**ОГЛАВЛЕНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc193563446)

[**ГЛАВА 1. ПРЕДМЕТНАЯ ОБЛАСТЬ.** 6](#_Toc193563447)

[**1.1.** **Исходная постановка задачи** 6](#_Toc193563448)

[**1.2. Среда разработки** 7](#_Toc193563449)

[**1.3. Техническое задание** 8](#_Toc193563450)

[**1.4 Руководство по стилю** 14](#_Toc193563451)

[**1.5 UML-диаграмма** 15](#_Toc193563452)

[**1.7 Схема структуры** 20](#_Toc193563453)

[**2.1 Разработка логической модели данных** 21](#_Toc193563454)

[**2.2. Добавление таблиц к базе данных** 22](#_Toc193563455)

[**2.3. Просмотр связей на диаграмме базы данных** 23](#_Toc193563456)

[**2.4. Определение ограничений, налагаемые на данные** 23](#_Toc193563457)

[**2.5. Определение объема и типов данных** 25](#_Toc193563458)

[**2.6. Определение способа использования данных** 27](#_Toc193563459)

[**2.7. Разработка физической структуры данных** 28](#_Toc193563460)

[**2.7.1. Создание таблицы в БД** 29](#_Toc193563461)

[**2.7.2. Данные таблиц** 29](#_Toc193563462)

[**ГЛАВА 3. РАЗРАБОТКА ПРИЛОЖЕНИЯ** 34](#_Toc193563463)

[**3.1. Разработка пользовательского интерфейса** 34](#_Toc193563464)

[**3.2. Листинг программы** 40](#_Toc193563465)

[**3.4. Руководство пользователя** 45](#_Toc193563466)

[**ГЛАВА 4. РАЗРАБОТКА 1С** 55](#_Toc193563467)

[**4.1 Разработка подсистем** 55](#_Toc193563468)

[**4.2 Разработка справочников** 56](#_Toc193563469)

[**4.3 Разработка документов и регистров** 56](#_Toc193563470)

[**4.4 Разработка отчетов** 56](#_Toc193563471)

# **ВВЕДЕНИЕ**

База данных системы автостанции представляет собой организованную систему хранения и управления информацией, касающейся расписания движения транспорта, продажи билетов, учета пассажиров и управления автопарком. Эта база данных служит важным инструментом для автоматизации процессов, повышения эффективности работы автостанции и улучшения обслуживания пассажиров.

Автостанции, как и другие транспортные предприятия, нуждаются в защите и оптимизации своих данных. Они занимаются организацией движения транспорта, продажей билетов, управлением автопарком и предоставлением информации пассажирам. Для успешной работы автостанции необходима надежная система учета, которая обеспечит:

Организацию и структурирование информации: Базы данных позволяют аккуратно организовывать информацию о маршрутах, расписании, транспортных средствах, водителях, билетах и пассажирах. Это значительно облегчает поиск нужной информации и ускоряет процесс обслуживания.

Обеспечение целостности данных: Важно, чтобы данные о расписании, доступных билетах и состоянии автопарка были точными и актуальными. База данных помогает предотвратить дублирование записей и обеспечивает согласованность информации.

Совместное использование данных: Множество сотрудников может одновременно работать с базой данных, что упрощает процессы продажи билетов, управления расписанием и взаимодействия с пассажирами. Это повышает эффективность работы автостанции.

Повышение эффективности бизнес-процессов: С автоматизацией рутинных задач, таких как формирование расписания, продажа билетов и управление автопарком, автостанция может сосредоточиться на предоставлении качественных услуг пассажирам.

В рамках данной работы будет использоваться Microsoft SQL Server для создания и управления базой данных, а также Visual Studio для разработки приложения, которое будет взаимодействовать с этой базой. Такой подход поможет создать эффективную систему управления данными для автостанции.

Тестирование базы данных

Тестирование базы данных является ключевым этапом разработки, целью которого является обеспечение ее соответствия требованиям и корректной работы. Тестирование включает в себя несколько аспектов:

Функциональное тестирование: Проверяет правильность выполнения функций базы данных и их соответствие техническому заданию.

Тестирование производительности: Анализирует, насколько быстро и эффективно база данных может обрабатывать запросы, что особенно важно в условиях высокой нагрузки, например, в часы пик.

Тестирование нагрузки: Определяет, как база данных справляется с большим числом одновременных запросов, что критично для поддержания качества обслуживания пассажиров.

Тестирование безопасности: Выявляет уязвимости базы данных и обеспечивает защиту информации от несанкционированного доступа, что имеет первостепенное значение для защиты данных о пассажирах, водителях и транспортных средствах.

Тестирование позволяет выявить и устранить ошибки на ранних этапах, что значительно улучшает качество базы данных и снижает риск проблем в будущем.

Основные задачи, которые планируется решить с помощью новой системы

Автоматизация бизнес-процессов: Улучшение процессов формирования расписания, продажи билетов, управления автопарком и взаимодействия с пассажирами.

Повышение прозрачности и контроля: Создание единой системы учета всех операций, что позволит отслеживать каждый этап работы автостанции, включая движение транспорта и продажу билетов.

Улучшение коммуникации: Облегчение обмена информацией между различными отделами автостанции, а также с пассажирами и водителями.

Снижение издержек: Автоматизация рутинных операций снизит затраты на бумажную документацию, ошибки ручной обработки данных и другие потери.

Таким образом, разработка базы данных для системы автостанции — это не только ответ на современные вызовы цифровизации, но и важный шаг к улучшению качества обслуживания пассажиров, повышению эффективности работы и конкурентоспособности на рынке транспортных

# **ГЛАВА 1. ПРЕДМЕТНАЯ ОБЛАСТЬ.**

# **Исходная постановка задачи**

Разрабатывается приложение с целью более удобного предоставления отчета о прибыли и затратах автостанции, а также улучшить удобство пользования автостанции для граждан

Таблица 1

Система автостанция

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Поле | Тип | Размер | Описание |
| 1 | ID Marsryt | Числовой | 10 | Регистрационный номер маршрута в базе |
| 2 | Pynkt posadku | Текстовый | 25 | Названия посадки |
| 3 | Pynkt Vusadki | Текстовый | 25 | Название высадки |
| 4 | ID Zapravku | Числовой | 10 | Регистрационный номер заправки |
| 5 | ID Avtobus | Числовой | 10 | Регистрационный номер автобуса |
| 6 | Cena na 1 zapravky | Числовой | 10 | Цена за одну заправку |
| 7 | ID Bulet | Числовой | 10 | Регистрационный номер билета в базе |
| 8 | Cena | Числовой | 10 | Цена билета |
| 9 | ID Marsryt | Числовой | 10 | Регистрационный номер маршрута |
| 10 | ID Proezd | Числовой | 10 | Регистрационный номер проезда |
| 11 | Polychen | Дата |  | Дата получение билета |
| 12 | FIO | Текстовый | 25 | ФИО Гражданина с проездным |

**1.2. Среда разработки**

**Visual Studio** — это одна из самых популярных интегрированных сред разработки приложение, разработанная Microsoft. Она предлагает широкий набор инструментов и функций, которые делают процесс разработки программного обеспечения более эффективным и удобным.

Почему я выбрал Visual? В это программе есть плюсы которые достаточно сильно улучшают и упрощают процесс разработки вот собственно и они:

* + Многофункциональность и поддержка языков
  + Мощные инструменты разработки
  + Интеграция с системами контроля версий
  + Гибкость и расширяемость
  + Простота в использовании

**SQL Server Management Studio** — это мощная интегрированная среда для работы с Microsoft SQL Server. Она предоставляет обширный набор инструментов для управления базами данных и позволяет выполнять множество задач, связанных с разработкой и администрированием.

Так же и с Sql server Management Studio, вот пункты которые я считаю наиболее привлекательными которые я заметил при выборе этого приложения для разработки:

* Удобный интерфейс
* Поддержка различных версий SQL Server
* Построение и визуализация схем баз данных
* Управление безопасностью и доступом
* Импорт и экспорт данных
* Большое сообщество и поддержка

**1.3. Техническое задание**

УТВЕРЖДАЮ УТВЕРЖДАЮ

Директор ГБПОУ «ВВК» Директор филиала

\_. \_ . \_ \_\_\_\_\_\_\_/Арсеньев А.В./ \_. \_. \_ \_\_\_\_\_\_/Михайлов А.С./

СОГЛАСОВАНО

Зам. директора ГБПОУ «ВВК»

\_ . \_ . \_ \_\_\_\_\_\_\_\_\_\_\_\_/ Бордодымова Т.В./

Система автостанций

Техническое задание

ЛИСТ УТВЕРЖДЕНИЯ

02068120.508100.027-02 2А 01.М-ЛУ

Листов 6

Представители разработчика

Начальник отдела ИУ

\_ . \_ . \_ \_\_\_\_\_\_\_/Яковлева Ж.С./

Руководитель разработки

Ведущий программист

\_ . \_ . \_ \_\_\_\_\_\_\_/Козлов Р.А./

2024

УТВЕРЖДЕНО Приложение № 1 02068120.508100.027-022А 01.М-ЛУ к Договору № 2 от 02.04.06

Система автостанций

Техническое задание

02068120.508100.027-02 2А 01.М

Листов 6

2024

**Введение**

Система автостанции предназначена для помощи пассажирам в поиске информации о расписании движения транспорта, маршрутах, наличии билетов и услугах, предоставляемых автостанцией. В программе можно будет посмотреть, где находится автостанция, какие маршруты доступны и какие услуги предоставляются.

**Основание для разработки**

Документы: План разработки

Организация: Система автостанций. Документ одобрен 15.01.2025

**Назначение разработки**

Функциональное назначение программы заключается в хранении и предоставлении информации о расписании, маршрутах и услугах автостанции пользователям. Приложение должно эксплуатироваться заказчиком (организацией, управляющей автостанцией).

**Требование к программе или программному изделию**

**Требования к функциональным характеристикам**

Программа должна содержать в себе следующие функции:

* Регистрация пользователей при необходимости (если не получается зарегистрироваться пользователю)
* Редактирование базы данных
* Главное меню со всеми кнопками для выбора любой категории и её редактирование
* Возможность переключать, добавлять, удалять, сохранять записи в базу данных через приложение
* Предоставление поддержки в случае затруднений со стороны пользователя
* Связь администратора с разработчиком в случае неисправности приложения
* Регистрация в приложении
* Главное меню с кнопками для выбора нужной категории
* Переключатели для просмотра нескольких записей

**Требование к надежности**

Надежное функционирование и защита данных уже включена в приложение, но заказчик должен обеспечить:

* Бесперебойного питания и сервера
* Использование лицензионного программного обеспечения
* Установка всех последних обновлений

**Условие эксплуатации**

* Использовать его только в законных целях.
* Нести ответственность за публикуемый контент.
* Не нарушать права интеллектуальной собственности.
* Принимать приложение "как есть" без гарантий.
* Соблюдать изменения в условиях использования.
* Принять возможность прекращения доступа при нарушении условий

**Требования к составу и параметрам технических средств**

* Сервер БД (SQL Server)
* Процессор ARM64 или x64; Рекомендуется использовать четырехъядерный процессор или лучше. Процессоры ARM 32 не поддерживаются.
* Не менее 4 ГБ ОЗУ. На используемые ресурсы влияет множество факторов; для стандартных профессиональных решений рекомендуется 16 ГБ ОЗУ.
* Место на жестком диске: от 850 МБ до 5 ГБ свободного места в зависимости от кол-ва заполненной информации. Для повышения производительности рекомендуется устанавливать Windows и Comics Store на твердотельном накопителе (SSD).
* Операционная система: Windows 7,10.

**Требования к информационной и программной совместимости**

* Проектирование базы данных должно быть выполнено вместе с разработкой приложения.
* Приложение 1 разрабатывается в интегрированной среде разработки 1с на языке программирования 1с
* Приложение 2 разрабатывается в интегрированной среде разработки Visual Studio на языке программирования C#

**Требования к маркировке и упаковке**

Маркировка содержит имя разработчика, дата реализации приложения, электронную почту и номер телефона разработчика

Приложение после разработки экспортируется на внешний накопитель, или на облачное хранилище

**Требование к транспортировке и хранению**

В случае необходимости передачи внешнего накопителя на расстоянии, требуется транспортировка почтой. В случае передачи облачным хранилищем, сохранить его.

Сохранить его на компьютер для личного пользования и освоения приложения, или загрузить на сервер и запустить приложение в сети интернет.

**Требования к программной документации**

Состав документации, который должен быть составлен:

1. Руководство оператора
2. Руководство пользователя
3. Руководство системного администратора
4. Техническое задание

**Технико-экономические показатели**

Экономическая эффективность не рассчитывается

**Стадии и этапы разработки**

Разработка технического задания;

Разработка документации;

Проектирование БД и приложения;

Реализация;

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждение настоящего технического задания.

На стадии разработки документации должен быть выполнен этап разработки проектной документации.

На стадии проектировании БД и приложения должен быть выполнен этап создания, кодирования, разработки БД и приложения.

На стадии реализации должен быть выполнен этап выпуска и реализации приложения на свет.

**Порядок контроля и приемки**

Испытание программы должно проводиться рядом методик, согласованные разработчиком с заказчиком (так же соблюдение требования к надежности)

При прохождении испытательного срока, Заказчик совместно с разработчиком подписывает акт приемки приложения в эксплуатацию

**Приложение**

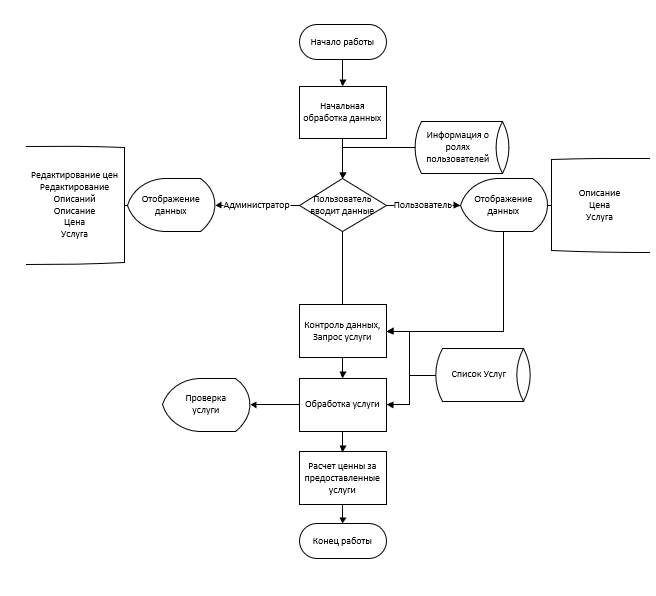


Рис. 1 Алгоритм

## **1.4 Руководство по стилю**

**Общие требования**

Не допускать орфографические и грамматические ошибки. Размер форм, элементов на форме должны быть одинаковы, они не должны отличаться.

**Использование логотипа**

Логотип используется в каждой форме справа в верхнем углу экрана, а также на главной странице

**Шрифт**

Шрифт будет использоваться Times New Roman, диапазон примерных размеров от 14 до 20.

**Основные цвета**

|  |  |
| --- | --- |
| #4472C4 | #C9C9C9 |
|  |  |

**Дополнительные**

|  |  |
| --- | --- |
| #595959 | #8EAADB |
|  |  |

**Для текста**

|  |  |
| --- | --- |
| #FFFFFF | #000000 |
|  |  |

## **1.5 UML-диаграмма**

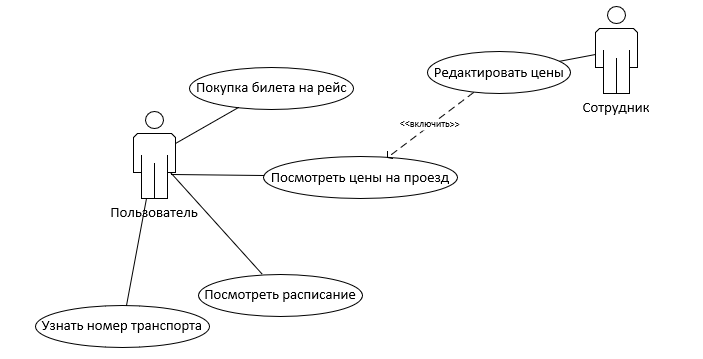
****

Рис. 2 Диаграмма вариантов использования

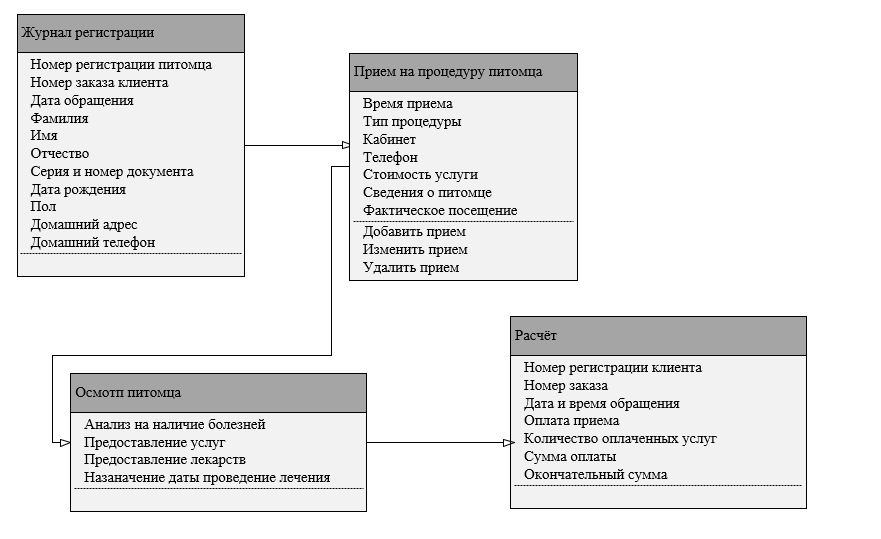


Рис. 3 Диаграмма классов

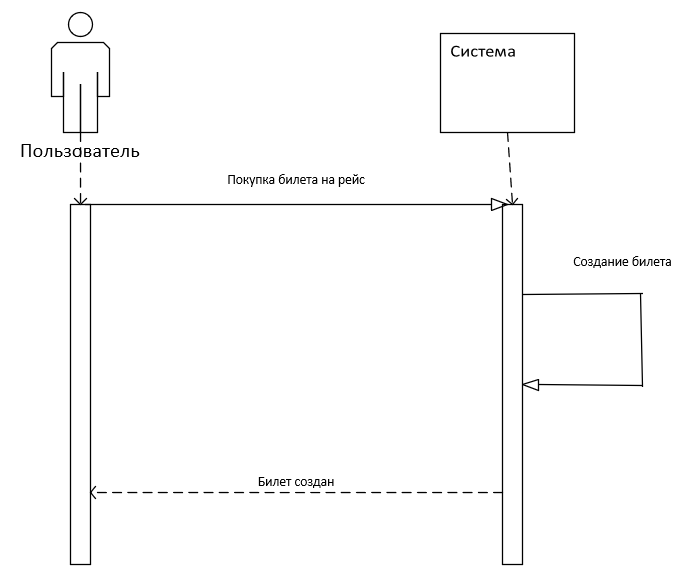


Рис. 4 Диаграмма поведения

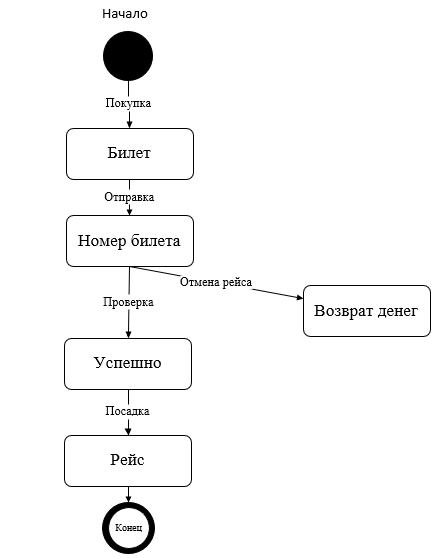


Рис. 5 Диаграмма состояния

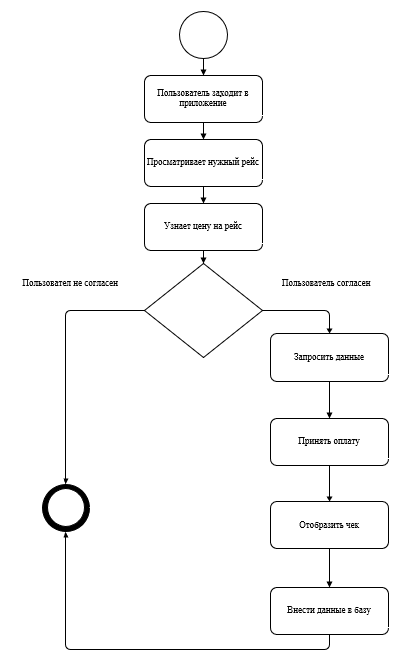


Рис. 6 Диаграмма деятельности

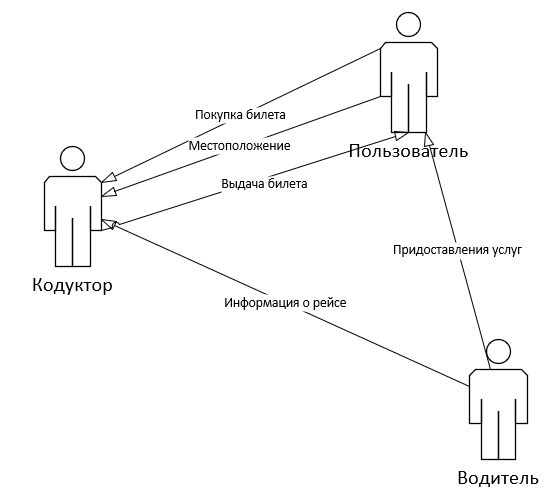


Рис. 7 Диаграмма взаимодействия

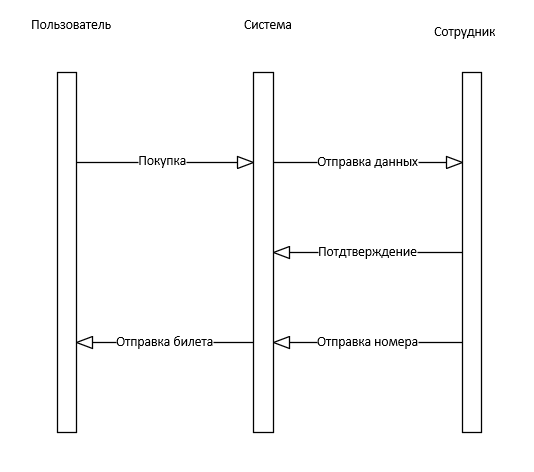


Рис. 8 Диаграмма последовательности

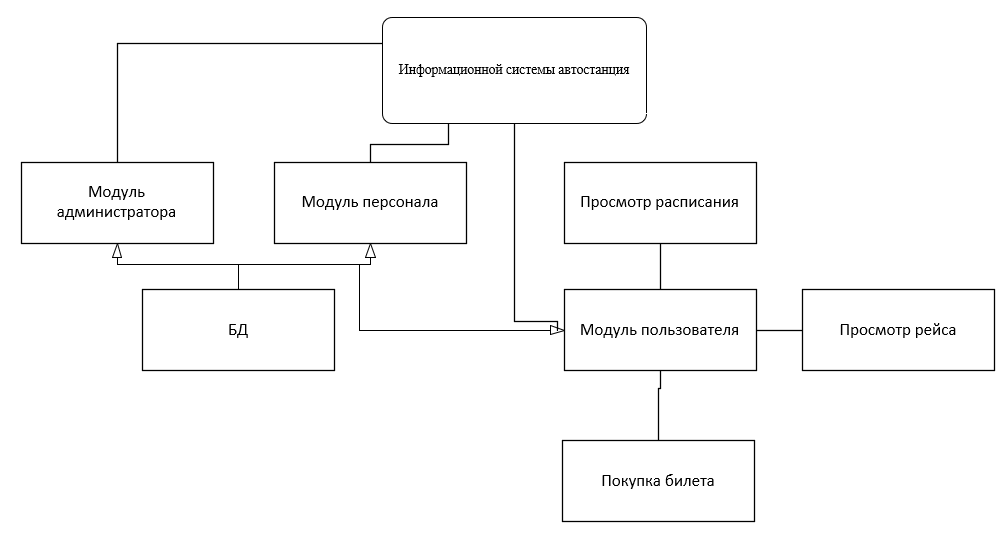


Рис. 9 Диаграмма компонентов

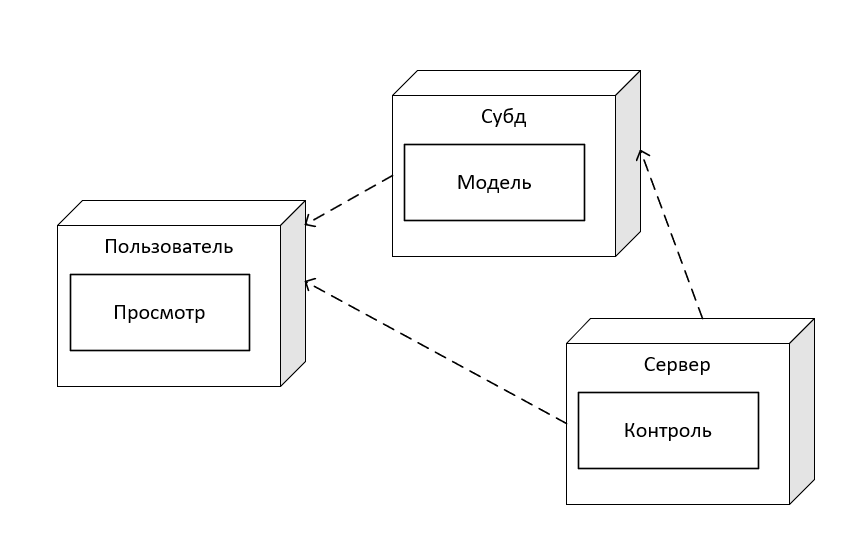


Рис. 10 Диаграмма развертывания

**1.6 Wireframe**

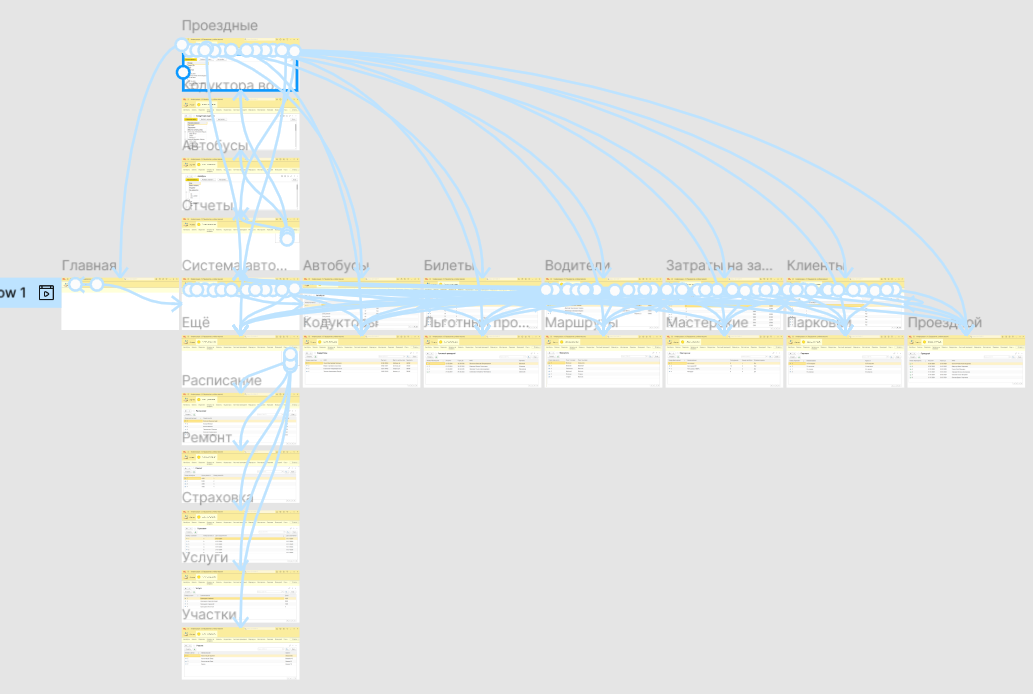
****

Рис. 11 Warframe

Cсылка на проект:

<https://www.figma.com/design/7RhA4ZiTrl7PZ8xvizqnci/Figma-basics?node-id=1669-162202&p=f&t=oyLrzWgy28fr4Wkp-0>

# **1.7 Схема структуры**

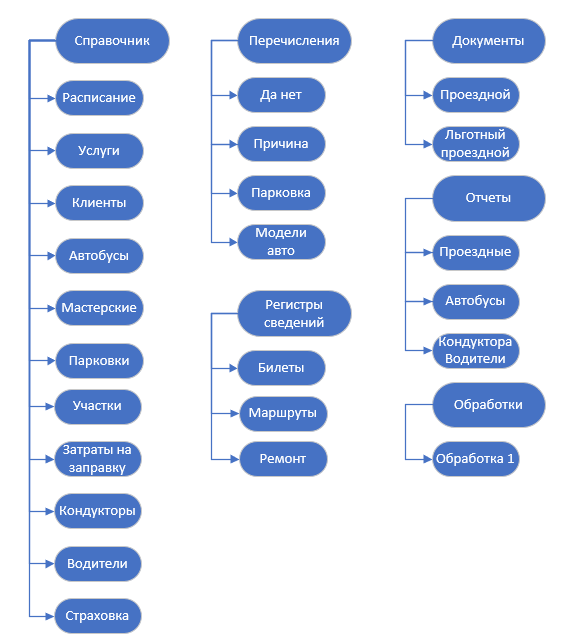
****

Рис. 12 Комплексная схема**ГЛАВА 2. РАЗРАБОТКА БАЗЗ ДАННЫХ**

**2.1 Разработка логической модели данных**

Сущность «Затраты на заправку»: Номер заправки, Номер автобуса, Цена за одну запчасть, Цена за месяц.

Сущность «Услуги»: Номер услуги, Наименование, Цены.

Сущность «Проездной»: Номер Проездного, Получен, Годен до, ФИО.

Сущность «Парковка»: Номер Парковки, Наименование, Адрес.

Сущность «Участок»: Номер участка, Наименование, Адрес.

Сущность «Водитель»: Номер водителя, ФИО, Паспорт, Место жительства, Зарплата.

Сущность «Страховка»: Номер страховки, Номер автобуса, Дата оформления, Дата окончания.

Сущность «Автобус»: Номер автобуса, Модель, Вмещает, На ремонте.

Сущность «Льготный проездной»: Номер Лпроездного, Получен, Годен до, ФИО.

Сущность «Клиенты»: Номер Клиента, ФИО, Наличие ремонта.

Сущность «Мастерские»: Номер Мастерской, Наименование, Сотрудник, Номер автобуса, Отремонтированные.

Сущность «Маршрут»: Номер маршрута, Пункт посадки, Пункт высадки.

Сущность «Ремонт»: Номер автобуса, Цена ремонта.

Сущность «Расписание»: Номер Расписания, Номер пункта, Время.

Сущность «Кондуктор»: Номер кондуктора, ФИО, Паспорт, Место жительства, Зарплата.

**Уникальный идентификатор сущностей.**

Сущность «Затраты на заправку»: Номер заправки.

Сущность «Услуги»: Номер услуги.

Сущность «Проездной»: Номер Проездной.

Сущность «Парковка»: Номер Парковки.

Сущность «Участок»: Номер участка.

Сущность «Водитель»: Номер водителя.

Сущность «Страховка»: Номер страховки.

Сущность «Автобус»: Номер автобуса.

Сущность «Льготного проездного»: Номер Лпроездной.

Сущность «Клиенты»: Номер Клиента.

Сущность «Мастера»: Номер Мастера.

Сущность «Маршрут»: Номер маршрута.

Сущность «Ремонт»: Номер автобуса.

Сущность «Расписание»: Номер расписания.

Сущность «Кондуктор»: Номер кондуктора.

## **2.2. Добавление таблиц к базе данных**

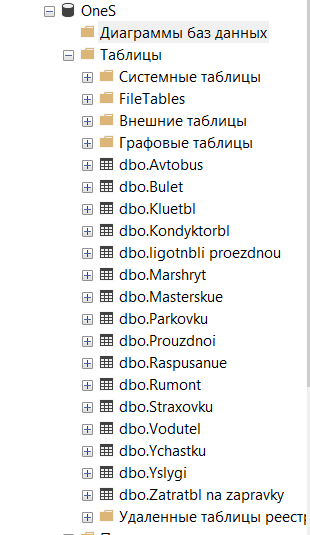


Рис. 13 Список таблиц

## **2.3. Просмотр связей на диаграмме базы данных**



Рис. 14 Диаграмма базы данных

## **2.4. Определение ограничений, налагаемые на данные**

Таблица «Затраты на заправки»

Номер заправки: int, уникальное значение, не может быть null.

Номер автобуса: int, уникальное значение, не может быть null.

Цена за одну заправки: int цена заправки.

Цена за месяц: int цена за месяц заправки.

Таблица «Услуги»

Номер услуги: int, уникальное значение, не может быть null.

Наименование: varchar(255).

Цена: int стоимость услуги.

Таблица «Проездной»

Номер Проездной: int, уникальное значение, не может быть null.

Получен: varchar(255).

Годен до: varchar(255).

ФИО: varchar(255).

Таблица «Парковка»

Номер парковки: int, уникальное значение, не может быть null.

Наименование: varchar(255).

Адрес: varchar(255).

Таблица «Участок»

Номер участка: int, уникальное значение, не может быть null.

Наименование: varchar(255).

Адрес: varchar(255).

Таблица «Водитель»

Номер водителя: int, уникальное значение, не может быть null.

ФИО: varchar(255).

Паспорт: varchar(255).

Место жительства: varchar(255).

Зарплата: int.

Таблица «Страховка»

Номер страховки: int, уникальное значение, не может быть null.

Номер автобуса: int, уникальное значение, не может быть null.

Дата оформления: varchar(255).

Дата окончания: varchar(255).

Таблица «Автобус»

Номер автобуса: int, уникальное значение, не может быть null.

Модель: varchar(255).

Вместимость: int.

На ремонте: varchar(255).

Таблица «Мастерские»

Номер Мастерские: int, уникальное значение, не может быть null.

Наименование: varchar(255).

Сотрудников: int, уникальное значение, не может быть null.

Номер автобуса: int, уникальное значение, не может быть null.

Отремонтированные: varchar(255).

Таблица «Маршрут»

Номер маршрута: int, уникальное значение, не может быть null.

Пункт посадки: varchar(255).

Пункт высадки: varchar(255).

Таблица «Ремонт»

Номер автобуса: int, уникальное значение, не может быть null.

Цена ремонта: float или decimal.

Таблица «Расписание»

Номер: int, уникальное значение, не может быть null.

Номер пункта: int, уникальное значение, не может быть null.

Время: varchar(255).

Таблица «Кондуктор»

Номер кондуктора: int, уникальное значение, не может быть null.

ФИО: varchar(255).

Паспорт: varchar(255).

Место жительства: varchar(255).

Зарплата: int.

**2.5. Определение объема и типов данных**

Для разработки ПО «Ветеринарной аптеки» следует выделить след категории данных. Это позволить организовать и структурировать информацию о товарах, услугах и цене для нуждающихся в этом питомцах. Категории данных включают:

* Автобусы
* Билеты
* Водители
* Затраты на заправку
* Клиенты
* Кондукторы
* Льготный проездной
* Маршруты
* Мастерские
* Парковки
* Проездной
* Расписание
* Ремонт
* Страховка
* Услуги
* Участки

Описание таблиц

Таблица 2

|  |  |
| --- | --- |
| **Название** | **Описание** |
| Автобусы | Сущность «Затраты на заправку»: Номер заправки. |
| Билеты | Сущность «Услуги»: Номер услуги. |
| Водители | Сущность «Проездной»: Номер Проездной. |
| Затраты на заправку | Сущность «Парковка»: Номер Парковки. |
| Клиенты | Сущность «Участок»: Номер участка. |
| Кондукторы | Сущность «Водитель»: Номер водителя. |
| Льготный проездной | Сущность «Страховка»: Номер страховки. |
| Маршруты | Сущность «Автобус»: Номер автобуса. |
| Мастерские | Сущность «Льготного проездного»: Номер Лпроездной. |
| Парковки | Сущность «Клиенты»: Номер Клиента. |
| Проездной | Сущность «Мастера»: Номер Мастера. |
| Расписание | Сущность «Маршрут»: Номер маршрута. |
| Ремонт | Сущность «Ремонт»: Номер автобуса. |
| Страховка | Сущность «Расписание»: Номер расписания. |
| Услуги | Сущность «Кондуктор»: Номер кондуктора. |
| Участки | Сущность «Затраты на заправку»: Номер заправки. |

Таблица 3

Объем данных таблиц

|  |  |
| --- | --- |
| **Название таблицы** | **Объем данных** |
| Автобусы | 20 |
| Билеты | 20 |
| Водители | 20 |
| Затраты на заправку | 20 |
| Клиенты | 20 |
| Кондукторы | 20 |
| Льготный проездной | 20 |
| Маршруты | 20 |
| Мастерские | 20 |
| Парковки | 20 |

Продолжение. Таблица 3

|  |  |
| --- | --- |
| Проездной | 20 |
| Расписание | 20 |
| Ремонт | 20 |
| Страховка | 20 |
| Услуги | 20 |

## **2.6. Определение способа использования данных**

Для разработки программного обеспечения для учета и управления системой автостанции важно определить роли и задачи двух категорий пользователей: Администратор и Пользователь. Ниже приведены описания этих пользователей и их задач.

Администратор

Администратор имеет полный доступ ко всем функциям системы, отвечает за управление пользователями, настройку платформы и контроль работы автостанции.

Основные задачи:

Управление пользователями:

Создание и редактирование аккаунтов пользователей.

Назначение прав доступа (например, доступ к расписанию, управлению автопарком и т.д.).

Настройка системы:

Добавление и редактирование маршрутов, расписания движения транспорта.

Управление информацией о транспортных средствах (автобусы, микроавтобусы и т.д.).

Мониторинг и отчетность:

Анализ данных о продаже билетов, загруженности маршрутов и использовании автопарка.

Формирование отчетов по запросам руководства.

Обслуживание данных:

Резервное копирование данных для предотвращения потери информации.

Обеспечение безопасности данных (защита от несанкционированного доступа).

Пользователь

Пользователь — это пассажир или лицо, заинтересованное в получении информации о расписании, маршрутах и услугах автостанции.

Основные задачи:

Просмотр информации:

Ознакомление с расписанием движения транспорта, маршрутами и доступными услугами.

Поиск информации о ближайших рейсах и стоимости билетов.

Оформление заказов:

Покупка билетов через систему (добавление в корзину и оформление заказа).

Получение электронных билетов или кодов для посадки.

## **2.7. Разработка физической структуры данных**

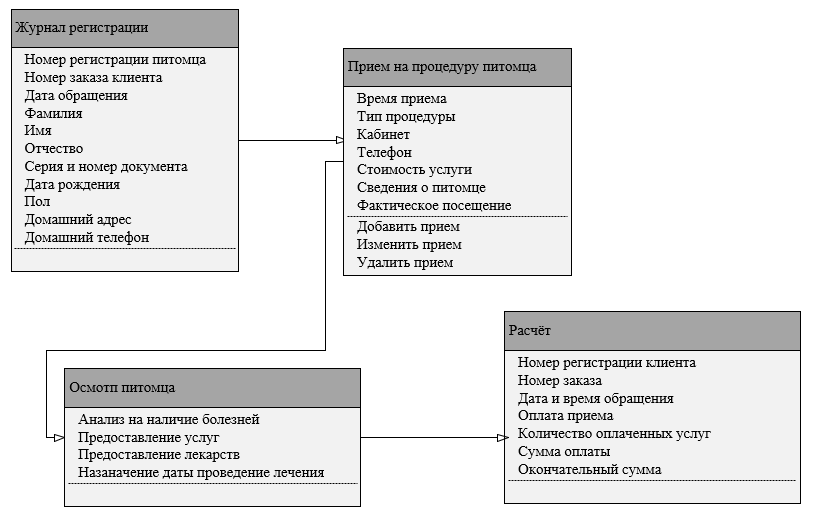


Рис. 15 Физическая структура данных

## **2.7.1. Создание таблицы в БД**

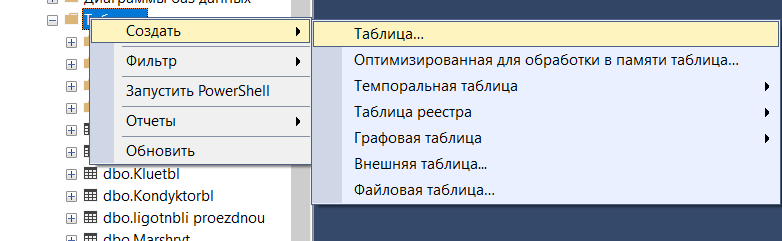


Рис. 16 Создание таблицы

## **2.7.2. Данные таблиц**

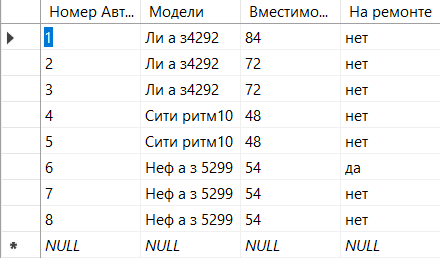


Рис. 17 Таблица «Автобус»

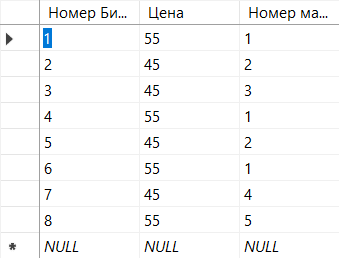


Рис. 18 Таблица «Билеты»

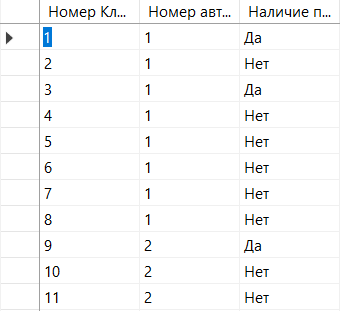


Рис. 19 Таблица «Клиенты»

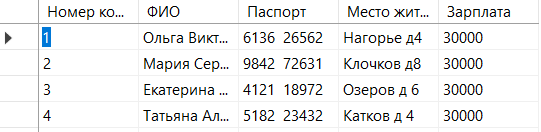


Рис. 20 Таблица «Кондукторы»

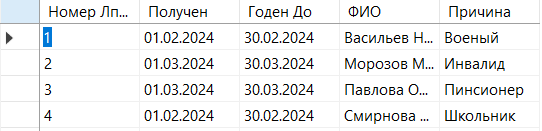


Рис. 21 Таблица «Льготный проездной»

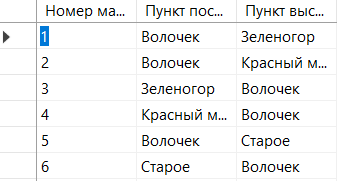


Рис. 22 Таблица «Маршрут»

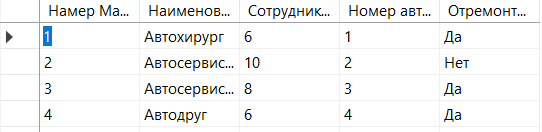


Рис. 23 Таблица «Мастерские»

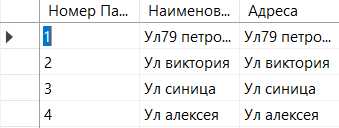


Рис. 24 Таблица «Парковки»

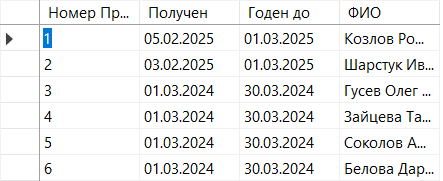


Рис. 25 Таблица «Проездной»

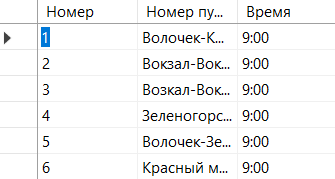


Рис. 26 Таблица «Расписание»

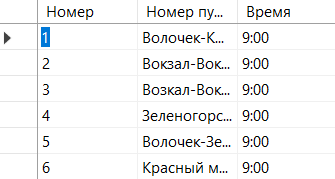


Рис. 27 Таблица «Ремонт»

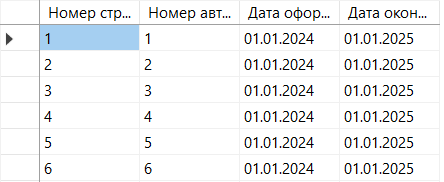


Рис. 28 Таблица «Страховки»

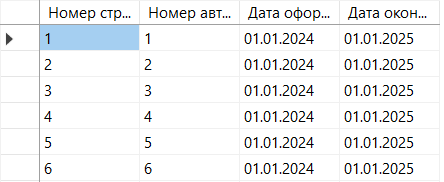


Рис. 29 Таблица «Водитель»

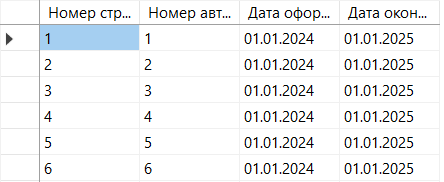


Рис. 30 Таблица «Участки»

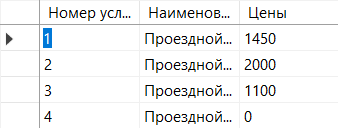


Рис. 31 Таблица «Услуги»

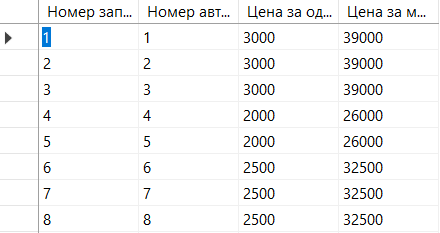


Рис. 32 Таблица «Затраты на заправку»

# **ГЛАВА 3. РАЗРАБОТКА ПРИЛОЖЕНИЯ**

## **3.1. Разработка пользовательского интерфейса**

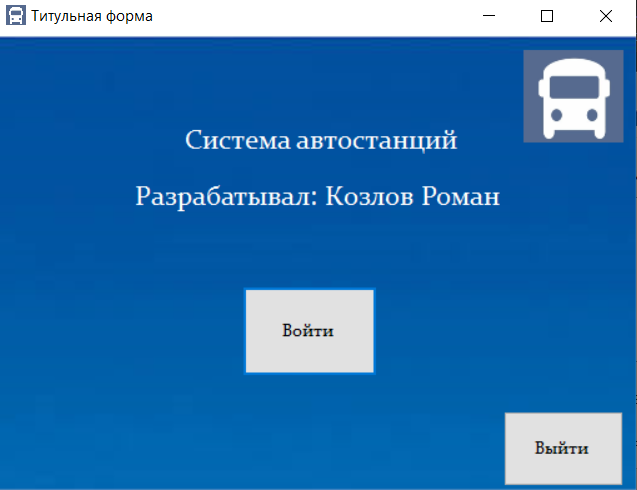


Рис. 33 Титульная форма

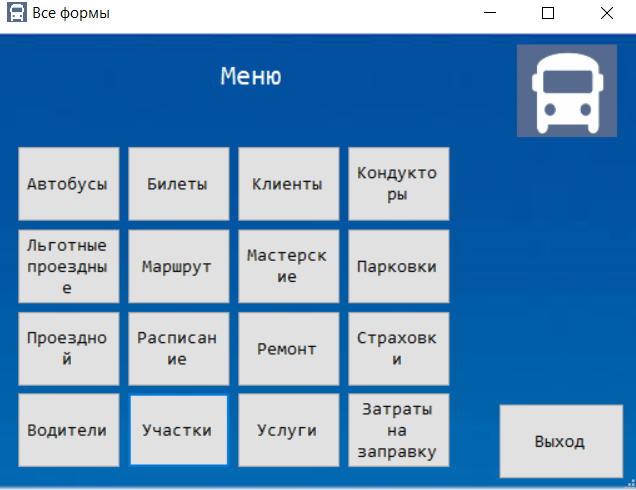


Рис. 34 Форма Всех форм

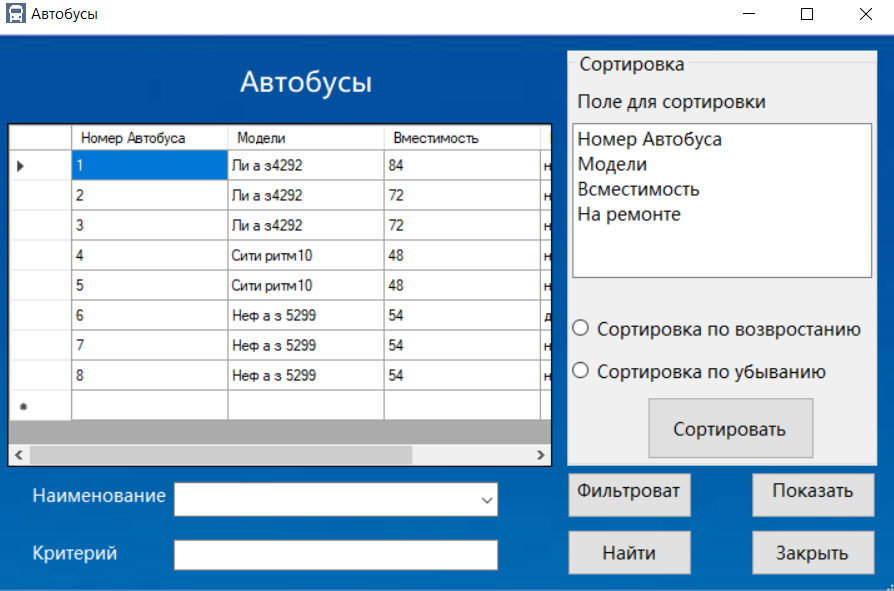


Рис. 35 Форма Автобус

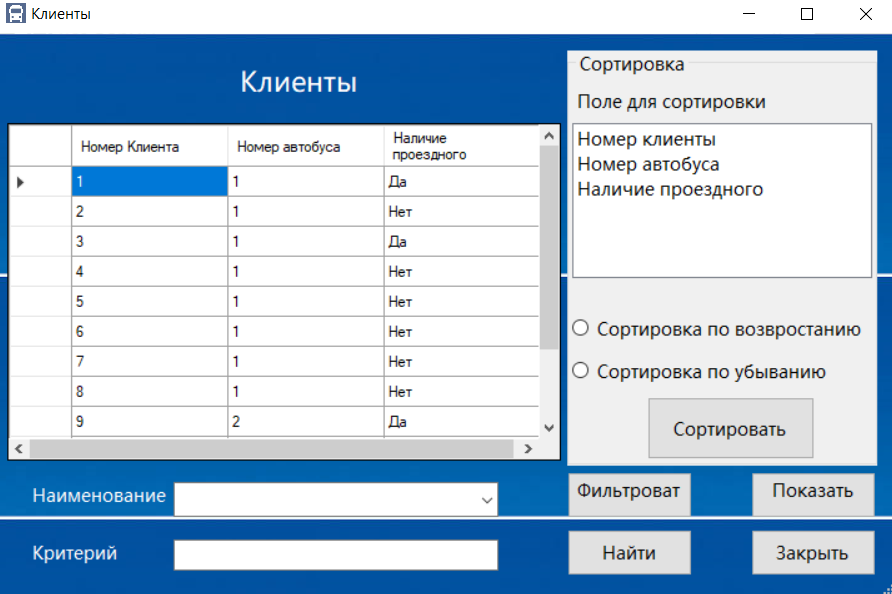


Рис. 36 Форма Клиенты

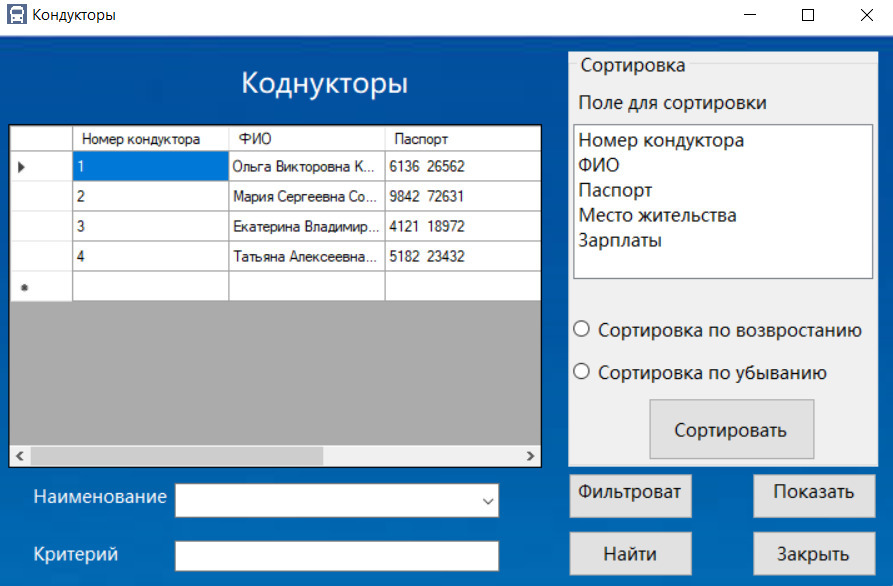


Рис. 37 Форма Кондукторы

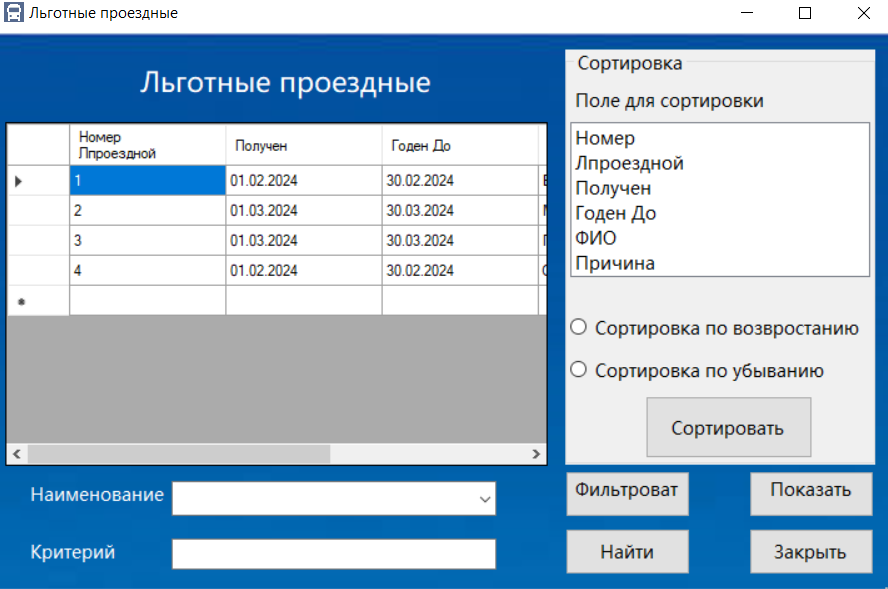


Рис. 38 Форма Льготные проездные

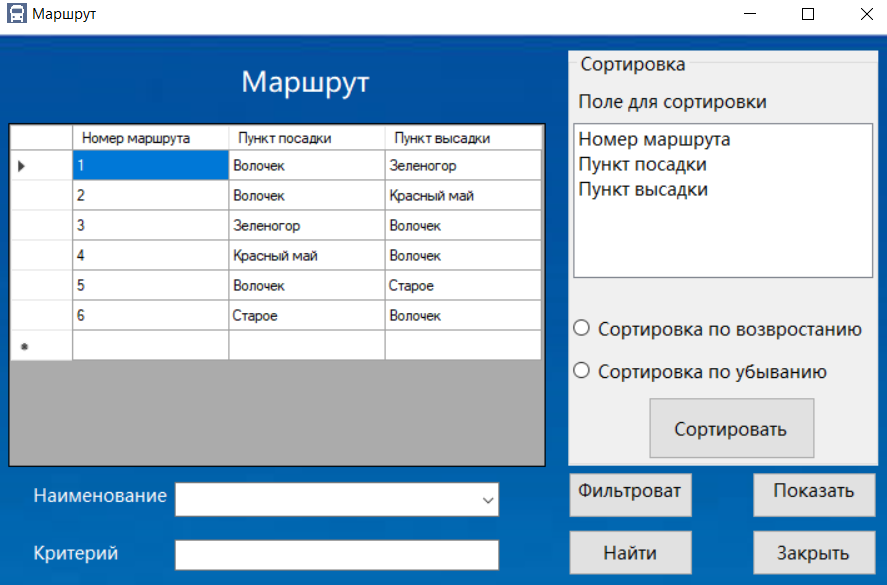


Рис. 39 Форма Маршруты

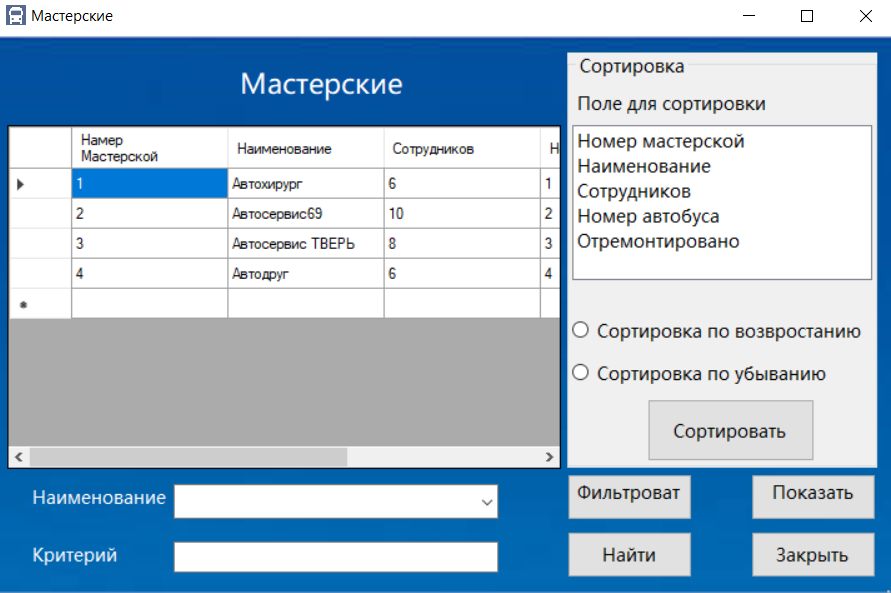


Рис. 40 Форма Мастерские

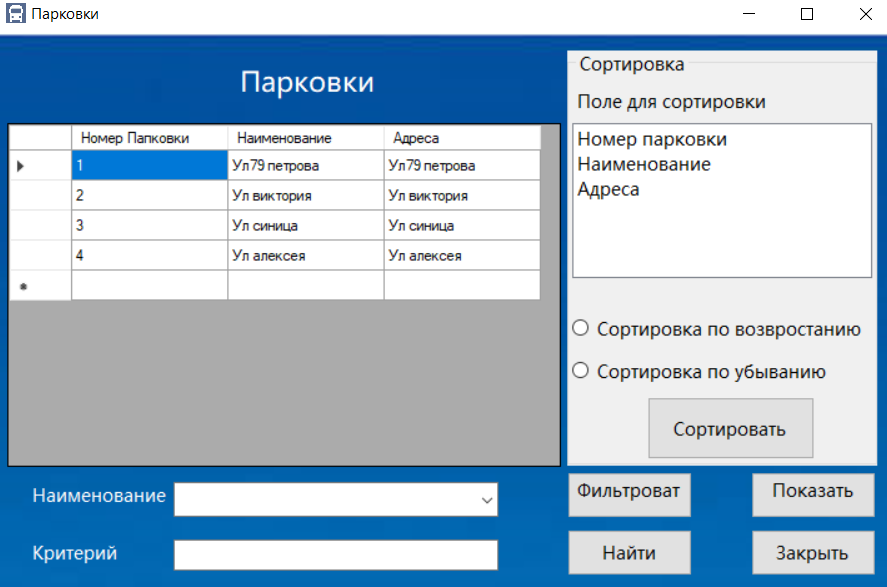


Рис. 41 Форма Парковки

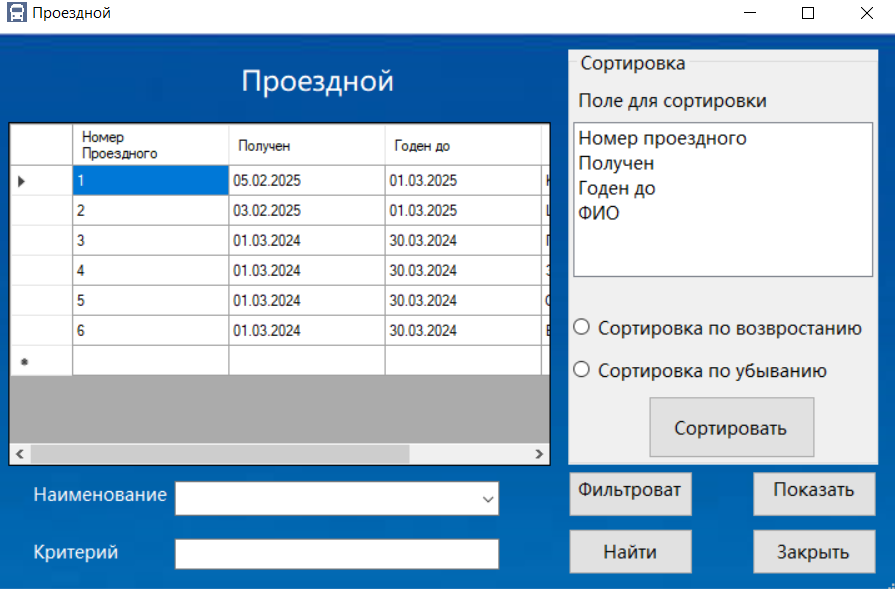


Рис. 42 Форма Проездной

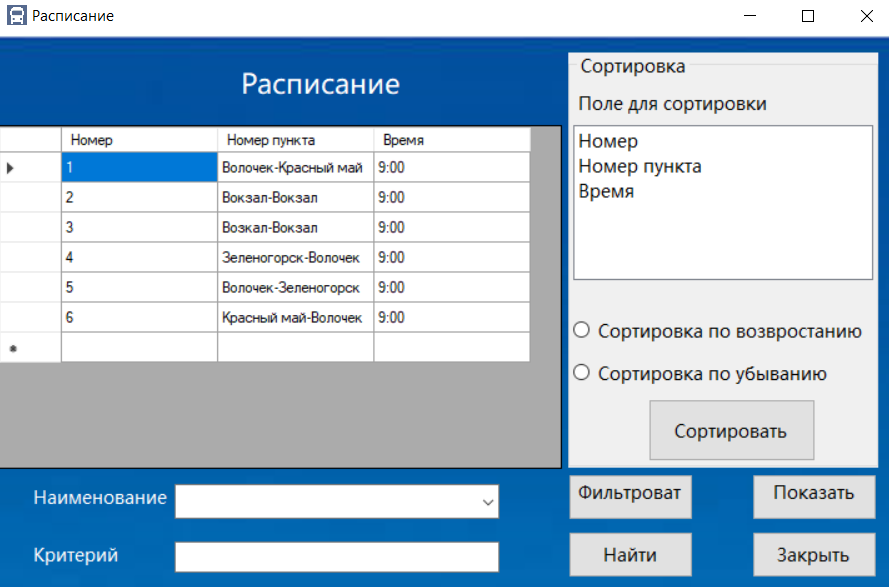


Рис. 43 Форма Расписание

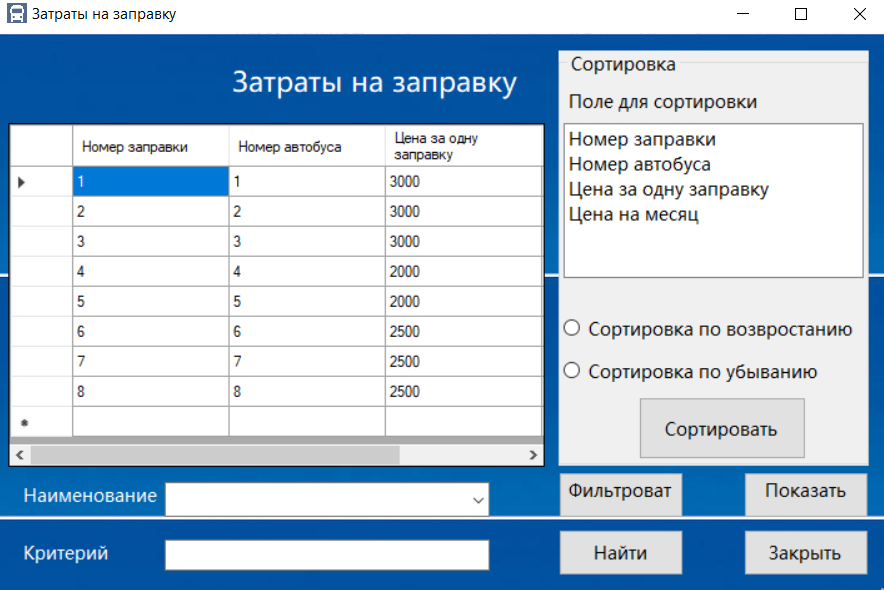


Рис. 44 Форма Затраты на заправку

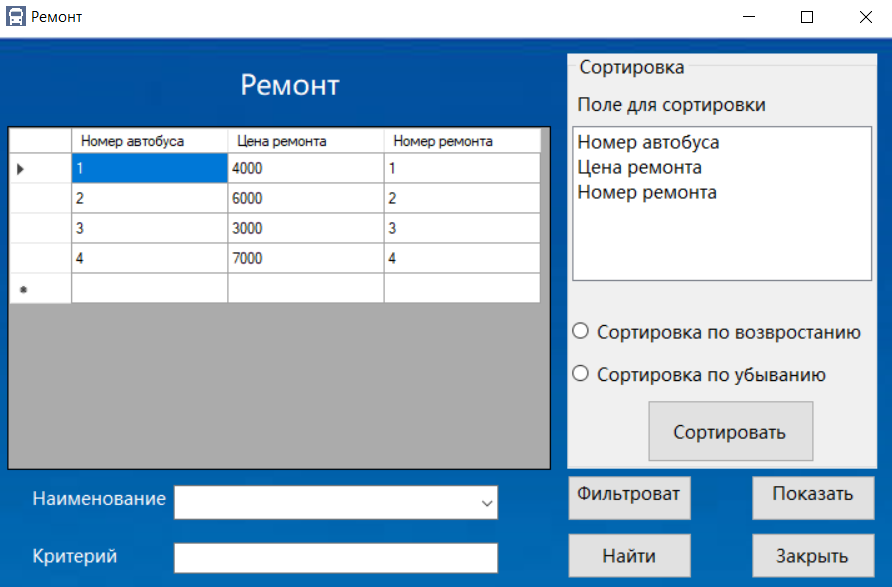


Рис. 45 Форма Ремонт

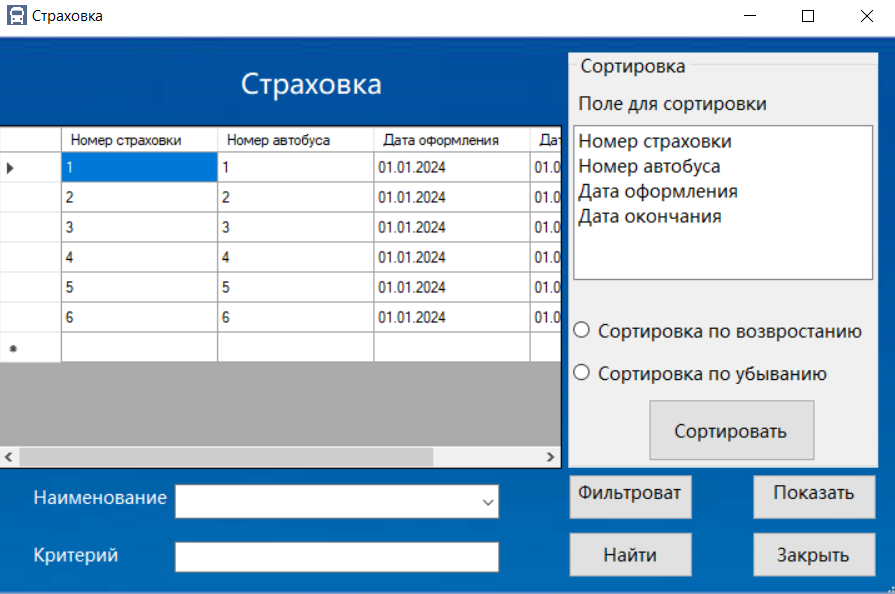


Рис. 46 Форма Страховка

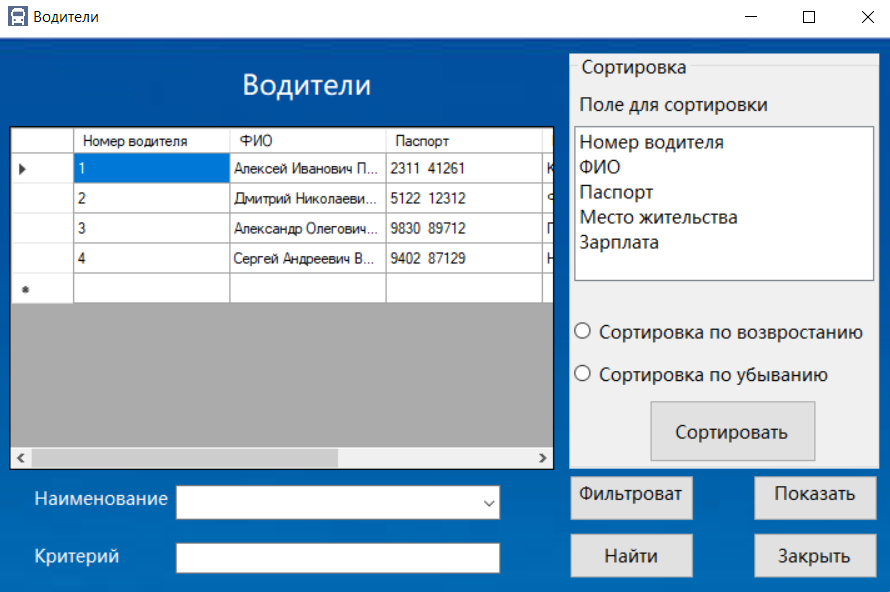


Рис. 47 Форма Водитель

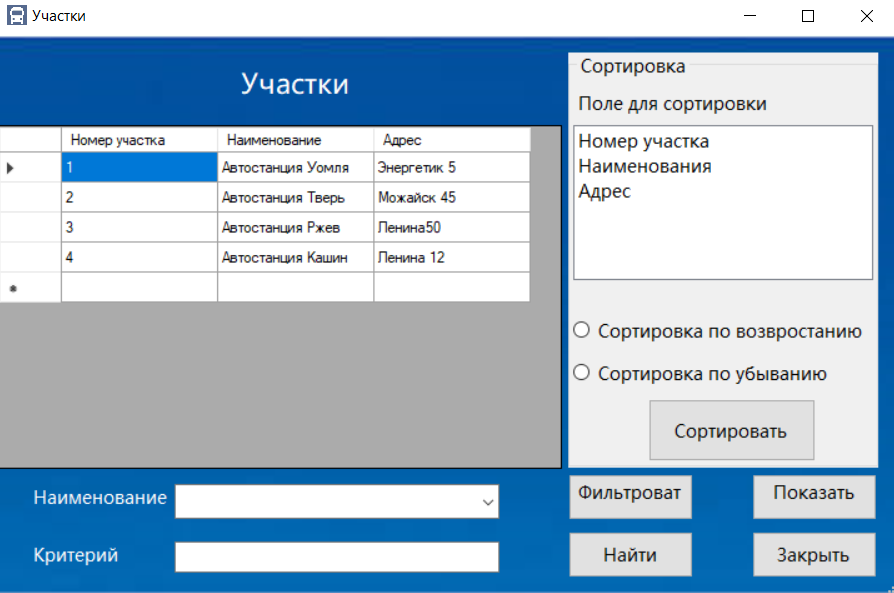


Рис. 48 Форма Участки

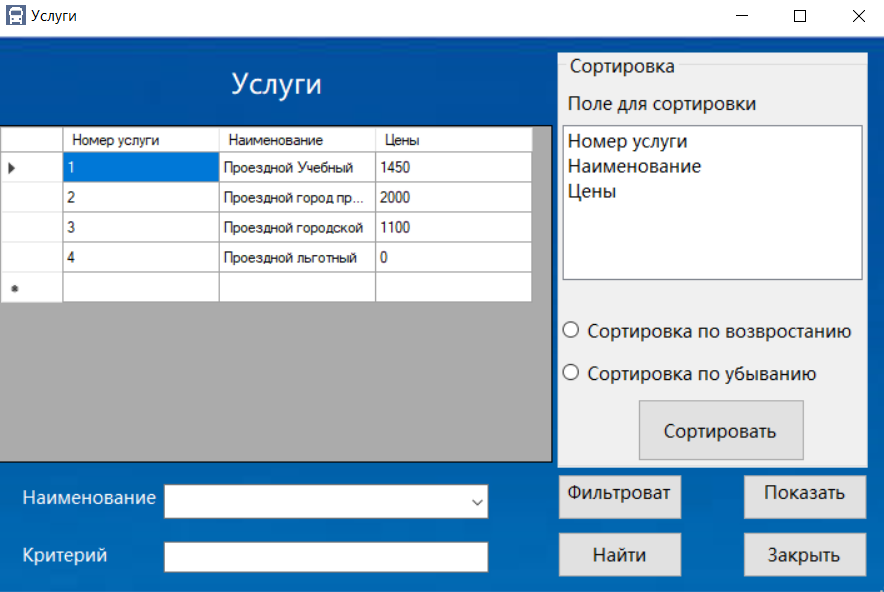


Рис. 49 Форма Услуги

## **3.2. Листинг программы**

Титульная форма

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Avtoctation

{

public partial class Tutl : Form

{

public Tutl()

{

InitializeComponent();

}

private void Exit\_Click(object sender, EventArgs e)

{

Close();

}

private void Voitu\_Click(object sender, EventArgs e)

{

ALLFORMS frm = new ALLFORMS();

this.Hide();

frm.ShowDialog();

this.Show();

}

private void label1\_Click(object sender, EventArgs e)

{

}

}

}

Форма Всех форм

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Avtoctation

{

public partial class ALLFORMS : Form

{

public ALLFORMS()

{

InitializeComponent();

}

private void Exit\_Click(object sender, EventArgs e)

{

Close();

}

private void Zatratblnazapravky\_Click(object sender, EventArgs e)

{

Zatratblnazapravky frm = new Zatratblnazapravky();

frm.ShowDialog();

}

private void Rumont\_Click(object sender, EventArgs e)

{

Rumont frm = new Rumont();

frm.ShowDialog();

}

private void button4\_Click(object sender, EventArgs e)

{

Ctraxovka frm = new Ctraxovka();

frm.ShowDialog();

}

private void Masterskue\_Click(object sender, EventArgs e)

{

Masterskue frm = new Masterskue();

frm.ShowDialog();

}

private void Parkovku\_Click(object sender, EventArgs e)

{

Parkovku frm = new Parkovku();

frm.ShowDialog();

}

private void Kluent\_Click(object sender, EventArgs e)

{

Kluentbl frm = new Kluentbl();

frm.ShowDialog();

}

private void Autobus\_Click(object sender, EventArgs e)

{

Autobus frm = new Autobus();

frm.ShowDialog();

}

private void Bulet\_Click(object sender, EventArgs e)

{

Bulet frm = new Bulet();

frm.ShowDialog();

}

private void lgotproezd\_Click(object sender, EventArgs e)

{

Lgotproezdnble frm = new Lgotproezdnble();

frm.ShowDialog();

}

private void Koductors\_Click(object sender, EventArgs e)

{

Kondyctors frm = new Kondyctors();

frm.ShowDialog();

}

private void Yslygu\_Click(object sender, EventArgs e)

{

Yslygi frm = new Yslygi();

frm.ShowDialog();

}

private void Marsryt\_Click(object sender, EventArgs e)

{

Marsryt frm = new Marsryt();

frm.ShowDialog();

}

private void Raspusanue\_Click(object sender, EventArgs e)

{

Raspusanue frm = new Raspusanue();

frm.ShowDialog();

}

private void Proezd\_Click(object sender, EventArgs e)

{

Proezd frm = new Proezd();

frm.ShowDialog();

}

private void Vodutelu\_Click(object sender, EventArgs e)

{

Vodutelu frm = new Vodutelu();

frm.ShowDialog();

}

private void Ychastku\_Click(object sender, EventArgs e)

{

Ychastku frm = new Ychastku();

frm.ShowDialog();

}

}

}

Форма Автобусы

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Avtoctation

{

public partial class Autobus : Form

{

public Autobus()

{

InitializeComponent();

}

private void Autobus\_Load(object sender, EventArgs e)

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "oneSDataSet.Avtobus". При необходимости она может быть перемещена или удалена.

this.avtobusTableAdapter.Fill(this.oneSDataSet.Avtobus);

}

private void polesorturovku\_SelectedIndexChanged(object sender, EventArgs e)

{

Corturovka.Enabled = true;

}

private System.Windows.Forms.DataGridViewColumn COL;

private void Corturovka\_Click(object sender, EventArgs e)

{

COL = new System.Windows.Forms.DataGridViewColumn();

switch (polesorturovku.SelectedIndex)

{

case 0:

COL = dataGridViewTextBoxColumn1;

break;

case 1:

COL = dataGridViewTextBoxColumn2;

break;

case 2:

COL = dataGridViewTextBoxColumn3;

break;

}

if (SortVozvrostanue.Checked)

avtobusDataGridView.Sort(COL,

System.ComponentModel.ListSortDirection.Ascending);

else

avtobusDataGridView.Sort(COL,

System.ComponentModel.ListSortDirection.Descending);

}

private void fultr\_Click(object sender, EventArgs e)

{

avtobusBindingSource.Filter = "WorkerID='" + naumenov.Text + "'";

}

private void showall\_Click(object sender, EventArgs e)

{

avtobusBindingSource.Filter = "";

}

private void naitu\_Click(object sender, EventArgs e)

{

{

for (int i = 0; i < avtobusDataGridView.ColumnCount - 1; i++)

{

for (int j = 0; j < avtobusDataGridView.RowCount - 1; j++)

{

avtobusDataGridView[i, j].Style.BackColor = Color.White;

avtobusDataGridView[i, j].Style.ForeColor = Color.Black;

}

}

for (int i = 0; i < avtobusDataGridView.ColumnCount - 1; i++)

{

for (int j = 0; j < avtobusDataGridView.RowCount - 1; j++)

{

if (avtobusDataGridView[i,

j].Value.ToString().IndexOf(textBox1.Text) != -1)

{

avtobusDataGridView[i, j].Style.BackColor = Color.AliceBlue;

avtobusDataGridView[i, j].Style.ForeColor = Color.Blue;

}

}

}

}

}

private void Exit\_Click(object sender, EventArgs e)

{

Close();

}

}

}

## **3.4. Руководство пользователя**

**1. Введение**

Приложение «Система автостанции» разработано для упрощения процесса поиска информации о расписании, маршрутах и услугах автостанции. Оно включает функции для просмотра расписания, покупки билетов, а также предоставления информации о доступных услугах. Это приложение подходит как для пассажиров, так и для сотрудников автостанции, желающих оптимизировать свою работу.

**1.2. Краткое описание возможностей**

Основные возможности приложения включают:

Просмотр расписания и маршрутов: Пользователи могут быстро находить информацию о ближайших рейсах и маршрутах.

Покупка билетов: Возможность оформления и оплаты билетов через приложение.

Поиск и фильтрация информации: Удобные инструменты для поиска по направлениям, времени отправления и другим критериям.

Управление услугами: Для администраторов — возможность добавлять и редактировать информацию о маршрутах, расписании и услугах.

**1.3. Уровень подготовки пользователя**

Приложение «Система автостанции» разработано с учетом потребностей пользователей с разным уровнем подготовки. Пользователи должны иметь базовые навыки работы с мобильными устройствами или компьютерами для качественного взаимодействия с приложением.

**1.4. Перечень эксплуатационной документации**

Для эффективного использования приложения рекомендуется ознакомиться с документами:

Руководство по установке и настройке приложения: Информация о процессе загрузки, установки и первоначальной настройки приложения.

Инструкция по созданию и редактированию профиля пользователя: Пошаговое руководство по созданию и редактированию личного профиля.

Рекомендации по использованию услуг автостанции: Советы и рекомендации по поиску информации и покупке билетов.

Пожалуйста, ознакомьтесь с вышеуказанными документами для более глубокого понимания функционала приложения и повышения эффективности его использования.

**2. Назначение и условия применения**

**2.1. Виды деятельности и функции**

Приложение «Система автостанции» предназначено для автоматизации следующих видов деятельности и функций:

Поиск информации о расписании и маршрутах: Пользователи могут искать и фильтровать актуальные данные по различным критериям, включая направление, время отправления и тип транспорта.

Покупка билетов: Возможность оформления и оплаты билетов через приложение.

Управление услугами: Для администраторов — добавление и редактирование информации о маршрутах, расписании и услугах.

Аналитика и отчетность: Представление аналитической информации о загруженности маршрутов и продаже билетов.

**2.2. Условия применения**

Для обеспечения эффективной работы приложения «Система автостанции» необходимо соблюдение следующих условий:

Оборудование: Приложение совместимо с большинством современных мобильных устройств и настольных компьютеров, работающих на операционных системах iOS, Android и Windows.

Технические средства: Рекомендуется наличие устройства с минимальными системными требованиями: процессор не ниже 1 ГГц, 2 ГБ оперативной памяти, поддержка интернет-соединения.

Операционная среда: Установка приложения требует наличия соответствующих операционных систем, поддерживающих платформы для мобильных приложений и веб-браузеров.

Общесистемные программные средства: Установка и использование приложения предполагает наличие актуальных версий браузеров и поддержку C#.

Входная информация и носители данных: Для работы с приложением пользователю необходимо иметь доступ к интернету и учетной записи.

База данных: Приложение использует облачное хранилище для хранения информации о маршрутах, расписании и пользователях, что требует наличия постоянного интернет-соединения.

Подготовка специалистов: Пользователи приложения должны иметь базовые навыки работы с компьютером и смартфонами.

Соблюдение этих условий позволит максимально эффективно использовать приложение «Система автостанции» для поиска информации и покупки билетов.

**3. Подготовка к работе**

**3.1. Состав и содержание дистрибутивного носителя данных**

Дистрибутивный носитель данных для приложения «Система автостанции» включает в себя:

Установочный файл: Файл приложения в формате APK для Android, IPA для iOS или EXE для Windows, предназначенный для установки на соответствующих устройствах.

Документация: Руководство пользователя и краткие инструкции по установке, содержащие основные рекомендации и советы по работе с приложением.

Образцы данных: Примеры маршрутов и расписания, которые помогут новым пользователям быстрее понять, как эффективно использовать приложение.

Обновления: Записи для обновления приложения, если оно уже установлено. Обновления могут содержать исправления ошибок, новые функции и улучшения производительности.

**3.2. Порядок загрузки данных и программ**

Чтобы загрузить и установить приложение «Система автостанции», выполните следующие шаги:

Загрузка для мобильных устройств:

Откройте магазин приложений (Google Play для Android или App Store для iOS).

Введите в строке поиска «Система автостанции».

Выберите приложение из списка и нажмите на кнопку «Установить» или «Получить».

Дождитесь завершения загрузки и установки приложения.

Загрузка для настольных компьютеров:

Перейдите на официальную веб-страницу приложения.

Найдите раздел «Скачать» или «Загрузить для Windows».

Нажмите на ссылку для скачивания установочного файла.

Запустите загруженный файл и следуйте инструкциям установщика для завершения установки.

После установки рекомендуется перезапустить устройство для применения всех изменений.

**3.3. Порядок проверки работоспособности**

Чтобы убедиться, что приложение «Система автостанции» успешно установлено и работает корректно, выполните следующие действия:

Запуск приложения: Найдите значок приложения на своем устройстве и запустите его.

Проверка входа в систему: Попробуйте войти в приложение, используя созданную учетную запись. Если у вас еще нет учетной записи, зарегистрируйтесь, следуя инструкциям на экране.

Тестирование функциональности:

Создайте тестовый запрос на поиск маршрута.

Проверьте возможность покупки билета.

Убедитесь, что все функции работают корректно.

После выполнения всех этих шагов вы сможете уверенно использовать приложение «Система автостанции» для поиска информации и покупки билетов.

**4. Описание операций**

**4.1. Создание профиля пользователя**

Наименование: Создание профиля пользователя

Условия выполнения: Доступ к приложению и наличие учетной записи пользователя.

Подготовительные действия:

Вход в приложение.

Сбор необходимой информации о личных данных.

Основные действия:

Выбрать раздел «Профиль» в главном меню приложения.

Нажать кнопку «Создать новый профиль».

Заполнить поля с личной информацией.

Нажать кнопку «Сохранить».

Заключительные действия:

Просмотреть созданный профиль для корректности информации.

При необходимости отредактировать и сохранить изменения.

**4.2. Поиск маршрутов**

Наименование: Поиск маршрутов

Условия выполнения: Способность подключения к интернету и наличие актуальной базы данных о маршрутах.

Подготовительные действия:

Вход в приложение.

Установка фильтров для поиска (по направлению, времени отправления и т.д.).

Основные действия:

Перейти в раздел «Маршруты».

Изучить полученный список маршрутов и выбрать интересующие предложения.

При необходимости сохранить информацию о маршруте или оформить билет.

**4.3. Покупка билета**

Наименование: Покупка билета

Условия выполнения: Наличие выбранного маршрута и доступ к интернету.

Подготовительные действия:

Убедиться в наличии всех необходимых данных для оформления билета.

Основные действия:

Перейти в раздел «Корзина».

Проверить список выбранных билетов и их количество.

Нажать кнопку «Оформить заказ» и следовать инструкциям для выбора способа оплаты.

Заключительные действия:

Подтвердить заказ и сохранить информацию о билете.

**4.4. Просмотр истории покупок**

Наименование: Просмотр истории покупок

Условия выполнения: Наличие активной учетной записи и доступ к интернету.

Подготовительные действия:

Вход в приложение.

Основные действия:

Перейти в раздел «История покупок».

Ознакомиться с ранее приобретенными билетами.

Проанализировать частоту поездок и общее количество потраченных средств.

Заключительные действия:

Сохранить важные данные для дальнейшего использования в планировании поездок.

Эти операции составляют основу функциональности приложения «Система автостанции», предоставляя пользователям удобные инструменты для поиска информации и покупки билетов.

**5.** **Аварийные ситуации**

Таблица 5

Авариные ситуации

|  |  |  |  |
| --- | --- | --- | --- |
| **Класс ошибки** | **Ошибка** | **Описание ошибки** | **Требуемые действия пользователя** |
| Отказ сервера | Сервер не найден. | Невозможно отобразить страницу. | Обратиться к технической поддержке (HelpDesk) или администратору системы. |
| Ошибка регистрации | Требуется ввести действительное имя пользователя. | При регистрации на портале не введено имя пользователя. | Ввести имя пользователя. |
| Требуется ввести пароль для регистрации. | При регистрации не введен пароль. | Ввести пароль. |
| Сбой аутентификации. | Неверно введено имя пользователя или пароль. | Повторить попытку. Если ошибка сохраняется, обратиться к администратору. |
| Несанкционированное изменение | Обнаружено несанкционированное вмешательство в данные. | Изменения в профиле, выполненные незнакомыми пользователями. | Сменить пароль от учетной записи. Обратиться в техническую поддержку. |

**6. Рекомендации по освоению**

**6.1. Общие рекомендации**

Перед началом работы в приложении «Система автостанции» рекомендуется ознакомиться с интерфейсом, изучив основные разделы, такие как:

Расписание: Просмотр актуального расписания движения транспорта.

Маршруты: Поиск информации о доступных маршрутах и направлениях.

Билеты: Покупка и управление билетами.

История покупок: Просмотр ранее приобретенных билетов.

Профиль: Управление личными данными и настройками учетной записи.

Это поможет быстрее ориентироваться в функционале приложения и находить нужную информацию.

Создание учетной записи

Рекомендуется создать учетную запись сразу после установки приложения. Это позволит получить доступ к персонализированным функциям, таким как:

Сохранение истории покупок: Возможность просматривать ранее приобретенные билеты и планировать будущие поездки.

Создание профиля пользователя: Добавление личных данных для удобства оформления билетов и получения уведомлений.

Получение уведомлений: Информация об изменениях в расписании, акциях и специальных предложениях.

Для создания учетной записи выполните следующие шаги:

Откройте приложение и перейдите в раздел «Профиль».

Нажмите кнопку «Создать учетную запись».

Заполните необходимые поля (имя, электронная почта, пароль и т.д.).

Подтвердите регистрацию, следуя инструкциям на экране.

После создания учетной записи вы сможете полноценно использовать все возможности приложения «Система автостанции».

**6.2. Контрольный пример**

Для практического освоения приложения можно использовать следующий контрольный пример:

Цель: Найти подходящий маршрут и оформить билет.

Шаги выполнения:

Создание профиля пользователя:

Перейдите в раздел «Профиль».

Нажмите на кнопку «Создать новый профиль».

Заполните все обязательные поля: имя, контактная информация, электронная почта.

Сохраните профиль.

Поиск маршрутов:

Перейдите в раздел «Маршруты».

Используйте фильтры для поиска по направлению, времени отправления и типу транспорта.

Просмотрите доступные маршруты и выберите подходящий.

Покупка билета:

Перейдите в раздел «Билеты».

Выберите маршрут и нажмите кнопку «Купить билет».

Следуйте инструкциям для выбора места и способа оплаты.

Просмотр истории покупок:

Перейдите в раздел «История покупок».

Проверьте информацию о ранее приобретенных билетах.

При необходимости сохраните данные для дальнейшего использования.

**6.3. Правила запуска и выполнения**

Запуск приложения:

Убедитесь, что ваше устройство подключено к интернету.

Запустите приложение, нажав на его значок на экране.

Введите свои учетные данные для входа, если у вас уже есть учетная запись.

Выполнение операций:

Следуйте интуитивно понятному интерфейсу: все разделы и кнопки обозначены понятно, что облегчает выполнение операций.

Используйте доступные подсказки и инструкции в приложении в случае возникновения затруднений.

Если возникли вопросы или сложности, не стесняйтесь обращаться в техническую поддержку.

# **ГЛАВА 4. РАЗРАБОТКА 1С**

## **4.1 Разработка подсистем**

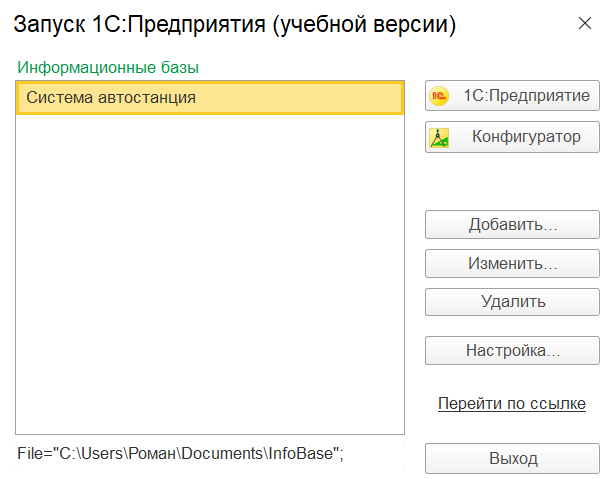
****

Рис. 50 Окно входа в систему

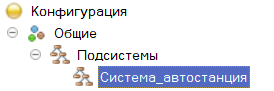
****

Рис. 51 Список подсистем

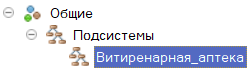


Рис. 52 Созданная подсистема

## **4.2 Разработка справочников**

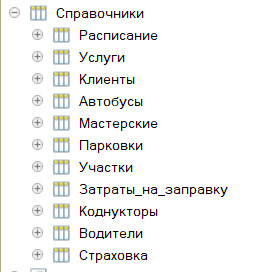


Рис. 53 Создание справочников

## **4.3 Разработка документов и регистров**

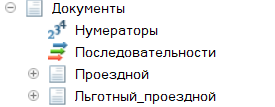


Рис. 54 Создание документации

## **4.4 Разработка отчетов**

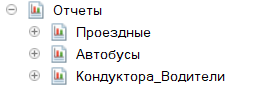


Рис. 55 Создание отчета

**СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Бессмертный, И. А. Системы искусственного интеллекта : учеб. пособие для СПО / И. А. Бессмертный. — 2-е изд., испр. и доп. — М. : Издательство Юрайт, 2018. — 130 с.
2. Гниденко, И. Г. Технология разработки программного обеспечения : учеб. пособие для СПО / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — М. : Издательство Юрайт, 2017. — 235 с.
3. Гордеев, С. И. Организация баз данных в 2 ч. Часть 2 : учебник для вузов / С. И. Гордеев, В. Н. Волошина. — 2-е изд., испр. и доп. — М. : Издательство Юрайт, 2019. — 501 с.
4. Жмудь, В. А. Моделирование замкнутых систем автоматического управления : учеб. пособие для академического бакалавриата / В. А. Жмудь. — 2-е изд., испр. и доп. — М. : Издательство Юрайт, 2019. — 128 с.
5. Зыков, С. В. Программирование. Объектно-ориентированный подход : учебник и практикум для академического бакалавриата / С. В. Зыков. — М. : Издательство Юрайт, 2019. — 155 с.
6. Иванов, В. М. Интеллектуальные системы : учеб. пособие для СПО / В. М. Иванов ; под науч. ред. А. Н. Сесекина. — М. : Издательство Юрайт, 2019. — 93 с.
7. Иванов, В. М. Интеллектуальные системы : учеб. пособие для вузов / В. М. Иванов ; под науч. ред. А. Н. Сесекина. — М. : Издательство Юрайт, 2017. — 91 с.
8. Кубенский, А. А. Функциональное программирование : учебник и практикум для академического бакалавриата / А. А. Кубенский. — М. : Издательство Юрайт, 2019. — 348 с.
9. Кудрина, Е. В. Основы алгоритмизации и программирования на языке c# : учеб. пособие для СПО / Е. В. Кудрина, М. В. Огнева. — М. : Издательство Юрайт, 2019. — 322 с.