ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ КОСТРОМСКОЙ ОБЛАСТИ

Областное государственное бюджетное профессиональное Образовательное учреждение

«Костромской энергетический техникум имени Ф.В.Чижова» (ОГБПОУ «КЭТ им.Ф.В.Чижова»)

ЗАЩИЩЕНО ОЦЕНКА

Куратор специальности

09.02.07 Информационные

системы и программирование

О.В. Ищук

УПРАВЛЕНИЕ СЛУЖБОЙ ПАССАЖИРСКИХ ПЕРЕВОЗОК

Пояснительная записка к курсовому проекту

по ПМ.05 Проектирование и разработка

информационных систем

КЭТ 09.02.07 КП 007 ПЗ

Руководитель КП Разработал

Преподаватель КЭТ студент группы 3-2ИС

Ищук Д.Н. Левченко М.Ю.

Кострома, 2024

ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ КОСТРОМСКОЙ ОБЛАСТИ

Областное государственное бюджетное профессиональное Образовательное учреждение

«Костромской энергетический техникум имени Ф.В.Чижова» (ОГБПОУ «КЭТ им.Ф.В.Чижова»)

|  |  |
| --- | --- |
| Специальность 09.02.07 Информационные системы и программирование  Квалификация «Специалист по информационным системам» | **УТВЕРЖДЕНО**  цикловой комиссией  « 9 » января 2024 года Куратор специальности  О.В. Ищук |

ЗАДАНИЕ

Для курсового проектирования по МДК 05.02 «Разработка кода информационных систем» студенту 3 курса группы 3-2 ИС

|  |  |
| --- | --- |
|  | Левченко Максиму Юрьевичу |
|  | (фамилия, имя, отчество) |
| Тема задания: | Управление службой пассажирских перевозок |

При выполнении курсового проекта на указанную тему должны быть представлены :

I Пояснительная записка

|  |  |
| --- | --- |
| Введение | |
| 1 Анализ предметной области и описание технологии проектирования ИС | |
| 1.1 Описание предметной области | |
| 1.2 Определение целевых групп пользователей | |
| 1.3 Определение ограничений проектного решения | |
| 1.4 Анализ предметной области | |
| 2 Определение требований к системе | |
| 2.1 Определение требований к функциональности приложения | |
| 2.2 Определение требований к графическому интерфейсу | |
| 2.1.1 Определение требований к дизайну | |
| 2.1.2 Определение требований к эргономике | |
| 2.3 Определение требований к надежности системы | |
| 3 Описание этапов разработки ИС | |
| 3.1 Описание исходных данных | |
| 3.2 Описание физической модели базы данных | |
| 3.3 Схема БД | |
| 3.4 Описание структуры приложения | |
| 3.5 Разработка макетов страниц | |
| 3.6 Описание дизайна приложения | |
| 3.7 Описание приемов верстки | |
| 3.8 Анализ и описание технологий программирования | |
| 3.9 Выбор домена и хостинга | |
| 3.10 Описание этапов тестирования программного решения | |
| 4 Руководство пользователя | |
| 5 Руководство администратора | |
| Заключение | |
| Список используемых источников | |
| Приложение А. Листинг программы | |
| Приложение Б. Результаты тестирования (TEST-CASE) | |

II Приложение, размещенное на сервере.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Дата выдачи: | « » | 20 г. | Срок окончания: | « » | 20\_г |

|  |  |
| --- | --- |
| Руководитель курсового проектирования Ищук Д.Н |  |
|  |  |

**СОДЕРЖАНИЕ**

[Основная часть 6](#_Toc168646243)

[1.Анализ предметной области 6](#_Toc168646244)

[2.Определение требований к системе 9](#_Toc168646245)

[3.Описание этапов разработки ИС 10](#_Toc168646246)

[4.Структура клиенткой части (Wireframe) 13](#_Toc168646247)

[5.Анализ и описание технологий программирования 17](#_Toc168646248)

[6.Проектирование API 19](#_Toc168646249)

# Введение

В современном мире, характеризующемся стремительными темпами глобализации и технологического прогресса, управление службой пассажирского транспорта становится ключевым аспектом обеспечения эффективности и комфорта перемещения граждан. Развитие информационных технологий оказывает значительное воздействие на организацию и функционирование этой отрасли, предоставляя новые возможности для оптимизации процессов и повышения качества предоставляемых услуг.

Актуальность темы обусловлена не только растущей потребностью в эффективных транспортных решениях, но и увеличением числа городского населения, что ставит перед управленческими структурами транспортных служб серьезные вызовы. Эффективное управление службой пассажирского транспорта становится неотъемлемым элементом развития устойчивых и интеллектуальных городов.

В контексте данного исследования, особое внимание уделяется роли новых информационных технологий в транспортной сфере. Внедрение цифровых инноваций, таких как системы онлайн-бронирования, системы мониторинга и управления транспортным потоком, вносит существенные изменения в организацию и взаимодействие участников транспортного процесса.

Целью данной курсовой работы является проектирование информационной системы по управлению службой пассажирского транспорта.

# Основная часть

1. Анализ предметной области

Управление службой пассажирских перевозок включает в себя координацию и организацию транспортных услуг для обеспечения эффективного и безопасного перемещения пассажиров. Основная цель состоит в том, чтобы обеспечить высокий уровень обслуживания при минимальных затратах.

Основные функции:

1. Регистрация и аутентификация: система должна предоставлять механизм регистрации и аутентификации для пользователей; разграничение доступа на уровне ролей.
2. Управление пользовательским профилем: возможность изменения профиля пользователем.
3. Планирование и заказ транспорта: функционал для планирования и заказа транспорта; возможность выбора маршрута.
4. Управление транспортным парком: функционал для добавления, удаления и обновления информации о транспортных средствах.

Технологии проектирования ИС:

1. ASP.NET Web API: Для создания RESTful веб-сервисов, обеспечивающих взаимодействие между клиентскими приложениями и сервером.
2. MS SQL Server: Для управления базой данных.
3. Git: Для управления версиями кода и совместной работы.

Основные элементы предметной области:

1. Маршруты и расписания

* Маршруты: Планы движения транспортных средств, определяющие остановки.
* Расписание: Время отправления и прибытия транспортных средств на каждой остановке.

1. Транспортные средства:

* Автобусы, микроавтобусы: Различные типы транспортных средств, используемые для перевозки пассажиров.

1. Информационные системы:

* Система управления транспортом: Позволяет планировать маршруты, управлять транспортными средствами.
* Система информационного обслуживания пассажиров: Предоставляет пассажирам информацию о расписании, изменениях маршрутов и доступных услугах.

Целевые группы пользователей:

1. Администраторы системы

* Задачи: Управление доступом к системе, настройка параметров работы, обеспечение безопасности данных.

1. Менеджеры по планированию

* Задачи: Составление и оптимизация маршрутов и расписаний, анализ пассажиропотока.

1. Водители

* Задачи: Выполнение рейсов в соответствии с расписанием.

1. Пассажиры

* Задачи: Использование услуг пассажирских перевозок, получение информации о расписании и изменениях.

Определение ограничений проектного решения:

1. Ограничения производительности:

* Система должна обеспечивать высокую производительность при обработке большого объема данных в режиме реального времени, включая данные GPS-трекинга, расписаний и информации о транспортных средствах.

1. Ограничения по совместимости:

* Система должна быть совместима с различными устройствами и операционными системами, используемыми пользователями, включая настольные компьютеры, планшеты и смартфоны.

1. Ограничения по безопасности:

* Необходимость соблюдения стандартов безопасности, таких как шифрование данных при передаче и хранении.

1. Ограничения по масштабируемости:

* Система должна быть масштабируемой для поддержки увеличения количества пользователей, транспортных средств и объема данных по мере роста службы пассажирских перевозок.

## 2. Определение требований к системе

Система управления службой пассажирских перевозок должна обеспечивать широкий спектр функциональных возможностей, поддерживать многопользовательский режим работы и удовлетворять потребности различных категорий пользователей. Основные функциональные требования включают:

1. Управление маршрутами и расписаниями:

* Создание, редактирование и удаление маршрутов.
* Составление и обновление расписаний для различных маршрутов.
* Оптимизация маршрутов с учетом пассажиропотока и дорожных условий.

1. Управление транспортными средствами:

* Учет транспортных средств, их технических характеристик и состояния.
* Планирование и учет технического обслуживания и ремонтов.

1. Обслуживание пассажиров:

* Информирование пассажиров о расписании и изменениях в маршрутах.
* Продажа билетов через веб-портал.

Дизайн и эргономика графического интерфейса системы должны обеспечивать удобство и эффективность работы для всех категорий пользователей. Основные требования включают:

1. Удобство использования:

* Интуитивно понятный и простой интерфейс для всех категорий пользователей.
* Логичная навигация и структура меню, обеспечивающая быстрый доступ к основным функциям.
* Минимизация количества шагов для выполнения основных операций.

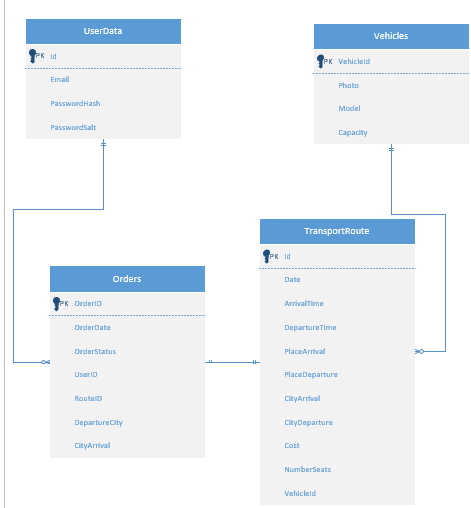
1. Дизайн и эргономика:

* Современный и привлекательный дизайн, соответствующий корпоративному стилю.
* Использование стандартных элементов интерфейса и паттернов, известных пользователям.

## 3. Описание этапов разработки ИС

Физическая модель данных является важным компонентом проектирования базы данных. Она описывает, как данные будут храниться на физическом уровне, включая используемые устройства хранения, структуры данных и методы доступа. Физическая модель должна быть разработана таким образом, чтобы обеспечить эффективность, надежность и масштабируемость базы данных.

Рисунок 1. ER-диаграмма



Сущности:

* Пользователи (UserData):
* Идентификатор пользователя (PK\_UserID): Целое число, первичный ключ.
* Имя пользователя (UserName): Строка текста.
* Хеш пароля (PasswordHash): Строка текста.
* Соль пароля (PasswordSalt): Строка текста.
* Транспортные средства (Vehicles):
* Идентификатор транспортного средства (PK\_VehicleID): Целое число, первичный ключ.
* Фото транспорта (Photo): Строка текста
* Модель (Model): Строка текста.
* Вместимость (Capacity): Целое число.
* Заказы (Orders):
* Идентификатор заказа (PK\_OrderID): Целое число, первичный ключ.
* Дата заказа (Date): Дата и время.
* Время прибытия (ArrivalTime): Дата и время.
* Время убытия (DepartureTime): Дата и время.
* Место прибытия (PlaceArrival): Строка текста.
* Место отправления (PlaceDeparture): Строка текста.
* Город прибытия (CityArrival): Строка текста.
* Город убытия (CityDeparture): Строка текста.
* Цена (Cost): Число.
* Количество мест (NumberSeats): Число.
* Идентификатор транспортного средства (VehicleID): Целое число, внешний ключ, ссылается на таблицу Vehicles.

Связи:

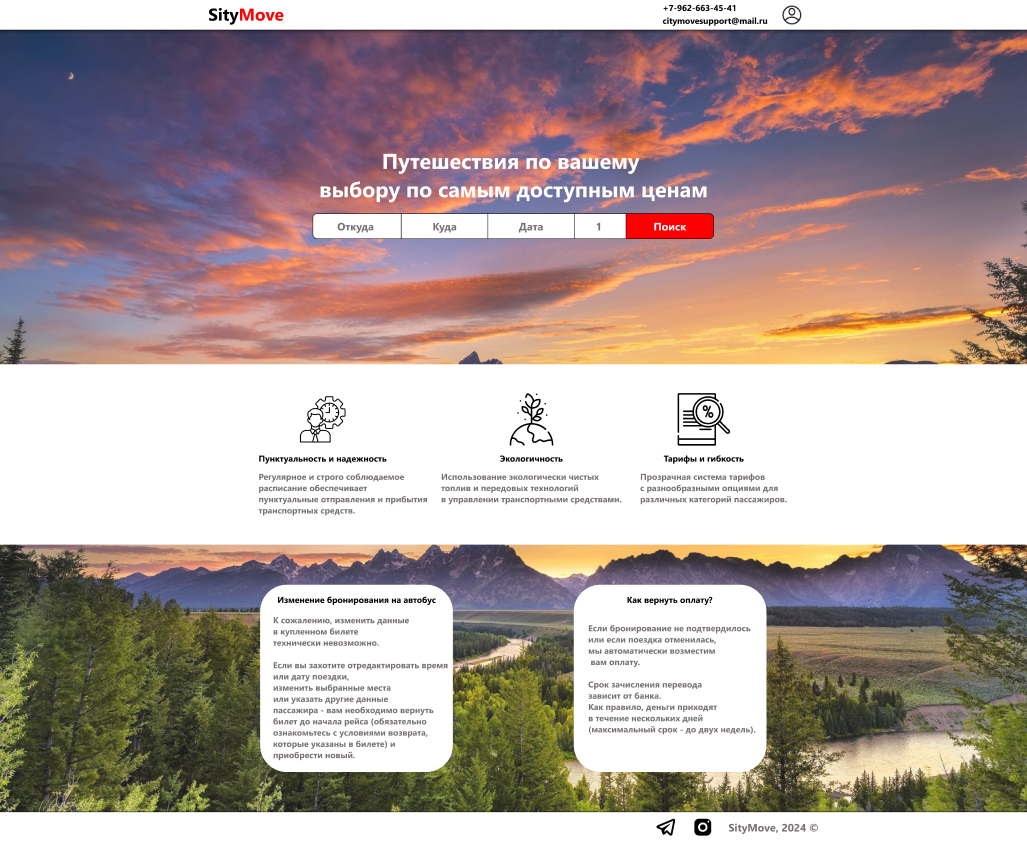
* Один пользователь может иметь несколько заказов.
* Одно транспортное средство может использоваться для выполнения нескольких заказов.
* Один заказ может быть выполнен только на одном транспортном средстве.

Описание:

* Таблица UserData хранит информацию о пользователях системы.
* Таблица Vehicles хранит информацию о транспортных средствах, используемых для выполнения заказов.
* Таблица Orders хранит информацию о заказах, которые необходимо выполнить.

## 4. Структура клиенткой части (Wireframe)

Рисунок 2. Главная страница

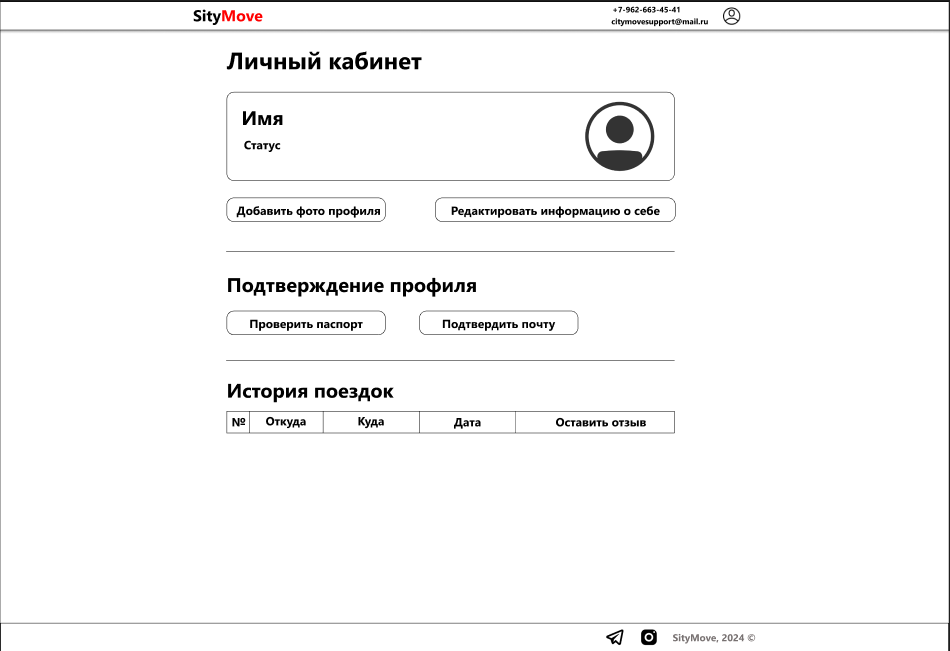


Строка поиска: Строка поиска размещена в центре экрана. Она позволяет пользователям искать маршруты по пункту назначения, дате и другим критериям.

Блоки с преимуществами компании: Блоки с преимуществами компании размещены под строкой поиска. Они описывают пунктуальность и надежность компании, ее экологичность, тарифы и гибкость.

Блок с информацией о бронировании: Блок с информацией о бронировании размещен под блоками с преимуществами компании. Он содержит информацию о том, как изменить или отменить бронирование.

Рисунок 3. Личный кабинет

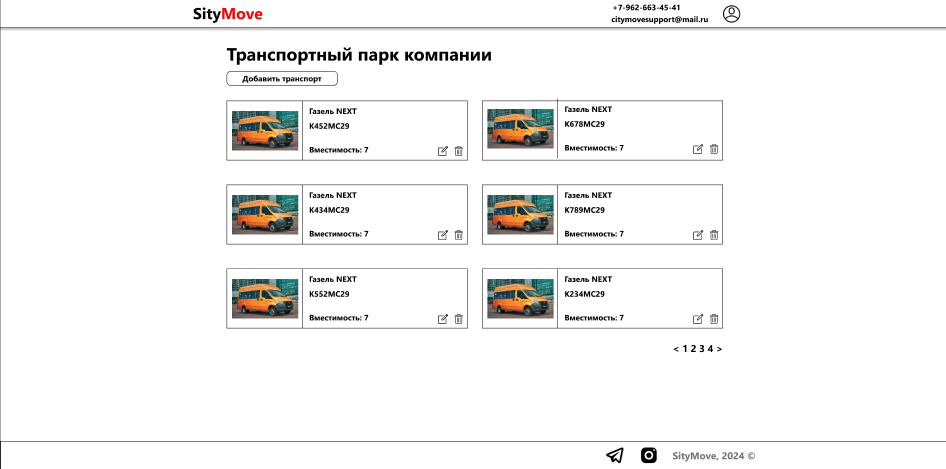


Профиль пользователя: Профиль пользователя отображает информацию о пользователе, такую как его имя, фамилию, статус, фото.

Мои бронирования: Список бронирований пользователя отображает все бронирования, которые он сделал. Пользователь может просмотреть информацию о каждом бронировании, изменить или отменить его, как и на скриншоте.

Настройки: Настройки позволяют пользователю добавить фото профиля, редактировать информацию о себе, проверить паспорт, подтвердить почту.

Рисунок 4. Транспортный парк компании

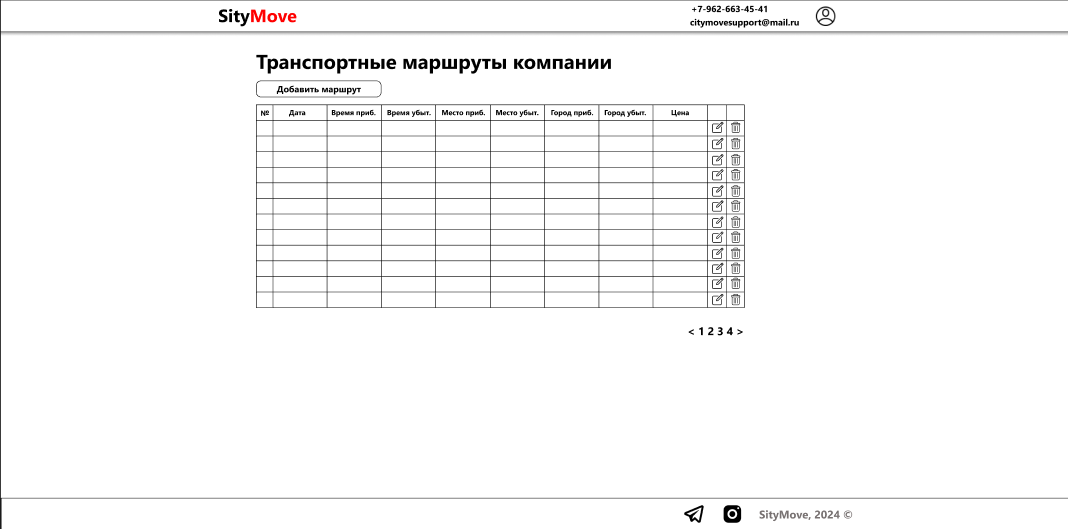


Кнопка "Добавить транспорт": При нажатии на эту кнопку открывается страница для добавления нового транспортного средства в парк компании.

Кнопка "Редактирования": При нажатии на эту кнопку открывается страница для редактирования информации о транспортом средстве.

Кнопка "Удаление": При нажатии на эту кнопку происходит удаление транспортного средства.

Рисунок 5. Транспортные маршруты копании



Кнопка "Добавить маршрут": При нажатии на эту кнопку открывается страница для добавления нового транспортного маршрута компании.

Кнопка "Редактирования": При нажатии на эту кнопку открывается страница для редактирования информации о маршруте.

Кнопка "Удаление": При нажатии на эту кнопку происходит удаление маршрута.

## 5. Анализ и описание технологий программирования

Клиентская часть

* HTML (HyperText Markup Language):

Основной язык разметки для создания веб-страниц и веб-приложений. Определяет структуру веб-страницы, используя элементы и теги.

* CSS (Cascading Style Sheets):

Язык описания внешнего вида HTML-документов. Позволяет задавать стили (цвета, шрифты, отступы и т.д.) для элементов HTML. Поддерживает адаптивный дизайн (responsive design) через медиа-запросы.

* JavaScript:

Основной язык программирования для клиентской части веб-приложений. Используется для создания интерактивного и динамичного контента на веб-странице. Поддерживает манипуляцию DOM (Document Object Model), работу с событиями, AJAX-запросы для взаимодействия с сервером без перезагрузки страницы.

Серверная часть:

* ASP.NET Core:

Кроссплатформенный фреймворк для разработки веб-приложений и API от Microsoft. Высокая производительность и поддержка модульной структуры приложения. Поддерживает инъекцию зависимостей, middleware и многие другие современные концепции веб-разработки.

* ASP.NET Core Web API:

Специальный набор инструментов для создания RESTful API на платформе ASP.NET Core. Использует контроллеры для обработки HTTP-запросов и возвращения HTTP-ответов. Поддержка форматов JSON и XML для обмена данными.

Используемые плагины и библиотеки:

* Entity Framework Core:

ORM (Object-Relational Mapping) для работы с базами данных. Обеспечивает взаимодействие с базами данных с помощью объектов .NET, поддерживает миграции и запросы на языке LINQ.

* Swagger:

Инструмент для генерации документации API. Позволяет автоматически генерировать интерфейсы для тестирования API прямо из браузера.

* Newtonsoft.Json:

Популярная библиотека для работы с JSON в .NET. Поддерживает сериализацию и десериализацию объектов .NET в JSON и обратно.

* BCrypt.Net-Next:

Библиотека для хэширования паролей с использованием алгоритма BCrypt. Обеспечивает безопасное хранение паролей и защиту от атак на основе радужных таблиц.

Использование HTML, CSS и JavaScript для клиентской части обеспечивает создание динамичного и интерактивного интерфейса, а ASP.NET Core Web API на серверной части позволяет создавать производительные и надежные RESTful API. Включение библиотек и плагинов, таких как Entity Framework Core, Swagger, Newtonsoft.Json и BCrypt.Net-Next, значительно упрощает разработку, тестирование и поддержку приложения, обеспечивая высокую производительность, безопасность и удобство в работе.

## 6. Проектирование API

API (Application Programming Interface) — это интерфейс для взаимодействия между различными программными приложениями. API определяет методы и данные, которые могут быть запрошены или отправлены между системами, что позволяет приложениям общаться друг с другом. API может быть использован для различных целей, включая доступ к данным, функциональным возможностям или сервисам другой системы.

Таблица 1. Запросы API

|  |  |  |
| --- | --- | --- |
| HTTP-запрос | Конечная точка | Действие |
| POST | /Authentication/Authorization | Авторизация пользователя |
| POST | /Authentication/Registration | Регистрация пользователя |
| GET | /TransportRoutes/RoutesSearch | Поиск маршрута по заданным требованиям |
| GET | /TransportRoutes/RoutesList | Вывод всех маршрутов компании |
| POST | /TransportRoutes/RoutesAdd | Добавление маршрута |
| GET | /TransportRoutes/RoutesInfoID | Поиск маршрута по ID |
| PUT | /TransportRoutes/EditRoutes | Редактирование информации о маршруте |
| DELETE | /TransportRoutes/DeleteRoutes | Удаление маршрута |
| GET | /Vehicles/VehicleList | Вывод всех транспортных средств компании |
| GET | /Vehicles/VehicleInfoID | Поиск транспортного средства по ID |
| POST | /Vehicles/VehicleAdd | Добавление транспортного средства |
| PUT | /Vehicles/EditVehicle | Редактирование информации о транспортном средстве |
| DELETE | /Vehicles/DeleteVehicle | Удаление транспортного средства |