МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное

учреждение высшего образования  
«Ивановский государственный энергетический университет

имени В.И. Ленина»

Кафедра Программного обеспечения компьютерных систем

**Система «Система продажи билетов на междугородние автобусные рейсы»**

Пояснительная записка к курсовому проекту

по дисциплинам: «Проектирование и архитектура ПС», «Конструирование ПО»

на 62 листах

**Выполнил студент гр. 3-42 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Федоров К.А.**

**Проверили**

**к.т.н., доцент каф. ПОКС \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Игнатьев Е. Б.**

**ст. пр. ПОКС \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Мочалов А.С.**

**ст. пр. ПОКС \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Фомина О.В.**

**Иваново, 2024**

**Аннотация**

пояснительной записки к курсовому проекту «Система продажи билетов на междугородние автобусные рейсы»

Исполнитель: Федоров К.А.

Руководители: Игнатьев Е. Б., Мочалов А.С., Фомина О.В.

Документ содержит: 62 листов, 42 рисунков.

Ключевые слова: СИСТЕМА ПРОДАЖИ БИЛЕТОВ, БАЗА ДАННЫХ, БИЛЕТ, АВТОВОКЗАЛ.

Учебный курсовой проект выполняется по дисциплинам «Проектирование и архитектура ПС», «Конструирование ПО» и посвящен разработке проекта системы по доставке пиццы. Пояснительная записка содержит следующие документы:

1. Задание.
2. Система продажи билетов на междугородние автобусные рейсы. Техническое задание.
3. Система продажи билетов на междугородние автобусные рейсы. Технорабочий проект.

Содержание

[Термины, определения и сокращения 6](#_Toc185799742)

[Задание 8](#_Toc185799743)

[Введение 10](#_Toc185799744)

[Система продажи билетов на междугородние автобусные рейсы. Техническое задание 11](#_Toc185799745)

[1 Общие сведения 11](#_Toc185799746)

[1.1 Наименование системы 11](#_Toc185799747)

[1.2 Заказчик и Разработчик системы 11](#_Toc185799748)

[1.3 Основание для разработки 11](#_Toc185799749)

[1.4 Плановые сроки начала и окончания работы по разработке проекта 11](#_Toc185799750)

[1.5 Сведения об источниках и порядке финансирования работ 11](#_Toc185799751)

[1.6 Порядок оформления и предъявления Заказчику результатов работ 11](#_Toc185799752)

[2 Назначение и цели создания системы 12](#_Toc185799753)

[2.1 Назначение системы 12](#_Toc185799754)

[2.2 Цели создания системы 12](#_Toc185799755)

[3 Характеристика объектов автоматизации 12](#_Toc185799756)

[3.1 Объекты автоматизации 12](#_Toc185799757)

[3.2 Концептуальная модель предметной области 13](#_Toc185799758)

[4 Требования к системе 14](#_Toc185799759)

[4.1 Требования к системе в целом 14](#_Toc185799760)

[4.1.1 Требования к структуре и функционированию системы 14](#_Toc185799761)

[1.1.1 Требования к численности и квалификации персонала системы 14](#_Toc185799762)

[1.1.2 Требования к надежности 15](#_Toc185799763)

[1.1.3 Требования безопасности 15](#_Toc185799764)

[1.1.4 Требования к эргономике и технической эстетике 15](#_Toc185799765)

[1.1.5 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы 16](#_Toc185799766)

[1.1.6 Требования к защите информации от несанкционированного доступа 16](#_Toc185799767)

[1.1.7 Требования по сохранности информации при авариях 16](#_Toc185799768)

[1.1.8 Требования к патентной чистоте 17](#_Toc185799769)

[1.2 Требования к функциям, выполняемым системой 17](#_Toc185799770)

[1.2.1 Модель вариантов использования 17](#_Toc185799771)

[1.2.2 Спецификация варианта использования «Купить билет» 20](#_Toc185799772)

[1.3 Требования к видам обеспечения 23](#_Toc185799773)

[1.3.1 Требования к информационному обеспечению 23](#_Toc185799774)

[1.3.2 Требования к лингвистическому обеспечению 23](#_Toc185799775)

[1.3.3 Требования к программному обеспечению 23](#_Toc185799776)

[1.3.4 Требования к техническому обеспечению 23](#_Toc185799777)

[1.3.5 Требования к организационному обеспечению 24](#_Toc185799778)

[2 Состав и содержание работ по созданию системы 25](#_Toc185799779)

[3 Порядок контроля и приемки системы 25](#_Toc185799780)

[4 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие 26](#_Toc185799781)

[5 Требования к документированию 26](#_Toc185799782)

[5.1 Требования к составу документов 26](#_Toc185799783)

[5.2 Требования к оформлению документов 27](#_Toc185799784)

[5.2.1 Технорабочий проект 27](#_Toc185799785)

[6 Источники разработки 27](#_Toc185799786)

[Система доставки пиццы. Технорабочий проект 29](#_Toc185799787)

[1 Общие положения 29](#_Toc185799788)

[1.1 Наименование системы 29](#_Toc185799789)

[1.2 Основание для разработки 29](#_Toc185799790)

[1.3 Перечень организаций, участвующих в разработке системы, сроки выполнения этапов 29](#_Toc185799791)

[1.4 Цели, назначение и области использования 29](#_Toc185799792)

[1.5 Подтверждение соответствия проектных решений действующим нормам и правилам техники безопасности, пожаро- и взрывобезопасности 30](#_Toc185799793)

[1.6 Нормативно-технические документы, использованные при проектировании 30](#_Toc185799794)

[2 Описание процесса деятельности 30](#_Toc185799795)

[3 Основные технические решения 31](#_Toc185799796)

[3.1 Архитектура системы 31](#_Toc185799797)

[3.2 Решения по режимам функционирования, диагностированию работы системы 32](#_Toc185799798)

[3.3 Решения по численности, квалификации и функциям персонала, режимам его работы, порядку взаимодействия 32](#_Toc185799799)

[3.4 Сведения об обеспечении заданных в ТЗ потребительских характеристик системы, определяющих ее качество 33](#_Toc185799800)

[3.4.1 Надёжность 33](#_Toc185799801)

[3.4.2 Требования к эргономике и технической эстетике 33](#_Toc185799802)

[3.4.3 Защищенность информации 33](#_Toc185799803)

[3.5 Состав функций, реализуемых системой 33](#_Toc185799804)

[3.6 Решения по комплексу технических средств 35](#_Toc185799805)

[3.7 Решения по информационному обеспечению 36](#_Toc185799806)

[3.8 Физическая модель БД 36](#_Toc185799807)

[3.8.1 Скрипт БД 37](#_Toc185799808)

[3.9 Решения по программному обеспечению 40](#_Toc185799809)

[3.9.1 Моделирование эскизной реализации варианта использования «купить билет» 40](#_Toc185799810)

[4 Руководство пользователя 44](#_Toc185799811)

[4.1 Вход в приложение 44](#_Toc185799812)

[4.2 Регистрация нового клиента 45](#_Toc185799813)

[4.3 Просмотр списка рейсов и их выбор клиентом 47](#_Toc185799814)

[4.4 Панель навигации для пассажира 49](#_Toc185799815)

[4.5 Изменение профиля пассажира 50](#_Toc185799816)

[4.6 Список купленных билетов 50](#_Toc185799817)

[4.7 Просмотр расписания 52](#_Toc185799818)

[4.8 Управление автобусами администратором 52](#_Toc185799819)

[4.9 Управление пользователями администратором 53](#_Toc185799820)

[4.10 Управление водителями администратором 55](#_Toc185799821)

[4.11 Управление расписанием администратором 57](#_Toc185799822)

[4.12 Управление маршрутами администратором 58](#_Toc185799823)

[4.13 Управление рейсами администратором 59](#_Toc185799824)

[1.1 Генерация отчетов менеджером 60](#_Toc185799825)

[5 Мероприятия по подготовке к вводу системы в действие 62](#_Toc185799826)

[5.1 Мероприятия по приведению информации к виду, пригодному для обработки на ЭВМ 62](#_Toc185799827)

[5.2 Мероприятия по обучению и проверке квалификации персонала 62](#_Toc185799828)

[6 Источники разработки 62](#_Toc185799829)

# Термины, определения и сокращения

В настоящем документе использованы термины и определения, предусмотренные ГОСТ 34.003-90 [1]. Кроме того, использован ряд терминов и определений, не предусмотренных указанным ГОСТ:

|  |  |
| --- | --- |
| Термин | Определение |
| Купленный билет | Билет, который был успешно оплачен и оформлен для конкретного рейса. |
| Просмотр доступных рейсов | Процесс выбора рейса пассажиром перед покупкой билета. |
| Профиль пассажира | Данные пользователя, включая личную информацию и сохранённые карты для оплаты. |
| Печать билета | Процесс создания физической или электронной копии купленного билета для предъявления при посадке. |
| Маршрут | Направление движения автобуса, включающее начальный и конечный пункты. |
| Фильтр рейсов | Процесс поиска рейсов по заданным параметрам: начальный пункт, конечный пункт и дата отправления. |
| Банковская карта | Платёжный инструмент, привязанный к профилю пользователя для оплаты билетов. |
| Ошибка при покупке билета | Ситуация, при которой покупка билета не может быть завершена из-за отсутствия карты или неверно введённых данных пассажира. |

Ниже приводятся сокращения, использованные в Документе:

| Сокращение | Определение |
| --- | --- |
| ИГЭУ | ФГБОУ ВПО «Ивановский государственный энергетический университет имени В.И. Ленина» |
| БД | База данных |
| НСД | Несанкционированный доступ |
| ОС | Операционная система |
| ПК | Персональный компьютер |
| ПО | Программное обеспечение |
| ПОКС | Кафедра программного обеспечения компьютерных систем ИГЭУ |
| СУБД | Система управления базами данных |
| ТЗ | Техническое задание |
| DOM | Domain Object Model (модель объектов предметной области) |
| GUI | Графический пользовательский интерфейс (Graphical User Interface) |
| MS | Корпорация Microsoft |
| UI | Пользовательский интерфейс (User Interface) |
| UML | Унифицированный язык моделирования (Unified Modeling Language) |

# Задание

по курсовому проекту студента гр. 3-42 Федорова Кирилла Андреевича

1. Тема: Система продажи билетов на междугородние автобусные рейсы
2. Срок сдачи студентом работы: 25 декабря 2024 г.
3. Исходные данные:

Необходимо разработать систему управления автобусными билетами для организации, занимающейся продажей и администрированием автобусных перевозок. Пользователями системы являются пассажиры и администраторы.

Пассажиры могут выбирать доступные рейсы, указывая маршрут, дату и время отправления, а также покупать билеты для себя или для другого пассажира, вводя необходимые данные: имя, фамилию, отчество и паспортные данные. После покупки пассажиры могут просматривать список своих билетов, содержащий информацию о маршруте, дате и времени отправления, номере автобуса, водителе и стоимости билета. Пассажиры также могут редактировать свой профиль, изменяя имя, фамилию, отчество, номер телефона, email, паспортные данные, логин и пароль. Система позволяет добавлять и редактировать банковские карты для оплаты билетов.

**Администраторы** имеют широкий функционал для управления данными системы. Они могут добавлять, редактировать и удалять автобусы, указывая модель, номерной знак и вместимость. Администраторы могут управлять пользователями, добавляя новых, редактируя существующих и удаляя их. Также они могут управлять водителями, вводя их личные данные, включая имя, фамилию, отчество, номер лицензии и телефон.

Система позволяет администраторам создавать и управлять маршрутами, задавая начальный и конечный пункты и дистанцию. На основе маршрутов администраторы могут создавать расписания, указывая автобусы, водителей, время начала и окончания поездок, а также интервалы движения.

Администраторы могут автоматически создавать рейсы на весь день по выбранному расписанию, задавая цену билетов и рассчитывая время отправления и прибытия. Все созданные рейсы можно просматривать и удалять при необходимости.

Система также включает функционал для формирования отчетов. Администраторы могут создавать отчеты по продажам билетов за определенный период, анализировать популярность маршрутов, просматривать выполненные рейсы и генерировать отчеты по купленным билетам за месяц. Все отчеты можно распечатать для дальнейшего использования. Система предоставляет удобный интерфейс для всех пользователей и гарантирует актуальное обновление данных в режиме реального времени, обеспечивая автоматизацию процесса покупки билетов и управления автобусными перевозками.

1. Дата выдачи задания: 23 сентября 2024 г.

Задание выдали:

доц. каф. ПОКС \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Игнатьев Е.Б.

ст. пр. ПОКС \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Мочалов А.С.

ст. пр. ПОКС \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Фомина О.В.

Задание принял к исполнению:

студент гр. 3-42 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Федоров К.А.

# Введение

Ежедневно может осуществляться до нескольких десятков или сотен продаж автобусных билетов, причем предполагается, что пассажиры могут приобретать билеты не только для себя, но и для других лиц, указывая их данные.

Перед руководством организации встали задачи сокращения времени на выбор и оформление билетов пассажирами, а также автоматизации процесса управления автобусными рейсами. Кроме того, необходимо вести учет всех продаж билетов в базе данных организации для анализа работы системы и автоматической генерации отчетов по поездкам и продажам.

Поэтому создание системы, автоматизирующей процесс покупки билетов и управления рейсами, является актуальной задачей.

Система основана на приложении WPF, которое обеспечивает доступ к базе данных для пассажиров и администраторов. Пассажиры могут регистрироваться и входить в систему для выбора подходящих рейсов. Приложение извлекает из базы данных информацию о доступных маршрутах и расписаниях. Пассажиры могут выбрать рейс, указать детали поездки (откуда, куда, дата и время отправления), а также оформить покупку билета для себя или другого пассажира, заполнив необходимые данные. После успешной покупки билеты сохраняются и становятся доступными для просмотра во вкладке «Мои билеты».

Система позволяет входить не только пассажирам, но и администраторам. Администраторы могут управлять всей системой: добавлять, редактировать и удалять автобусы, водителей, маршруты и расписания. Администраторы также могут создавать рейсы на весь день по расписанию, устанавливая цену билетов и автоматически формируя рейсы на основе заданного интервала движения.

Кроме того, система предоставляет возможность для генерации отчетов. Администраторы могут формировать отчеты по продажам билетов за определенный период, анализировать популярность маршрутов и количество проданных билетов, а также генерировать отчеты по купленным билетам за месяц. Отчеты можно распечатать для дальнейшего использования.

Таким образом, система автоматизирует процесс покупки билетов, управления рейсами и формирования отчетов, обеспечивая удобный и эффективный интерфейс для пассажиров и администраторов.

# Система продажи билетов на междугородние автобусные рейсы. Техническое задание

# Общие сведения

## Наименование системы

Полное наименование системы — «Система продажи билетов на междугородние автобусные рейсы».

Условное обозначение системы — «BTS» (Bus Ticketing Systemров).

## Заказчик и Разработчик системы

Заказчик системы: Федеральное государственное бюджетное образовательное учреждение высшего образования «Ивановский государственный энергетический университет имени В.И. Ленина» (ИГЭУ); 153003, г. Иваново, ул. Рабфаковская, д. 34.

Разработчик системы: Федоров Кирилл Андреевич, студент группы 3-42.

## Основание для разработки

Разработка ведется на основании задания на курсовой проект по дисциплинам: «Проектирование и архитектура ПС», «Конструирование ПО».

Задание утверждено на заседании кафедры ПОКС 01.09.2024 и выдано преподавателями кафедры Игнатьевым Е.Б., Мочаловым А.С., Фоминой О.В.

## Плановые сроки начала и окончания работы по разработке проекта

Начало: 23 сентября 2024 г.

Окончание: 25 декабря 2024 г.

## Сведения об источниках и порядке финансирования работ

Финансирование работ отсутствует.

## Порядок оформления и предъявления Заказчику результатов работ

Разработчик оформляет результаты работ над проектом в виде технорабочего проекта и передает их Заказчику.

# Назначение и цели создания системы

## Назначение системы

BTS-система предназначена для автоматизации процесса продажи билетов на междугородние автобусные рейсы, управления расписаниями и рейсами, а также формирования отчетов по продажам билетов за определенный период.

## Цели создания системы

Основными целями создания системы являются:

- упрощение процесса покупки билетов пассажирами;

- предоставление возможности покупки билетов для себя или для другого пассажира;

- повышение управляемости за счет возможности проведения анализа данных о продажах билетов и формирования отчетов;

- автоматизация управления автобусами, маршрутами, расписаниями и водителями;

- обеспечение контроля за продажами билетов и рейсами в реальном времени.

# Характеристика объектов автоматизации

## Объекты автоматизации

Объектами автоматизации являются процессы оформления билетов пассажирами, управления рейсами администраторами, а также процесс создания отчетов по поездкам за конкретный период.

Пассажиры просматривают доступные междугородние автобусные рейсы и приобретают билеты. После выбора рейса и подтверждения заказа билет сохраняется в системе. Если билет приобретается для другого пассажира, указываются его личные данные.

Администраторы управляют расписаниями, автобусами, водителями и пользователями системы. Они добавляют новые рейсы, редактируют информацию о расписаниях, автобусах и маршрутах, а также создают отчеты по продажам билетов и выполненным рейсам за определенный период.

Каждый билет содержит информацию о рейсе, включая номер рейса, дату и время отправления, пункт отправления и прибытия, стоимость билета, данные пассажира (фамилия, имя, отчество), а также дату и время покупки.

Система позволяет в режиме реального времени отслеживать наличие свободных мест на рейсы и изменять статус рейсов. Впоследствии администраторы могут формировать отчеты по проданным билетам и выполненным поездкам для анализа работы транспортной компании

## Концептуальная модель предметной области

В результате обследования предметной области была разработана модель объектов предметной области (Domain Object Model – DOM), описывающая классы предметной области и связи между ними.

Рис. 1 представляет диаграмму классов для концептуальных классов.

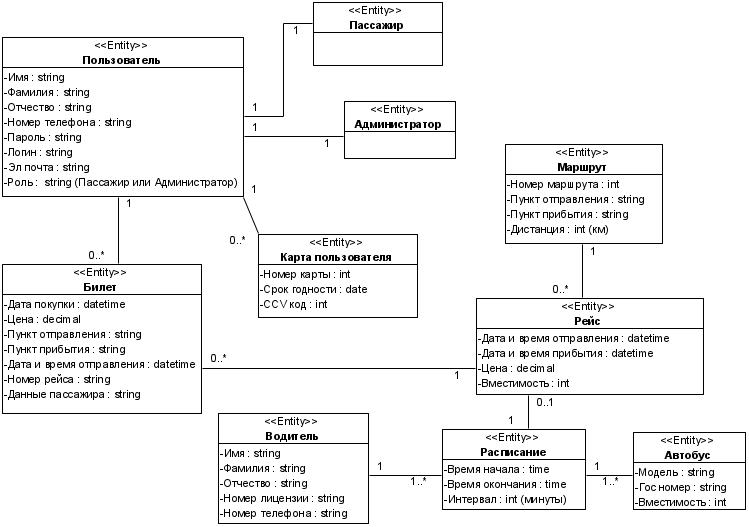


Рисунок 1 – Концептуальная модель предметной области

# Требования к системе

## Требования к системе в целом

### Требования к структуре и функционированию системы

**Архитектура системы**

Система BTS должна иметь двухуровневую клиент-серверную архитектуру:

На уровне хранения данных размещается сервер БД, а на прикладном уровне – насыщенное клиентское приложение.

**Информационный обмен между компонентами системы**

Входящие в состав BTS подсистемы в процессе функционирования должны обмениваться информацией на основе открытых форматов обмена данными по протоколам на основе TCP/IP.

**Перспективы развития, модернизации системы**

Система должна поддерживать возможность дальнейшей модернизации как программного обеспечения, так и комплекса технических средств. Необходимо предусмотреть:

Возможность масштабирования системы для увеличения производительности и поддержки большего количества пользователей.

Расширение функционала для администраторов, включая возможность автоматического формирования дополнительных отчетов по продажам и управлению рейсами.

Возможность интеграции дополнительных функций для пассажиров, таких как уведомления о статусе рейсов и билетов.

### Требования к численности и квалификации персонала системы

Для эксплуатации BTS должны быть предусмотрены следующие роли пользователей:

1. Администратор;
2. Пассажир;

Администратор должен обладать высоким уровнем квалификации и опытом работы с базами данных, а также навыками управления данными и формирования отчетов.

Пользователи системы (Администратор и пассажир) должны иметь опыт работы с персональным компьютером на базе операционных систем Microsoft Windows на уровне уверенного пользователя и свободно осуществлять базовые операции в стандартных Windows-приложениях.

Рекомендуемая численность для эксплуатации системы PD:

- Администратор – 1 штатная единица;

- Пассажир – число клиентов не определяется организацией и не должно быть ограничено.

### Требования к надежности

Система должна быть доступна 24 часа каждый день недели. Не должно быть никакого связанного с БД времени простоя. О любых запланированных простоях почтового сервера из-за профилактик BTS-пользователи должны быть уведомлены, по крайней мере, за 24 часа.

Отказ компонентов программного обеспечения не должен ставить под угрозу корректность и целостность БД. Пользователь должен иметь возможность повторно начать программу после отказа и найти информацию БД непротиворечивой и не повреждённой в результате отказа.

Система должна обеспечивать корректную обработку ситуаций, вызванных недопустимыми и несогласованными значениями входных данных. В указанных случаях пользователю должны выдаваться соответствующие уведомления, после чего система должна возвращаться в рабочее состояние.

Для обеспечения устойчивости к отказам электроснабжения все устройства хранения и обработки информации должны быть подключены к электросети через источники бесперебойного питания.

### Требования безопасности

Все технические решения, использованные при создании системы, а также при определении требований к аппаратному обеспечению, должны соответствовать действующим нормам и правилам техники безопасности, пожаробезопасности и взрывобезопасности, а также охраны окружающей среды при эксплуатации.

### Требования к эргономике и технической эстетике

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав системы, должно осуществляться посредством визуального графического интерфейса (GUI). Интерфейс системы должен быть понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм. Навигационные элементы должны быть выполнены в удобной для пользователя форме. Средства редактирования информации должны удовлетворять принятым соглашениям в части использования функциональных клавиш, режимов работы, поиска, использования оконной системы.

### Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

Система должна быть рассчитана на эксплуатацию в составе программно-технического комплекса Заказчика. Техническая и физическая защита аппаратных компонентов системы, носителей данных, бесперебойное энергоснабжение, резервирование ресурсов, текущее обслуживание реализуется техническими и организационными средствами, предусмотренными в ИТ инфраструктуре Заказчика.

Для нормальной эксплуатации разрабатываемой системы должно быть обеспечено бесперебойное питание персональных компьютеров (ПК). При эксплуатации системы должна быть обеспечена соответствующая стандартам хранения носителей и эксплуатации ПК температура и влажность воздуха.

### Требования к защите информации от несанкционированного доступа

Разрабатываемая система должна обеспечивать разграничение доступа на уровне отдельных программных модулей и структур данных. Компоненты подсистемы защиты от НСД должны обеспечивать:

– идентификацию пользователя;

– проверку полномочий пользователя при работе с системой;

– разграничение доступа пользователей на уровне задач и информационных массивов.

Разрабатываемая система должна использовать «слепые» пароли (при наборе пароля его символы не показываются на экране либо заменяются одним типом символов).

Регистрация для администраторов не должна быть предусмотрена в рамках данной системы, так как она проходит не через публичный интерфейс, а через внутреннюю систему управления организации.

### Требования по сохранности информации при авариях

Программное обеспечение Системы должно восстанавливать свое функционирование при корректном перезапуске аппаратных средств. Должна быть предусмотрена возможность организации автоматического и (или) ручного резервного копирования данных системы средствами системного и базового программного обеспечения (ОС, СУБД), входящего в состав программно-технического комплекса Заказчика.

Приведенные выше требования не распространяются на компоненты системы, разработанные третьими сторонами и действительны только при соблюдении правил эксплуатации этих компонентов, включая своевременную установку обновлений, рекомендованных производителями покупного программного обеспечения.

### Требования к патентной чистоте

Установка системы в целом, как и установка отдельных частей системы, не должна предъявлять дополнительных требований к покупке лицензий на программное обеспечение сторонних производителей, кроме программного обеспечения, указанного в разделе 4.3.3.

## Требования к функциям, выполняемым системой

### Модель вариантов использования

На рис. 2 представлена диаграмма вариантов использования для разрабатываемой системы продажи билетов на междугородние автобусные рейсы. В системе существует два типа пользователей: Пассажир и Администратор.

Пассажир – это пользователь системы, который может выполнять различные действия, связанные с приобретением билетов и управлением своим профилем. Для начала работы пассажир проходит авторизацию в системе (вариант использования «Войти в систему»), при необходимости также доступна регистрация нового пользователя (вариант использования «Зарегистрироваться»). После входа в систему пассажир может управлять своим профилем (вариант использования «Управление профилем»), что включает возможность редактировать данные (например, имя, пароль, номер телефона и паспортные данные), а также добавить банковскую карту для оплаты билетов.

Основная функция для пассажира – это поиск и покупка билетов. Пассажир может просматривать доступные рейсы (вариант использования «Просмотр доступных рейсов») и выбрать рейс на конкретную дату и время отправления. После выбора рейса пассажир может приобрести билет (вариант использования «Купить билет»), включая возможность покупки билета для себя или других пассажиров. Для удобства пассажир также может просматривать информацию о своих купленных билетах (вариант использования «Просмотр билетов») и распечатывать их для предъявления при посадке (вариант использования «Печать билетов»). Помимо этого, пассажир имеет возможность просматривать расписание рейсов (вариант использования «Просмотр расписания»), чтобы узнать время отправления и прибытия автобусов.

Администратор – это пользователь системы с расширенными правами доступа, который выполняет задачи по управлению данными и функционалом системы. Администратор может управлять рейсами (вариант использования «Управлять рейсами»), включая добавление новых рейсов, редактирование существующих и их удаление. Также администратор имеет возможность управлять автобусами (вариант использования «Управлять автобусами»), водителями (вариант использования «Управлять водителями») и маршрутами (вариант использования «Управлять маршрутами»).

Для анализа и отчетности администратор может просматривать отчеты (вариант использования «Просмотр отчетов»), а также генерировать отчеты по проданным билетам или выполненным рейсам и распечатывать их (варианты использования «Генерация отчетов» и «Печать отчетов»). Кроме того, администратор может управлять пользователями системы (вариант использования «Управлять пользователями»), добавляя новых пользователей, редактируя их данные или удаляя учетные записи.

Таким образом, система предоставляет пассажирам удобный функционал для поиска рейсов, покупки и печати билетов, управления профилем и просмотра расписаний. Администраторы, в свою очередь, могут управлять всеми аспектами системы: рейсами, автобусами, водителями, маршрутами, пользователями и отчетами, что обеспечивает эффективное функционирование системы и анализ её работы.

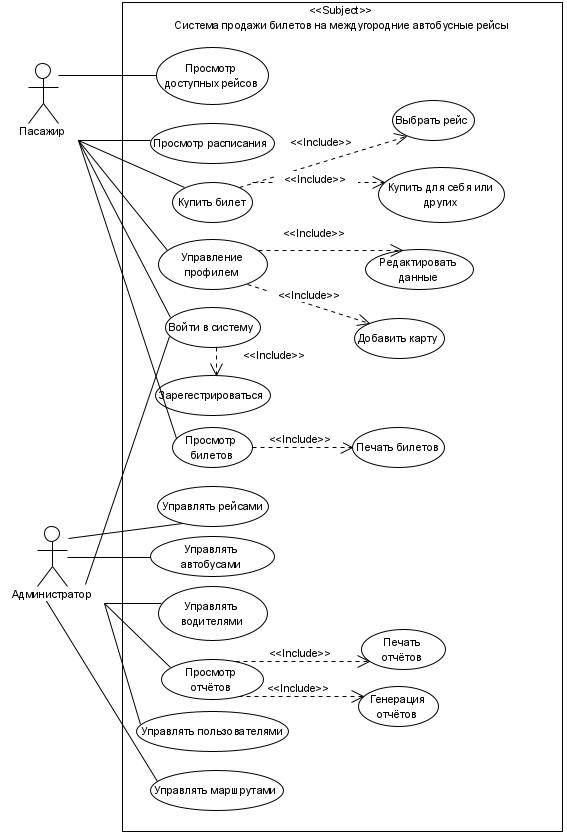


Рисунок 2 – Диаграмма вариантов использования

### Спецификация варианта использования «Купить билет»

**Краткое описание**: описывается процесс покупки билета пассажиром на междугородний автобусный рейс. Пассажир может выбрать покупку билета для себя или для другого человека.

Область действия: система как «чёрный ящик».

Уровень цели: цель пользователя («уровень моря»).

**Основной поток событий**:

1. Пассажир решает купить билет.

2.Система отображает доступные рейсы.

3.Пассажир выбирает интересующий рейс.

4.Система предлагает выбрать, для кого покупается билет:

* Для себя.
* Для другого человека.

5.Если билет покупается для себя:

* Система использует данные пассажира из его профиля.

6.Если билет покупается для другого человека:

* Система запрашивает:
  + Имя,
  + Фамилию,
  + Отчество,
  + Паспортные данные.

7.Система проверяет, добавлена ли у пассажира банковская карта:

* Если карта добавлена, переходит к следующему шагу.
* Если карты нет, система предлагает добавить карту.

8.Пассажир выбирает банковскую карту для оплаты.

9.Система проверяет наличие свободных мест на выбранный рейс.

10.Система выполняет оплату билета.

11.Система уменьшает количество свободных мест на выбранный рейс.

12.Система уведомляет пассажира об успешной покупке билета.

13.Система предоставляет возможность распечатать билет.

**Альтернативные потоки**:

7А. Банковская карта не добавлена

1.Система обнаруживает, что у пассажира нет добавленной банковской карты.

2.Система предлагает пассажиру добавить карту.

3.После добавления карты процесс покупки продолжается с шага 8..

9А. Нет свободных мест

1.Система обнаруживает, что на выбранный рейс нет свободных мест.

2.Система уведомляет пассажира об отсутствии свободных мест.

3.Вариант использования завершается неуспешно.

10А. Ошибка при оплате

1.Система обнаруживает ошибку при выполнении оплаты.

2.Система уведомляет пассажира о неудачной попытке оплаты.

3.Вариант использования возвращается к шагу 8 для выбора другой карты.

**Предусловия**: Клиент должен войти в систему.

**Гарантии успеха**: Билет будет успешно куплен, количество свободных мест на рейс уменьшится, и пассажир сможет распечатать билет..

**Минимальные гарантии**: Если банковская карта отсутствует, произошла ошибка при оплате или нет свободных мест на рейс, система уведомит пассажира о проблеме, и билет не будет оформлен.

На рис. 3 ниже представлена диаграмма деятельности для варианта использования «Купить билет».

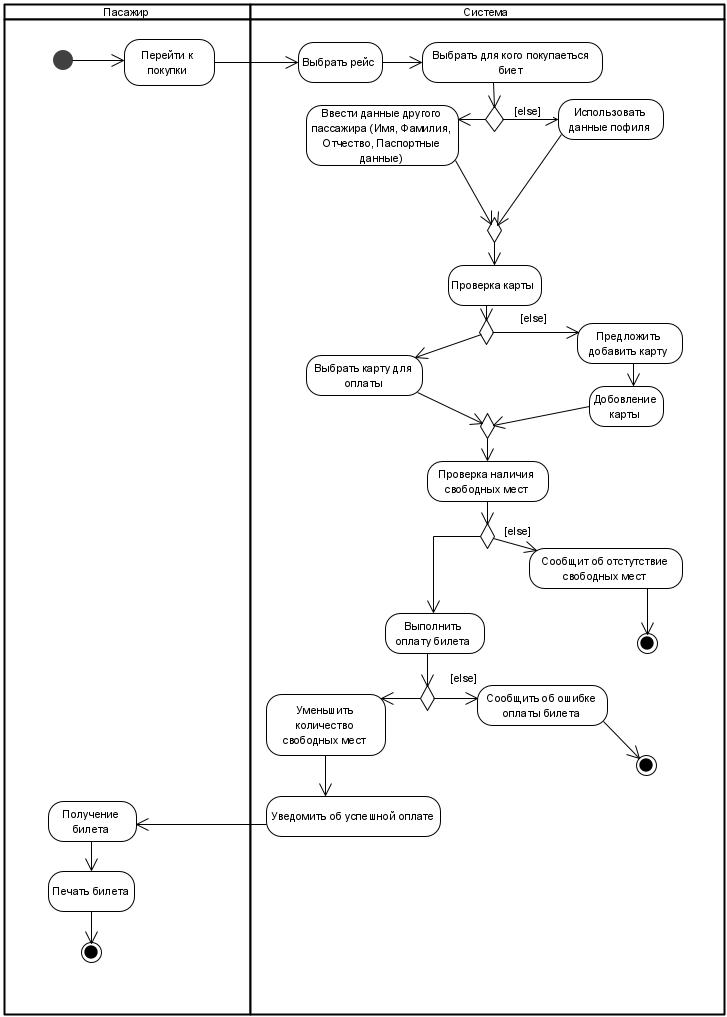


Рисунок 3 – Диаграмма деятельности для варианта использования «Купить билет»

## Требования к видам обеспечения

### Требования к информационному обеспечению

Уровень хранения данных в Системе должен быть построен на платформе реляционной СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД.

База данных предназначена для хранения:

* сведений о сотрудниках организации и клиентах,
* сведений о заказах, строках заказов и их содержимом,
* сведений о пиццах и ингредиентах.

### Требования к лингвистическому обеспечению

Программное обеспечение системы должно быть разработано на языке программирования C# с использованием .NET 8.0.

Все прикладное программное обеспечение системы для организации взаимодействия с пользователем должно использовать русский язык (язык интерфейса).

### Требования к программному обеспечению

Проект должен использовать СУБД PostgreSQL 16.2, но он должен быть легко перестраиваемым для других реляционных БД.

Система должна использовать C# и Entity Framework с провайдером Npgsql для доступа из программы к базе данных.

Должен использоваться шаблон проектирования MVVM.

Система должна быть реализована на основе WPF.

Разрабатываемая Система должна быть рассчитана на функционирование в следующей программной среде:

**Серверная группа**

ПО, устанавливаемое на компьютеры серверной группы:

1. Базовая ОС – Microsoft Windows Server 2022.

2. Система управления базами данных – PostgreSQL 16.2.

**Рабочие станции**

Типовое программное обеспечение, устанавливаемое на рабочие станции (ПК): базовая операционная система ̶ Windows 10, среда выполнения .NET 8.0.0 или выше.

### Требования к техническому обеспечению

Техническое обеспечение системы должно максимально и наиболее эффективным образом использовать существующие технические средства.

В состав комплекса должны входить следующие технические средства:

1) сервер БД;

2) персональные компьютеры (ПК) пользователей.

Минимальные требования к характеристикам компонентов технического обеспечения, при которых значения временных параметров Системы должны соответствовать предъявленным в ТЗ требованиям (п. 4.2.3):

1) для сервера БД:

– процессор – 4 х Intel Xeon 3 ГГц;

– объем оперативной памяти – 64 Гб;

– дисковая подсистема – 4 х 500 Гб;

– сетевой адаптер – 1000 Мбит/с.

2) для ПК пользователя:

– процессор – 2 x Intel Core i3 1.5 ГГц;

– объем оперативной памяти – 4 Гб;

– дисковая память – 128 Гб;

– сетевой адаптер – 100 Мбит/с.

### Требования к организационному обеспечению

Организационное обеспечение системы должно быть достаточным для эффективного выполнения персоналом возложенных на него обязанностей при осуществлении автоматизированных и связанных с ними неавтоматизированных функций системы продажи билетов на междугородние автобусные рейсы.

Заказчиком должны быть подготовлены изменения в положения о структурных подразделениях, в которых будет эксплуатироваться система.

Заказчиком должны быть подготовлены изменения к действующим должностным инструкциям для персонала, который будет участвовать в эксплуатации системы.

Должностные инструкции должны определять функциональные обязанности и ответственность сотрудников, участвующих в обслуживании и эксплуатации системы:

1. Основными обязанностями Администратора являются:

* управление пользователями системы, включая добавление, редактирование и удаление данных пользователей;
* управление рейсами, включая добавление новых рейсов и редактирование существующих;
* управление автобусами и водителями, включая внесение новой информации и удаление устаревших данных;
* создание и редактирование расписаний для автобусов;
* создание маршрутов междугородних рейсов;
* генерация отчетов по проданным билетам, выполненным рейсам и другим аспектам работы системы;
* печать отчетов для проведения анализа работы организации.

1. Основными обязанностями Пассажира являются:

* просмотр доступных рейсов и расписаний автобусов;
* покупка билетов на выбранные рейсы для себя или других лиц;
* добавление банковских карт для оплаты билетов;
* просмотр информации о купленных билетах;
* распечатка билетов для предъявления при посадке.

Заказчиком должен быть подготовлен приказ о приёмке системы BTS и вводе её в эксплуатацию с указанием ответственных за эксплуатацию системы.

# Состав и содержание работ по созданию системы

Разработка и сдача проекта должна вестись по этапам (#).

|  |  |  |  |
| --- | --- | --- | --- |
| **Наименование этапа** | **Результаты этапа** | **Дата начала этапа** | **Дата завершения этапа** |
| 1. Разработка, согласование и утверждение технического задания | Техническое задание | 23.09.2024 | 05.10.2024 |
| 2. Разработка, согласование и утверждение технорабочего проекта | Технорабочий проект | 05.10.2024 | 15.12.2024 |
| 3. Подготовка презентации и защита проекта | Пояснительная записка и презентация | 16.12.2024 | 25.12.2024 |

Таблица 5.1 – Этапы работ над проектом

# Порядок контроля и приемки системы

По окончании работ проект принимается Приёмной комиссией.

Заседание комиссии проводится в конце 5-го семестра перед зачётной неделей.

Проект подлежит защите. Защита проекта проводится Разработчиком перед членами Приёмной комиссии.

Приёмная комиссия назначается из числа преподавателей кафедры ПОКС.

Комиссии предоставляется полностью оформленная и подписанная пояснительная записка и презентация.

Проект принимается, если он удовлетворяет всем пунктам данного технического задания. По результатам защиты проекта Приёмная комиссия выставляет оценку.

# Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

Для ввода Системы в действие необходимо:

1. подготовить у Заказчика всё необходимое техническое обеспечение;
2. установить на сервер и клиентские ПК системное, базовое и прикладное ПО;
3. установить на сервер БД;
4. ввести данные в справочники БД:

* сведения о пассажирах и администраторах,
* сведения об автобусах и водителях,
* сведения о маршрутах и расписаниях,
* сведения о статусах рейсов и билетов

1. подготовить организационное обеспечение;
2. провести испытания системы.

# Требования к документированию

## Требования к составу документов

По окончании работ над проектом все разработанные документы объединяются в Пояснительную записку. Она должна содержать:

1. титульный лист,
2. аннотацию,
3. оглавление,
4. термины, определения и сокращения, использованные в Пояснительной записке,
5. Задание,
6. введение,
7. Техническое задание,
8. Технорабочий проект;
9. список литературы.

Заказчику предоставляются:

1. Пояснительная записка в формате MS Word 2016;
2. Пояснительная записка, распечатанная на бумаге формата А4 – 1 экземпляр;
3. Презентация (в формате MS PowerPoint), демонстрирующая основные проектные решения.

## Требования к оформлению документов

Техническое задание и Технорабочий проект оформляются в соответствии с ГОСТ 34.201-89 [2], ГОСТ 34.601-90 [3], ГОСТ 34.602-89 [4] и РД 50-34.698.90 [5].

### Технорабочий проект

Технорабочий проект должен содержать следующие разделы:

1. общие положения;
2. описание процесса деятельности;
3. основные технические решения;
4. мероприятия по подготовке объекта автоматизации к вводу системы в действие;
5. схема функциональной структуры;
6. описание автоматизируемых функций;
7. описание комплекса технических средств;
8. описание программного обеспечения;
9. описание информационного обеспечения;
10. описание организационной структуры;
11. руководство пользователя.

# Источники разработки

При разработке ТЗ использовались следующие источники:

1. ГОСТ 34.003-90. Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения.
2. ГОСТ 34.201-89. Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем.
3. ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.
4. ГОСТ 34.602-89. Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.
5. РД 50-34.698.90. Методические указания. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов.

# Система доставки пиццы. Технорабочий проект

# Общие положения

## Наименование системы

Полное наименование системы — «Система продажи билетов на междугородние автобусные рейсы».

Условное обозначение системы — «BTS» (Bus Ticketing System).

## Основание для разработки

Разработка ведется на основании задания на курсовой проект по дисциплинам: «Проектирование и архитектура ПС», «Конструирование ПО».

Задание утверждено на заседании кафедры ПОКС 01.09.2024 и выдано преподавателями кафедры Игнатьевым Е.Б., Мочаловым А.С, и Фоминой О.В.

## Перечень организаций, участвующих в разработке системы, сроки выполнения этапов

Заказчик системы: Федеральное государственное бюджетное образовательное учреждение высшего образования «Ивановский государственный энергетический университет имени В.И. Ленина» (ИГЭУ); 153003, г. Иваново, ул. Рабфаковская, д. 34.

Разработчик системы: Федоров Кирилл Андреевич, студент группы 3-42.

Сроки выполнения этапов работы над проектом показаны в таблице 1.1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Наименование этапа** | **Результаты этапа** | **Дата начала этапа** | **Дата завершения этапа** |
| 1. Разработка, согласование и утверждение технорабочего проекта | Технорабочий проект | 05.10.2024 | 15.12.2024 |
| 2. Подготовка презентации и защита проекта | Пояснительная записка и презентация | 16.12.2024 | 25.12.2024 |

Таблица 1.1 – Этапы работ над проектом

## Цели, назначение и области использования

Основными целями создания системы являются:

* упрощение процесса покупки билетов пассажирами;
* предоставление возможности покупки билетов для себя или для другого пассажира;
* повышение управляемости за счет возможности проведения анализа данных о продажах билетов и формирования отчетов;
* автоматизация управления автобусами, маршрутами, расписаниями и водителями;
* обеспечение контроля за продажами билетов и рейсами в реальном времени.

BTS-система предназначена для автоматизации процесса продажи билетов на междугородние автобусные рейсы, управления расписаниями и рейсами, а также формирования отчетов по продажам билетов за определенный период.

## Подтверждение соответствия проектных решений действующим нормам и правилам техники безопасности, пожаро- и взрывобезопасности

Все технические решения, использованные при разработке проекта, соответствуют действующим нормам и правилам техники безопасности, пожаробезопасности и взрывобезопасности, а также охраны окружающей среды при эксплуатации.

## Нормативно-технические документы, использованные при проектировании

При проектировании использовалась спецификация текущей версии UML – Unified Modeling Language (OMG UML), v. 2.5.1. Дата релиза: декабрь 2017 года.- http://www.omg.org/spec/UML/2.5.1/PDF.

# Описание процесса деятельности

Система BTS автоматизирует процессы покупки билетов пассажирами, их обработки администраторами и создания отчетов по проданным билетам за определенный период.

Пассажиры могут просматривать доступные рейсы и расписание движения автобусов. После выбора интересующего рейса пассажиры совершают покупку билетов. При покупке билета пассажир может выбрать, для кого приобретается билет (для себя или другого человека). Если билет приобретается для другого пассажира, необходимо ввести данные этого человека (имя, фамилию, отчество и паспортные данные). После успешной покупки билет можно распечатать для предъявления при посадке.

Администраторы управляют данными в системе: добавляют автобусы, водителей, маршруты и расписание. Администраторы также могут управлять пользователями и генерировать отчеты по проданным билетам и выполненным рейсам за конкретный период.

Таким образом, система обеспечивает автоматизацию процесса покупки билетов, управления рейсами и отчетности, что облегчает работу как для пассажиров, так и для администраторов.

# Основные технические решения

## Архитектура системы

Система BTS имеет двухуровневую клиент-серверную архитектуру (см. рис. 3.1).

На уровне данных размещается хранение данных, что содержит СУБД и БД, предназначенная для хранения данных о пользователях системы, ассортименте пиццерии и заказах клиентов.

На прикладном уровне размещается исполняемый код приложения и среда выполнения Microsoft .Net 8.0, автоматизирующие процедуры управления заказами.

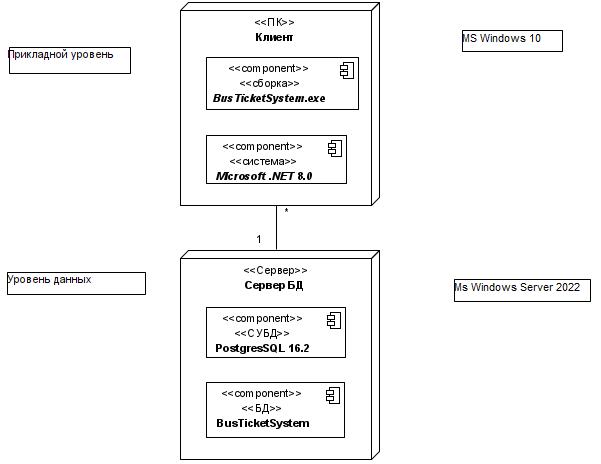


Рис. 3.1 – Диаграмма развертывания

Работа с пользовательским интерфейсом выполняется по шаблону MVVM (рис. 3.2).

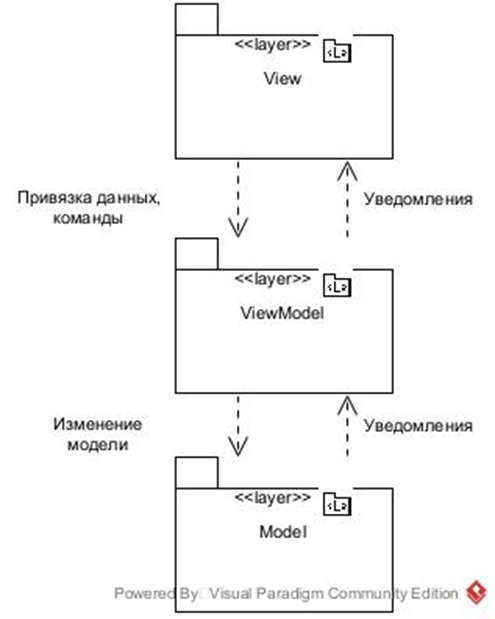


Рисунок 3.2 – Структурный шаблон MVVM

Слой View (представление) определяет визуальный интерфейс, через который пользователь взаимодействует с приложением.

Слой Model (модель) описывает используемые данные.

Слой ViewModel (модель представления) служит прослойкой между моделью и представлением посредством механизма привязки данных, благодаря чему модель и представление напрямую не связаны.

## Решения по режимам функционирования, диагностированию работы системы

В работе системы можно выделить следующие режимы работы:

1) работа неавторизованного пользователя (при входе в систему и регистрации);

2) работа авторизованного пользователя.

Специальных процедур диагностирования состояния компонентов системы не предусмотрено.

## Решения по численности, квалификации и функциям персонала, режимам его работы, порядку взаимодействия

BTS – многопользовательское приложение.

Авторизация пользователя и его права доступа к различным средствам приложения управляются централизованно из БД, которая соединена с прикладной программой.

Посредством приложения доступна регистрация клиентов организации (публичный интерфейс), регистрация администраторов через данный интерфейс предусмотрена не будет.

Не требуется никакого обучения для пользователя, знакомого с компьютером, чтобы он мог использовать данную систему. Простого объяснения цели и основных особенностей приложения будет достаточно, чтобы использовать программу.

## Сведения об обеспечении заданных в ТЗ потребительских характеристик системы, определяющих ее качество

### Надёжность

Для обеспечения устойчивости к отказам электроснабжения все устройства хранения и обработки информации подключаются к электросети через источники бесперебойного питания соответствующей мощности.

### Требования к эргономике и технической эстетике

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав системы, должно осуществляться посредством визуального графического интерфейса (GUI). Интерфейс системы должен быть понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм. Навигационные элементы должны быть выполнены в удобной для пользователя форме. Средства редактирования информации должны удовлетворять принятым соглашениям в части использования функциональных клавиш, режимов работы, поиска, использования оконной системы.

### Защищенность информации

Защита от несанкционированного доступа обеспечивается в системе BTS за счёт парольной аутентификации Пользователя при входе в систему и разделением прав доступа для трех классов пользователей:

1) Пассажир;

2) Администратор;

Для восстановления работоспособности системы после сбоев предусмотрено резервное копирование БД средствами СУБД PostgreSQL 16.2.

## Состав функций, реализуемых системой

На диаграмме вариантов использования (Рис. 3.3) приведены функции системы.

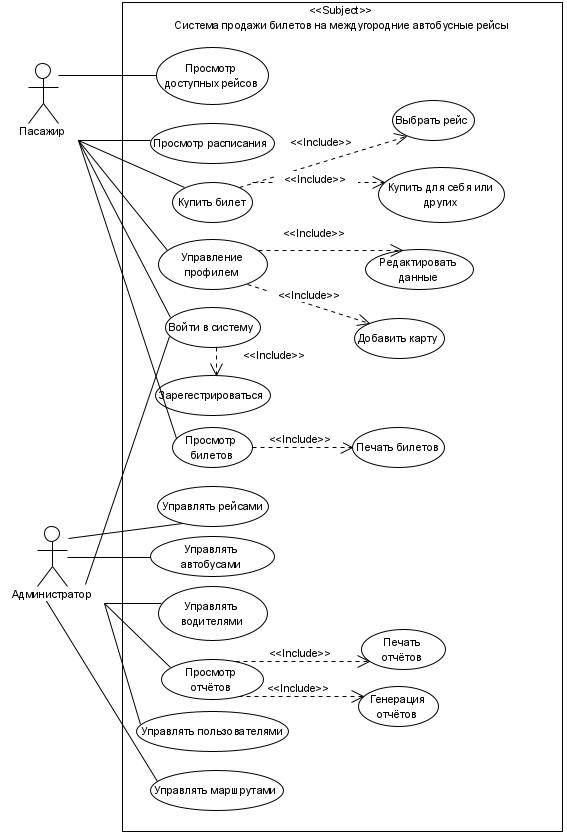


Рисунок 3.3 – Диаграмма вариантов использования

Определены следующие функции для абстрактного Пользователя:

1. Войти в Систему;
2. Редактировать профиль.

Предполагаются такие функции (варианты использования) для пассажира:

1. Зарегистрироваться в Системе
2. Просмотреть доступные рейсы
3. Выбрать рейс (включает выбор даты и времени отправления)
4. Купить билет (включает следующие варианты использования):

* Выбрать, для кого покупается билет (для себя или другого пассажира)
* Выбрать карту для оплаты

1. Просмотреть купленные билеты (включает возможность распечатать билет)
2. Добавить карту для оплаты
3. Просмотреть расписание рейсов

Выделены следующие функции для Курьера:

1. Управлять рейсами (добавление, редактирование, удаление рейсов)
2. Управлять автобусами (добавление, редактирование, удаление автобусов)
3. Управлять водителями (добавление, редактирование, удаление водителей)
4. Управлять маршрутами (создание, редактирование, удаление маршрутов)
5. Управлять пользователями (добавление, редактирование, удаление пользователей)
6. Создавать расписание рейсов
7. Просматривать отчеты по билетам и рейсам

* Генерировать отчеты (например, по проданным билетам за месяц)
* Печатать отчеты

## Решения по комплексу технических средств

В соответствии с ТЗ:

1) в качестве сервера БД будет использован компьютер, имеющийся у автовокзала и установленный в специальном серверном помещении;

2) в качестве ПК пользователей будут использованы компьютеры, установленные в настоящее время на рабочих местах на автовокзале, а также в любом месте у клиентов;

3) все компьютеры объединены локальной вычислительной сетью с пропускной способностью не менее 100 Мбит/с.

Характеристики сервера БД:

– процессор – 4 х Intel Xeon 3 ГГц;

– объем оперативной памяти – 64 Гб;

– дисковая подсистема – 4 х 500 Гб;

– сетевой адаптер – 100 Мбит/с.

Характеристики ПК пользователя:

– процессор – 2 x Intel Core i3 1.5 ГГц;

– объем оперативной памяти – 4 Гб;

– дисковая память – 128 Гб;

– сетевой адаптер – 100 Мбит/с.

## Решения по информационному обеспечению

Компонентом информационного обеспечения является системная БД «BusTicketSystem».

БД существует заранее и загружена данными относительно пользователей, автобусов, водителей, маршрутов и расписаний. Прикладная программа системы продажи билетов извлекает эту информацию из БД, предоставляя пассажирам информацию о доступных рейсах и расписании, организует процессы регистрации пассажиров и авторизации всех пользователей, обновляя информацию о пользователях, хранящуюся в базе данных.

Система также организует процесс покупки билетов пассажирами и управления рейсами администраторами, занося соответствующие сведения в базу данных. Администраторы могут управлять данными о рейсах, автобусах, водителях и маршрутах, а также генерировать отчеты для анализа работы системы.

## Физическая модель БД

На рисунке 3.4 представлена физическая модель БД, построенная в приложении pgAdmin.

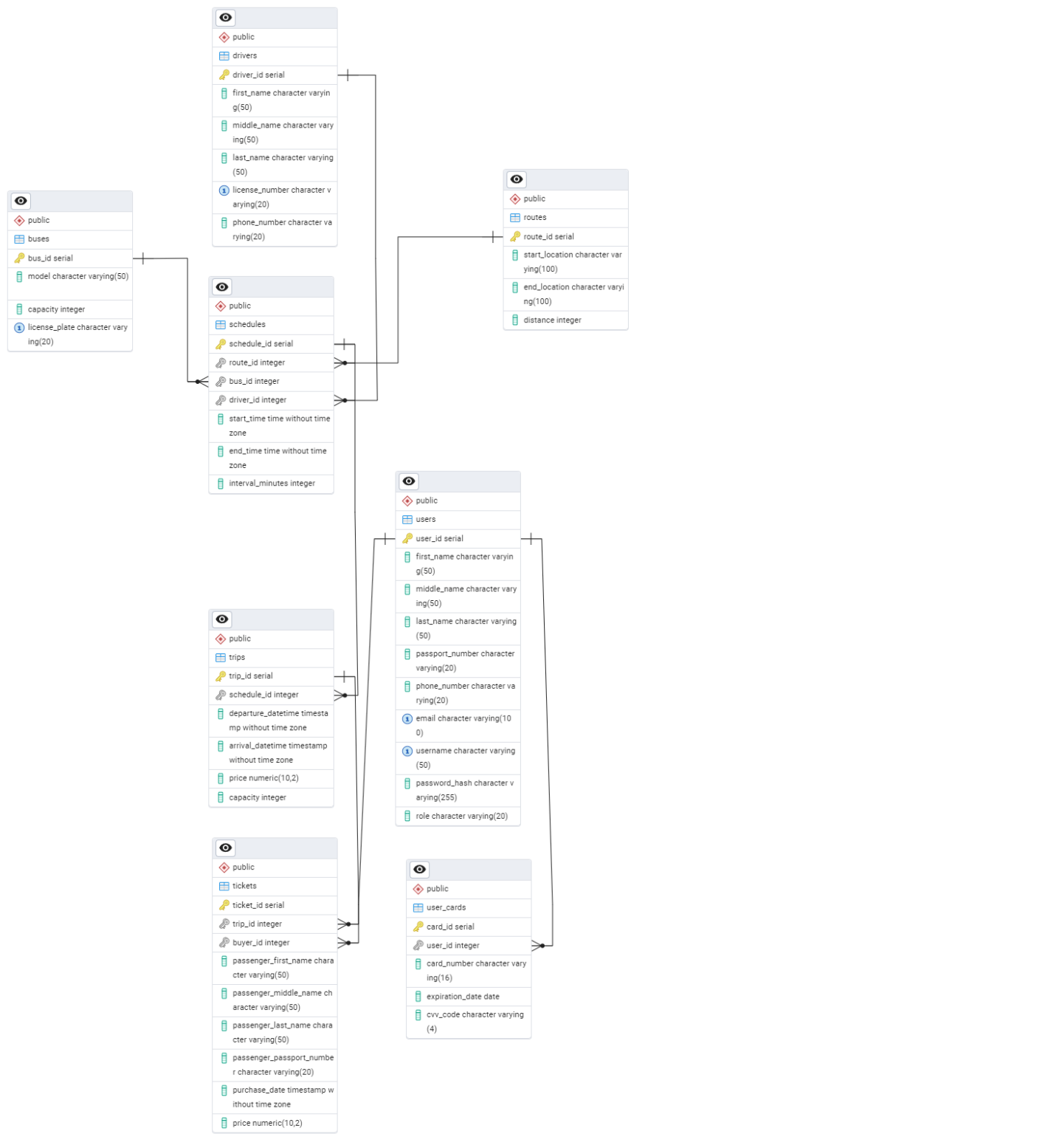


Рисунок 3.4 – Физическая модель базы данных

### Скрипт БД

CREATE TABLE Users (

user\_id SERIAL PRIMARY KEY,

first\_name VARCHAR(50) NOT NULL,

middle\_name VARCHAR(50),

last\_name VARCHAR(50) NOT NULL,

passport\_number VARCHAR(20),

phone\_number VARCHAR(20),

email VARCHAR(100) UNIQUE NOT NULL,

username VARCHAR(50) UNIQUE NOT NULL,

password\_hash VARCHAR(255) NOT NULL,

role VARCHAR(20) CHECK (role IN ('Passenger', 'Administrator')) DEFAULT 'Passenger'

);

CREATE TABLE User\_Cards (

card\_id SERIAL PRIMARY KEY,

user\_id INTEGER REFERENCES Users(user\_id) ON DELETE CASCADE,

card\_number VARCHAR(16) NOT NULL,

expiration\_date DATE NOT NULL,

cvv\_code VARCHAR(4) NOT NULL

);

CREATE TABLE Buses (

bus\_id SERIAL PRIMARY KEY,

model VARCHAR(50) NOT NULL,

capacity INTEGER NOT NULL,

license\_plate VARCHAR(20) UNIQUE NOT NULL

);

CREATE TABLE Drivers (

driver\_id SERIAL PRIMARY KEY,

first\_name VARCHAR(50) NOT NULL,

middle\_name VARCHAR(50),

last\_name VARCHAR(50) NOT NULL,

license\_number VARCHAR(20) UNIQUE NOT NULL,

phone\_number VARCHAR(20)

);

CREATE TABLE Routes (

route\_id SERIAL PRIMARY KEY,

start\_location VARCHAR(100) NOT NULL,

end\_location VARCHAR(100) NOT NULL,

distance INTEGER

);

CREATE TABLE Schedules (

schedule\_id SERIAL PRIMARY KEY,

route\_id INTEGER REFERENCES Routes(route\_id) ON DELETE CASCADE,

bus\_id INTEGER REFERENCES Buses(bus\_id) ON DELETE CASCADE,

driver\_id INTEGER REFERENCES Drivers(driver\_id) ON DELETE SET NULL,

start\_time TIME NOT NULL,

end\_time TIME NOT NULL,

interval\_minutes INTEGER NOT NULL

);

CREATE TABLE Trips (

trip\_id SERIAL PRIMARY KEY,

schedule\_id INTEGER REFERENCES Schedules(schedule\_id) ON DELETE CASCADE,

departure\_datetime TIMESTAMP NOT NULL,

arrival\_datetime TIMESTAMP NOT NULL,

price NUMERIC(10, 2) NOT NULL,

capacity INTEGER NOT NULL -- Вместимость автобуса для конкретного рейса

);

CREATE TABLE Tickets (

ticket\_id SERIAL PRIMARY KEY,

trip\_id INTEGER REFERENCES Trips(trip\_id) ON DELETE CASCADE,

buyer\_id INTEGER REFERENCES Users(user\_id) ON DELETE SET NULL,

passenger\_first\_name VARCHAR(50) NOT NULL,

passenger\_middle\_name VARCHAR(50),

passenger\_last\_name VARCHAR(50) NOT NULL,

passenger\_passport\_number VARCHAR(20) NOT NULL,

purchase\_date TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

price NUMERIC(10, 2) NOT NULL

);

## Решения по программному обеспечению

### Моделирование эскизной реализации варианта использования «купить билет»

На рисунке 3.5 представлена диаграмма последовательности для варианта использования «Купить билет».

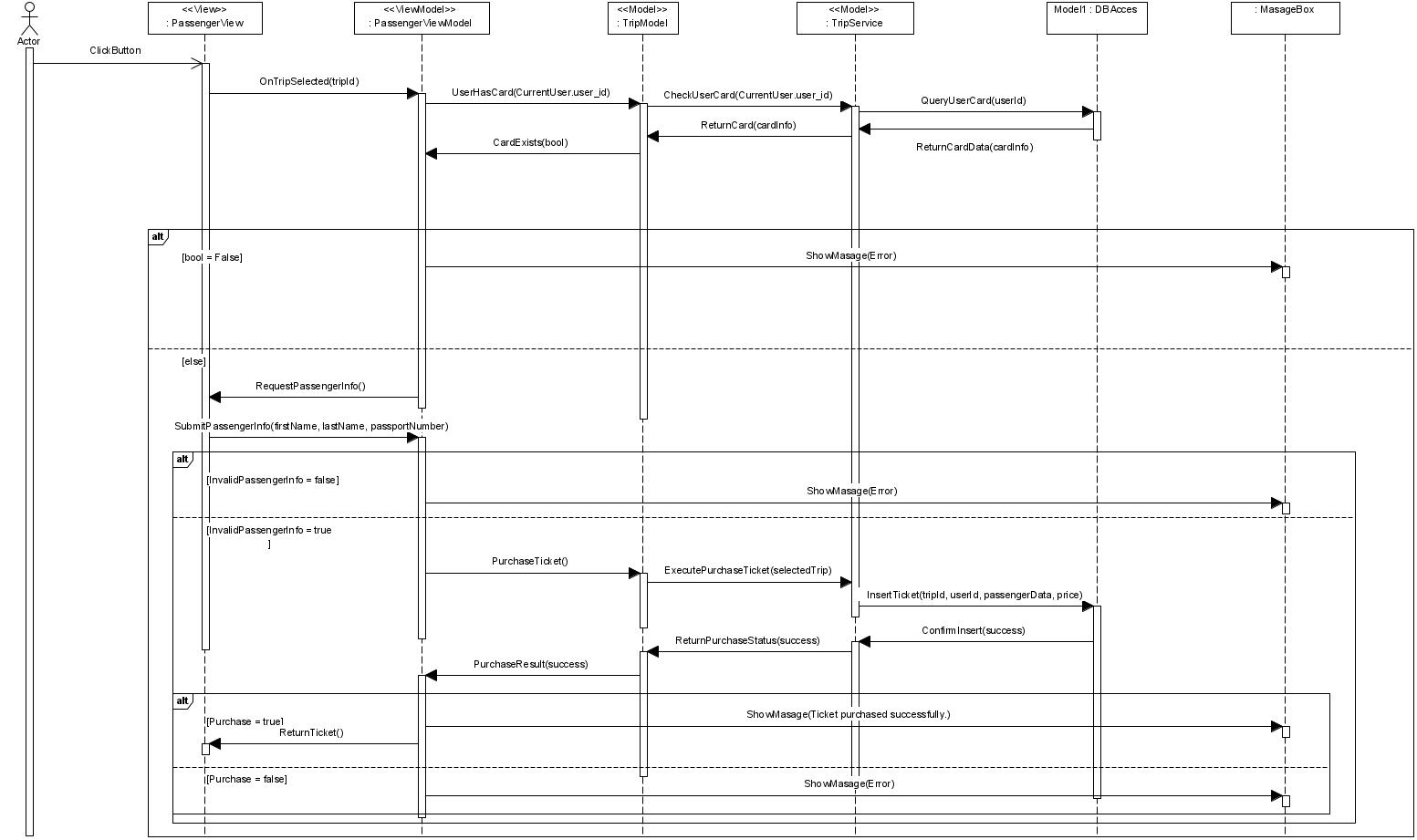


Рисунок 3.5 – Диаграмма последовательности для варианта использования «Купить билет»

Пользователь решает приобрести билет на автобусный рейс. Если у пользователя нет привязанной карты для оплаты, то вариант использования завершается неуспешно, при этом выводится соответствующее сообщение (ShowMessage(Error)). В противном случае вызывается метод CheckUserCard() объекта класса TripService, который проверяет наличие привязанной карты в базе данных через DBAccess. Если карта найдена, система запрашивает подтверждение данных пассажира для оформления билета.

Если пользователь покупает билет для другого пассажира, вызывается метод RequestPassengerInfo() для ввода необходимых данных: имя, фамилия и паспортные данные. Если хотя бы одно из полей данных пассажира пустое или неверное, вариант использования завершается неуспешно, и отображается сообщение об ошибке (ShowMessage(Error)). В противном случае вызывается метод PurchaseTicket() объекта класса TripModel для выполнения покупки билета.

Метод PurchaseTicket() вызывает ExecutePurchaseTicket(), который обращается к слою доступа к данным (DBAccess), вставляя запись о покупке билета через метод InsertTicket(). Этот метод сохраняет данные о рейсе, пассажире, цене и уменьшает количество свободных мест в рейсе. После успешного сохранения возвращается статус успешной покупки (ReturnPurchaseStatus(success)), и пользователю выводится сообщение об успешной покупке (ShowMessage(Ticket purchased successfully)).

Если покупка завершилась неуспешно (например, из-за ошибки в базе данных или отсутствия мест), система отображает сообщение об ошибке (ShowMessage(Error)). После успешной покупки обновляется список доступных рейсов (RefreshAvailableTrips()) и билетов пользователя (RefreshUserTickets()), чтобы отразить изменения в данных.

На рис. 3.7 представлена диаграмма последовательности, демонстрирующая процесс проверки наличия привязанной карты, ввода данных пассажира и выполнения покупки билета. В ходе этого процесса PassengerViewModel взаимодействует с TripModel, TripService и слоем доступа к данным (DBAccess), чтобы организовать покупку билета и обновить данные системы.

На рисунке 3.9 представлена диаграмма коммуникации для варианта использования «Купить билет».

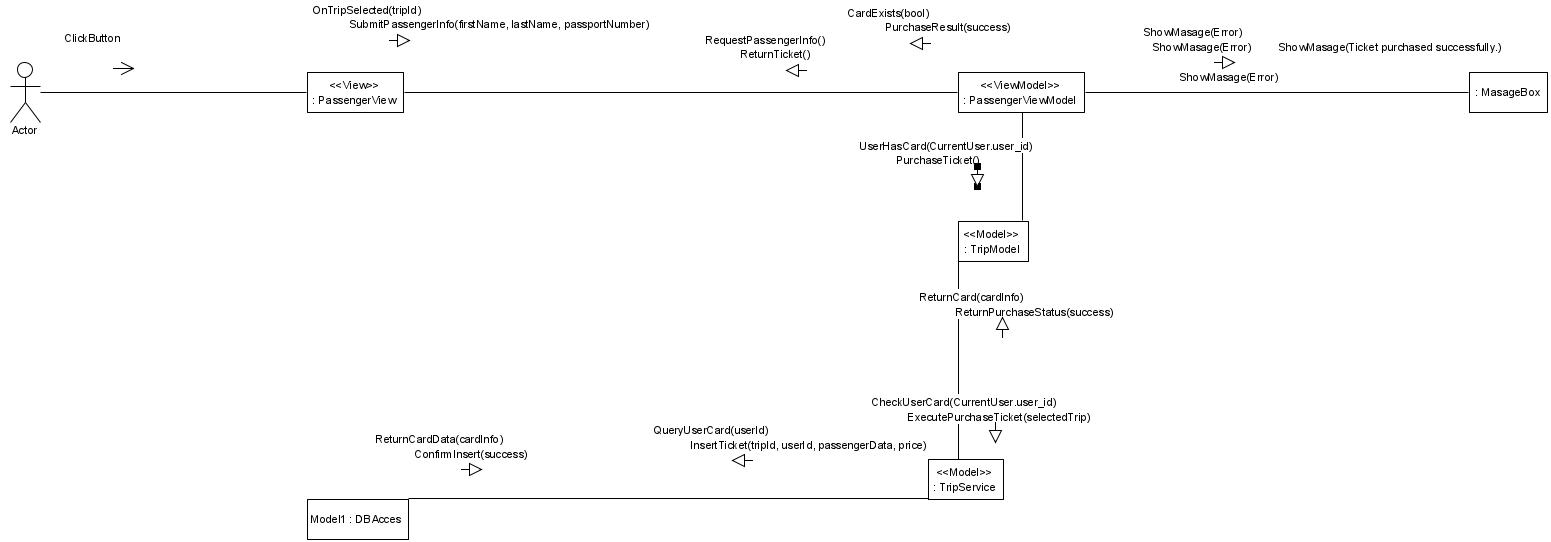


Рисунок 3.6 – Диаграмма коммуникации для варианта использования «Купить билет»

На рис. 3.7 представлена диаграмма классов для варианта использования «Купить билет».

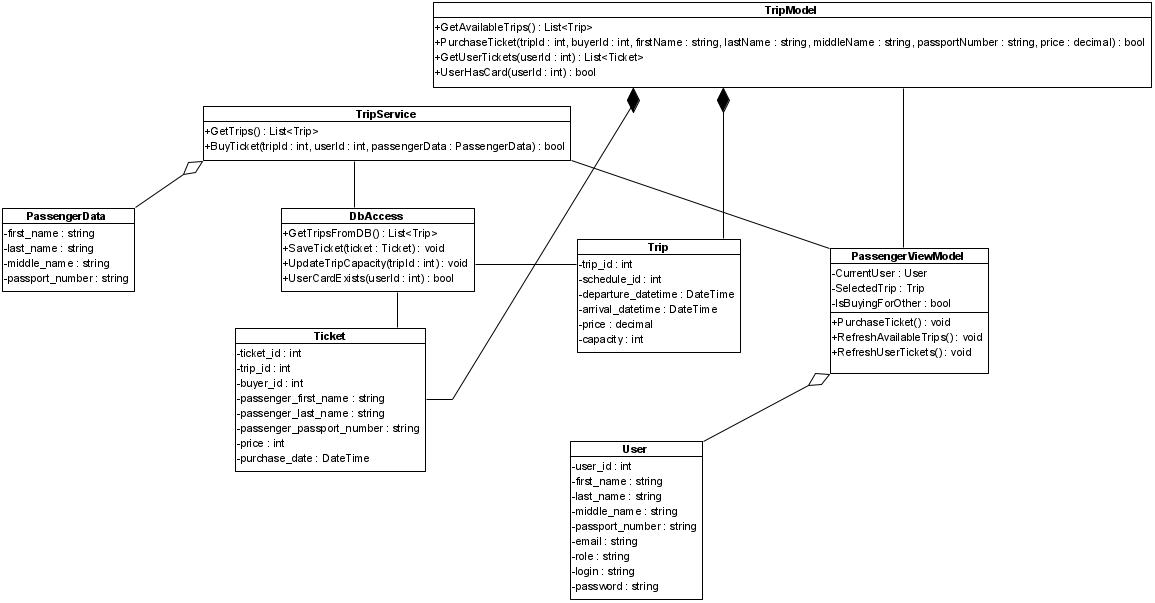


Рисунок 3.7 – Диаграмма классов для варианта использования «Оформить заказ»

На рисунке 3.8 представлена диаграмма деятельности для метода «Найти рейс» класса PassangertViewModel.

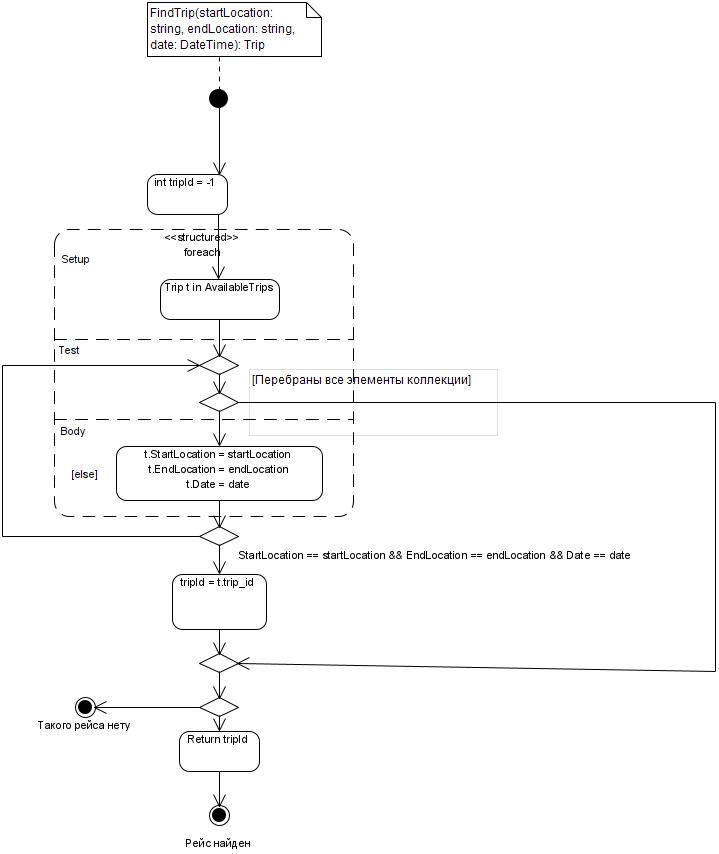


Рисунок 3.9 – Диаграмма деятельности для метода «Найти рейс»

На рис. 3.10 ниже представлена диаграмма состояний для класса PassangertViewModel.

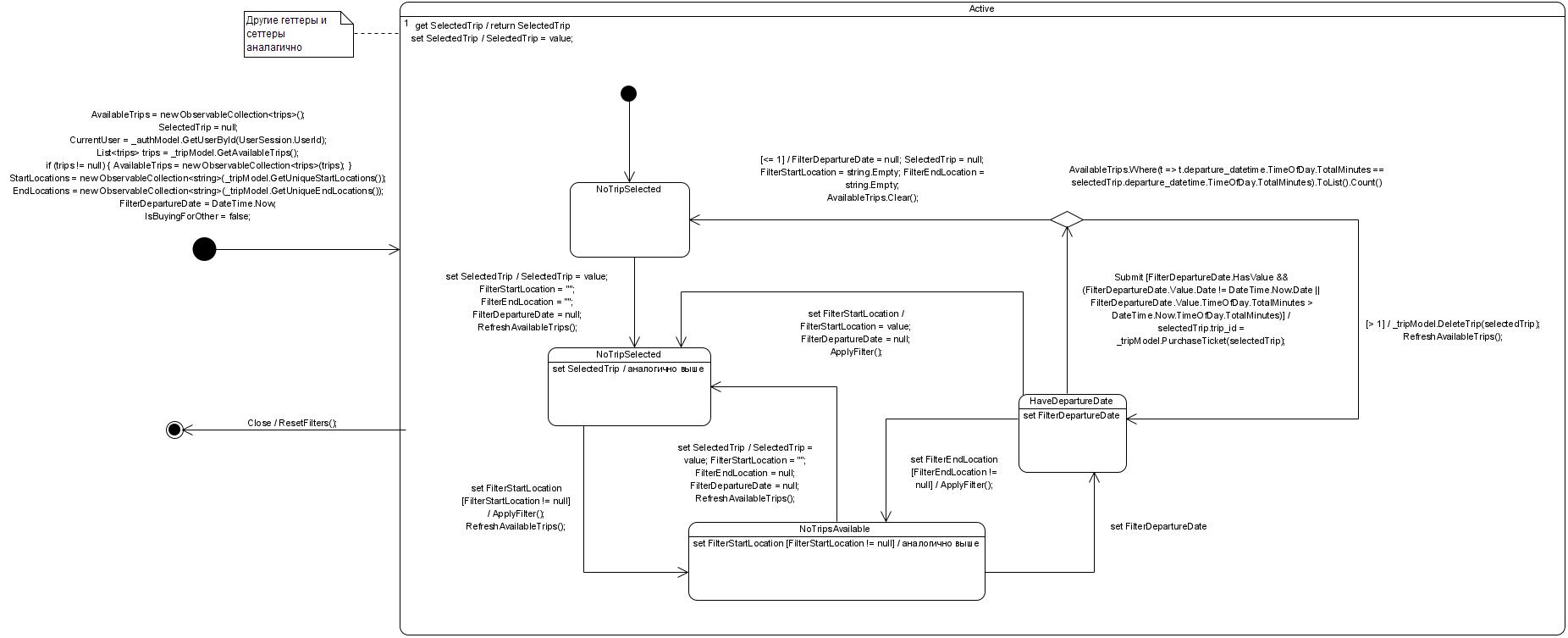


Рисунок 3.12 – Диаграмма состояний для класса PassangertViewModel

Заказ билета изначально пребывает в состоянии «Рейс не выбран». При установке фильтров для поиска рейса (например, начального и конечного пункта отправления, даты отправления), вызывается метод ExecuteApplyFilter, который фильтрует доступные рейсы. Если фильтры применены успешно, пользователь выбирает рейс, и состояние переходит в «Рейс выбран».

На этом этапе выполняется проверка наличия привязанной карты пользователя через метод UserHasCard(userId). Если карта отсутствует, система остаётся в состоянии «Рейс выбран», выводя сообщение о необходимости добавить карту. Если карта есть, пользователь вводит данные пассажира, после чего состояние переходит в «Ввод данных пассажира».

Далее вызывается метод ExecutePurchaseTicket, и система переходит в состояние «Покупка билета». Если покупка билета проходит успешно, выполняется обновление списка доступных рейсов (RefreshAvailableTrips) и билетов пользователя (RefreshUserTickets), а состояние переходит в «Билет приобретён». В случае ошибки при покупке состояние остаётся на этапе «Рейс выбран», выводя соответствующее сообщение об ошибке.

Пользователь может сбросить фильтры, что приводит к состоянию «Рейс не выбран». Также система может завершить работу, возвращая пользователя к начальному состоянию и сбрасывая фильтры через метод ResetFilters.

# Руководство пользователя

## Вход в приложение

При запуске приложения PizzaDelivery.exe откроется окно входа в приложение (рис. 4.1). Пользователю будет необходимо указать имя пользователя (логин) и пароль, после чего нажать на кнопку «Войти». Если указанные данные окажутся неверными, то отобразится соответствующее уведомление об этом пользователя (рис.4.2).

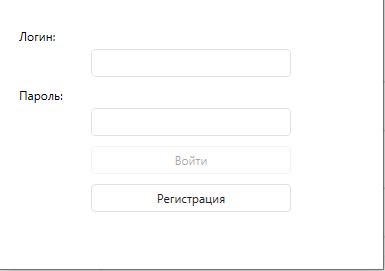


Рисунок 4.1 – Вход в приложение

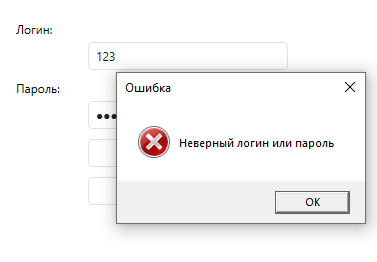


Рисунок 4.2 – Уведомление пользователя об ошибке входа

## Регистрация нового клиента

Для регистрации нового пользователя в качестве клиента необходимо в окне входа нажать на кнопку «Регестрация», тогда откроется окно регистрации нового клиента (рис. 4.3). Для регистрации нужно будет указать свои личные данные, в том числе номер телефона, адрес электронной почты и паспортные данные, а также придумать имя пользователя (логин) и пароль, после чего нажать на кнопку «Зарегистрироваться». При возникновении ошибки регистрации возникнет сообщение с уведомлением об этом пользователя.

Пользователь может перейти обратно к окну входа в приложение, закрыв окно.

Все поля являються обязательные для заполнения при регистрации поля.

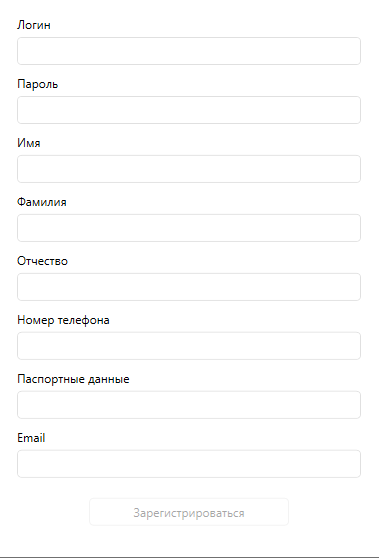


Рисунок 4.3 – Регистрация в приложении

В случае возникновения ошибки при входе будет отображено сообщение об ошибке (рис. 4.4).

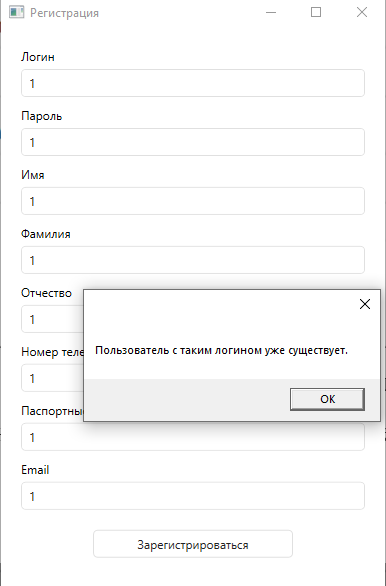


Рисунок 4.4 – Уведомление пользователя об ошибке регистрации

## Просмотр списка рейсов и их выбор клиентом

После входа в приложение или регистрации пассажир оказывается в окне просмотра рейсов(рис. 4.5).

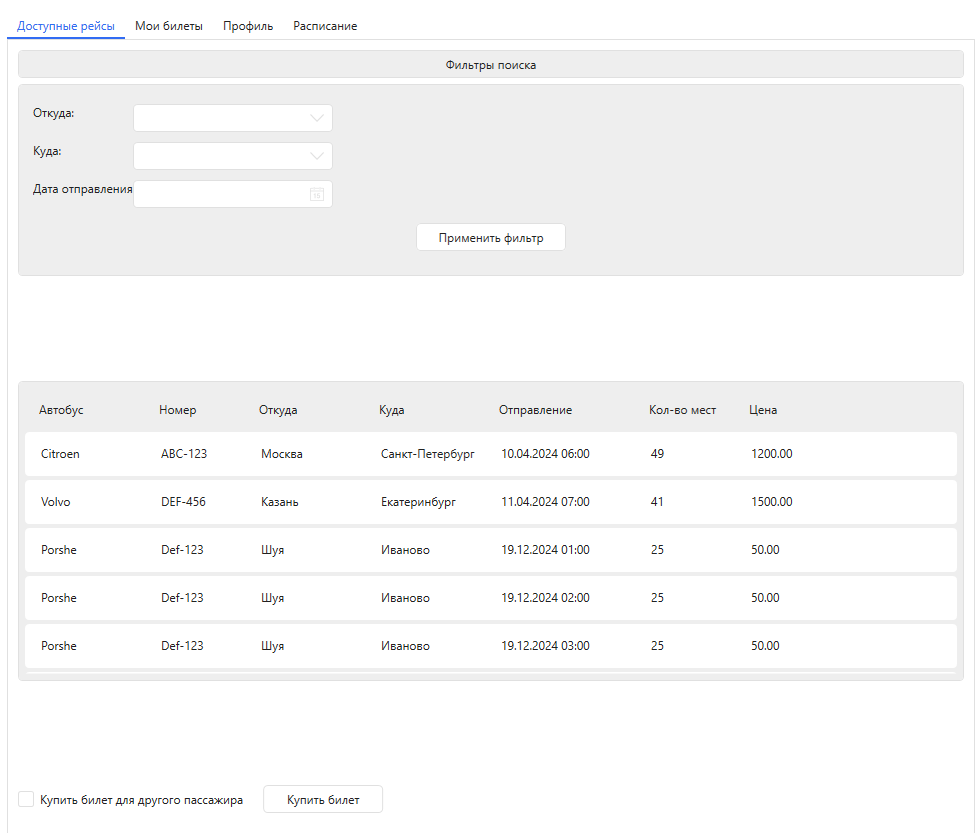


Рисунок 4.5 – Показ рейсов

Одиночным кликом мыши по рейсу клиент выбирает рейс (рис. 4.6).

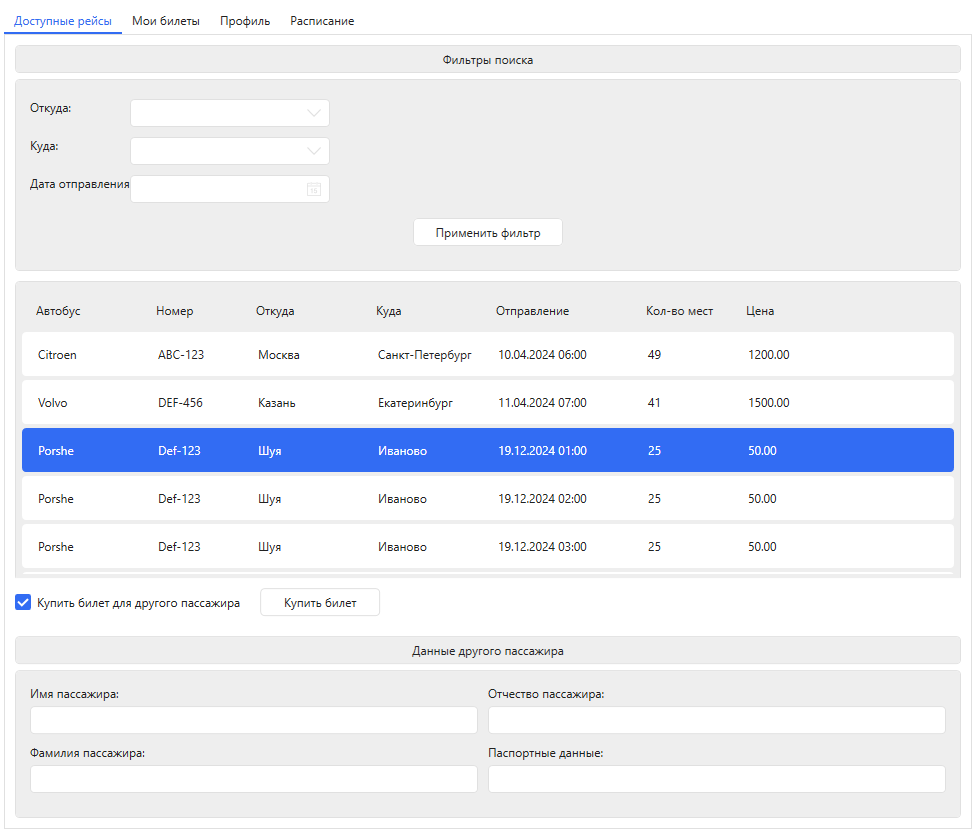


Рисунок 4.6 – Выбор рейса

Во время выбора рейса клиент может поставить флаг, что покупает билет для другого пассажира и указать его данные.

Для покупки выбранного билета «Купить билет». Процесс выбора можно прервать (тогда билет не будет куплен), если нажать на другой рейс или перейти в другую вкладку.

## Панель навигации для пассажира

На протяжении всего процесса работы авторизованного пользователя сверху окна будет находиться панель навигации (рис. 4.7).



Рисунок 4.7 – Панель навигации

Панель содержит 4 вкладки. Первая вкладка для просмотра рейсов. Вторая вкладка для просмотра купленных билетов. Третья вкладка для просмотра профиля. Четвёртая вкладка для просмотра расписания.

## Изменение профиля пассажира

Окно изменения профиля представлено на рис 4.8 ниже.

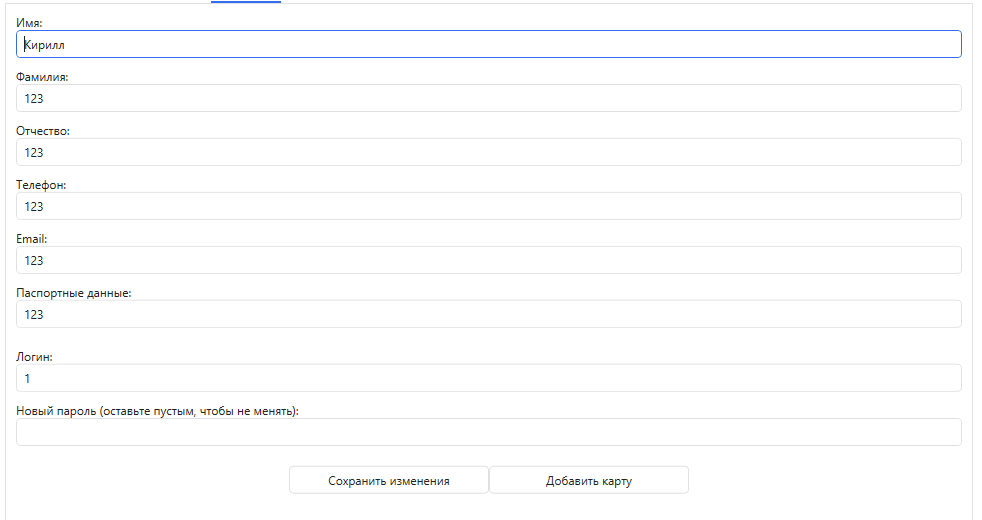


Рисунок 4.8 – Окно изменения профиля

Здесь пассажир может изменить предоставленную ранее им самим информацию о себе, в том числе и пароль. Если пароль окажется новым, то он также будет заменен.

## Список купленных билетов

В Списки купленных билетов представлены билеты, купленные пассажиром (рис. 4.9).

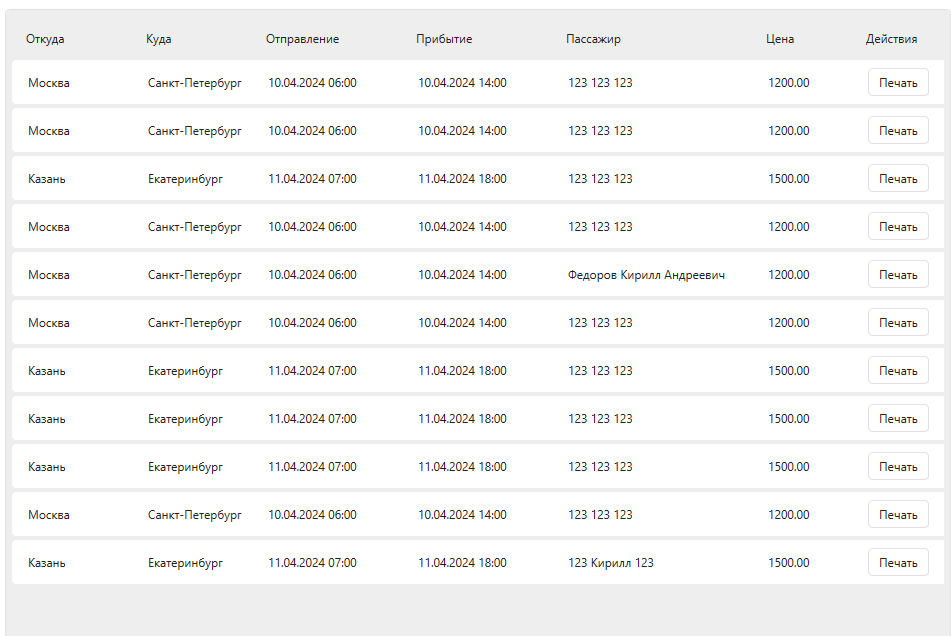


Рисунок 4.9 – Список купленных билетов

Клиент может распечатать купленный билет, нажав на кнопку «печать» (рис. 4.10).

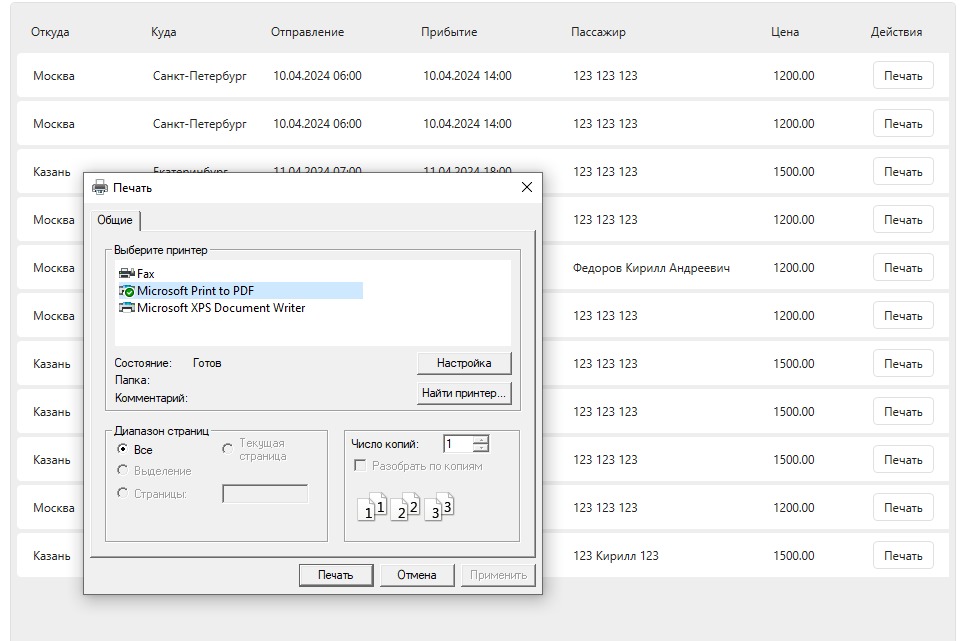


Рисунок 4.10 – Печать билета

## Просмотр расписания

В расписании представлены режим работы автобусов (рис. 4.11).

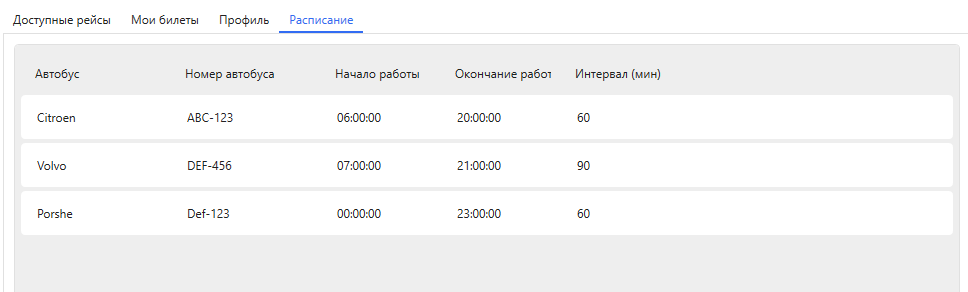


Рисунок 4.11 – Расписание

## Управление автобусами администратором

После входа в приложение администратору станут доступны все автобусы, присутствующие на автовокзале (рис. 4.12). Для администратора доступна панель навигации сверху, на которой расположены вкладки: Управление автобусами, управление пользователями, Управление водителями, Управление расписанием, Создание маршрутов, Создание рейсов и Отчёты.

Администратор может изменять автобусы изменив данные в таблице и нажав на кнопку изменить. Так же администратор может добавлять автобусы заполнив данные в форме и нажав на кнопку «Добавить» (рис. 4.13)

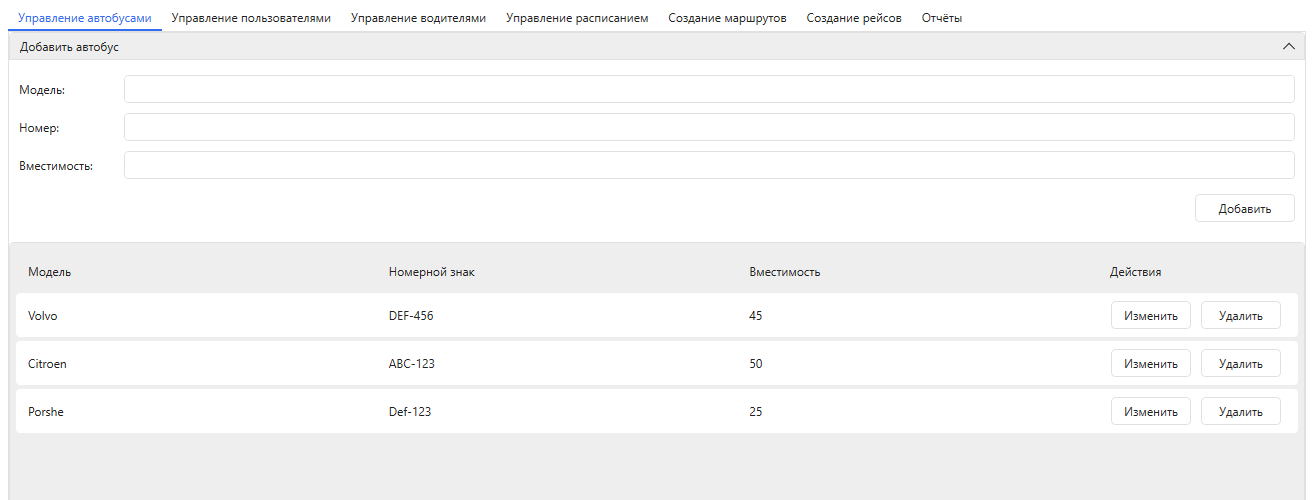


Рисунок 4.12 – Список автобусов

По нажатии на кнопку «Удалить» Администратор может удалить выбранный автобус (рис. 4.14) и подтвердить действие.

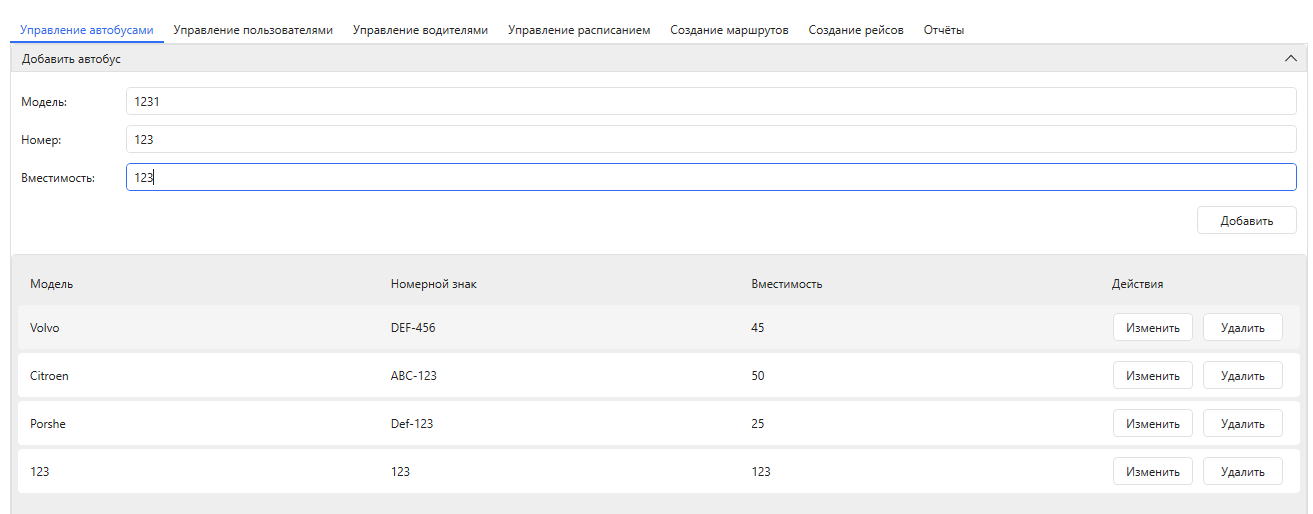


Рисунок 4.13 – Добавление автобуса

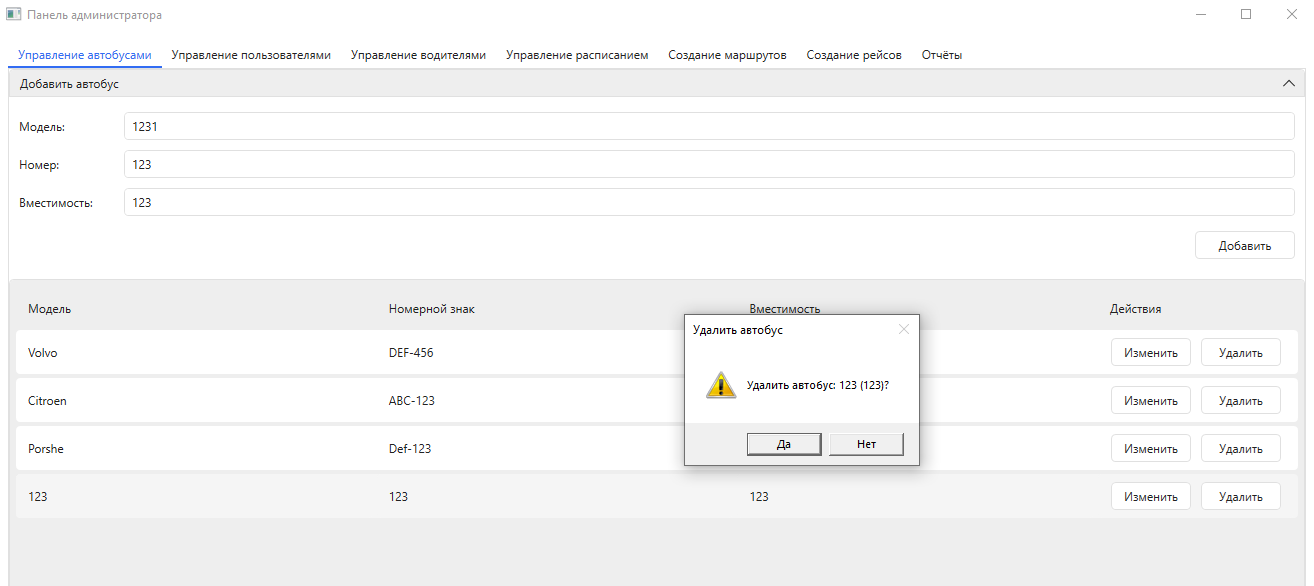


Рисунок 4.14 – Удаление автобуса

## Управление пользователями администратором

Администратору может редактировать, добавлять и удалять пользователей, зарегистрированных на автовокзале (рис. 4.15).

Администратор может изменять данные пользователя нажав на кнопку изменить, исправив данные в форме и нажав на кнопку сохранить (рис. 4.16 – 4.17). Так же администратор может добавлять пользователей нажав на кнопку «Добавить нового пользователя», заполнив данные в форме и нажав на кнопку «Сохранить» (рис. 4.18)

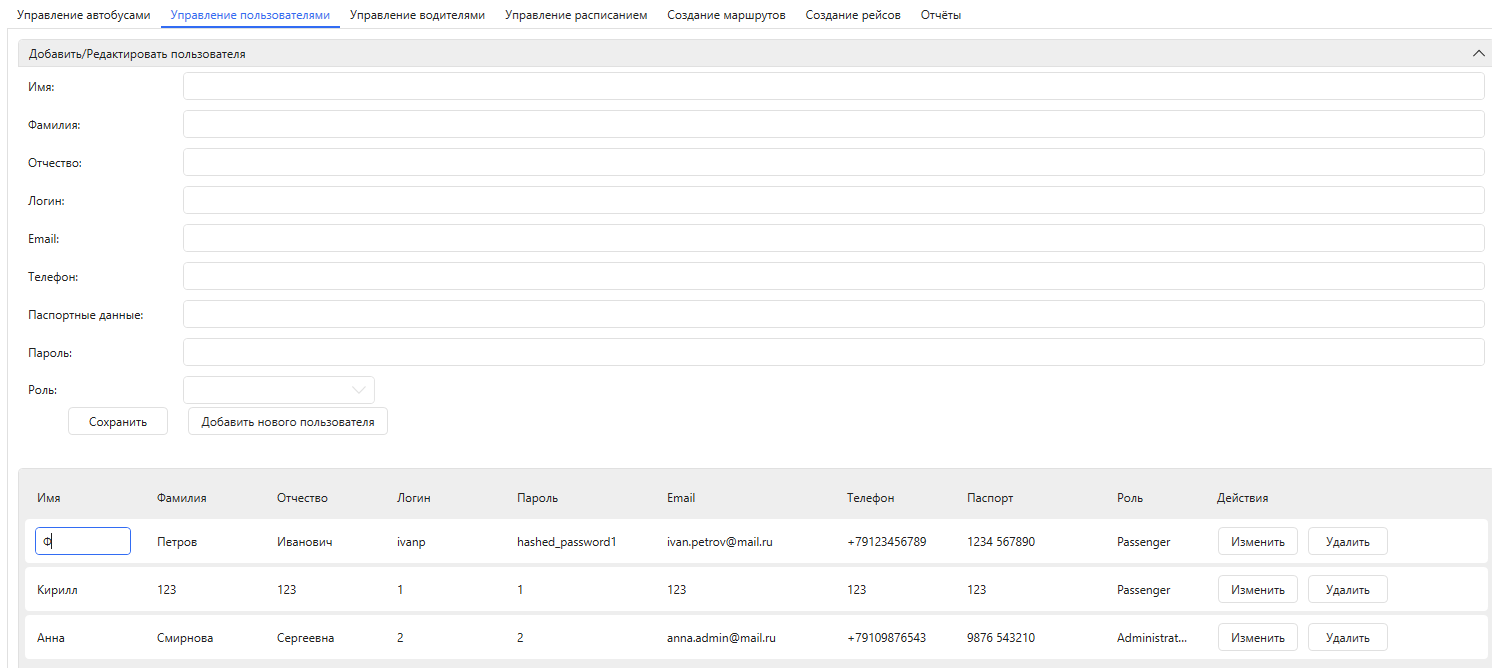


Рисунок 4.15 – Список пользователей

Аналогично по нажатии на кнопку «Удалить» Администратор может удалить пользователя и подтвердить действие.

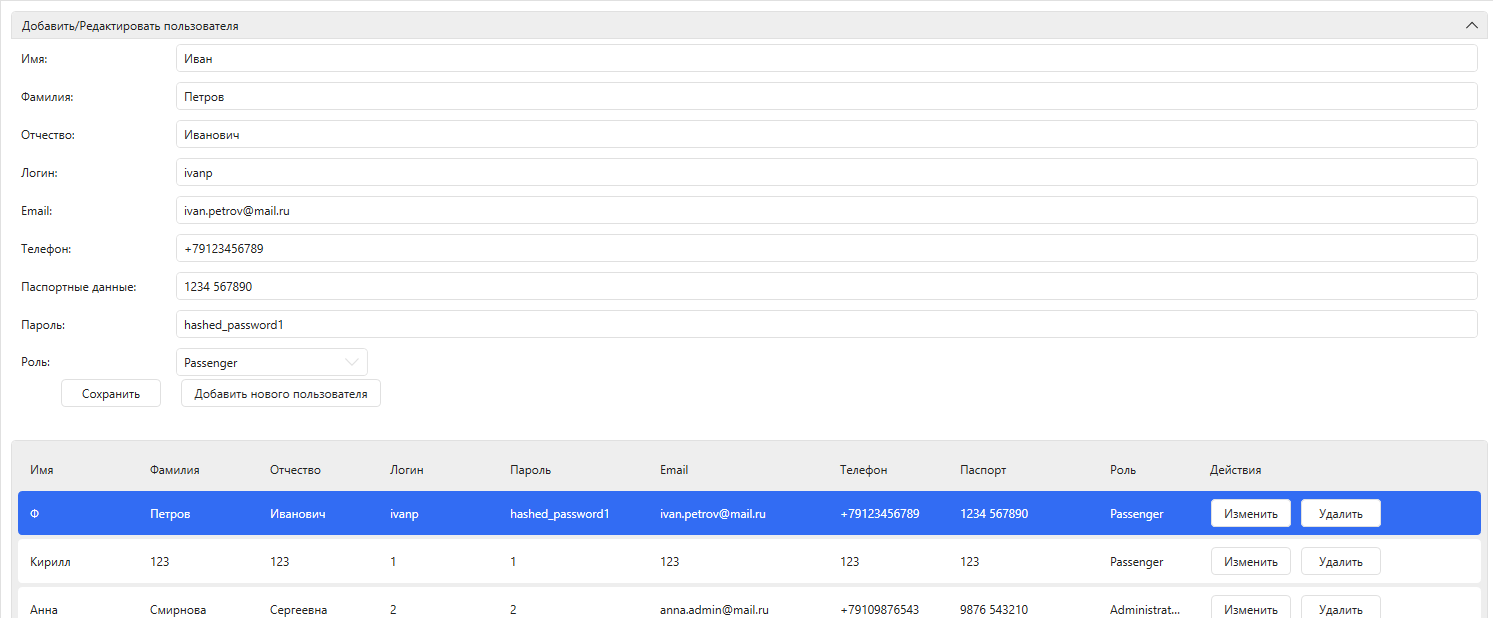


Рисунок 4.16 – Изменения данных пользователя

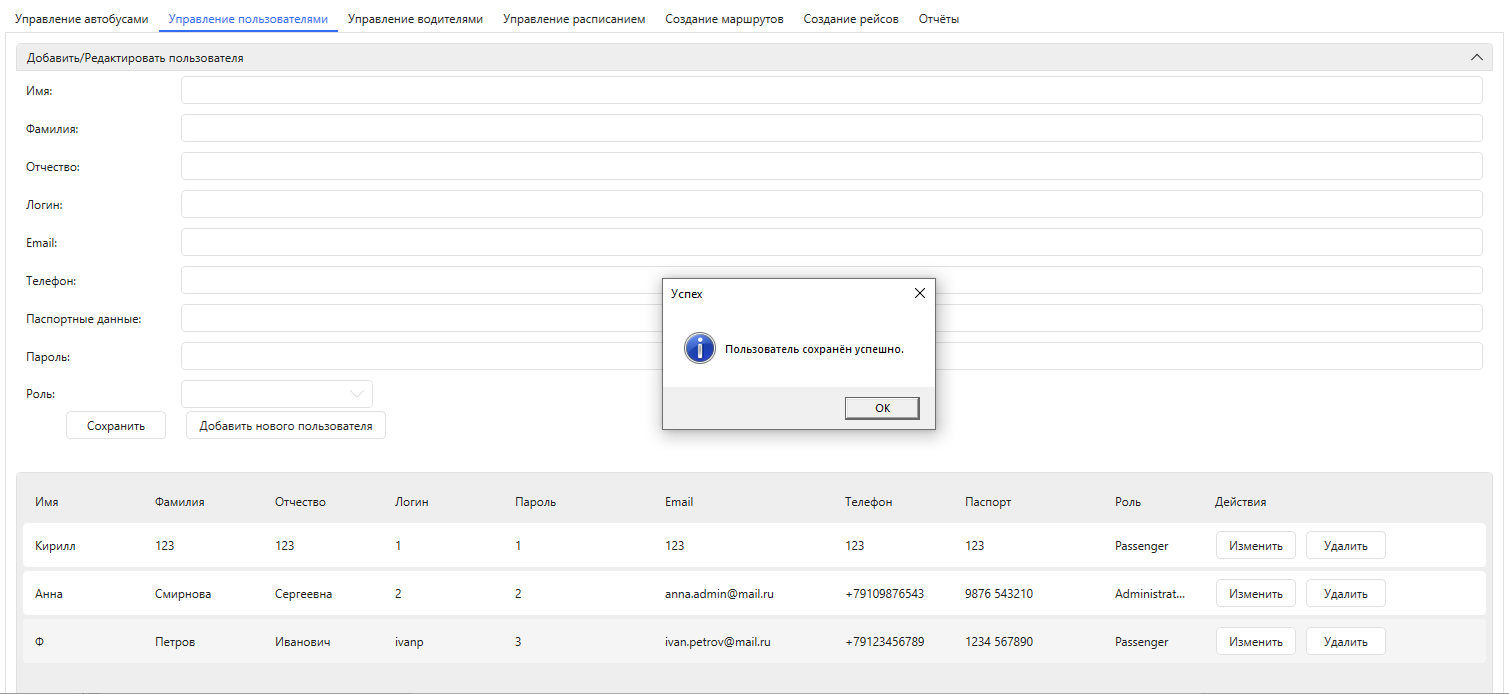


Рисунок 4.17 – Сохранение изменений

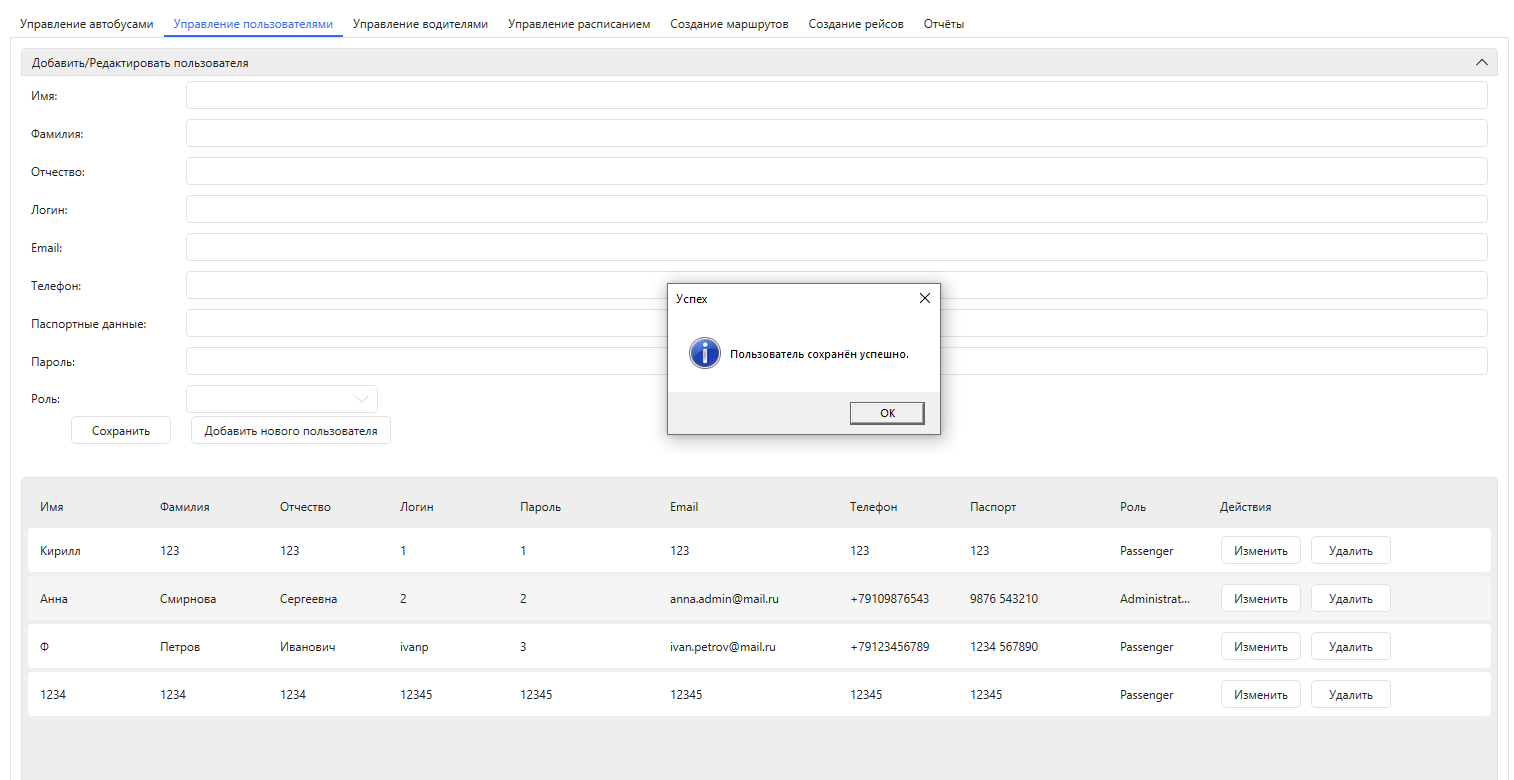


Рисунок 4.18 – Добавление нового пользователя

## Управление водителями администратором

Администратору может редактировать, добавлять и удалять водителей, зарегистрированных на автовокзале (рис. 4.19).

Администратор может изменять данные водителей нажав на кнопку изменить, исправив данные в форме и нажав на кнопку сохранить (рис. 4.20 – 4.21). Так же администратор может добавлять пользователей нажав на кнопку «Добавить нового водителя», заполнив данные в форме и нажав на кнопку «Сохранить» (рис. 4.22)

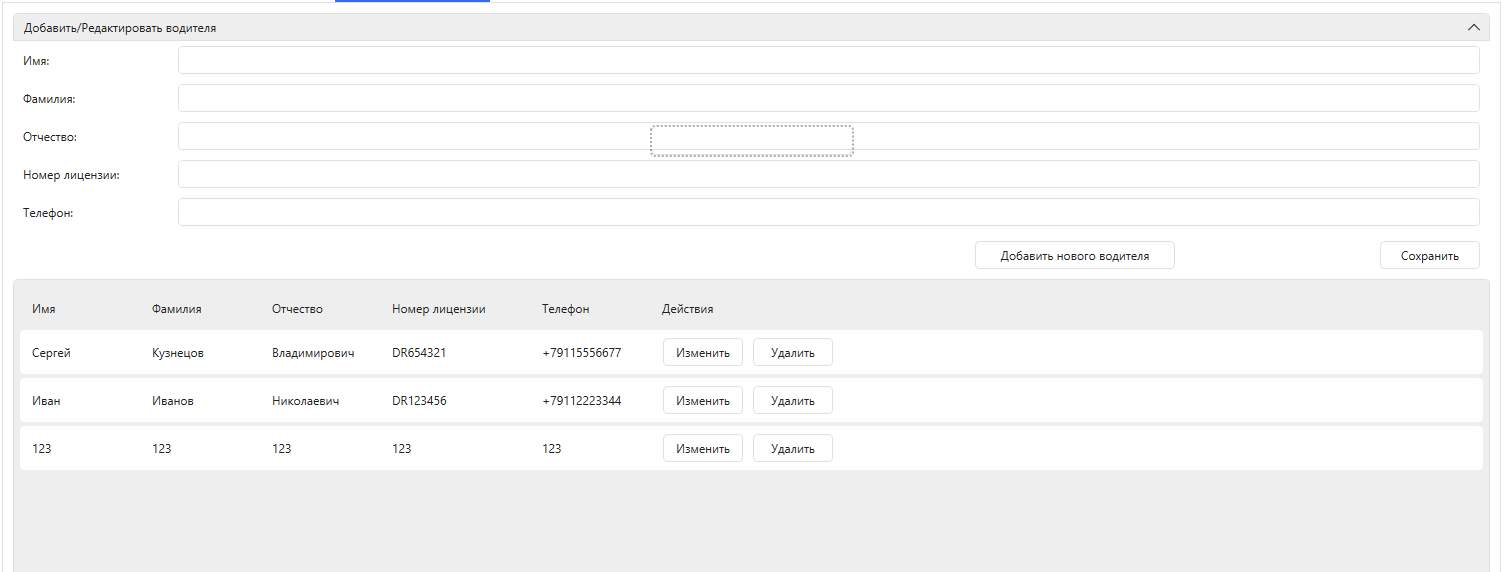


Рисунок 4.15 – Список водителей

Аналогично по нажатии на кнопку «Удалить» Администратор может удалить водителя и подтвердить действие.

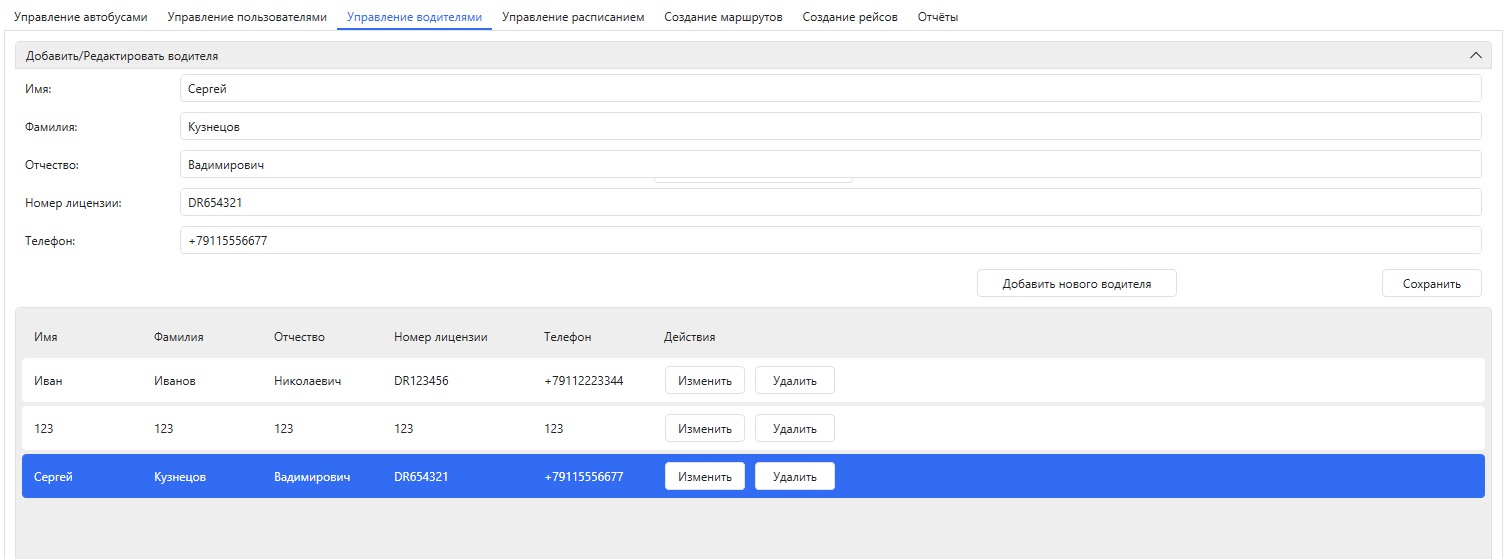


Рисунок 4.20 – Изменения данных водителя



Рисунок 4.21 – Сохранение изменений

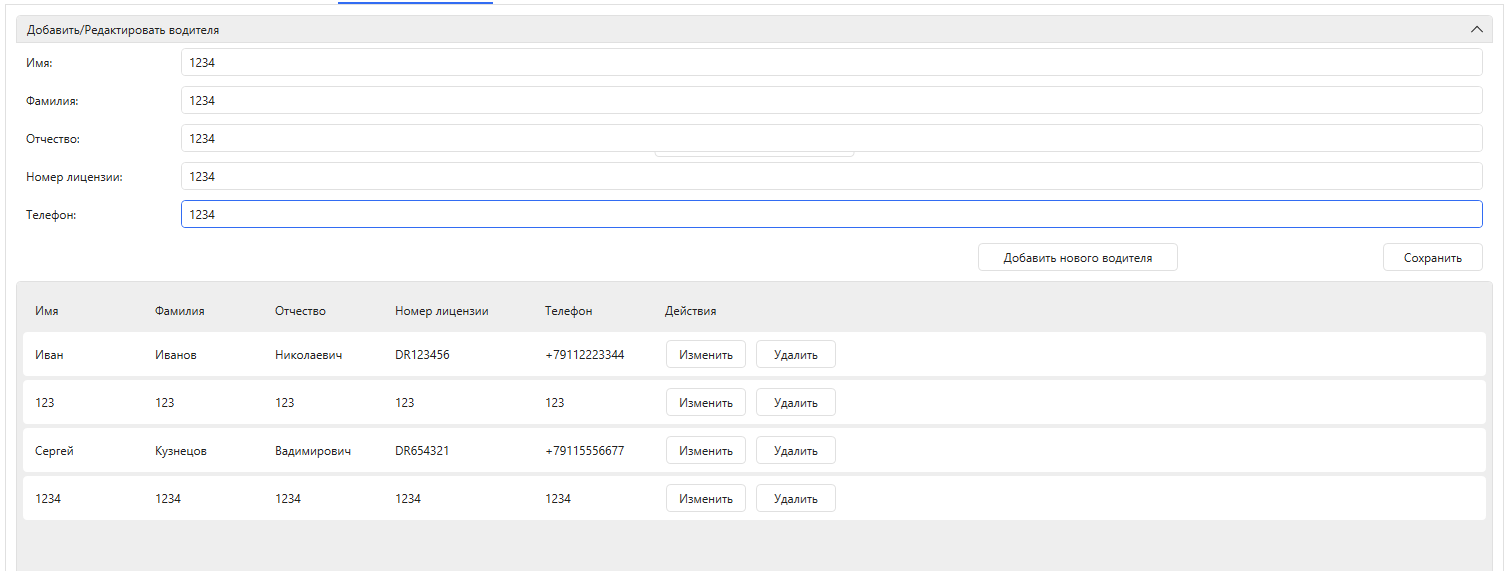


Рисунок 4.22 – Добавление нового водителя

## Управление расписанием администратором

Администратор может добавлять и удалять расписание, используемое на автовокзале (рис. 4.22).

Администратор может добавлять расписание заполнив данные в форме и нажав на кнопку «Добавить» (рис. 4.23)

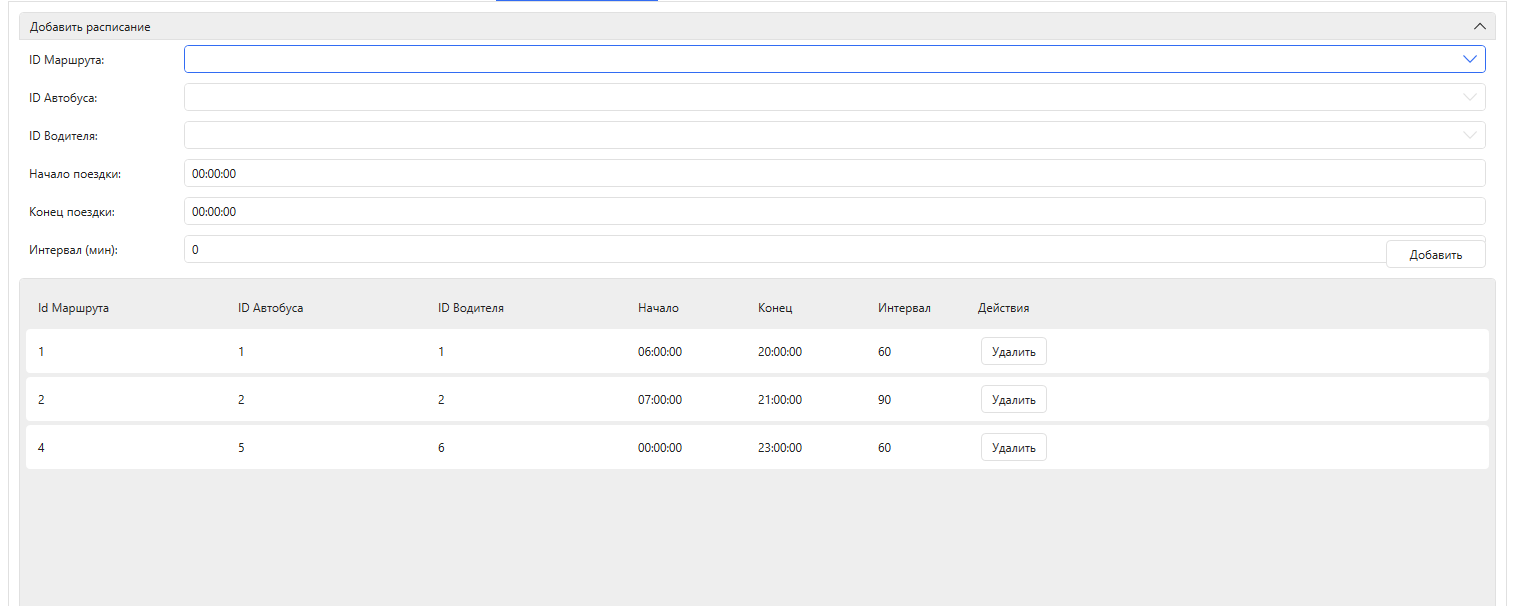


Рисунок 4.22 – Список расписания

Аналогично по нажатии на кнопку «Удалить» Администратор может удалить расписание и подтвердить действие.

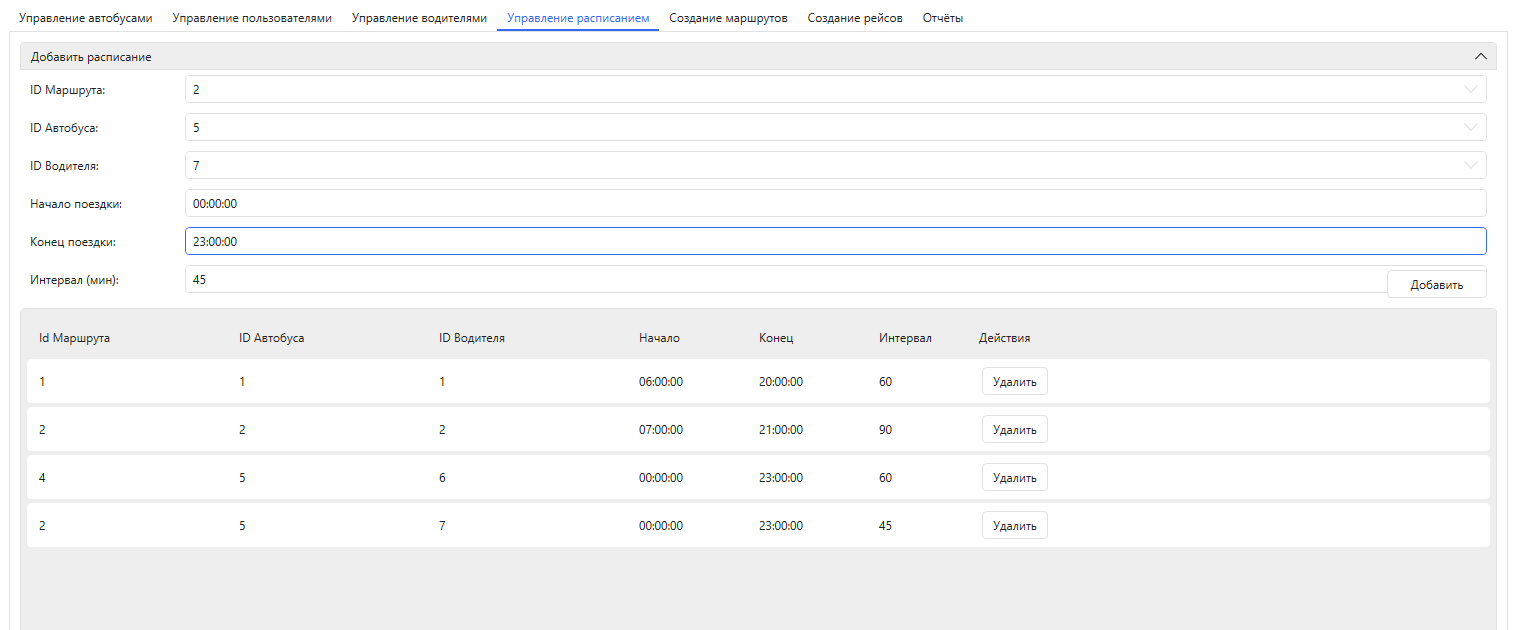


Рисунок 4.23 – Добавление нового расписания

## Управление маршрутами администратором

Администратор может добавлять и удалять маршруты, используемые на автовокзале (рис. 4.24).

Администратор может добавлять расписание заполнив данные в форме и нажав на кнопку «Добавить» (рис. 4.25)

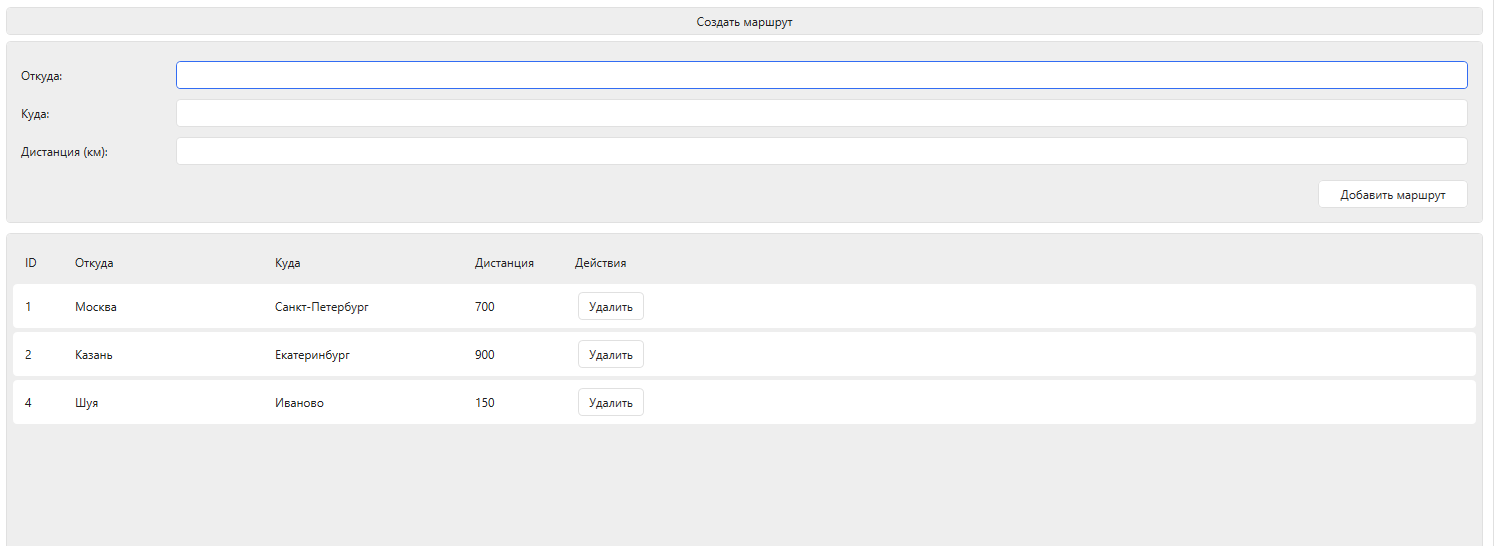


Рисунок 4.24 – Список маршрутов

Аналогично по нажатии на кнопку «Удалить» Администратор может удалить маршруты и подтвердить действие.

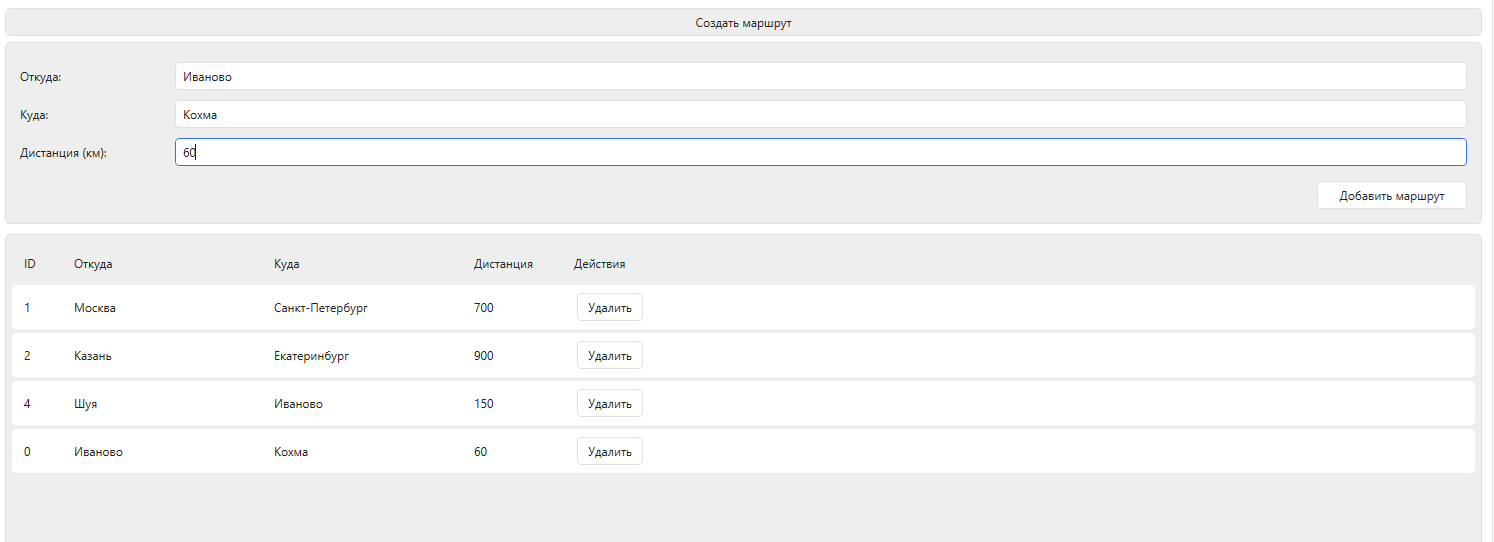


Рисунок 4.25 – Добавление нового маршрута

## Управление рейсами администратором

Администратор может добавлять рейсы на целый день и удалять рейсы, используемые на автовокзале (рис. 4.26).

Администратор может добавлять рейсы выбрав id расписания, указав цену и нажав на кнопку «Добавить» (рис. 4.27)

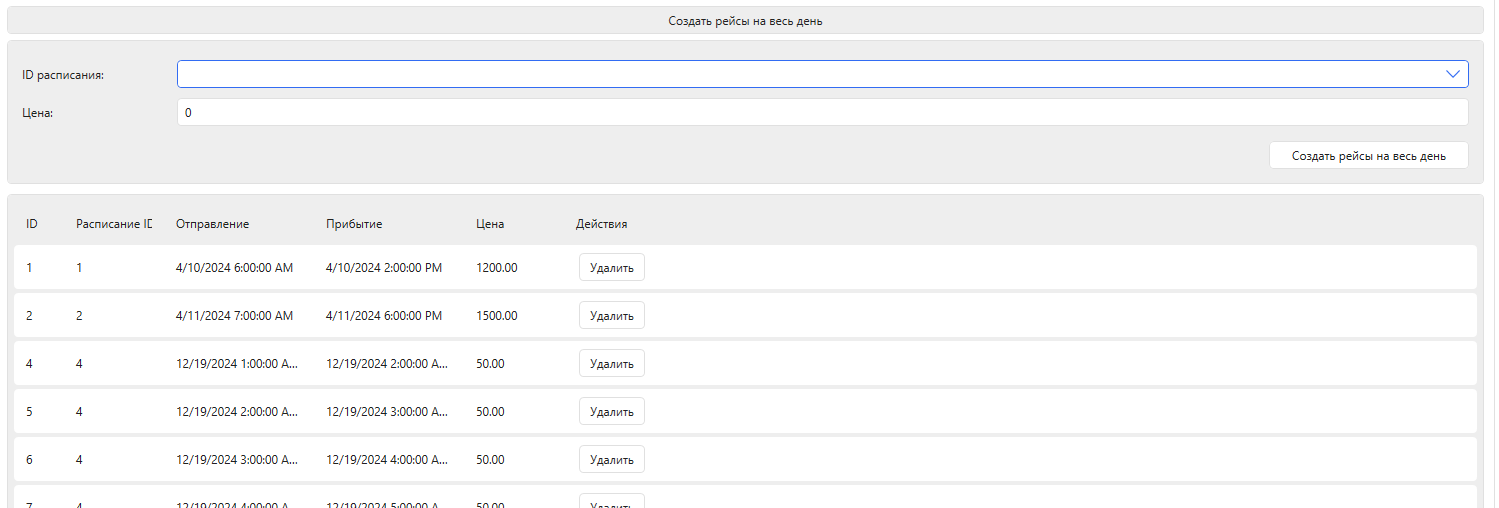


Рисунок 4.26 – Список рейсов

Аналогично по нажатии на кнопку «Удалить» Администратор может удалить рейсы и подтвердить действие.

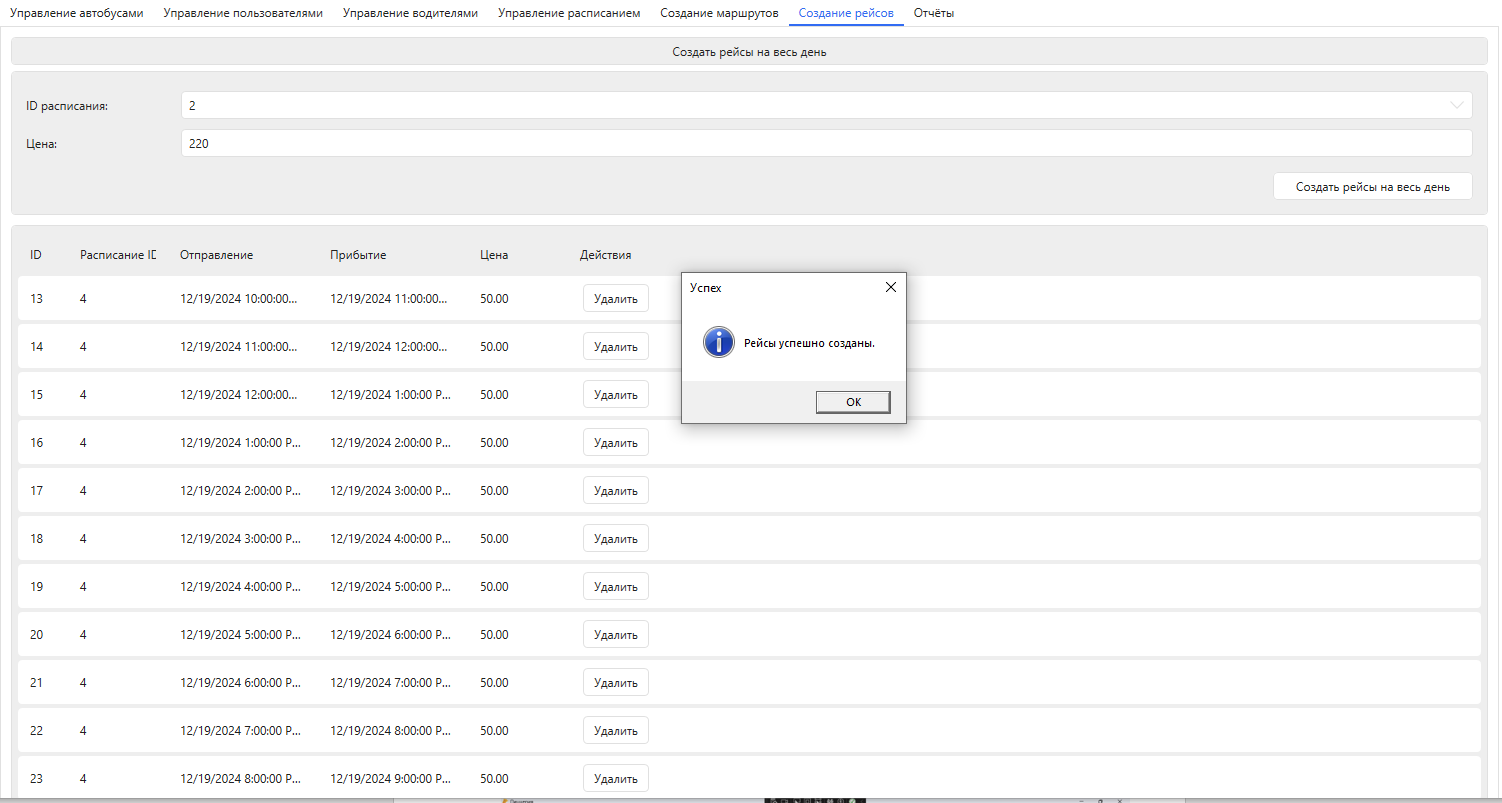


Рисунок 4.27 – Добавление новых рейсов

## Генерация отчетов менеджером

Для администратора доступна функция генерации отчетов (рис. 4.28). Администратор может выбрать нужный ему, по нажатии на кнопку выбранного отчёта он будет сгенерирован.

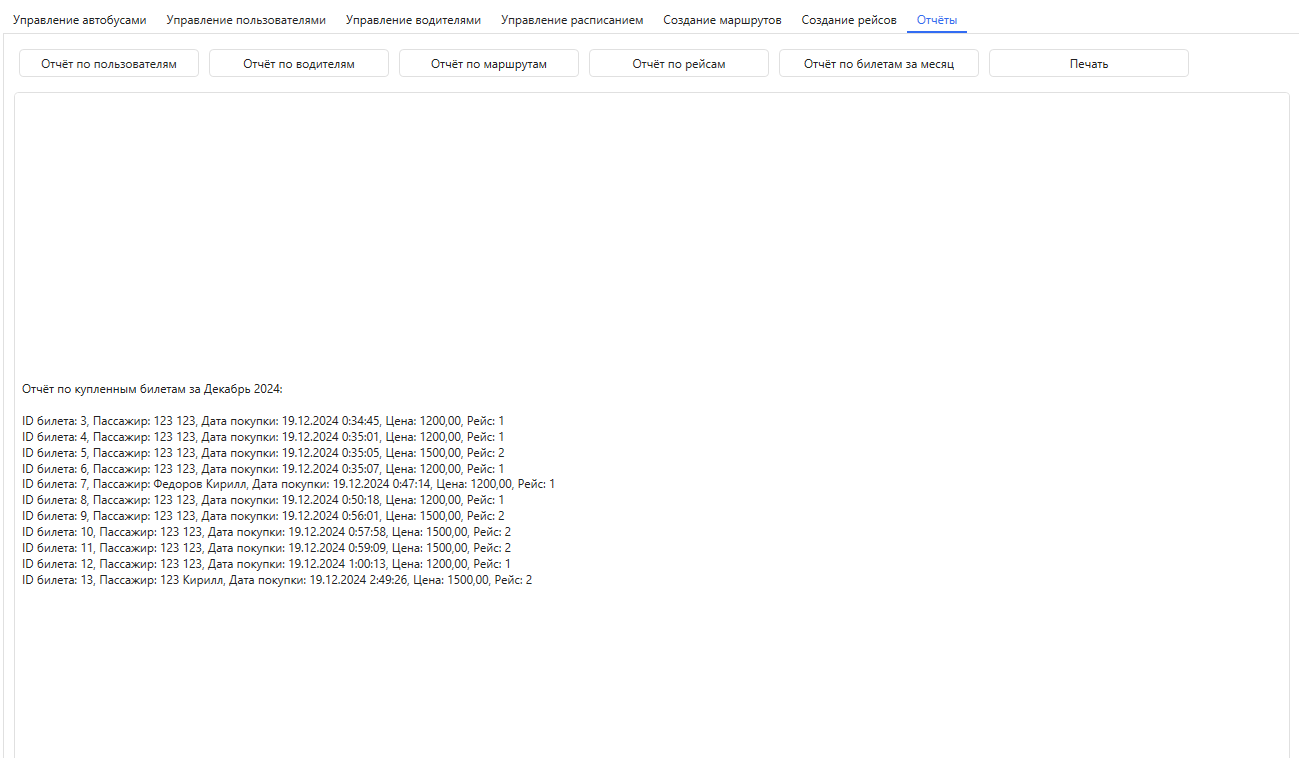


Рисунок 4.28 – Сгенерированный отчет

По нажатии на кнопку «Печать», откроется окно для печати файлов (рис. 4.29), где нужно будет указать принтер или выбрать сохранить в PDF. Отчеты сохраняются в формате PDF. Расширение pdf будет применено автоматически.

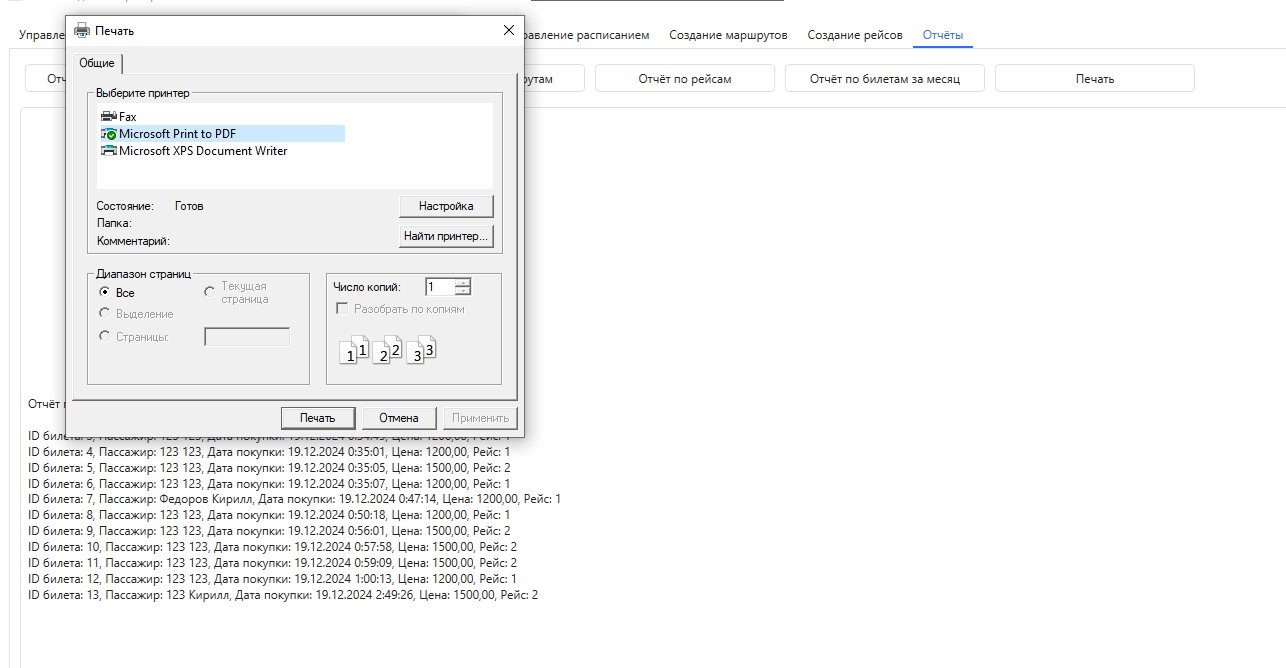


Рисунок 4.29 – Сохранение отчета

Пример просмотра сохраненного в формате PDF отчета представлен на рис. 4.30 ниже.

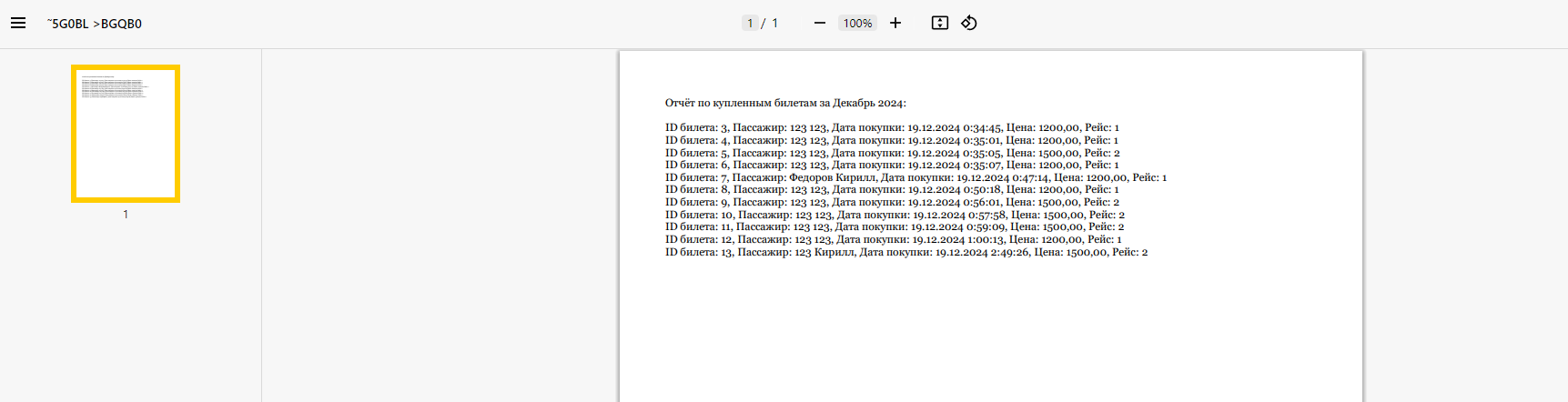


Рисунок 4.30 – Просмотр документа

# Мероприятия по подготовке к вводу системы в действие

Для ввода Системы в действие необходимо:

1. подготовить у Заказчика всё необходимое техническое обеспечение;
2. установить на сервер и клиентские ПК системное, базовое и прикладное ПО;
3. установить на сервер БД;
4. провести испытания системы.

## Мероприятия по приведению информации к виду, пригодному для обработки на ЭВМ

Для ввода системы в эксплуатацию необходимо выполнить первичное наполнение БД:

1. ввести в БД сведения о Сотрудниках организации
2. ввести в БД сведения о пиццах и ингредиентах.

## Мероприятия по обучению и проверке квалификации персонала

Специального обучения Пользователей работе с Системой не требуется. Достаточно показать сотрудникам, как выбирать и помечать доставленными заказы.

# Источники разработки

При разработке технорабочего проекта использовались следующие источники:

1. ГОСТ 34.003-90. Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения.
2. ГОСТ 34.201-89. Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем.
3. ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.
4. ГОСТ 34.602-89. Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.
5. РД 50-34.698.90. Методические указания. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов.
6. Documents associated with Unified Modeling Language (UML), v2.5.1. Release date: December 2017.- http://www.omg.org/spec/UML/2.5.1/PDF.