**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФГБОУ ВО «ЗАПОЛЯРНЫЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ  
ФЕДОРОВСКОГО"  
Факультет электроэнергетики экономики и управления  
Кафедра информационных систем и технологий**

**Кафедра**  «Информационные системы и технологии»

                                                                      УТВЕРЖДАЮ:

                                                                  Зав. кафедрой

                                                                                               \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

                                                                                                   (подпись, фамилия и инициалы)

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к курсовому проекту (работе) по дисциплине

«Технологии программирования» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

         на тему: «Мобильное приложение системы заявок НМЗ XIS»

Автор курсового проекта (работы)     Шикан А.В.

                                                                                                                                                (подпись, дата, фамилия и инициалы)

Направление подготовки (специальности) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_09.03.02\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(код, наименование)

Обозначение курсового проекта (работы)     Группа ИС-21 \_\_\_\_\_\_\_\_\_\_\_

Руководитель проекта (работы) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_    \_\_\_\_\_\_Доцент Беляев И. С.\_\_\_\_\_\_

                                                                     (подпись, дата)                                        (фамилия, инициалы)

Проект (работа) защищен(а) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Оценка \_\_\_\_\_\_\_\_\_\_\_\_

                                                                                                              (дата)

Члены комиссии : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

                                                        (подпись, дата)                                               ( фамилия, инициалы)

                                          \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

                                                        (подпись, дата)                                               ( фамилия, инициалы)

                                          \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

                                                        (подпись, дата)                                               ( фамилия, инициалы)

Норильск, 2025

**Оглавление**

[ВВЕДЕНИЕ 3](#_Toc187767824)

[Глава 1. АКТУАЛЬНОСТЬ РАЗРАБОТКИ СИСТЕМЫ ЗАЯВОК 5](#_Toc187767825)

[1.1 Анализ предметной области 5](#_Toc187767826)

[1.2 Анализ диаграмм IDEF0 (AS-IS) 6](#_Toc187767827)

[1.3 Анализ диаграмм DFD (AS-IS) 12](#_Toc187767828)

[1.4 Патентный поиск 14](#_Toc187767829)

[1.5 Вывод по главе 17](#_Toc187767830)

[Глава 2. ИНФОЛОГИЧЕСКОЕ МОДЕЛИРОВАНИЕ 19](#_Toc187767831)

[2.1 Формирование и утверждение документа по стратегии 19](#_Toc187767832)

[2.2.1 Рабочее резюме 19](#_Toc187767833)

[2.2.2 Описание старой системы 19](#_Toc187767834)

[2.2.3 Взаимодействующие проекты 19](#_Toc187767835)

[2.2.4 Деловое и финансовое субсидирование 19](#_Toc187767836)

[2.2.5 Побуждения 19](#_Toc187767837)

[2.2.6 Границы проекта 20](#_Toc187767838)

[2.2.7 Рабочий план проекта 20](#_Toc187767839)

[2.2.8 Заключение 20](#_Toc187767840)

[2.3 Анализ диаграммы IDEF0 (TO-BE) 21](#_Toc187767841)

[2.4 Построение диаграмм DFD (TO-BE) 28](#_Toc187767842)

[2.5 Вывод по главе 31](#_Toc187767843)

[Глава 3. ПРОЕКТИРОВАНИЕ СТРУКТУРЫ БАЗЫ ДАННЫХ СИСТЕМЫ ЗАЯВКИ НМЗ XIS 33](#_Toc187767844)

[3.1 Логическая модель 33](#_Toc187767845)

[3.2 Физическая модель 35](#_Toc187767846)

[3.3 Инфологическое проектирование БД 37](#_Toc187767847)

[3.4 Бизнес-процесс 41](#_Toc187767848)

[3.5 Архитектура 44](#_Toc187767849)

[3.6 Алгоритмы 47](#_Toc187767850)

[Глава 4. ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА САЙТА СИСТЕМЫ ЗАЯВОК НМЗ XIS 50](#_Toc187767851)

[4.1 Мобильное приложение системы заявок НМЗ XIS 50](#_Toc187767852)

[БИБЛИОГРАФИЧЕСКИЙ СПИСОК 61](#_Toc187767853)

[ПРИЛОЖЕНИЕ 63](#_Toc187767854)

# **ВВЕДЕНИЕ**

В данной курсовой работе представлена разработка мобильного приложения, призванного автоматизировать и оптимизировать процесс подачи и обработки заявок на модернизацию системы «НМЗ XIS». Созданное приложение отличается удобным и интуитивно понятным интерфейсом, который обеспечивает эффективное взаимодействие пользователей с системой.

Инициатором разработки выступил отдел АСУ ТП, отметивший необходимость решения проблемы ручной обработки заявок. Новое программное обеспечение нацелено на упрощение рабочих процессов, повышение их производительности и снижение временных затрат.

В основе этого инструмента лежит созданная база данных, которая обеспечивает не только хранение заявок на модернизацию системы НМЗ XIS, но и их структурирование для более оперативного анализа и принятия решений. Разработанное приложение также предоставляет пользовательский интерфейс, способствующий более удобному взаимодействию с системой и данными.

В результате внедрения данного программного решения предполагается существенное улучшение эффективности управления процессом подачи и обработки заявок на модернизацию системы "НМЗ XIS". Создание данного приложения открывает новые перспективы и подходы к работе при модернизации системы, повышая степень автоматизации и оптимизируя процессы работы персонала.

Целью данной курсовой работы является создание мобильного приложения для обработки заявок для модернизации системы «НМЗ XIS», что позволяет экономить средства, рабочую силу и время для управления заявками.

Нашими задачами данной курсовой работы являются:

1. Изучение текущей системы НМЗ XIS для возможности интеграции нашего мобильного приложения.
2. Моделирование будущего мобильного приложения для управления заявками.
3. Проектирование и разработка базы данных с учетом потребностей модернизации заявок.
4. Разработка серверной части (бэкенд) мобильного приложения с использованием Python.
5. Разработка клиентской части мобильного приложения с использованием React Native.
6. Настройка контейнеризации backend мобильного приложения с использованием Docker.
7. Интеграция MinIO для управления и хранения данных.
8. Проведение тестирования и отладки созданного мобильного приложения.
9. Подготовка документации для пользователей и разработчиков.

# **Глава 1. АКТУАЛЬНОСТЬ РАЗРАБОТКИ СИСТЕМЫ ЗАЯВОК**

## **Анализ предметной области**

Разработка мобильного приложения для автоматизации процесса подачи и обработки заявок на модернизацию системы НМЗ XIS является важным шагом в оптимизации работы сотрудников и улучшении качества управления заявками. В текущий момент на Надеждинском металлургическом заводе им. Б. И. Колесникова наблюдается необходимость в автоматизации этого процесса, так как все работы по обработке заявок осуществляются вручную, что приводит к ряду проблем: высокой трудоемкости, временным затратам и вероятности ошибок.

Ручная обработка заявок на модернизацию системы НМЗ XIS сопровождается рядом значительных трудностей. Высокие временные и трудовые затраты на обработку заявок приводят к снижению эффективности работы сотрудников. Риск возникновения ошибок и неточностей при обработке данных затрудняет принятие правильных решений.

Для решения этих проблем требуется система, которая позволит автоматизировать процесс подачи и обработки заявок, обеспечивая при этом высокую надежность и удобство использования. Система должна поддерживать авторизацию пользователей, автоматизированное принятие заявок, работу с самими заявками.

В проектируемой системе ключевыми сущностями являются роли пользователей, данные пользователей, заявки, вложения и услуги в заявке. Роли пользователей определяют права доступа и возможности взаимодействия с системой. Данные пользователей включают личную информацию и контактные данные, необходимые для авторизации и идентификации. Заявки представляют собой запросы на получение определенных услуг, содержат описание проблемы или услуги, сроки выполнения и текущий статус. Вложения включают файлы и документы, которые прикрепляются к заявкам для предоставления дополнительной информации. Услуги описывают различные действия и операции, которые система может выполнить. Услуги в заявке детализируют запросы пользователей и указывают конкретные услуги, которые они хотят получить.

Архитектура мобильного приложения была реализована с использованием React Native, и бэкенд, развернутый в кластере Docker контейнеров с использованием Python. Данные хранятся в PostgreSQL, что обеспечивает надежность и масштабируемость, а для управления большими объемами неструктурированных данных используется MinIO.

Анализ предметной области позволяет выявить текущие проблемы и требования к новой системе. Разработка мобильного приложения для системы заявок НМЗ XIS обеспечит значительное улучшение качества и эффективности работы, сократит время и усилия, необходимые для управления заявками, и повысит оперативность в принятии решений.

## **Анализ диаграмм IDEF0 (AS-IS)**

Для описания модели процесса обработки заявок в данной работе используются диаграммы бизнес-процессов IDEF0 модели AS-IS. Нотация IDEF0 (функциональная модель) представляет собой совокупность иерархических, упорядоченных и взаимосвязанных диаграмм, каждая из которых является единицей описания системы, где отличительной особенностью является её акцент на соподчинённость объектов.

Контекстная диаграмма (Рис. 1.1) представляет собой исследуемый объект предметной области под названием «Текущая система обработки заявок HM3 XIS».

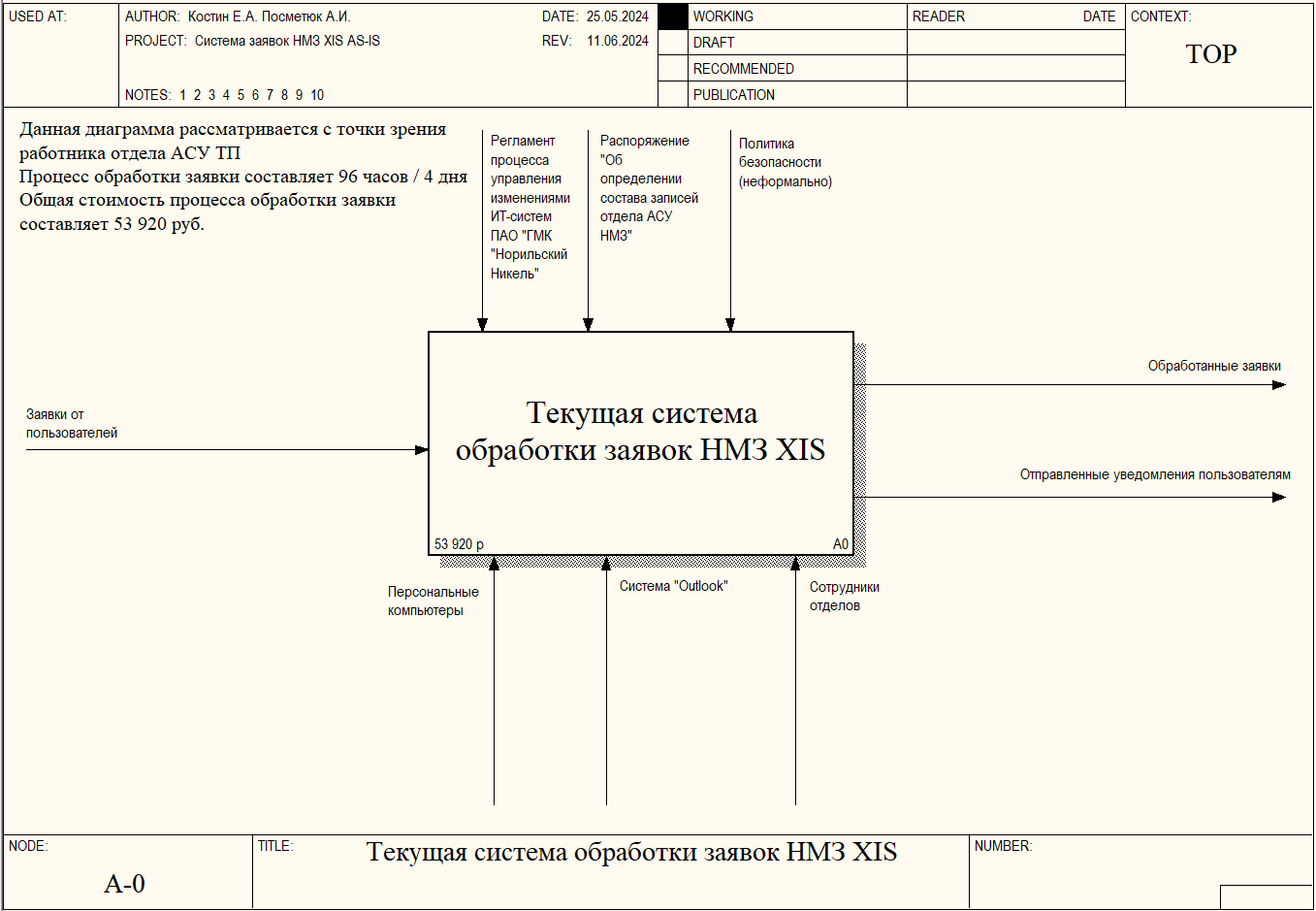


Рисунок 1.1 – Диаграмма IDEF0 AS-IS A-0

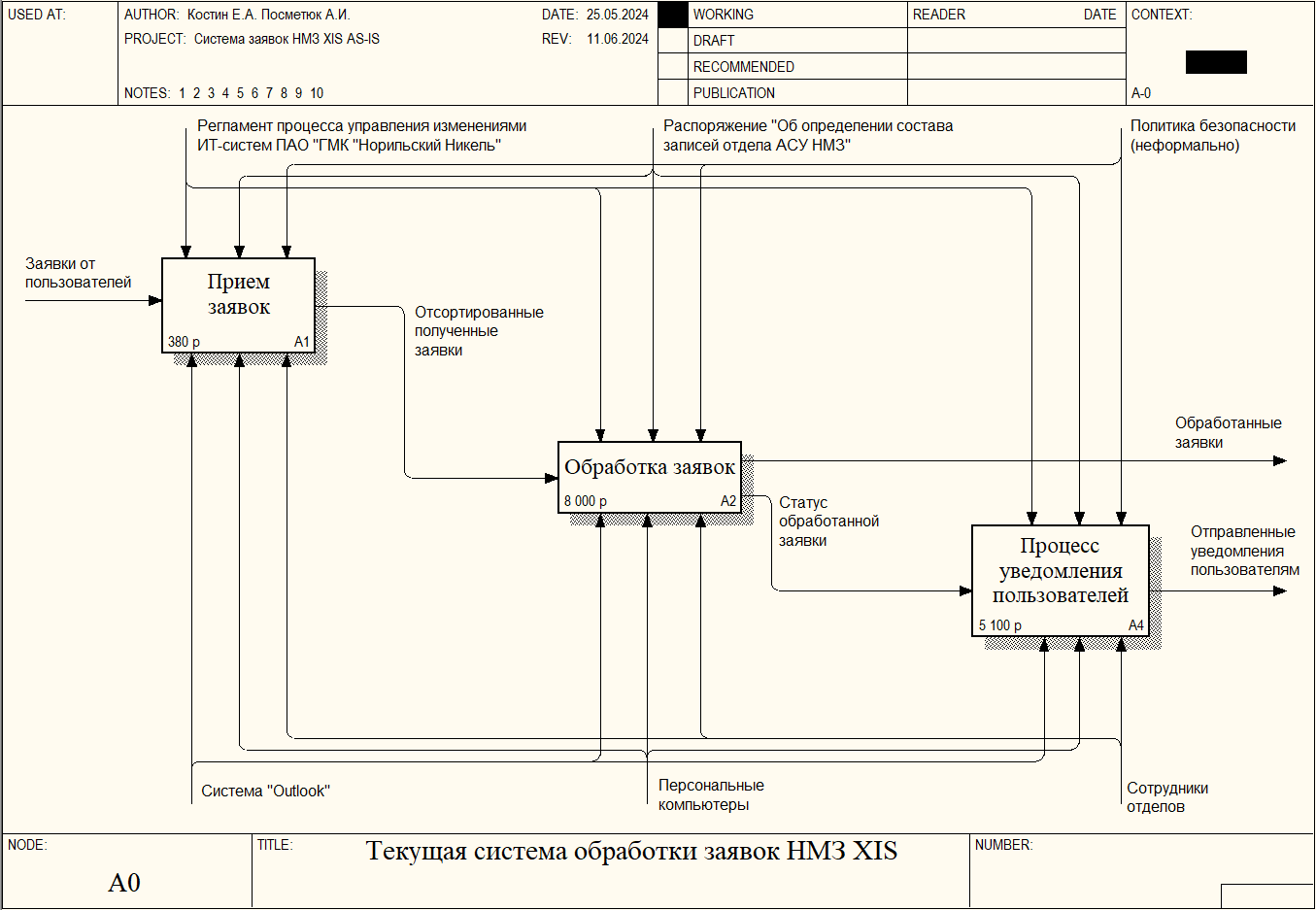


Рисунок 1.2 – Диаграмма IDEF0 AS-IS A0

В декомпозиции диаграммы (Рис. 1.2) показаны основные элементы процесса, включающие приём заявок и их обработку. Основные документы, регулирующие процесс, включают регламент управления изменениями ИТ-систем и распоряжения о составе записей отдела. Обработка заявок осуществляется при помощи персональных компьютеров и системы "Outlook".

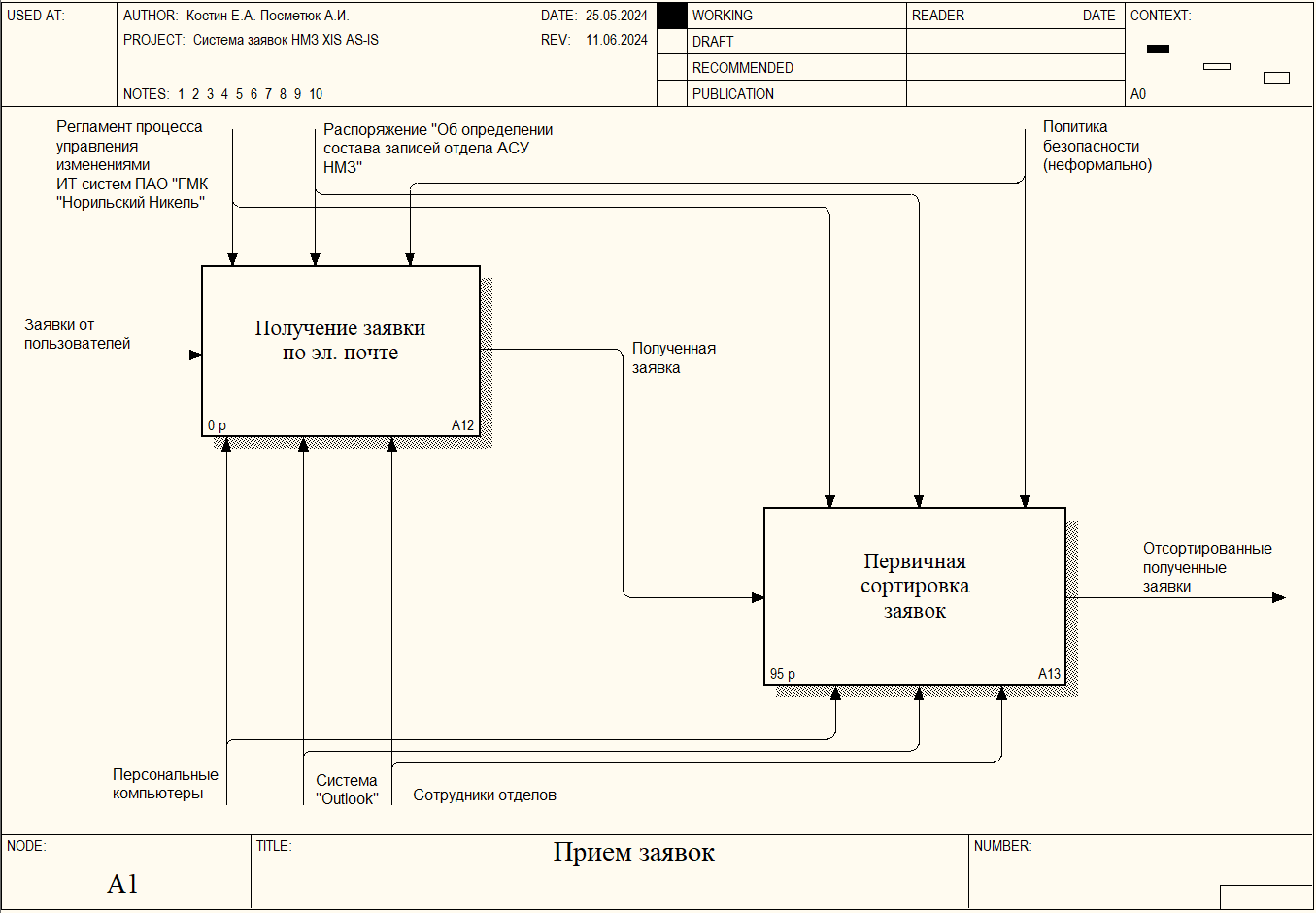


Рисунок 1.3 – Диаграмма IDEF0 AS-IS A1

Диаграмма (Рис. 1.3) описывает получение заявок по электронной почте, их первичную сортировку и передачу для дальнейшей обработки.

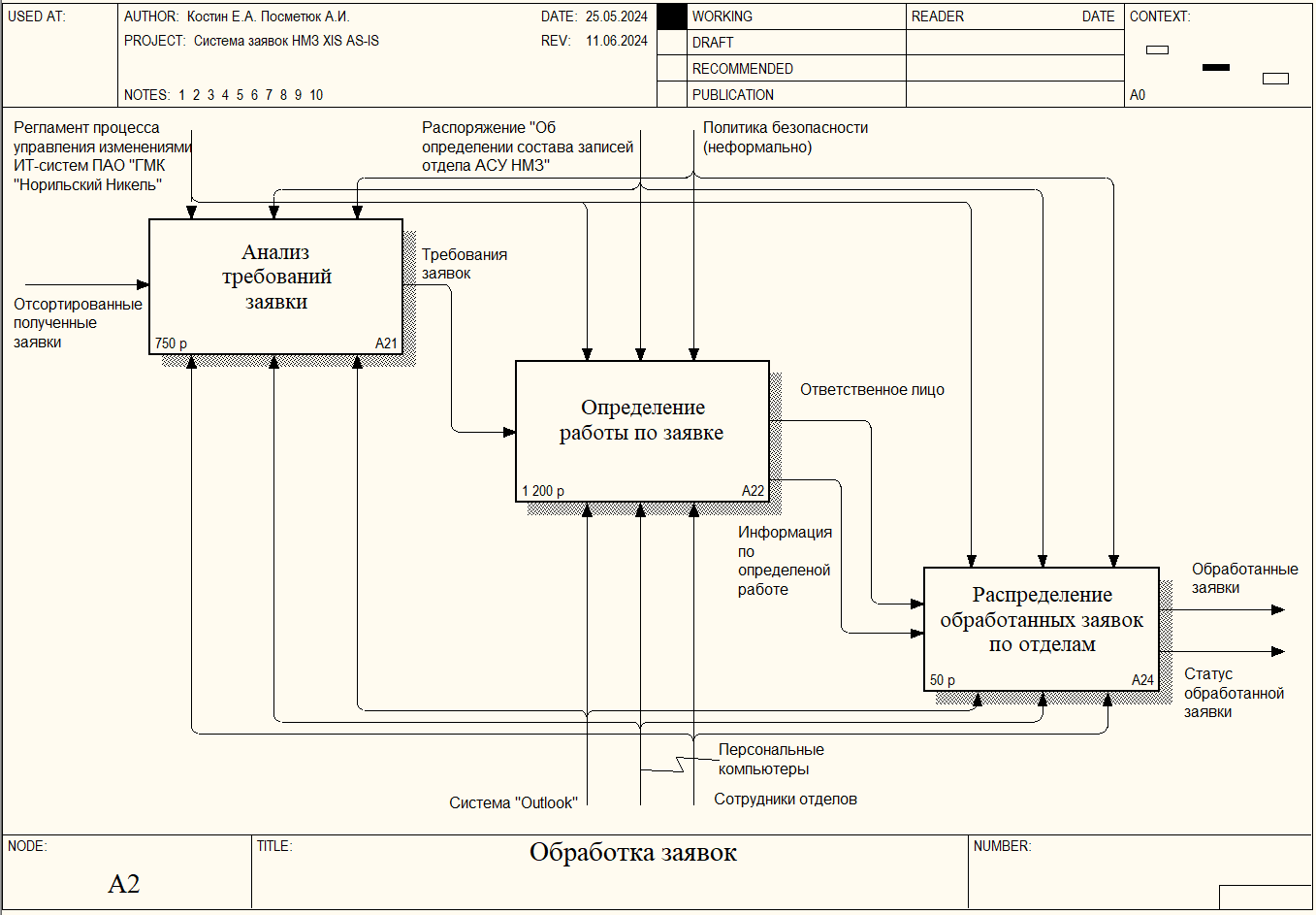


Рисунок 4 – Диаграмма IDEF0 AS-IS A2

Диаграмма (Рис. 1.4) включает в себя анализ требований заявки, определение работы по заявке и распределение обработанных заявок по отделам. Этот процесс позволяет определить ответственных лиц и дальнейшие действия для каждой заявки.

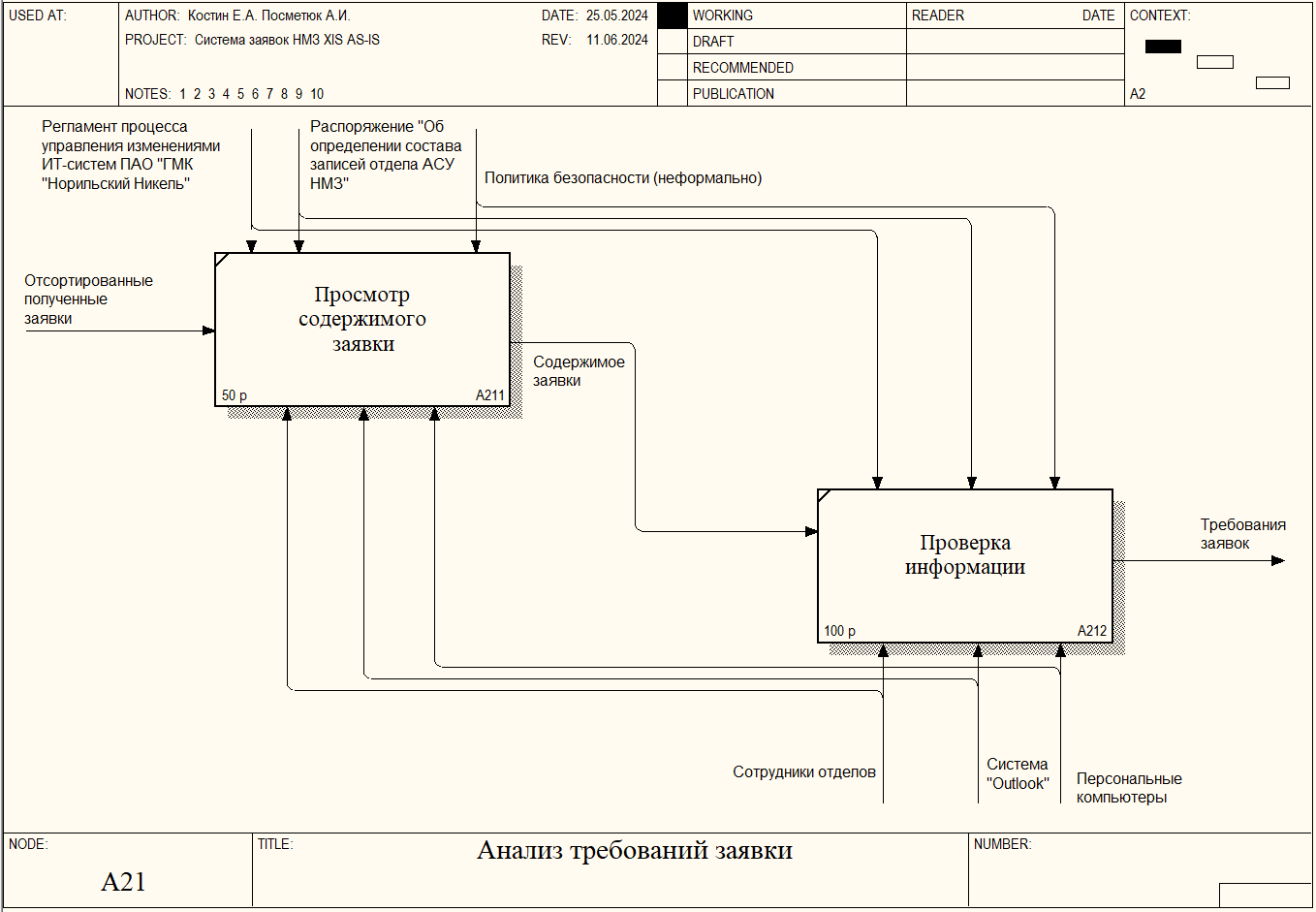


Рисунок 1.5 – Диаграмма IDEF0 AS-IS A21

Диаграмма (Рис. 1.5) включает просмотр содержимого заявки и проверку информации о данной заявке.

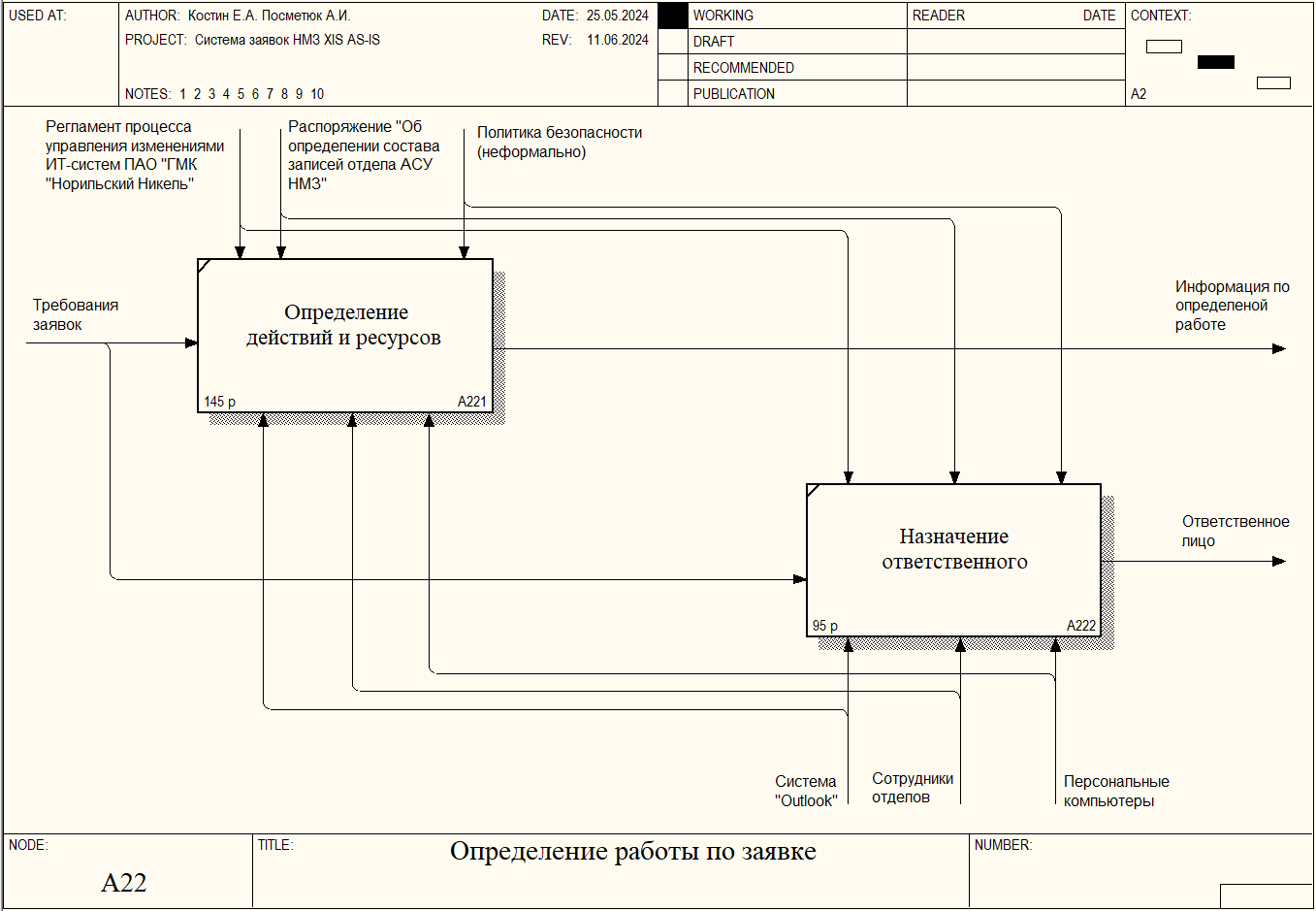


Рисунок 1.6 – Диаграмма IDEF0 AS-IS A22

На данной диаграмме (Рис. 1.6) назначаются ответственные лица за выполнение заявок, а также распределяются ресурсы.

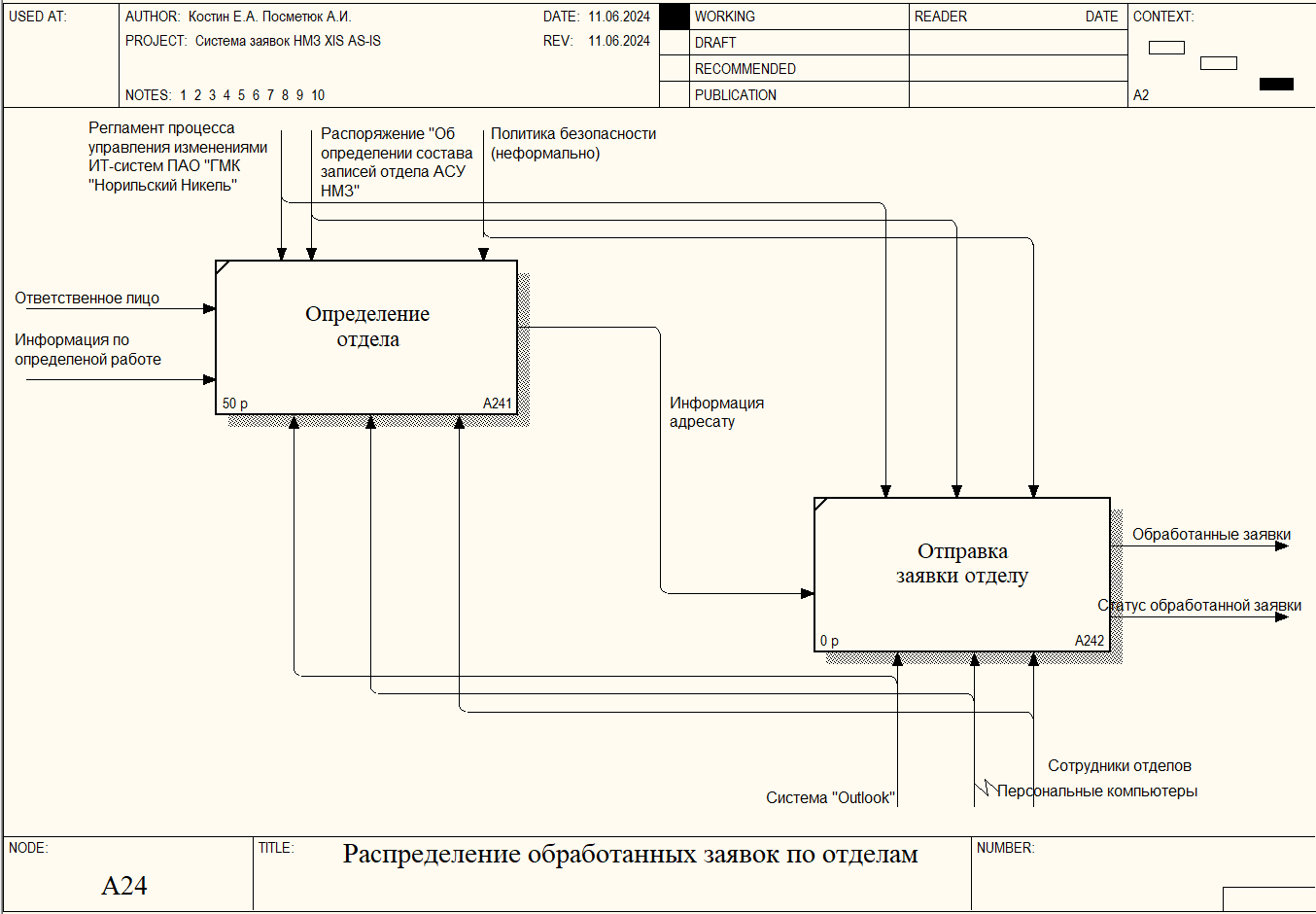


Рисунок 1.7 – Диаграмма IDEF0 AS-IS A24

Диаграмма (Рис. 1.7) включает определение отдела, который получит заявку и сам процесс отправки заявки.

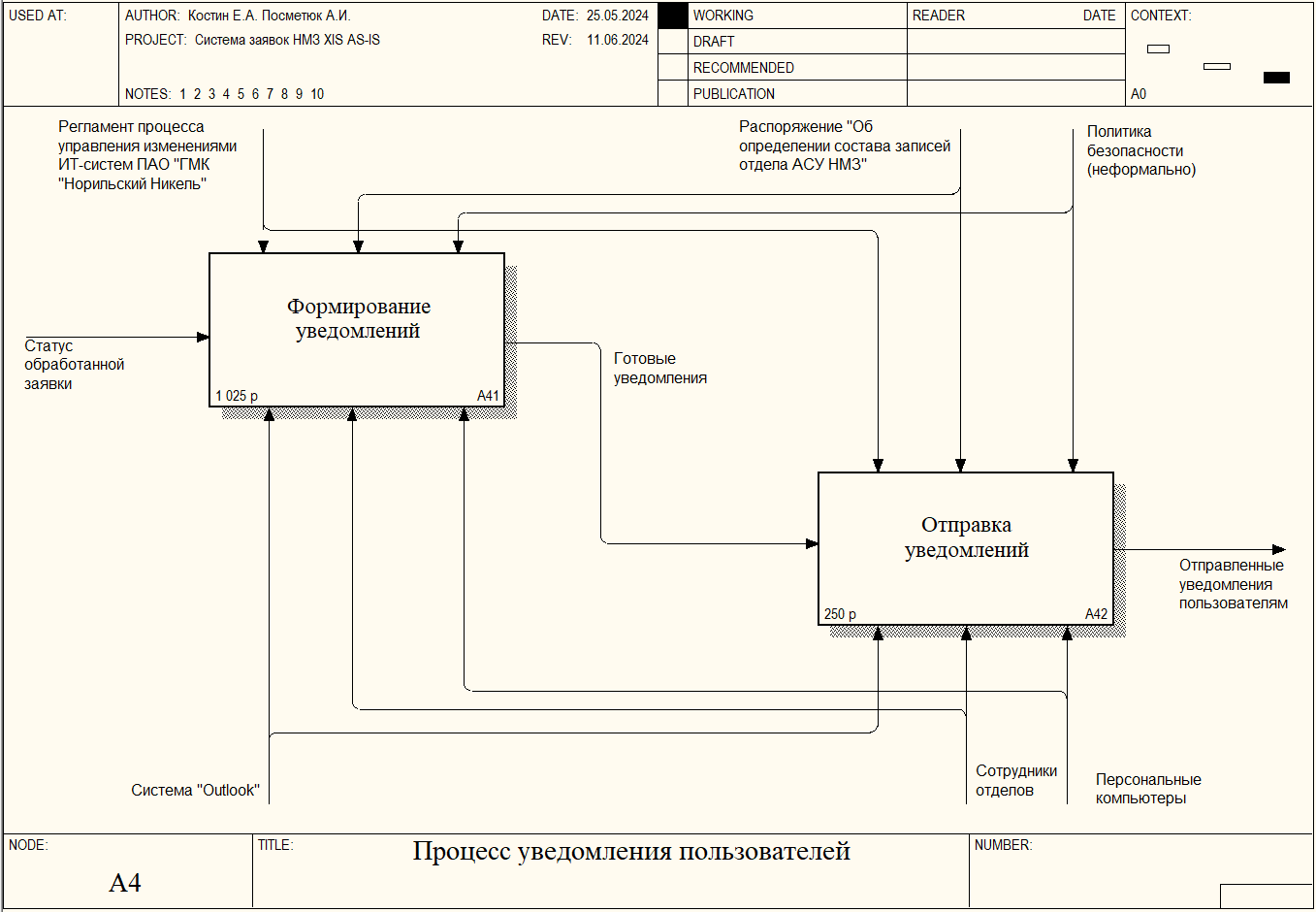


Рисунок 1.8 – Диаграмма IDEF0 AS-IS A4

Диаграмма (Рис. 1.8) включает в себя формирование уведомлений и отправку готовых уведомлений пользователям. Процесс основан на статусе обработанной заявки и использует системы "Outlook" для уведомления пользователей.

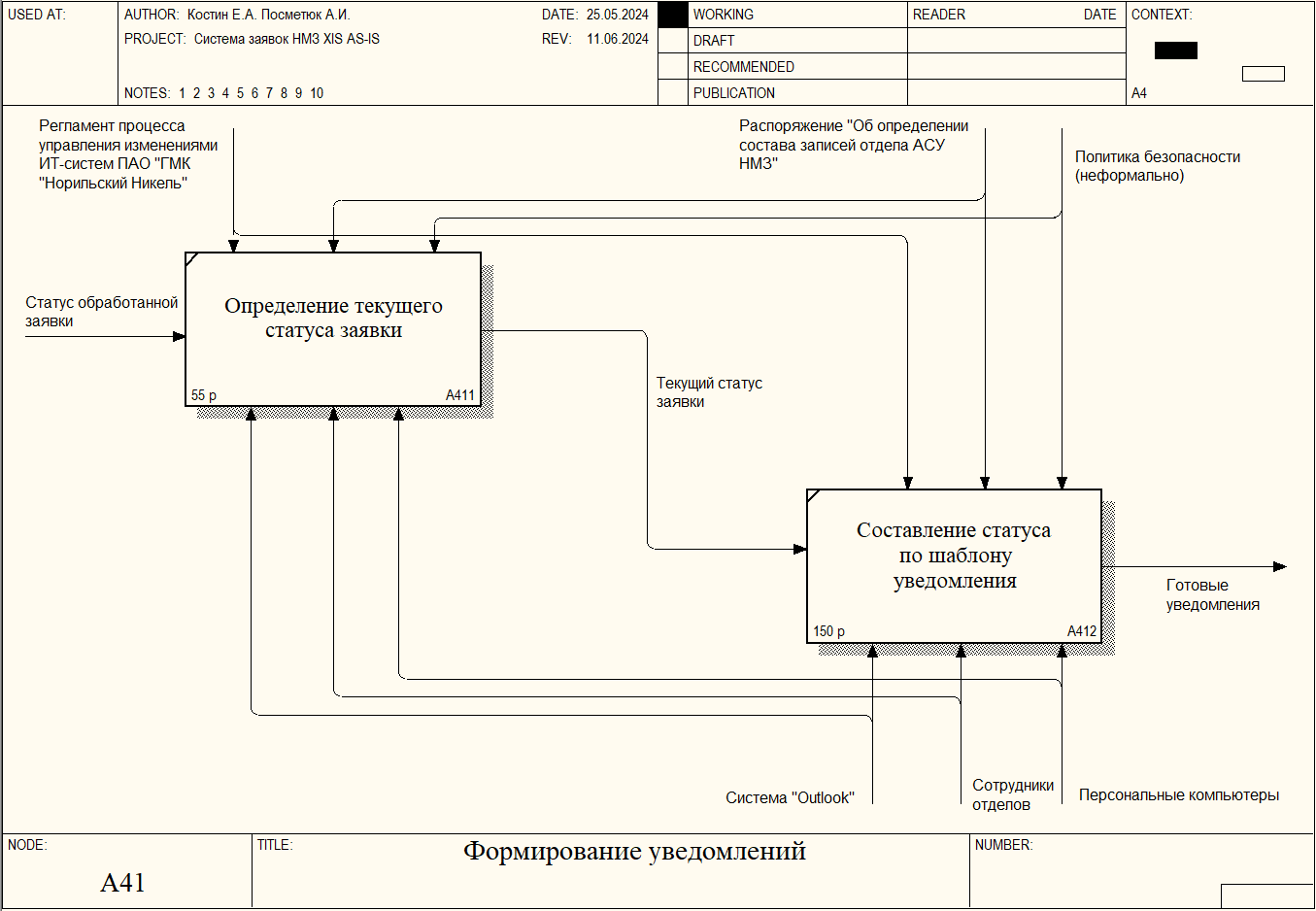


Рисунок 1.9 – Диаграмма IDEF0 AS-IS A41

Диаграмма (Рис. 1.9) включает в себя определение текущего статуса заявки и составление уведомления по шаблону, а затем отправку составленных уведомлений.

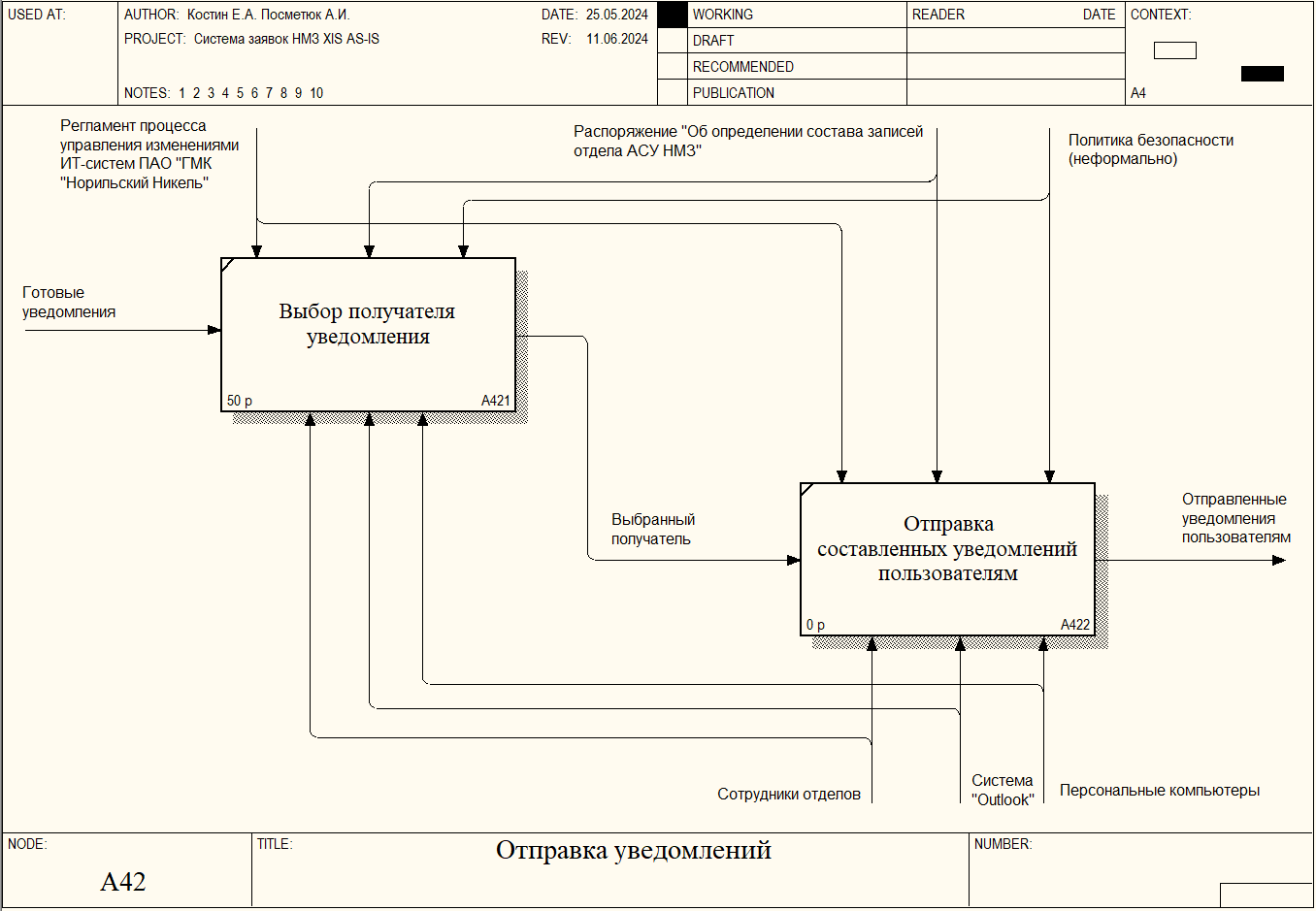


Рисунок 1.10 – Диаграмма IDEF0 AS-IS A42

Диаграмма (Рис. 1.10) включает в себя определение текущего статуса заявки и подготовку информации для уведомления.

Основной целью автоматизации работы является упрощение и ускорение процесса обработки заявок и автоматическое уведомление пользователей.

## **Анализ диаграмм DFD (AS-IS)**

Для описания модели процесса обработки заявок в данной работе используются диаграммы бизнес-процессов DFD модели AS-IS. DFD представляет собой совокупность иерархических, упорядоченных и взаимосвязанных диаграмм, каждая из которых является единицей описания системы, где отличительной особенностью является её акцент на взаимодействия со внешними сущностями и хранилищами.

Контекстная диаграмма (Рис. 1.11) представляет собой исследуемый объект предметной области под названием «Текущая система обработки заявок HM3 XIS».

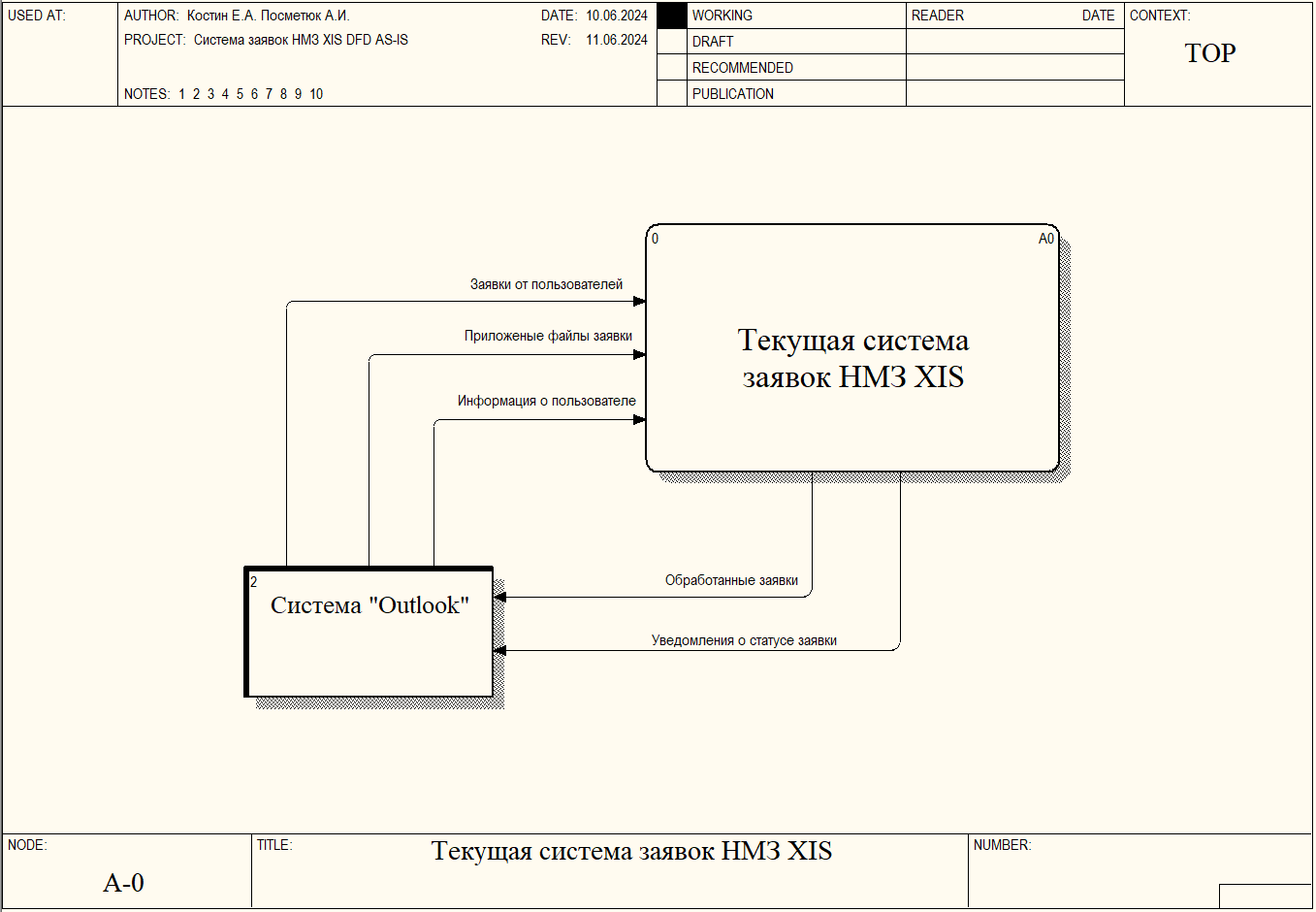


Рисунок 1.11 – Диаграмма DFD AS-IS A-0

В данной диаграмме показаны основные элементы процесса, включающие приём заявок, их обработку и процесс уведомления пользователей.

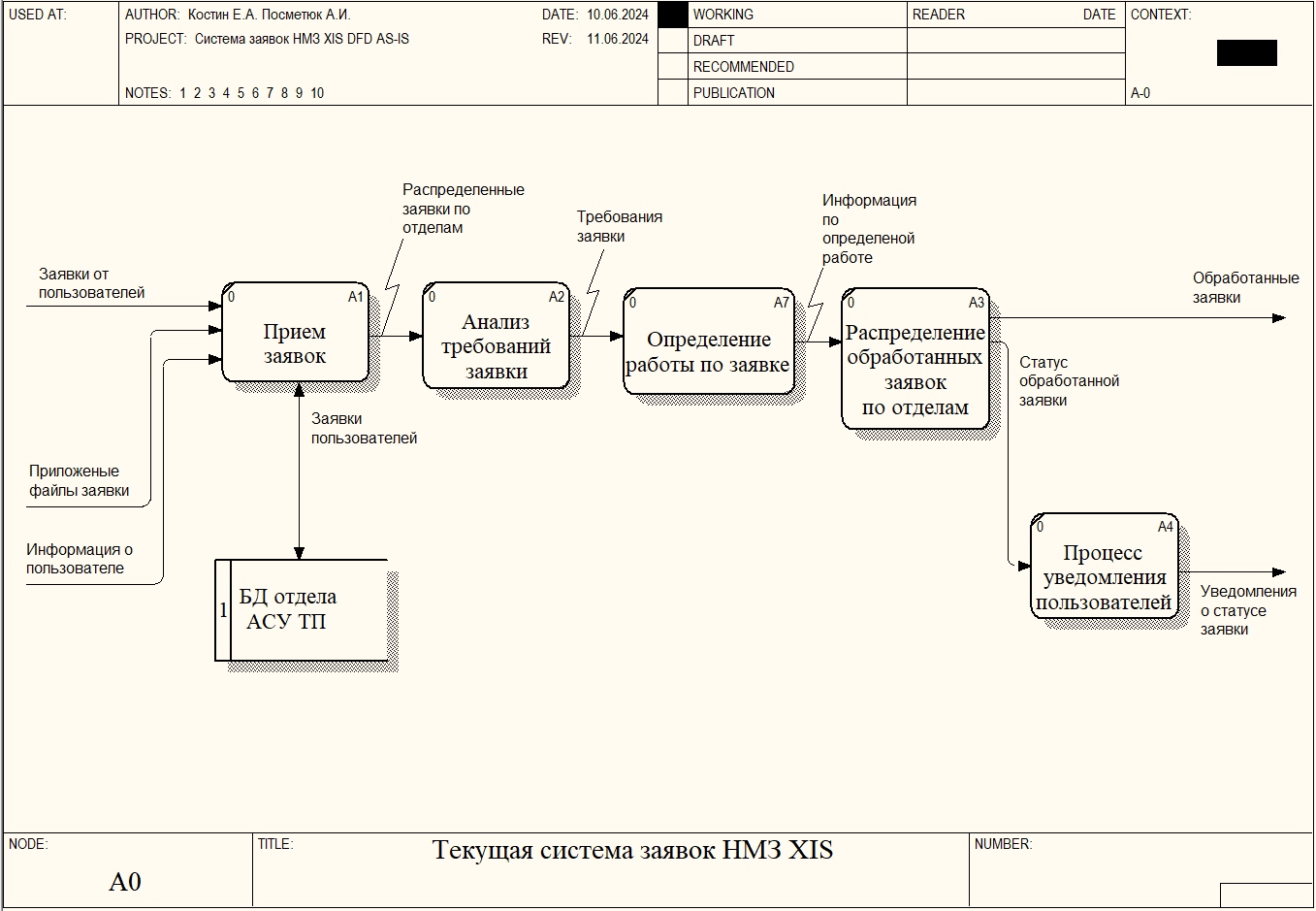


Рисунок 1.12 – Диаграмма DFD AS-IS A0

На декомпозиции диаграммы (Рис. 1.12) представлены следующие основные этапы процесса:

1. Приём заявок:
   1. Приём заявок от пользователей, включая приложенные файлы и информацию о пользователях.
   2. Сохранение заявок в базе данных отдела АСУ ТП.
2. Анализ требований заявки:
   1. Оценка и анализ требований, указанных в заявке.
   2. Формирование требований для дальнейшей обработки.
3. Определение работы по заявке:
   1. Определение конкретной работы, необходимой для выполнения заявки.
   2. Распределение обработанных заявок по отделам.
4. Процесс уведомления пользователей:
   1. Уведомление пользователей о статусе их заявок.

Примеры описания элементов диаграммы:

* Приём заявок

На этом этапе система получает заявки от пользователей, сохраняет приложенные файлы и информацию о пользователях. Все данные сохраняются в базе данных отдела АСУ ТП.

* Анализ требований заявки

Заявки проходят этап анализа, где оцениваются указанные в них требования. Это необходимо для определения конкретной работы по каждой заявке.

* Определение работы по заявке

На этом этапе система определяет необходимую работу для выполнения заявки и распределяет обработанные заявки по соответствующим отделам.

* Процесс уведомления пользователей

Пользователи получают уведомления о статусе их заявок. Это осуществляется через систему "Outlook".

## **Патентный поиск**

Для автоматизированного управления системой заявок на модернизацию НМЗ XIS, программное обеспечение должно содержать функционал, позволяющий обрабатывать, хранить и анализировать информацию о заявках, а также обеспечивать взаимодействие с пользователями. Основные требования включают комфортное управление статусами заявок при использовании мобильного интерфейса.

**Zendesk** — это web-приложение для управления заявками и поддержки клиентов. Оно позволяет пользователям подавать заявки через различные каналы (веб, email, чат) и обеспечивает автоматическую обработку заявок.

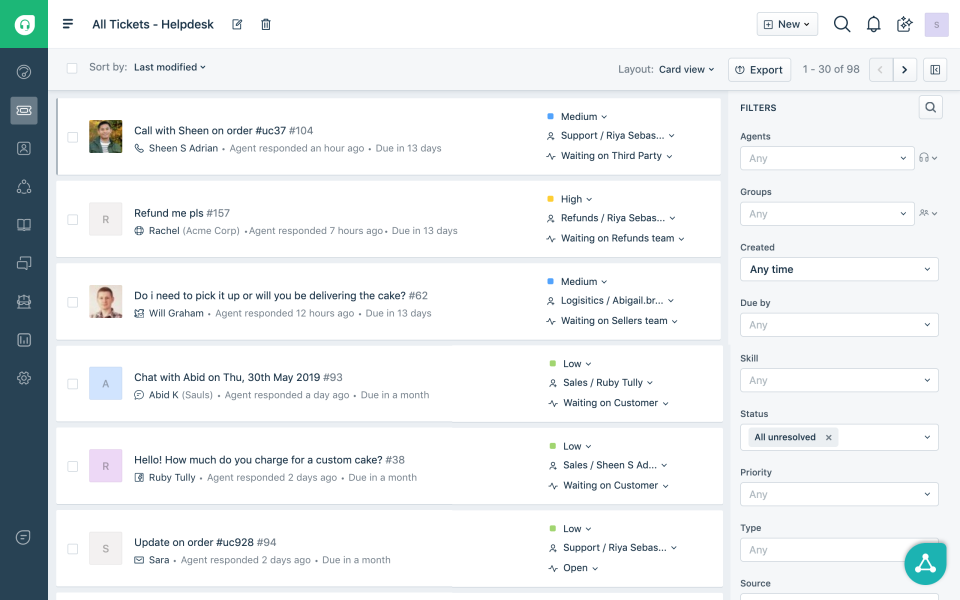


Рисунок 1.13 – Интерфейс приложения Zendesk

Преимущества: интуитивно понятный интерфейс, широкие возможности интеграции с другими системами.

Недостатки: Высокая стоимость, сложность в настройке и развертывании, требует значительных ресурсов для поддержки и обучения пользователей.

**Okdesk** — это система тикетов, предлагающая мобильное приложение для инженеров на платформах Android и iOS.

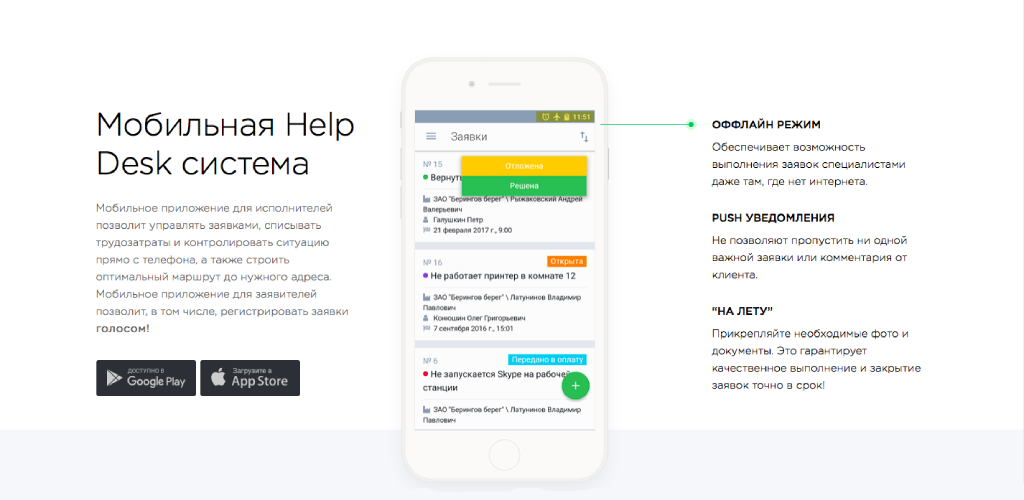


Рисунок 1.14 – Интерфейс приложения Okdesk

Преимущества: Приложение обеспечивает интуитивно понятный интерфейс, облегчающий работу инженеров, своевременное информирование пользователей о статусе заявок.

Недостатки: Ограниченные возможности для крупномасштабных проектов, некоторые пользователи могут столкнуться с нехваткой определенных функций, необходимых для специфических задач.

**Freshdesk** - платформа для управления клиентскими заявками, популярная среди компаний, предоставляющих услуги.

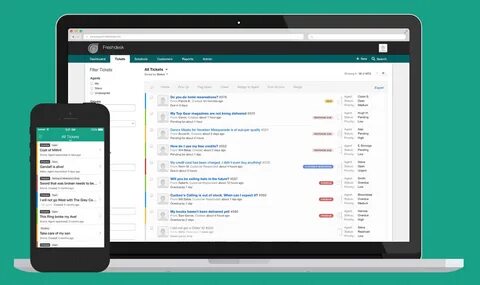


Рисунок 1.15 – Интерфейс приложения Freshdesk

Преимущества: Простота в использовании и настройке, полный функционал доступен с телефона.

Недостатки: Некоторые функции доступны только в платных тарифах, лучше подходит для работы с внешними клиентами.

Сравнение функциональных особенностей перечисленных программ представлен в таблице №1.1.

Таблица №1.1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Функционал** | **Система заявок НМЗ XIS** | **Zendesk** | **Okdesk** | **Freshdesk** |
| |  | | --- | | Мобильное приложение |  |  | | --- | |  | | + | + | + | + |
| Управление заявками | + | + | + | + |
| Интеграция с внешними системами | + | - | - | + |

Из перечисленных программ только текущая система заявок НМЗ XIS обладает всеми необходимыми функциями для полного управления процессом подачи и обработки заявок на модернизацию системы. Особенным фактором является интеграция с существующими системами.

Таким образом, текущая система заявок НМЗ XIS представляет собой оптимальное решение для автоматизации и улучшения процессов управления заявками, обеспечивая высокую эффективность и удобство работы.

## **Вывод по главе**

В данной главе была рассмотрена актуальность разработки мобильного приложения для автоматизации процесса подачи и обработки заявок на модернизацию системы НМЗ XIS. Обоснована необходимость автоматизации данного процесса для улучшения качества управления заявками и повышения эффективности работы сотрудников на Надеждинском металлургическом заводе им. Б. И. Колесникова.

Был проведен анализ текущих проблем ручной обработки заявок, выявлены основные трудности, связанные с высокой трудоемкостью, временными затратами и рисками возникновения ошибок. Установлено, что для решения этих проблем требуется система, которая обеспечит автоматизацию процесса подачи и обработки заявок.

Описание архитектуры проектируемого мобильного приложения включает в себя использование технологий React Native, бэкенд на базе Python и кластер Docker контейнеров, а также хранилище данных PostgreSQL и систему управления неструктурированными данными MinIO.

Был проведен сравнительный анализ аналогичных мобильных приложений, таких как Zendesk, Okdesk и Freshdesk, с целью выявления их преимуществ и недостатков. Сравнение функциональных возможностей данных систем позволило определить уникальные аспекты и требования к новой системе заявок НМЗ XIS.

Основные сущности проектируемой системы, такие как роли пользователей, данные пользователей, заявки, вложения, услуги и услуги в заявке, были детально описаны. Также был представлен анализ диаграмм IDEF0 (AS-IS) и контекстных диаграмм нотации DFD (AS-IS), которые отразили текущее состояние процессов обработки заявок на заводе.

В заключение, была проведена оценка патентного поиска аналогичных систем и выявлены недостатки существующих решений. На основе анализа было сделано заключение, что новая система заявок НМЗ XIS должна включать все необходимые функции для полного управления процессом подачи и обработки заявок, обеспечивая высокую надежность и удобство использования.

# **Глава 2. ИНФОЛОГИЧЕСКОЕ МОДЕЛИРОВАНИЕ**

## **Формирование и утверждение документа по стратегии**

Документ по стратегии разрабатывается для создания четкого представления о целях и ожидаемых результатах проекта. Он описывает границы новой системы и включает в себя план реализации проекта.

## **Рабочее резюме**

Для автоматизации процесса подачи и обработки заявок на модернизацию системы НМЗ XIS предлагается разработать мобильного приложение с использованием технологий React Native, Python и Docker. В текущей системе все заявки обрабатываются вручную, что приводит к высоким трудозатратам и увеличению временных затрат.

## **Описание старой системы**

На данный момент процесс подачи и обработки заявок на модернизацию системы НМЗ XIS осуществляется вручную. Это требует значительных временных и трудовых затрат, а также приводит к частым ошибкам и неточностям.

## **Взаимодействующие проекты**

Новое мобильное приложение будет интегрироваться с существующими корпоративными системами НМЗ XIS для обеспечения полной автоматизации процессов подачи и обработки заявок.

## **Деловое и финансовое субсидирование**

Этот проект имеет важное значение для оптимизации рабочих процессов на НМЗ XIS. Он обеспечит упрощение работы с заявками, уменьшение вероятности ошибок и снижение трудозатрат. Финансирование на текущий момент не предусмотрено.

## **Побуждения**

Сотрудники НМЗ XIS в настоящее время используют для своих задач множество различных инструментов, включая Microsoft Outlook. Хотя эти инструменты позволяют сократить время на выполнение некоторых задач, общая работа остается трудоемкой и требует значительных временных затрат. Автоматизация процесса подачи и обработки заявок поможет преодолеть эти трудности, значительно уменьшив временные затраты и повысив общую эффективность.

## **Границы проекта**

Разрабатываемая система должна решать следующие задачи:

1. Автоматизация подачи заявок.
2. Удобная реализация обработки заявок.
3. Интеграция с существующими системами НМЗ XIS для обеспечения полного контроля и управления процессом.

## **Рабочий план проекта**

План проекта разработан в Microsoft Excel. Общая продолжительность выполнения проекта составляет 77 дней. Стоимость проекта составляет 54 362 рубля. (исходя из среднедневного размера стипендии студента: 14840/21 \* 77).



Рисунок 2.1 - Рабочий план проекта

## **Заключение**

Проект будет реализован с использованием современных технологий, таких как React Native, Python и Docker. В рамках проекта будут созданы модели данных и информационных потоков, спроектированы базы данных и конфигурация системы. Документ по стратегии описывает план реализации проекта, его границы и перспективы развития, обеспечивая четкое руководство для успешного выполнения всех задач.

## **Анализ диаграммы IDEF0 (TO-BE)**

При внедрении новой информационной системы появится новый механизм «Система заявок НМЗ XIS», который позволит автоматизировать процесс подачи и обработки заявок.

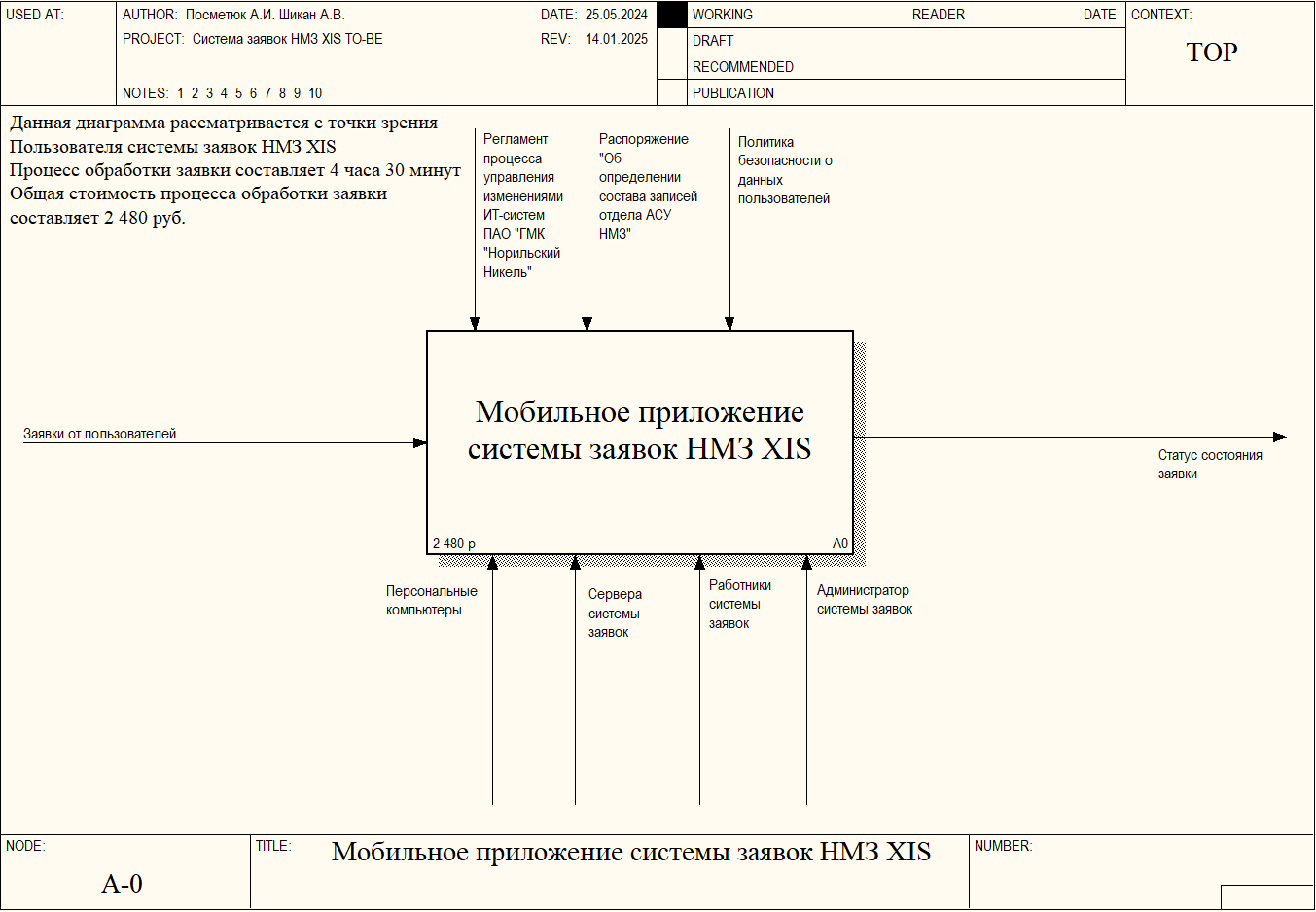


Рисунок 2.2 – Диаграмма IDEF0 TO-BE A-0

Контекстная диаграмма представляет собой исследуемый объект предметной области под названием «Система заявок НМЗ XIS».

В данной диаграмме показаны основные элементы процесса, включающие регистрацию заявок и их обработку. Основные документы, регулирующие процесс, включают регламент управления изменениями ИТ-систем, распоряжения о составе записей отдела и политику безопасности данных пользователей. Обработка заявок осуществляется при помощи персональных компьютеров и серверов системы заявок.

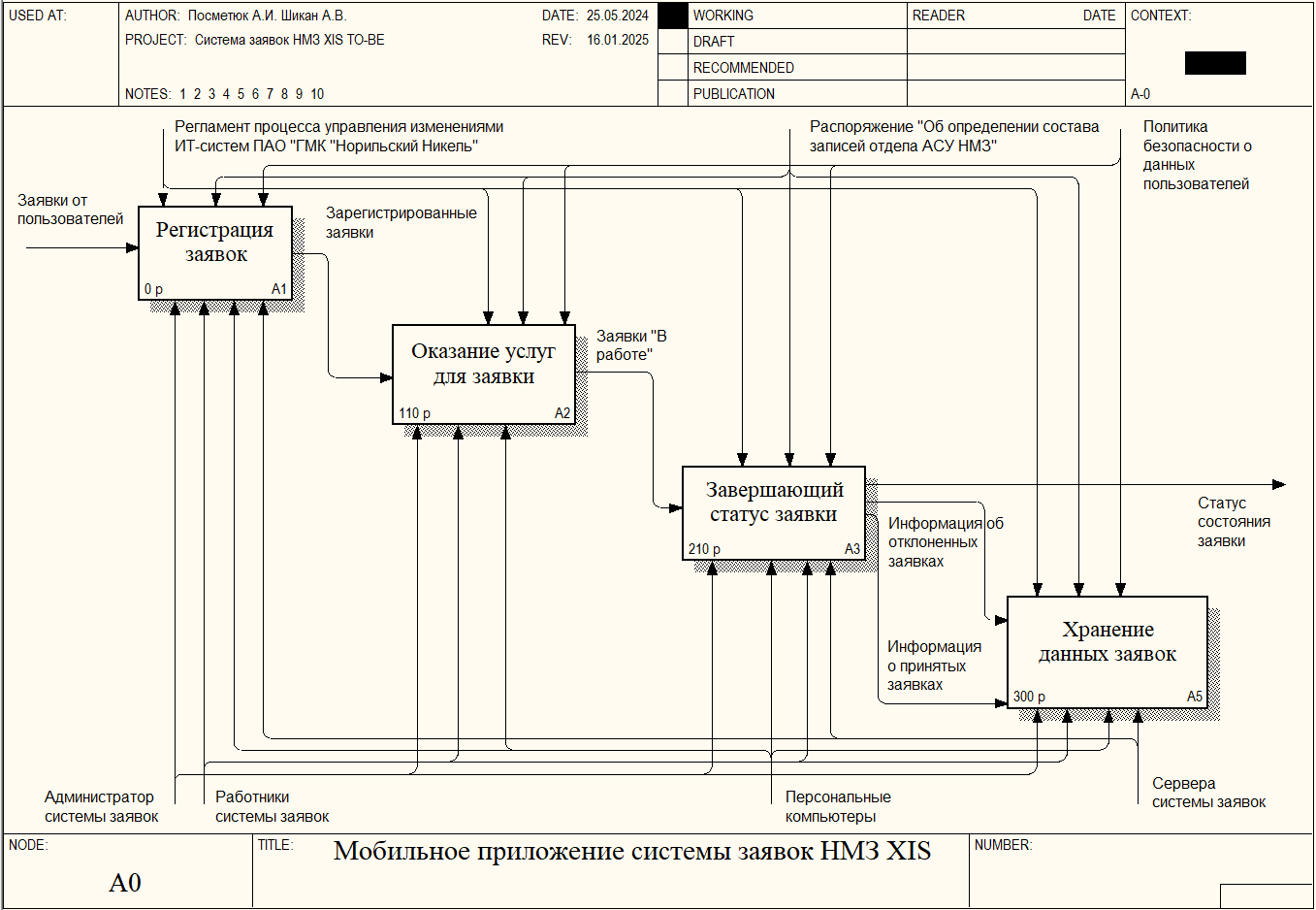


Рисунок 2.3 – Диаграмма IDEF0 TO-BE A0

Этот процесс включает в себя получение заявок от пользователей, их регистрацию, обработку и анализ требований.

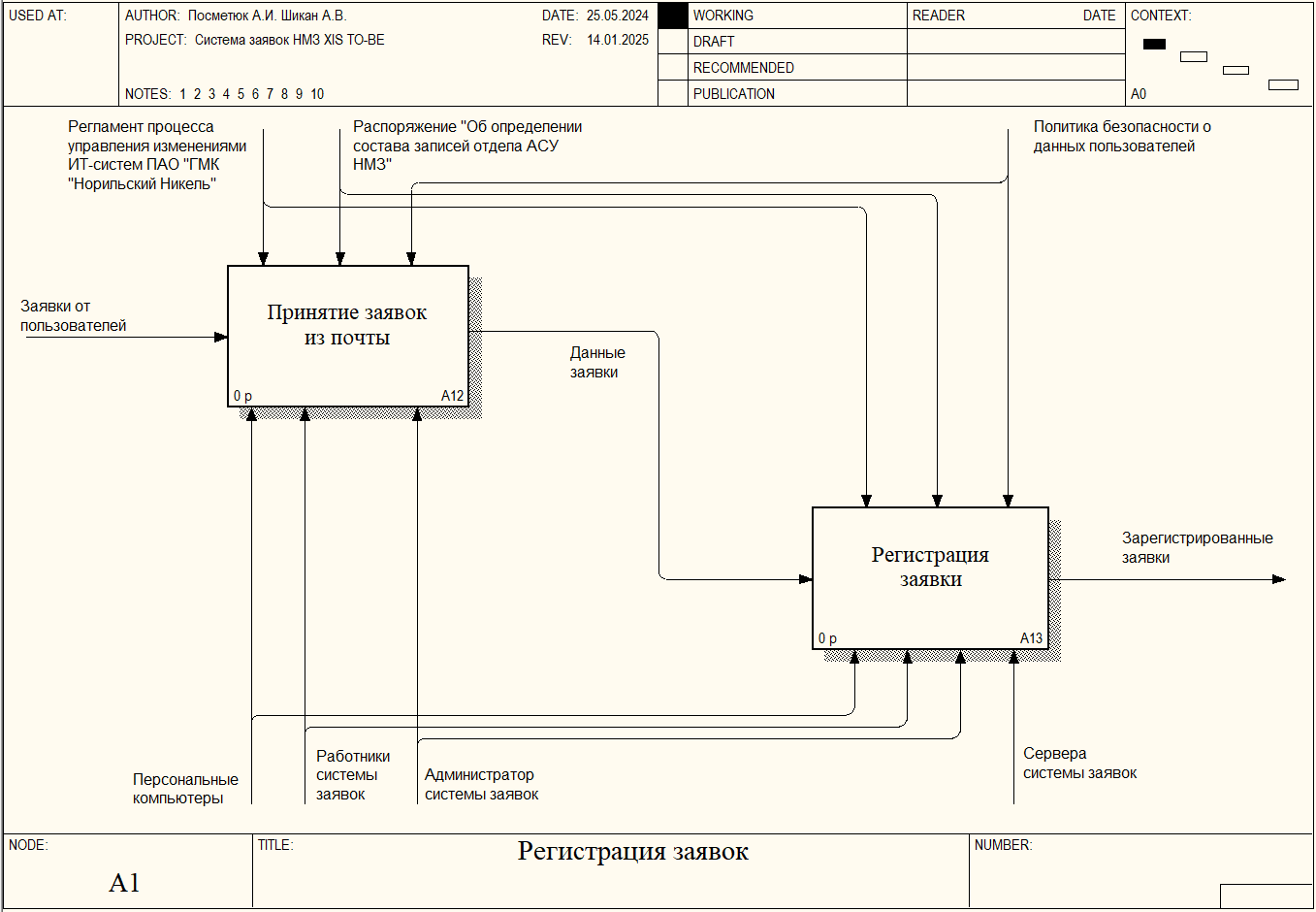


Рисунок 2.4 – Диаграмма IDEF0 TO-BE A1

Диаграмма показывает процесс регистрации заявок. Основные этапы включают принятие заявок из почты, проверку корректности данных, передачу данных заявки, создание единого образа заявок с данными и передачу образа заявок в систему.

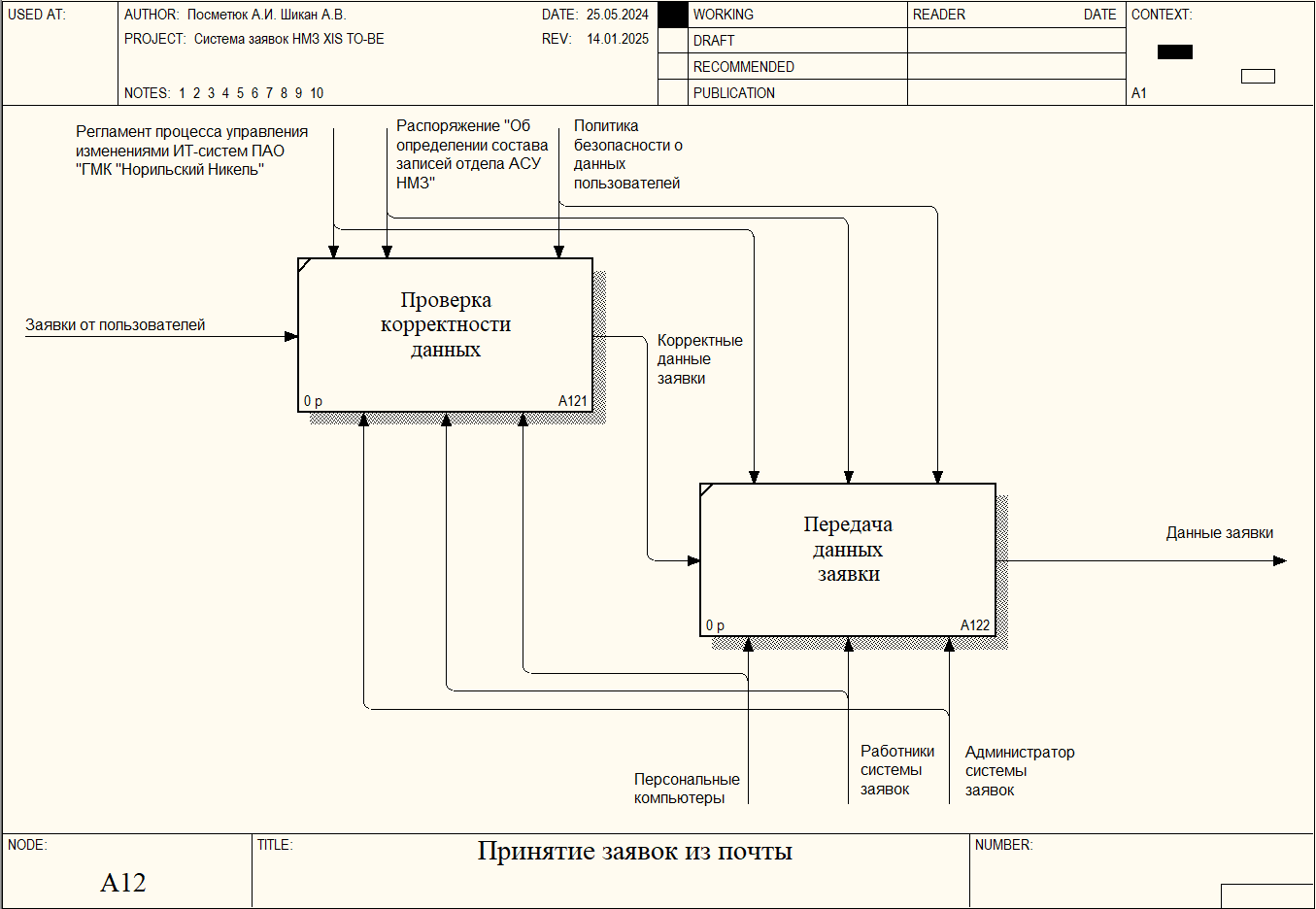


Рисунок 2.5 – Диаграмма IDEF0 TO-BE A12

Диаграмма детализирует процесс принятия заявок из почты. Включает проверку корректности данных и передачу данных заявки.

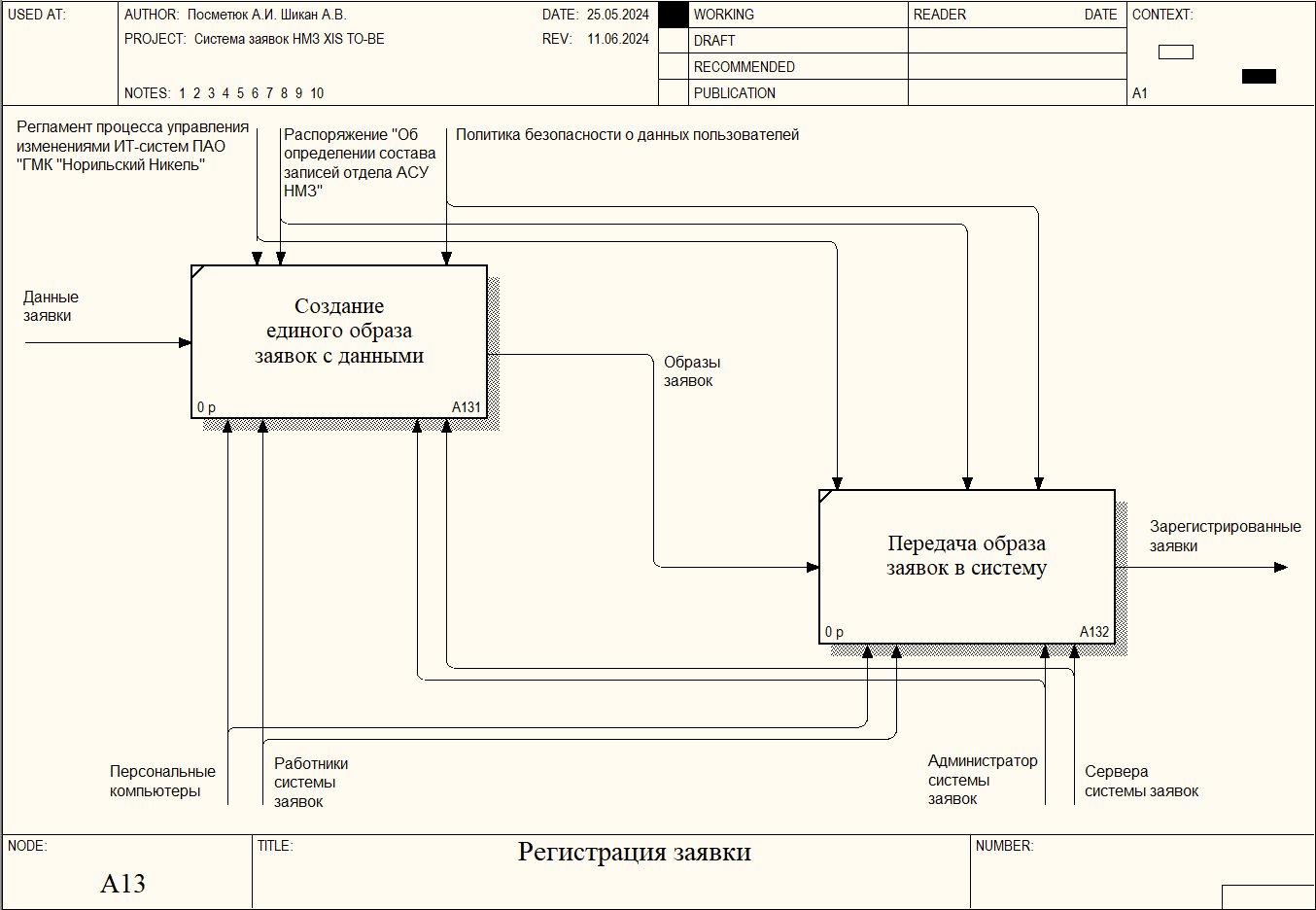


Рисунок 2.6 – Диаграмма IDEF0 TO-BE A13

Диаграмма описывает процесс создания единого образа заявок с данными и передачу образа заявок в систему.

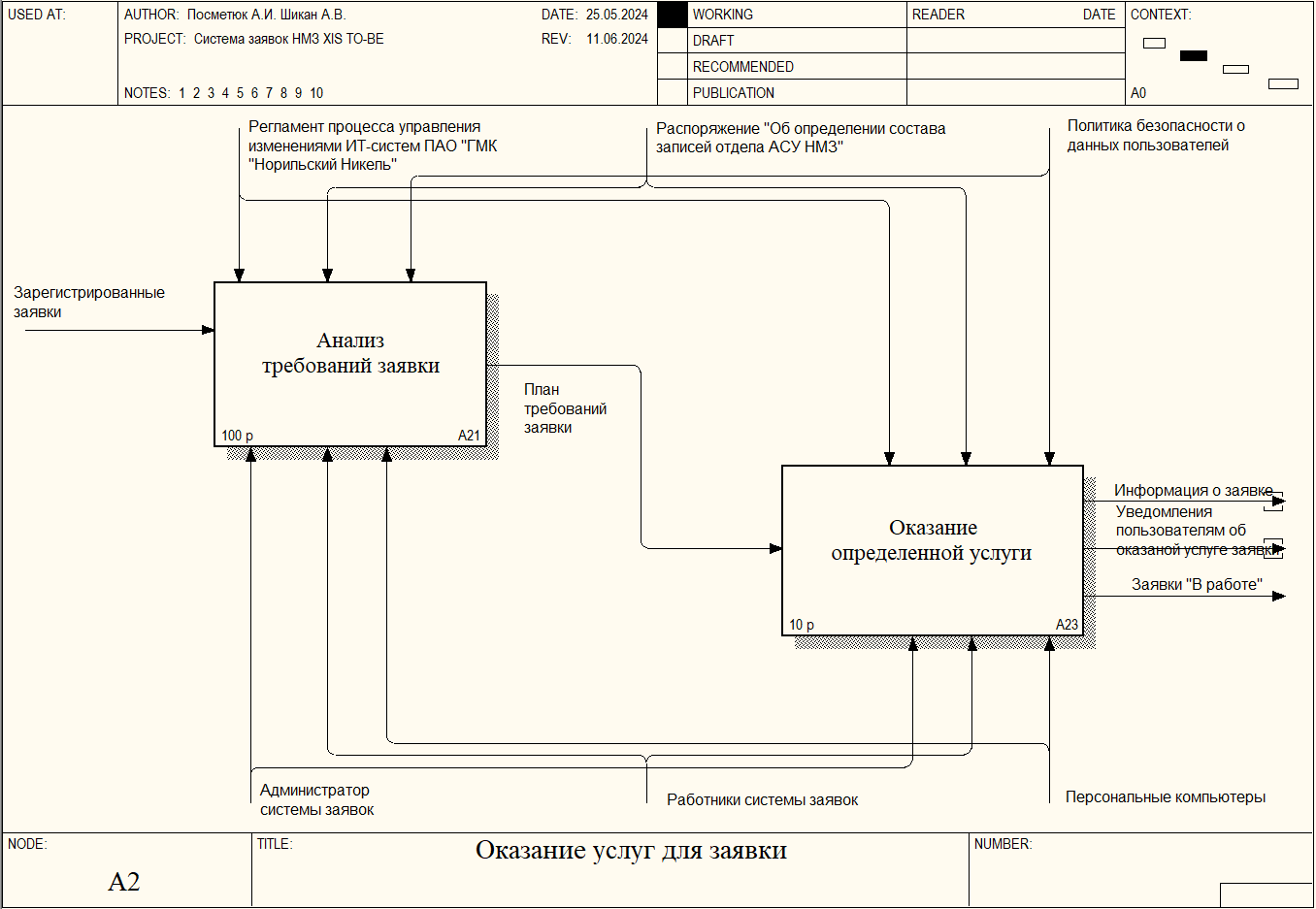


Рисунок 2.7 – Диаграмма IDEF0 TO-BE A2

Диаграмма описывает процесс оказания услуг для заявки. Включает анализ требований заявки и оказание определенной услуги.

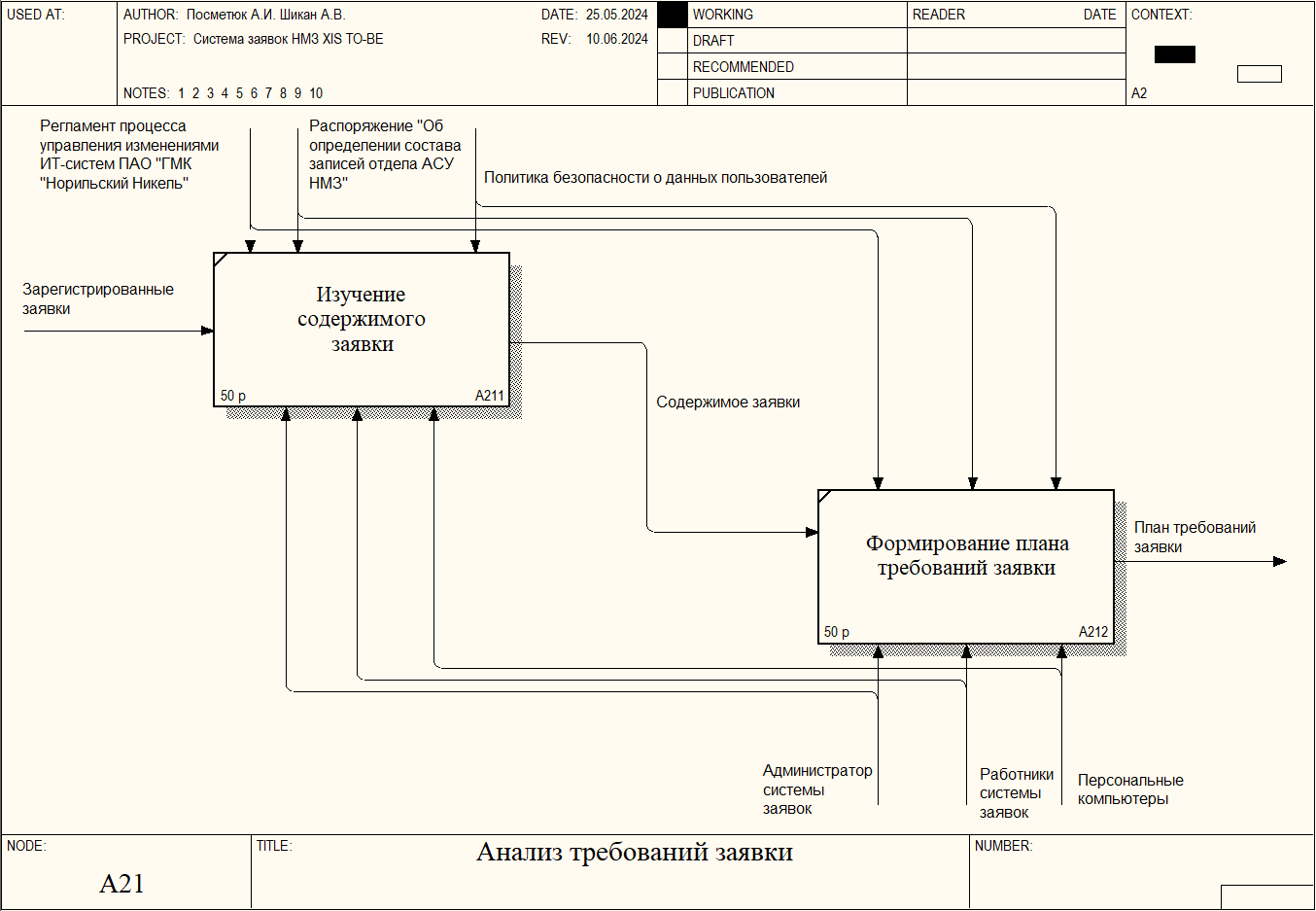


Рисунок 2.8 – Диаграмма IDEF0 TO-BE A21

Диаграмма детализирует анализ требований заявки. Включает изучение содержания заявки и формирование плана требований заявки.

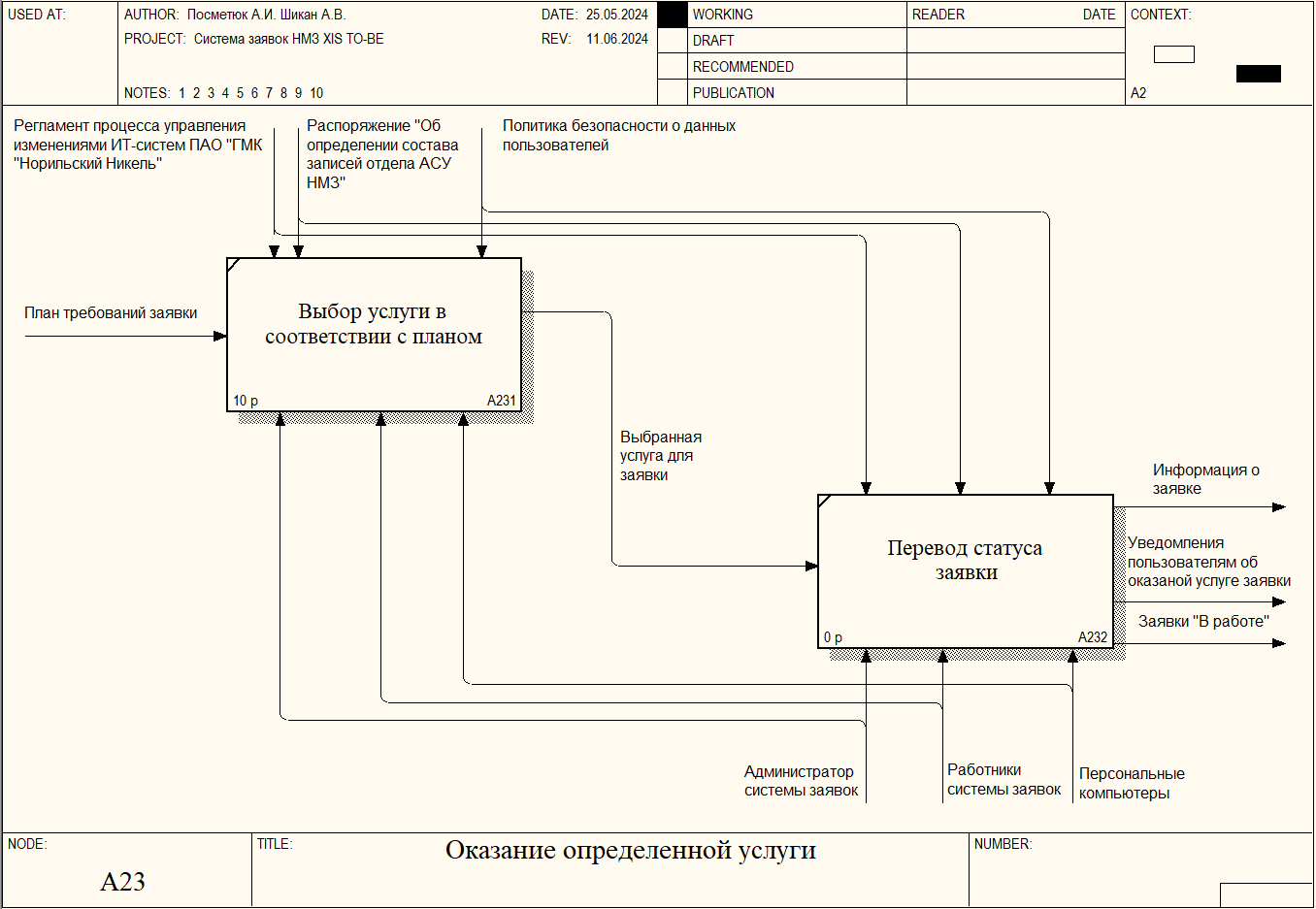


Рисунок 2.9 – Диаграмма IDEF0 TO-BE A23

Диаграмма описывает процесс оказания определенной услуги. Включает выбор услуги в соответствии с планом и перевод статуса заявки.

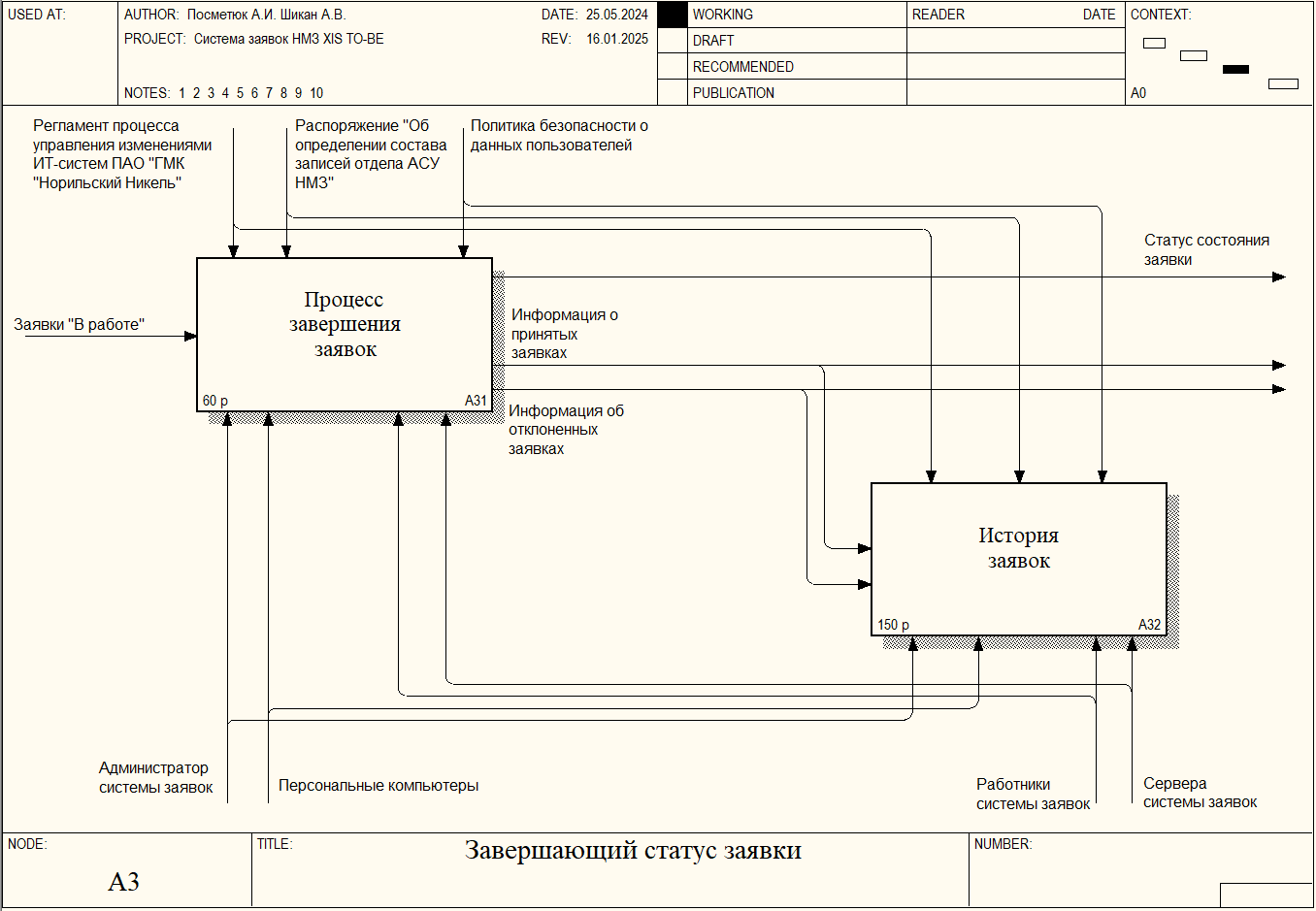


Рисунок 2.10 – Диаграмма IDEF0 TO-BE A3

Диаграмма описывает процесс завершения заявок. Включает анализ обработанных заявок и принятие решения по заявке.

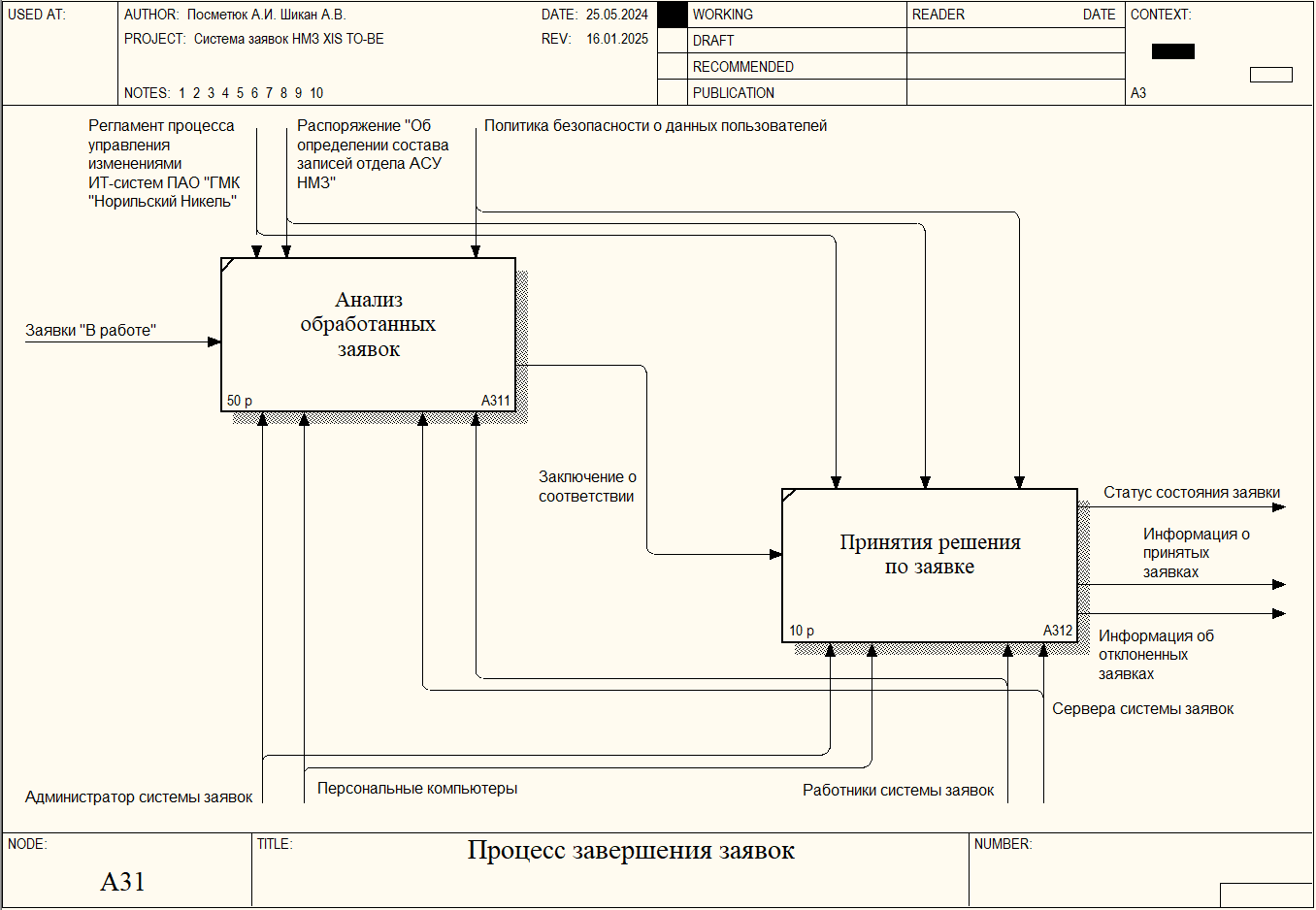


Рисунок 2.11 – Диаграмма IDEF0 TO-BE A31

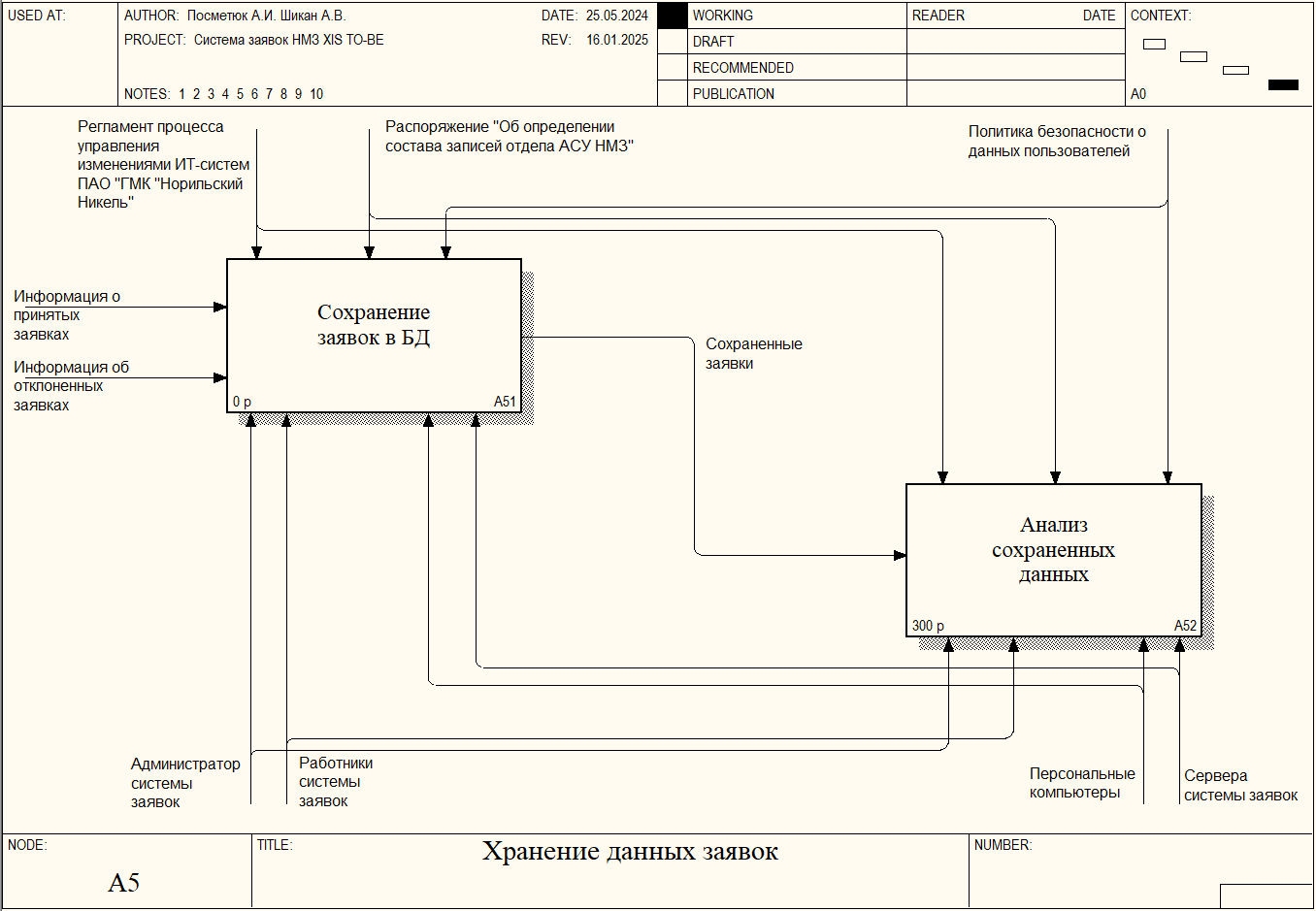


Рисунок 2.16 – Диаграмма IDEF0 TO-BE A5

Диаграмма показывает сохранение заявок в базе данных и анализ сохранённых данных. Это обеспечивает надёжное хранение и анализ данных для дальнейшего использования.

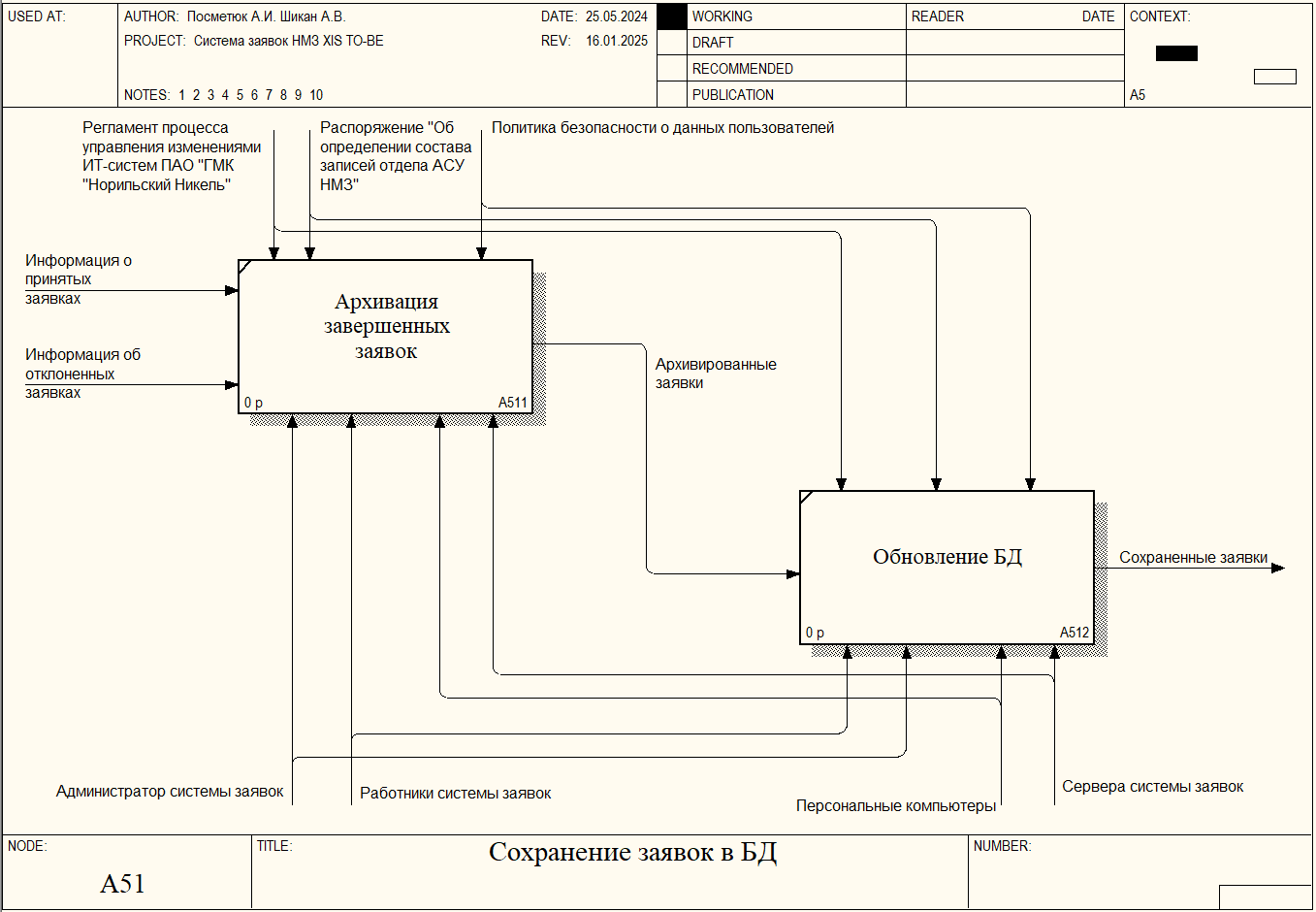


Рисунок 2.17 – Диаграмма IDEF0 TO-BE A51

Диаграмма описывает сохранение завершённых заявок в базе данных и обновление базы данных. Это позволяет сохранять историю заявок и поддерживать актуальность данных.

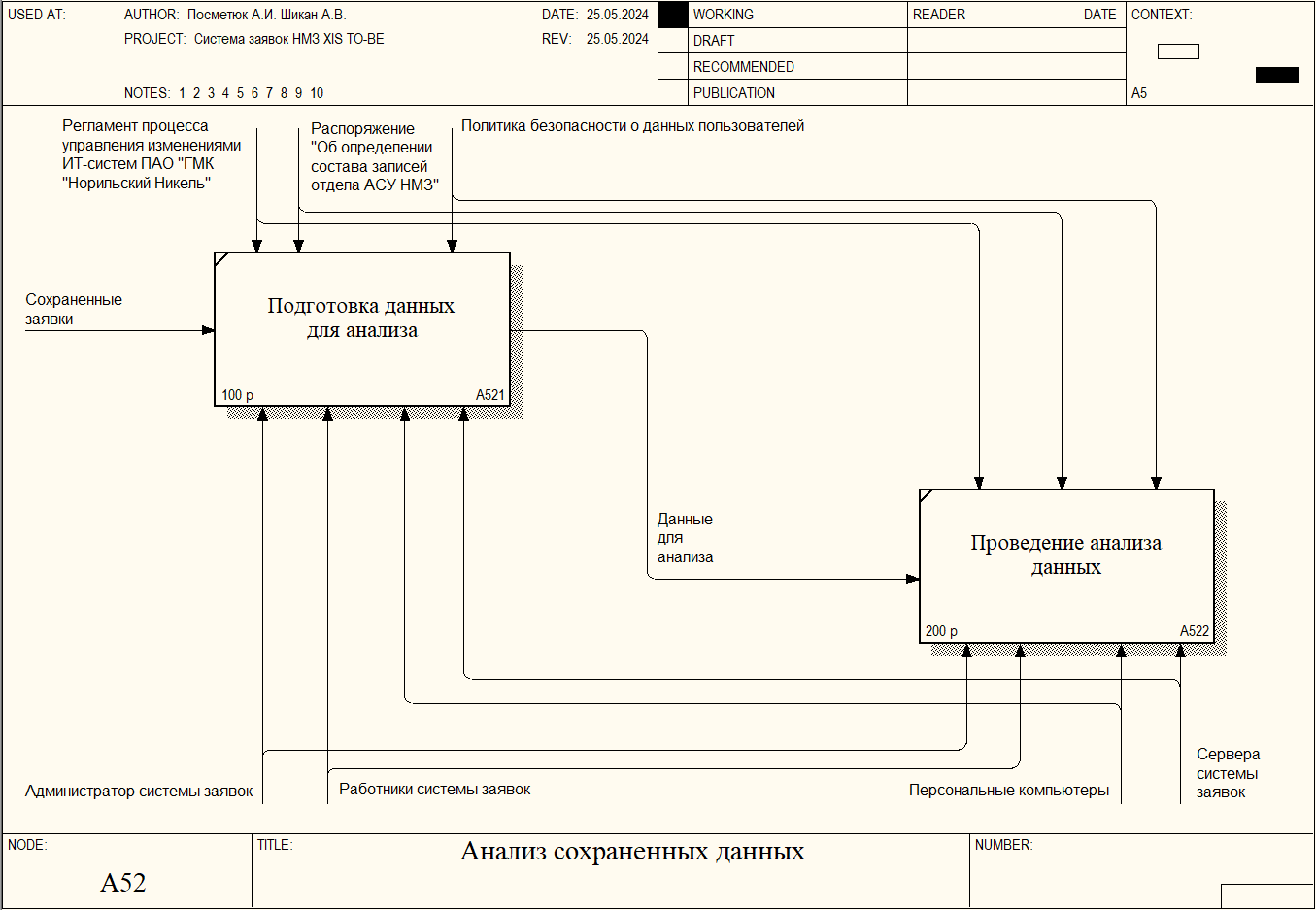


Рисунок 2.18 – Диаграмма IDEF0 TO-BE A52

Диаграмма описывает подготовку данных для анализа и проведение анализа данных. Это позволяет извлекать полезную информацию из сохранённых данных для принятия управленческих решений.

Основные изменения и их влияние

Внедрение системы заявок НМЗ XIS изменит структуру процессов. Вместо ручного ввода и обработки заявок будет использоваться автоматизированная система, которая значительно сократит временные и трудовые затраты.

С внедрением системы заявок НМЗ XIS временные затраты на процесс «Подача и регистрация заявок» снизятся, «Обработка и анализ заявок» будут выполнены быстрее, «Формирование отчетов» станет более эффективными.

Обновленная структура процессов:

1. Обработка и анализ заявок:
   1. Система автоматически распределяет заявки по ответственным лицам.
   2. Процесс анализа требований и выполнения задач ускоряется за счет автоматизации.
2. Формирование отчетов в веб версии:
   1. Система генерирует отчеты по заявкам, их статусу и результатам, что упрощает мониторинг и контроль.
   2. Отчеты формируются на основе актуальных данных, обеспечивая точность и своевременность.

Внедрение системы заявок НМЗ XIS позволит автоматизировать ключевые процессы подачи и обработки, что значительно повысит эффективность работы, снизит временные и трудовые затраты, а также улучшит качество обслуживания пользователей.

## **2.4 Построение диаграмм DFD (TO-BE)**

Для описания модели процесса обработки заявок в данной работе используются диаграммы бизнес-процессов DFD модели TO-BE.

Контекстная диаграмма (Рис. 2.19) представляет собой исследуемый объект предметной области под названием «Система заявок HM3 XIS».

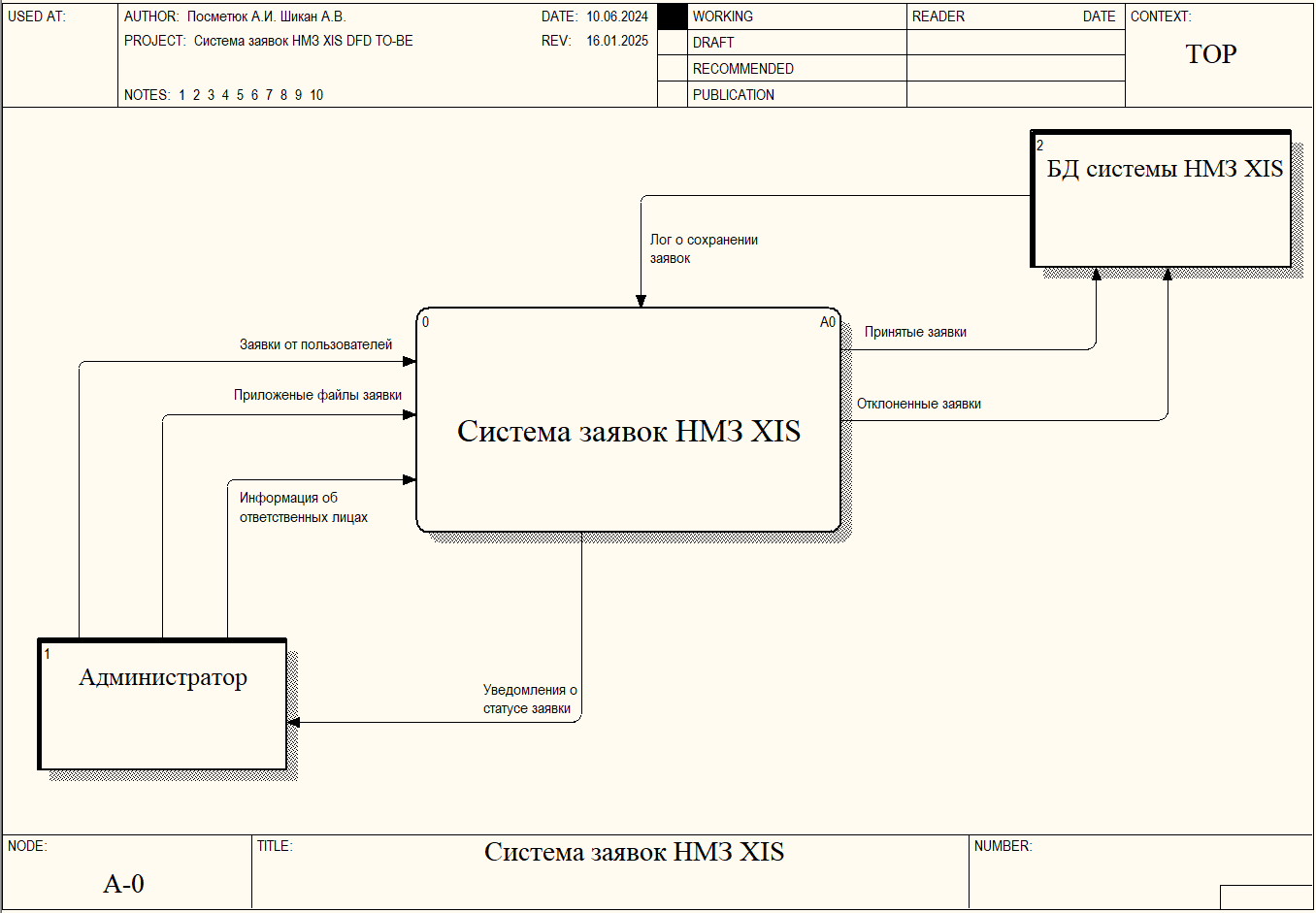


Рисунок 2.19 – Диаграмма DFD TO-BE A-0

В декомпозиции диаграммы показаны основные элементы процесса, включающие приём заявок и их обработку.

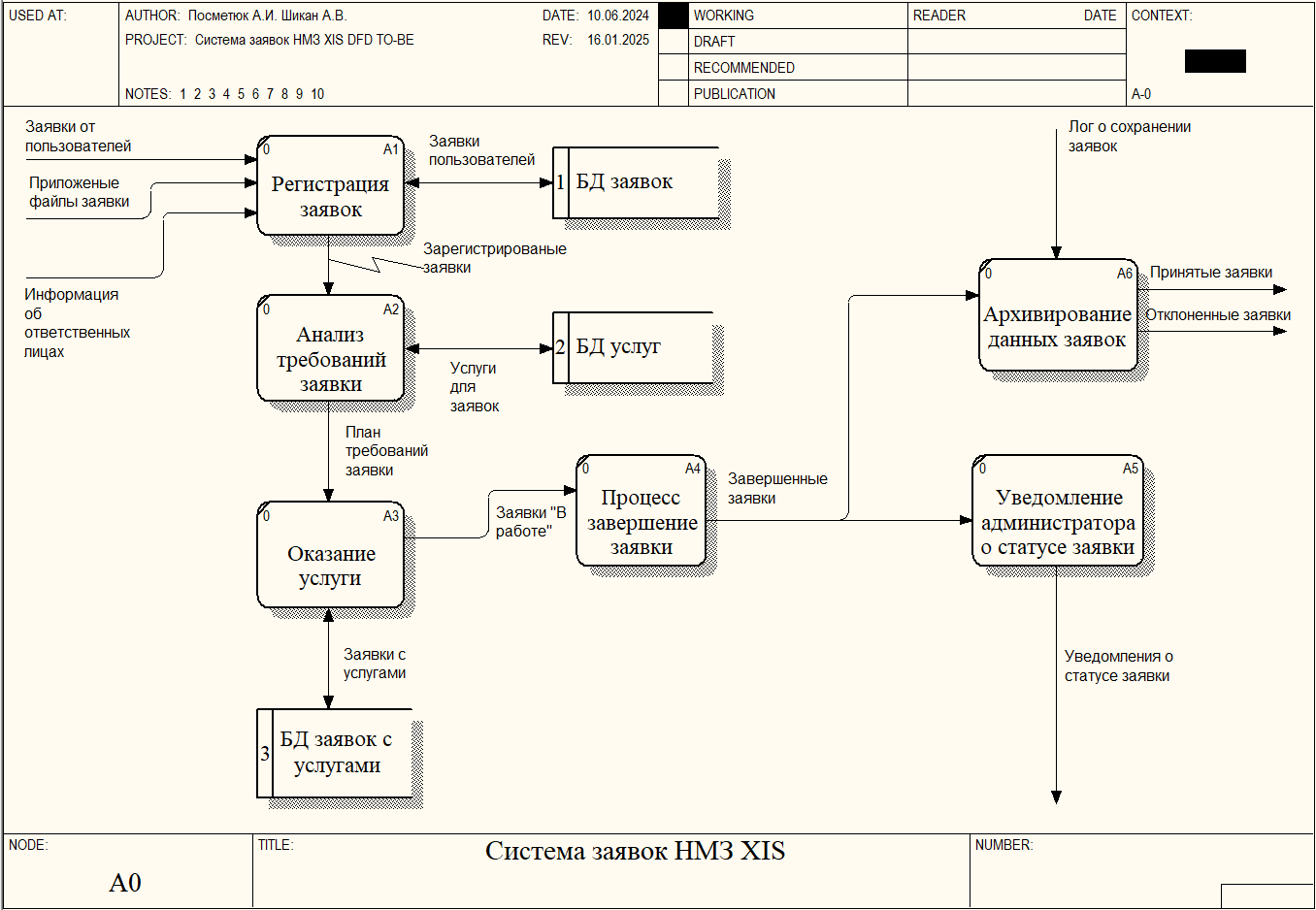


Рисунок 2.20 – Диаграмма IDEF0 TO-BE A0

На декомпозиции диаграммы представлены следующие основные этапы процесса:

1. Регистрация заявок:
   1. Приём заявок от пользователей, включая приложенные файлы и информацию о пользователях.
   2. Сохранение заявок в базе данных заявок.
2. Анализ требований заявки:
   1. Оценка и анализ требований, указанных в заявке.
   2. Формирование плана требований для дальнейшей обработки.
3. Оказание услуги:
   1. Определение конкретной услуги, необходимой для выполнения заявки.
   2. Выполнение услуги согласно заявке.
4. Процесс завершения заявки:
   1. Завершение обработки заявки.
   2. Архивирование данных заявки в базе данных.

Примеры описания элементов диаграммы:

* Регистрация заявок

На этом этапе система получает заявки от пользователей, сохраняет приложенные файлы и информацию о пользователях. Все данные сохраняются в базе данных заявок.

* Анализ требований заявки

Заявки проходят этап анализа, где оцениваются указанные в них требования. Это необходимо для формирования плана требований для дальнейшей обработки.

* Оказание услуги

На этом этапе система определяет необходимую услугу для выполнения заявки и выполняет её согласно заявке.

* Процесс завершения заявки

Завершение обработки заявки включает в себя выполнение всех необходимых работ и архивирование данных заявки в базе данных.

Такой стиль написания позволяет четко структурировать описание процесса и облегчает понимание каждого этапа обработки заявок.

## **2.5 Вывод по главе**

В данной главе было проведено инфологическое моделирование процесса подачи и обработки заявок на модернизацию системы НМЗ XIS. Обоснована необходимость автоматизации данного процесса для улучшения качества управления заявками и повышения эффективности работы сотрудников на Надеждинском металлургическом заводе им. Б. И. Колесникова.

Был сформирован документ по стратегии, который описывает цели и ожидаемые результаты проекта, границы новой системы и план ее реализации. В рабочем резюме предложено разработать мобильное приложение с использованием технологий React Native, Python, Docker, MiniO и Minikube для автоматизации процесса подачи и обработки заявок, что позволит сократить трудозатраты и уменьшить количество ошибок.

Проведен анализ текущей системы, в которой все заявки обрабатываются вручную, что приводит к высоким временным и трудовым затратам и частым ошибкам. Обоснована необходимость интеграции нового мобильного приложения с существующими корпоративными системами НМЗ XIS для обеспечения полной автоматизации процесса подачи, обработки.

В разделе делового и финансового субсидирования подчеркнуто, что проект имеет важное значение для оптимизации рабочих процессов на НМЗ XIS, несмотря на отсутствие финансирования. Сотрудники в настоящее время используют множество различных инструментов, однако общая работа остается трудоемкой и требует значительных временных затрат. Автоматизация процесса подачи и обработки заявок поможет преодолеть эти трудности, значительно уменьшив временные затраты и повысив общую эффективность.

Определены границы проекта, включающие автоматизацию подачи заявок, удобную реализацию их обработки, а также интеграцию с существующими системами НМЗ XIS для обеспечения полного контроля и управления процессом. Разработан рабочий план проекта с общей продолжительностью 77 дней и стоимостью 54 362 рубля.

Построены диаграммы IDEF0 TO-BE и DFD TO-BE, которые детализируют процесс автоматизации и описывают основные элементы процесса подачи и обработки. Эти диаграммы были сравнены с соответствующими диаграммами модели AS-IS, что позволило выявить основные изменения и их влияние.

По итогам анализа и моделирования установлено, что внедрение системы заявок НМЗ XIS приведет к значительным улучшениям в работе. Это позволит значительно повысить эффективность работы, снизить временные и трудовые затраты, а также улучшить качество обслуживания пользователей.

# **Глава 3. ПРОЕКТИРОВАНИЕ СТРУКТУРЫ БАЗЫ ДАННЫХ СИСТЕМЫ ЗАЯВКИ НМЗ XIS**

## **Логическая модель**

Логическая модель данных нашего мобильного приложения для модернизации системы НМЗ XIS представляет собой абстрактное описание данных, необходимых для управления заявками. Основными сущностями модели являются Роли, Пользователи, Заявки, Вложения и Услуги, а также связи между ними. Каждая сущность включает в себя набор атрибутов, которые определяют её свойства и характеристики.

Пользователи системы могут иметь различные роли, которые определяют их уровень доступа и возможности в системе. Роли обеспечивают механизм управления доступом и позволяют систематизировать права пользователей, обеспечивая безопасность и разграничение полномочий.

Заявки являются центральной сущностью системы, описывающей запросы на модернизацию системы НМЗ XIS. Каждая заявка имеет уникальный идентификатор, название, описание и различные статусы, позволяющие отслеживать её жизненный цикл от создания до завершения. Заявки могут содержать вложения, такие как документы и другие файлы, которые помогают в их обработке и выполнении. Вложения привязаны к заявкам и обеспечивают дополнительную информацию, необходимую для выполнения задач.

Услуги представляют собой действия или задачи, которые могут быть выполнены в рамках заявки. Они имеют свои уникальные идентификаторы, названия и описания, что позволяет чётко определить их суть и предназначение. Связь услуг с заявками позволяет структурировать процесс выполнения задач и обеспечивать гибкость в управлении заявками.

Таким образом, логическая модель данных нашего мобильного приложения демонстрирует, как данные о пользователях, их ролях, заявках, вложениях и услугах связаны между собой. Это позволяет структурировать данные, упрощать их управление и обеспечивать эффективное функционирование системы в целом.

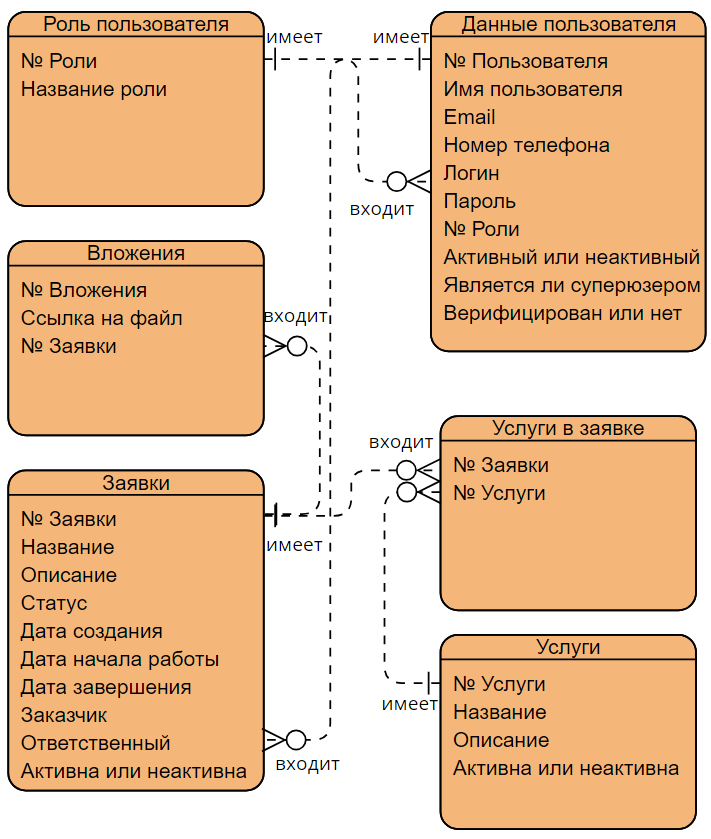


Рисунок 3.1 – Логическая модель базы данных системы

## **Физическая модель**

Физическая модель данных для мобильного приложения модернизации системы НМЗ XIS описывает, как логическая модель данных реализована на уровне конкретной СУБД PostgreSQL. Она включает в себя подробное описание структуры таблиц, типов данных, индексов и связей между таблицами, обеспечивая оптимальное хранение и доступ к данным.

Таблица "role" содержит информацию о ролях пользователей в системе, определяя их права доступа. Поле id используется в качестве первичного ключа, что обеспечивает уникальность каждой записи.

Таблица "user" хранит данные о пользователях системы, включая их личные данные и параметры безопасности. Внешний ключ role\_id связывает пользователя с его ролью, что позволяет управлять доступом и правами.

Таблица "application" содержит информацию о заявках на модернизацию системы НМЗ XIS. Она включает в себя идентификаторы, описания, статусы и даты, что позволяет отслеживать жизненный цикл каждой заявки от создания до завершения. Внешние ключи id\_executor и id\_application связывают заявки с исполнителями и вложениями соответственно.

Таблица "attachment" хранит ссылки на файлы, прикрепленные к заявкам. Поле id\_application используется для связи с соответствующей заявкой, обеспечивая целостность данных и упрощая доступ к связанным документам.

Таблица "service" описывает различные услуги, которые могут быть связаны с заявками. Она включает в себя идентификаторы, названия и описания услуг, что позволяет структурировать процесс выполнения задач.

Таблица "application\_service" связывает заявки с услугами, позволяя учитывать, какие услуги предоставляются в рамках каждой заявки. Внешние ключи application\_id и service\_id обеспечивают связь между таблицами "application" и "service".

Физическая модель данных является ключевым элементом при реализации базы данных в конкретной СУБД. Она определяет структуру хранения данных, индексы и связи, обеспечивая оптимальную производительность и надежность системы. В контексте нашего проекта физическая модель данных PostgreSQL позволяет эффективно управлять информацией о пользователях, ролях, заявках, вложениях и услугах, что способствует успешной модернизации системы НМЗ XIS.

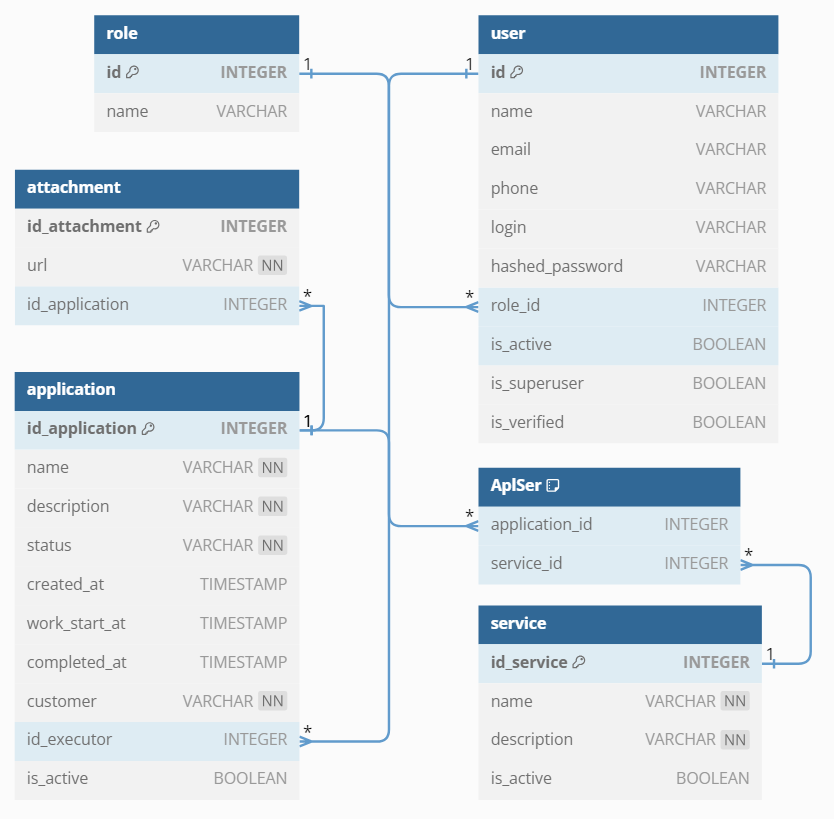


Рисунок 3.2 – Физическая модель базы данных системы

* 1. **Инфологическое проектирование БД**

Инфологическое проектирование базы данных представляет собой процесс сбора и анализа необходимой информации, которая будет использована в создаваемой информационной системе. Данный процесс включает в себя разработку моделей, которые помогают структурировать и визуализировать данные для эффективного управления информацией. Наша инфологическая модель включает такие сущности, как «Роль пользователя», «Данные пользователя», «Вложения», «Заявки», «Услуги в заявке» и «Услуги».

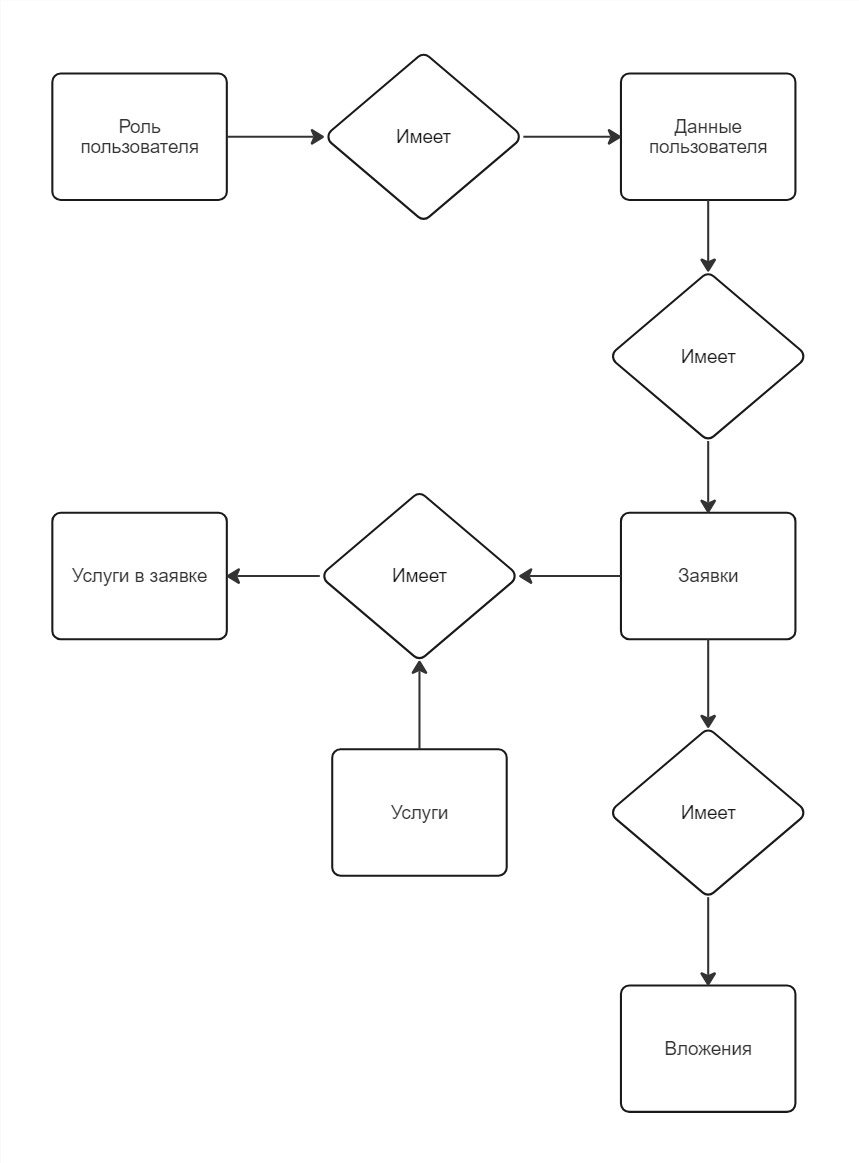


Рисунок 3.3 – Инфологическая модель базы данных системы

Выделение ключевых объектов системы:

1. **Роль пользователя**
   1. № Роли
   2. Название роли
2. **Данные пользователя**
   1. № Пользователя
   2. Имя пользователя
   3. Email
   4. Номер телефона
   5. Логин
   6. Пароль
   7. № Роли
   8. Активный или неактивный
   9. Является ли суперпользователем
   10. Верифицирован или нет
3. **Вложения**
   1. № Вложения
   2. Ссылка на файл
   3. № Заявки
4. **Заявки**
   1. № Заявки
   2. Название
   3. Описание
   4. Статус
   5. Дата создания
   6. Дата начала работы
   7. Дата завершения
   8. Заказчик
   9. Ответственный
   10. Активна или неактивна
5. **Услуги в заявке**
   1. № Заявки
   2. № Услуги
6. **Услуги**
   1. № Услуги
   2. Название
   3. Описание
   4. Активна или неактивна

Составление и комментирование уточнения ER-диаграммы (с атрибутами):

«Роль пользователя» обязательно имеет «Данные пользователя», а именно пользователя, которому принадлежит роль.

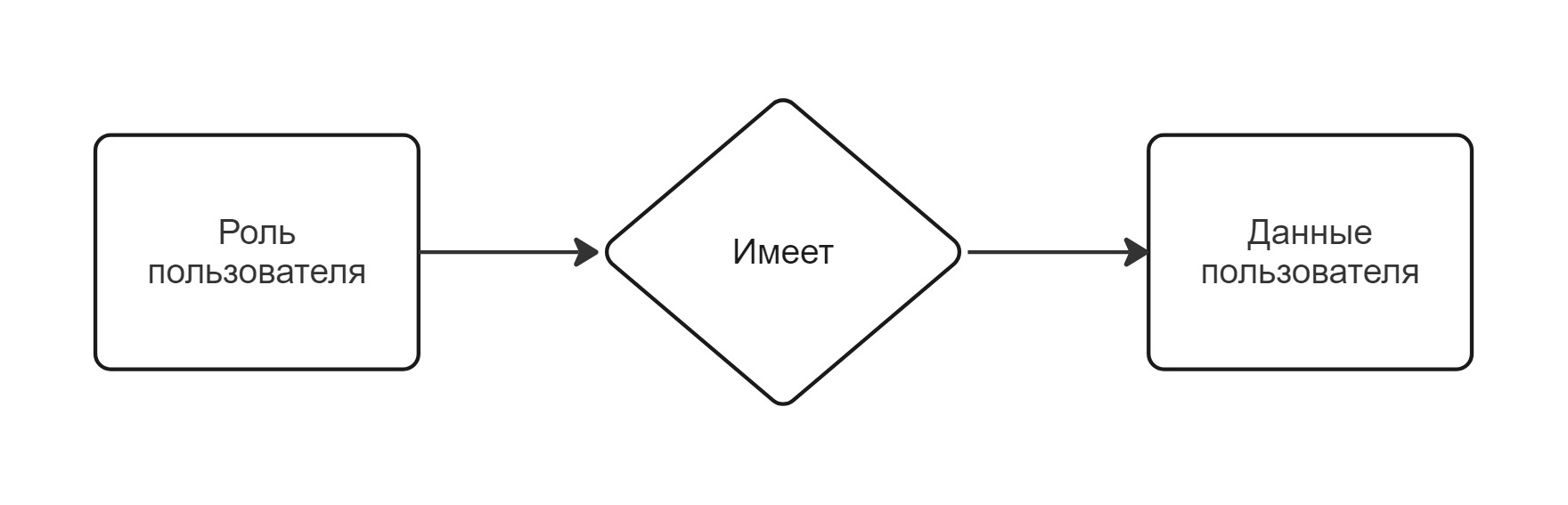


Рисунок 3.4 - Отношение «Роль пользователя» – «Данные пользователя»

К «Данные пользователя» обязательно принадлежит «Заявки», а именно заявки, где пользователь является ответственным.

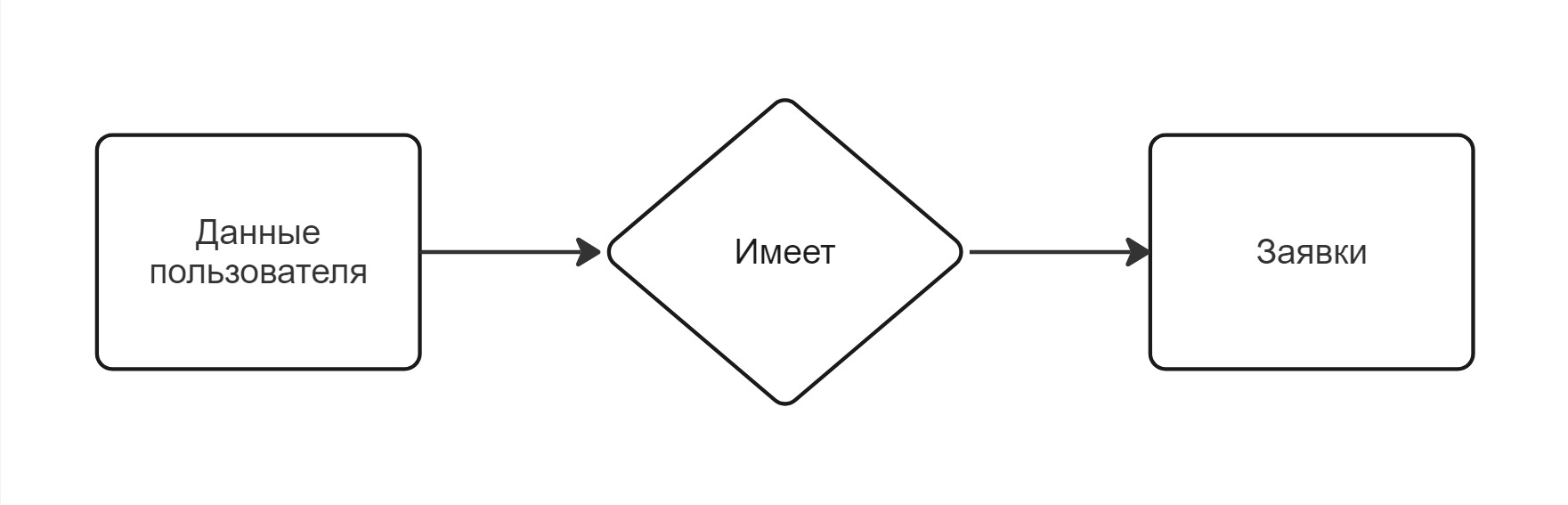


Рисунок 3.5 - Отношение «Данные пользователя» – «Заявки»

У каждой «Заявки» есть отношение к «Вложения», которые входят в заявку.

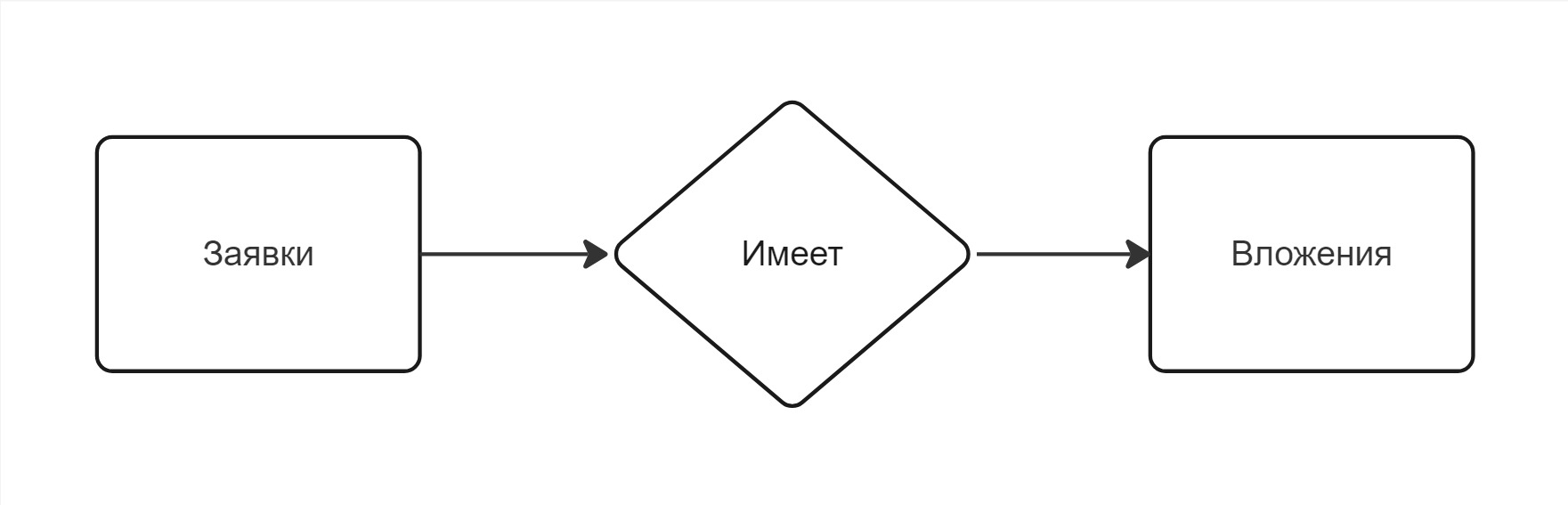
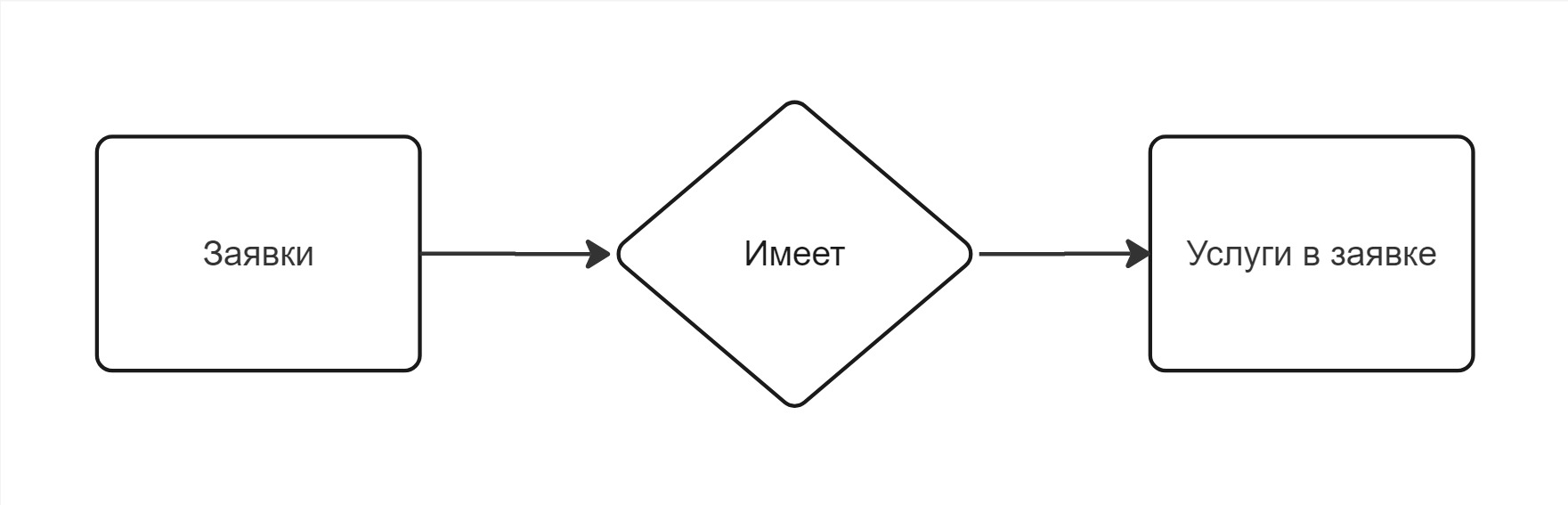


Рисунок 3.6 - Отношение «Заявки» – «Вложения»

В «Услуги в заявке» обязательно содержится содержимое «Заявки».

 Рисунок 3.7 - Отношение «Заявки» – «Услуги в заявки»

В «Услуги в заявке» обязательно содержится содержимое «Услуги».

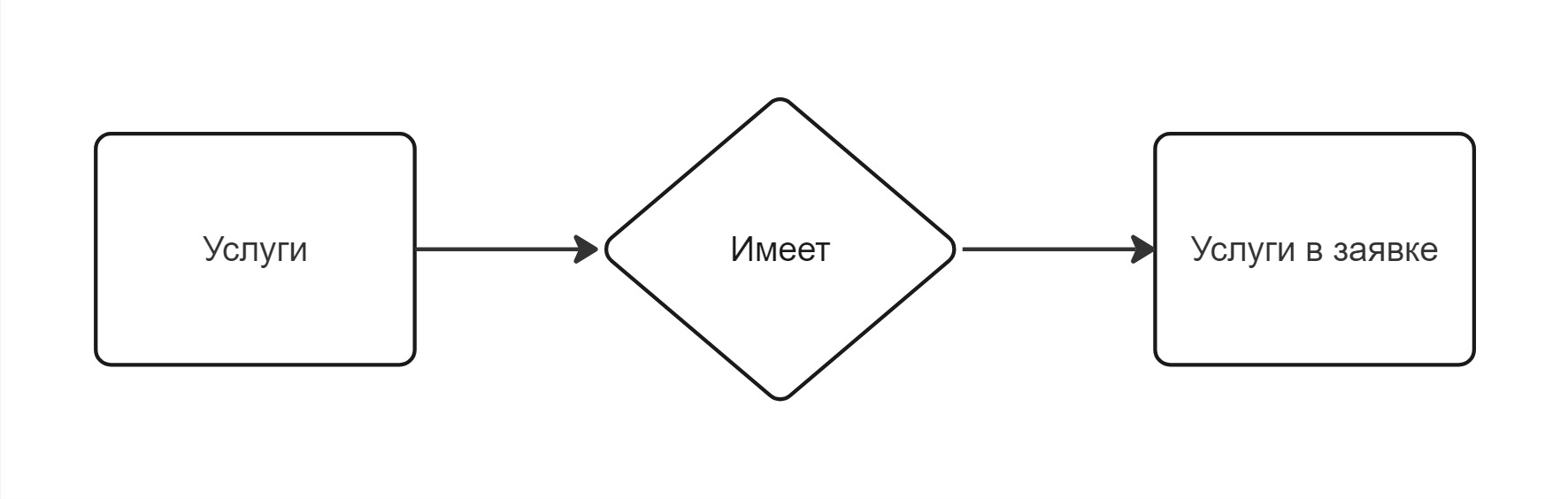


Рисунок 3.8 - Отношение «Услуги» – «Услуги в заявки»

# **3.4 Бизнес-процесс**

В рамках непрерывного стремления к оптимизации технологических процессов, важным шагом стала разработка нового мобильного приложения, предназначенного для автоматизации и систематизации процессов подачи, обработки и управления заявками на модернизацию системы НМЗ XIS. Это приложение обеспечивает более эффективное взаимодействие с заявками благодаря простоте и четкой структурированности.

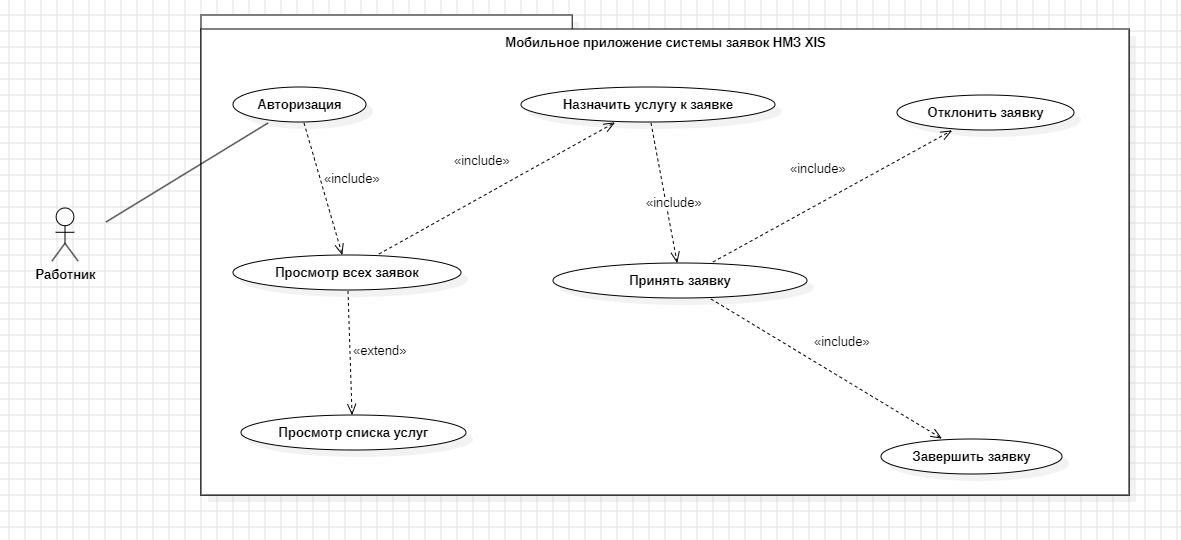
Приложение разработано как простая система управления заявками с возможностями авторизации, которая позволяет пользователям отправлять заявки системе по почте и получать ответ от приложения со статусом принятой заявки. Все заявки в системе классифицируются по четырём основным фильтрам:

1. Все заявки — общий список всех поданных заявок.
2. Наши заявки — заявки, поданные отделом или авторизованным пользователем.
3. Заявки в работе — заявки, которые в настоящий момент обрабатываются.
4. Завершенные заявки — заявки, которые были успешно обработаны или отклонены.

Каждая заявка маркирована соответствующим тегом, что облегчает её идентификацию и позволяет автоматически сортировать заявки в соответствии с указанными категориями.

Интерфейс приложения разработан таким образом, чтобы обеспечить максимальное удобство и простоту в использовании. Работники приложения могут легко принимать новые заявки в работу, завершить и отклонить, что делает процесс управления заявками гибким и адаптивным. После отправки, заявка фиксируется и переходит к модерации, где она может быть одобрена или отклонена ответственными лицами.

Для наглядного представления процессов управления заявками в приложении используются различные виды диаграмм. Диаграмма прецедентов (рис. 3.9) отображает взаимодействие пользователей с системой и основные сценарии использования приложения. Эта диаграмма помогает понять различные пользовательские роли и действия в системе.

Рисунок 3.9 - Диаграмма прецедентов

Далее, диаграмма деятельности (рис. 3.10) показывает последовательность действий пользователя при работе с заявками, начиная от создания и заканчивая отправкой на модерацию. Это помогает пользователям визуализировать процесс подачи и обработки заявок.



Рисунок 3.10 - Диаграмма деятельности

Особенно важной является диаграмма состояний заявки (рис 3.11), которая отражает все возможные состояния заявки от момента создания до её закрытия, включая переходы между статусами в зависимости от действий пользователя или архивиста. Эта диаграмма иллюстрирует жизненный цикл заявки, позволяя всем участникам процесса видеть текущее состояние и возможные следующие шаги.

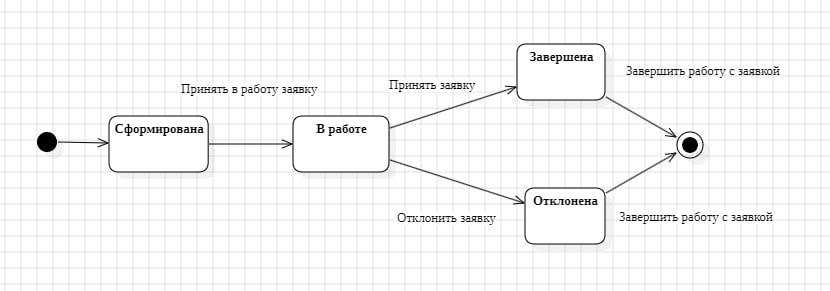


Рисунок 3.11 - Диаграмма состояний заявки

Таким образом, разработанное мобильное приложение не только существенно упрощает процесс подачи и обработки заявок на модернизацию системы НМЗ XIS, но и способствует более организованной и продуктивной работе всех участников процесса, повышая общую эффективность и удобство работы.

# **3.5 Архитектура**

В основе мобильной разработки лежит современная технология React Native. React Native предлагает значительные преимущества в скорости разработки благодаря быстрой перезагрузке модулей и оптимизированной сборке. Это позволяет нам эффективно управлять состоянием приложения и ускоряет процесс отладки и развертывания, сделав разработку интерфейса более гибкой и доступной.

Бекенд приложения развертывается в кластере Docker контейнеров, что обеспечивает однородность исполнения и изоляцию от внешних зависимостей, унифицируя развертывание на различных платформах. Docker облегчает масштабирование приложения и его обновление благодаря стандартизации окружения через контейнеры, включающие все необходимые зависимости. Конфигурация контейнеров задается в файле docker-compose.yml, что позволяет легко управлять сервисами в рамках проекта и детализировать параметры их запуска.

Сердцем системы хранения данных является СУБД PostgreSQL, выбранная за её надежность и масштабируемость. ER-диаграмма (рис. 3.13) показывает сложную структуру базы данных, включая таблицы пользователей, ролей, приложений и связанных с ними документов. Кроме того, для хранения больших объемов неструктурированных данных, таких как изображения и мультимедийные файлы, используется объектное хранилище MinIO. Это решение обеспечивает высокую производительность и отказоустойчивость, критически важные для современных мобильных приложений.

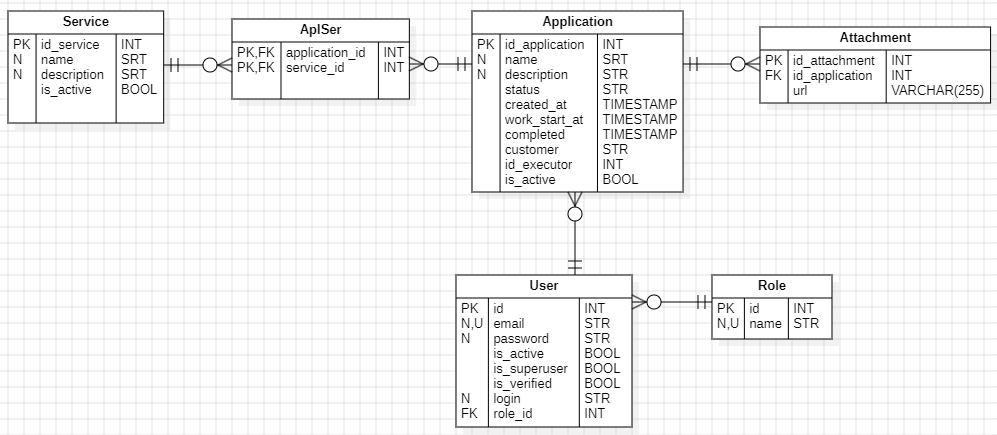


Рисунок 3.13 - ER диаграмма

Диаграмма классов бекенда (рис. 3.14) демонстрирует связи между основными моделями данных и их отображение в базе данных. Модель 'Application' связывается с моделями 'User' и 'Attachment', где каждое приложение может содержать несколько вложений, таких как документы или изображения. Эти связи управляются через таблицы связей, позволяя нам гибко обрабатывать множество отношений и обеспечивать целостность данных в приложении. Интеграция с внешними сервисами также реализована через специализированные связи, например, сервер статических файлов для хранения изображений.

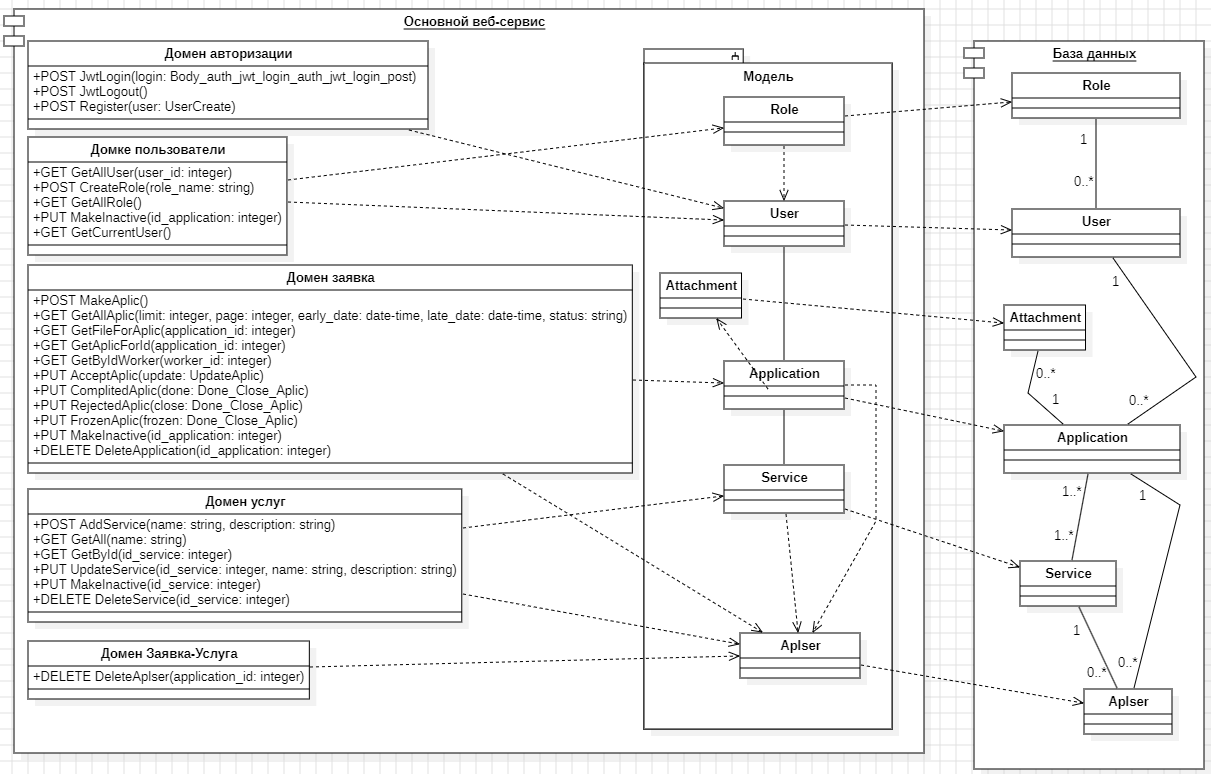
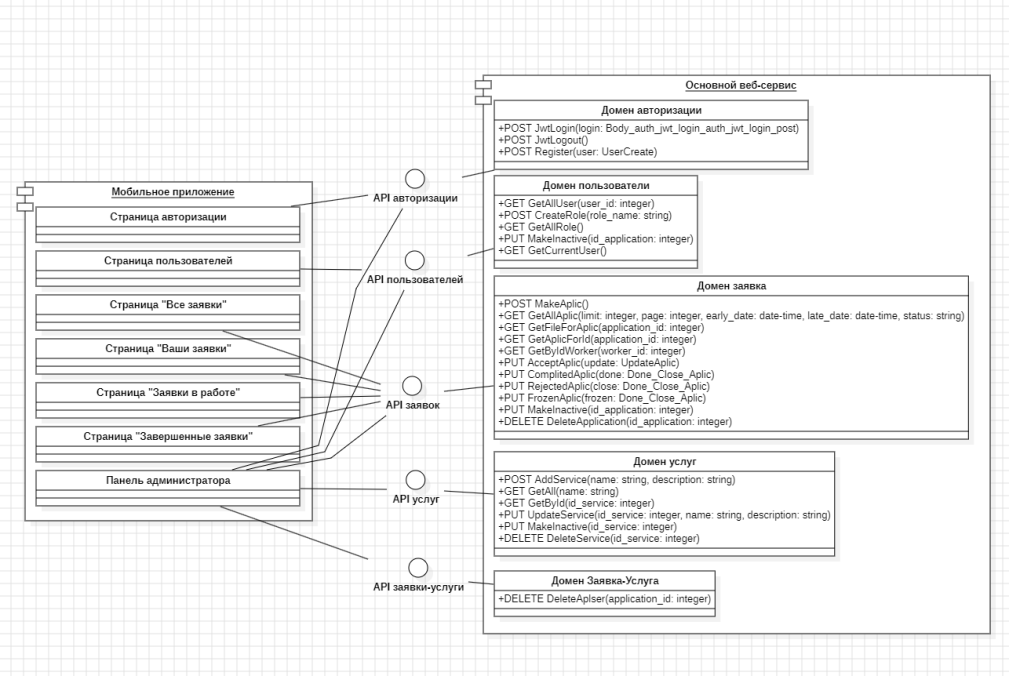


Рисунок 3.14 - Диаграмма классов бекенда

Таким образом, развертывание и архитектура нашего приложения представляют собой комплексное решение, которое охватывает не только разработку пользовательского интерфейса и бекенд-логику, но и детальное планирование инфраструктуры данных и безопасности. Это обеспечивает высокую производительность, надежность и масштабируемость системы, которая может адаптироваться к изменяющимся бизнес-требованиям и расти вместе с потребностями пользователя.

Связь мобильного приложения и бекенда отражена на диаграмме классов мобильного приложения (рис. 3.15). Панель администратора имеет доступ ко всем страницам так как администратор контролирует абсолютно все процессы работы нашего мобильного приложения.

Рисунок 3.15 *-* Диаграмма классов мобильного приложения

# **3.6 Алгоритмы**

В начале бизнес-процесса происходит аутентификация работника. Для этого он отправляет запрос на мобильный web-сервис через React-приложение, передавая в нем почту, логин и пароль. Если данные корректны, в ответном запросе пользователь получает токен, что позволяет ему продолжить работу с системой. В случае ошибки в данных пользователю предоставляется информация о необходимости корректировки входных данных.

Далее, работник запрашивает текущий список активных заявок и доступных услуг, используя свой доступный токен для аутентификации запросов. Документы и информация о заявках возвращаются в JSON формате, что упрощает их отображение и дальнейшую обработку в графическом интерфейсе.

Когда работник выбирает активные заявки, он может выбрать для них статус «в работе», «заморожена» на неопределенный срок, или «отклонить», позже он смотрит вложение в заявке, ими могут быть различные файлы такие как Jpeg изображения, word файл, pdf и другие. После полного анализа заявки работник ее завершает и переходит к следующей.

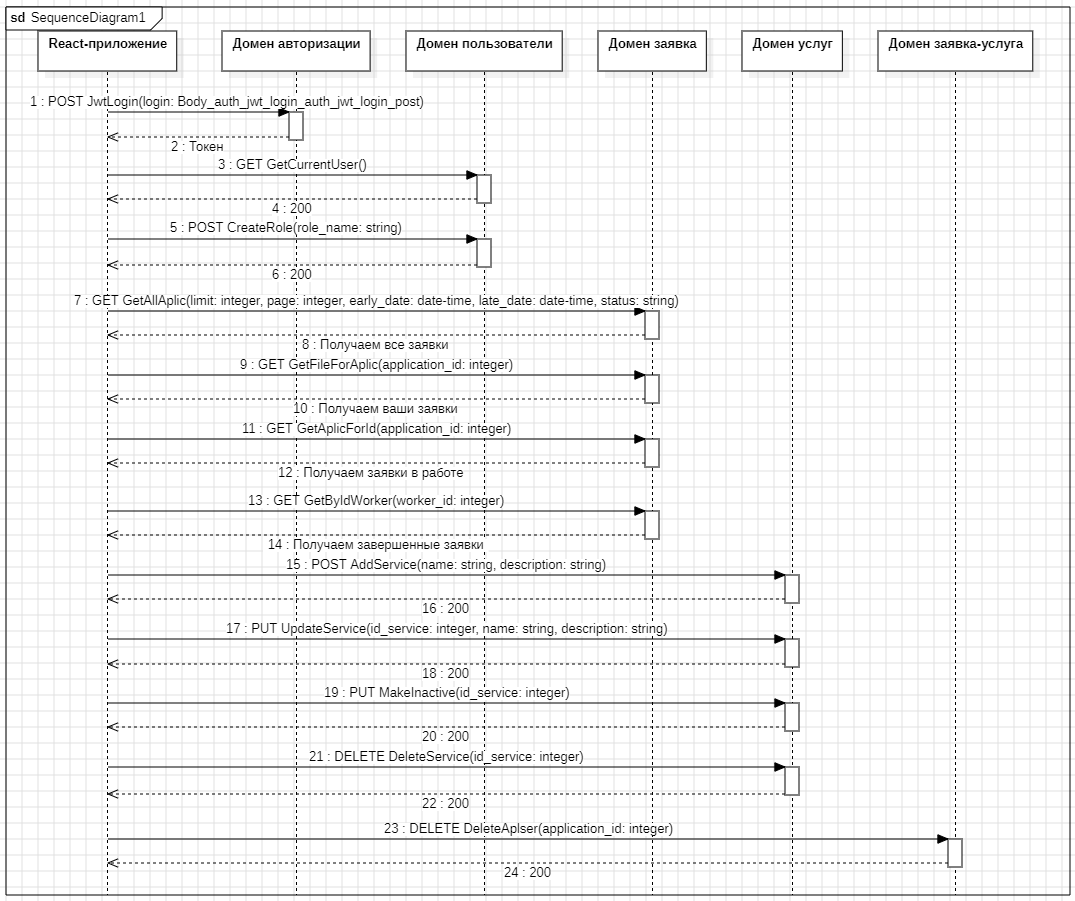


Рисунок 3.16 - Диаграмма последовательности

Процесс рассмотрения и управления заявками осуществляется через графический интерфейс администратора. Администраторы имеют возможность просматривать все поданные заявки, фильтровать их по критериям, указанным ранее, таким как «Все заявки», «Ваши заявки», «Заявки в работе» и «Завершенные заявки». Они также могут одобрять или отклонять заявки, а также управлять ими — создавать новые, редактировать существующие или удалять их. Каждое из этих действий представляет собой отдельный вызов API к мобильному web-сервису, что гарантирует централизованное управление и обработку данных.

Весь процесс интегрирован с основным мобильным web-сервисом, который обрабатывает логику бизнес-процессов и взаимодействует с различными доменами системы, включая пользователей, заявки и услуги. Эта модульная структура обеспечивает гибкость и масштабируемость системы, позволяя эффективно обрабатывать запросы и управлять данными в соответствии с развивающимися бизнес-требованиями.

Данная архитектура представляет собой комплексное решение, позволяющее не только эффективно обрабатывать операции в режиме реального времени, но и гарантирует высокий уровень безопасности и удобства для пользователей при работе с системой.

# **Глава 4. ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА МОБИЛЬНОГО ПРИЛОЖЕНИЯ СИСТЕМЫ ЗАЯВОК НМЗ XIS**

# **4.1 Мобильное приложение системы заявок НМЗ XIS**

Начальный экран приложения (рис. 4.1) встречает пользователя формой авторизации и не дает доступа к функционалу сайта без входа в систему.

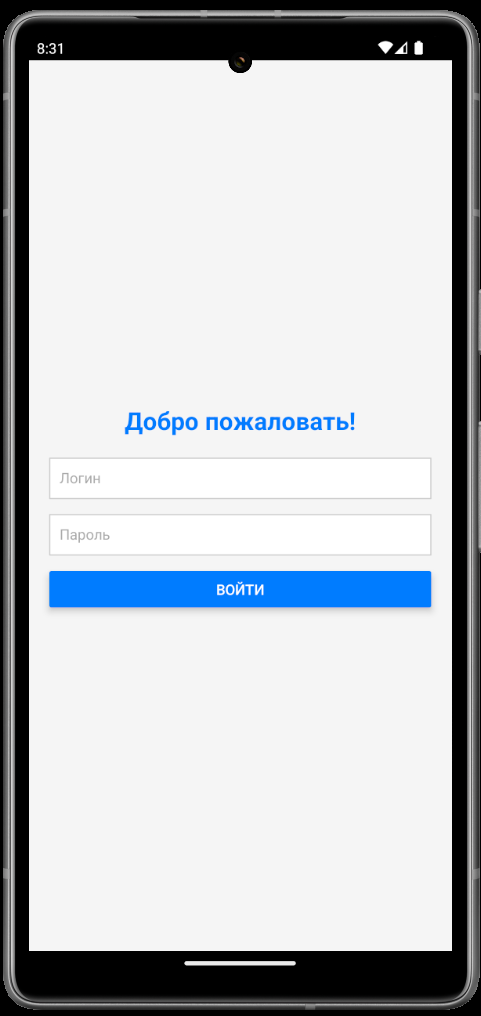


Рисунок 4.1 – Авторизация пользователя

После авторизации пользователь переходит на главный экран приложения (рис. 4.2) с разными элементами интерфейса.

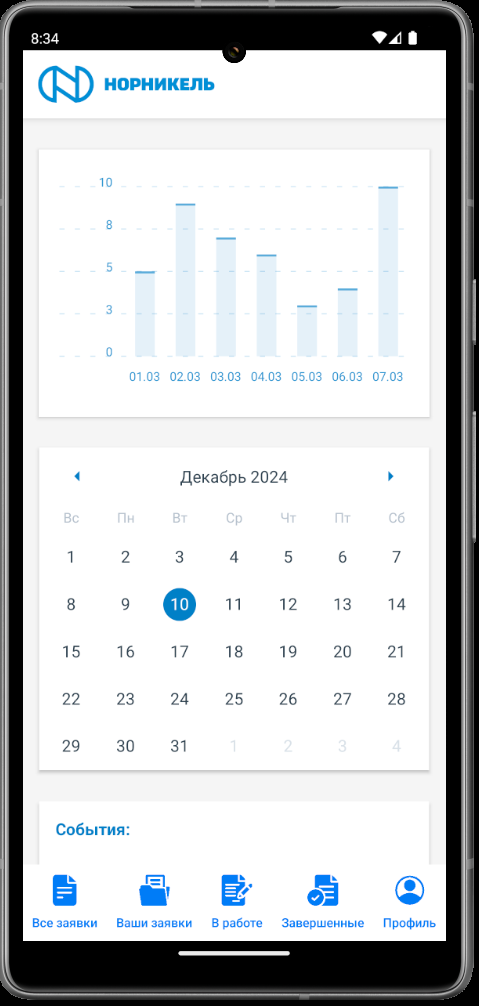


Рисунок 4.2 – Главный экран приложения

Хеадер (рис. 4.3) содержит содержит логотип «Норникеля».



Рисунок 4.3 – Хеадер приложения

Футер экрана (рис. 4.4), который переносит на страницу с таблицами заявок, каждая из кнопок фильтрует содержимое таблиц в соответствии с названием: «Все заявки», «Ваши заявки», «Заявки в работе» и «Завершенные заявки».

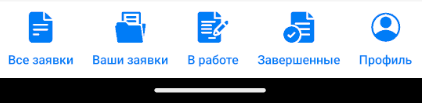


Рисунок 4.4 – Футер приложения

Экран профиля (рис. 4.5) содержит «аватар», фамилию с инициалами и должность пользователя, ниже представлена более подробная информация, а именно, «полное имя» пользователя, «Email адрес», «номер телефона» и «адрес» проживания.

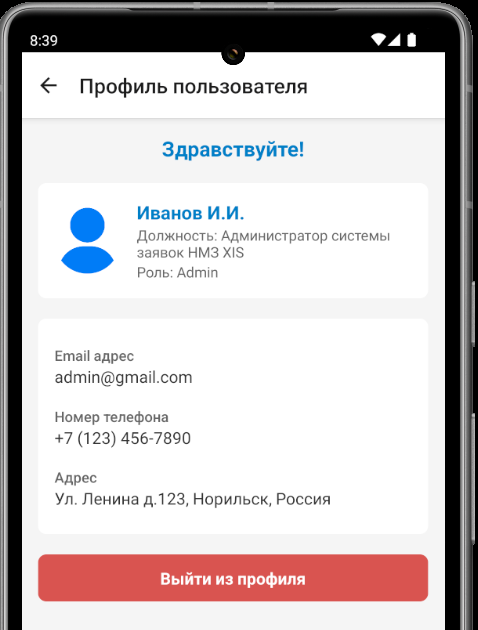


Рисунок 4.5 – Экран профиля пользователя

На главном экране также представлено окно графика (рис. 4.6) количества выполненных заявок на сайте в день.

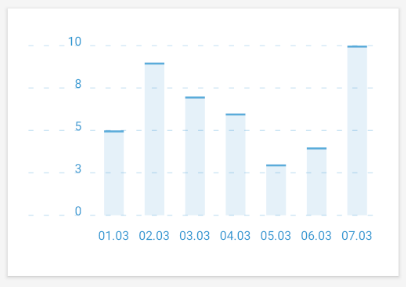


Рисунок 4.6 – График выполнения заявок

Ниже окно с календарем (рис. 4.7) и с заметками (рис. 4.8) где находятся напоминания для пользователя, которые сортируются по времени и показываются в конкретную дату, пользователь получает заметки от руководителя или добавляет их самостоятельно.

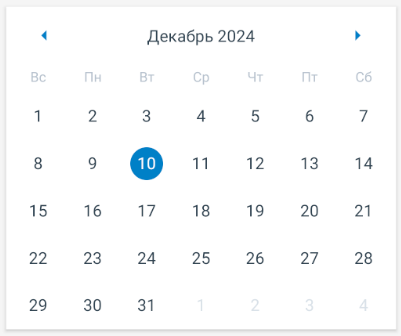
****

Рисунок 4.7 – Окно с календарем и заметками

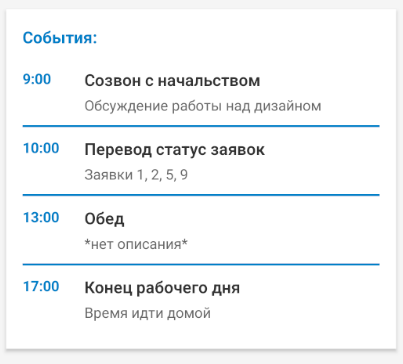


Рисунок 4.8 – График выполнения заявок

При переходе на экран «Все заявки» (рис. 4.9) открывается экран со всеми заявками которые есть в приложении, каждая заявка имеет название, описание, статус, ответственный и дата создания заявки, также на странице реализована пагинация, которая оставляет на странице 5 заявок и переводит оставшиеся заявки на следующие, а над заявками есть кнопка сортировки (рис. 4.10) которая позволяет сортировать список заявки по признаками которые имеют заявки.

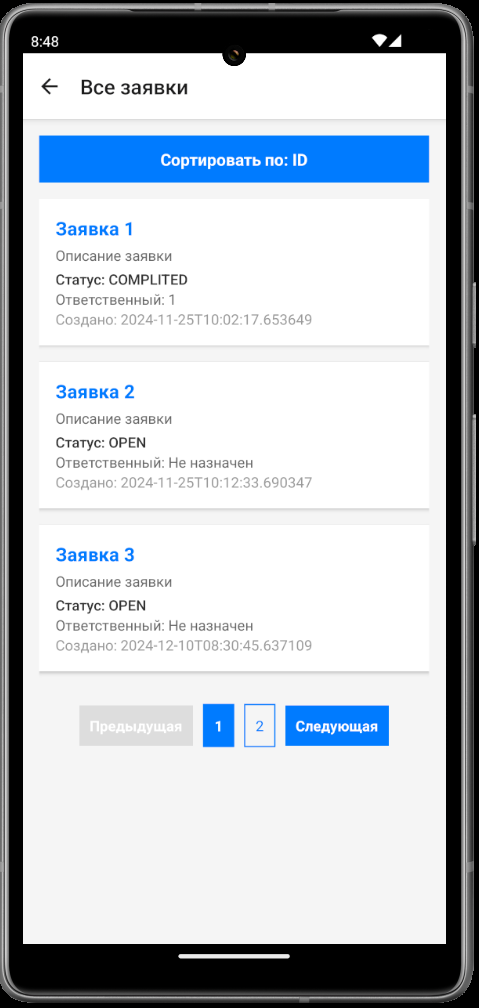
****

Рисунок 4.9 – Экран «Все заявки»

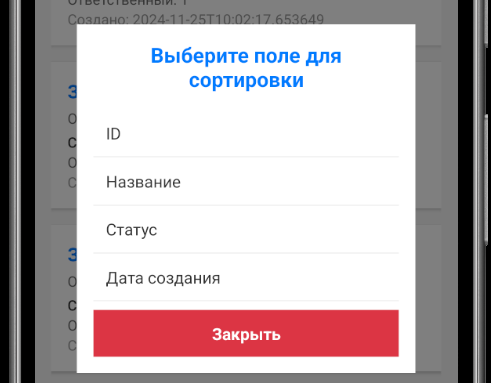


Рисунок 4.10 – Окно сортировки списка заявок

На экране «Ваши заявки» (рис. 4.11) отображаются только те заявки, за которые ответственен авторизированный пользователь, в данном примере показаны только заявки с id авторизированного пользователя, а именно «1».

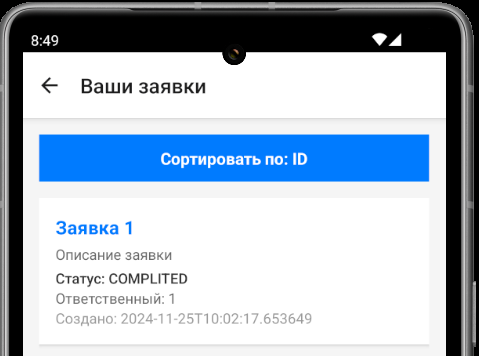


Рисунок 4.11 - Экран «Ваши заявки»

На экране «Заявки в работе» (рис. 4.12) отображаются только те заявки, которые имеют статус «ATWORK».

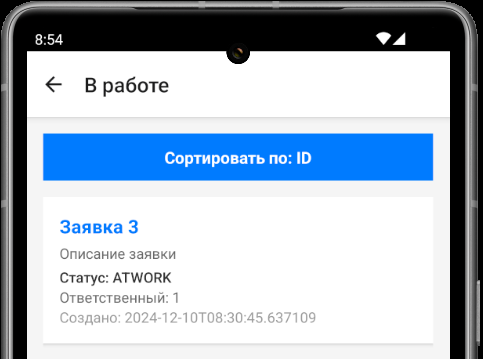


Рисунок 4.12 - Экран «Заявки в работе»

На последней экране «Завершенные заявки» (рис. 4.13) отображаются только заявки со статусом «COMPLITED».

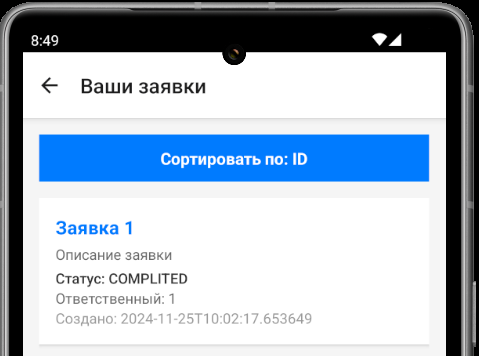


Рисунок 4.13 - Экран «Завершенные заявки»

При нажатии на заявку мы переходим на экран самой заявки «Детали заявки» (рис. 4.14), где находится уже полная информация о заявке, ее содержимое, также ниже расположена кнопка с возможностью выбора оказываемой услуги «Оказать услугу» выбора услуг (рис. 4.15) появится другая кнопка «Сменить статус» (рис. 4.16), позволяющая сменить статус заявки на «COMPLITED» или «REJECTED» (рис. 4.17).

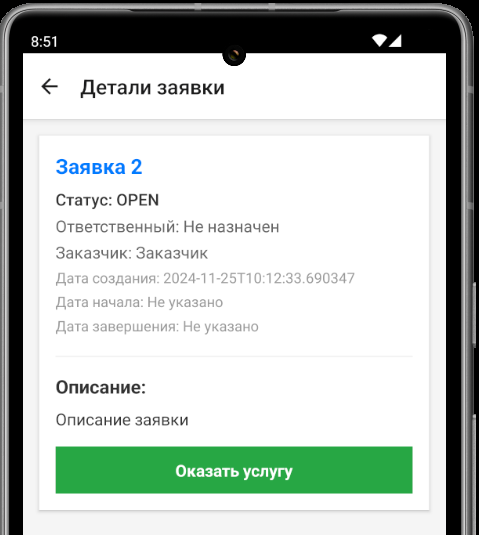


Рисунок 4.14 – Экран «Детали заявки»

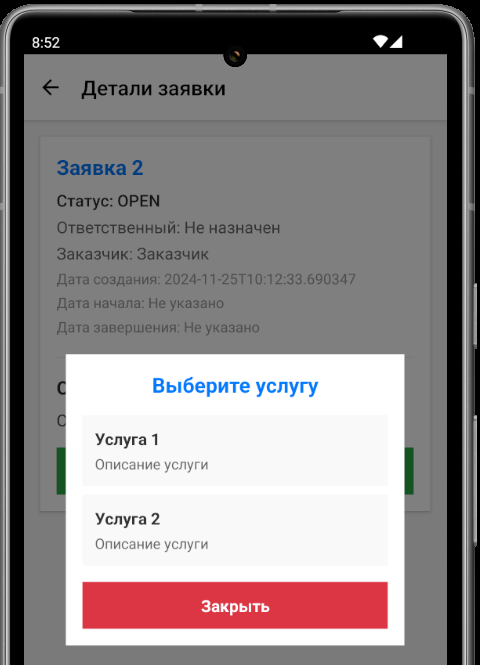


Рисунок 4.15 – Выбор услуг

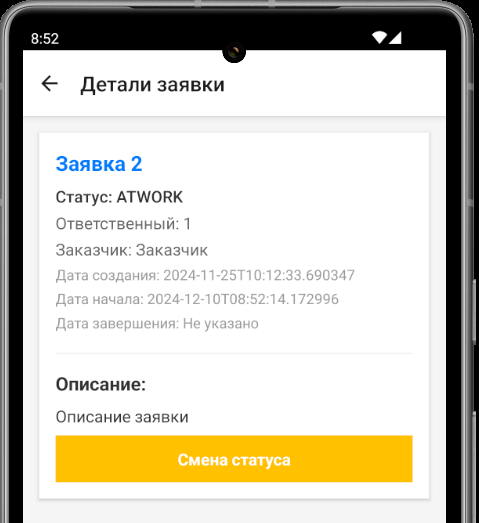


Рисунок 4.16 – Экран «Детали заявки» с кнопкой «Смена статуса»

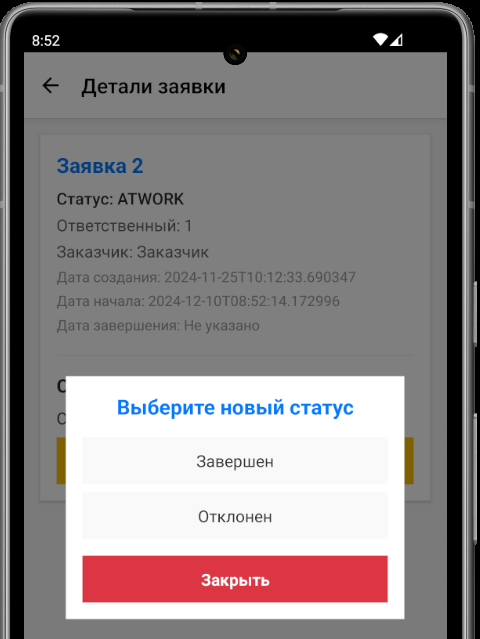


Рисунок 4.17 – Выбор статуса

# **БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

1. Карпова, Т.С. Базы данных. Модели, разработка, реализация; СПб: Питер, 2001. - 304 c.

2. Данелян, Т.Я. Теория систем и системный анализ: Учебно-методический комплекс / Т.Я. Данелян. - М.: Ленанд, 2016. - 360 c.

3. Кориков, А.М. Теория систем и системный анализ: Учебное пособие / А.М. Кориков, С. Н. Павлов. - М.: Инфра-М, 2016. - 416 c.

4. Хомяков, П.М. Системный анализ: Экспресс-курс лекций / П. М. Хомяков. - М.: Ленанд, 2017. - 214 c.

5. Методические указания к выполнению курсового проекта по дисциплинам базы данных, управление данными: учебное пособие НГТУ, 2007 г...

6. Трипутина, В.В. Проектирование баз данных с помощью Case-средства ErWin. Методические указания к выполнению лабораторных работ: учебное пособие

7. С. Г. Фомичёва, А. А. Маслова. Клиент-серверные базы данных. Курс лекций для студентов специальностей 171900 и 351400 всех форм обучения / Норильский индустриальный институт. – Норильск, 2004. – 141 с.

8. Доусон М. Программируем на Python. – СПб.: Питер, 2014. – 416 с.

9. Лутц М. Изучаем Python, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 1280 с.

10. Лутц М. Программирование на Python, том I, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с.

11. Лутц М. Программирование на Python, том II, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с.

12. Саймон Ригс, Ханну Кросинг «Администрирование Postgre SQL 9» ДМК Пресс, 2012, 368 стр., пер. с англ. Самохвалова Е.

13. Дарнелл, Р. JavaScript: справочник / Р. Дарнелл. - М.: Питер, 2018. - 192 c.

14. Пьюривал, Сэмми Основы разработки web-приложений / Сэмми Пьюривал. - М.: Питер, 2020. - 568 c.

15. Николас, Закас JavaScript. Оптимизация производительности / Закас Николас. - М.: Символ-плюс, 2019. - 944 c.

16. Макфарланд, Дэвид JavaScript. Подробное руководство / Дэвид Макфарланд. - М.: Эксмо, 2020. - 608 c.

17. Кантелон, М. Node.js в действии / М. Кантелон. - М.: Питер, 2020. - 788 c.

18. Бэнкс, Порселло- React и Redux: функциональная web-разработка 2018 Алекс Бэнкс, Ева Порселло- 336 с.

19. React [Электронный ресурс]. URL: https://reactjs.org/ (дата обращения: 18.05.2021)

20. Документация PostgreSQL и Postgres Pro [Электронный ресурс]. URL: https://postgrespro.ru/docs(дата обращения: 10.05.2021)

# **ПРИЛОЖЕНИЕ**

**1.** **Методы мобильного web-сервиса**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Метод** | **Описание** | **URL** | **Входные данные** | **Выходные данные** |
| **1.1. User** | | | | | |
| 1.1.1 | GET | Получаем всех пользователей | http://localhost:9999/api/user/get\_all\_user | user\_id | [  {  "id": 0,  "email": "string",  "is\_active": true,  "is\_superuser": false,  "is\_verified": false,  "login": "string",  "role\_id": 0  }  ] |
| 1.1.2 | POST | Создаем роль | http://localhost:9999/api/user/create\_role?role\_name=1 | role\_name | {  "id": 0,  "name": "string"  } |
| 1.1.3 | GET | Получаем все роли | http://localhost:9999/api/user/get\_all\_role | No parameters | [  {  "id": 0,  "name": "string"  }  ] |
| 1.1.4 | PUT | Сделать неактивным пользователя | http://localhost:9999/api/user/make\_inactive?id=5 | id | "string" |
| 1.1.5 | GET | Получаем авторизированного пользователя устройства | http://localhost:9999/api/user/current | No parameters | {  "id": 0,  "email": "string",  "is\_active": true,  "is\_superuser": false,  "is\_verified": false,  "login": "string",  "role\_id": 0  } |
| **1.2. Aplication** | | | | | |
| 1.2.1 | POST | Создаем заявку автоматически | http://localhost:9999/api/aplication/make\_apli | No parameters | "string" |
| 1.2.2 | POST | Создаем заявку вручную | http://localhost:9999/api/aplication/make\_apli\_admin?name=1&description=1&customer=1 | name  description  customer | "string" |
| 1.2.3 | GET | Получаем все заявки | http://localhost:9999/api/aplication/get\_all\_aplic?limit=1&page=1&early\_date=1&late\_date=1&status=1 | limit  page  early\_date  late\_date  status | [  {  "status": "OPEN",  "id\_application": 5,  "work\_start\_at": null,  "customer": "MuRchi4me <pendospoganiy@gmail.com>",  "is\_active": true,  "description": "Пример текста содержимого",  "created\_at": "2024-06-10T08:51:37.811866",  "name": "Проблемы туда сюда",  "completed\_at": null,  "id\_executor": null  }  ] |
| 1.2.4 | GET | Получаем все закрепленные файлы заявки | http://localhost:9999/api/aplication/get\_file\_for\_aplic?application\_id=1 | application\_id | [  "http://localhost:9000/bucket5/121.xlsx?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=minio%2F20240610%2Fus-east-1%2Fs3%2Faws4\_request&X-Amz-Date=20240610T085137Z&X-Amz-Expires=604800&X-Amz-SignedHeaders=host&X-Amz-Signature=2307e47d04b3bbd9ddcb67dda5bf365039d6913670b4f5eebdbfa0f39d5dc6fc",  "http://localhost:9000/bucket5/323.docx?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=minio%2F20240610%2Fus-east-1%2Fs3%2Faws4\_request&X-Amz-Date=20240610T085137Z&X-Amz-Expires=604800&X-Amz-SignedHeaders=host&X-Amz-Signature=8667e43f02b4a8114943048060fba18d397acf16fbbc90f61f8e884e3dda4acd"  ] |
| 1.2.5 | GET | Получаем заявку по ее id | http://localhost:9999/api/aplication/get\_apllic\_for\_id?application\_id=1 | application\_id | {  "status": "OPEN",  "id\_application": 5,  "work\_start\_at": null,  "customer": "MuRchi4me <pendospoganiy@gmail.com>",  "is\_active": true,  "description": "Пример текста содержимого",  "created\_at": "2024-06-10T08:51:37.811866",  "name": "Проблемы туда сюда",  "completed\_at": null,  "id\_executor": null  },  []  ] |
| 1.2.6 | GET | Получаем заявку по id ответственного | http://localhost:9999/api/aplication/get\_by\_idworker?worker\_id=1 | worker\_id | {  "status": "OPEN",  "id\_application": 5,  "work\_start\_at": null,  "customer": "MuRchi4me <pendospoganiy@gmail.com>",  "is\_active": true,  "description": "Пример текста содержимого",  "created\_at": "2024-06-10T08:51:37.811866",  "name": "Проблемы туда сюда",  "completed\_at": null,  "id\_executor": 2  },  []  ] |
| 1.2.7 | PUT | Принимаем заявку и оказываем услугу | http://localhost:9999/api/aplication/accept\_aplic | {  "application\_id": 0,  "new\_status": "Open",  "id\_service": [  0  ]  } | "string" |
| 1.2.8 | PUT | Заявка в статусе «Завершена» | http://localhost:9999/api/aplication/done\_aplic | {  "application\_id": 0,  "new\_status": "Open"  } | "string" |
| 1.2.9 | PUT | Заявка в статусе «Отклонена» | http://localhost:9999/api/aplication/close\_aplic | { "application\_id": 0,  "new\_status": "Open"  } | "string" |
| 1.2.10 | PUT | Сделать заявку неактивной | http://localhost:9999/api/aplication/make\_inactive?id\_application=1 | id\_application | "string" |
| 1.2.11 | DELETE | Удалить заявку | http://localhost:9999/api/aplication/delete\_application?id\_application=3 | id\_application | "string" |
| **1.3. Service** | | | | | |
| 1.3.1 | POST | Добавить услугу | http://localhost:9999/api/service/add\_service?name=1&description=1 | name  description | [  {  "id\_service": 2,  "name": "1",  "description": "1",  "is\_active": true  }  ] |
| 1.3.2 | GET | Получаем все услуги | http://localhost:9999/api/service/get\_all?name=1 | name | [  {  "id\_service": 1,  "name": "1",  "description": "1",  "is\_active": true  }  {  "id\_service": 2,  "name": "1",  "description": "1",  "is\_active": true  }  ] |
| 1.3.3 | GET | Получаем услуги по id | http://localhost:9999/api/service/get\_by\_id?id\_service=1 | id\_service | [  {  "id\_service": 2,  "name": "1",  "description": "1",  "is\_active": true  }  ] |
| 1.3.4 | PUT | Изменить услугу | http://localhost:9999/api/service/update\_service?id\_service=2&name=1&description=1 | id\_service  name  description | [  {  "id\_service": 2,  "name": "1",  "description": "1",  "is\_active": true  }  ] |
| 1.3.5 | PUT | Сделать услугу неактивной | http://localhost:9999/api/service/make\_inactive?id\_service=2 | id\_service | "string" |
| 1.3.6 | DELETE | Удалить услугу | http://localhost:9999/api/service/delete\_service?id\_service=2 | id\_service | "string" |
| **1.4. Aplser** | | | | | |
| 1.4.1 | POST | Удалить все услуги у заявки | http://localhost:9999/api/aplser/delete\_aplser?application\_id=1 | application\_id | "string" |
| **1.5. Auth** | | | | | |
| 1.5.1 | POST | Залогиниться | http://localhost:9999/api/auth/jwt/login | grant\_type  username  password  scope  client\_id  client\_secret | access-control-allow-credentials: true access-control-allow-origin: http://localhost:9999 date: Mon,10 Jun 2024 10:34:07 GMT server: uvicorn vary: Origin |
| 1.5.2 | POST | Разлогиниться | http://localhost:9999/api/auth/jwt/logout |  | access-control-allow-credentials: true access-control-allow-origin: http://localhost:9999 date: Mon,10 Jun 2024 10:34:30 GMT server: uvicorn vary: Origin |
| 1.5.3 | POST | Зарегистрировать пользователя | http://localhost:9999/api/auth/register | {  "email": "admin",  "password": "string",  "is\_active": true,  "is\_superuser": false,  "is\_verified": false,  "login": "string",  "role\_id": 1  } | {  "id": 2,  "email": "admin",  "is\_active": true,  "is\_superuser": false,  "is\_verified": false,  "login": "string",  "role\_id": 1  } |

**2. Функциональные требования:**

2.1. Страница с формой авторизации. На данной странице отображается окно с формой заполнения для авторизации.

2.1.1. Доступна гостям

2.1.2. Действия:

2.1.2.1. Войти – перенаправляет на страницу 5.2

2.2. Главное меню**.** На этой странице находится текст с приветствием, если участник не авторизирован, и меню со списком доступных пунктов.

2.2.1. Доступна всем авторизированным пользователям.

2.2.2. Действия:

2.2.2.1. Модальное окно авторизированного пользователя

2.2.2.2. Переход на страницы 2.3, 2.4, 2.5 и 2.6

2.3. Страница со списком всех заявок. На этой странице располагается список всех заявок и панель их фильтрации.

2.3.1. Доступна всем авторизированным пользователям.

2.3.2. Выводится информация о заявках в виде карточек:

2.3.2.1. Название заявки.

2.3.2.2. Описание заявки.

2.3.2.3. Дата создания заявки.

2.3.2.4. Дата начала.

2.3.2.5. Дата завершения.

2.3.2.6. Заказчик.

2.3.2.7. Ответственный.

2.3.2.8. Статус.

2.3.2.9. Подробнее.

2.3.3. Действия:

2.3.3.1. Подробнее – перенаправляет на страницу 5.7.

2.3.3.2. Вернуться на главную страницу – позволяет пользователю вернутся на главную страницу.

2.4 Страница со списком своих заявок. На этой странице отображаются только заявки, поданные авторизированным пользователем.

2.4.1. Доступна только авторизированным пользователям.

2.4.2. Выводится информация о заявках в виде карточек:

2.4.2.1. Название заявки.

2.4.2.2. Описание заявки.

2.4.2.3. Дата создания заявки.

2.4.2.4. Дата начала.

2.4.2.5. Дата завершения.

2.4.2.6. Заказчик.

2.4.2.7. Ответственный.

2.4.2.8. Статус.

2.4.2.9. Подробнее.

2.4.3 Действия:

2.4.3.1. Подробнее – перенаправляет на страницу 5.7.

2.4.3.2. Вернуться на главную страницу – позволяет пользователю вернутся на главную страницу.

2.5. Страница со списком заявок в работе. На этой странице отображаются только заявки, которые находятся в работе.

2.5.1. Доступна только авторизированным пользователям.

2.5.2. Выводится информация о заявках в виде карточек:

2.5.2.1. Название заявки.

2.5.2.2. Описание заявки.

2.5.2.3. Дата создания заявки.

2.5.2.4. Дата начала.

2.5.2.5. Дата завершения.

2.5.2.6. Заказчик.

2.5.2.7. Ответственный.

2.5.2.8. Статус.

2.5.2.9. Подробнее.

2.5.3. Действия:

2.5.3.1. Подробнее – перенаправляет на страницу 5.7.

2.5.3.2. Вернуться на главную страницу – позволяет пользователю вернутся на главную страницу.

2.6. Страница со списком завершенных заявок. На этой странице отображаются только завершенные заявки.

2.6.1. Доступна только авторизированным пользователям.

2.6.2. Выводится информация о заявках в виде карточек:

2.6.2.1. Название заявки.

2.6.2.2. Описание заявки.

2.6.2.3. Дата создания заявки.

2.6.2.4. Дата начала.

2.6.2.5. Дата завершения.

2.6.2.6. Заказчик.

2.6.2.7. Ответственный.

2.6.2.8. Статус.

2.6.2.9. Подробнее.

2.6.3. Действия:

2.6.3.1. Подробнее – перенаправляет на страницу 5.7.

2.6.3.2. Вернуться на главную страницу – позволяет пользователю вернутся на главную страницу.

2.7. Страница с подробным описанием заявки. На этой странице отображается полная информация о выбранной заявке, включая прикрепленные файлы.

2.7.1. Доступна только авторизированным пользователям.

2.7.1. Выводится информация о заявке в построчном формате

2.7.2.1. Название заявки.

2.7.2.2. Описание заявки.

2.7.2.3. Дата создания заявки.

2.7.2.4. Дата начала.

2.7.2.5. Дата завершения.

2.7.2.6. Заказчик.

2.7.2.7. Ответственный.

2.7.2.8. Описание.

2.7.3 Действия:

2.7.3.1. Выбрать услугу – позволяет выбрать статус для услуги, список берется из бэкэнда и перевести статус заявки в «В работе».

2.7.3.2. Изменить статус заявки – позволяет изменить статус заявки на «завершена» или «отклонена».

2.8. Страница редактирования/создания заявки. На этой странице администратор может создать новую заявку или изменить существующую.

2.8.1 Доступна только администраторам.

2.8.2 Выводится информация о заявке в построчном формате:

2.8.2.1. Название заявки.

2.8.2.2. Описание заявки.

2.8.2.3. Дата создания заявки.

2.8.2.4. Дата начала.

2.8.2.5. Дата завершения.

2.8.2.6. Заказчик.

2.8.2.7. Ответственный.

2.8.2.8. Описание.

2.8.3. Действия:

2.8.3.1. Выбрать услугу – позволяет выбрать статус для услуги, список берется из бэкэнда и перевести статус заявки в «В работе».

2.8.3.2. Изменить статус заявки – позволяет изменить статус заявки на «завершена» или «отклонена».

2.8.3.3. Сделать неактивной – сделает заявку неактивной, и она не будет отображаться в таблице.

2.8.3.4. Удалить заявку – удаляет заявку полностью.

2.8.3.5. Удалить статусы заявки – удаляет все статусы заявки.

2.9. Страница управлением пользователями. На этой странице администратор управляет пользователями и создавать роли.

2.9.1 Доступна только администраторам.

2.9.2 Выводится информация о всех пользователях:

2.9.2.1. ID.

2.9.2.2. Email.

2.9.2.3. Логин.

2.9.2.4. Роль.

2.9.2.5. Активен.

2.9.2.6. Действие.

2.9.3 Выводится информация о всех ролях:

2.9.3.1. ID.

2.9.3.2. Название.

2.9.4. Действия:

2.8.3.1. Деактивировать пользователя – позволяет выбрать пользователя чей аккаунт станет неактивным.

2.8.3.2. Создать роль – позволяет создать новую роль.

2.10. Страница со списком всех заявок. На этой странице располагается список всех заявок и панель их фильтрации.

2.10.1. Доступна только администраторам.

2.10.2. Выводится информация о заявках в виде карточек:

2.10.2.1. Название заявки.

2.10.2.2. Описание заявки.

2.10.2.3. Дата создания заявки.

2.10.2.4. Дата начала.

2.10.2.5. Дата завершения.

2.10.2.6. Заказчик.

2.10.2.7. Ответственный.

2.10.2.8. Статус.

2.10.2.9. Подробнее.

2.10.3. Действия:

2.10.3.1. Подробнее – перенаправляет на страницу 5.7.

2.10.3.1. Создать новую заявку – позволяет создать новую заявку с полями «название», «описание» и «заказчик».

2.10.3.3. Создать новую заявку auto – создает заявку автоматически путем Email.

2.10.3.4. Режим auto mode – автоматически принимает заявки, если такие есть, каждые 2 минуты.

2.10.3.5. Вернуться на главную страницу – позволяет администратору вернутся на главную страницу.

2.11. Страница управлением услугами. На этой странице администратор управляет услугами для заявок.

2.11.1 Доступна только администраторам.

2.11.2 Выводится информация о всех услугах:

2.11.2.1. ID.

2.11.2.2. Название.

2.11.2.3. Описание.

2.11.3. Действия:

2.11.3.1. Поиск услуги по ID – позволяет найти нужную услугу по ID.

2.11.3.2. Редактировать – позволяет создать новую роль.

2.11.3.3. Сделать неактивной – позволяет сделать заявку неактивной.

2.11.3.4. Удалить – позволяет удалить заявку.

**3. Требования к программному обеспечению:**

3.1. Серверная часть

- ОС: Linux (2.x.x) / Windows (7 и выше)

- Docker: (4.23.x и выше)

- Web-сервер MinIO: (8.4.x и выше)

- Web-сервер Redis: (7.2 и выше)

- Python: 3.11.5

- СУБД PostgreSQL: (16.0 и выше)

3.2. Клиентская часть

- ОС: Android

**4. Требования к аппаратному обеспечению:**

4.1. Серверная часть

- Процессор: минимум 4-ядерный с частотой от 2 ГГц

- Оперативная память: от 8 Гб

- Место на жестком диске: от 1 Гб

4.2. Клиентская часть

- Процессор: с частотой от 2 ГГц

- Оперативная память: от 1 Гб