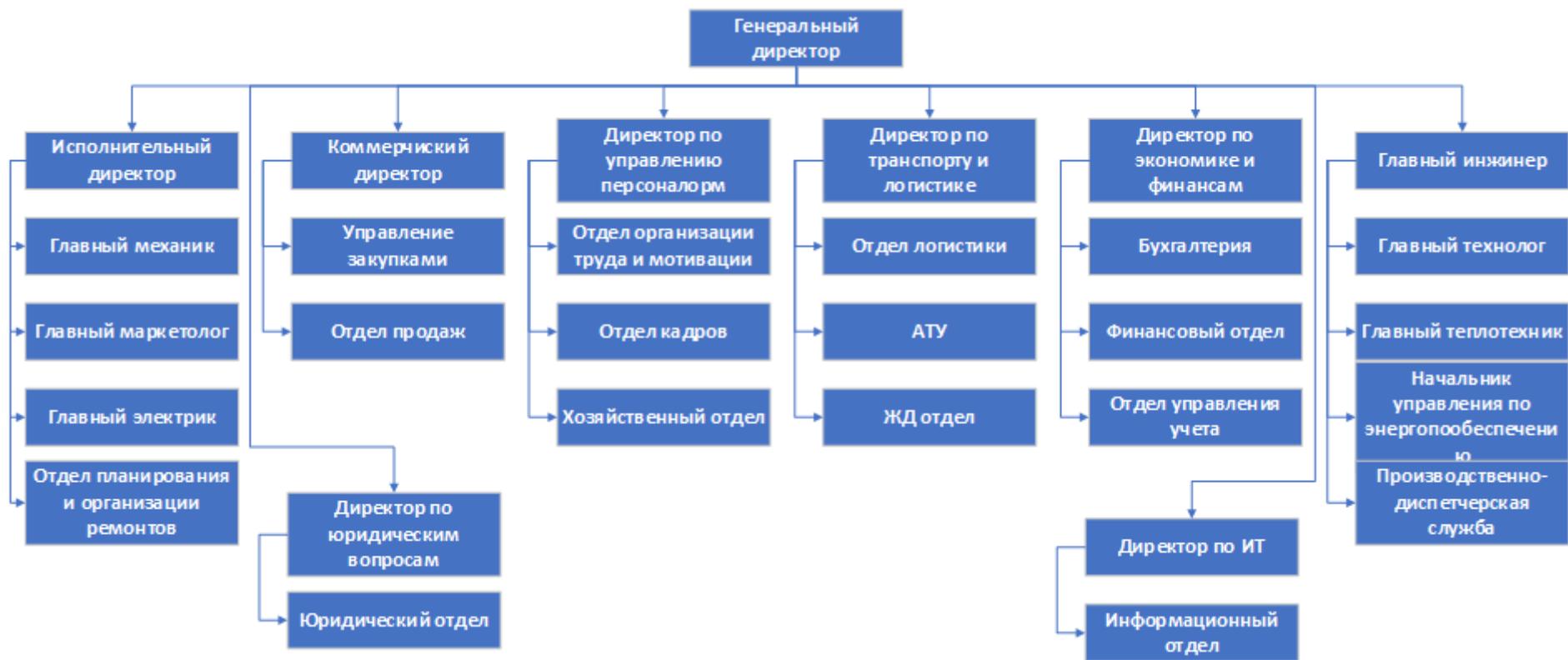


Рисунок 1 - Схема организационной структуры АО «Тольяттиазот»

Анализ текущей организационной структуры и методов управления заявками выявил ряд проблем, требующих решения для повышения эффективности работы информационного отдела. Внедрение новой информационной системы, автоматизирующей процессы и обеспечивающей интеграцию с другими корпоративными системами, позволит значительно улучшить качество управления заявками и сократить временные затраты.

## **1.2 Выбор методологии и технологии концептуального моделирования**

«BPMN на сегодняшний день является одним из самых часто используемых методов описания бизнес-процессов. Данный метод использует события, действия, развилки и многое другое. BPMN позволяет достаточно просто визуализировать бизнес-процессы. Основой моделирования является именно язык описания процессов. Важно осознавать, что это именно язык, имеющий семантику, правила и законы, которые надо изучать» [11].

Методология объектно-ориентированного анализа и проектирования, базирующиеся на стандартах языка UML.

«Основной данной методологии является объектная модель. Большинство современных методов методологии построены на применении языка UML (Unified Modeling Language). UML - это унифицированный язык, который обеспечивает стандартизированный подход к представлению, определению, документированию и проектированию систем, что делает его широко используемым в современных методологиях разработки» [20].

Создавать модель UML можно программами, такими как: Draw.io, Gliffy, Visual Paradigm и др.

Объектно-ориентированный анализ и проектирование подразумевает использование диаграмм классов, компонентов и размещений, позволяющие моделировать архитектуру, иерархию и структуру системы. Также

используются диаграммы поведения, которые позволяют описать поведение классов в системе.

Методология IDEF0 представляется процесс взаимодействия как нечто, что получает на вход информацию для обработки, этой информацией может быть, к примеру, звонок клиента. А на выходе получается итог обработки данной информации, к примеру, покупка товара. Данная методология также включает в себя стратегии и процедуры необходимые для выполнения (управляющие), а также описывает необходимые для проведения работы ресурсы (механизмы).

Создавать модель IDEF0 можно программами, такими как: ERWIN, BPWIN, Visio и др. В моем случае будет использоваться бесплатное программное обеспечение Ramus [10].

Модель IDEF0 можно декомпозировать, разбив на дочерние диаграммы, которые более подробно описывают процессы, происходящие во время работы.

DFD модель является дополнением к IDEF0. Она служит для того, чтобы визуально показать все процессы в виде данных. Это необходимо при проектировании ИСУ. Диаграмма DFD позволяет отобразить процесс движения данных между системными объектами. В диаграмме существует 4 элемента:

- процессы изменения данных (обработка и др.);
- объект, который взаимодействует с данными;
- хранилище для данных;
- поток данных, отображающий направления и сами данные.

Модель IDEF3 так же является дополнением IDEF0. Она описывает сценарии бизнес-процесса и последовательность действий с детальным описанием происходящих в нём событий. Данная модель очень важна, так как успех предприятия зависит от хорошо проработанных, на этапе концептуальных разработок бизнес-процессов.

### 1.3 Разработка и анализ модели бизнес-процесса «КАК ЕСТЬ»

Модель IDEF0 представляет собой функциональную структуру объекта, то есть производимые им действия и связи между этими действиями. Модель «КАК ЕСТЬ» является отражением текущего состояния бизнес-процесса.

Разработанная модель на контекстном уровне содержит один блок «Работа информационного отдела» (рисунок 2).

Входами являются информация заявки, которая содержит следующую информацию:

- id заявки;
- наименование – название заявки, отражающее основную суть записи;
- описание – в этом поле расписывается цель обращения;
- обоснование – основание, на котором необходимо принять заявку;
- файл – можно прикрепить файл, если это необходимо;
- ожидаемая дата реализации – инициатор указывает дату, до которой, желательно, реализовать его запрос.

и информация об клиенте в которой присутствует информация:

- id клиента,
- имя клиента.

Управление включает в себя внутренние регламенты и стандарты, необходимые для организации и эффективного управления процессами.

Механизмы - это инициатор, ИТ специалист и аналитик. Они исполняют процессы работы информационного отдела.

Выходом является решенная заявка. Которая представляет собой результат выполнения всех необходимых работ.

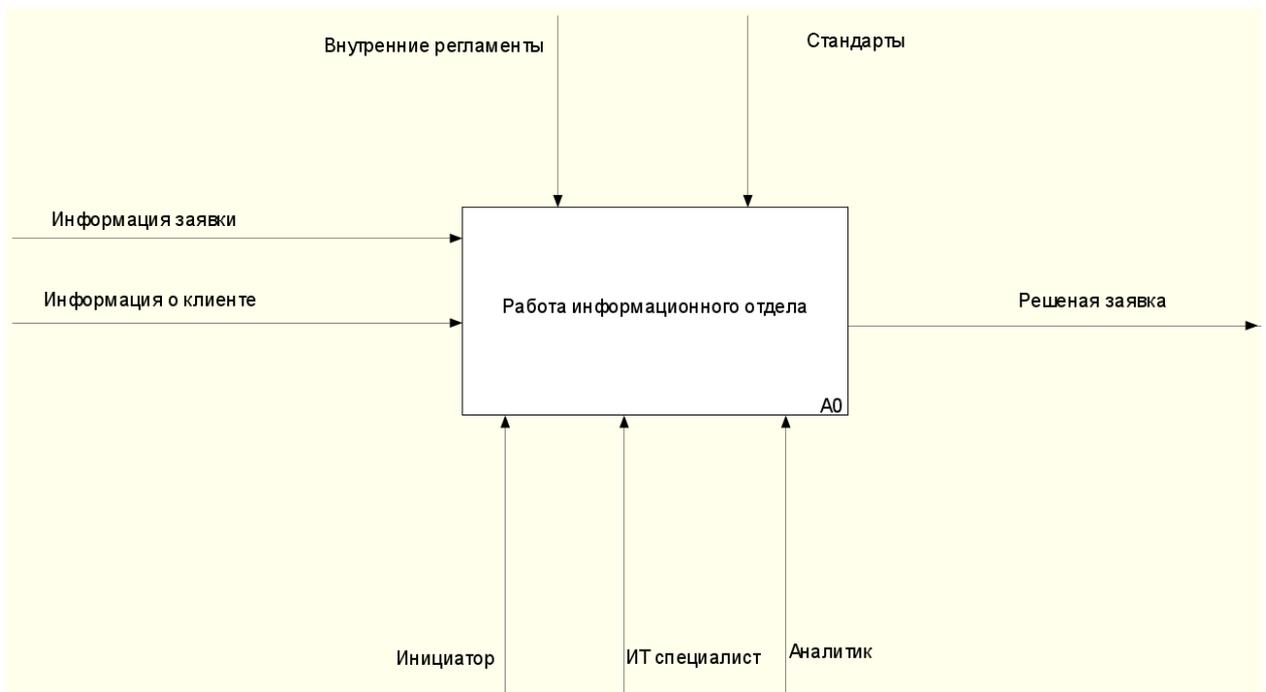


Рисунок 2 - Диаграмма AS-IS «Работа информационного отдела» (Родитель)

После того, как описали контекстную диаграмму, необходимо разбить ее на подсистемы и описывать уже их. Данный процесс называется декомпозицией и показан на рисунке 3.

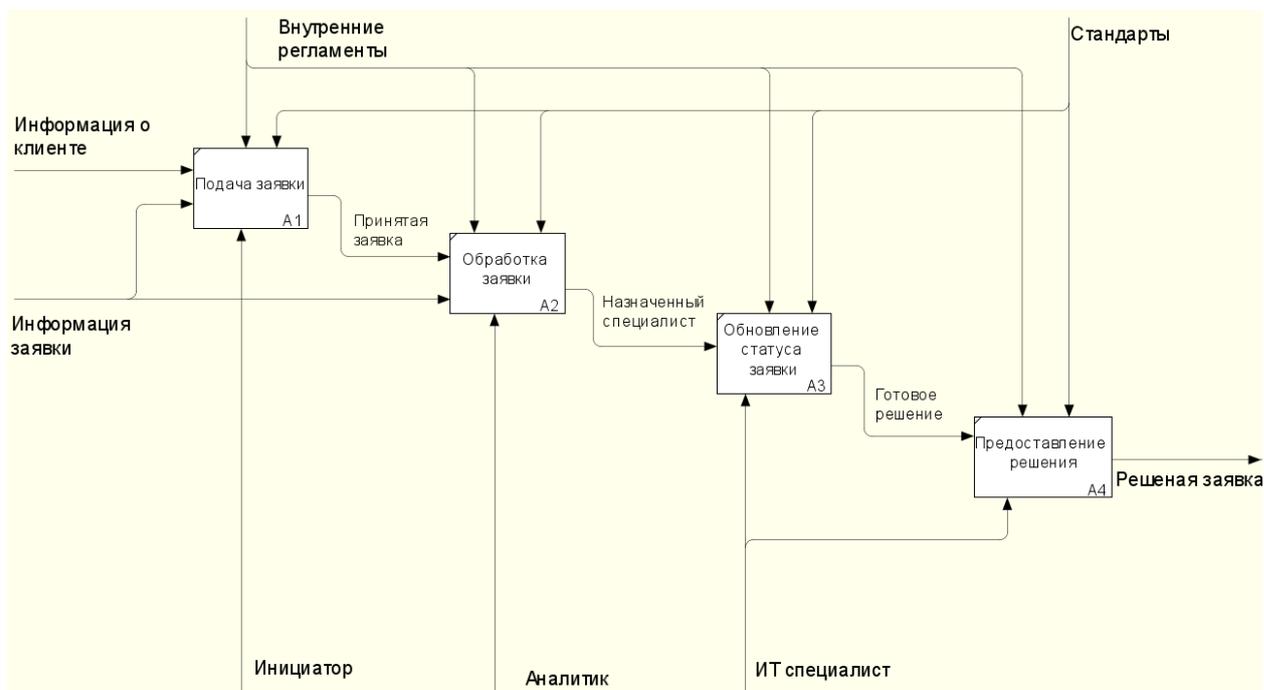


Рисунок 3 - Диаграмма AS-IS «Работа информационного отдела» (Дочерняя)

Функция «Работа информационного отдела» в информационном отделе АО «Тольяттиазот» заключается в приеме и обработке заявок от инициаторов. Важно подчеркнуть, что эта функция представляет собой комплексный процесс, который включает в себя несколько ключевых этапов:

- подача заявки: на этом этапе инициатор оформляет заявку. Это информация от инициаторов об ошибке или неисправности или необходимости в установке необходимого программного обеспечения. Заявка включает в себя описание обращения и обоснование этого обращения.
- обработка заявки: аналитик изучает заявку и назначает программиста для выполнения заявки. После чего программист может выполнить данную ему заявку.
- обновление статуса заявки: после того как программист выполнит все необходимые работы с заявкой, он может изменить статус на «закрыт».

- предоставление решения: после изменения статуса заявки на «закрыт», инициатор может считать, что все необходимые работы выполнены и может проверить качество выполненной работы.

Информационный отдел также поддерживает свою деятельность через внутренние регламенты и стандарты, которые определяют трудовые процессы, безопасность, качество и другие аспекты бизнеса.

Спецификация модели «Как есть» представлена в таблице 1.

Таблица 1 – Спецификация модели бизнес-процесса управления деятельностью информационного отдела бизнес-процесса «Как есть»

Элемент модели	Описание
Входы	Информация заявки Информация о клиенте
Выходы	Решенная заявка
Управление	Внутренние регламенты Стандарты
Исполнители	Инициатор ИТ специалист Аналитик

Данная схема бизнес-процесса имеет недостаток в виде отсутствия автоматизации некоторых процессов. Отсутствуют автоматическое уведомление программистов о появлении у них новой задачи, что может быть неудобно во время большой нагрузки. Какую-то заявку программист может не увидеть.

Всех программистов уведомляют аналитики о новых заявках. Что так же неудобно, поскольку на одного аналитика будет приходиться несколько программистов. Внедрение новой ИС для заявок является важным шагом для повышения эффективности работы информационного отдела. Помимо преимуществ для программистов и аналитиков, внедрение новой информационной системы также сделает процессы более удобными и эффективными для инициаторов.

## 1.4 Разработка модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ»

После выявления недостатков необходимо разработать модель бизнес-процесса «КАК ДОЛЖНО БЫТЬ», которая основана на анализе текущей модели «КАК ЕСТЬ». Цель этой модели состоит в устранении выявленных недостатков и улучшении процессов в организации. В результате внедрения данной модели произойдет автоматизация бизнес-процессов.

Добавим новый механизм – Информационная система (рисунок 4).

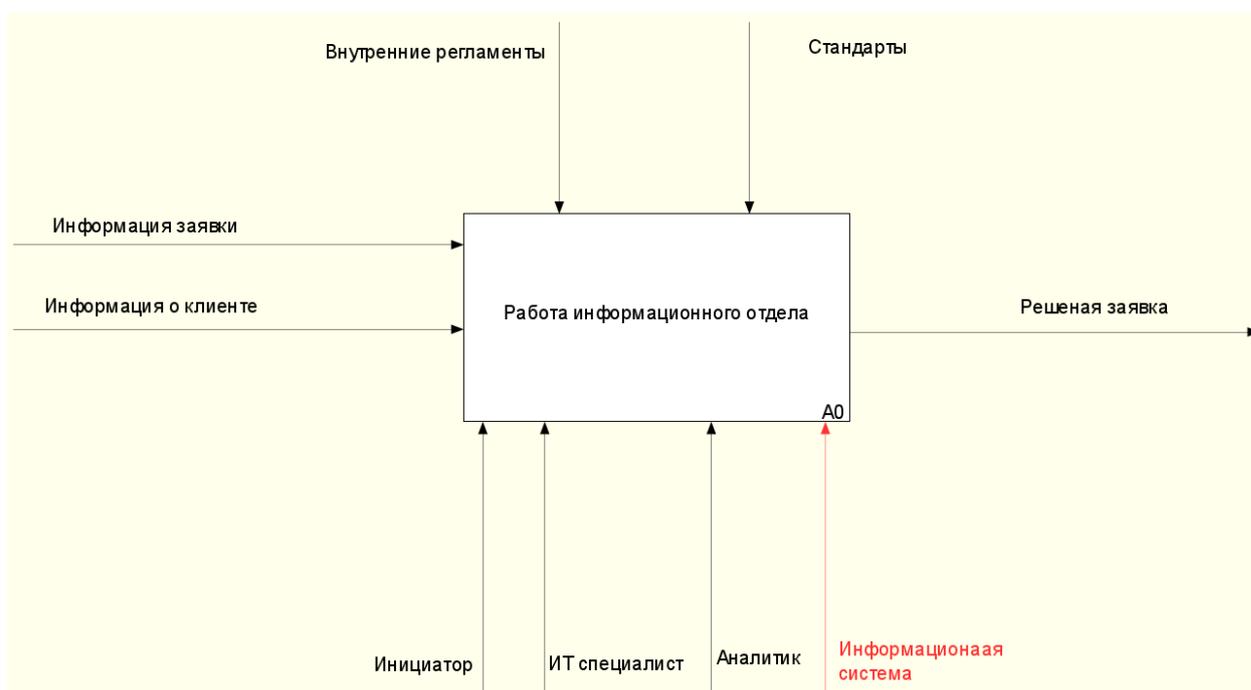


Рисунок 4 - Диаграмма IDEF0 КАК ДОЛЖНО БЫТЬ (Родитель)

На рисунке 5, показывающим декомпозицию организации в модели «КАК ДОЛЖНО БЫТЬ», предложено добавить новый механизм – Информационная система. Это дополнение, основанное на выявленных недостатках и анализе текущего состояния, будет способствовать внедрению автоматизации и повышению эффективности работы.

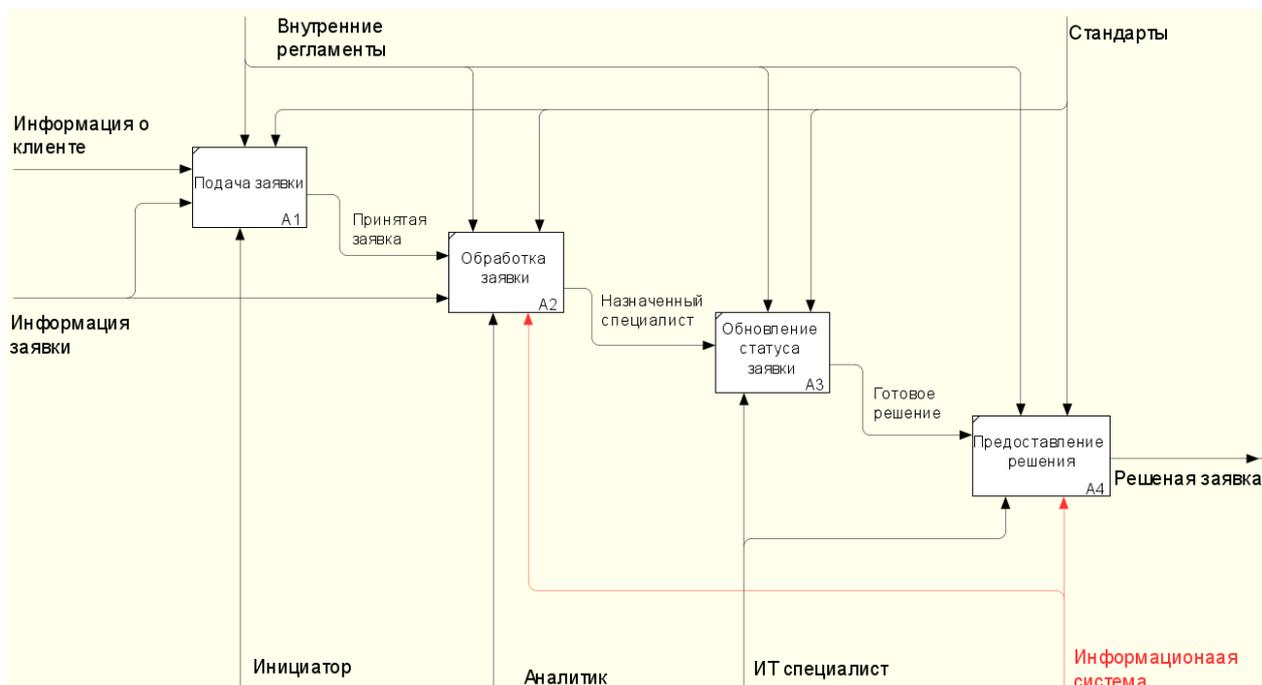


Рисунок 5 - Диаграмма IDEF0 КАК ДОЛЖНО БЫТЬ (Дочерняя)

На основе выявленных недостатков и анализа модели "КАК ЕСТЬ" предлагается внедрение нового механизма - Информационной системы, представленной на декомпозиции информационного отдела АО «Тольяттиазот». Это позволит автоматизировать процессы и повысить эффективность работы, предоставляя более оптимальные и современные методы выполнения бизнес-процессов.

### 1.5 Разработка требований к ПО

Для определения требований к информационной системе управления заявками информационного отдела используется методология FURPS+.

«FURPS+ - это метод, который помогает оценить приоритетные требования после того, когда были поняты потребности клиента.

Аббревиатура FURPS означает функциональность, удобство использования, надежность, производительность и возможность поддержки.

Методология FURPS+ классифицирует требования, учитывая различные типы функциональных и нефункциональных требований» [14].

«Функциональные требования определяют действия, которые система должна выполнять. В тоже время нефункциональные требования устанавливают условия и ограничения, в которых эти функции должны осуществляться.

Дополнительные спецификации охватывают различные аспекты, такие как юридические нормы, критерии качества, возможности поддержки и критерии производительности системы. Объем дополнительных спецификаций зависит от всех нефункциональных требований.» [4]

Таблица 2 содержит основные требования к информационной системе, учитывающей особенности методологии FURPS+.

«Рекомендации по программному и техническому обеспечению серверной части:

- операционные системы Linux, Windows Server;
- СУБД SQLite версии 3.36.0;
- веб-сервер Apache HTTP Server версии 2.4.58;
- процессор с минимальным числом ядер 8 (16 логических потоков), с тактовой частотой от 3,5 ГГц. и больше;
- минимальный объем HDD: 128 Гб и выше для документов (зависит от объема хранимых документов);
- оперативная память: минимум 64 Гб и выше;
- стабильный канал связи: минимальная скорость 10 Мб/сек.

В состав программного обеспечения клиентской части должны входить:

- операционная система Windows 10;
- обновленные версии следующих веб-браузеров: Yandex Browser, MS Internet Explorer, MS Edge, Google Chrome, Mozilla Firefox.

Техническое обеспечение клиентской части должно обеспечивать соответствие требованиям, установленным для указанного программного обеспечения.»[15]

Таблица 2– Требования к информационной системе

№	Требование	Статус	Полезность	Риск	Стабильность
Functionality — Функциональные требования					
1	Обработка заявок в режиме онлайн	Одобрено	Критическая	Средний	Средняя
Usability— Требования к удобству использования					
2	Дружественный интуитивный интерфейс	Одобрено	Критическая	Средний	Средняя
Reliability— Требования к надежности					
3	Допустимая частота/периодичность сбоев: 1 раз в 300 часов	Одобрено	Важная	Средний	Средняя
4	Среднее время сбоев: 1 раб. день	Одобрено	Важная	Средний	Средняя
5	Возможность восстановления системы после сбоев: 1 раб. день	Одобрено	Важная	Средний	Средняя
6	Режим работы: 24/7	Одобрено	Важная	Средний	Средняя
Performance — Требования к производительности					
7	Допустимое количество одновременно работающих пользователей: 500	Предложенное	Важная	Средний	Средняя
8	Время реакции на возникновение аварийной ситуации: 10 секунд	Предложенное	Важная	Средний	Средняя
№	Требование	Статус	Полезность	Риск	Стабильность
Supportability — Требования к поддержке					
9	Время устранения критических проблем: в течение рабочего дня	Предложенное	Важная	Средний	Средняя
Проектные ограничения					
10	Применение современных веб-технологий	Предложенное	Критическая	Средний	Средняя
11	Низкая стоимость владения	Предложенное	Критическая	Средний	Средняя

Разработанный список требований служит основой для разработки информационной системы управления заявками информационного отдела АО «Тольяттиазот».

## 1.6 Постановка задачи на разработку

Разработка информационной системы управления заявками имеет важную цель – это увеличение эффективности обработки новых заявок за определенную единицу времени. Такая система обеспечит сбор и анализ статистических данных о производительности сотрудников. Это позволит оптимизировать рабочие процессы, а также повысит мотивацию сотрудников.

Применение такой информационной системы позволит оптимизировать процессы принятия заявок и ускорит обработку записей. В результате, планируется увеличение уровня удовлетворенности за счет более оперативного выполнения запросов. Так же, система будет способствовать накоплению данных для анализа бизнес-процессов, что даст возможности для оптимизации и улучшения работы предприятия.

Выводы по главе 1.

В рамках первой главы выпускной работы было проведено исследование по вопросу изучения действующих в настоящий момент операций информационного отдела АО "Тольяттиазот" и бизнес-процессов, связанных с ними.

Используя методологию IDEF0, разработаны диаграммы «как есть», которые отображают текущие операционные процессы информационного отдела. Проведенный анализ выявил отсутствие автоматического уведомления специалистов информационного отдела о появлении новых задач, что приводит к не результативности выполнения поставленных задач в периоды повышенной нагрузки на отдел.

Действующая в настоящий момент модель бизнес-процессов не соответствует современным стандартам. Она заметно замедляет оперативность информационного отдела, что отрицательно сказывается на продуктивности и эффективности деятельности всей организации.

Для повышения эффективности работы информационного отдела, необходимо разработать информационную систему направленную на устранение выявленных недостатков существующей бизнес-системы.

На основе проведенного анализа принято решение: при разработки новой системы использовать высокоуровневый веб-фреймворк Django на языке программирования Python. [18]

На диаграммах «как должно быть» показаны изменения логической модели предметной области с учетом внедрения информационной системы, которые были выявлены в процессе исследования.

На основании вышеизложенного сформулирована задача для дипломной работы: разработка информационной системы управления заявками информационного отдела компании, которая отвечает всем требованиям АО «Тольяттиазот».

## **Глава 2 Проектирование системы управления заявками информационного отдела.**

### **2.1 Разработка логической модели автоматизированной системы управления заявками.**

Логическая модель информационной системы (ИС), представляющая собой ключевой инструмент, широко применяется специалистами для оценки эффективности программного продукта. «Логическое проектирование ИС включает набор диаграмм UML, которые графически отображают взаимосвязи между ресурсами, действиями, выводами и результатами программы.

Логическая модель ИС определяет:

- описание того, как должен функционировать проект разработки ИС;
- методы и принципы, на которых основана программа;
- связь результатов (как краткосрочных, так и долгосрочных) с системной деятельностью и процессами.

Для разработки диаграмм логической модели информационной системы управления (ИСУ) деятельностью информационного отдела используется методология проектирования RUP (Rational Unified Process) и CASE-средство IBM Rational Rose. На этапе разработки функциональных требований к системе в методологии RUP создается диаграмма вариантов использования.

Диаграмма вариантов использования представляет собой высокоуровневое представление требований, существующих в системе.

Диаграммы вариантов использования делают требования более доступными и понятными, позволяя быстро визуализировать то, что должна делать система» [5].

Анализ текущей модели бизнес-процессов информационной системы управления заявками в ИТ отделе компании выявил следующие роли: инициатор, ИТ специалист и аналитик.

«Ниже приведены возможные варианты использования, описанные в таблицах с 3 по 9 в рамках методологии RUP.

Таблица 3 – Описание прецедента: Авторизация/Регистрация

Прецедент: Регистрация/Авторизация
ID: 1
Краткое описание: Авторизация и/или регистрация пользователя
Главный субъект: Инициатор
Второстепенный субъект: ИТ специалист, Аналитик
Предусловие: Отсутствует
Основной поток: Пользователь авторизируется или регистрируется в ИСУ
Постусловие: Отсутствует
Альтернативные потоки: Отсутствуют

Таблица 4 – Описание прецедента: Оформление заявки

Прецедент: Оформление заявки
ID: 2
Краткое описание: оформление заявки с последующей отправкой в работу
Главный субъект: Инициатор
Второстепенный субъект: ИТ специалист, Аналитик
Предусловие: Отсутствует
Основной поток: Пользователь оформляет заявку
Постусловие: Отсутствует
Альтернативные потоки: Отсутствуют

Таблица 5 – Описание прецедента: Оформление заявки

Прецедент: Получение информации о заявке
ID: 3
Краткое описание: Получение подробной информации заявки
Главный субъект: ИТ специалист
Второстепенный субъект: Инициатор, Аналитик
Предусловие: Отсутствует
Основной поток: ИТ специалист изучает заявку
Постусловие: Отсутствует
Альтернативные потоки: Отсутствуют

Таблица 6 – Описание прецедента: Отслеживание статуса заявки

Прецедент: Отслеживание статуса заявки
ID: 4
Краткое описание: Отслеживание изменения статуса заявки
Главный субъект: Инициатор
Второстепенный субъект: ИТ специалист, Аналитик
Предусловие: Отсутствует
Основной поток: Инициатор отслеживает изменение статуса своей заявки
Постусловие: Отсутствует
Альтернативные потоки: Отсутствуют

Таблица 7 – Описание прецедента: Изменение статуса заявки

Прецедент: Изменение статуса заявки
ID: 5
Краткое описание: Изменение статуса заявки
Главный субъект: ИТ Специалист
Второстепенный субъект: Инициатор, Аналитик
Предусловие: Отсутствует
Основной поток: ИТ специалист изменяет статуса заявки, над которой работает
Постусловие: Отсутствует
Альтернативные потоки: Отсутствуют

Таблица 8 – Описание прецедента: Исполнение заявки

Прецедент: Исполнение заявки
ID: 6
Краткое описание: ИТ специалист выполняет задание из заявки
Главный субъект: ИТ Специалист
Второстепенный субъект: Инициатор, Аналитик
Предусловие: Отсутствует
Основной поток: ИТ специалист проводит работы для выполнения заявки
Постусловие: Отсутствует
Альтернативные потоки: Отсутствуют

Таблица 9 – Описание прецедента: Получение готового решения

Прецедент: Исполнение заявки
ID: 7
Краткое описание: ИТ специалист выполняет задание из заявки
Главный субъект: ИТ Специалист
Второстепенный субъект: Инициатор, Аналитик
Предусловие: Отсутствует
Основной поток: ИТ специалист проводит работы для выполнения заявки
Постусловие: Отсутствует
Альтернативные потоки: Отсутствуют

«Диаграмма вариантов использования ИСУ, изображенная на рисунке 6, показывает функциональные возможности системы».[19]

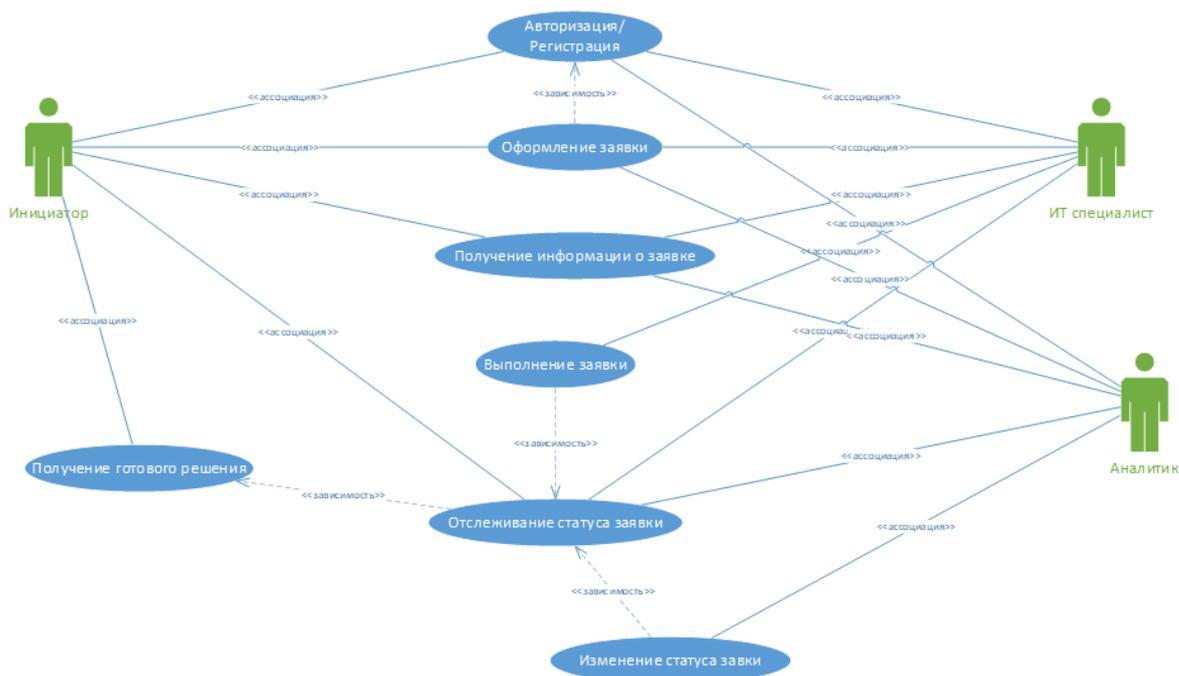


Рисунок 6 – Диаграмма вариантов использования

Для более лучшего понимания ключевого бизнес-процесса в данной области, была разработана диаграмма последовательностей, которая детально описывает этапы взаимодействия между участниками.

На рисунке 7 представлена диаграмма последовательностей основного бизнес-процесса. Диаграмма деятельности представляет собой графическое изображение рабочего процесса, разбитого на этапы. Данная диаграмма позволяет не только рассматривать различные варианты дальнейших действий, но и проследить последовательность шагов по отдельно выбранному конкретному варианту действий [2].



Рисунок 7 – Диаграмма последовательности

Диаграмма деятельности (рисунок 8) представляет последовательность действий, которые необходимы для успешного выполнения заявки. Важно отметить, что при оформлении заявки система проводит проверку на корректность заполнения формы. Это позволяет уменьшить вероятность ошибок при оформлении заявки [3].

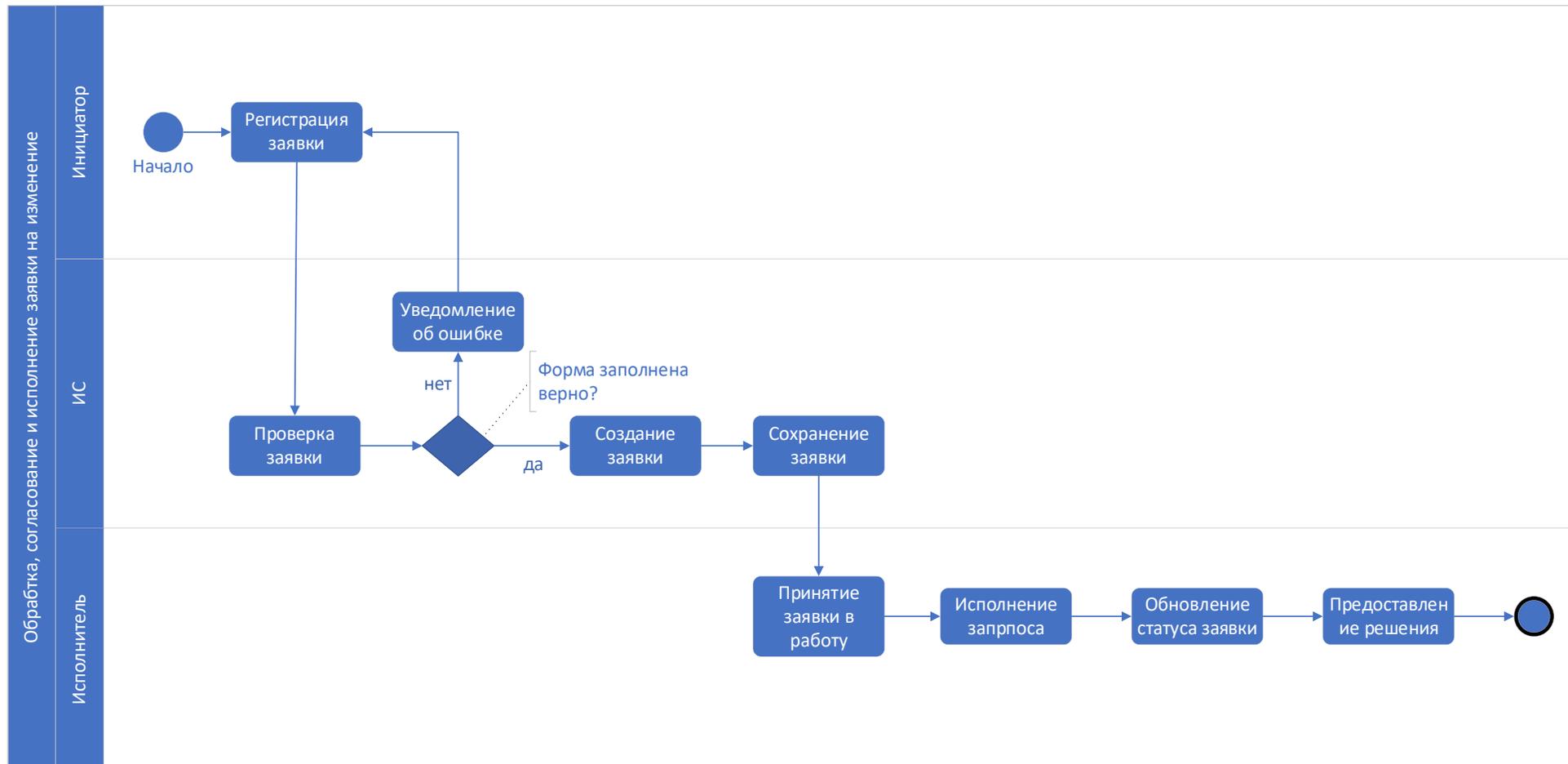


Рисунок 8 – Диаграмма деятельности

Диаграмма классов, изображенная на рисунке 9, описывает основные объекты и их взаимосвязи в системе. В ней показаны классы объектов, такие как исполнитель, инициатор и заявки. Исполнитель может просматривать информацию о заявках и менять их статус, в то время как инициатор может просматривать информацию о заявках и создавать новые заявки [1].

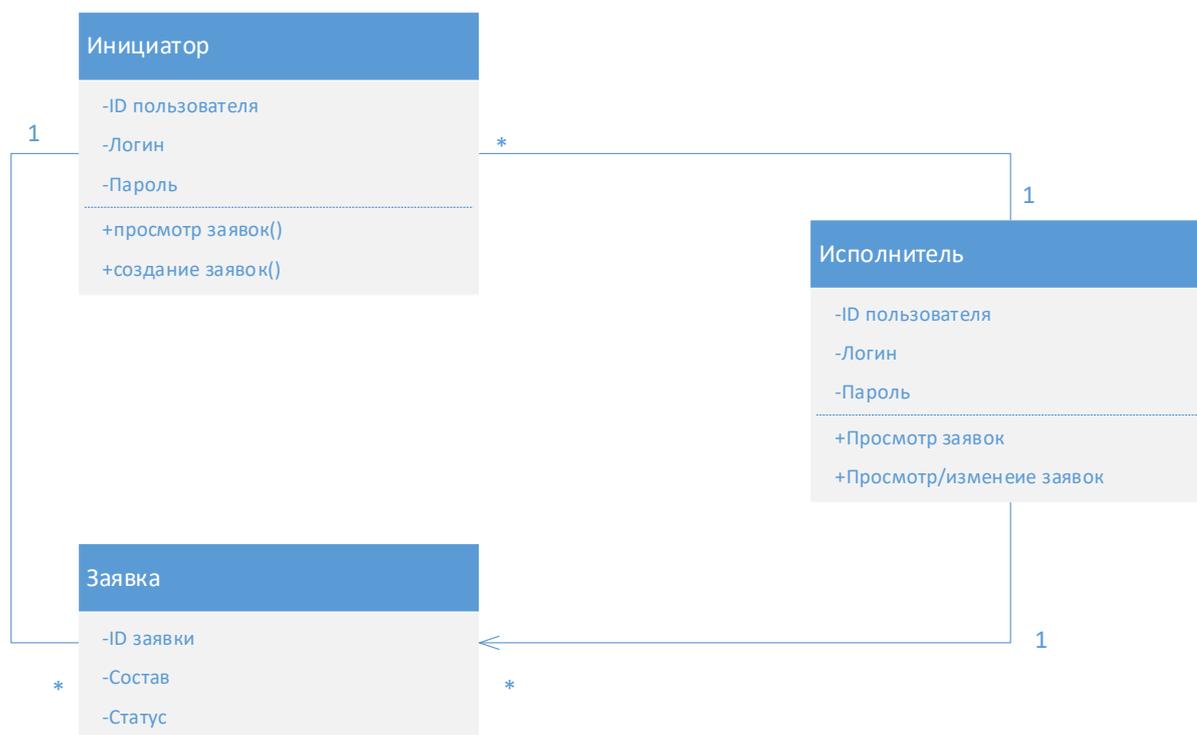


Рисунок 9 – Диаграмма классов

Эта диаграмма играет ключевую роль в понимании функциональности и взаимосвязей в информационной системе. Диаграмма помогает в дальнейшем проектировании и разработке системы с учетом всех ключевых аспектов.

## 2.2 Разработка логической модели данных автоматизированной системы управления заявками.

Исходя из поставленной задачи и информации, полученной из диаграммы логической модели, была разработана логическая модель данных, изображенная на рисунке 10.

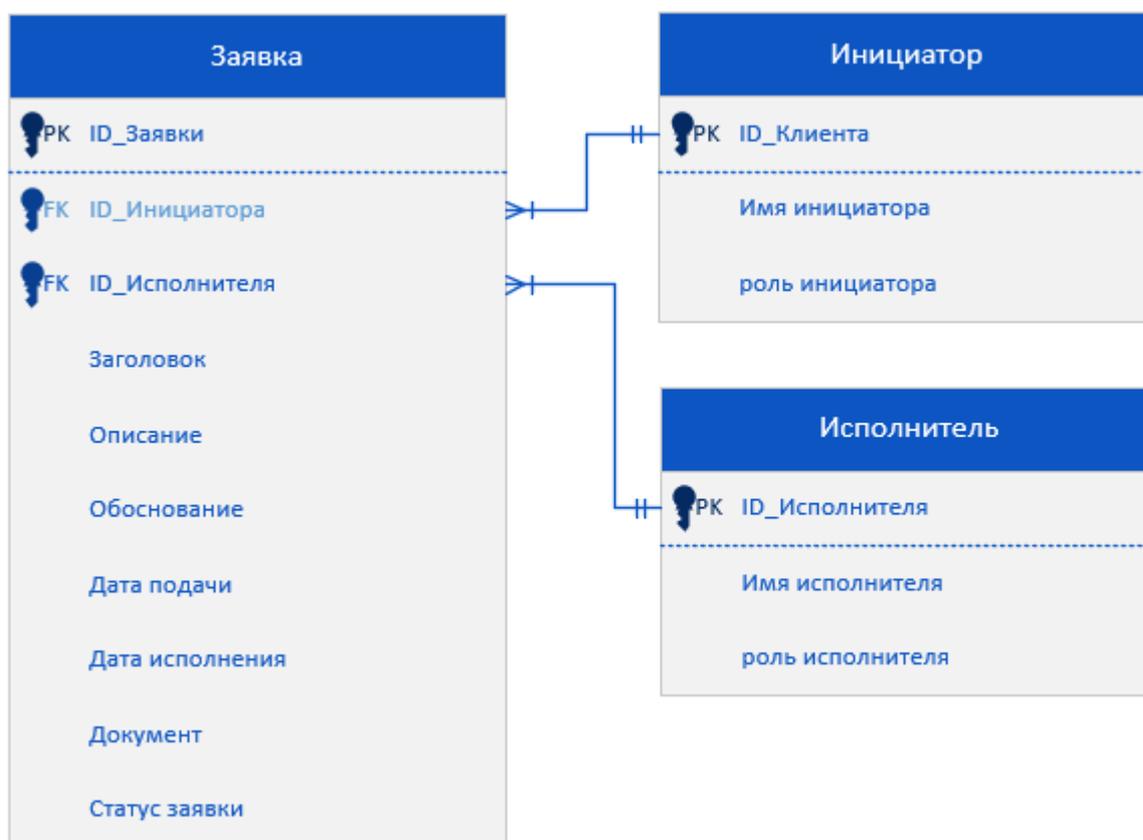


Рисунок 10 - Логическая модель данных

«Логическая модель данных показывает взаимоотношения между сущностями с помощью связей» [6].

### 2.3 Разработка физической модели данных автоматизированной ИСУ заявками.

«В соответствии с логической моделью данных была разработана физическая модель для базы данных SQLite. Физическое проектирование включает в себя аспекты хранения информации и способы доступа к ней.» [13]. Физическая модель данных для СУБД SQLite представлена на рисунке 11

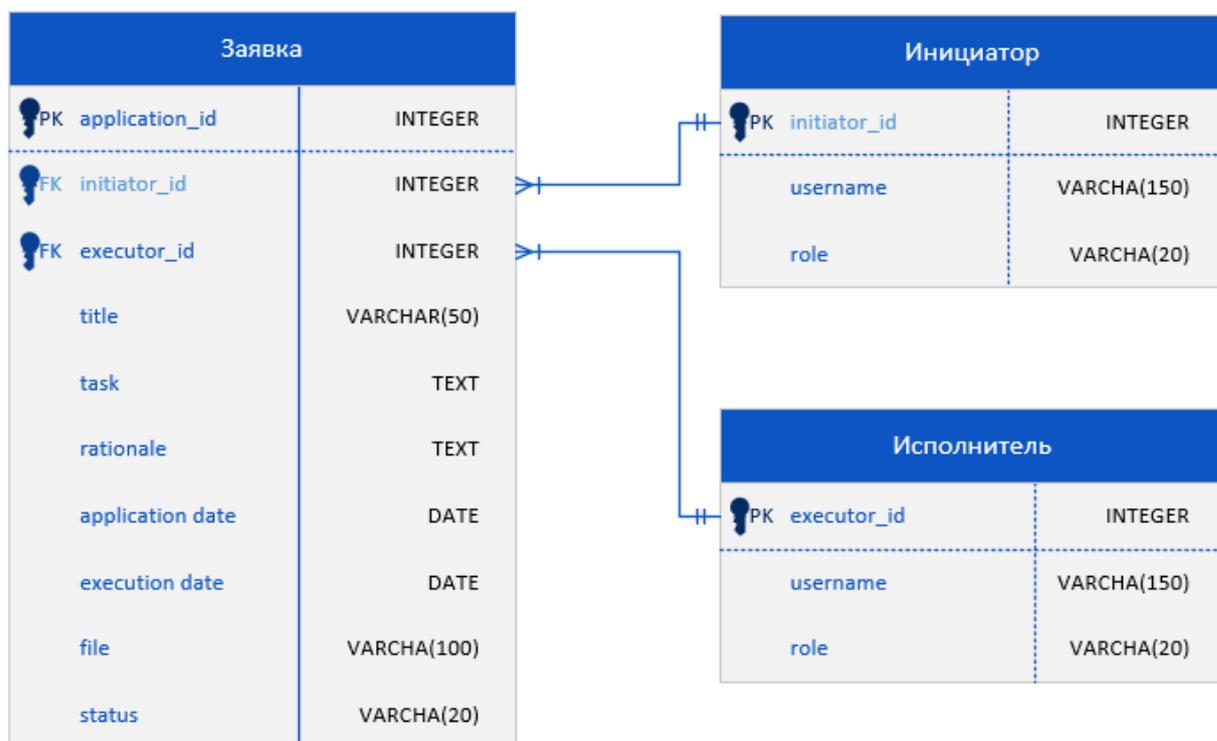


Рисунок 11 - Физическая модель данных

На этапе физического проектирования базы данных администратор определяет, как данные будут группироваться, какие индексы использовать и как обеспечить доступ к ним.

Также стоит учитывать особенности самой СУБД, включая ограничения на именование объектов и поддерживаемые типы данных.

## 2.4 Требования к аппаратно-программному обеспечению автоматизированной системы управления заявками информационного отдела организации

Информационная система управления заявками информационного отдела АО «Тольяттиазот» будет разрабатываться на высокоуровневом веб-фреймворке Django на языке программирования Python.

Минимальные системные требования для установки веб-приложения на сервере и компьютерах сотрудников включают следующее оборудование:

а) сервер:

- 1) процессор Intel Xeon Silver 4210R (16 ядер);
- 2) 64 Гб оперативной памяти ECC DDR4;
- 3) SSD-накопитель: 2 Тб.

б) персональные компьютеры:

- 1) процессор Intel Core i3-9100 (4 ядра);
- 2) материнская плата ASUS Prime B360M-K;
- 3) 16Гб оперативная память DDR4-2400;
- 4) встроенная графика Intel UHD 630;
- 5) SSD-накопитель Kingston A400 480Гб.

Операционные системы, поддерживаемые для установки, включают Microsoft Server 2022 или Windows 10.

Проанализировав имеющееся на балансе предприятия техническое оборудование и программное обеспечение в АО "Тольяттиазот", можно сделать вывод о соответствии технических возможностей компании требованиям для внедрения информационной системы. Приобретение дополнительного системного оборудования или программного обеспечения не требуется.

Выводы по главе 2.

Во второй главе дипломной работы была выполнена логическая разработка информационной системы. В этот процесс входило создание логических моделей для программного обеспечения и базы данных. Диаграмма вариантов использования помогла определить ключевых пользователей и функции системы, в то время как диаграмма деятельности детально представила последовательность действий для каждого сценария.

Для более детального описания этапов основного бизнес-процесса была создана диаграмма последовательностей, а диаграмма классов показала иерархическую структуру системы. Особое внимание уделялось анализу выходных данных, что являлось основной целью разработки системы.

Для проектирования базы данных применялась методология IDEF1X. С помощью концептуальной ER-модели были выделены важные объекты и их связи. Логическая модель данных описала объекты предметной области и их взаимодействия.

В ходе разработки были учтены современные методологии и лучшие практики в области проектирования информационных систем, что позволило создать прочную основу для последующей реализации. В рамках логического проектирования также были предусмотрены механизмы масштабирования и адаптации системы под потенциальные изменения в бизнес-процессах компании. Это обеспечит гибкость и устойчивость системы в долгосрочной перспективе, что является важным фактором для успешной эксплуатации информационной системы в динамично развивающейся компании

На основании результатов логического проектирования и учитывая наличие соответствующего оборудования и программного обеспечения в АО «Тольяттиазот», можно переходить к следующему этапу — физической реализации программного продукта.

## **Глава 3 Реализация проектных решений системы управления заявками информационного отдела организации и тестирование информационной системы управления.**

### **3.1 Выбор средств разработки**

Для реализации проекта системы управления заявками информационного отдела организации АО «Тольяттиазот» необходимо выбрать соответствующие средства разработки, такие как языки программирования, системы управления базами данных (СУБД) и фреймворки. «Рассмотрим различные варианты и выберем наиболее подходящие из них:

а) языки программирования:

1) Python:

- Python - высокоуровневый язык программирования, известный своей простотой и читаемостью кода. Он поддерживает множество библиотек и фреймворков для веб-разработки, научных вычислений и автоматизации;
- достоинства: легкость в освоении, большое сообщество, множество библиотек, высокая производительность при разработке прототипов. Python также обеспечивает быструю разработку и легкую интеграцию с другими системами;
- недостатки: меньшая скорость выполнения по сравнению с компилируемыми языками, такими как C++ или Java.

2) Java:

- Java - объектно-ориентированный язык программирования, широко используемый для разработки корпоративных приложений. Он обеспечивает высокую производительность и надежность;

- достоинства: кроссплатформенность, высокая производительность, большое количество инструментов и библиотек;
- недостатки: сложный синтаксис, длительное время компиляции и запуска.

### 3) JavaScript:

- JavaScript - язык программирования, преимущественно используемый для создания интерактивных веб-страниц. В сочетании с фреймворками, такими как Node.js, он также подходит для серверной разработки;
- достоинства: кроссплатформенность, высокая скорость выполнения, большое сообщество и множество библиотек;
- недостатки: возможные проблемы с безопасностью, динамическая типизация может приводить к ошибкам, сложность управления большими проектами.

## б) СУБД:

### 1) MySQL:

- MySQL - реляционная система управления базами данных с открытым исходным кодом. Она популярна благодаря своей надежности и производительности;
- достоинства: высокая производительность, поддержка ACID, широкое использование и поддержка;
- недостатки: ограниченная масштабируемость, сложность настройки репликации.

### 2) PostgreSQL:

- PostgreSQL - мощная, объектно-реляционная СУБД с открытым исходным кодом, известная своей надежностью и широкими функциональными возможностями;
- достоинства: поддержка сложных запросов, расширяемость, высокая надежность и масштабируемость;

- недостатки: сложная настройка и администрирование по сравнению с MySQL.

### 3) SQLite:

- SQLite - встраиваемая СУБД, не требующая отдельного серверного процесса. Она идеально подходит для небольших и встраиваемых приложений;
- достоинства: простота использования, минимальные требования к настройке, высокая производительность для небольших объемов данных. SQLite также поддерживает транзакции ACID, что обеспечивает надежность данных. Для большинства малых и средних проектов производительности SQLite достаточно, а при необходимости можно легко мигрировать на более мощную СУБД, такую как PostgreSQL;
- недостатки: ограниченные возможности параллельного доступа, не подходит для крупных проектов с высокой нагрузкой.

### в) фреймворки:

#### 1) Django (Python):

- Django - высокоуровневый веб-фреймворк на языке Python, который поощряет быстрое развитие и чистый, прагматичный дизайн;
- достоинства: полноценный фреймворк, высокая скорость разработки, встроенная панель администратора. Django также обладает мощными инструментами для работы с базами данных и масштабирования приложений. Для проектов средней сложности и выше Django предоставляет все необходимые инструменты для надежной и масштабируемой разработки;
- недостатки: монотонная архитектура, может быть избыточна для небольших проектов.

## 2) Spring (Java):

- Spring - мощный фреймворк для создания корпоративных приложений на языке Java. Он предоставляет все необходимое для создания надежных и масштабируемых приложений;
- достоинства: гибкость, поддержка инверсии управления, обширная экосистема;
- недостатки: крутая кривая обучения, сложность конфигурации.

## 3) Node.js (JavaScript):

- Node.js - среда выполнения для JavaScript, которая позволяет использовать JavaScript для серверной разработки;
- достоинства: высокая производительность, большое сообщество, асинхронная природа;
- недостатки: ограниченная поддержка многопоточности, возможные проблемы с управляемостью кода в больших проектах.

Ниже представлена таблица 10, в которой суммированы основные достоинства и недостатки рассмотренных языков программирования, систем управления базами данных и фреймворков. Это позволит наглядно сравнить их и обосновать выбор средств для разработки проекта.

Исходя из анализа, для разработки информационной системы управления заявками организации АО «Гольягтиазот» были выбраны следующие средства:

- язык программирования: Python;
- СУБД: SQLite;
- веб-фреймворк: Django.

Таблица 10 - Сравнение языков программирования, СУБД и фреймворков.

Средство	Достоинства	Недостатки
<b>Язык программирования</b>		
Python	Легкость освоения, большое сообщество, множество библиотек, быстрая разработка	Меньшая скорость выполнения
Java	Кроссплатформенность, высокая производительность	Сложный синтаксис, длительное время компиляции
JavaScript	Кроссплатформенность, высокая скорость выполнения	Возможные проблемы с безопасностью, сложность управления
<b>СУБД</b>		
MySQL	Высокая производительность, поддержка ACID	Ограниченная масштабируемость
PostgreSQL	Поддержка сложных запросов, высокая надежность	Сложная настройка и администрирование
SQLite	Простота использования, высокая производительность, поддержка ACID	Ограниченные возможности параллельного доступа
<b>Фреймворк</b>		
Django	Полноценный фреймворк, высокая скорость разработки, мощные инструменты для масштабирования	Монотонная архитектура
Spring	Гибкость, обширная экосистема	Крутая кривая обучения
Node.js	Высокая производительность, асинхронная природа	Ограниченная поддержка многопоточности

Эти средства были выбраны по следующим причинам:

- Python был выбран благодаря своей простоте, легкости освоения и большому количеству библиотек, что позволяет быстро разрабатывать и отлаживать код. Python также обеспечивает легкую интеграцию с другими системами и технологиями;
- SQLite идеально подходит для небольших и средних проектов благодаря своей простоте использования и высокой

производительности. Он не требует отдельного серверного процесса, что упрощает настройку и администрирование. При необходимости система может быть легко масштабирована и мигрирована на более мощную СУБД, такую как PostgreSQL;

- Django предоставляет мощные инструменты для быстрой и эффективной разработки веб-приложений. Он включает в себя все необходимые компоненты, что сокращает время разработки. Django также обеспечивает высокую надежность и масштабируемость, что важно для долгосрочного успеха проекта.»[12]

Выбранные средства разработки обеспечивают высокое качество и эффективность разрабатываемой системы, а также позволяют легко масштабировать систему при росте нагрузки и объема данных, что делает их оптимальными для разрабатываемого проекта.

### **3.2 Функциональность автоматизированной системы управления заявками.**

У каждого пользователя есть своя роль в информационной системе. Это позволяет разграничить обязанности и сделать интерфейс удобным и подстроенным под определенные задачи [17].

Так на главной странице, пользователь-администратор, (рисунок 12) находятся заявки от всех инициаторов и исполнителей. Чтобы было проще найти нужную заявку был создан фильтр по статусам заявок, по именам инициаторов и по именам исполнителей. Это значительно увеличивает производительность работы администратора. Он может отследить статус любой заявки, от любого инициатора, а также скорость выполнения этих заявок исполнителями.

## Заявки

Статус: Инициатор: Исполнитель: [Применить фильтр](#)

### Настройка уведомлений об изменениях в законодатель

Настройка автоматического уведомления по почте ил...

Инициатор: Александр\_Коротков

[Подробнее](#)

### Дубликаты в справочнике транспорта Логистической п

Реализовать проверку номеров вводимых транспортны...

Инициатор: Пётр\_Харитонов

Рисунок 12 - Главная страница. Вид администратора

На главной странице обычного пользователя (рисунок 13) находятся только заявки самого пользователя, что упрощает и оптимизирует отслеживание своих заявок.

## Заявки

Статус: Инициатор: Исполнитель: [Применить фильтр](#)

### Дубликаты в справочнике транспорта Логистической п

Реализовать проверку номеров вводимых транспортны...

Инициатор: Пётр\_Харитонов

[Подробнее](#)

### Фирменный стиль на сайте

Привести корпоративный сайт в соответствие с корп...

Инициатор: Пётр\_Харитонов

Рисунок 13 - Главная страница. Вид пользователя

На главной странице пользователя программиста (рисунок 14) находятся только заявки, которые должен выполнить программист. Это упрощает программистам поиск своих задач и оптимизирует процесс информирования программистов о новых заказах, которые они должны выполнить.

Пользователь: Александр\_Коротков (programmer)

[Главная](#) [Добавить запись](#) [Выйти](#)

## Заявки

Статус:

Инициатор:

Исполнитель:

[Применить фильтр](#)

### Дубликаты в справочнике транспорта Логистической п

Реализовать проверку номеров вводимых транспортны...

Инициатор: Пётр\_Харитонов

[Подробнее](#)

### Выпуск машиночитаемых доверенностей

Реализация автоматизированного выпуска МЧД внутри...

Рисунок 14 - Главная страница. Вид программиста

Так же установлен постоянный фильтр заявок по статусам, что сильно оптимизирует работу сотрудников, использующих данную ИС.

Записи со статусом «не назначен» находятся выше всех. Это статус означает, что задание поступило недавно и его необходимо взять в работу.

Задачи со статусом «в работе» находятся по середине. Данный статус означает, что заявка принята в работу и определен исполнитель.

Заявка со статусом «закрыт» находится ниже всех. Статус означает, что все необходимые работы были выполнены. Выполненные записи не нуждаются в особом внимании, так как вся необходимая работа была выполнена. Эти записи остаются для аналитиков.

Все авторизированные пользователи могут оставлять записи в веб-приложении (рисунок 15). Чтобы добавить запись, нужно заполнить все поля – название, описание, обоснование, ожидаемая дата реализации. Так же можно прикрепить файл, если это необходимо. После того, как инициатор нажмет на кнопку запустить в работу, его запись отобразится на главной странице у пользователей с ролями администратор, аналитик и у самого инициатора.

Пользователь: Пётр\_Харитонов (user)

[Главная](#) [Добавить запись](#) [Выйти](#)

## Создать запись

Наименование

Отчет BI (закупки ТМЦ)

Описание

Реализовать отчет в Power BI по прослеживанию процесса закупки от подачи потребности до поступления ТМЦ на склад и выдаче непосредственному заказчику

Обоснование

В целях контроля сроков исполнения потребностей, для расчета KPI отдела закупок

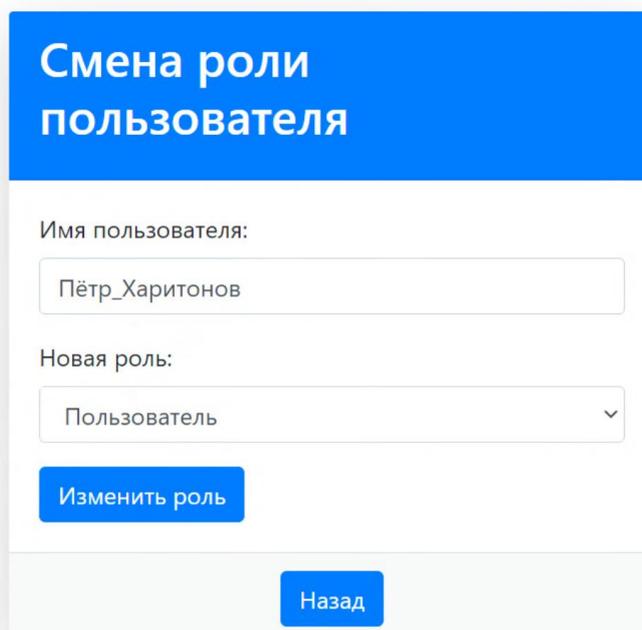
Файл:  Файл не выбран

Дата подачи заявки: 3 июня 2024 г.

Ожидаемая дата реализации:

Рисунок 15 – Страница создания заявок

На рисунке 16 представлена страница для смены роли пользователя. Эта страница доступна только администратору. Она необходимо для назначения ролей пользователям. У каждой роли есть свои привилегии и уровни доступа.



**Смена роли  
пользователя**

Имя пользователя:

Новая роль:

[Изменить роль](#)

[Назад](#)

Рисунок 16 – Страница смены роли пользователя

Аналитика необходима для эффективного управления ресурсами [16]. При помощи анализа данных о количестве заявок в различных статусах и учета числа инициаторов и исполнителей можно получить ценную информацию о загруженности, эффективности обработки заявок и распределении рабочей нагрузки (рисунок 17).

## Аналитика заявок

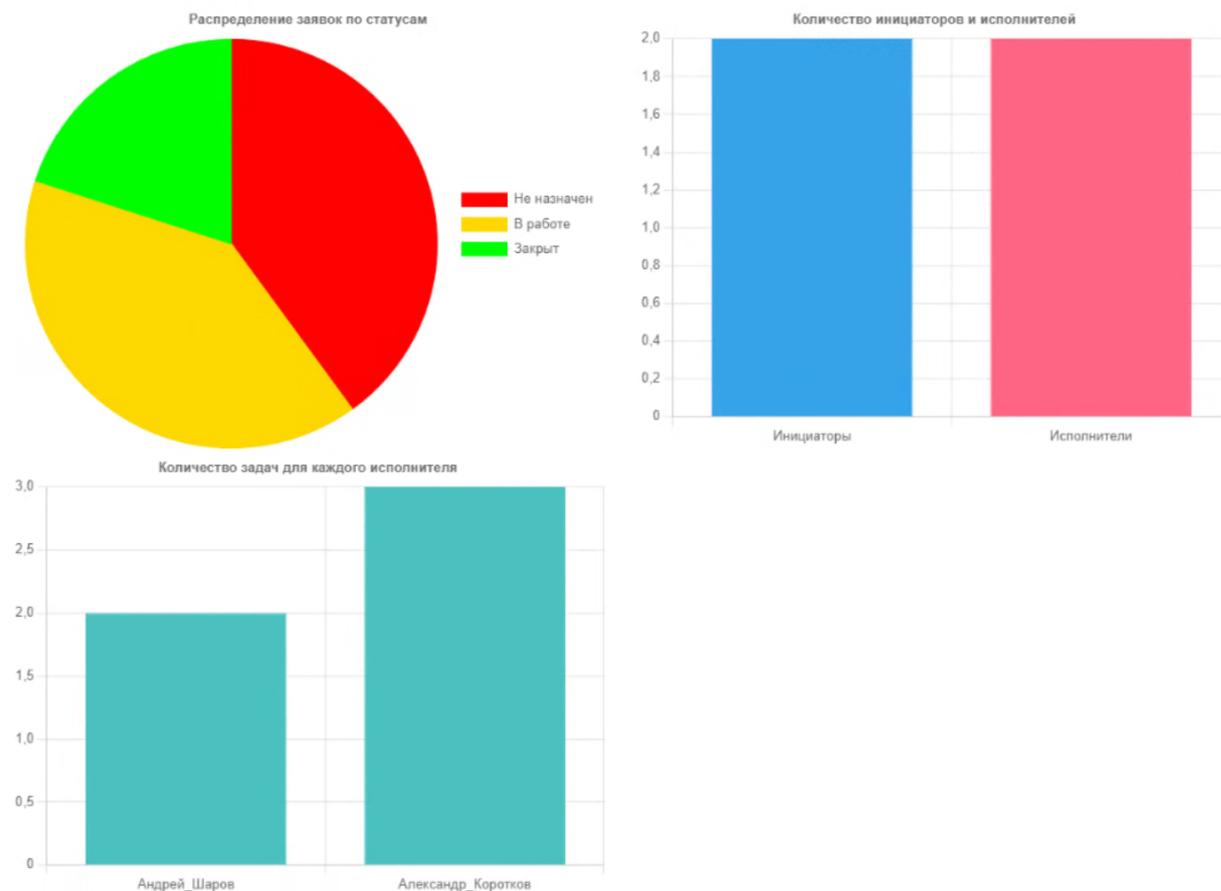


Рисунок 17 – Аналитика

На рисунке 18 представлена регистрация в веб приложении. Для регистрации нужно ввести имя пользователя, почту и пароль. После заполнения всех полей, необходимо нажать на зеленую кнопку «Зарегистрироваться», чтобы завершить регистрацию нового пользователя.

# Регистрация

---

Имя пользователя:

Обязательное поле. Не более 150 символов. Только буквы, цифры и символы @/./+/-/\_.

Email:

Пароль:

- Пароль не должен быть слишком похож на другую вашу личную информацию.
- Ваш пароль должен содержать как минимум 8 символов.
- Пароль не должен быть слишком простым и распространенным.
- Пароль не может состоять только из цифр.

Подтверждение пароля:

Для подтверждения введите, пожалуйста, пароль ещё раз.

Рисунок 18 – Регистрация

Рисунок 19 показывает окно входа в систему. Если у пользователя уже есть аккаунт в веб-приложении, то клиент может заполнить поля авторизации и пользоваться веб-приложением в полной мере.

Пользователь: AnonymousUser ()

---

# Вход

---

Имя пользователя:

Пароль:

[Забыл пароль](#)

Рисунок 19 – Вход в систему

В случае, если пользователь забыл пароль для входа в систему, то предусмотрена функция сброс пароля (рисунок 20). Чтобы восстановить пароль, необходимо ввести почту, которую клиент вводил при регистрации. После чего клиенту придет письмо на указанную почту, с ссылкой для восстановления пароля. Данная функция сбрасывает старый пароль, чтобы пользователь мог ввести новый и продолжить пользоваться своей учетной записью.

# Сброс пароля

Адрес электронной почты:

[Сбросить](#)

Рисунок 20 – Сброс пароля

Описание функциональных возможностей подтверждает обоснованность применения разработанного проектного решения для автоматизации управленческих процессов в информационной системе.

### **3.3 Тестирование информационной системы управления заявками АО «Тольяттиазот».**

Тестирование программного обеспечения представляет собой процесс исследования, направленный на предоставление заинтересованным лицам сведений о качестве проверяемого продукта или услуги. Это исследование также может дать объективную независимую оценку программного обеспечения, что позволит компаниям оценить и понять риски, связанные с его внедрением

«Методы тестирования включают процесс выполнения программы или приложения с целью обнаружения ошибок программного обеспечения и проверки пригодности программного продукта для использования. Для тестирования информационной системы управления заявками в компании АО «Тольяттиазот» был разработан следующий сценарий:

Программа и методика тестирования ИСУ заявками АО «Тольяттиазот»:

а) Объект тестирования:

Объектом тестирования является программное обеспечение информационной системы управления заявками в АО

«Тольяттиазот». Состав предоставляемого для тестирования программного обеспечения должен обеспечивать проведение тестирования в полном объеме согласно настоящей программе и соответствовать принятой стратегии тестирования.

б) Цель тестирования:

Целью тестирования ИСУ является:

- 1) Проверка программного обеспечения ИСУ на соответствие утвержденному проекту разработки и внедрения системы.
- 2) Проверка работоспособности программы и выявление возможных ошибок. » [8]

«В качестве дополнительных результатов тестирования могут рассматриваться следующие:

- 1) Качество интерфейса пользователя ИСУ.
- 2) Качество информационного обмена между отдельными модулями ИСУ.

в) Общие положения:

Настоящая программа и методика тестирования программного обеспечения (ПО) ИСУ предназначены для разработчика ПО и инженера по эксплуатации ИСУ для проведения ими тестирования ПО ИСУ. Тестирование ПО ИСУ проводится на территории компании АО «Тольяттиазот». Тестирование ПО ИСУ проводит разработчик при участии инженера по эксплуатации ИСУ.

г) Методика тестирования:

Тестирование ПО ИСУ проводится по методу бета-тестирования (beta testing). Бета-тестирование - это пользовательское приемочное тестирование или тестирование конечных пользователей. На этом этапе разработки ПО приложения проходит тестирование в реальном времени целевой аудиторией программного обеспечения. Опыт ранних пользователей перенаправляется обратно разработчикам,

которые вносят окончательные изменения перед выпуском программного обеспечения на коммерческой основе.»[7]

### Сценарий тестирования

Цель: Проверка работоспособности и функциональности информационной системы управления заявками.

– Вход в систему

Пользователь открывает приложение и вводит свои учетные данные для входа в систему (рисунок 21).

Пользователь: AnonymousUser ()

[Главная](#) [Регистрация](#) [Войти](#)

## Вход

Имя пользователя:

Пароль:

[Войти](#)

[Забыл пароль](#)

Рисунок 21 - Вход в систему

– Сброс пароля

Если пользователь забыл пароль, он использует функцию сброса пароля. Для этого вводит свой email, на который приходит письмо со ссылкой для восстановления пароля (рисунок 22).

## Сброс пароля

Адрес электронной почты: 

Рисунок 22 - Сброс пароля

### – Создание новой заявки

Пользователь создает новую заявку, заполняя необходимые поля (наименование, описание, обоснование и т.д.) (рисунок 23).

## Создать запись

Наименование

Описание

Обоснование

Файл:  Файл не выбран

Дата подачи заявки: 1 июня 2024 г.

Ожидаемая дата реализации:  

Рисунок 23 - Создание новой заявки

### – Просмотр списка заявок

Пользователь просматривает список всех заявок, отсортированных по различным критериям (статус, инициатор, исполнитель) (рисунок 24).

## Заявки

Статус: Инициатор: Исполнитель: 


### Фирменный стиль на сайте

Привести корпоративный сайт в соответствие с корп...

Инициатор: Пётр\_Харитонов

[Подробнее](#)

Рисунок 24 - Просмотр списка заявок

По результатам тестирования составляется протокол по установленной форме (таблица 11).

Таблица 11 - Протокол тестирования ИСУ заявками АО «Тольяттиазот»

№ этапа	Описание	Результат тестирования
1	Вход в систему	Успешный вход
2	Проверка компонентов интерфейса	Без замечаний
3	Сброс пароля	Сброс пароля выполнен успешно
4	Создание новой заявки	Заявка создана успешно
5	Просмотр списка заявок	Все заявки отображаются корректно
6	Проверка работоспособности ПО	Программное обеспечение работает корректно и стабильно

Тестирование разработанного модуля информационной системы управления заявками для АО «Тольяттиазот» показало его полную функциональность. При интеграции с другими модулями корпоративной информационной системы компании не было обнаружено ошибок, что подтверждает его успешную работоспособность

Выводы по главе 3.

В третьей главе была реализована проектная разработка информационной системы управления заявками для информационного отдела АО «Тольяттиазот». Были выбраны наиболее подходящие средства разработки, такие как языки программирования и системы управления базами данных, что позволило создать функциональную систему с разграничением ролей пользователей. Внедрение системы привело к значительному улучшению процесса управления заявками, обеспечив более высокую производительность и контроль над рабочим процессом. Проведенные тестирования подтвердили эффективность и надежность разработанного решения. Таким образом, достигнуты все поставленные цели, что подтверждает успех выполнения проектных решений и их внедрение в производственную среду организации.

## Заключение

В ходе выполнения данной выпускной квалификационной работы была разработана информационная система управления заявками информационного отдела АО «Тольяттиазот». Проведенные исследования и разработки позволили достичь поставленных целей и решить обозначенные задачи.

Для достижения цели работы были выполнены следующие задачи:

- проведен анализ предметной области, определены основные процессы и требования к системе. Были проанализированы текущие методы управления заявками и выявлены их недостатки, что позволило определить направления для улучшения;
- на основе анализа предметной области были разработаны логические и физические модели данных, а также модели бизнес-процессов. Система была спроектирована с учетом требований к функциональности и надежности.

Разработанная система была успешно реализована и интегрирована в инфраструктуру компании. Тестирование показало высокую работоспособность и соответствие всех модулей заявленным требованиям. Система доказала свою эффективность в управлении заявками, улучшая оперативность и качество обработки информации.

Практическая значимость работы заключается в возможности использования разработанной системы в реальных условиях, чтобы повысить эффективности работы информационного отдела. Внедрение системы позволит оптимизировать процессы управления заявками, сократить время на обработку запросов и улучшить взаимодействие между сотрудниками отдела.

Результаты работы представляют интерес как для научного сообщества, занимающегося вопросами автоматизации бизнес-процессов, так и для практиков, внедряющих информационные системы в крупных организациях.

Выявленные в ходе работы нюансы и подходы могут быть полезны для дальнейших исследований и разработок в данной области.

Цели и задачи выпускной квалификационной работы были успешно достигнуты. Разработанная информационная система управления заявками информационного отдела АО «Тольяттиазот» соответствует всем предъявленным требованиям и может быть рекомендована к внедрению.

В ходе работы было выявлено несколько важных аспектов, которые могут стать основой для будущих улучшений. В частности, дальнейшая оптимизация интерфейса системы может значительно повысить удобство её использования для конечных пользователей. Также возможно расширение функциональности системы за счет интеграции с другими корпоративными системами и внедрения дополнительных аналитических инструментов. Эти направления могут стать предметом дальнейших исследований и разработок.

Таким образом, результаты данной работы подтверждают, что разработанная система способна существенно улучшить процессы управления заявками в информационном отделе АО «Тольяттиазот», обеспечивая более высокую производительность и контроль над рабочим процессом.

## Список используемой литературы и используемых источников

1. Диаграмма деятельности UML (activity diagram). Руководство с примерами. [Электронный ресурс]. URL: [https://itonboard.ru/analysis/664-diagramma\\_deyatelnosti\\_rukovodstvo\\_dlja\\_nachinajushhih/](https://itonboard.ru/analysis/664-diagramma_deyatelnosti_rukovodstvo_dlja_nachinajushhih/)
2. Диаграммы деятельностей [Электронный ресурс]. URL: <https://studfile.net/preview/4418072/page:2/>
3. Использование диаграммы классов UML при проектировании и документировании программного обеспечения [Электронный ресурс]. URL: <https://habr.com/ru/articles/572234/>
4. Краткое описание нотации BPMN [Электронный ресурс]. URL: <https://habr.com/ru/companies/auriga/articles/667084/>
5. Леоненков А. В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose: учебное пособие. М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2020. 317 с.
6. Логическая модель предметной области [Электронный ресурс]. URL: <http://analyst.by/diagrams/logicheskaya-model-predmetnoy-oblasti>
7. Майерс, Гленфорд Дж. Искусство тестирования программ. Пер. с англ. – 3-е изд., испр. и доп. – М.: Вильямс, 2011. – 400 с.
8. Методы тестирования и отладки программного обеспечения / Е. Е. Карпович. - М. : Изд. Дом НИТУ «МИСиС», 2020. - 136 с.
9. Организационная структура: обзор популярных схем [Электронный ресурс]. URL: <https://www.unicraft.org/blog/7626/organizacionnaya-structura/>
10. Программа Ramus [Электронный ресурс]. URL: <https://ru.freedownloadmanager.org/Windows-PC/Ramus-Educational.html>
11. Репин В. В. Моделирование бизнес-процессов в нотации BPMN. Пособие для начинающих. М. : Перо, 2018. 80с.

12. Романенко В. В., Елизаров А. И. Основы разработки программного обеспечения : учебное методическое пособие / В. В. Романенко, А. И. Елизаров. – Томск : ФДО, ТУСУР, 2021. – 91 с.

13. Создание физической модели данных [Электронный ресурс]. URL: <https://kazedu.com/referat/98566/7>

14. С чего начать работу над новым проектом? Проектирование приложений: FURPS, Use Case, User Story... [Электронный ресурс]. URL: <https://egorzimowski.com/2022/11/09/how-start-program-design-use-cases-and-requirements/>

15. Стандарты и шаблоны для ТЗ на разработку ПО [Электронный ресурс]. URL: <https://habr.com/ru/articles/328822/>

16. How to create an analytics dashboard in a Django app [Электронный ресурс]. URL: <https://www.freecodecamp.org/news/how-to-create-an-analytics-dashboard-in-django-app/>

17. Multi-Role User Authentication in Django Rest Framework [Электронный ресурс]. URL: <https://dev.to/forhadakhan/multi-role-user-authentication-in-django-rest-framework-3nlp>

18. ONLINE ATTENDANCE WITH PYTHON FACE RECOGNITION AND DJANGO FRAMEWORK [Электронный ресурс]. URL: <https://elibrary.ru/item.asp?id=63540524>

19. Rational Unified Process - это легко: Руководство по RUP для практиков. / П. Кролл, Ф. Крачтен - М : КУДИЦ-Образ. 2004. 432 с.

20. UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition) Martin Fowler / England : 2003. 208p