



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

**Институт информационных технологий (ИТ)
Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

КУРСОВАЯ РАБОТА

по дисциплине: Проектирование клиент-серверных систем

по профилю: Разработка программных продуктов и проектирование информационных систем

направления профессиональной подготовки: 09.03.04 «Программная инженерия»

Тема: Проектирование ИС «Трамвайное депо»

Студент: Фомичев Роман Алексеевич

Группа: ИКБО-20-21

Работа представлена к защите _____ (дата) _____ /Фомичев Р.А. /
(подпись и ф.и.о. студента)

Руководитель: Рачков Андрей Владимирович, ст. преп.

Работа допущена к защите _____ (дата) _____ /Рачков А.В. /
(подпись и ф.и.о. рук-ля)

Оценка по итогам защиты: _____

_____/_____/_____/_____/_____/_____/

(подписи, дата, ф.и.о., должность, звание, уч. степень двух преподавателей, принявших защиту)

Руководитель курсовой работы: ст. преп. Рачков Андрей Владимирович

Фомичев Р.А., Курсовая работа направления подготовки «Программная инженерия» на тему «Проектирование ИС „Трамвайное депо“»: М., 2024 г., МИРЭА – Российский технологический университет (РТУ МИРЭА), Институт информационных технологий (ИТ), кафедра инструментального и прикладного программного обеспечения (ИиППО) – 29 стр., 15 рис., 15 источн.

Ключевые слова: проектирование, трамвайное депо, клиент-серверные системы, UML, IDEF0, Entity Relationship, База данных, Модель, Бизнес процесс, Диаграмма.

Целью работы является проектирование комплексной информационной системы для автоматизации процессов трамвайного депо. Разработаны модели процессов и архитектура клиент-серверной системы, логическая модель базы данных, а также техническая документация.

Fomichev R.A., Course Work for the program "Software Engineering" on the topic "Design of the information system "Tram depot": Moscow, 2024, MIREA – Russian Technological University (RTU MIREA), Institute of Information Technologies (IT), Department of Instrumental and Applied Software (IiPPO) – 29 pages, 15 figures, 15 sources.

Keywords: design, tram depot, client-server systems, UML, IDEF0, Entity Relationship, Database, Model, Business Process, Diagram.

The purpose of the work is to design a comprehensive information system to automate tram depots processes. Process models, client-server system architecture, database`s logical model, and technical documentation were developed.

РТУ МИРЭА: 119454, Москва, пр-т Вернадского, д. 78

Кафедра инструментального и прикладного программного обеспечения (ИиППО)

Тираж: 1 экз. (на правах рукописи)

Файл: «ПЗ_ПКСС_ИКБО_20_21_ФомичевРА.docx», исполнитель Фомичев Р.А.

© Фомичев Р.А.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1 РАЗРАБОТКА КОНЦЕПТУАЛЬНОЙ МОДЕЛИ КЛИЕНТ-СЕРВЕРНОЙ СИСТЕМЫ.....	7
1.1 Идентификация предметной области автоматизации	7
1.2 Выбор методологии и технологии концептуального моделирования клиент-серверной системы.....	7
1.3 Разработка и анализ модели бизнес-процесса «Как есть».....	8
1.4 Разработка модели бизнес-процесса «Как должно быть»	11
1.5 Разработка требований к клиент-серверной системе.....	13
1.6 Обзор и анализ аналогичных клиент-серверных систем	14
1.7 Постановка задачи на разработку новой клиент-серверной системы ...	15
1.8 Общие сведения	15
1.9 Вывод.....	17
2 РАЗРАБОТКА ЛОГИЧЕСКОЙ МОДЕЛИ КЛИЕНТ-СЕРВЕРНОЙ СИСТЕМЫ.....	18
2.1 Выбор методологии и технологии логического моделирования клиент-серверной системы.....	18
2.2 Разработка диаграмм логической модели клиент-серверной системы .	20
2.3 Разработка модели клиент-серверных потоков в системе.....	24
2.4 Разработка логической модели.....	24
2.5 Вывод.....	25
ЗАКЛЮЧЕНИЕ	26
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ	27

ВВЕДЕНИЕ

В современных условиях автоматизация процессов и цифровизация являются важнейшими направлениями развития различных сфер деятельности. Эффективная организация работы и оптимизация бизнес-процессов позволяют существенно повысить продуктивность предприятий и улучшить пользовательский опыт. Одной из таких областей, требующей модернизации, является управление трамвайным депо, где сложности в координации персонала, планировании маршрутов, техническом обслуживании и финансовом учете создают значительные трудности.

Эти проблемы обуславливаются сложностью и громоздкостью традиционных подходов к управлению процессами, что приводит к задержкам в ремонте, неравномерной загрузке маршрутов и недостаточной прозрачности финансовых потоков. Для их решения целесообразно использовать современные информационные системы с клиент-серверной архитектурой, которые обеспечивают эффективное управление всеми аспектами работы депо.

Цель данной курсовой работы – проектирование клиент-серверной системы, позволяющей автоматизировать ключевые процессы трамвайного депо. Это позволит повысить оперативность принятия решений, минимизировать ошибки, связанные с человеческим фактором, и улучшить качество обслуживания.

Для достижения поставленной цели необходимо решить следующие задачи:

- 1) провести анализ предметной области;
- 2) описать пользователей, входные и выходные потоки;
- 3) описать существующие процессы «AS-IS»;
- 4) создать функциональную модель клиент-серверной системы «TO-BE»;

- 5) разработать архитектуру клиент-серверной системы;
- 6) создать модель базы данных клиент-серверной системы;
- 7) оформить пояснительную записку по курсовой работе;
- 8) подготовить презентацию выполненной курсовой работы.

В ходе выполнения работы были использованы следующие методы: сравнение, анализ, классификация, обобщение, описание и моделирование.

Работа состоит из введения, оглавления, аннотации, глоссария, двух основных разделов, заключения и списка использованных источников.

1 РАЗРАБОТКА КОНЦЕПТУАЛЬНОЙ МОДЕЛИ КЛИЕНТ-СЕРВЕРНОЙ СИСТЕМЫ

1.1 Идентификация предметной области автоматизации

В качестве основы для автоматизации было выбрано трамвайное депо, включающее несколько подразделений: ремонтные мастерские, диспетчерскую, и зоны хранения запасных частей. Обычно управление депо осуществляется различными сотрудниками, каждый из которых отвечает за свой участок работы, организуя процессы на основе установленных регламентов.

1.2 Выбор методологии и технологии концептуального моделирования клиент-серверной системы

Для дальнейшего построения бизнес-процессов необходимо выбрать методологии моделирования [1,2,3]. Для решения этой задачи была построена сводная таблица 1, в которой представлены наиболее распространённые варианты [4].

Таблица 1 – Характеристики методологий моделирования бизнес-процессов

Методология	Назначение	Преимущества	Недостатки
IDEF0	Моделирование функциональных аспектов верхнего уровня	Структурированность, стандартизация, простота	Не подходит для временных аспектов
DFD	Анализ поток данных между процессами	Детализация информационных потоков	Ограниченность в функциональном моделировании
IDEF3	Документирование сценариев и последовательность событий	Фокус на временных аспектах	Сложность интеграции с другими методологиями
UML	Объектно-ориентированное моделирование сложных систем	Гибкость, поддержка множества диаграмм	Сложность для анализа простых процессов
BPMN	Моделирование бизнес-процессов с высокой детализацией	Универсальность, согласованность между бизнесом и ИТ	Высокая сложность, избыточность

После проведения анализа преимуществ и недостатков была выбрана методология IDEF0, так как она универсальна и интуитивно понятная, что позволит на дальнейших этапах создания системы использовать её для обозначения планируемого функционала для остальных участников проекта, также данная методология ставит фокус на функциональности, что является главной целью итоговой информационной системы и не мало важно то, что IDEF0 является международным стандартом, что упрощает интеграцию модели в существующие процессы.

Для дальнейшего построения бизнес-процессов по методологии IDEF0 было выбрано CASE-средство [5] Ramus, которое представляет удобный и понятный интерфейс для построения различных диаграмм в том числе и IDEF0.

1.3 Разработка и анализ модели бизнес-процесса «Как есть»

Для проведения анализа существующих процессов была построена диаграмма по методологии IDEF0, которая представлена на рисунках 1.1 – 1.6.

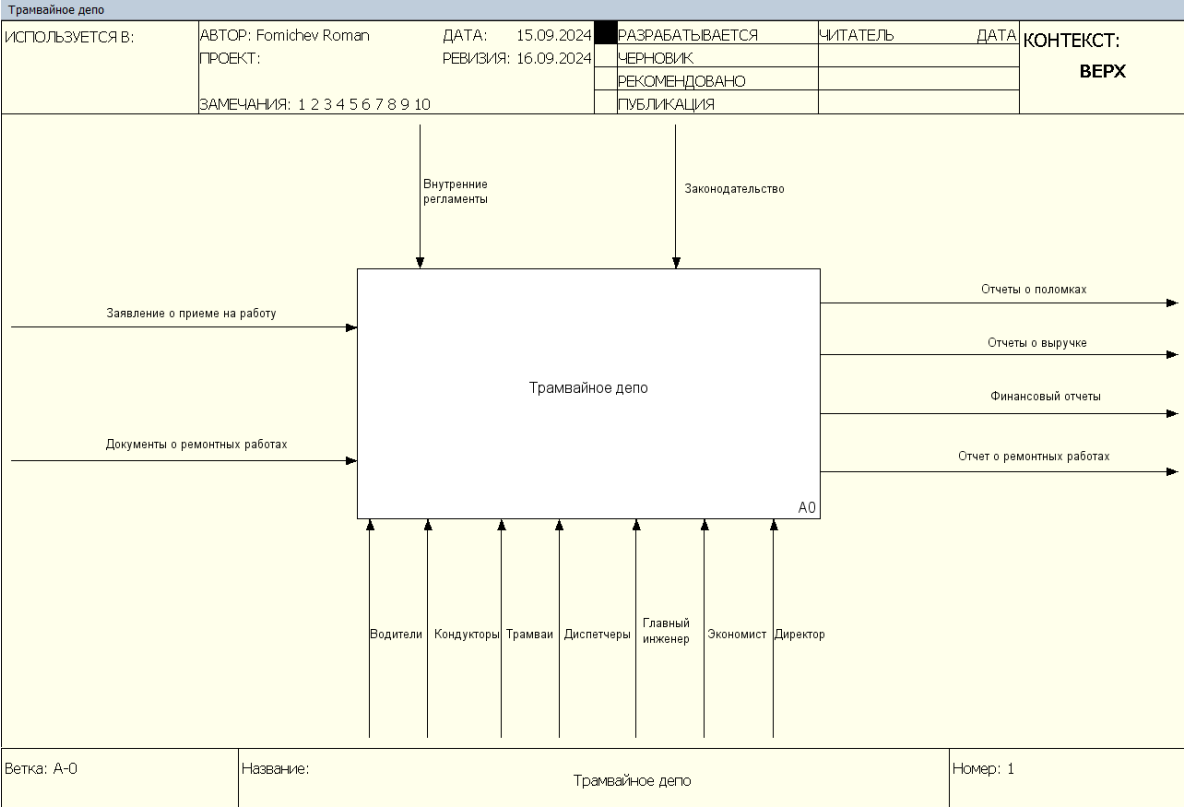


Рисунок 1.1 – Верхний уровень диаграммы

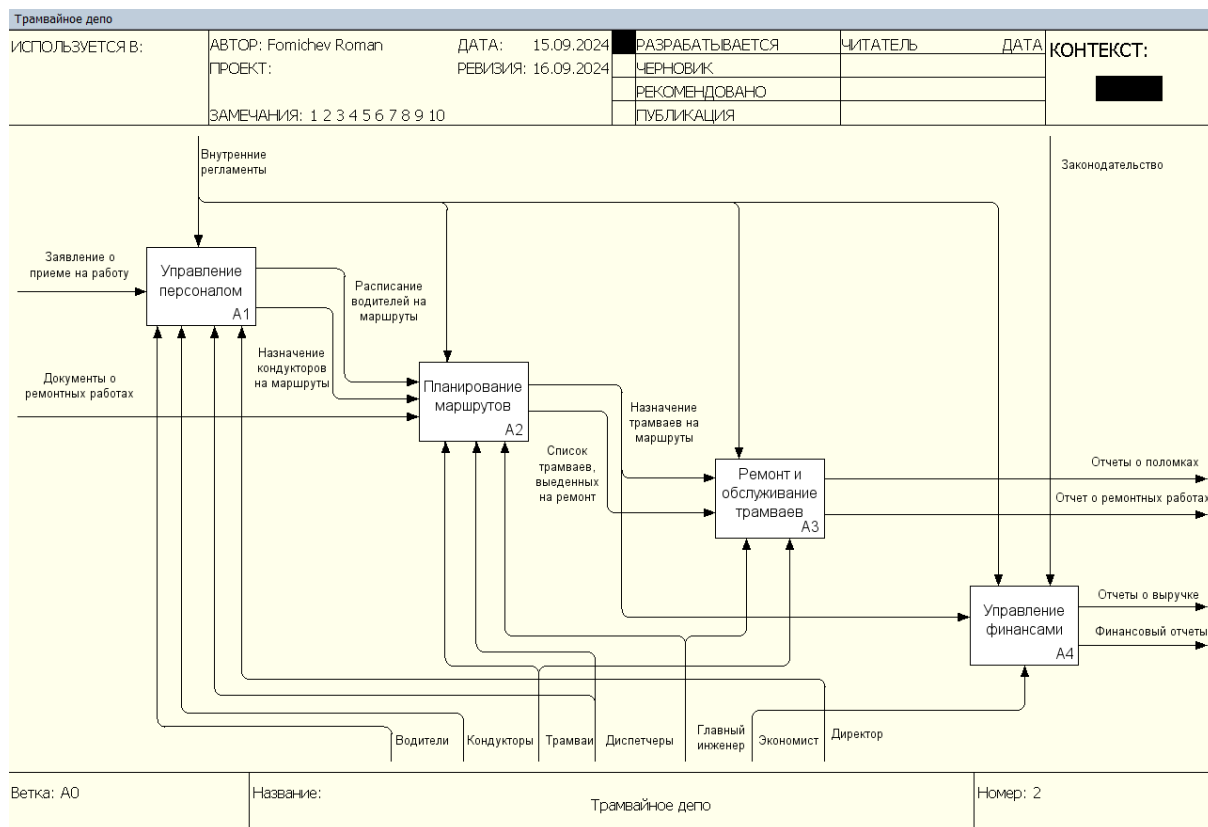


Рисунок 1.2 – Декомпозиция процесса «Трамвайное депо»

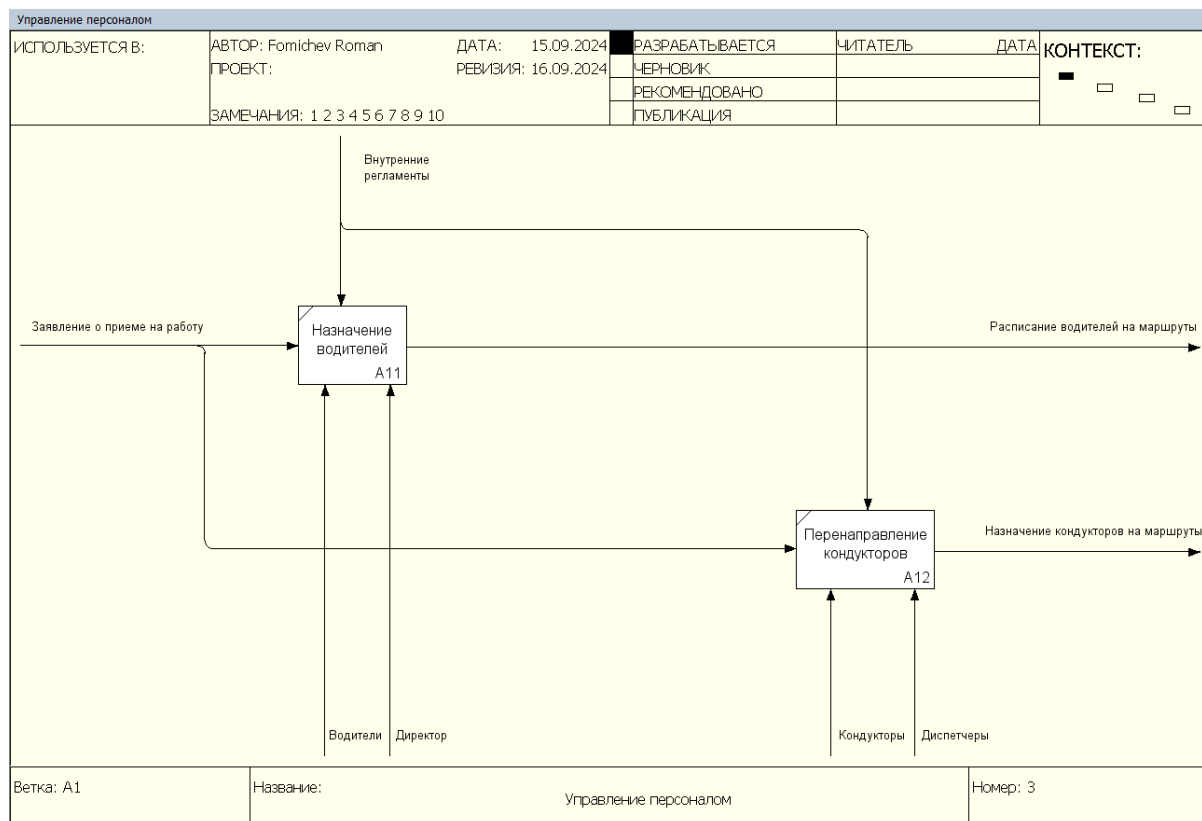


Рисунок 1.3 – Декомпозиция процесса «Управление персоналом»

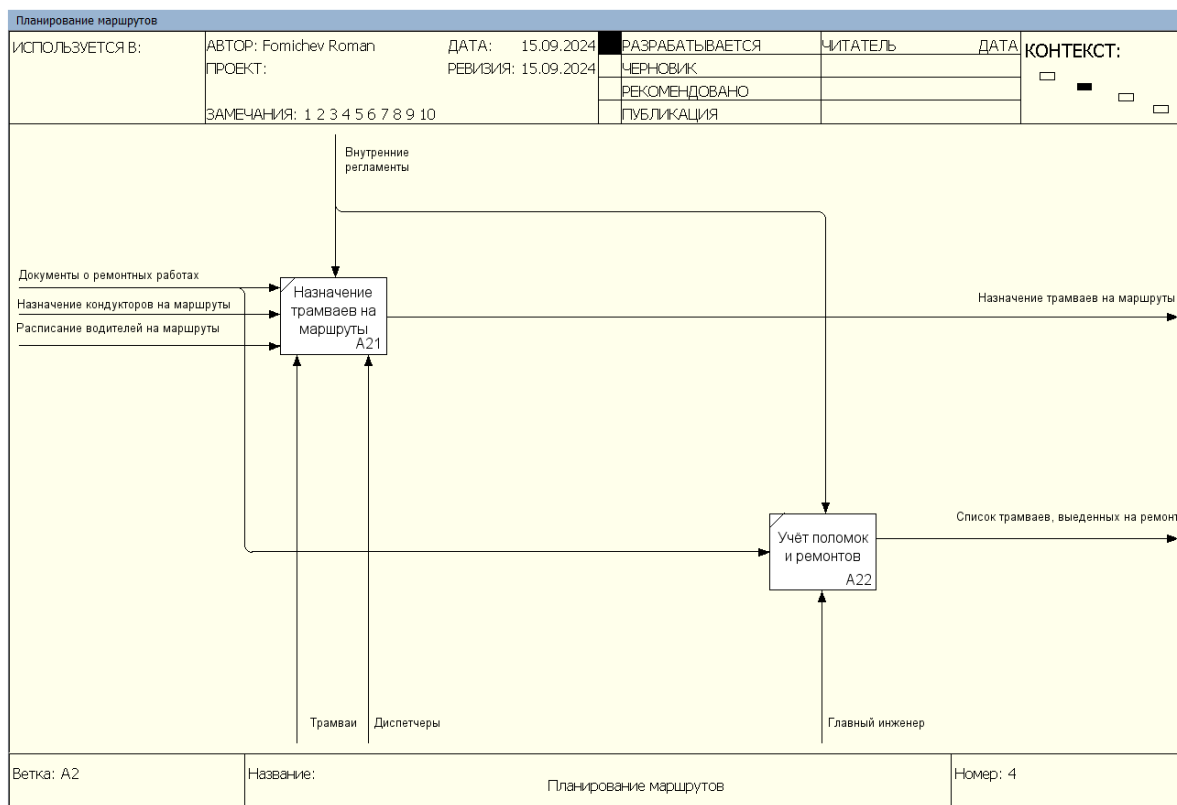


Рисунок 1.4 – Декомпозиция процесса «Планирование маршрутов»

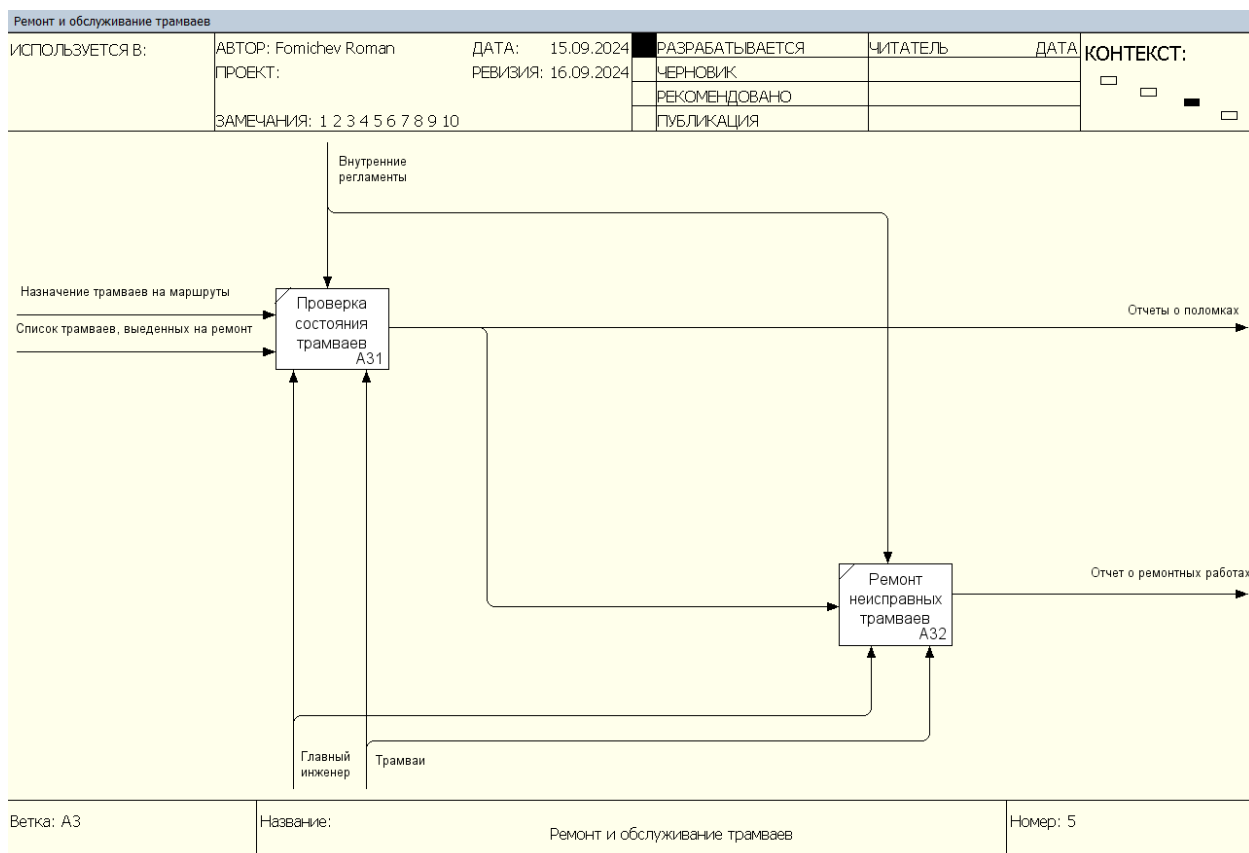


Рисунок 1.5 – Декомпозиция процесса «Ремонт и обслуживание трамваев»

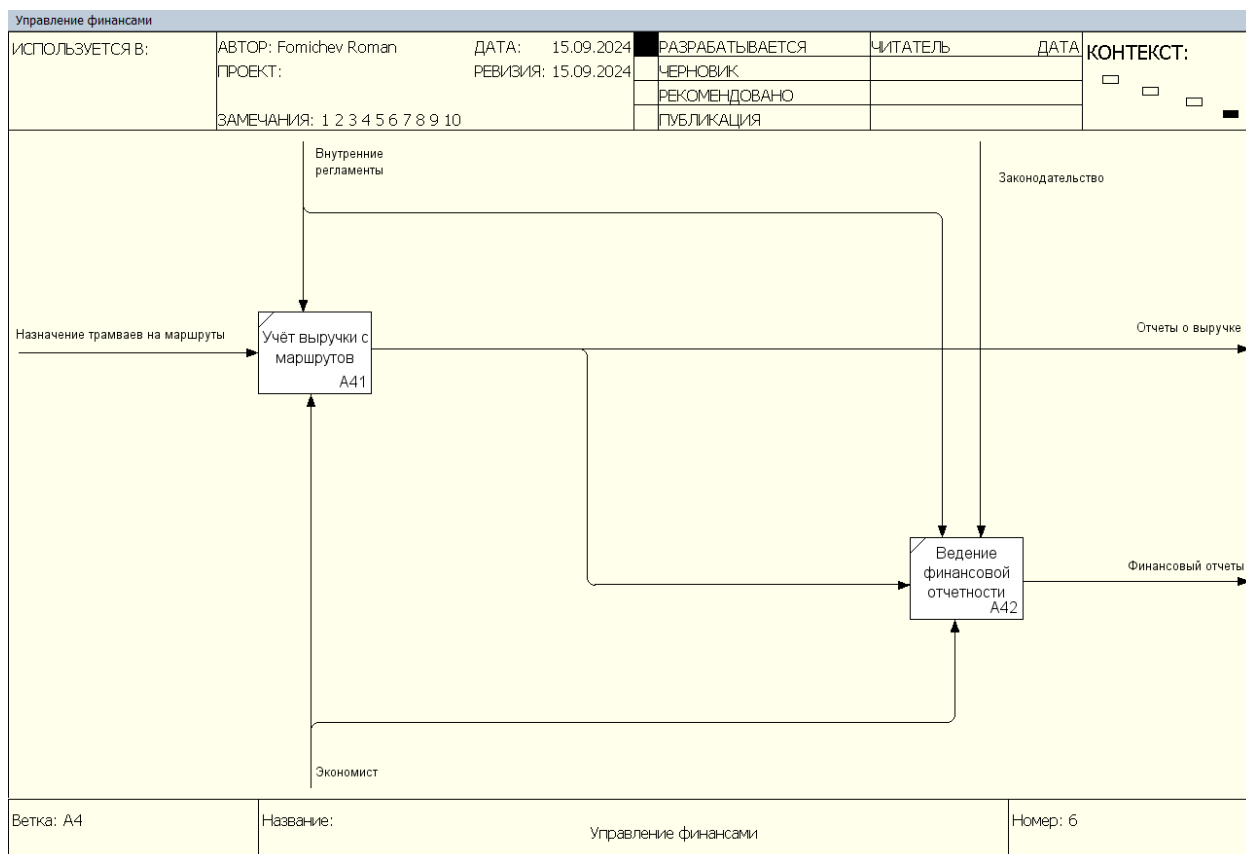


Рисунок 1.6 – Декомпозиция процесса «Управление финансами»

В результате рассмотрения настоящей работы система, а также анализа построенного процесса - «Трамвайное депо», было выявлено, что основные процессы в рамках, которых происходит замедление общей работы системы связаны с поиском неисправностей трамвая. Для оптимизации поиска неисправностей трамвая было выдвинуто предложение автоматизировать систему сбора информации о неисправностях, путем добавления датчиков, что позволит уменьшить время на поиск неисправностей.

1.4 Разработка модели бизнес-процесса «Как должно быть»

Для описания предложенной автоматизации была создана диаграмма по методологии IDEF0, в рамках которой было добавлено участие автоматизированной системы поиска неисправностей. Диаграмма представлена на рисунках 1.7 – 1.12.

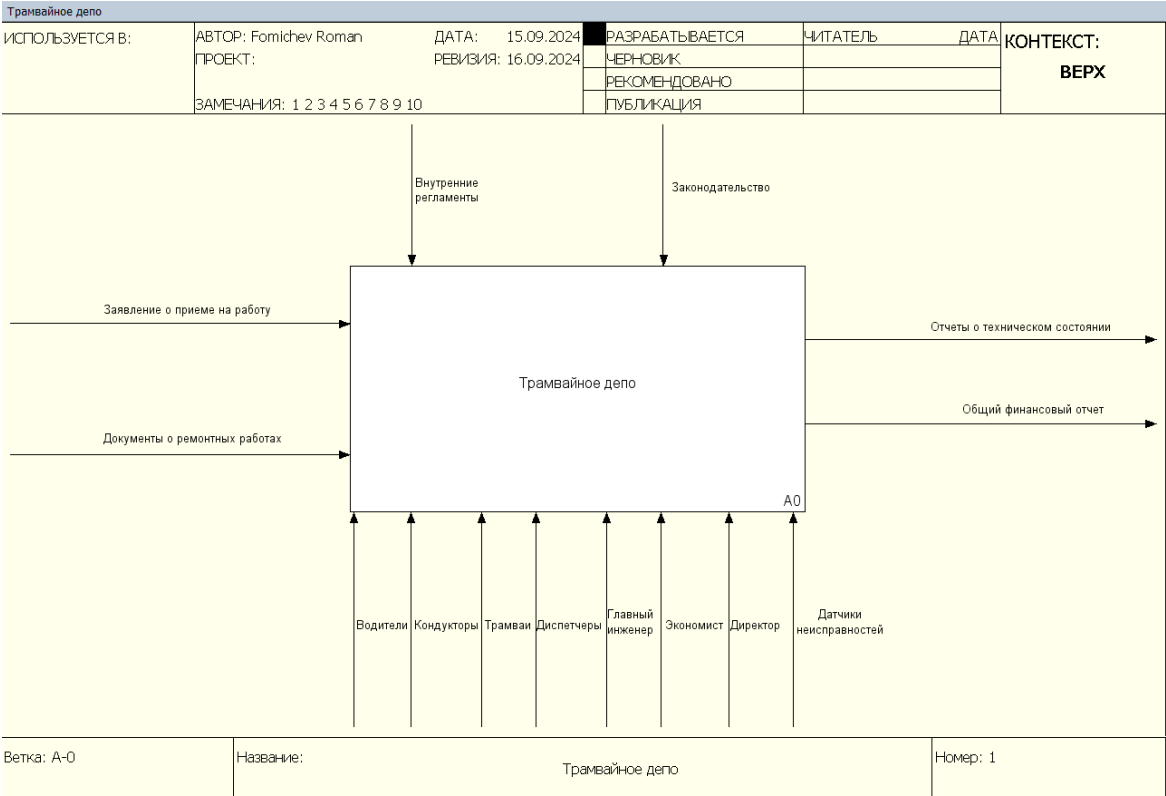


Рисунок 1.7 – Модифицированная диаграмма процесса «Трамвайное депо»

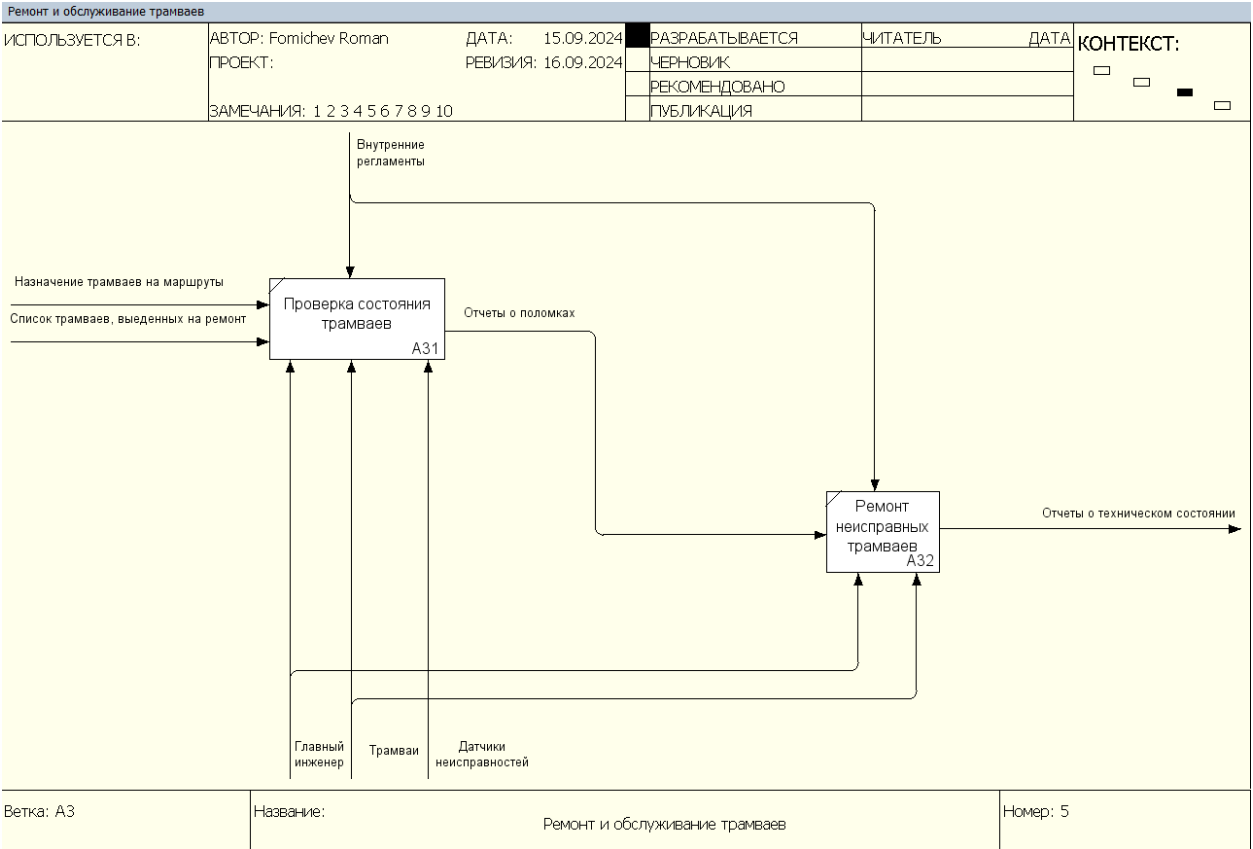


Рисунок 1.8 – Декомпозиция процесса «Ремонт и обслуживание трамваев»

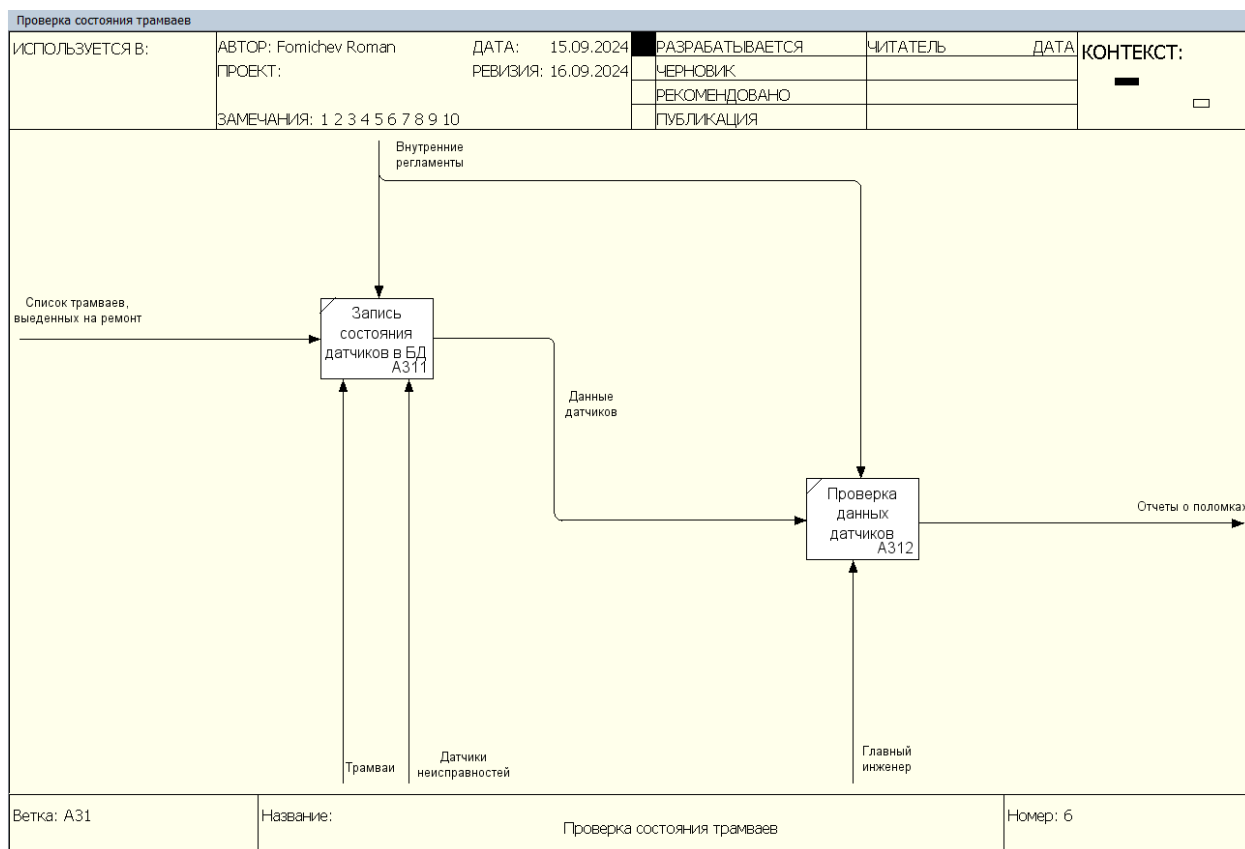


Рисунок 1.9 – Декомпозиция процесса «Проверка состояния трамваев»

Полученные диаграммы бизнес-процесса показывают те процессы, в рамках которых предполагается проводить автоматизацию с помощью использования автоматизированной системы поиска неисправностей для работы с неисправными трамваями.

1.5 Разработка требований к клиент-серверной системе

В результате анализа получившихся бизнес-процессов с участием системы автоматизации были сформулированы требования к информационной системе в соответствии с технологией FURPS+:

1) functionality, функциональность:

- регистрация пользователей;
- привязка аккаунта к уникальным идентификаторам организаций;
- хранение данных в распределенных базах данных;
- передача данных между пользователем и оператором;
- администрирование технического обслуживания;

2) usability, удобство пользования:

- наличие справочной информации пользователя;
- наличие руководства администратора системы;
- наличие инструкций для сотрудников;

3) reliability, надежность:

- обеспечение резервного копирования;
- использование нескольких экземпляров с балансировщиком

нагрузки;

4) performance, производительность:

- допустимое количество одновременно работающих пользователей до 40;
- суточная пропускная способность системы в рамках организации 20000 операций;

5) supportability, поддержка:

- инсталляция на территории организации;
- сбор данных со всех экземпляров;
- обеспечение автоматического обновления.

На основе описанных выше требований в дальнейшем будет сформировано техническое задание.

1.6 Обзор и анализ аналогичных клиент-серверных систем

На основе выдвинутых раньше требований были оценены несколько существующих систем[6]. Так как на данный момент системы по автоматизации трамвайных депо недостаточно распространены и не находятся в открытом доступе, были рассмотрены 3 системы:

- «1С: Управление автотранспортом» - приложение для управления ресурсами транспортных предприятий;
- «Infor EAM» - приложение для управления техническим обслуживанием;

– «IBM Maximo» - система для учета неисправностей и планировании технических работ.

Результаты представлены в таблице 2.

Таблица 2 – Обзор аналогичных клиент-серверных систем

Критерий	«1С: Управление автотранспортом»	«Infor EAM»	«IBM Maximo»
Регистрация пользователей	Присутствует	Присутствует	Отсутствует
Привязка аккаунта к уникальным идентификаторам организации	Отсутствует	Отсутствует	Отсутствует
Планирование и управление процессами	Присутствует	Отсутствует	Присутствует
Техническое обслуживание и ремонт	Отсутствует	Присутствует	Отсутствует
Наличие справочной информации	Присутствует	Присутствует	Присутствует
Пропускная способность	10000 операций в сутки	40000 операций в сутки	Неограниченная
Возможность установки на территории организации	Отсутствует	Отсутствует	Присутствует

В результате проведения оценки критериев для систем, являющихся лидерами в своём сегменте управления техническим обслуживанием, было получено подтверждение необходимости требований по функциональности и пользовательскому опыту на основе наличия их у популярных клиент-серверных систем.

1.7 Постановка задачи на разработку новой клиент-серверной системы

На основе описанных выше требований и проведенного анализа аналогичных клиент-серверных систем было составлено техническое задание [7], представленное далее.

1.8 Общие сведения

Наименование системы: Автоматизированная информационная система «Трамвайное депо».

Цель создания системы: Автоматизация процессов поиска неисправностей трамваев, уменьшение времени технического обслуживания трамваев.

Назначение разработки

Система предназначена для автоматизации технического обслуживания трамваев.

Требования к системе

Требования к системе включают в себя [8, 9, 10]:

1) функциональные требования:

- регистрация пользователей и учет их данных;
- привязка аккаунта к уникальным идентификаторам организаций;
- автоматическое снятие показаний датчиков;
- администрирование технического обслуживания;
- передача данных между пользователями и операторами;

2) требования к удобству пользования:

- наличие интуитивного пользовательского интерфейса;
- наличие справочной информации для пользователей;
- руководства по эксплуатации для администратора и сотрудников.

3) требования к надёжности:

- резервное копирование данных;
- использование нескольких экземпляров системы с балансировкой

нагрузки;

4) требования к производительности:

- одновременная работа до 40 пользователей в рамках организации;
- пропускная способность системы в рамках одной организации: 20

000 операций в сутки;

5) требования к поддержке:

- возможность локальной установки на территории организации;
- обеспечение автоматического обновления.

Состав и параметры системы

Система должна состоять из 4х обязательных компонентов [11]:

- клиентский модуль, для доступа пользователей к приложению;
- модуль для обработки запросов, данных, сообщений между пользователями;
- распределенная система хранения данных о пользователях и неисправностях;
- балансировщик нагрузки для распределения входящих запросов.

Требования к интерфейсу

Интерфейс должен быть совместим со всеми современными браузерами с версией, чья дата выхода не позднее 2014 года, также должен поддерживаться выбор языка и интеграция с системами идентификации пользователя.

1.9 Вывод

В рамках раздела была оценена область применения автоматизации трамвайного депо, были рассмотрены возможные системы аналоги, а также сформулированы требования, на основании которых будет производится дальнейшая разработка клиент-серверной системы.

2 РАЗРАБОТКА ЛОГИЧЕСКОЙ МОДЕЛИ КЛИЕНТ-СЕРВЕРНОЙ СИСТЕМЫ

На этапе, когда ясны требования информационной системе, необходимо разработать логическую модель клиент-серверной системы для дальнейшей реализации проекта [12].

2.1 Выбор методологии и технологии логического моделирования клиент-серверной системы

В современном мире существует множество методологий и технологий, применяемых для логического моделирования клиент-серверных систем. Основные из них включают структурный анализ и проектирование, объектно-ориентированный анализ и проектирование, а также использование языка унифицированного моделирования (UML).

Структурный анализ и проектирование (SADT, DFD) фокусируется на функциональной декомпозиции системы и потоках данных между процессами. Этот подход подходит для понимания функциональных требований, но может быть недостаточно эффективен при разработке сложных систем с высоким уровнем взаимодействия компонентов.

Объектно-ориентированный анализ и проектирование (ООАП) основывается на концепциях объектов, классов и их взаимодействий. Он способствует повышению повторного использования кода и улучшению модульности системы.

Унифицированный язык моделирования (UML) является стандартизированным языком визуального моделирования, поддерживающим множество диаграмм для представления различных аспектов системы. Он широко принят в индустрии и обеспечивает гибкость и стандартизацию в процессе разработки.

Для обоснования выбора наиболее подходящей методологии проведем сравнительный анализ указанных подходов, представленный в таблице 3.

Таблица 3 – Сравнительный анализ методологий логического моделирования

Критерий	Структурный анализ	ООАП	UML
Модульность	Низкая	Высокая	Высокая
Повторное использование	Ограниченное	Высокое	Высокое
Степень стандартизации	Низкая	Средняя	Высокая
Инструментальная поддержка	Ограниченная	Хорошая	Отличная
Простота освоения	Высокая	Средняя	Средняя
Поддержка сложных системных	Низкая	Высокая	Высокая

Исходя из проведенного анализа, объектно-ориентированный анализ и проектирование с использованием языка UML является наиболее подходящей методологией для разработки логической модели клиент-серверной системы. Это обусловлено тем, что UML является международным стандартом, что облегчает коммуникацию между разработчиками и другими заинтересованными сторонами. Кроме того, он обеспечивает гибкость и масштабируемость, поддерживая моделирование как небольших, так и крупных систем с высокой сложностью. Также следует отметить, что существует множество программных средств, поддерживающих UML, что ускоряет процесс разработки и снижает вероятность ошибок. Наконец, совместимость с объектно-ориентированными языками программирования облегчает переход от модели к реализации, что сокращает время разработки и повышает качество конечного продукта.

В заключение можно сказать, что выбор методологии объектно-ориентированного анализа и проектирования с использованием UML обоснован ее преимуществами в контексте разработки современных клиент-серверных систем. Это позволит создать гибкую, масштабируемую и понятную логическую модель, которая станет прочной основой для последующей реализации системы.

2.2 Разработка диаграмм логической модели клиент-серверной системы

Для лучшего понимания процессов необходимых для функционирования системы и её структуры были созданы несколько диаграмм.

Диаграмма классов, представленная на рисунке 2.1, отражает примерный состав классов, используемых для управления системой.

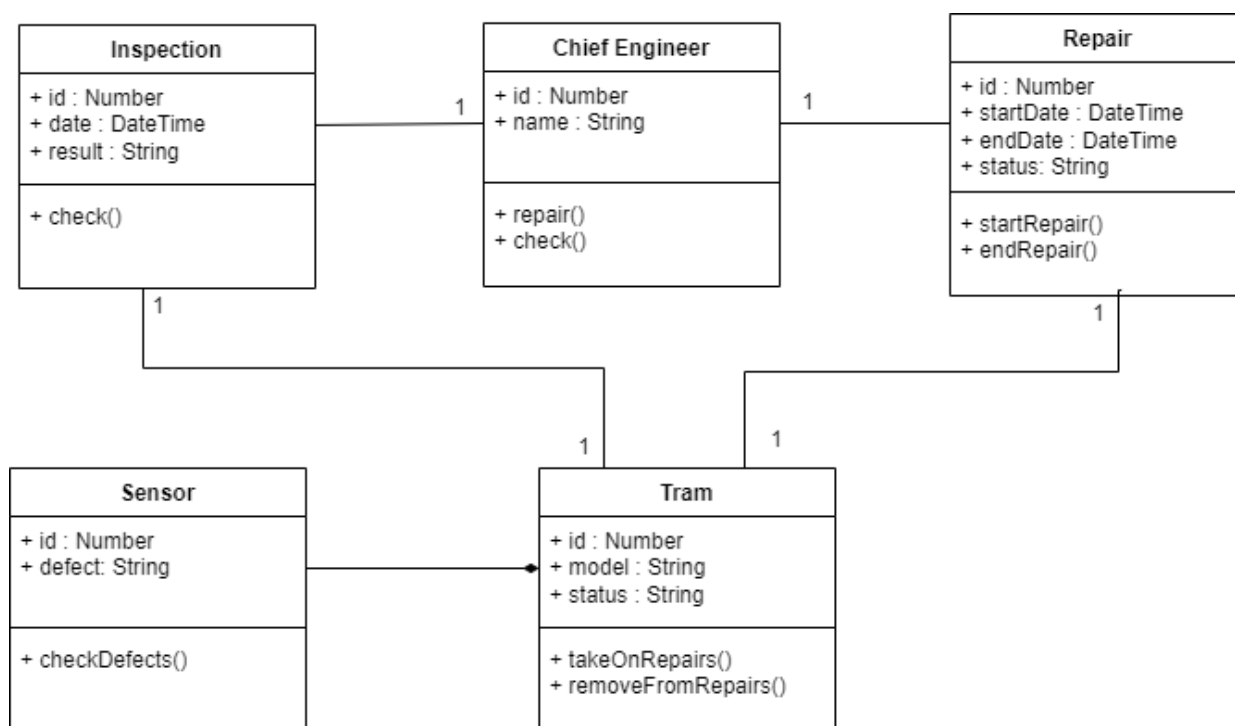


Рисунок 2.1 – Диаграмма классов

Для отражения основного процесса системы и взаимодействия между различными её элементами в рамках него, была построена диаграмма последовательности обработки запроса пользователя, которая представлена на рисунке 2.2.

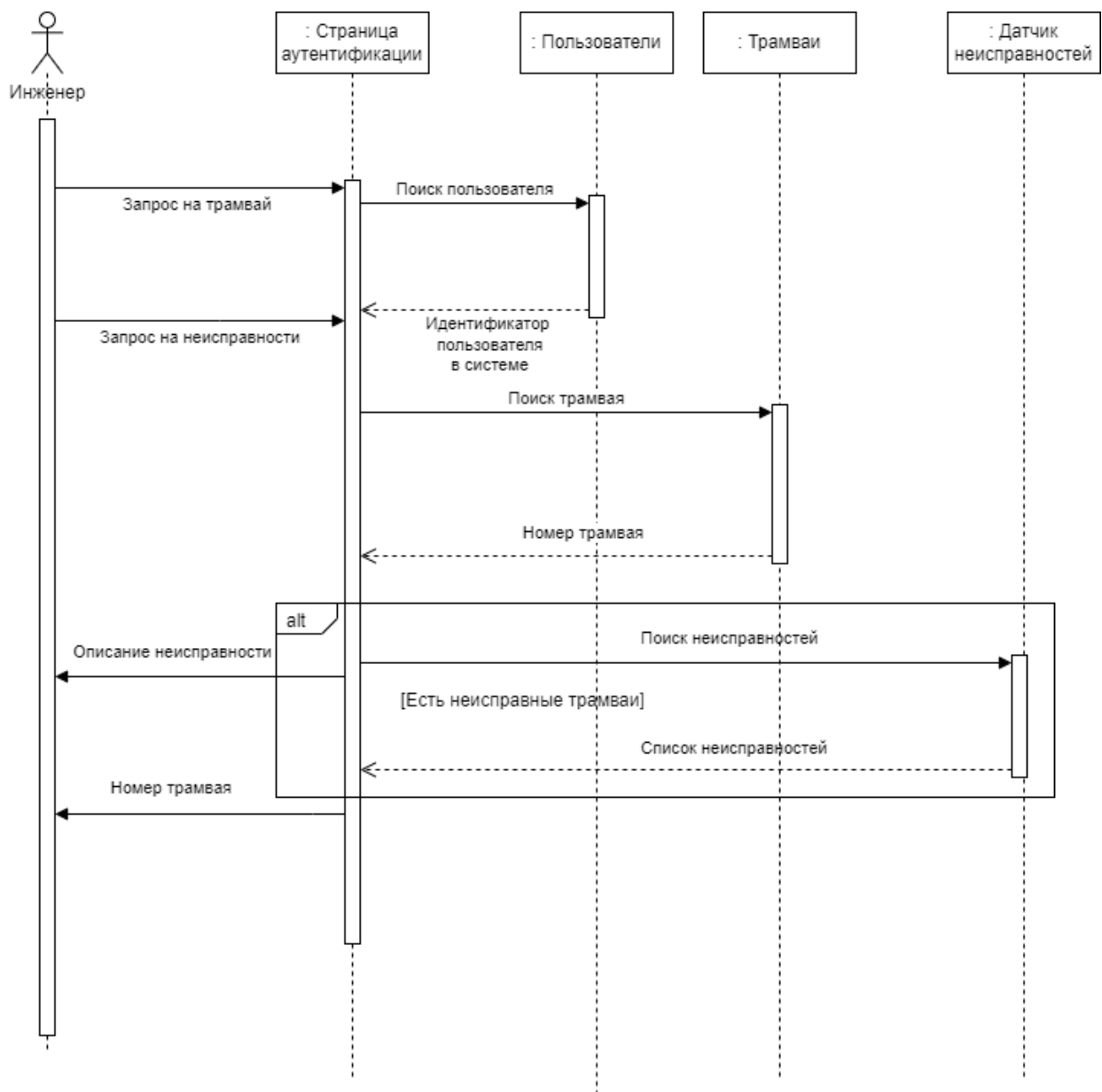


Рисунок 2.2 – Диаграмма последовательности

Также для данного процесса была сформирована диаграмма обзора взаимодействий, представленная на рисунке 2.3, которая позволяет лучше понять последовательность событий и различные варианты их ветвлений.

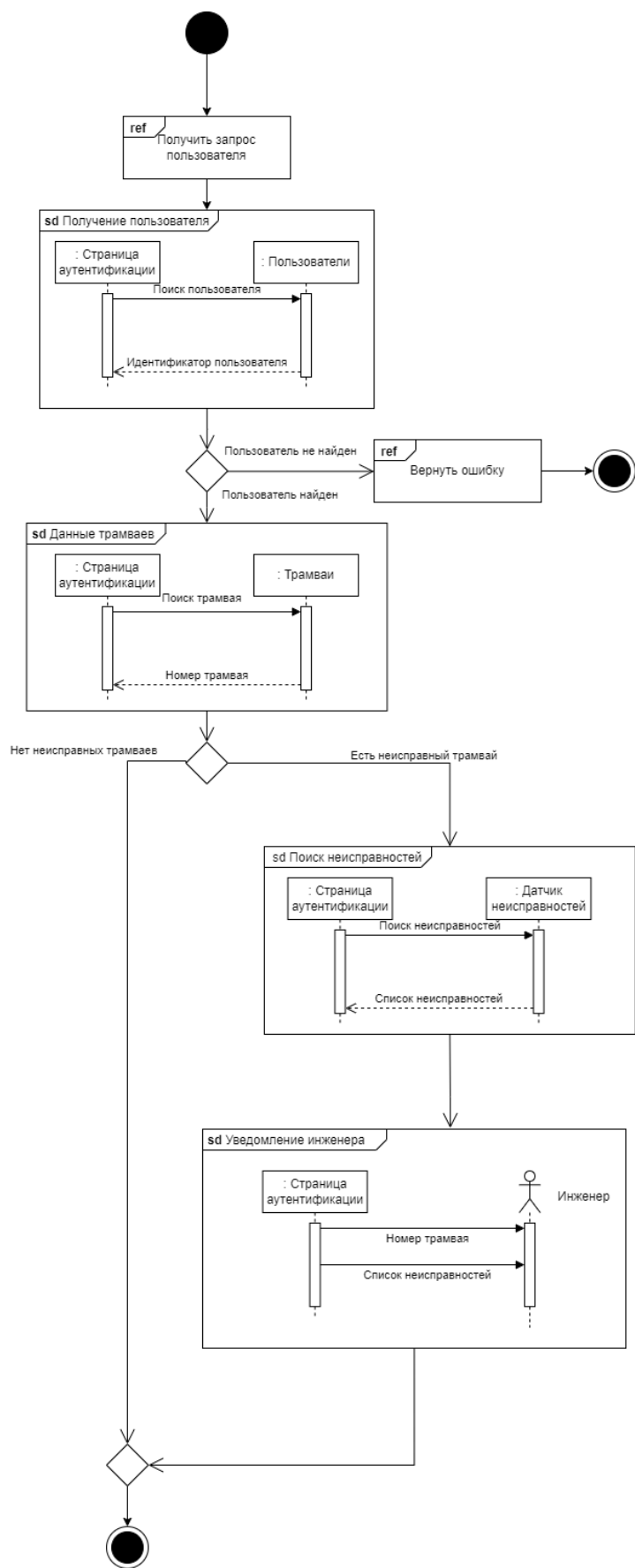


Рисунок 2.3 – Диаграмма обзора взаимодействий

В качестве демонстрации различных зависимостей между элементами системами [13] была создана диаграмма пакетов, представленная на рисунке 2.4.

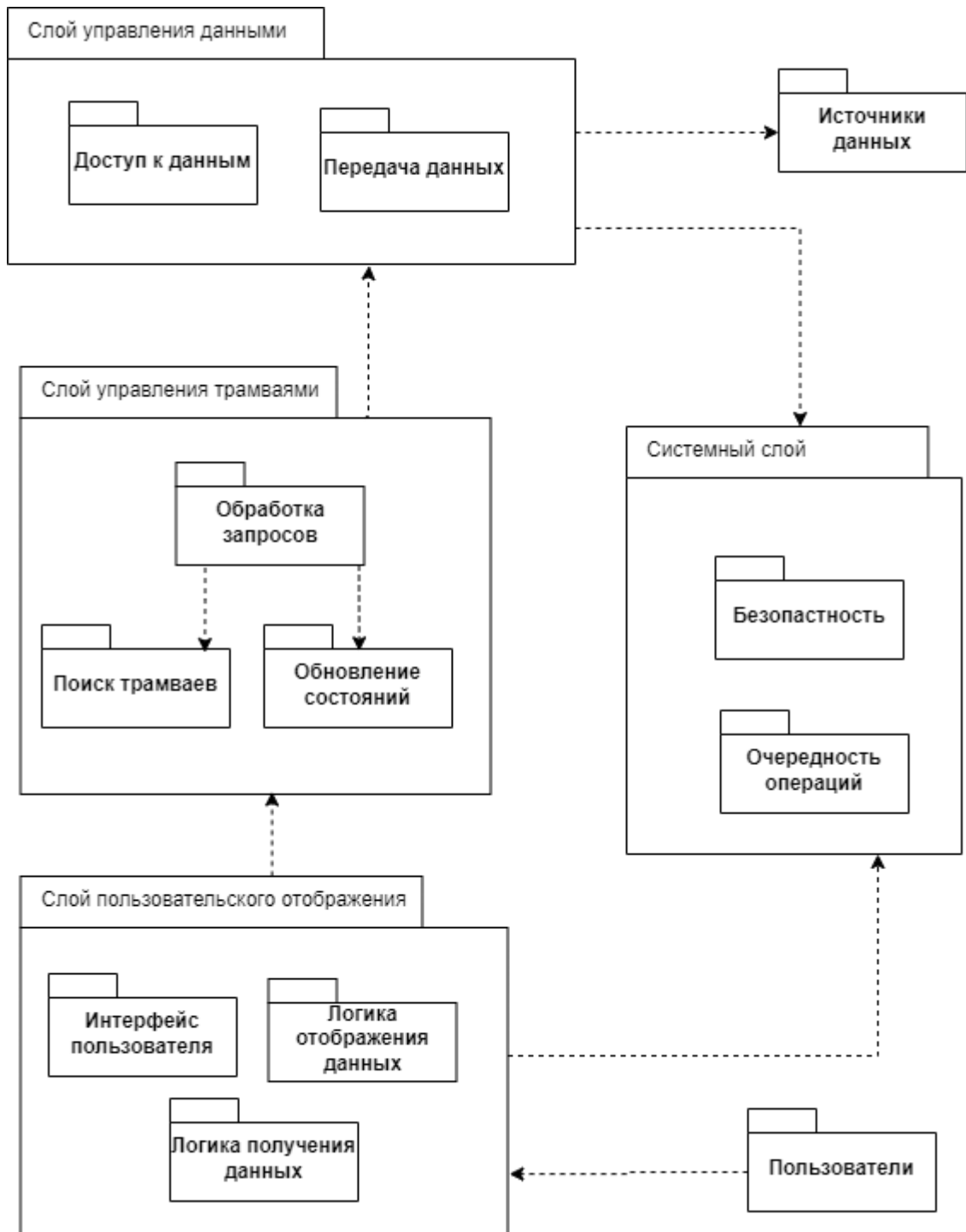


Рисунок 2.4 – Диаграмма пакетов

С помощью представленных выше диаграмм был описан основной процесс системы – обработка запроса пользователя, что позволяет лучше

понимать взаимодействие различных компонентов, а также последовательность и порядок событий, происходящих в системе в рамках этого процесса, что в дальнейшем будет использовано при разработке.

2.3 Разработка модели клиент-серверных потоков в системе

Для отображения потоков в системе была построена DFD модель процесса обработки запроса пользователя. Данная модель отражает события, происходящие в системе в рамках этого процесса, а также данные, циркулирующие между различными элементами системы. Результат построения представлен на рисунке 2.5.

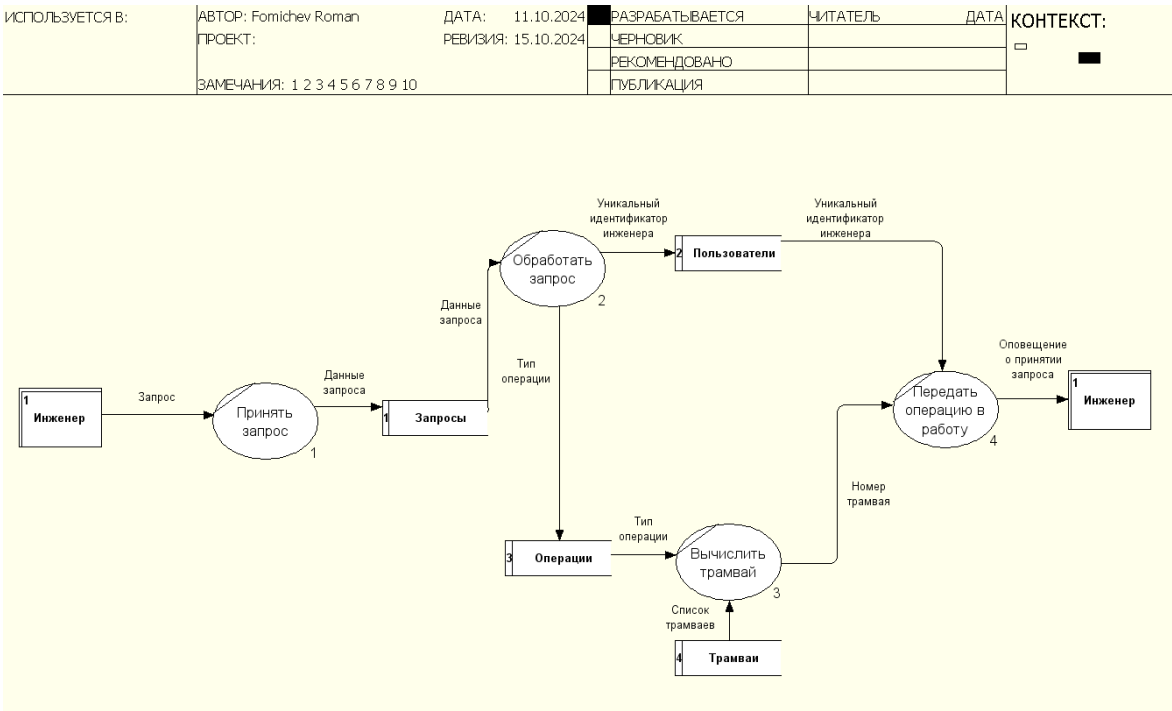


Рисунок 2.5 – DFD-модель обработки запроса

Данная модель будет использована в дальнейшем для построения логической модели данных ИС, для более детального отображения их связи.

2.4 Разработка логической модели

Основываясь на предыдущих диаграммах и моделях, была сформирована логическая модель данных информационной системы [14, 15],

представленная на рисунке 2.6, которая позволяет детально рассмотреть связи между различными сущностями системы.

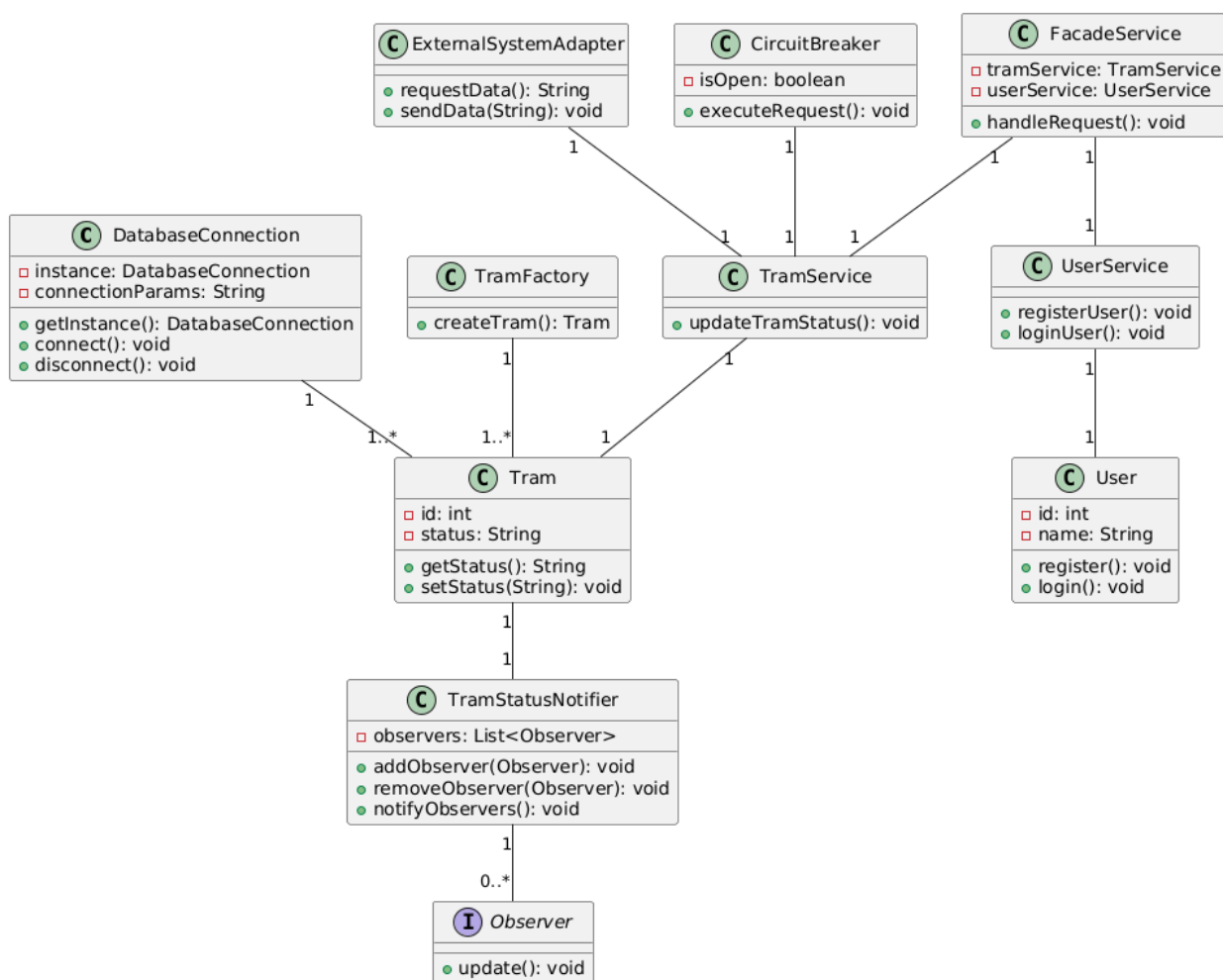


Рисунок 2.6 – Логическая модель данных ИС

2.5 Вывод

На основании требований, полученных в результате анализа в первом разделе, была проделана работа по анализу различных частей системы для формирования различных диаграмм, описывающих основные бизнес-процессы, происходящие в рамках автоматизации, взаимодействие различных элементов системы, а также логическую структуру сущностей.

ЗАКЛЮЧЕНИЕ

В результате выполнения данной курсовой работы была успешно достигнута цель - проектирование комплексной информационной системы, включающей в себя подсистемы для работы с пользователями и организацией технического обслуживания.

Для достижения данной цели был проведен анализ предметной области, описаны потоки данных в рамках системы, проанализированы существующие процессы, а также была создана функциональная модель будущей автоматизированной системы.

В ходе курсовой работы была разработана архитектура клиент-серверной системы, создана логическая модель данных информационной системы, а также были созданы различные диаграммы, описывающие предполагаемое течение процессов в автоматизированной системе, их взаимодействие с элементами системы и с окружающей инфраструктурой. Полученные навыки и знания позволили эффективно спроектировать и оформить комплексную информационную систему, включающую в себя подсистемы для работы с пользователями и организацией технического обслуживания.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Гвоздева Т. В., Баллод Б. А.. Проектирование информационных систем. Стандартизация [Электронный ресурс]:учебное пособие. - Санкт-Петербург: Лань, 2019. - 252 с. – Режим доступа: <https://e.lanbook.com/book/115515>.
2. Рочев К. В.. Информационные технологии. Анализ и проектирование информационных систем [Электронный ресурс]:учебное пособие. - Санкт-Петербург: Лань, 2019. - 128 с. – Режим доступа: <https://e.lanbook.com/book/122181>.
3. Вейцман В. М.. Проектирование информационных систем [Электронный ресурс]:учебное пособие. - Санкт-Петербург: Лань, 2019. - 316 с. – Режим доступа: <https://e.lanbook.com/book/122172>.
4. Остроух А. В., Суркова Н. Е.. Проектирование информационных систем [Электронный ресурс]:монография. - Санкт-Петербург: Лань, 2019. - 164 с. – Режим доступа: <https://e.lanbook.com/book/118650>.
5. Современные CASE средства проектирования систем: сайт. – URL: http://window.edu.ru/resource/616/73616/files/kulyabov-korolkova_formal-methods.pdf (дата обращения 11.10.2024). – Текст: электронный.
6. Типовые модели систем: сайт. – URL: <http://kgau.ru/istiki/umk/pis/l37.htm> (дата обращения 15.11.2024). – Текст: электронный.
7. Стандарты разработки: сайт. – URL: <http://venec.ulstu.ru/lib/disk/2017/460.pdf> (дата обращения 13.10.2024). – Текст: электронный.
8. Эксплуатация и сопровождение проекта: сайт. – URL: <http://kgau.ru/istiki/umk/pis/l7.htm> (дата обращения 09.11.2024). – Текст: электронный.
9. Тестирование и контроль программных систем: сайт. – URL: <https://xreferat.com/33/2759-1-sushnost-i-osobennosti-ispol-zovaniya->

instrumental-nogo-programmnogo-obespecheniya.html (дата обращения 24.11.2024). – Текст: электронный.

10. Контроль и корректировка кода: сайт. – URL: <https://studfile.net/preview/2790134> (дата обращения 16.10.2024). – Текст: электронный.

11. Проектирование и разработка Java приложений и систем: сайт. – URL: <https://dic.academic.ru/dic.nsf/ruwiki/608820> (дата обращения 13.10.2024). – Текст: электронный.

12. Object Constraints Language: сайт. – URL: <https://ami.nstu.ru/~vms/lecture/lecture12/lecture12.htm> (дата обращения 10.11.2024). – Текст: электронный.

13. Архитектура ЭВМ и систем: сайт. – URL: <https://arxiv.org/ftp/arxiv/papers/1802/1802.06769.pdf> (дата обращения 18.10.2024). – Текст: электронный.

14. Проектирование ER-диаграмм: сайт. – URL: http://citforum.ru/cfin/prcorpsys/infosistpr_09.shtml (дата обращения 12.11.2024). – Текст: электронный.

15. Реляционные СУБД: сайт. – URL: <https://compress.ru/article.aspx?id=10082> (дата обращения 11.10.2024). – Текст: электронный.