

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

Институт математики, физики и информационных технологий  
(наименование института полностью)

---

Кафедра «Прикладная математика и информатика»  
(наименование)

09.03.03 Прикладная информатика

(код и наименование направления подготовки, специальности)

---

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

---

## **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)**

на тему «Автоматизация процессов создания и передачи товаросопроводительной документации на железнодорожные грузоперевозки».

Обучающийся \_\_\_\_\_ А.М. Курлянов \_\_\_\_\_  
(И.О. Фамилия) (личная подпись)

Руководитель \_\_\_\_\_ Н.Н. Казаченок \_\_\_\_\_  
(ученая степень, звание, И.О. Фамилия)

Тольятти 2023

## Аннотация

Тема выпускной квалификационной работы - автоматизация процессов создания и передачи товаросопроводительной документации на железнодорожные грузоперевозки.

Цель работы – разработать приложение, автоматизирующее процесс создания и передачи сопроводительной документации на железнодорожные грузоперевозки.

Задачи работы:

- изучить деятельность компании ООО «Информсистемы»;
- провести проектирование и реализовать приложение, которое обеспечит автоматизированное создание сопроводительной документации на железнодорожные грузоперевозки.

В первой главе работы было проведено исследование деятельности компании по грузовым перевозкам. В ходе исследования также были изучены бизнес-процессы компании и документы, с которыми работают на этапе составления документации на железнодорожные перевозки. На основе полученной информации были смоделированы бизнес-процессы в нотации IDEF0.

Вторая глава описывает структуру приложения, которая автоматизирует процесс создания и передачи документации на грузоперевозки. Также спроектированы все необходимые диаграммы с помощью нотации UML. На основе полученных знаний были выделены объекты приложения и построена структура базы данных.

В третьей главе описывается физическое проектирование приложения, что включает выбор архитектуры приложения, разработку физической модели данных и описание модулей приложения.

Бакалаврская работа состоит из 62 страниц текста, 24 рисунка, 4 таблиц и 23 источников.

## Оглавление

Введение.....	5
Глава 1 Исследование деятельности компании ООО «Информсистемы».....	7
1.1 Анализ деятельности компании ООО «Информсистемы».....	7
1.2 Концептуальное моделирование процессов компании ООО «Информсистема».....	9
1.2.1 Выбор технологии концептуального моделирования.....	11
процессов компании .....	11
1.2.2 Разработка и анализ модели бизнес-процесса «как есть» .....	12
1.2.3 Обоснование необходимости автоматизированного варианта .....	14
решения .....	14
1.3 Анализ существующих разработок на предмет соответствия сформулированным требованиям .....	16
1.3.1 Сравнительная характеристика существующих разработок .....	17
1.4 Разработка модели бизнес-процесса «как должно быть».....	20
1.5 Постановка задачи на разработку проекта создания АИС .....	22
Глава 2 Логическое проектирование приложения.....	27
2.1 Выбор технологии логического моделирования .....	27
2.2 Логическая модель приложения.....	28
2.3 Информационное обеспечение приложения .....	31
2.3.1 Характеристика нормативно-справочной, входной и оперативной информации .....	31
2.3.2 Характеристика выходной информации .....	32
2.4 Проектирование базы данных .....	33
2.4.1 Выбор технологии проектирования базы данных .....	33
2.4.2 Разработка концептуальной модели данных .....	34
2.4.3 Обоснование вида логической модели .....	37
2.4.4 Разработка логической модели данных .....	38

2.5 Требования к аппаратно-программному обеспечению.....	39
Глава 3 Физическое проектирование приложения .....	41
3.1 Выбор архитектуры приложения .....	41
3.2 Выбор технологии разработки программного обеспечения .....	42
3.3 Выбор системы управления базой данных приложения.....	43
3.4 Разработка физической модели данных .....	46
3.5 Разработка приложения.....	48
3.5.1 Описание модулей приложения .....	48
3.6 Описание функциональности приложения .....	49
3.7 Обоснование экономической эффективности разработки .....	52
приложения.....	52
Заключение .....	59
Список используемой литературы и используемых источников.....	60

## Введение

Работа посвящена разработке приложения, которое автоматизирует процесс создания и передачи документации на железнодорожные грузоперевозки.

Грузовые перевозки занимают важную и значительную роль в современном мире, остаются важными аспектами для мировой экономики.

Мировая экономика стремительно развивается, и грузовые перевозки остаются одним из главных способов перемещения товаров по всему миру.

Ключевой составляющей успешного бизнеса стала эффективная логистика. Грузовые перевозки позволяют доставлять товары в различные точки мира, минимизируя затраты и сокращая время доставки.

Развитие интернет-торговли повысило потребность в грузоперевозках. Онлайн-магазины и маркетплейсы регулярно отправляют товары по всему миру, в частности, и по железной дороге, что стимулирует развитие грузовых перевозок.

Мировые компании стремятся создать мировые цепи поставок для снижения затрат и повышения эффективности производства.

В мире грузовые перевозки остаются движущей силой мировой экономики и торговли. Их актуальность будет продолжать расти, особенно с учетом постоянных изменений в технологиях и требованиях к устойчивости и эффективности в перевозках.

Внедрение современных информационных технологий и автоматизированных систем в данной отрасли приносит значительные преимущества и оптимизирует весь цикл работы компании [1].

Цель работы - разработать приложение, автоматизирующее процесс создания и передачи сопроводительной документации на железнодорожные грузоперевозки.

Задачи работы:

- изучить деятельность компании ООО «Информсистемы»;

- провести проектирование приложения, которое обеспечит автоматизированное создание сопроводительной документации на железнодорожные грузоперевозки;
- реализовать приложение для грузовой компании ООО «Информсистемы».

Объектом исследования являются процессы создания и передачи товаросопроводительной документации в грузовой компании ООО «Информсистемы».

Предметом исследования является автоматизация процессов создания и передачи товаросопроводительной документации на железнодорожные перевозки.

Выпускная квалификационная работа состоит из введения, трех разделов, заключения, списка используемой литературы и источников.

В первой главе работы было проведено исследование деятельности компании по грузовым перевозкам. В ходе исследования также были изучены бизнес-процессы компании и документы, с которыми работают на этапе составления документации на железнодорожные перевозки. На основе полученной информации были смоделированы бизнес-процессы в нотации IDEF0 [2].

Вторая глава описывает структуру приложения, которая автоматизирует процесс создания и передачи документации на грузоперевозки. Также спроектированы все необходимые диаграммы с помощью нотации UML. На основе полученных знаний были выделены объекты приложения и построена структура базы данных.

В третьей главе описывается физическое проектирование приложения, что включает выбор архитектуры приложения, разработку физической модели данных и описание модулей приложения.

# **Глава 1 Исследование деятельности компании ООО «Информсистемы»**

## **1.1 Анализ деятельности компании ООО «Информсистемы»**

Компания «Информсистема» основана компанией предпринимателей, которые видели потенциал в сфере грузоперевозок. Компания создавалась в период растущего спроса на услуги перевозок товаров [3].

В начале своей деятельности компания «Информсистема» специализировалась в конкретных видах грузоперевозок, таких как автомобильные перевозки, железнодорожные перевозки и воздушные грузоперевозки. Важно было построить репутацию надежного перевозчика.

Со временем компания расширила свой бизнес, добавляя новые виды грузоперевозок и услуги, чтобы удовлетворить разнообразные потребности клиентов. Спектр услуг добавил в себя международные перевозки, логистические услуги, складское хранение и другие.

Для повышения эффективности компания старается следовать развитию новых технологий и внедрять в бизнес-процессы, такие как системы отслеживания грузов, автоматизированные системы управления логистикой и другие.

В компании есть несколько отделов, которые в совокупности обеспечивают процесс перевозок товаров.

Ниже описаны основные отделы и их функции, связанные с приемкой и отгрузкой грузов.

Генеральный директор - высший руководитель компании, отвечающий за стратегическое управление и принятие ключевых бизнес-решений.

Директор по логистике и транспорту - отвечает за общее управление логистическими операциями и контроль над транспортными ресурсами.

Отдел продаж и маркетинга.

Директор по продажам - отвечает за привлечение клиентов, заключение договоров и управление отношениями с клиентами.

Маркетинг-менеджер - разрабатывает маркетинговые стратегии и рекламные кампании.

Отдел логистики.

Логистический менеджер - осуществляет планирование и координацию логистических операций, включая выбор маршрутов и способов доставки.

Специалист по складской логистике - отвечает за управление складами, хранение и инвентаризацию грузов.

Диспетчеры - отслеживают и управляют передвижением грузовых транспортных средств в режиме реального времени.

Отдел транспорта.

Директор по транспорту - отвечает за управление транспортными ресурсами, включая автомобили, поезда, суда или самолеты, в зависимости от специфики компании.

Менеджер по транспортным ресурсам - занимается распределением и обслуживанием транспортных средств.

Отдел качества и безопасности.

Менеджер по качеству и безопасности - отвечает за соблюдение стандартов качества и безопасности во всех логистических операциях.

Отдел информационных технологий.

ИТ-директор - занимается разработкой и поддержкой информационных систем, используемых для управления процессами и отслеживания грузов.

Бухгалтерия и финансы.

Финансовый директор - отвечает за управление финансами компании, включая бюджетирование, учет и финансовую отчетность.

Отдел кадров.

Директор по персоналу - занимается наймом, обучением и управлением персоналом компании.

Служба клиентской поддержки.



Менеджеры по обслуживанию клиентов - работают с клиентами, принимая заказы, предоставляя информацию и решая возникающие вопросы и проблемы.

Структура компании ООО «Информсистема» представлена на рисунке 1.

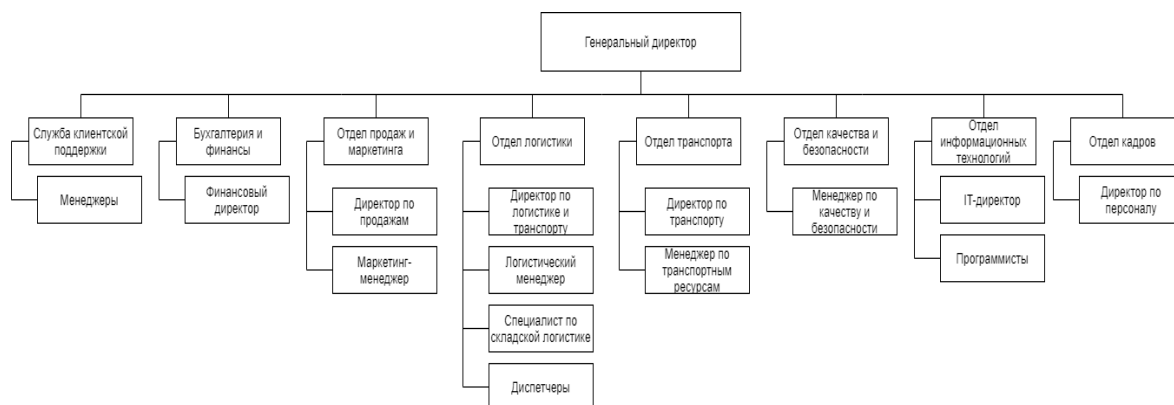


Рисунок 1 – Организационная структура компании ООО «Информсистема»

Все отделы компании и сотрудники выполняют ключевые функции, направленные на эффективное управление и выполнение логистических задач, включая приемку и отгрузку грузов, чтобы обеспечить клиентов качественными и своевременными услугами.

## 1.2 Концептуальное моделирование процессов компании ООО «Информсистема»

Основные бизнес-процессы компании, занимающейся грузовыми железнодорожными перевозками, включают в себя целый ряд операций и этапов, направленных на эффективное и безопасное перемещение грузов по железной дороге.

Опишем основные бизнес-процессы.

Клиент связывается с компанией для запроса грузоперевозки. Специалист по приему заказов регистрирует запрос, включая информацию о типе груза, его объеме, маршруте и сроках доставки. Проверяются наличие и актуальность всех необходимых документов и разрешений для груза. Производится расчет стоимости перевозки на основе параметров груза и выбранного маршрута. Заключается договор между клиентом и компанией, устанавливающий условия и сроки перевозки.

Планирование и логистика происходит с помощью логистического менеджера, который анализирует заказы и графики перевозок. Выбирается оптимальный маршрут с учетом расстояния, времени и особенностей груза. Устанавливается график движения поезда, включая дату и время отправки и прибытия. Оформляются необходимые разрешения и документы для движения по маршруту.

Подготовка и погрузка грузов проходит согласно всем стандартам. Осуществляется весовой контроль для точного определения веса груза. Груз погружается на железнодорожные вагоны с соблюдением всех стандартов и правил безопасности. Груз закрепляется и зафиксирован на вагонах, чтобы предотвратить его перемещение во время транспортировки.

Отслеживание грузов осуществляется по установленному маршруту и графику. Специалисты по отслеживанию следят за передвижением поезда в режиме реального времени с помощью систем GPS и телеметрии. Постоянно обновляется информация о местоположении и статусе груза. В случае задержек или нестандартных ситуаций, специалисты принимают меры для их устранения и информируют клиентов.

Прием и разгрузка грузов осуществляется по прибытии поезда на конечный пункт назначения, начинается процесс разгрузки. Грузы выгружаются с вагонов и проходят процедуры проверки на целостность и отсутствие повреждений. Оформляются все необходимые документы о приеме груза, их подписывают и утверждают. Грузы передаются клиенту или далее направляются по маршруту в случае необходимости.

### **1.2.1 Выбор технологии концептуального моделирования процессов компании**

Для моделирования процессов компании, включая бизнес-процессы, существует множество инструментов и методик. Рассмотрим несколько распространенных инструментов для моделирования бизнес-процессов.

BPMN (Business Process Model and Notation) представляет собой стандартный язык для моделирования бизнес-процессов. BPMN позволяет создавать графические диаграммы, отображающие шаги процесса, потоки данных, роли и связи между ними. Популярными инструментами для BPMN моделирования являются Bizagi, Lucidchart, и Signavio.

ARIS (Architecture of Integrated Information Systems). Это инструмент для управления бизнес-процессами и моделирования организационной структуры. ARIS предоставляет мощные средства для анализа и оптимизации процессов.

Microsoft Visio предоставляет инструменты для создания разнообразных диаграмм, включая бизнес-процессы, организационные структуры и диаграммы потоков данных. Это широко используемое приложение для моделирования.

IBM Blueworks Live - платформа для моделирования и автоматизации бизнес-процессов. Blueworks Live предоставляет инструменты для создания, анализа и оптимизации процессов.

ARIS Cloud облачная версия инструмента ARIS, которая позволяет командам совместно работать над моделированием и анализом бизнес-процессов.

Camunda — это open-source платформа для автоматизации бизнес-процессов, но также предоставляет инструменты для моделирования процессов. Особенно полезен, если вам нужна автоматизация бизнес-логики.

Для более сложных процессов может потребоваться использование программного обеспечения для симуляции, такого как AnyLogic, Simul8 или

Arena. Эти инструменты позволяют создавать модели и проводить численные эксперименты для оптимизации процессов.

Для небольших компаний или начальных этапов моделирования процессов можно использовать таблицы Excel или Google Sheets для создания простых диаграмм и процессов.

DEF (Integrated DEFinition) - это набор методологий и инструментов для моделирования бизнес-процессов и систем, который широко используется в области инженерии и управления. Он был разработан как стандарт США и имеет несколько версий, таких как IDEF0, IDEF1X, IDEF3 и другие, каждая из которых ориентирована на определенные аспекты моделирования и анализа.

### **1.2.2 Разработка и анализ модели бизнес-процесса «как есть»**

Существует несколько причин, почему IDEF хороший выбор для моделирования бизнес-процессов:

- стандартизация;
- контекстуальное моделирование;
- анализ и оптимизация;
- интеграция с другими методологиями;
- инструментальная поддержка;
- универсальность.

На рисунке 2 показаны процессы «как есть», компании ООО «Информсистема» в нотации IDEF0. Моделирование осуществлялось с помощью программного продукта VpWin 7.0, который позволяет легко моделировать диаграммы, проводить декомпозиции на разные уровни. Также программный продукт позволяет осуществлять моделирование в нескольких разных нотациях, таких как IDEF0, DFD и IDEF3.

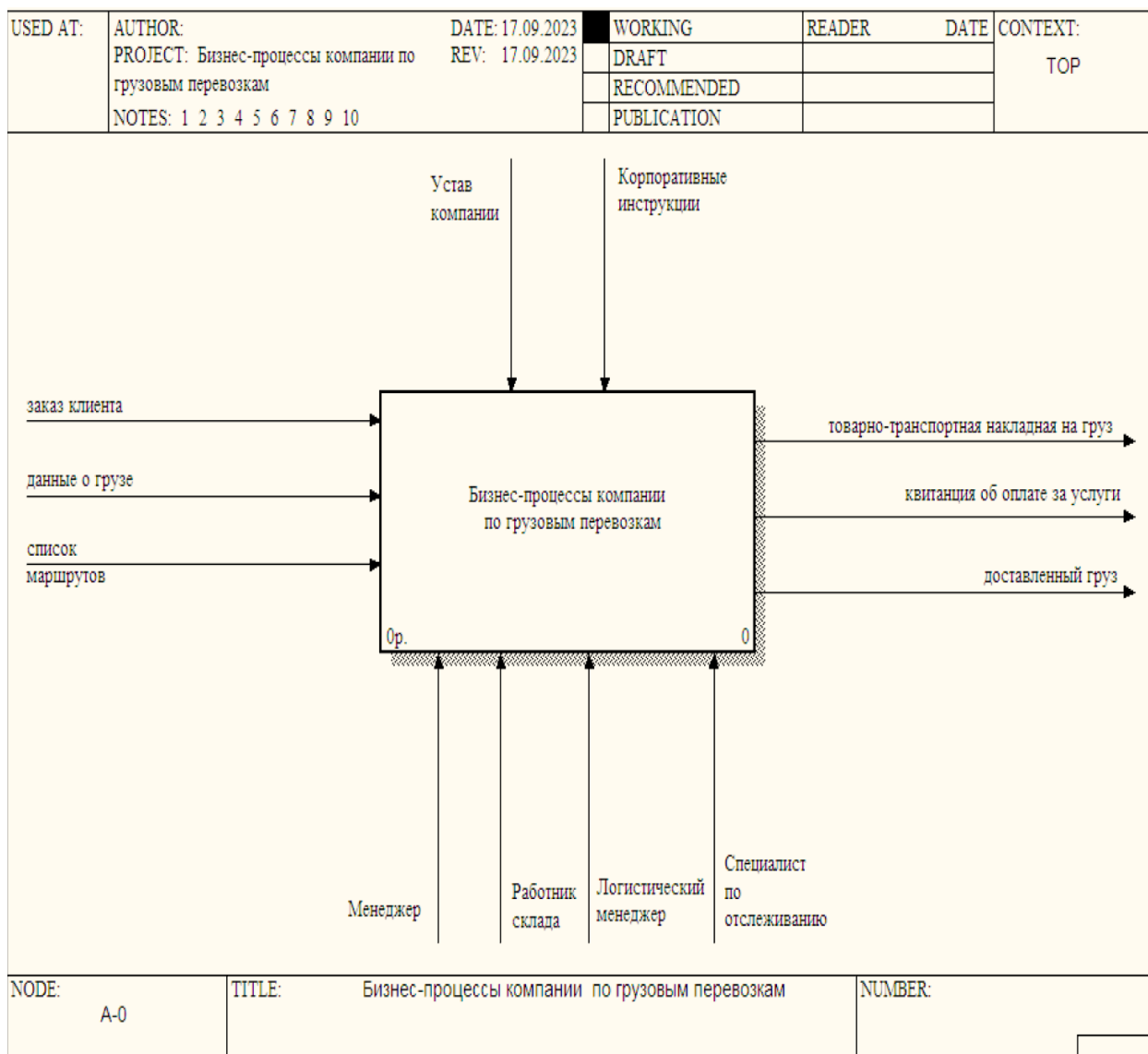


Рисунок 2 – Контекстная диаграмма процессов компании, модель «как есть»

Бизнес-процессы начинают исполняться после получения заказа клиента, после чего идет регистрация заказа и происходит регистрация груза, в которую входит замеры и взвешивание груза, для оценки стоимости.

Далее происходит выбор оптимального маршрута доставки груза и подготовка груза к отправке. После отправки груза специалист отслеживает доставку до места назначения.

Подробный процесс приема и отгрузки груза показан на рисунке 3.

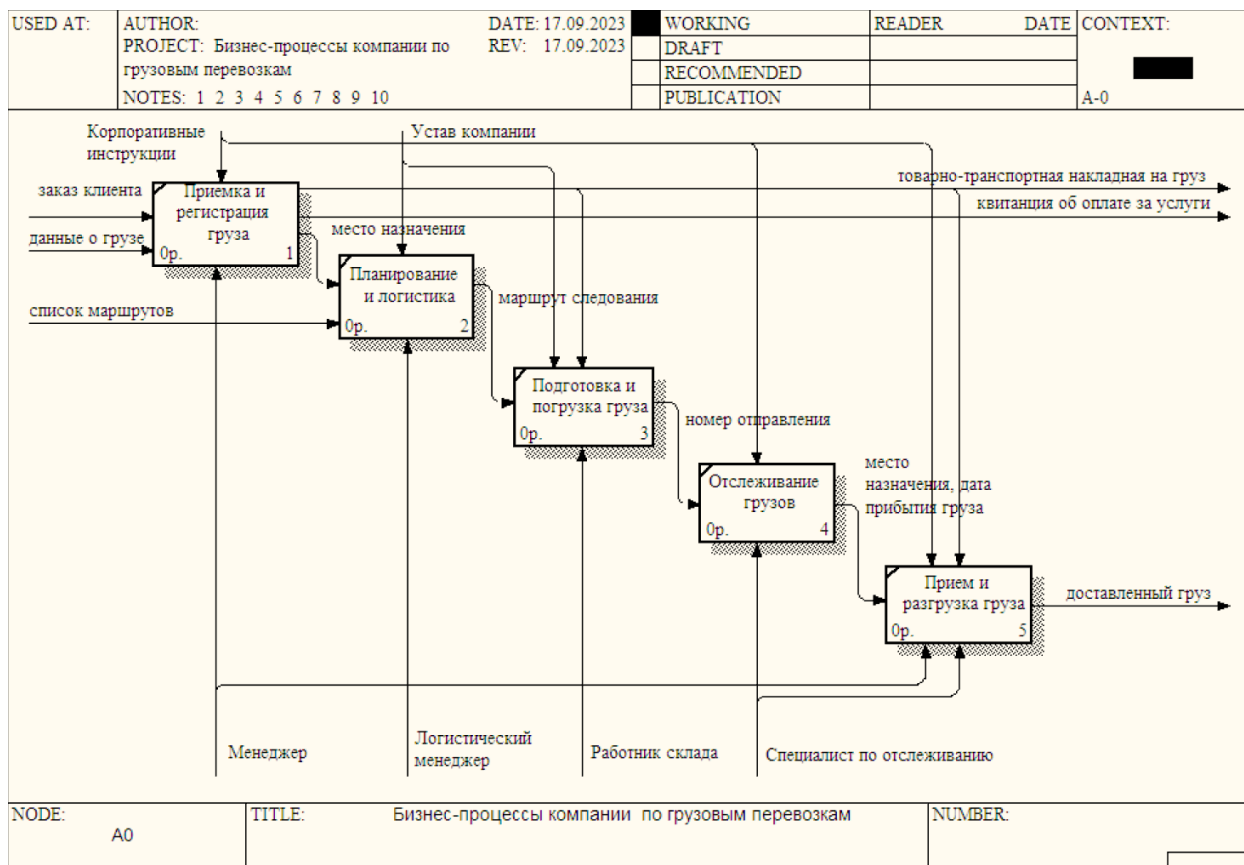


Рисунок 3 – Декомпозиция процессов компании, модель «как есть»

Отгрузка грузов в компании осуществляются согласно расписанию маршрутов поездов, прием и регистрация происходит достаточно быстро, однако процесс тормозит ручное заполнение и передача документов на груз, таких как товарно-транспортная накладная, которые обеспечивают сохранность груза и актуальную информацию о грузе.

### 1.2.3 Обоснование необходимости автоматизированного варианта решения

Процесс создания и передачи товарно-транспортной накладной (ТТН) при железнодорожной перевозке без автоматизации включает в себя несколько этапов и довольно много ручных операций.

Менеджер заполняет ТТН, в которой указывает данные о грузе, отправителе, получателе, маршруте, количестве и характеристиках груза, а также другие необходимые сведения.

ТТН может потребовать согласования с получателем, чтобы подтвердить прием груза и другие условия доставки. ТТН подписывается отправителем и железнодорожной компанией, что подтверждает условия перевозки и ответственность за груз.

Груз погружается на железнодорожные вагоны, при этом должны соблюдаться правила безопасности и укладки груза. Менеджер передает железнодорожной компании оригинал ТТН, который будет сопровождать груз во время перевозки.

В зависимости от характера груза и требований железнодорожной компании могут потребоваться дополнительные документы и разрешения. Груз отправляется по маршруту, указанному в ТТН. Машины или поезда следуют по железной дороге с учетом графика движения. Железнодорожная компания отслеживает движение груза и обновляет информацию о его местоположении.

По прибытии на конечный пункт назначения груз разгружается с вагонов. Получатель проверяет груз на целостность и соответствие указанным в ТТН данным, затем подписывает ТТН как подтверждение получения груза.

Получатель возвращает оригинал ТТН отправителю. Отправитель и получатель архивируют копии ТТН и других связанных документов для будущих потребностей и контроля.

Этот процесс предполагает множество ручных операций и документооборота, что может быть затратным и времязатратным. Он также подвержен риску ошибок при заполнении и обработке бумажных документов. Автоматизация этого процесса с использованием специализированных информационных систем и электронных ТТН может значительно упростить его и уменьшить вероятность ошибок [4].

В процессе приемки и отгрузки грузов при железнодорожных перевозках входят различные документы, которые играют важную роль в

организации и учете грузов. Опишем входящие документы на этапах приемки и отгрузки грузов.

При приемке грузов поступает заявка на перевозку. Это документ, который отправитель (грузоотправитель) предоставляет железнодорожной компании, чтобы заказать перевозку. Заявка содержит информацию о грузе, его объеме, массе, маршруте и других деталях.

Договор между грузоотправителем и железнодорожной компанией, который устанавливает условия перевозки, стоимость, ответственность сторон и другие важные детали.

Товарно-транспортная накладная (ТТН). Этот документ является основным для железнодорожных перевозок. В ТТН указываются данные о грузе, отправителе, получателе, маршруте, количестве груза, весе и объеме, а также другие сведения. Она подписывается отправителем и железнодорожной компанией, что подтверждает условия перевозки.

Оригинал ТТН, который отправитель передает железнодорожной компании при приемке груза, также используется при отгрузке. Она служит для подтверждения факта приемки и условий перевозки.

Документы, такие как разрешения на перевозку опасных грузов или другие специализированные разрешения, могут потребоваться в случае перевозки особых видов грузов.

Эти документы играют важную роль в процессе приемки и отгрузки грузов, так как они фиксируют условия перевозки, стоимость, ответственность и другие ключевые параметры.

### **1.3 Анализ существующих разработок на предмет соответствия сформулированным требованиям**

Целесообразно провести анализ существующих разработок на рынке, предлагающие подобные программные решения.



Рассмотрим несколько готовых программных решений, которые предлагает рынок IT-разработок.

Критериями оценки программных разработок будет надежность, безопасность данных, наличие пользовательского интерфейса и функций для создания и передачи сопроводительной документации на железнодорожные перевозки.

### **1.3.1 Сравнительная характеристика существующих разработок**

«Умная Логистика - CRM-сервис для автоматизации рабочей деятельности компаний-грузоперевозчиков. Дает возможность существенно ускорить формирование документации, быстро проанализировать действия сотрудников при необходимости, вести бухгалтерский учет. Кроме того, с помощью данной программы можно определить общее состояние предприятия. В информационной базе содержатся все требуемые показатели» [9].

«Цена на ПО варьируется от 2990 до 14500 рублей в месяц» [4].

«CRM-продукт, позволяющий оптимизировать работу отдела по транспорту. Обладает расширенным функционалом, направленным на успешное решение определенных задач. В их числе» [5]:

- отслеживание транспорта компании;
- формирование путей маршрута;
- формирование заявок на груз;
- формирование счетов.

Достоинства:

- автоматизация;
- улучшенная видимость;
- оптимизация маршрутов;
- управление изменениями;
- генерация документации;
- аналитика и отчетность;
- удобство использования.

Недостатки:

- зависимость от технических средств;
- интеграция;
- безопасность данных;
- обучение пользователей;
- зависимость от интернет-соединения.

Пример приложения на рисунке 4.

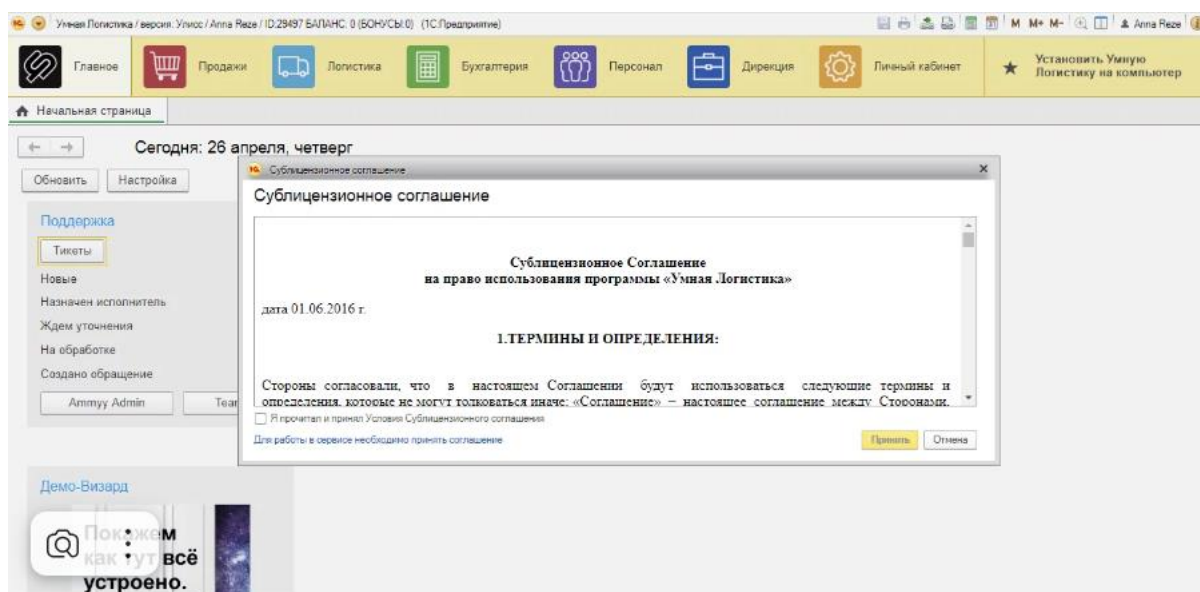


Рисунок 4 – Приложение «Умная Логистика»

«КиберЛог - удобный сервис-облако, позволяющий управлять транспортировкой. Система помогает упростить взаимодействие сторон, участвующие в бизнес-процессах. Предназначена для осуществления документооборота между заказчиками и исполнителями, обмена информацией, контроля выполнения договорных условий и формирования необходимой отчетности» [10].

«За использование ПО предстоит заплатить всего 400-450 рублей. Предоставляется пробный период 30 дней» [8].

Достоинства:

- автоматизация;
- улучшенная видимость;
- оптимизация ресурсов;
- интеграция;
- управление изменениями.

Недостатки:

- сложность внедрения;
- интеграционные проблемы;
- зависимость от технических средств;
- безопасность данных;
- зависимость от интернет-соединения.

Пример окна представлен на рисунке 5.

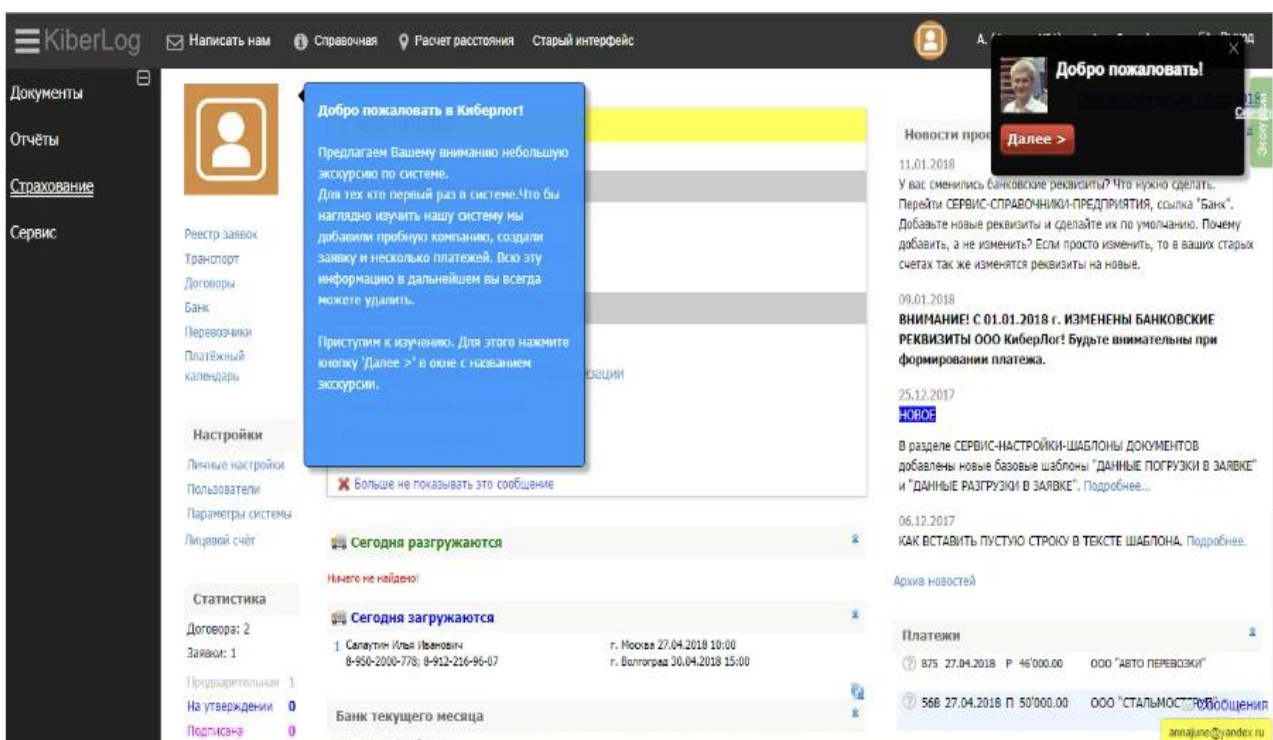


Рисунок 5 – Пример окна приложения «КиберЛог»

«Муравьиная логистика - многофункциональная программа для работы логистов по грузоперевозкам (построение и составление маршрутов

доставки). С ее помощью можно планировать и отслеживать соответствие заданным маршрутам, формировать наиболее подходящие пути, наблюдать за выполнением задач» [11].

Достоинства:

- автоматизация;
- улучшенная видимость;
- оптимизация ресурсов;
- интеграция;
- управление изменениями.

Недостатки:

- сложность внедрения;
- интеграционные проблемы;
- зависимость от технических средств;
- безопасность данных;
- зависимость от интернет-соединения.

«Рассмотренные решения больше представляют CRM-систему, которая контактирует с покупателем и предлагает ранее оказанные услуги или отслеживает предпочтения клиентов по оказанным услугам» [12].

Целесообразно будет провести разработку приложения для автоматизации процессов создания и передачи товаросопроводительной документации на железнодорожные грузоперевозки в отделе IT-технологий.

#### **1.4 Разработка модели бизнес-процесса «как должно быть»**

На основе построенных диаграмм процессов компании модели «как есть» были выявлены проблемы в процессах компании. Построим модели «как должно быть».

Рисунок 6 представляет контекстную диаграмму процессов компании, модель «как должно быть».

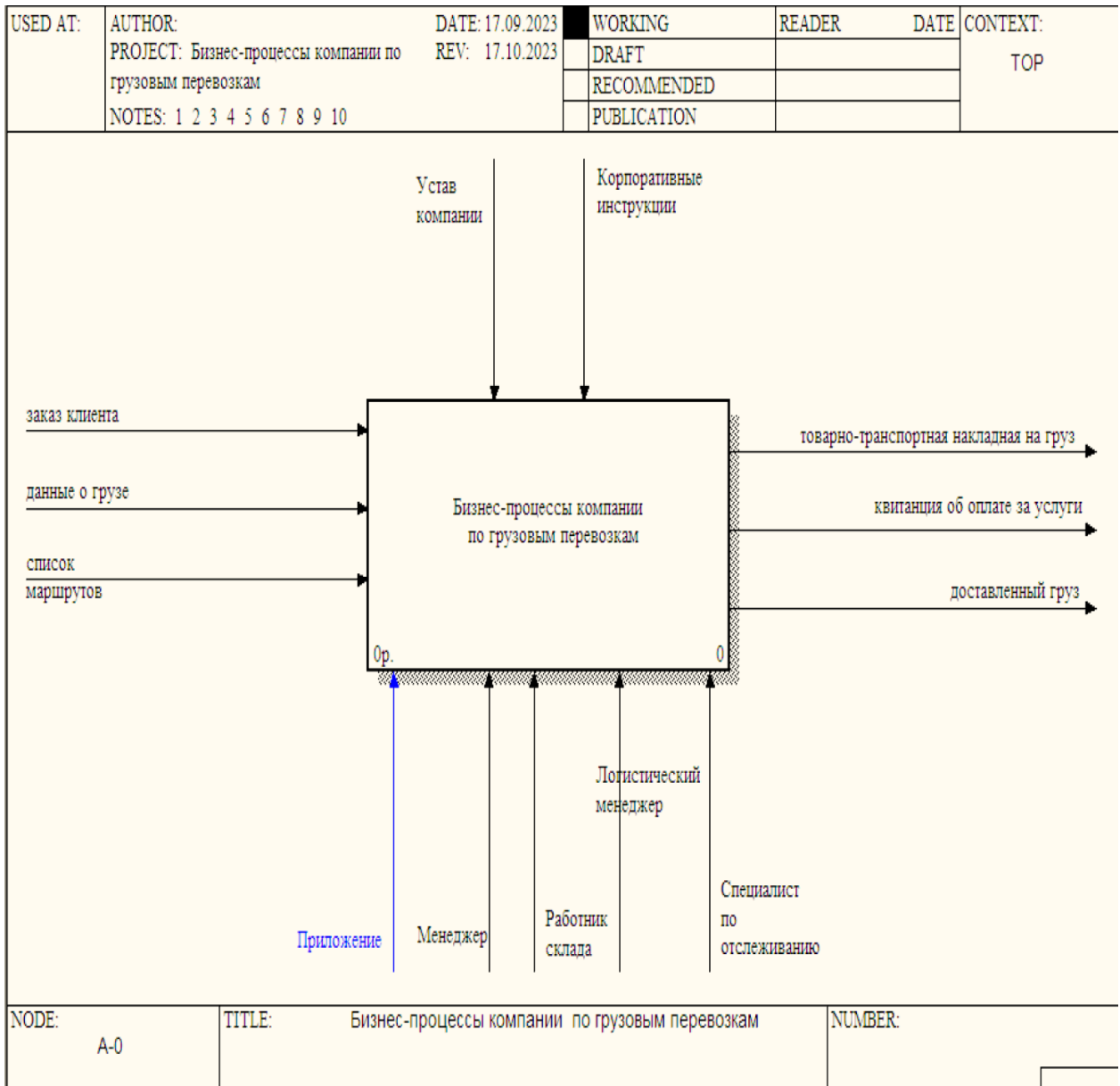


Рисунок 6 – Контекстная диаграмма процессов компании, модель «как должно быть»

На диаграмме показана информационная система, с помощью которой сотрудники компании будут осуществлять процессы.

Проведем декомпозицию диаграммы процессов компании по грузовым перевозкам «как есть», диаграмма показана на рисунке 7. Процесс разбит на несколько блоков подпроцессов, которые более детально показывают, как осуществляются прием и отправка грузов.

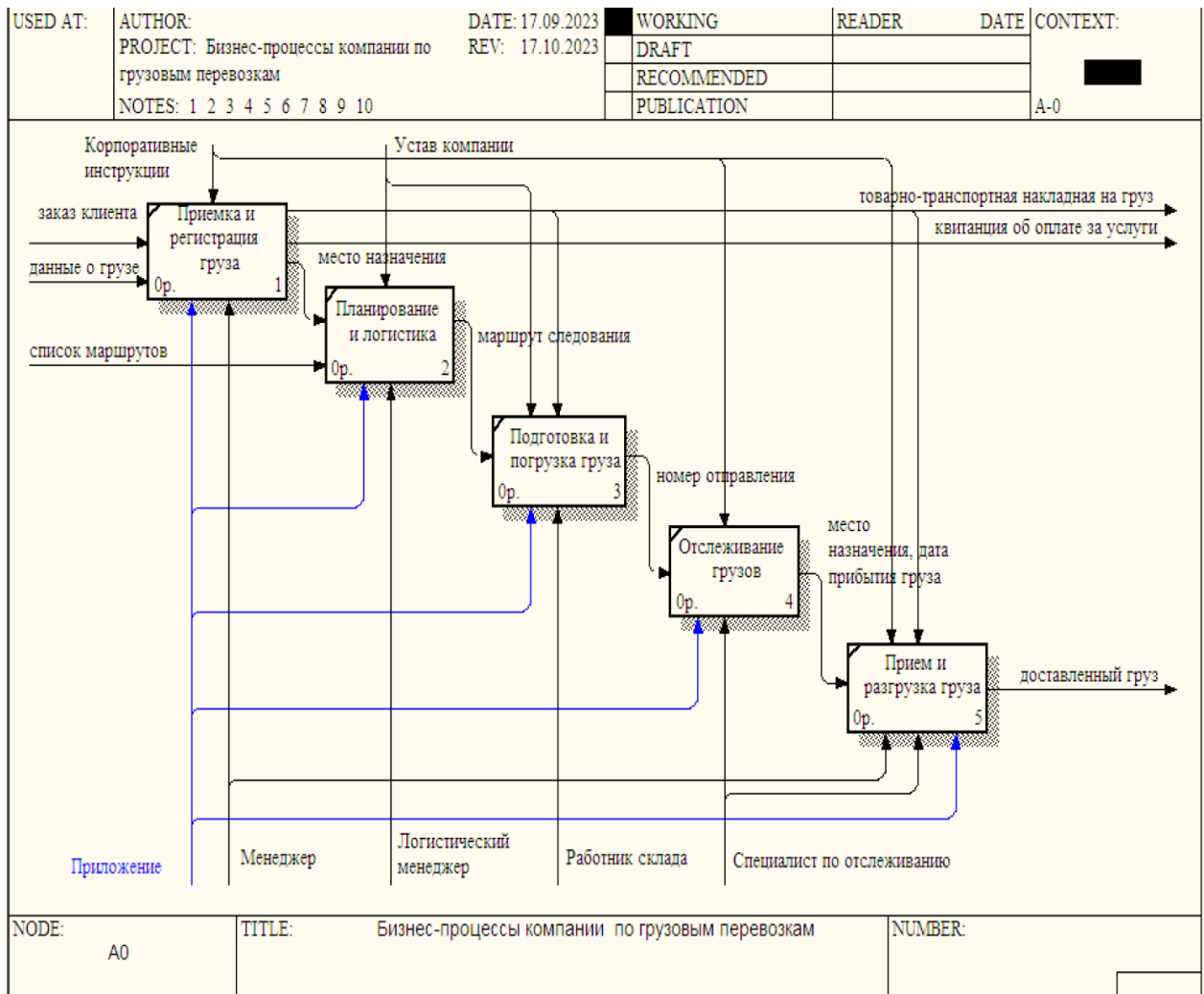


Рисунок 7 – Декомпозиция процессов компании, модель «как должно быть»

На рисунке 7 показана модель «как должно быть» где показано, что все сотрудники компании, которые участвуют в процессах, связанных с грузовыми перевозками, будут использовать информационную систему. Все документы, сопровождающие груз будут создаваться в системе и передаваться между отделами компании внутри системы.

## 1.5 Постановка задачи на разработку проекта создания АИС

### Введение

Техническое задание написано на разработку приложения для автоматизации процессов создания и передачи товаросопроводительной документации на железнодорожные грузоперевозки.

## Основание для разработки

Приложение разрабатывается на основании распоряжения директора компании ООО «Информсистема».

## Назначение

Цель приложения состоит в автоматизации и оптимизации бизнес-процессов, связанных с созданием, обработкой и передачей сопроводительной документации при железнодорожных грузоперевозках. Оно служит инструментом для управления и контроля всеми этапами перевозок, начиная от создания заказа и завершая передачей груза получателю.

## Требования к функциональным характеристикам.

Приложение будет позволять:

- создание и редактирование документов;
- управление информацией о грузе;
- интеграция с железнодорожными компаниями;
- управление данными и архивирование;
- генерация отчетов.

## Требования к надежности.

Приложение должно быть доступным для пользователей 24/7. Система должна быть спроектирована с учетом возможности бесперебойной работы даже в случае сбоев или отказов.

Приложение должно способствовать обработке высокой нагрузки, особенно в периоды пиковой активности, такие как праздничные сезоны или акции. Это предотвращает зависание или замедление работы системы.

Должна быть реализована система регулярного резервного копирования данных и механизмы восстановления в случае сбоев или потери данных.

Система должна гарантировать целостность данных и предотвращать их несанкционированное изменение или повреждение.

## Системные требования:

- Pentium IV;
- ОЗУ от 512 Мб.

Требования к программной и информационной совместимости.

Приложение должно быть реализовано так чтобы оно могло функционировать в ОС Windows 10 и версии выше.

Требования к программной документации

К готовому приложению должны быть предоставлены документы такие как: руководство пользователя и руководство администратора.

Технико-экономические показатели

Показатели выгоды сложно предсказать пока приложение не будет внедрено и протестировано в рабочем режиме.

Стадии и этапы разработки

Этапы разработки заранее согласовываются с заказчиком [14].

Таблица 1 – Стадии и этапы разработки

Стадии разработки	Содержание работ	Исполнитель этапа разработки
Формирование требований к разработке	Исследование предметной области	аналитик
	Обоснование актуальности разработки	аналитик
	Выбор бизнес-процесса для автоматизации	заказчик
	Определение требований	заказчик
Проектирование	Выбор модели и методологии проектирования и разработки	проектировщик
	Выбор инструментов разработки	проектировщик
	Проектирование программной структуры	Проектировщик
	Инфологическое проектирование	проектировщик



### Продолжение таблицы 1

Реализация	Конфигурация инструментов разработки	программист
	Программная реализация	программист
	Реализация базы данных	Программист
Тестирование	Модульное тестирование	программист
	Интеграционное тестирование	программист
Внедрение	Установка	программист
	Написание руководства пользователя	программист
	Написание руководства администратора	программист

#### Порядок контроля и приемки

Порядок контроля и приемки приложения включает в себя следующие этапы:

- подготовка к контролю и приемке;
- тестирование;
- устранение ошибок;
- аудит безопасности;
- приемочные тесты;
- оценка результатов;
- подписание акта приемки;
- внедрение [7].

В системе автоматизации железнодорожных грузоперевозок участвует несколько видов пользователей, каждый из которых выполняет свои функции и обладает определенными правами доступа.

#### Список основных категорий пользователей:

- администраторы системы;
- менеджеры.

Администратор системы в системе выполняет ряд ключевых функций,

которые обеспечивают надежное и эффективное функционирование приложения и обеспечивают безопасность, мониторинг и настройку системы.

Основные функции администратора:

- управление пользователями;
- настройка системы;
- ввод новых данных;
- резервное копирование и восстановление;
- поддержка пользователей.

Менеджер в системе выполняет ряд ключевых функций, которые связаны с управлением и координацией грузоперевозок.

Основные функции менеджера:

- создание новых заказов на грузоперевозку;
- редактирование существующих заказов;
- отслеживание статусов всех заказов;
- генерация необходимых сопроводительных документов.

Техническое задание значительно упростит и ускорит работу разработчиков приложения.

Выводы по первой главе

В главе был проведен обзор литературы, что позволило понять текущее состояние знаний в данной области. В главе были описаны методы, используемые для сбора и анализа данных.

Был проведен обзор существующих исследований и теорий, связанных с темой исследования. Это помогло определить области, которые еще не были исследованы или требуют дополнительного изучения.

Моделирование бизнес-процессов позволило провести анализ деятельности грузовой компании ООО «Информсистема», выяснилось, что процесс создания и передачи сопроводительной документации выполняется вручную. На основе чего было принято решение автоматизировать данный процесс путем разработки и внедрения приложения.

## **Глава 2 Логическое проектирование приложения**

### **2.1 Выбор технологии логического моделирования**

Логическое моделирование приложения представлено как процесс создания абстрактной схемы приложения, которая отражает его логику и структуру приложения.

В процессе логического моделирования определяются ключевые компоненты приложения, их взаимосвязи, логику работы, а также важные аспекты, такие как обработка данных, бизнес-правила и взаимодействие с пользователями. Цель состоит в том, чтобы создать абстрактное представление, которое будет понятным для всех участников проекта — разработчиков, дизайнеров, менеджеров и заказчиков.

Существует несколько технологий логического моделирования, рассмотрим несколько из них и выберем которое больше подходит для моделирования приложения.

Унифицированный язык моделирования UML, он достаточно широко используется программистами в области разработки ПО. С помощью UML можно построить различные диаграммы такие как диаграммы классов, вариантов использования и многие другие, отражающие логику программного обеспечения.

Entity-Relationship Diagrams применяется в большей степени при моделировании базы данных их сущностей и связей.

BRMN больше используется для моделирования бизнес-процессов организации, такой способ моделирования больше отражает шаги бизнес-процесса и взаимодействие участников в них.

Для логического моделирования приложения, которое будет автоматизировать процесс создания и передачи сопроводительной документации на железнодорожные грузоперевозки подходит язык

моделирования UML, который наиболее четко отразит логику приложения с помощью различных диаграмм.

## 2.2 Логическая модель приложения

Диаграмма прецедентов является важным инструментом моделирования в системной инженерии и помогает визуализировать функциональные возможности системы и взаимодействие её сущностей (пользователей). На рисунке 8 представлена диаграмма прецедентов для разрабатываемого приложения.

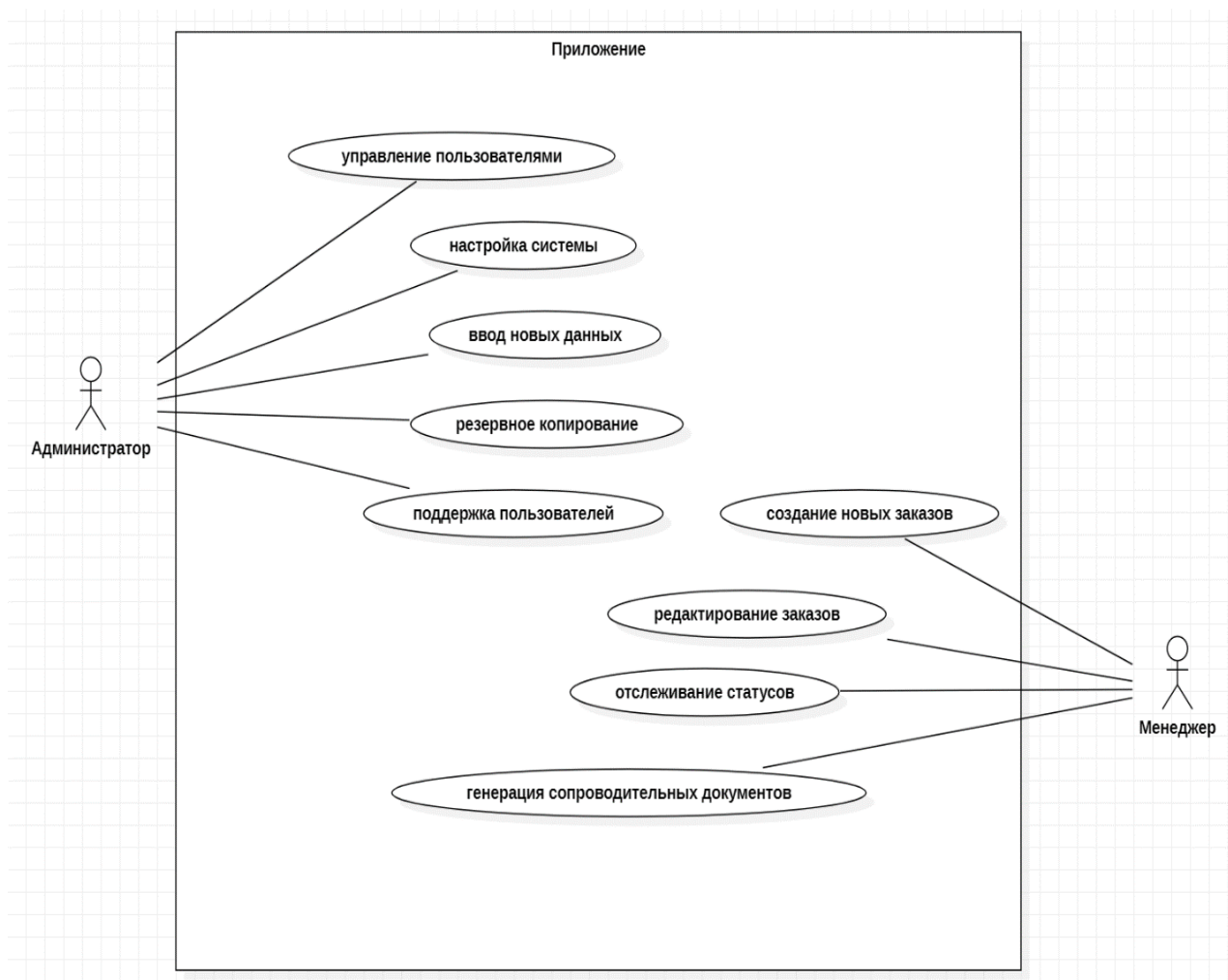


Рисунок 8 – Диаграмма вариантов использования

Функциональные возможности приложения для автоматизации процессов создания и передачи сопроводительной документации на железнодорожные грузоперевозки могут быть очень разнообразными, в зависимости от требований и целей системы [8]:

- создание и редактирование заказов;
- отслеживание статуса заказов;
- генерация сопроводительной документации;
- управление изменениями в заказе;
- мониторинг и анализ данных;
- администрирование и управление пользователями.

Нефункциональные требования к приложению:

- безопасность;
- производительность;
- надежность;
- удобство использования;
- скорость загрузки;
- интеграция;
- масштабируемость;
- отказоустойчивость;
- совместимость и переносимость.

Построим диаграмму последовательности процесса создания сопроводительного документа для железнодорожной грузоперевозки. На рисунке 9 представлена диаграмма, которая будет отражать процесс по шагам.

На диаграмме показано два объекта системы, это менеджер и приложение, в котором он будет создавать сопроводительные документы для отправки груза.

Все введенные данные будут автоматически сохраняться в базу данных после подтверждения их на сохранение.

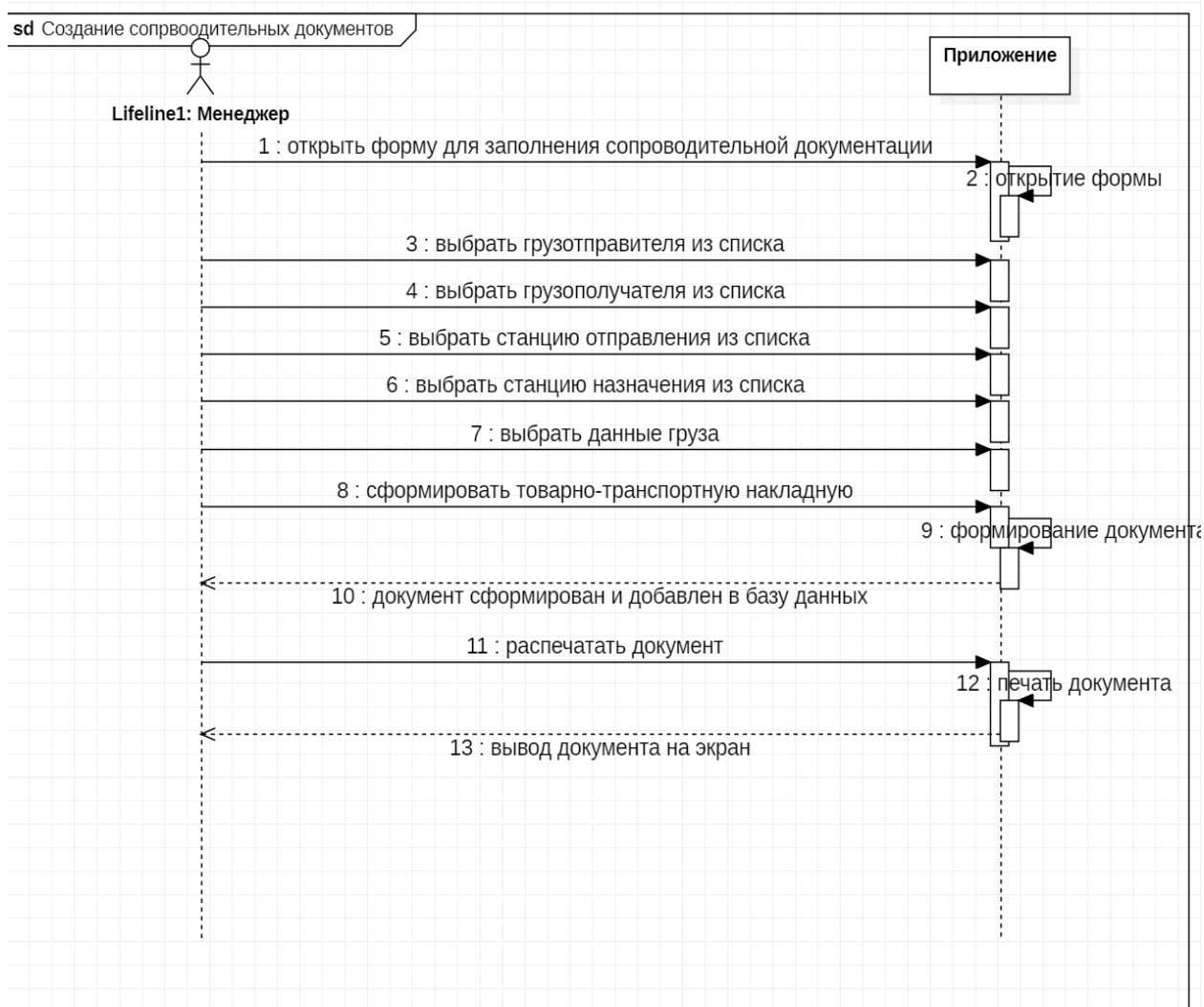


Рисунок 9 – Диаграмма последовательности процесса создания сопроводительных документов

На диаграмме показано как менеджер компании создает сопроводительные документы на груз, в приложении он выбирает данные грузополучателя, грузоотправителя и информацию о грузе, которые уже внесены в базу данных и доступны менеджеру при заполнении специальной формы документа.

При выборе данных из списка менеджер может с помощью приложения сохранить документ в базу данных и сформировать его в формат pdf после чего распечатать.

## **2.3 Информационное обеспечение приложения**

### **2.3.1 Характеристика нормативно-справочной, входной и оперативной информации**

Перевозка грузов по железной дороге имеет свои правила и нормативные документы.

Перевозка грузов железнодорожным транспортом в международном и транзитном сообщении через Российскую Федерацию осуществляется в соответствии с законодательством Российской Федерации и международными договорами, ратифицированными Российской Федерацией [9].

Технические условия размещения и крепления грузов, порядок и условия перевозок жидких грузов наливом, опасных грузов, не предусмотренные настоящими Правилами, устанавливаются международными соглашениями, принятыми в рамках Организации сотрудничества железных дорог и Совета по железнодорожному транспорту стран-участниц Содружества независимых государств и иными международными соглашениями.

Для осуществления перевозки грузов между перевозчиком и грузоотправителем заключается договор перевозки согласно статье 689 Гражданского кодекса Российской Федерации, по которому перевозчик обязуется своевременно и в сохранности доставить вверенный ему грузоотправителем груз, со станции отправления до станции назначения и выдать грузополучателю, а грузоотправитель обязуется оплатить перевозку груза и обеспечить его приемку.

Договор перевозки грузов оформляется составлением железнодорожной транспортной накладной [10].

Накладная с проставленным календарным штампом и выданная на ее основании перевозчиком грузоотправителю квитанция о приеме груза (при

перевозке в международном сообщении – дубликат накладной) подтверждают заключение договора перевозки.

Договор перевозки считается исполненным после получения грузополучателем или лицом, им уполномоченным железнодорожной транспортной накладной и подписания им дорожной ведомости, а также приемосдаточных документов о приеме груза.

Перевозка грузов по магистральной железнодорожной сети осуществляется перевозчиком на основании принятой заявки грузоотправителя на перевозку грузов. Перевозчики на основании заявок грузоотправителей формируют план перевозок грузов и предоставляют Национальному оператору инфраструктуры заявку на включение их поездов в график движения.

### **2.3.2 Характеристика выходной информации**

Железнодорожные товарно-транспортные накладные являются документами, регулирующими отправку и доставку грузов по железной дороге. Они содержат информацию о грузе, отправителе, получателе, станциях назначения и отправления, а также условиях перевозки.

Перевозочные документы являются неотъемлемой частью процесса перевозки груза. Они не только закрепляют условия перевозки, но и являются основным доказательством прав на груз и взаимоотношений между грузовладельцем и перевозчиком.

Оформление перевозочных документов — важный этап в процессе осуществления грузоперевозки. Эти документы необходимы как для заказчика перевозки, так и для перевозчика, поскольку осуществляют регулирование отношений между ними и подтверждают факт отправления и доставки груза.

Процесс оформления перевозочных документов включает в себя несколько этапов:

- составление договора о перевозке;
- подготовка накладной;



- оформление товаросопроводительных документов;
- формирование экспедиционного документа;
- составление акта приема-передачи груза.

Весь процесс оформления перевозочных документов тщательно контролируется и регулируется, чтобы обеспечить безопасность и надежность перевозки и защитить интересы заказчика и перевозчика. Корректно оформленные документы являются гарантией исполнения условий договора и важными инструментами при разрешении возникающих споров и претензий.

## **2.4 Проектирование базы данных**

### **2.4.1 Выбор технологии проектирования базы данных**

Существует несколько технологий проектирования базы данных:

- ER-моделирование;
- реляционное моделирование;
- нормализация данных;
- NoSQL-технологии;
- CASE-системы.

«ER-моделирование подходит для проектирования базы данных, так как с помощью моделирования можно показать сущности и описать их взаимосвязи в базе данных. ER-диаграммы помогают понять структуру данных и их взаимосвязи» [11].

«Логическая модель данных является визуальным графическим представлением структур данных, их атрибутов и связей. Логическая модель представляет данные таким образом, чтобы они легко воспринимались бизнес-пользователями. Проектирование логической модели должно быть свободно от требований платформы и языка реализации или способа дальнейшего использования данных.

При разработке используются требования к данным и результаты анализа для формирования логической модели данных. Логическую модель приводят к третьей нормальной форме, и проверяет ее на соответствие модели процессов» [12].

Для создания концептуальной и логической модели необходимо определить сущности, которые будут содержать данные приложения.

#### 2.4.2 Разработка концептуальной модели данных

Для приложения автоматизации создания сопроводительной документации грузоперевозок потребуется несколько сущностей, которые будут содержать информацию о грузоотправителе и грузополучателе, это могут быть как физические лица, так и юридические. Также необходимо хранить данные и станциях назначения и отправления.

Построим ER-модель данных, рисунок 10.

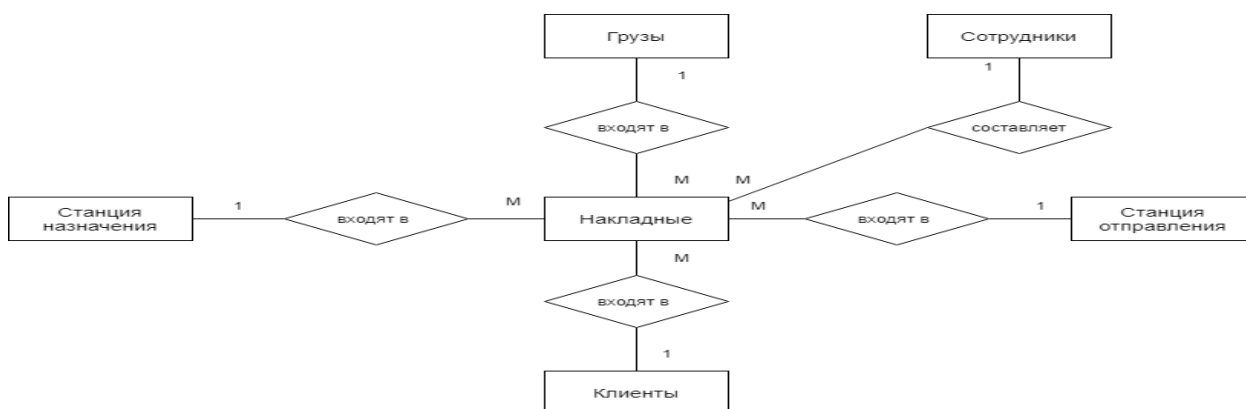


Рисунок 10 – ER-модель

Сущность Клиенты содержит один составной атрибут адрес, который состоит из простых атрибутов: город, улица, дом.

Сущность Сотрудник содержит один составной атрибут адрес, который состоит из простых атрибутов: город, улица, дом.

Многозначный атрибут – это список (массив) однотипных значений. Многозначных атрибутов в наших сущностях нет.

Выявим n-арные связи и представим их в виде слабой сущности на ER-диаграмме.

Слабой сущностью будет накладная.

Сущность Накладная имеет бинарную связь с сущностями: грузы, клиенты, станция назначения, станция отправления, сотрудники.

Создадим концептуальную модель с помощью специализированной программы для моделирования моделей базы данных ErWin. На рисунке 11 показана концептуальная модель данных.

Сущность Грузы имеет суррогатный ключ код груза, могут быть одинаковые наименования грузов, он же и является ключевым атрибутом. Внешних ключей, в сущности, нет.

«Сущность Клиенты имеет суррогатный ключ код клиента, могут быть однофамильцы, он же и является ключевым атрибутом. Внешних ключей, в сущности, нет» [3].

Сущность Станция назначения имеет суррогатный ключ код станции, так как могут быть одинаковые названия станций, он же и является ключевым атрибутом. Внешних ключей, в сущности, нет.

Сущность Станция отправления имеет суррогатный ключ код станции, так как могут быть одинаковые названия станций, он же и является ключевым атрибутом. Внешних ключей, в сущности, нет.

«Сущность Сотрудники имеет суррогатный ключ код сотрудника, могут быть однофамильцы, он же и является ключевым атрибутом. Внешних ключей, в сущности, нет» [2].

Сущность Накладная имеет суррогатный ключ номер, так как накладные могут повторяться, он же и является ключевым атрибутом. Внешним ключом, в сущности, будет код груза, код сотрудника, код станции назначения, код станции отправления, код грузополучателя, код грузоотправителя, которые обеспечат связь с сущностями станция назначения, станция отправления, сотрудники, клиенты, грузы.

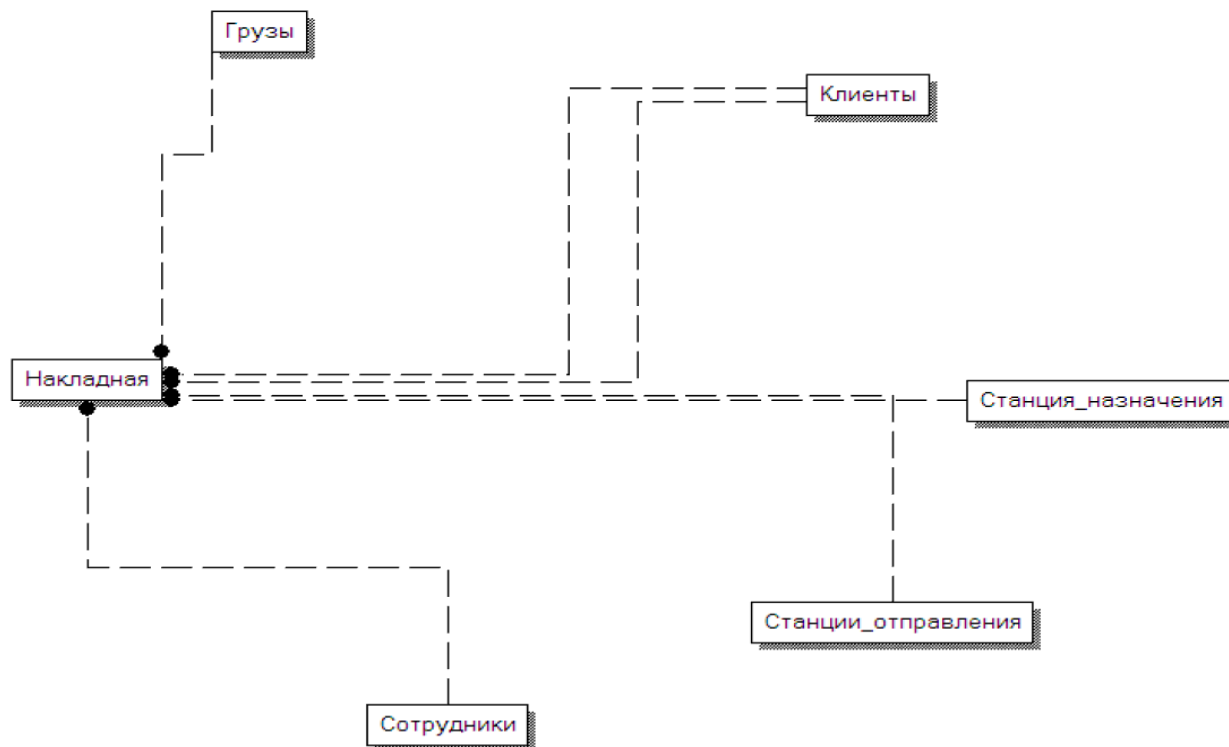


Рисунок 11 – ER-модель

### 2.4.3 Обоснование вида логической модели

Рассмотрим спецификации соединений сущностей.

Груз может существовать без накладной, до отправки, например.

Клиенты (грузоотправитель или грузополучатель) могут не иметь накладных.

Станции назначения или отправления могут не входить в накладную.

На рисунке 12 показана модель с первичными ключами.

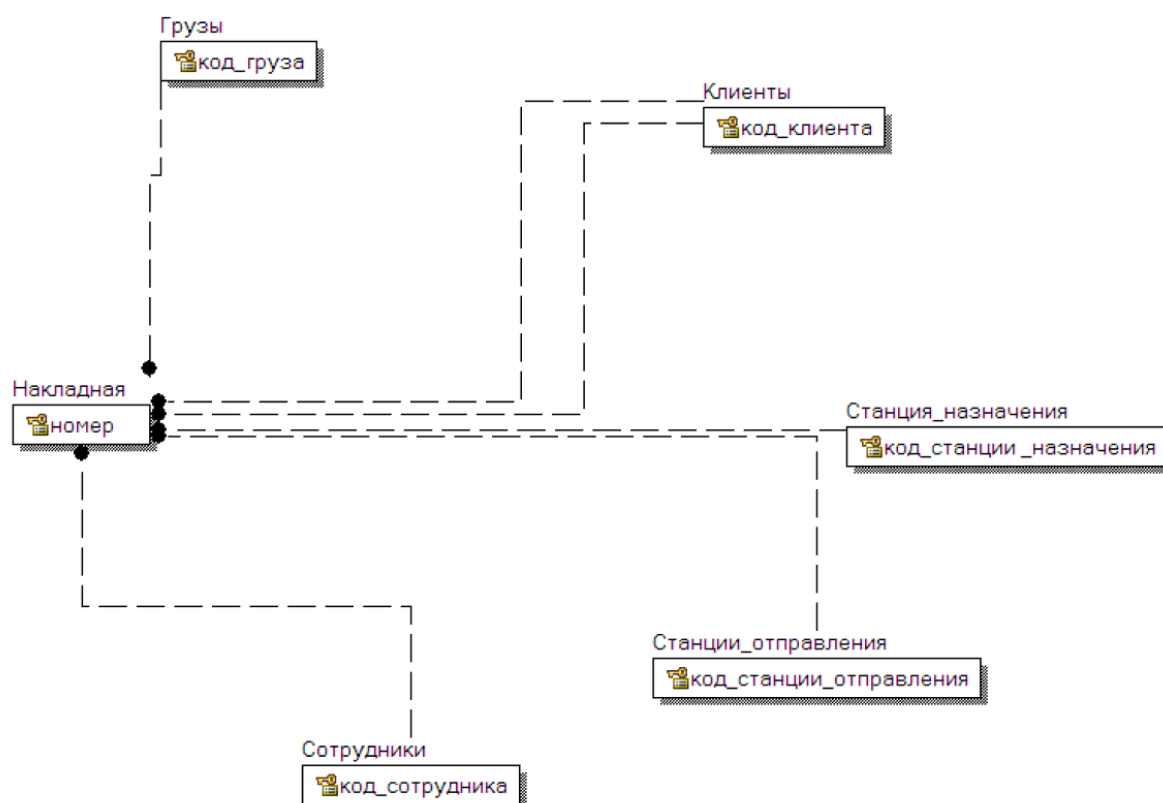


Рисунок 12 - Модель с первичными ключами

При исследовании не было обнаружено новых атрибутов. Разработаем поля для каждой сущности и определим их типы, чтобы построить логическую модель данных [13].

#### 2.4.4 Разработка логической модели данных

Каждая таблица, содержащая информацию, должна иметь поля и типы данных, разработаем глоссарий и представим в таблице 2, где будут показаны все ограничения таблиц, поля и типы данных.

Таблица 2 – Глоссарий

Имя	Смысл	Тип	Сущность	Ограничение
вес груза	вес отправляемого груза	текст	грузы	числовой
количество мест	количество мест	число	грузы	Числовой
объявленная стоимость	объявленная стоимость	денежный	грузы	денежный
описание	описание	текст	грузы	тестовый
фамилия	фамилия	текст	клиенты	тестовый
имя	имя	текст	клиенты	тестовый
отчество	отчество	текст	клиенты	тестовый
название компании	название компании	текст	клиенты	тестовый
адрес	адрес	текст	клиенты	тестовый
телефон	телефон	текст	клиенты	тестовый
тип клиента	тип клиента	текст	клиенты	тестовый
название	Название станции назначения	текст	станция назначения	тестовый
название	Название станции отправления	текст	станция отправления	тестовый
фамилия	фамилия	текст	сотрудники	тестовый
имя	имя	текст	сотрудники	тестовый
отчество	отчество	текст	сотрудники	тестовый
должность	должность	текст	сотрудники	тестовый
код груза	код груза	число	накладная	числовой
код сотрудника	код сотрудника	число	накладная	числовой
код станции назначения	код станции назначения	число	накладная	числовой
код станции отправления	код станции отправления	число	накладная	числовой
грузоотправитель	грузоотправитель	число	накладная	Числовой
грузополучатель	грузополучатель	число	накладная	числовой
дата отправления	дата отправления	дата	накладная	дата

Когда определены все типы данных и установлены ограничения можно построить логическую модель данных, представим на рисунке 13.

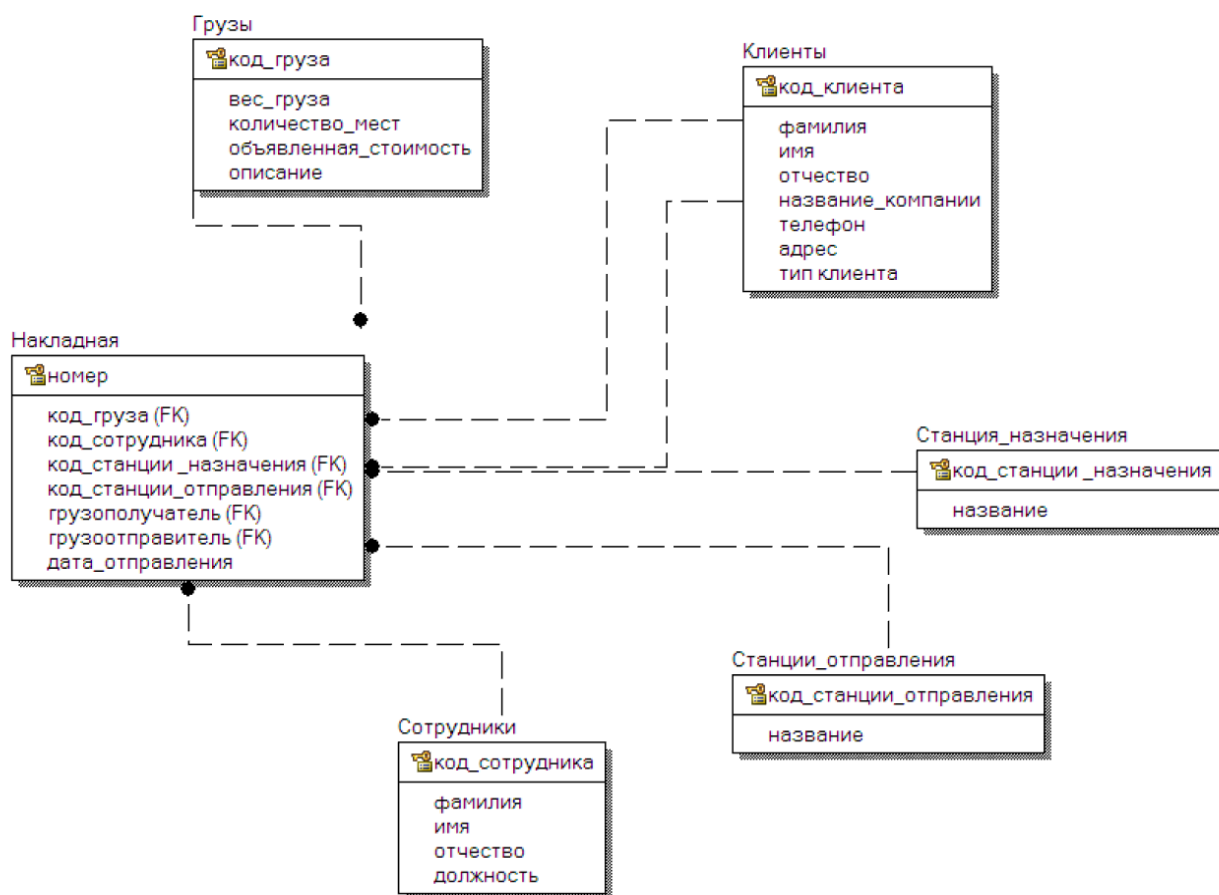


Рисунок 13 – Логическая модель данных

На основе выполненного логического проектирования приложения можно перейти к требованиям к аппаратно-программному обеспечению.

## 2.5 Требования к аппаратно-программному обеспечению

Для хранения данных необходимо хранилище данных с возможностью масштабирования для обеспечения эффективного хранения информации о грузоперевозках и сопроводительной документации. Для этой цели в компании необходимо приобрести и установить сервер с достаточным объемом оперативной памяти и процессорной мощности.

Взаимодействие между пользователем системы и ее модулями необходимо сетевое оборудование быстрое и надежное [14].

Операционная система должна поддерживать Windows Server и Linux.

Так как ранее были разработаны модели данных для реляционной базы данных, то для эффективного управления связанными данными должна использоваться СУБД по типу MySQL, PostgreSQL, или Microsoft SQL Server.

Для написания приложения может быть выбран любой язык программирования высокого уровня, для ускорения разработки можно использовать фреймворки, например, Django, Spring, ASP.NET.

Интерфейс пользователя должен быть понятным обычному пользователю не имеющего специального образования или навыков.

Архитектура приложения должна позволять масштабировать приложения на основе потребностей компании грузоперевозок.

#### Выводы по второй главе

Во второй главе проведено логическое проектирование приложения, в котором было проведено разработка и моделирование разных уровней модели данных. Определены все необходимые сущности, типы данных и связи между ними. На основе моделирования были представлены требования к аппаратно-программному обеспечению приложения.



## **Глава 3 Физическое проектирование приложения**

### **3.1 Выбор архитектуры приложения**

Рассмотрим несколько распространенных архитектур приложения, чтобы определить какая из них подойдет для приложения, которое будет разрабатываться для компании.

Монолитная архитектура характеризуется единым монолитным кодом, который развертывается единым приложением.

Архитектура имеет достоинства такие как простота разработки и тестирование, также легкость развертывания и масштабирования. Такая архитектура имеет и недостатки с трудностью в обновлении отдельных компонентов, также и затруднения в сопровождении с ростом функциональности [15].

Микросервисная архитектура подразумевает что приложение состоит из нескольких автономных сервисов, каждый из которых отвечает за определенную бизнес-функциональность.

Преимущества данной архитектуры заключаются в высокой гибкости и масштабируемости. Недостатки тоже есть, возможные проблемы согласованности данных между сервисами, а также дополнительные затраты на управление инфраструктурой.

Клиент-серверная архитектура имеет клиента и сервер.

Клиент — это устройство или программное обеспечение, которое обращается к серверу для получения услуг или ресурсов.

Сервер — это устройство или программное обеспечение, которое предоставляет услуги, ресурсы или данные клиентам.

Функциональность клиент-серверной архитектуры позволяет обрабатывать запросы от клиентов, управлять ресурсами и выполнять операции, необходимые для клиентов.

Клиент-серверная архитектура самая подходящая для реализации приложения компании.

### **3.2 Выбор технологии разработки программного обеспечения**

Для написания приложения нужно сравнить и выбрать наиболее оптимальный язык программирования. Рассмотрим C++, C# и Java, наиболее распространенные.

C++ – чрезвычайно мощный язык, содержащий средства создания эффективных программ практически любого назначения, от низкоуровневых утилит и драйверов до сложных программных комплексов самого различного назначения. Однако недостатки не подходят нам для разработки, такие как синтаксис, провоцирующий ошибки, препроцессор, унаследованный от C, очень примитивен, плохая поддержка модульности (по сути, в классическом Си модульность на уровне языка отсутствует, её обеспечение переложено на компоновщика) [16].

Будучи объектно-ориентированным языком, C# много перенял у Java и C++. Как и Java, C# изначально предназначался для веб-разработки, и примерно 75% его синтаксических возможностей такие же, как у Java.

Так как в приложении будет создаваться графический интерфейс и база данных, то этот язык подходит со своей функциональностью и гибкостью [17].

Язык Java зародился как часть проекта создания передового программного обеспечения для различных бытовых приборов.

Язык Java потребовался для создания интерактивных продуктов для сети Internet. Фактически, большинство архитектурных решений, принятых при создании Java, было продиктовано желанием предоставить синтаксис, сходный с C и C++. В Java используются практически идентичные соглашения для объявления переменных, передачи параметров, операторов и

для управления потоком выполнением кода. В Java добавлены все хорошие черты C++.

Приведем сравнительную таблицу языков программирования

Таблица 3 – Результаты сравнения языков программирования

Основные отличия	C++	C#	Java
1	2	3	4
Понятный синтаксис	-	+	+
Интеграция с базой данных	+	+	+
Скорость обработки данных	+	+	-
Функциональность и гибкость языка	-	+	-
Интеграция с фреймворками	-	+	+

Сравнительный анализ помог выбрать язык программирования C#, далее можно рассмотреть СУБД для разработки базы данных.

### 3.3 Выбор системы управления базой данных приложения

Для создания приложения выберем инструмент разработки, система будет небольшой и создаваться в учебных целях, поэтому создадим пользовательский интерфейс и выберем СУБД, где будет храниться информация по сопроводительной документации на грузы.

Для сравнения возьмем наиболее распространенные СУБД такие как: SQL Server, Access и PostgreSQL.

Рассмотрим самую простую СУБД. Access можно быстро изучить, любой пользователь найдет много информации как в ней работать, поэтому она доступна для пользователей с низкой квалификацией. СУБД содержит все необходимые инструменты для создания отчетов как простых, так и вложенных, форм простого формата и на основе нескольких таблиц с

данными. В основном Access используется для создания личных баз данных, не имеющих коммерческого распространения [18].

Достоинства:

- понятный интерфейс;
- наличие русской версии;
- возможность работать в конструкторе.

Недостатки СУБД:

- отсутствие серьезной защиты информации;
- ограниченный объем информации;
- отсутствие собственного языка программирования;
- отсутствие скорости при обработке больших данных.
- СУБД PostgreSQL[19].

PostgreSQL имеет большой функционал и имеет большую надежность хранения данных от несанкционированного доступа. Производительность СУБД также показывает хорошие показатели. Огромным плюсом является то, что СУБД имеется в открытом доступе и имеет открытый код, который можно посмотреть любой пользователь, разбирающийся в программировании.

Ниже представлены несколько функциональных возможностей СУБД:

- вложенные запросы;
- представления;
- ссылочная целостность – внешние ключи;
- наследственность;
- проверка совместимости версий.

СУБД постоянно развивается благодаря открытому доступу к исходному коду, программисты-создатели эффективно исправляют ошибки, возникшие в работе с СУБД.

Однако есть и минусы, как и любой другой системы управления данными. Главным минусом является плохая генерация с языками программирования.

СУБД SQL Server разработана Microsoft и расширяет свои возможности по сей день. Объем хранения данных, может расширяться по мере наполнения информацией, без заметного уменьшения быстродействия операций с записями в многопользовательском режиме [20].

Программисты компании изрядно потрудились над обеспеченностью безопасности хранения данных. Можно с уверенностью сказать, что данные защищены от несанкционированного доступа за счет интеграции сетевой безопасности с сервером безопасности. Благодаря тому, что данные хранятся на отдельном сервере они максимально защищены и пользователь может быть ограничен в доступе к работе с данными.

Пользователь получает ответы от SQL Server путем отправки запроса на сервер и получения данных в ответ. Как правило такие СУБД выбирают для обработки данных в сети интернет.

Обслуживание SQL Server не требует определенных знаний в работе с СУБД. Она также позволяет сохранять данные во время работы и выполнять резервное копирование.

Если информационная система или любое другое приложение, которое будет содержать базу данных будет реализовываться на языке программирования C#, то SQL Server будет идеальным вариантом для этого, так как оба эти продукта являются разработками Microsoft.

На основе рассмотренных выше систем управления данными можно сделать вывод и выбрать оптимальный вариант для приложения. СУБД Microsoft SQL Server подойдет больше всех, она имеет все доступные средства для разработки приложения. Представим базы данных в сравнительной таблице 4.

Таблица 4 – Результаты сравнения баз данных

Основные отличия	Access	MS SQL	PostgreSQL
1	2	3	4
Доступность	+	+	+
Хранение большого объема данных	-	+	+
Скорость обработки операций с большими данными	-	+	-
Защита от несанкционированного доступа	-	+	+
Простота использования	+	-	-

Сравнительный анализ СУБД показал, что с применением языка программирования С# подходящей будет СУБД Microsoft SQL Server.

### 3.4 Разработка физической модели данных

В выбранной СУБД создадим таблицы, которые были спроектированы ранее, создадим связи между ними, после чего получим физическую модель данных, рисунок 14.

Модель имеет таблицы:

- Consignors;
- Waybills;
- Employees;
- Consignees;
- DestinationStations;
- Cargos;
- DepartureStations.

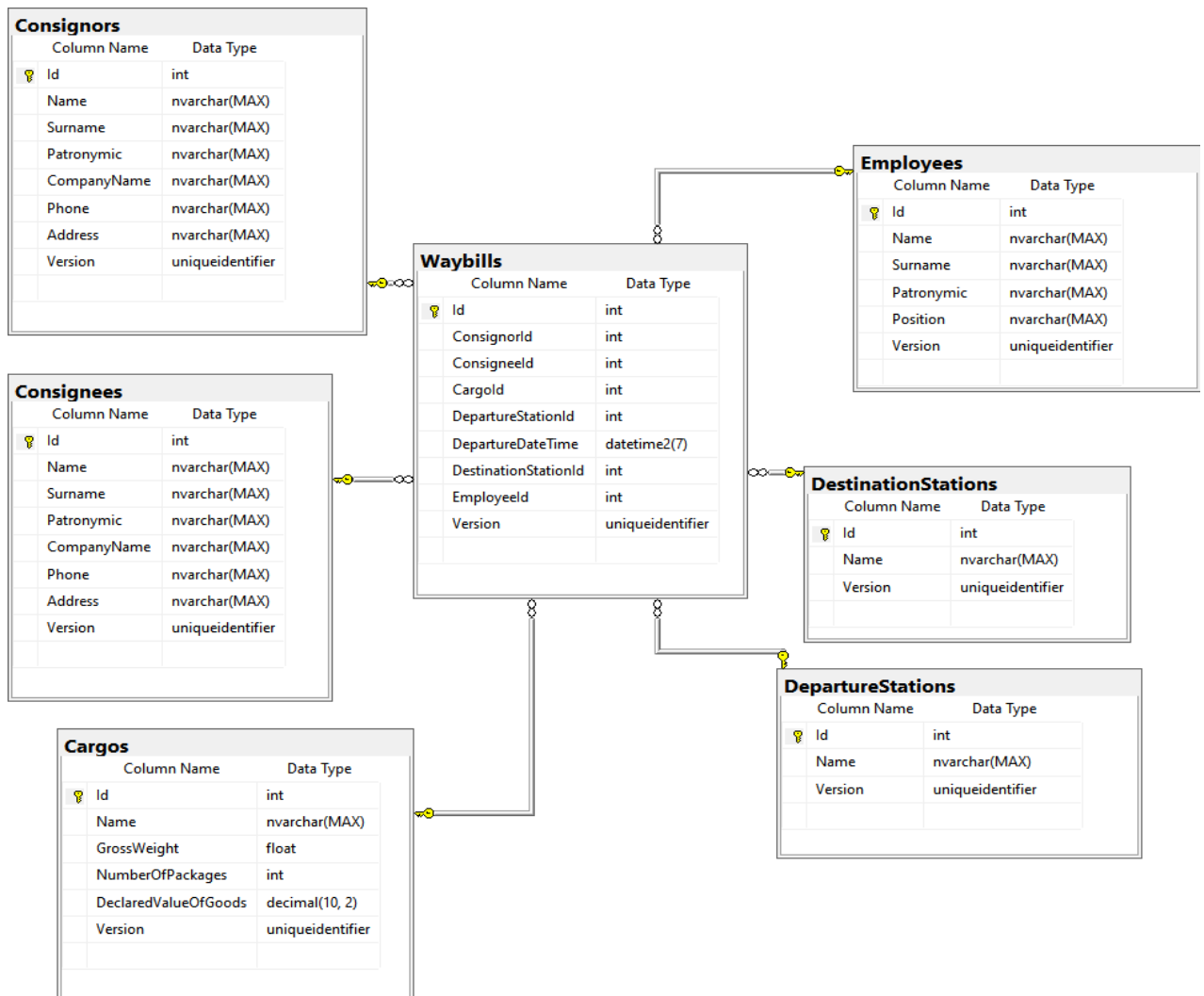


Рисунок 14 – Физическая модель данных

Физическая модель данных была построена с помощью инструментов СУБД Microsoft SQL Server. В модели представлены таблицы на латинском языке, так же, как и поля. Из рисунка 14 видно, что товарно-транспортная накладная имеет данные всех других созданных таблиц, кроме таблицы MigrationsHistory, которая содержит информацию об истории миграций данных. Все новые данные в базу данных будет вносить администратор приложения, который будет отвечать за ведение и сопровождение приложения.

### **3.5 Разработка приложения**

Приложение разработано в среде Visual Studio 2022 [21]. Visual Studio Code — это редактор исходного кода.

Редактор кода позволяет:

- работать с IntelliSense;
- выполнять отладку;
- удобно писать код;
- контролировать версии кода.

Также для VS Code есть расширения, которые позволяют подстраивать его под конкретные задачи и запускать даже на не очень мощных компьютерах.

Создание и интеграция данных в базу данных было осуществлено с помощью Entity Framework. Entity Framework — это решение для работы с базами данных, которое используется в программировании на языках семейства .NET. Оно позволяет взаимодействовать с СУБД с помощью сущностей (entity), а не таблиц. Сущности были созданы непосредственно в приложении, которые автоматически добавляют новые данные в базу данных [22].

При внесении новых данных администратором приложения, они автоматически сохраняются в базу данных, откуда их можно получить в любое время по средствам запросов. Менеджер использует данные при создании сопроводительных документов на грузоперевозки.

#### **3.5.1 Описание модулей приложения**

Структура приложения автоматизации создания и передачи сопроводительных документов на железнодорожные грузоперевозки имеет структуру, представленную на рисунке 15.



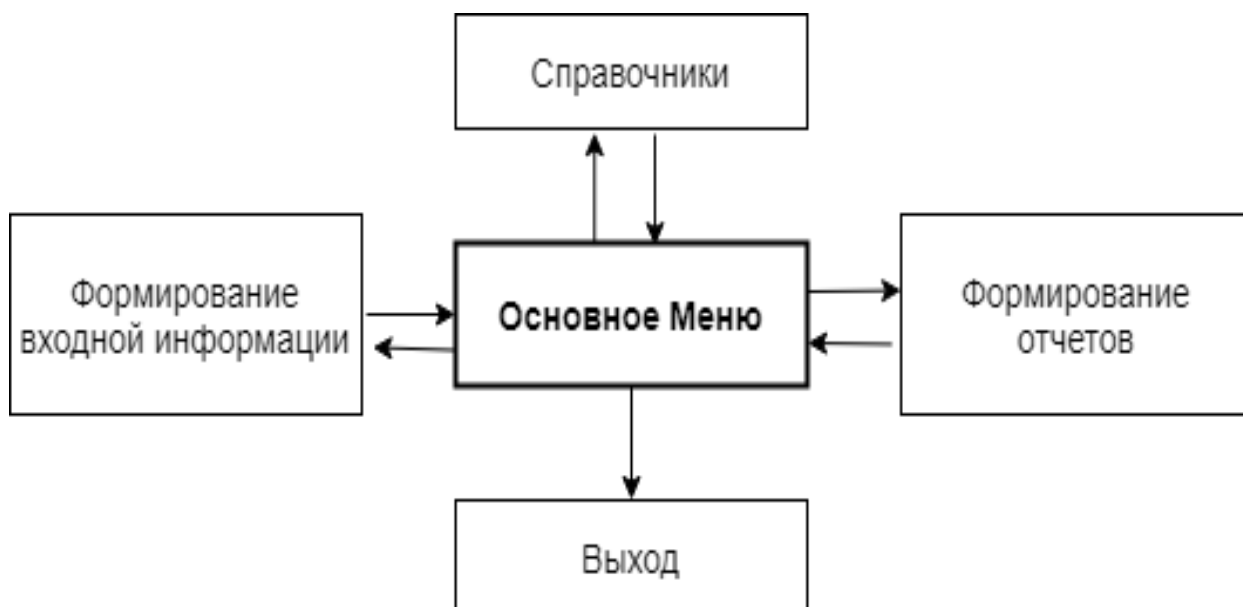


Рисунок 15 - Структура приложения

Работа менеджера с приложением начинается с вывода основного меню, где представлен модуль для создания и печати товарно-транспортной накладной на груз.

Модуль работы со справочниками обеспечивает получение и просмотр данных, которые хранятся в справочниках базы данных.

Модуль отчетов предоставляет возможность сформировать документ в удобный формат pdf и отправить его на печать [23].

### 3.6 Описание функциональности приложения

При открытии пользователем приложения загружается основное меню с формой для создания сопроводительных документов на грузоперевозки. Товарно-транспортная накладная создается из выпадающих списков, на рисунке 16 представлена форма.

Печать товарно-транспортной накладной

Грузоотправитель	▼
Грузополучатель	▼
Станция отправления	▼
Станция назначения	▼
Сотрудник	▼
Груз	▼

Распечатать ТТН

Рисунок 16 – Основное меню

Менеджер выбирает данные всех полей формы таких как: грузоотправитель, грузополучатель, станция отправления, станция назначения, сотрудник и данные о грузе.

На рисунке 17 показано как менеджер будет выбирать данные из выпадающих списков.

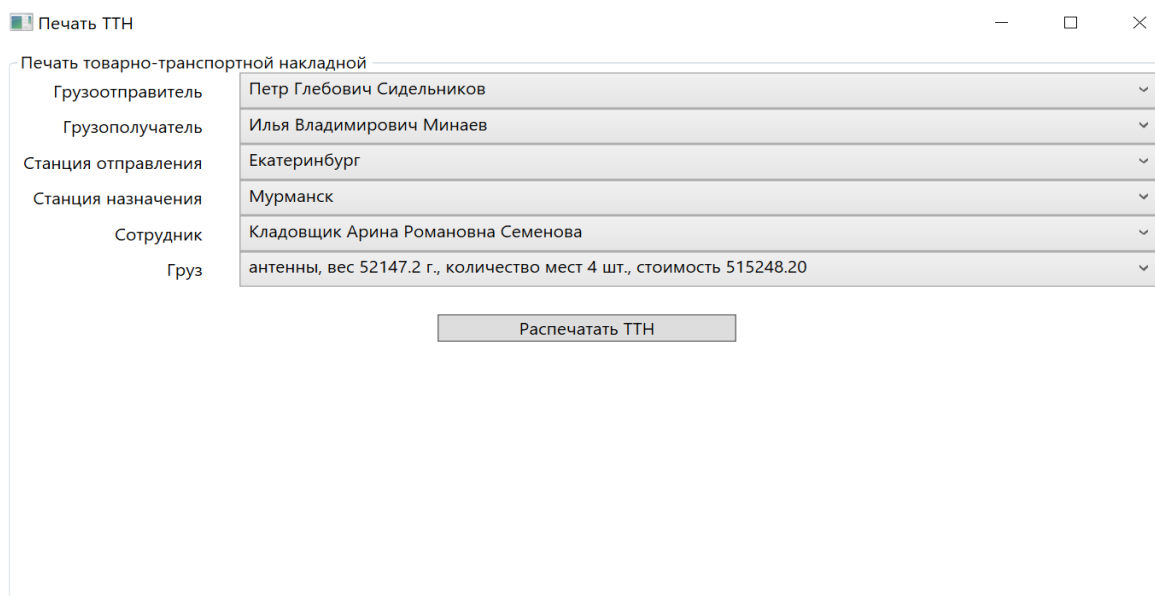
Печать ТТН

Печать товарно-транспортной накладной

Грузоотправитель	Петр Глебович Сидельников
Грузополучатель	Илья Владимирович Минаев
Станция отправления	Екатеринбург
Станция назначения	Мурманск
Сотрудник	▼
Груз	Кладовщик Михаил Петрович Никишин Старший кладовщик Оксана Григорьевна Шевелева Начальник склада Елена Филипповна Булкина Кладовщик Инна Максимовна Савинова Кладовщик Арина Романовна Семенова

Рисунок 17 – Заполнение данных

После выбора всех данных менеджер может распечатать ТТН, нажав кнопку, рисунок 18.



Печать товарно-транспортной накладной	
Грузоотправитель	Петр Глебович Сидельников
Грузополучатель	Илья Владимирович Минаев
Станция отправления	Екатеринбург
Станция назначения	Мурманск
Сотрудник	Кладовщик Арина Романовна Семенова
Груз	антенны, вес 52147.2 г., количество мест 4 шт., стоимость 515248.20

Распечатать ТТН

Рисунок 18 – Заполнение формы ТТН

После того как будет нажата кнопка «Распечатать ТТН», будет автоматически создан документ в pdf формате и сохранен в архив приложения, на рисунке 19 показано как происходит сохранение.

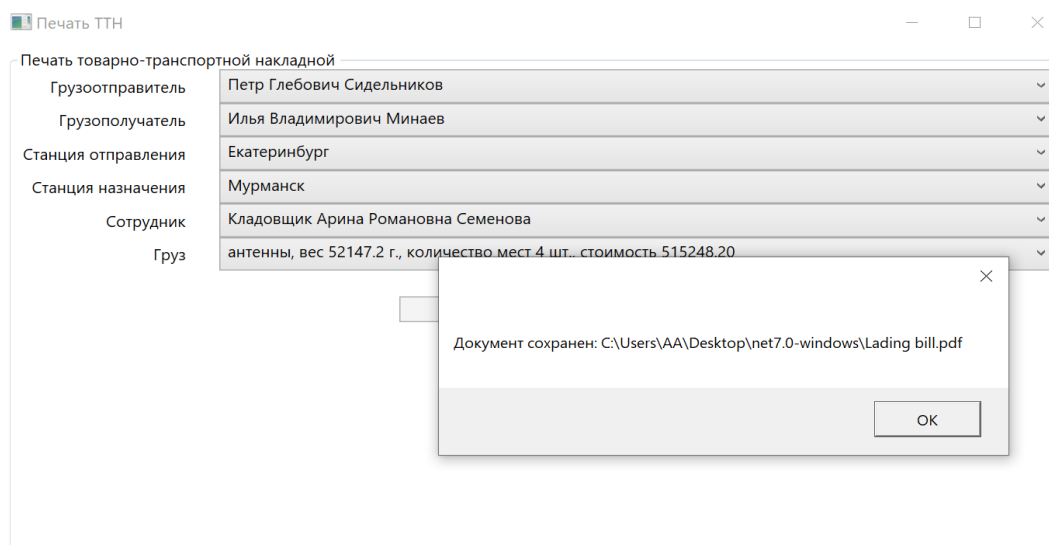


Рисунок 19 – Сохранение документа

Созданный документ можно открыть в любое время, на рисунке 20 показана товарно-транспортная накладная в формате pdf.

**ТОВАРНО-ТРАНСПОРТНАЯ НАКЛАДНАЯ**

Грузоотправитель: Петр Глебович Сидельников, г. Санкт-Петербург, ул. Невский проспект 3, +7 (333) 111-22-11  
ФИО, адрес, номер телефона

Грузополучатель: Илья Владимирович Минаев, г. Екатеринбург, ул. Иванова 45, +7 (456) 152-00-22  
ФИО, адрес, номер телефона

Станция отправления: Екатеринбург  
адрес станции отправления

Станция назначения: Мурманск  
адрес станции назначения

**1. ТОВАРНЫЙ РАЗДЕЛ (заполняется грузоотправителем)**

Код продукции (номенклатурный номер)	Вес брутто, гр.	Количество упаковок, шт.	Объявленная стоимость товаров, руб.	Наименование груза
1	2	3	4	5
3	52147.2	4	515248.20	автены

Отпуск груза произвел: Кладовщик Семенова Арина Романовна должность, ФИО      Подпись: \_\_\_\_\_      Груз получил: \_\_\_\_\_ должность, ФИО      Подпись: \_\_\_\_\_

Кладовщик Семенова Арина Романовна

Рисунок 20 – Созданная товарно-транспортная накладная

Созданный документ можно передавать внутри компании в электронном виде или распечатывать и отправлять вместе с грузом.

### 3.7 Обоснование экономической эффективности разработки приложения

Расчет затрат на разработку и внедрение приложения определяется путем считывания затрат на расходы написания приложения, описание и отладку, а также на внедрение по формуле (1).

$$C = C_{\text{ктс}} + C_{\text{алг}} + C_{\text{отл}} + C_{\text{вн}} \quad (1)$$

где  $C_{\text{ктс}}$  – затраты на приобретение комплекса технических средств;  
 $C_{\text{алг}}$  – расходы на написание алгоритма;

$C_{отл}$  – расходы на описание и отладку;

$C_{вн}$  – расходы на внедрение.

Затраты на приобретение комплекса технических средств составили 144000 рублей. Расходы на написание алгоритма составили 91980 рублей. Расходы на внедрение составили 324600 рублей. Расходы на отладку составили 712200 рублей. Так расходы составили  $C = 144000 + 91980 + 324600 + 712200 = 1\,272\,780$  рублей.

Рассчитаем расходы на написание алгоритма по формуле (2).

$$C_{алг} = Z_{празр} * B + От \quad (2)$$

где  $Z_{празр}$  – оклад разработчика;

$B$  – время, затраченное на разработку;

$От$  – отчисления на соц. страхование.

Определим параметры для вычислений. Оклад разработчика составляет 100000 рублей, время, затраченное на разработку, будет равно 15 дней, отчисления на социальное страхование составят 21000 рублей. Так расходы на написание алгоритма составят  $100000/22 * 15 + 21000 = 81980$  рублей

Расчет расходов на комплекс технических средств вычислим по формуле (3).

$$C_{ктс} = C_{комп} + C_{прин} + C_{др} \quad (3)$$

где  $C_{комп.}$  - стоимость компьютера;

$C_{прин.}$  - стоимость принтера;

$C_{др.}$  - стоимость других технических средств.

Стоимость компьютера равна 100000 рублей, стоимость принтера равна 24000 рублей, стоимость других технических средств равна 20000 рублей. Расходы на комплекс технических средств составляют  $100000 + 24000 + 20000 = 144000$  рублей.

Расчет затрат, связанных с внедрением программы рассчитаем по формуле (4).

$$C_{\text{вн}} = K_{\text{вн}} + Z_{\text{п.вн}} \quad (4)$$

где  $K_{\text{вн}}$  - стоимость машинного времени на время внедрения;

$Z_{\text{п.вн.}}$  - зарплата разработчика на время внедрения.

Стоимость машинного времени на время внедрения составила 211200 рублей, зарплата разработчика на время внедрения составила 113400 рублей, так затраты, связанные с внедрением программы равны  $211200 + 113400 = 324600$  рублей.

Стоимость машинного времени на время внедрения определяется по формуле (5).

$$K_{\text{вн}} = K * d * q \quad (5)$$

где  $K$  - время работы на ПК в день (час.);

$d$  – число дней работы на ПК;

$q$  - стоимость часа машинного времени.

Время работы на ПК в день равно 8 часов, число дней работы на ПК равно 22 дня, стоимость часа машинного времени равно 1200 рублей. Таким образом стоимость машинного времени на время внедрения равно  $8*22*1200 = 211200$  рублей

Заработная плата определяется по формуле (6).

$$Z_{\text{п.вн}} = Q_{\text{вн}} * \left(\frac{d}{D}\right) + \text{отч} \quad (6)$$

где  $Q_{\text{вн.}}$  - оклад разработчика;

$d$  – число дней работы на ПК;

$D$  – число рабочих дней в месяц;

отч. – отчисления на соц. страхование (26% от оклада разработчика системы).

Оклад разработчика равен 90000 рублей, число дней работы на ПК равно 22 дням, число рабочих дней в месяц равно 22 дня, отчисления на социальное страхование равно 23400 рублей. Так заработная плата будет равна  $90000 \cdot (22/22) + 23400 = 113400$  рублей.

Расчет расходов на написание и отладку программы будем производить по формуле (7).

$$C_{\text{отл}} = K_{\text{отл}} + K_{\text{зп}} \quad (7)$$

где  $K_{\text{отл}}$  – стоимость машинного времени, затраченного на отладку системы;

$K_{\text{з.п.}}$  – фонд заработной платы программиста, занятого отладкой и написанием программы.

Стоимость машинного времени, затраченного на отладку системы равна 211200 рублей, фонд заработной платы программиста, занятого отладкой и написанием программы равен 501100 рублей, таким образом расходы на написание и отладку программы равны  $211200 + 501100 = 712300$  рублей.

Фонд заработной платы программиста, занятого отладкой и написанием программы, определяется по формуле (8).

$$K_{\text{зп}} = K_{\text{м}} * O_{\text{разраб}} + \text{отч.} \quad (8)$$

где  $K_{\text{м}}$  – количество дней, потраченных на разработку;

$O_{\text{разраб.}}$  – оклад программиста за день;

отч. – отчисления на соц. страхование (21% от оклада программиста, занятого отладкой программы).

Количество дней, потраченных на разработку, будет равно 100, оклад

программиста за день составляет 5000 рублей, отчисления на социальное страхование равно 1050 рублей, получается фонд заработной платы программиста, занятого отладкой и написанием программы рассчитанного на 100 дней равен  $100 * 5000 + 1050 = 501050$  рублей.

Расчет эксплуатационных расходов, связанных с использованием программы основывается на времени использования ЭВМ под задачей и стоимостью машинного часа и рассчитывается по формуле (9).

$$E_{\text{экспл}} = T_{\text{мв}} * S_{\text{м}} \quad (9)$$

где  $T_{\text{мв}}$  – время использования ЭВМ под задачей;

$S_{\text{м}}$  – стоимость машинного часа.

Время использования ЭВМ под задачей составляет 1440 часов, стоимость машинного часа равна 200 рублей/час. Таким образом эксплуатационные расходы, связанные с использованием программы равны  $1440 * 200 = 288000$  рублей.

Годовой прирост прибыли от автоматизации определяется нормами эксплуатационных расходов, связанных с использованием программы, ежемесячным начислением на зарплату и основной зарплатой работников, решающих задачу вручную, коэффициентом, учитывающим дополнительную зарплату, и рассчитывается по формуле (10).

$$dE_{\text{экспл}} = (1 + Wo) * (1 + Wq) * B_{\text{з.пр}} - E_{\text{экспл}} \quad (10)$$

где  $E_{\text{экспл}}$  – эксплуатационные расходы, связанные с использованием программы;

$Wo$  – ежемесячные начисления на зарплату;

$Wq$  – коэффициент, учитывающий дополнительную зарплату;

$B_{\text{з.пр}}$  – основная зарплата работников, решающих задачу вручную.

Ежемесячные начисления на зарплату составили 0,35, коэффициент,



учитывающий дополнительную зарплату равен 0,15, основная зарплата работников составляет  $4 * 12 * 30000 = 1\,440\,000$  рублей, эксплуатационные расходы составили 288 000 рублей. Таким образом, годовой прирост прибыли составил  $(1+0,35) * (1+0,15) * 1\,440\,000 - 288\,000 = 1\,948\,000$  рублей.

Стоимость пакета программ определяется расчетом затрат на разработку программы, распределенной между числом пользователей, и рассчитывается по формуле (11).

$$S = \frac{C}{N} \quad (11)$$

где  $C$  – затраты на разработку программы;

$N$  – число пользователей.

Затраты на разработку программы составили 1 270 000 рублей на два пользователя. Так стоимость пакета программ составила 635 000 рублей.

Срок окупаемости определяется отношением капитальных вложений к экономии годовых эксплуатационных расходов и рассчитывается по формуле (12).

$$O = \frac{C}{dE_{\text{экспл}}} \quad (12)$$

где  $C$  – затраты на разработку;

$dE_{\text{экспл}}$  - экономия годовых эксплуатационных расходов.

Затраты на разработку составили 1 270 000 рублей, экономия годовых эксплуатационных расходов равна 1 948 000 рублей, таким образом срок окупаемости равен  $1\,270\,000 / 1\,948\,000 = 0,652$  лет (7 месяцев)

В процессе расчета была подсчитана экономическая эффективность от внедрения приложения.

Были рассчитаны затраты на разработку программного обеспечения. Произведен расчет капитальных вложений, требующихся для запуска

программного продукта с учетом стоимости и времени использования электронно-вычислительной машины для работы с конкретной задачей. Рассчитана экономия средств при решении специализированных задач на электронно-вычислительной машине.

На рисунке 24 представлен график окупаемости приложения.

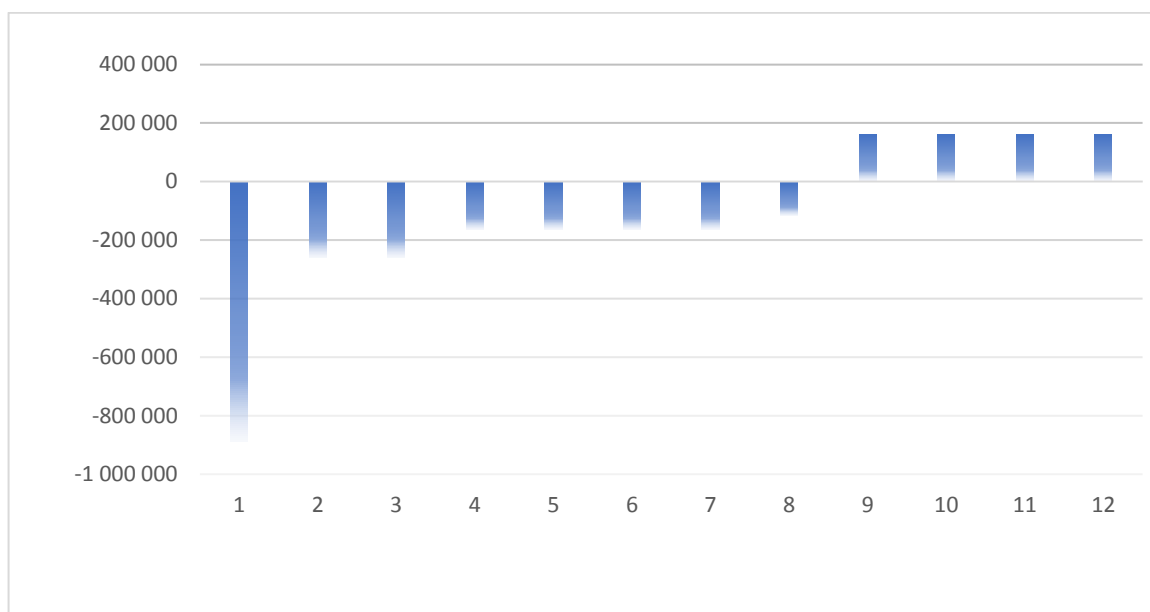


Рисунок 24 – График окупаемости приложения

Было рассчитано, что срок окупаемости внедрения приложения составляет 8 месяцев.

После всех расчетов определено, что внедрение данного приложения целесообразно и экономически выгодно.

#### Выводы по третьей главе

В третьей главе выполнены практические расчеты экономической эффективности разработки и внедрения приложения.

## Заключение

В ходе выполнения выпускной квалификационной работы описана разработка приложения для автоматизации создания сопроводительных документов на груз.

Проведен обзор существующих исследований и теорий, связанных с темой исследования. Это помогло определить области которые еще не были исследованы или требовали дополнительного изучения

Проведено исследование деятельности компании ООО «Информсистемы», выявлены бизнес-процессы, которые требуют автоматизации.

По результатам исследования было выяснено, что в автоматизации процессов больше всего нуждается процесс создания и передачи сопроводительных документов на железнодорожные грузоперевозки.

На основе проведенного анализа бизнес-процессов было проведено проектирование приложения, в котором было проведено разработка и моделирование разных уровней модели данных. Проектирование было выполнено в нотациях IDEF0 и UML.

На основе моделирования были представлены требования к аппаратно-программному обеспечению приложения. Для реализации приложения были выбраны инструменты разработки язык программирования С# и СУБД Microsoft SQL Server. Выбранные инструменты позволяли дальнейшее масштабирование архитектуры приложения на основе потребностей компании грузоперевозок.

Были проведены расчеты экономической эффективности разработки и внедрения приложения, на основе которых было выяснено, что разработанное приложение увеличит эффективность работы менеджеров компании и приведет к экономии средств.

Задачи ВКР выполнены в полном объеме, цель достигнута.

## Список используемой литературы и используемых источников

1. Волк В. К. Базы данных. Проектирование, программирование, управление и администрирование: учебник /В. К. Волк. — Санкт-Петербург: Лань, 2020. — 244 с.: ил.— (Учебники для вузов. Специальная литература). — Текст: непосредственный.
2. Какие данные используют операторы для контроля за перемещением грузов по железной дороге. [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/companies/pgk/articles/688416/> (дата обращения: 09.10.2023).
3. Как использовать Entity Framework. [Электронный ресурс]. – Режим доступа: [https://skillbox.ru/media/code/entity\\_framework/](https://skillbox.ru/media/code/entity_framework/) (дата обращения: 09.10.2023)
4. Концептуальное логическое и физическое моделирование данных. [Электронный ресурс]. – Режим доступа: <https://framework.ru/SID/datamodelling.html> (дата обращения: 09.10.2023).
5. Моделирование систем с использованием информационных технологий: учебн. пособие / В. Г. Лисиенко, Н. Г. Дружинина, О. Г. Трофимова, С. П. Трофимов. – Екатеринбург: УГТУ-УПИ, 2009. – 440 с.
6. Онлайн курс обучения программированию: методологии разработки. [Электронный ресурс]. – Режим доступа: <https://javarush.ru/groups/posts/647-metodologii-razrabotki-po> (дата обращения: 09.10.2023)
7. Отличия, достоинства и недостатки базы данных PostgreSQL: что такое PostgreSQL. [Электронный ресурс]. – Режим доступа: <https://oracle-patches.com/common/3214-что-такое-postgresql> (дата обращения: 09.10.2023)
8. Отправляем груз по железной дороге: документация. [Электронный ресурс]. – Режим доступа: <https://www.glavbukh.ru/art/30674-otpravlyaem-gruz-po-jeleznoy-doroge-dokumentatsiya> (дата обращения: 09.10.2023).

9. Попова-Коварцева, Д. А. Основы проектирования баз данных: учеб. пособие / Д.А. Попова-Коварцева, Е.В. Сопченко. – Самара: Изд-во Самарского университета, 2019. – 112 с.: ил. ISBN 978-5-7883-1450-1
10. Правил перевозок грузов железнодорожным транспортом. [Электронный ресурс]. – Режим доступа: [https://base.spinform.ru/show\\_doc.fwx?rgn=119287](https://base.spinform.ru/show_doc.fwx?rgn=119287) (дата обращения: 09.10.2023).
11. С GUI на C#: Кроссплатформенное приложение [Электронный ресурс]. - Режим доступа: [https://skillbox.ru/media/code/ne\\_windows\\_](https://skillbox.ru/media/code/ne_windows_) (дата обращения: 09.10.2023)
12. Тамре Л. Введение в тестирование программного обеспечения /пер. с англ. М.: Вильямс, 2018. 368 с.
13. Технология приема и выдачи грузов на железнодорожном транспорте. [Электронный ресурс]. – Режим доступа: [https://www.usurt.ru/uploads/data/index2/files/7\\_22/014\\_2\\_7\\_22.pdf](https://www.usurt.ru/uploads/data/index2/files/7_22/014_2_7_22.pdf) (дата обращения: 09.10.2023).
14. Этапы и методологии проектирования баз данных. [Электронный ресурс]. – Режим доступа: <https://studfile.net/preview/2674691/page:4/> (дата обращения: 09.10.2023).
15. Access: База данных. [Электронный ресурс]. – Режим доступа <https://www.microsoft.com/ru-ru/microsoft-365/access>(дата обращения: 09.10.2023)
16. GeekBrains – Язык программирования C#. [Электронный ресурс]. - Режим доступа: <https://geekbrains.ru/posts/yazyk-programmirovaniya-c-sharp-istoriya-specifika-mesto-na-rynke> (дата обращения: 09.10.2023)
17. Helpiks.org: Достоинства и недостатки языка [Электронный ресурс]. - Режим доступа: <https://helpiks.org/6-21879.html> (дата обращения: 09.10.2023)
18. Java-энциклопедия языков программирования: Java. [Электронный ресурс]. – Режим доступа: <http://progopedia.ru/language/java/>

(дата обращения: 09.10.2023)

19. SoftClipper: что такое SQL Server. [Электронный ресурс]. – Режим доступа: <https://softclipper.net/foxpro-i-sql/sravnenie-baz-dannykh-microsoft-sql-server-i-microsoft-visual-foxpro.html> (дата обращения: 09.10.2023)

20. SQL Server: руководство по SQL Server. [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/ru-ru/sql/sql-server/tutorials-for-sql-server-2016?view=sql-server-ver15> (дата обращения: 09.10.2023)

21. SQL Server: программное обеспечение. [Электронный ресурс]. – Режим доступа: <https://www.microsoft.com/ru-ru/sql-server/sql-server-downloads> (дата обращения: 09.10.2023)

22. Visual Studio 2019: программное обеспечение. [Электронный ресурс]. – Режим доступа: <https://visualstudio.microsoft.com/ru/vs/> (дата обращения: 09.10.2023)

23. Visual Studio Code. [Электронный ресурс]. – Режим доступа: <https://practicum.yandex.ru/blog/vsyo-o-visual-studio-code/> (дата обращения: 09.10.2023)