Министерство просвещения Республики Башкортостан

Государственное автономное профессиональное образовательное учреждение

Уфимский колледж статистики, информатики и вычислительной техники

ОТЧЁТ

по производственной практики

ПП.04.01 Производственная практика

по модулю ПМ.04 Сопровождение и обслуживание программного обеспечения компьютерных систем

Специальность

09.02.07 Информационные системы и программирование

Квалификация

Программист

|  |  |
| --- | --- |
| Методист отделения по УПР  /А.И. Файзулова  подпись ФИО | Руководитель практики от предприятия  / М.А. Рябов  подпись ФИО  М.П. |
| Руководитель практики от учебного заведения  / Д.А.Шарипова  подпись ФИО | Студент группы 22П-1  / В.Ю. Феденёв  подпись ФИО  « » 2025 год |

УФА – 2025 год

СОДЕРЖАНИЕ

|  |  |
| --- | --- |
|  | лист |
| Введение | 3 |
| 1. Характеристика предприятия и места практики | 5 |
| 1.1. Общая характеристика Группы компаний «ТрансТехСервис» | 5 |
| 1.2. Организационная структура и отдел информационных технологий | 5 |
| 2. Анализ аппаратного и программного обеспечения | 8 |
| 2.1. Анализ аппаратного обеспечения рабочего места | 8 |
| 2.2. Анализ программного обеспечения и сетевой инфраструктуры | 9 |
| 2.3. Анализ антивирусного обеспечения | 10 |
| 2.4. Настройка защиты системы стандартными средствами ОС | 11 |
| 2.4.1 Брандмауэр в операционной системе Windows | 11 |
| 2.4.2. Настройка брандмауэра в Linux (UFW) | 14 |
| 3. Проектирование Программного Обеспечения Для Решения Прикладной Задачи | 18 |
| 3.1 Постановка задачи. Техническое задание на разработку программного продукта | 18 |
| 3.1.1 Введение | 18 |
| 3.1.2 Основания для разработки | 18 |
| 3.1.3 Назначение разработки | 18 |
| 3.1.4 Требования к программе | 19 |
| 3.1.5 Требования к программной документации | 19 |
| 3.2 Описание программы | 20 |
| 3.2.1Архитектура | 20 |
| 3.2.2 Модульная схема и диаграмма базы данных | 26 |
| 3.3 Протокол тестирования разработанного программного продукта | 27 |
| 3.4 Руководство пользователя | 38 |
| Заключение | 48 |
| Список используемых источников | 49 |

ВВЕДЕНИЕ

Производственная практика является важнейшим этапом подготовки специалиста, позволяющим применить теоретические знания для решения реальных задач в условиях действующего предприятия.

Практика проходила с 27.11.2025 по 24.12.2025 в Группе компаний «ТрансТехСервис» (ТТС) – одном из крупнейших региональных автодилеров России. Местом прохождения практики был определен Отдел информационных технологий головного офиса в г. Уфе.

Целью практики являлось получение профессиональных навыков в области сопровождения, обслуживания и разработки программного обеспечения для автоматизации бизнес-процессов.

Для достижения цели были поставлены следующие задачи:

* ознакомиться со структурой, историей и основными бизнес-процессами предприятия.
* изучить состав, конфигурацию и принципы организации корпоративной IT-инфраструктуры.
* проанализировать используемое аппаратное и программное обеспечение, а также действующие политики безопасности.
* разработать действующий прототип внутреннего веб-приложения для автоматизации работы отдела продаж.

Актуальность выполняемой работы обусловлена тем, что в процессе продаж менеджерам требуется удобный и централизованный инструмент для учета автомобилей в наличии. Существующие методы (таблицы, разрозненные фотоальбомы) затрудняют быстрый поиск, обсуждение особенностей авто и ведение внутренней коммуникации. Автоматизация данного процесса позволяет повысить эффективность работы отдела продаж и качество обслуживания клиентов.

В ходе практики был выполнен полный цикл работ: от анализа предметной области и формулировки технического задания до реализации полнофункционального веб-приложения с использованием современных технологий.

1. Характеристика предприятия и места практики

1.1. Общая характеристика Группы компаний «ТрансТехСервис»

Компания «ТрансТехСервис» была основана в 1992 году в Набережных Челнах. Сегодня это динамично развивающаяся группа компаний, являющаяся официальным дилером ведущих автомобильных брендов. География присутствия охватывает несколько регионов России, включая Республику Башкортостан, со штаб-квартирой в г. Уфе.

Основные виды деятельности:

* реализация новых автомобилей и автомобилей с пробегом.
* полный комплекс сервисного обслуживания и ремонта.
* продажа оригинальных запасных частей и аксессуаров.
* предоставление сопутствующих услуг: страхование (КАСКО, ОСАГО), кредитование, лизинг.

1.2. Организационная структура и отдел информационных технологий

Организационная структура ГК «ТрансТехСервис» является дивизиональной. Упрощенная структура головного офиса представлена на рисунке 1.2.1

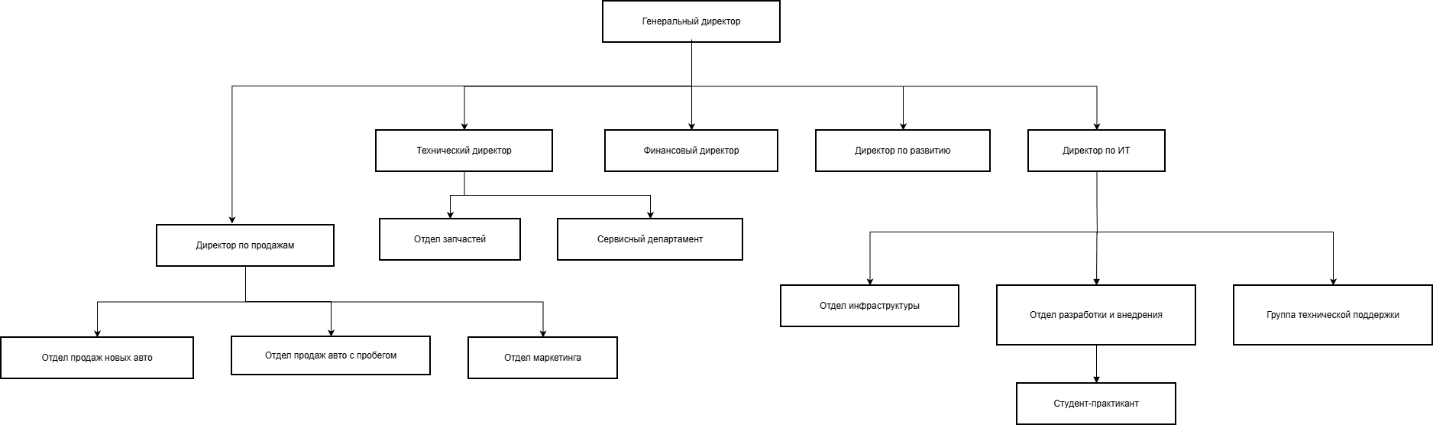


Рисунок 1.2.1 – Упрощенная организационная структура головного офиса ГК «ТТС»

Отдел информационных технологий (ИТ-отдел) является центральным звеном, обеспечивающим цифровизацию всех бизнес-процессов компании. В рамках отдела практика проходила в группе разработки и внедрения.

Основные функции группы разработки и внедрения:

* разработка, доработка и интеграция внутреннего программного обеспечения.
* автоматизация операционных и управленческих процессов.
* техническая поддержка и развитие корпоративных информационных систем.
* создание инструментов для аналитики данных.

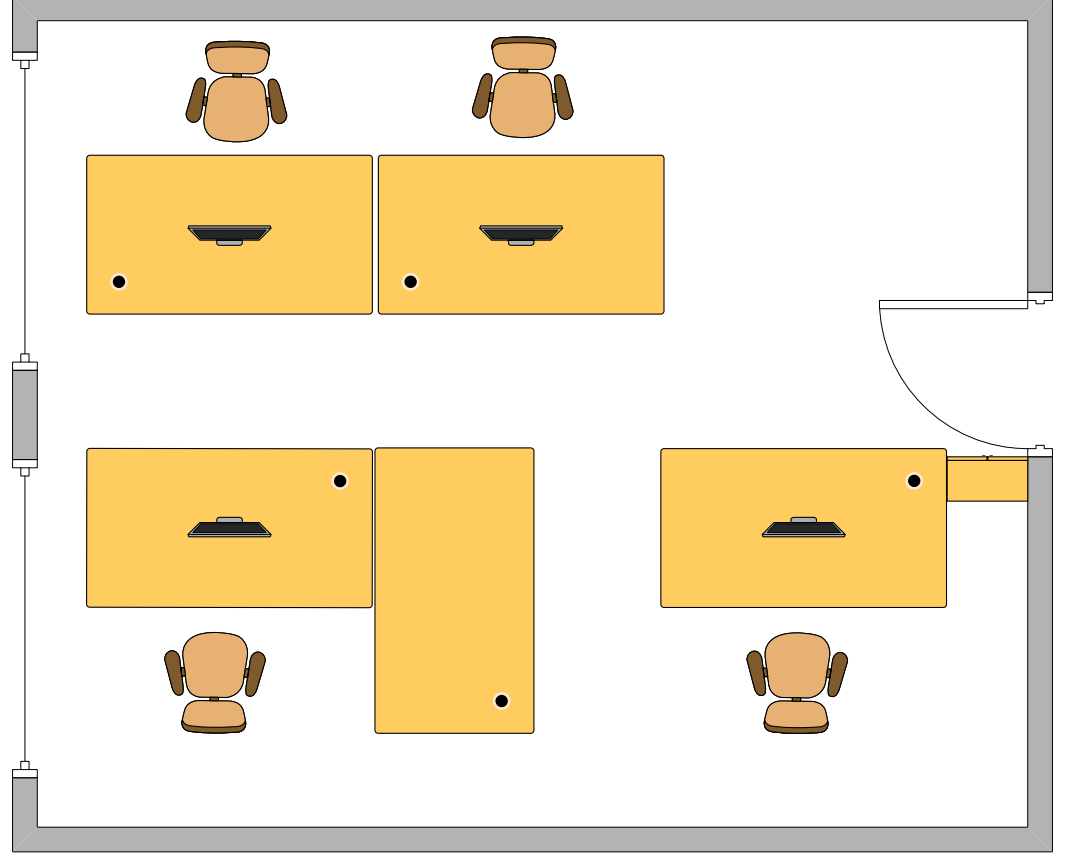
Рабочее место в отделе оснащено всем необходимым для выполнения задач по программированию и анализу. Схематичное расположение рабочих мест в секторе разработки представлена на рисунке 1.2.2 

Рисунок 1.2.2 – Схематичное расположение рабочих мест в секторе разработки ИТ-отдела

2. Анализ аппаратного и программного обеспечения

2.1. Анализ аппаратного обеспечения рабочего места

Для выполнения задач разработки было предоставлено рабочее место на базе высокопроизводительного персонального компьютера, собранного из современных компонентов, что обеспечивало комфортную работу со средами разработки, контейнеризацией и требовательными к ресурсам приложениями.

Конфигурация системного блока:

* операционная система: Microsoft Windows 11 Pro for Workstations.
* центральный процессор (CPU): AMD Ryzen 5 7500F (6 ядер, 12 потоков, базовая частота 3.7 ГГц).
* материнская плата: MSI PRO A620M-E.
* оперативная память (ОЗУ): KingBank 32 ГБ DDR5.
* системный накопитель: Gudga SSD M.2 NVMe, 1 ТБ.
* графический процессор (GPU): Palit NVIDIA GeForce RTX 4060 Dual OC, 8 ГБ GDDR6.

Периферийные устройства и ввод/вывод:

* монитор: ZOWIE by BenQ XL2540K (24.5”, разрешение Full HD (1920x1080), частота обновления 240 Гц).
* клавиатура: MadLions MAD 60HE.
* мышь: Lamzu Atlantis Mini.

Данная конфигурация обеспечивала высокую производительность при сборке проектов, работе с виртуальными окружениями, одновременном запуске нескольких экземпляров СУБД и веб-серверов, а также при тестировании интерфейсов. Использование NVMe-накопителя значительно сокращало время загрузки операционной системы и тяжелых сред разработки.

2.2. Анализ программного обеспечения и сетевой инфраструктуры

На рабочей станции было установлено следующее системное и прикладное программное обеспечение:

* операционная система: Microsoft Windows 11 Pro for Workstations.
* средства разработки: Visual Studio Code, VS Code, MySQL Worckbench.

Сетевая инфраструктура отдела:  
IT-отдел состоит из 4 рабочих станций, подключённых к глобальной сети интернет через маршрутизатор. Маршрутизатор обеспечивает:

* внутреннее взаимодействие между ПК
* подключение к глобальной сети Интернет (WAN)
* централизованное управление трафиком и базовую защиту периметра

Такая архитектура отвечает требованиям к масштабируемости, упрощенному управлению и минимизации точек отказа. Низкая задержка передачи данных критична для разработки и тестирования ПО, совместной работы с репозиториями и внешними сервисами. Схема сетевого обеспечение IT-отдела представлена на рисунке 2.2.1

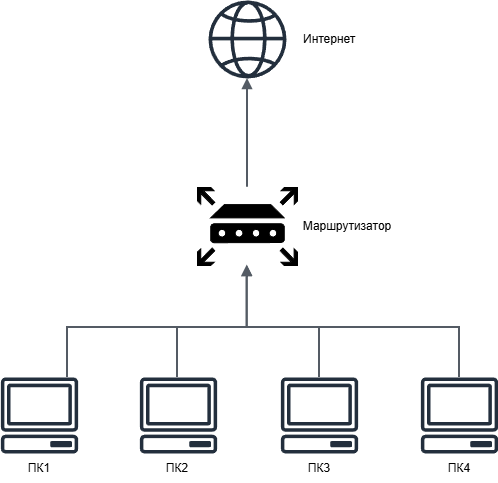


Рисунок 2.2.1 — Сетевое обеспечение IT-отдела

2.3. Анализ антивирусного обеспечения

Для обеспечения информационной безопасности инфраструктуры IT-отдела ГК «ТрансТехСервис» целесообразно внедрение корпоративного антивирусного комплекса. Основные задачи антивирусного ПО:

* защита рабочих станций и серверов от вирусов, троянов, руткитов и других вредоносных программ
* контроль интернет-трафика и почтовых сообщений
* централизованное управление политиками безопасности и обновлениями

Подходящим решением стал: Kaspersky Endpoint Security.

Ключевые возможности антивирусного комплекса:

* комплексное обнаружение угроз (сигнатурный и поведенческий анализ)
* прогностическая защита от новых, ранее неизвестных угроз
* многоуровневая проверка данных: файловая система, почта, приложения
* контроль интернет-активности: блокировка фишинговых сайтов и вредоносного трафика
* автоматическое обновление баз и систем без снижения производительности рабочих станций

Внедрение антивирусного решения позволяет минимизировать риски утечек данных, сбоев в работе критичных систем и финансовых потерь, связанных с кибератаками.

2.4. Настройка защиты системы стандартными средствами ОС

2.4.1 Брандмауэр в операционной системе Windows

Брандмауэр (межсетевой экран) Windows представляет собой встроенный механизм защиты операционной системы, предназначенный для контроля и фильтрации входящего и исходящего сетевого трафика на основе заданных правил безопасности. Основной задачей брандмауэра является предотвращение несанкционированного доступа к рабочей станции, защита от сетевых атак, вредоносного программного обеспечения и утечек данных.

Для доступа к настройкам брандмауэра в операционной системе Windows необходимо открыть «Панель управления» и перейти в раздел «Брандмауэр Защитника Windows». В данном разделе пользователь может включать или отключать межсетевой экран, а также настраивать параметры защиты для различных типов сетей: доменной, частной и общедоступной (рисунок 2.4.1.1).

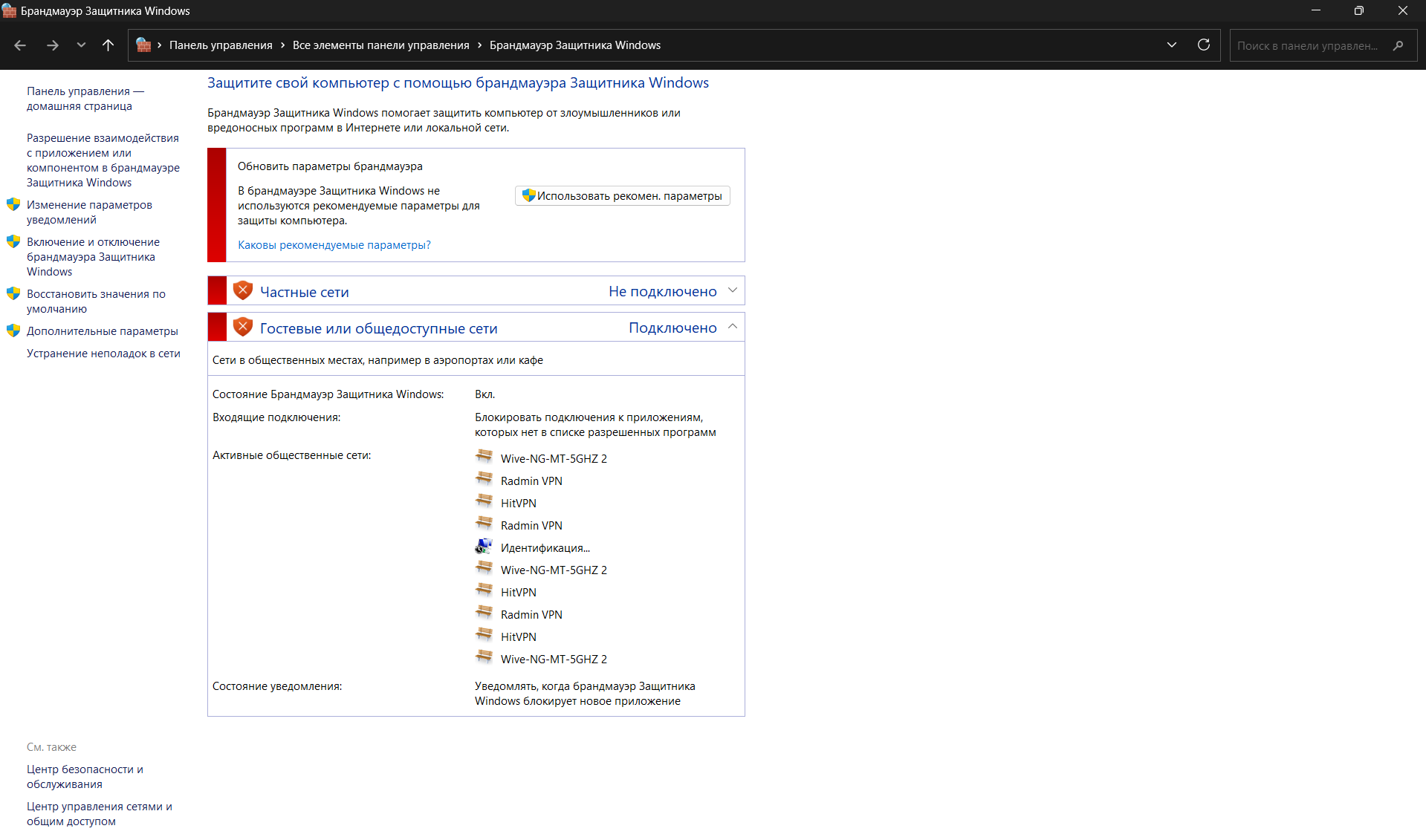
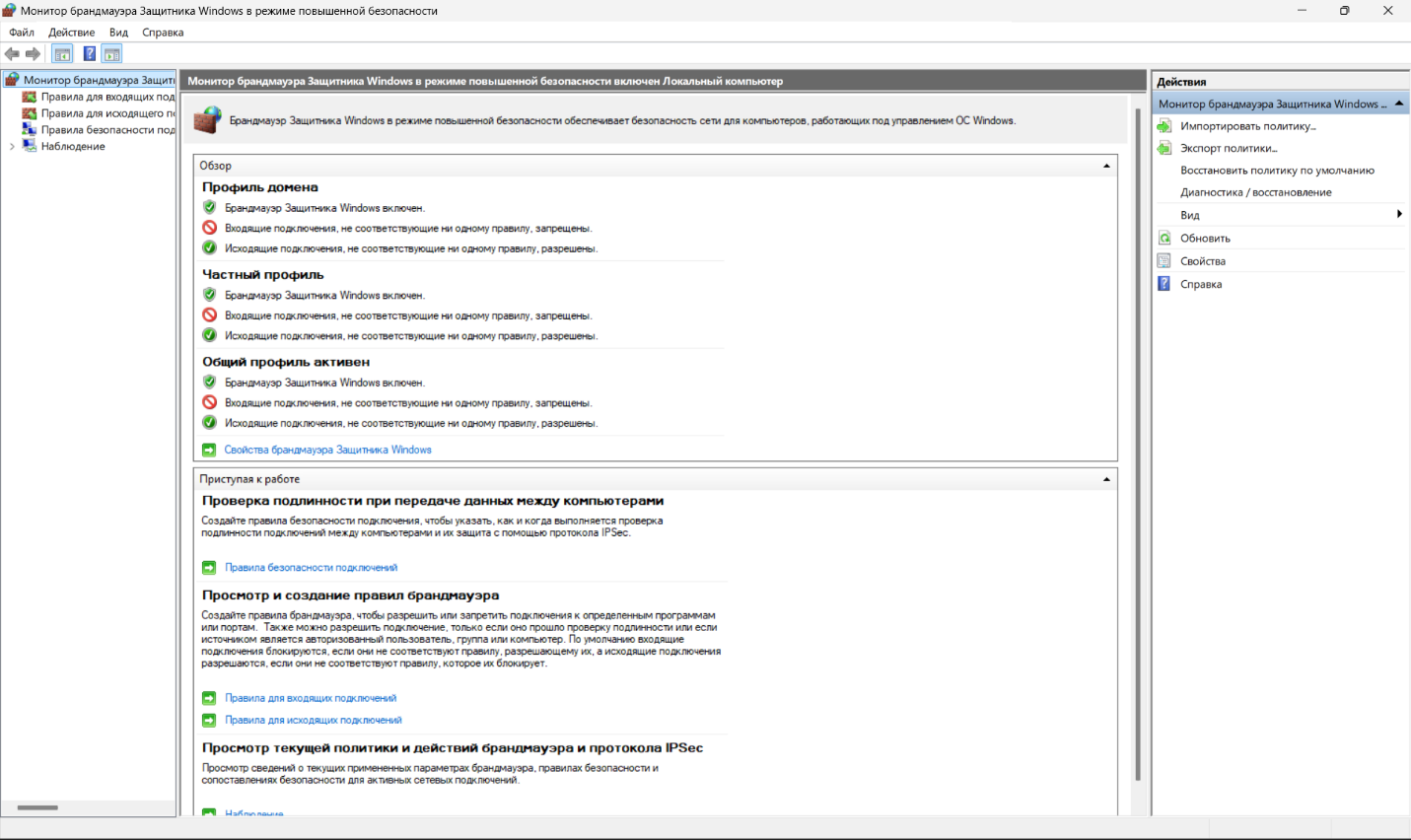


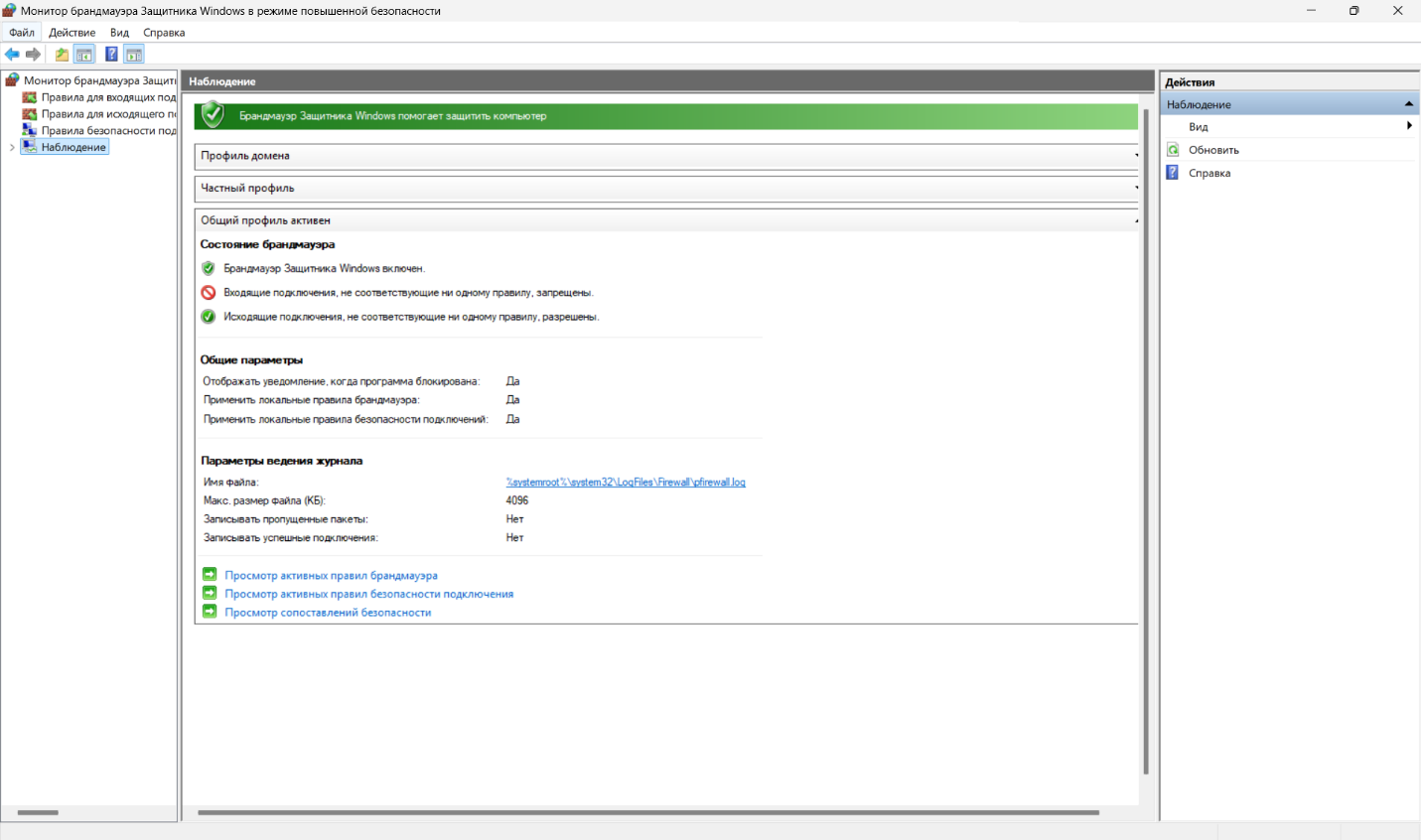
Рисунок 2.4.1.1 – Раздел настроек брандмауэра Защитника Windows

При включённом брандмауэре все входящие подключения по умолчанию блокируются, что значительно снижает риск несанкционированного доступа к системе. Разрешение сетевых соединений возможно только для доверенных приложений и сервисов, таких как локальный веб-сервер или система управления базами данных. Дополнительно пользователь может включить уведомления, которые отображаются при попытках приложений получить доступ к сети.

В разделе дополнительных параметров безопасности пользователь может создавать и редактировать правила для входящих и исходящих подключений. Данный интерфейс позволяет точно определить, какие приложения и службы имеют доступ к сетевым ресурсам (рисунок 2.4.1.2).

Рисунок 2.4.1.2 – Дополнительные параметры безопасности брандмауэра Windows

Для контроля текущей сетевой активности и применяемых правил используется встроенный монитор брандмауэра. В данном разделе отображаются активные подключения, задействованные правила и события безопасности, что позволяет оперативно выявлять подозрительную активность (рисунок 2.4.1.3).

Рисунок 2.4.1.3 – Мониторинг сетевых подключений в брандмауэре Windows

2.4.2. Настройка брандмауэра в Linux (UFW)

В Linux-дистрибутивах (Ubuntu/Debian) для управления межсетевым экраном часто используется UFW — Uncomplicated Firewall, который упрощает настройку правил и делает их более наглядными.

Преимущества UFW:

* простота использования (удобный синтаксис команд)
* быстрое создание правил для приложений, портов и сетевых интерфейсов
* возможность настройки по умолчанию («deny all» для входящих, «allow all» для исходящих)
* поддержка логирования и мониторинга активности

Проверка состояния межсетевого экрана

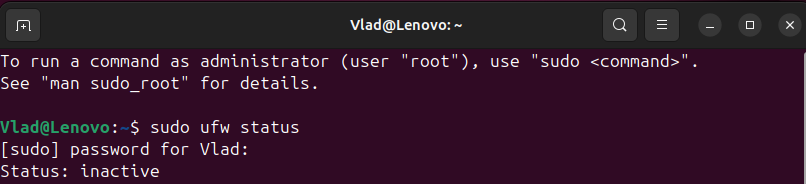
Для определения текущего состояния брандмауэра UFW и просмотра активных правил используется команда: sudo ufw status.  
В результате выполнения команды в терминале отображается текущее состояние службы. Если межсетевой экран отключён, выводится сообщение Status: inactive, что означает отсутствие активной фильтрации сетевого трафика. Данная команда применяется на начальном этапе настройки для оценки текущего уровня защищённости системы.

Рисунок 2.4.2.1 – Проверка состояния межсетевого экрана UFW

Активация брандмауэра UFW

Для включения межсетевого экрана и автоматического запуска службы при старте операционной системы используется команда: sudo ufw enable

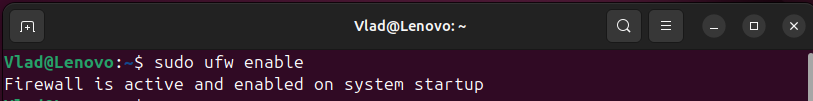
После выполнения команды UFW активируется и начинает применять правила фильтрации сетевого трафика. В терминале отображается сообщение о том, что брандмауэр включён и будет автоматически запускаться при загрузке системы. Это является обязательным шагом при вводе системы в эксплуатацию.

Рисунок 2.4.2.2 – Включение межсетевого экрана UFW

Деактивация брандмауэра

В случае необходимости временного отключения фильтрации сетевого трафика (например, для диагностики сетевых проблем) используется команда: sudo ufw disable

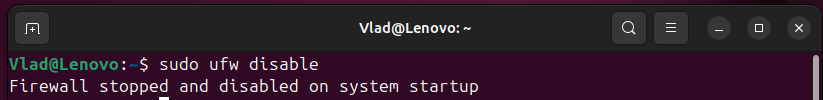
После выполнения команды межсетевой экран прекращает работу, а все правила временно перестают применяться. Использование данной команды рекомендуется только в контролируемых условиях, так как система остаётся без сетевой защиты.

Рисунок 2.4.2.3 – Отключение межсетевого экрана UFW

Установка базовой политики безопасности

Ключевым этапом настройки брандмауэра является определение политики по умолчанию. В соответствии с принципами информационной безопасности рекомендуется запрещать все входящие соединения и разрешать исходящие.

Для этого выполняются следующие команды: sudo ufw default deny incoming; sudo ufw default allow outgoing

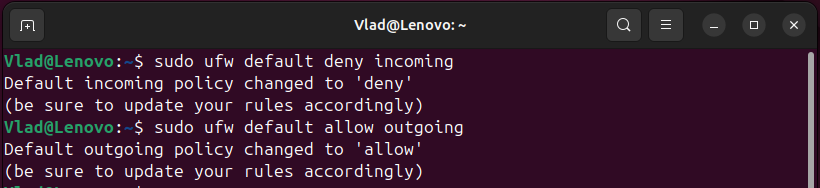
Такая конфигурация гарантирует, что устройство будет недоступно для несанкционированных подключений извне, при этом сохраняя возможность выхода в интернет и доступа к обновлениям и облачным сервисам.

Рисунок 2.4.2.4 – Настройка базовой политики безопасности UFW

Разрешение сетевых портов

Для корректной работы служб и удалённого администрирования необходимо явно разрешить доступ к определённым портам. В рамках IT-отдела используются следующие правила: sudo ufw allow 3306/tcp, sudo ufw allow 80,443/tcp

* порт 3306/tcp — применяется для работы с СУБД MySQL;
* порт 80/tcp — обеспечивают работу веб-сервисов по протоколам HTTP.

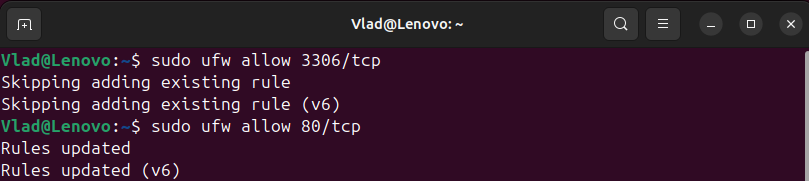
Добавленные правила позволяют сервисам функционировать без нарушения политики безопасности.

Рисунок 2.4.2.5 – Разрешение сетевых портов в UFW

Удаление правил фильтрации

В случае ошибочного добавления правила или при отсутствии необходимости в открытом доступе к определённому сервису используется команда удаления правила: sudo ufw delete allow 80

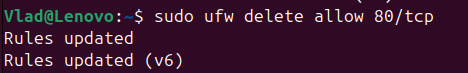
После выполнения команды соответствующее правило удаляется из конфигурации брандмауэра, а доступ к указанному порту блокируется. Это позволяет гибко управлять настройками безопасности и своевременно закрывать неиспользуемые сетевые сервисы.

Рисунок 2.4.2.6 – Удаление правила межсетевого экрана UFW

3. Проектирование Программного Обеспечения Для Решения Прикладной Задачи

3.1 Постановка задачи. Техническое задание на разработку программного продукта

3.1.1 Введение  
В рамках производственной практики была поставлена задача разработки веб-приложения «Каталог автомобилей ТТС» для автоматизации бизнес-процессов отдела продаж Группы компаний «ТрансТехСервис». Существовала необходимость в удобном инструменте для учёта автомобилей, онлайн-бронирования клиентами и централизованного управления инвентарём.

3.1.2 Основания для разработки  
Основанием для разработки программного продукта послужило задание на производственную практику по модулю ПМ.04, а также практическая потребность предприятия в автоматизации процессов учёта и бронирования автомобилей.

Разработка программного продукта выполнена в соответствии с требованиями предприятия с использованием современных веб-технологий. Архитектура и технологический стек были определены с учётом задач масштабируемости, безопасности и удобства сопровождения системы.

3.1.3 Назначение разработки  
Целью разработки является создание безопасного, интуитивно понятного и функционального веб-приложения, которое позволит:

* клиентам: просматривать каталог автомобилей, осуществлять онлайн-бронирование, управлять личным профилем и историей заказов.
* администраторам: управлять автомобилями (добавление, редактирование, удаление), отслеживать статусы (в наличии, забронировано, продано).

3.1.4 Требования к программе

Функциональные требования:

* + система регистрации и аутентификации пользователей (JWT).
  + публичный каталог автомобилей с фильтрацией (по цене, году, статусу, цвету) и сортировкой.
  + детальная страница автомобиля с условиями бронирования.
  + механизм бронирования на срок от 1 до 7 дней с автоматическим расчётом стоимости (1% от цены авто в день).
  + личный кабинет пользователя: просмотр истории бронирований, редактирование профиля (ФИО, телефон), смена пароля.
  + административная панель: CRUD-операции (создание, чтение, обновление, удаление) для автомобилей.

Нефункциональные требования:

* + клиентская часть: SPA-приложение на Vue.js с адаптивным интерфейсом.
  + серверная часть: RESTful API на C# (ASP.NET Core WebAPI).
  + база данных: MySQL.
  + безопасность: аутентификация по JWT-токенам, хеширование паролей.
  + производительность: отклик интерфейса < 500 мс при стандартной нагрузке.

3.1.5 Требования к программной документации  
В состав документации входят настоящее техническое задание, описание программы, протокол тестирования и руководство пользователя.

3.2 Описание программы

3.2.1Архитектура  
Приложение построено по клиент-серверной архитектуре с использованием шаблона MVC на стороне сервера и многослойной структуры.

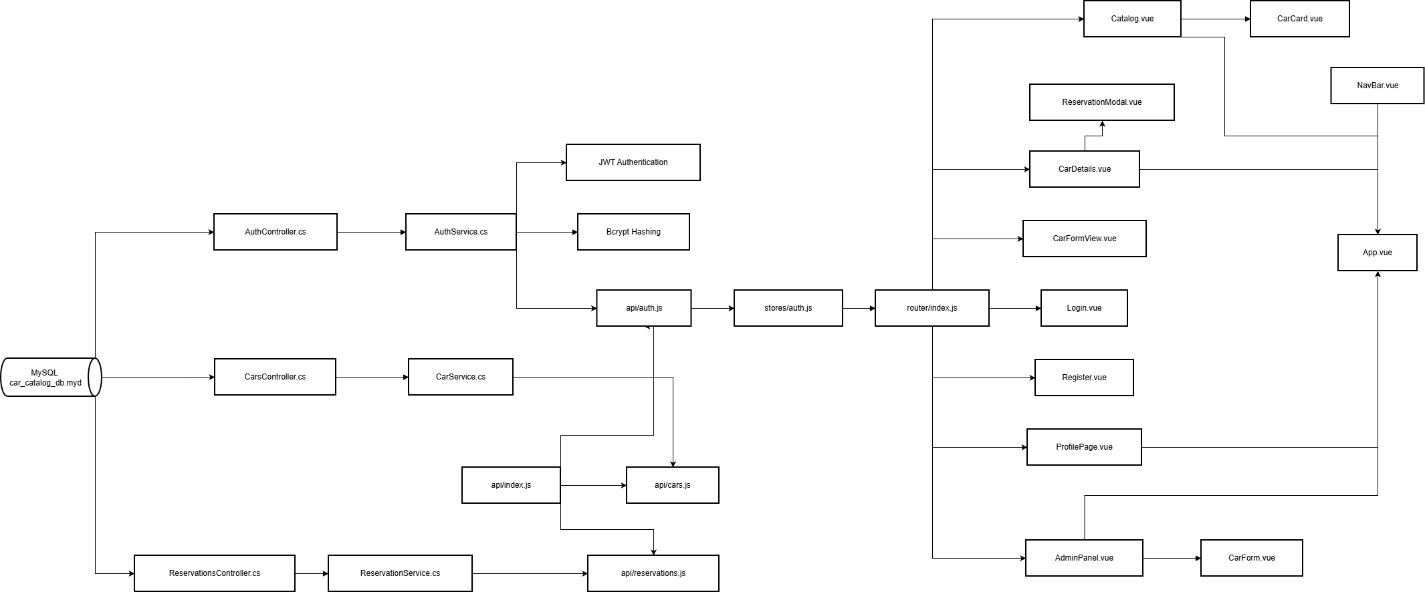
Модульная схема представлена на рисунке 3.2.1.1.

Рисунок 3.2.1.1 – Модульная схема

Таблица 3.2.1.1 – Описание модулей

|  |  |
| --- | --- |
| Модуль | Назначение |
| 1 | 2 |
| AuthController.cs | Реализация регистрации, аутентификации пользователей и генерации JWT-токенов |
| CarsController.cs | Управление каталогом автомобилей: получение, фильтрация, добавление, редактирование и удаление |

Продолжение таблицы 3.2.1.1

|  |  |
| --- | --- |
| 1 | 2 |
| ReservationsController.cs | Реализация логики бронирования автомобилей и управления статусами бронирований |
| AuthService.cs | Бизнес-логика аутентификации и авторизации пользователей |
| CarService.cs | Бизнес-логика работы с автомобилями и их фильтрации |
| ReservationService.cs | Бизнес-логика бронирования и расчёта стоимости |
| router/index.js | Маршрутизация и защита страниц по ролям пользователей |
| stores/auth.js | Управление состоянием аутентификации и хранение JWT-токена |
| api/auth.js | Взаимодействие клиентской части с API аутентификации |
| api/cars.js | Взаимодействие клиентской части с API каталога автомобилей |
| api/reservations.js | Взаимодействие клиентской части с API бронирований |

Описание процедур основных модулей представлено в таблице 3.2.1.2

Таблица 3.2.1.2 – Описание процедур

|  |  |  |
| --- | --- | --- |
| Модуль | Процедура | Назначение |
| 1 | 2 | 3 |
| AuthController.cs | Login() | |  | | --- | |  |  |  | | --- | | Аутентификация пользователя и выдача JWT-токена | |

Продолжение таблицы 3.2.1.2

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| AuthController.cs | Register() | |  | | --- | |  |  |  | | --- | | Регистрация нового пользователя в системе | |
| GetProfile() | Получение данных профиля авторизованного пользователя |
| UpdateProfile() | Обновление личных данных и пароля пользователя |
| CarsController.cs | GetAllCars() | |  | | --- | |  |  |  | | --- | | Получение списка автомобилей с фильтрацией и сортировкой | |
| GetCar() | Получение детальной информации об автомобиле |

Продолжение таблицы 3.2.1.2

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| CarsController.cs | CreateCar() | Добавление нового автомобиля (доступно администратору) |
| UpdateCar() | Редактирование информации об автомобиле |
| DeleteCar() | Удаление автомобиля из каталога |
| |  | | --- | |  |  |  | | --- | | ReservationsController.cs | | CreateReservation() | |  | | --- | |  |  |  | | --- | | Создание бронирования автомобиля | |
| GetMyReservations() | Получение истории бронирований текущего пользователя |
| GetReservation() | Получение информации о конкретном бронировании |

Продолжение таблицы 3.2.1.2

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| ReservationsController.cs | CancelReservation() | Отмена бронирования пользователем |
| GetCarReservations() | Получение списка бронирований конкретного автомобиля |
| UpdateReservationStatus() | Изменение статуса бронирования (администратор) |
| AuthService.cs | LoginAsync() | Проверка учетных данных пользователя |
| RegisterAsync() | |  | | --- | |  |  |  | | --- | | Создание нового пользователя и хеширование пароля | |
| GenerateJwtToken() | Формирование JWT-токена с пользовательскими claims |
| UpdateUserProfileAsync() | Обновление профиля пользователя с проверкой пароля |

Продолжение таблицы 3.2.1.2

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| |  | | --- | |  |  |  | | --- | | CarService.cs | | GetCarsAsync() | Формирование запроса к БД с учетом фильтров |
| GetCarAsync() | Получение информации об автомобиле из базы данных |
| CreateCarAsync() | Создание записи об автомобиле в БД |
| UpdateCarAsync() | Обновление данных автомобиля |
| DeleteCarAsync() | Удаление автомобиля из базы данных |
| ReservationsService.cs | CreateReservationAsync() | Проверка доступности и создание бронирования |
| GetUserReservationsAsync() | Получение бронирований пользователя |

Продолжение таблицы 3.2.1.2

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| ReservationsService.cs | GetReservationByIdAsync() | Получение бронирования по идентификатору |
| CancelReservationAsync() | Отмена бронирования и освобождение автомобиля |
| UpdateReservationStatusAsync() | Изменение статуса бронирования |

3.2.2 Модульная схема и диаграмма базы данных  
Система состоит из логически связанных модулей: аутентификации, каталога, бронирования, профиля и администрирования. Взаимосвязь сущностей в базе данных отражена на ER-диаграмме Рисунок 3.2.2.1.

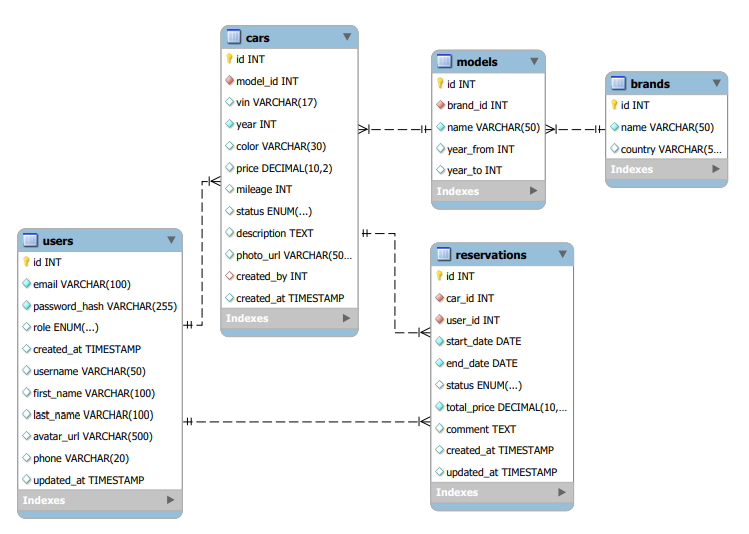
Таблица 3.2.2.1– Описание основных таблиц базы данных

Рисунок 3.2.2.1 – ER-диаграмма базы данных.

|  |  |  |
| --- | --- | --- |
| Таблица | Назначение | Ключевые поля |
| users | Хранение данных пользователей | id, email, password\_hash, role |
| cars | Каталог автомобилей | id, model\_id, year, price, status, vin |
| reservations | Бронирования | id, car\_id, user\_id, start\_date, end\_date, total\_price |
| brands | Справочник марок | id, name, country |
| models | Справочник моделей | id, brand\_id, name, year\_from, year\_to |

3.3 Протокол тестирования разработанного программного продукта

В ходе тестирования разработанного программного продукта были выполнены проверки корректности работы основных функциональных модулей системы на корректных и некорректных входных данных. Тестирование проводилось вручную по основным пользовательским сценариям, соответствующим требованиям технического задания.

В процессе тестирования не было выявлено ошибок, оказывающих критическое влияние на функционирование программного продукта и стабильность работы системы в целом. Все основные функции приложения работают корректно, обработка пользовательских действий и ошибок выполняется в соответствии с заложенной бизнес-логикой.

Разработанный программный продукт удовлетворяет всем предъявленным функциональным и нефункциональным требованиям, обладает удобным и интуитивно понятным пользовательским интерфейсом, а также обеспечивает корректную обработку исключительных ситуаций и защиту данных пользователей.

Таблица 3.3.1 – Тестирование регистрации нового пользователя

|  |  |  |
| --- | --- | --- |
| Поле | Описание | |
| 1 | 2 | |
| Дата теста | 20.12.2025 | |
| Приоритет | Высокий | |
| Заголовок теста | Регистрация пользователя с корректными данными | |
| Резюме испытания | Проверка создания учётной записи | |
| Этапы теста | 1 Запустить сервер и веб приложение;  2 нажать на кнопку «Зарегистрироваться»;  3 ввод почты в поле для ввода «Email»;  4 ввод пароля в поле для ввода «Пароль»;  5 ввод пароля в поле для ввода «Повторите пароль»;  4 нажатие на кнопку «Войти». | |
| Тестовые данные | 1 Email: qwe@test.com  2 Пароль: Qwerty123! | |
| Ожидаемый результат | Отображение уведомления «Регистрация прошла успешно! Перенаправление на главную страницу...», перенаправление на главную страницу. |
| Фактический результат | Соответствует ожидаемому. Отобразилось уведомление и перенаправило на главную. |

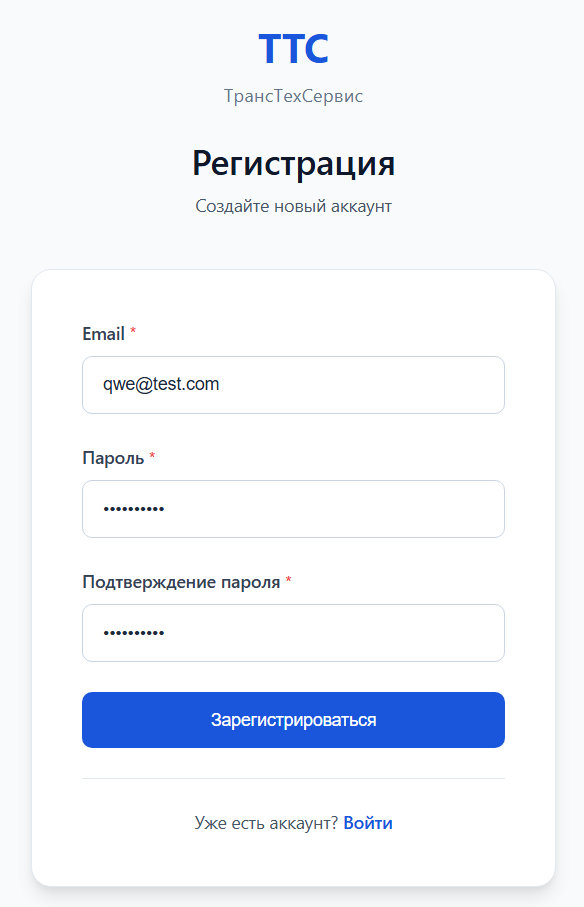
**Рисунок 3.3.1 – Тестирование регистрации нового пользователя

Таблица 3.3.2 – Тестирование входа в систему

|  |  |
| --- | --- |
| Поле | Описание |
| Дата теста | 20.12.2025 |
| Приоритет | Высокий |
| Заголовок теста | Вход пользователя с корректными данными |
| Резюме испытания | Проверка аутентификации пользователя |
| Этапы теста | 1 Запустить сервер и веб приложение;  2 ввод почты в поле для ввода «Email»;  3 ввод пароля в поле для ввода «Пароль»;  4 нажатие на кнопку «Войти». |
| Тестовые данные | 1 Email: qwe@test.com  2 Пароль: Qwerty123! |
| Ожидаемый результат | Успешный вход, переход на главную страницу |
| Фактический результат | Соответствует ожидаемому. Успешная авторизация, переход на главную страницу |

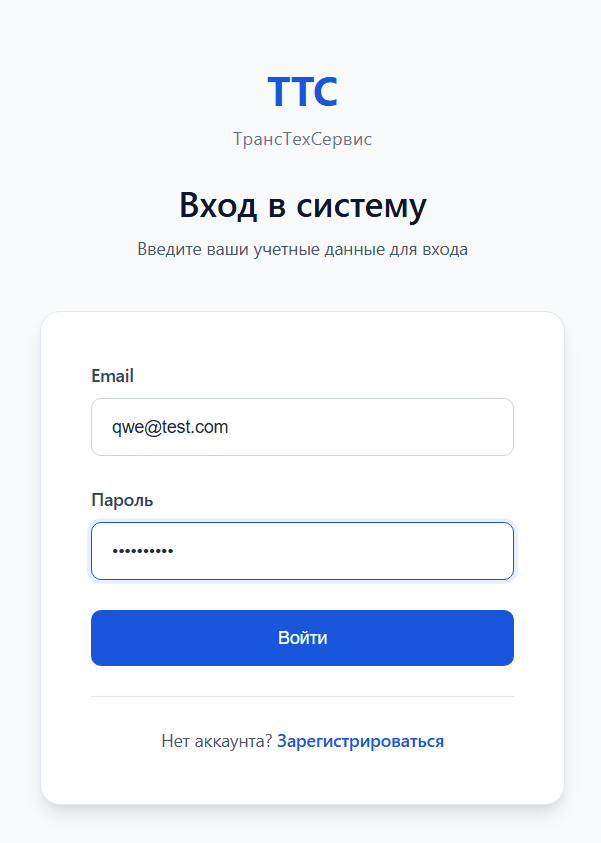
Рисунок 3.3.2 - Тестирование входа в систему

Таблица 3.3.3 – Тестирование бронирования автомобиля

|  |  |
| --- | --- |
| Поле | Описание |
| 1 | 2 |
| Дата теста | 20.12.2025 |
| Приоритет | Высокий |
| Заголовок теста | Бронирование автомобиля на допустимый срок |
| Резюме испытания | Проверка создания бронирования |

Продолжение таблицы 3.3.3

|  |  |
| --- | --- |
| 1 | 2 |
| Этапы теста | 1 Запустить сервер и веб приложение;  2 ввод почты в поле для ввода «Email»;  3 ввод пароля в поле для ввода «Пароль»;  4 нажатие на кнопку «Войти»;  5 нажатие на кнопку «Забронировать»;  6 выбрать дату окончания брони;  7 нажатие на кнопку «Забронировать». |
| Тестовые данные | Дата начала: 21.12.2025, Дата окончания: 22.12.2025 |
| Ожидаемый результат | Бронь создана, статус автомобиля изменён на «reserved» |
| Фактический результат | Соответствует ожидаемому. Автомобиль был забронирован и статус изменился на «reserved» |

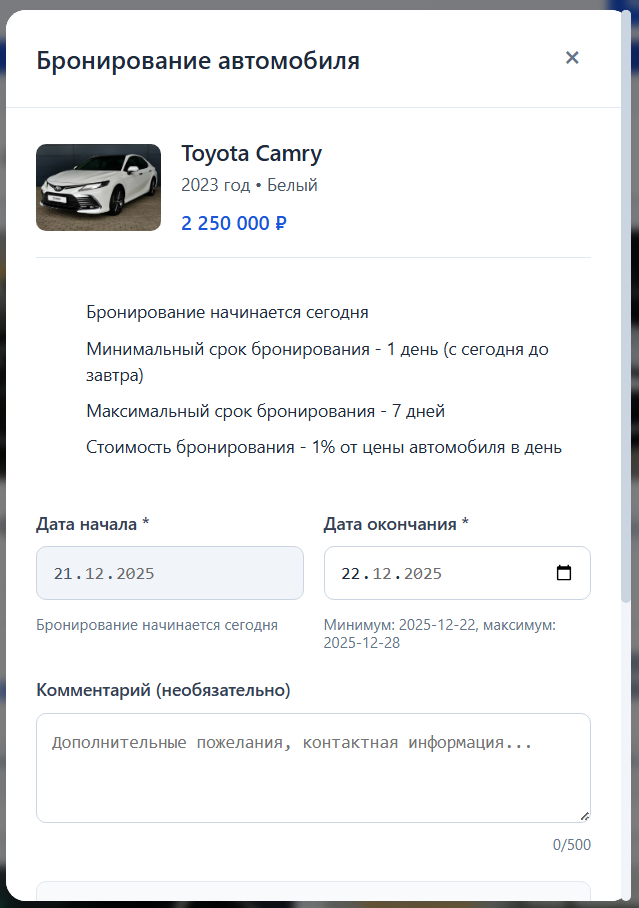
Рисунок 3.3.3 - Тестирование бронирования автомобиля

Таблица 3.3.4 – Тестирование фильтрации каталога

|  |  |
| --- | --- |
| Поле | Описание |
| 1 | 2 |
| Дата теста | 20.12.2025 |
| Приоритет | Средний |
| Заголовок теста | Фильтрация автомобилей по параметрам |
| Резюме испытания | Проверка работы фильтров каталога |
| Этапы теста | 1 Запустить сервер и веб приложение;  2 ввод почты в поле для ввода «Email»;  3 ввод пароля в поле для ввода «Пароль»;  4 нажатие на кнопку «Войти»;  5 ввести в фильтры цену и год авто. |

Продолжение таблицы 3.3.4

|  |  |
| --- | --- |
| 1 | 2 |
| Тестовые данные | Цена: 1–3 млн, Год: от 2020 |
| Ожидаемый результат | Отображаются автомобили, соответствующие условиям |
| Фактический результат | Соответствует ожидаемому. Отобразились автомобили соответствующие условиям, всего 9 автомобилей |

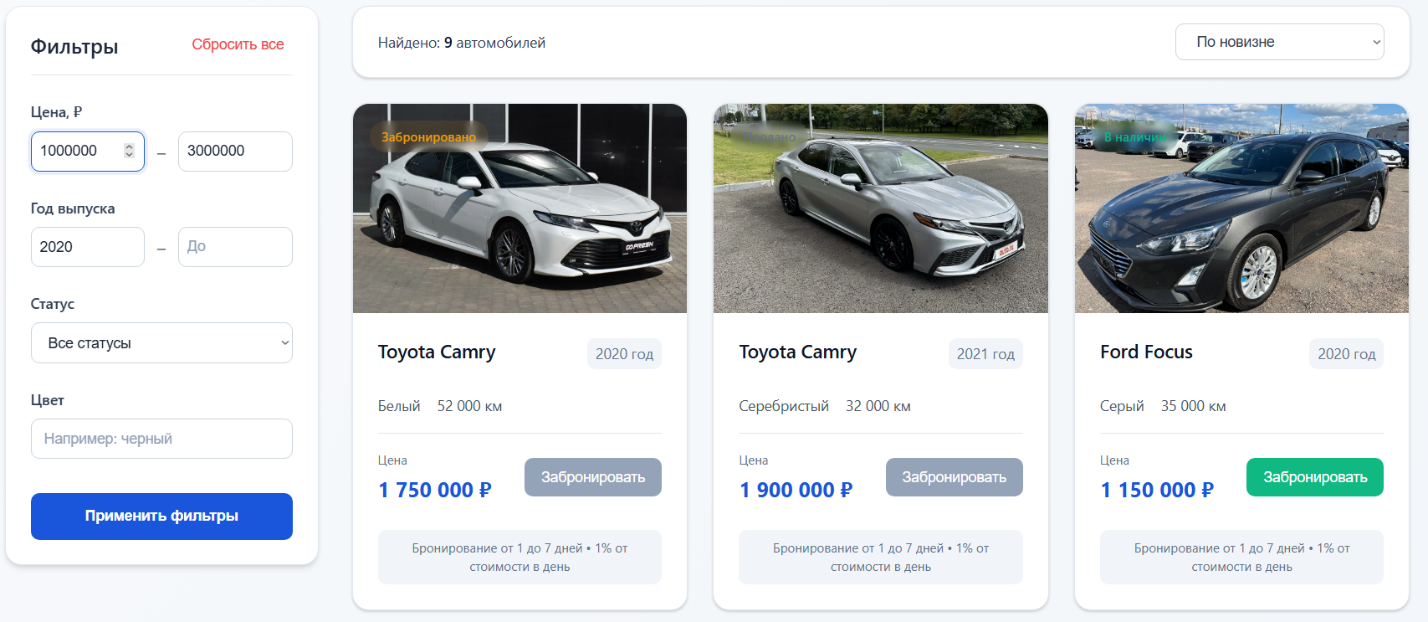


Рисунок 3.3.4 - Тестирование фильтрации каталога

Таблица 3.3.5 – Тестирование добавления автомобиля (администратор)

|  |  |
| --- | --- |
| Поле | Описание |
| 1 | 2 |
| Дата теста | 20.12.2025 |
| Приоритет | Высокий |
| Заголовок теста | Добавление нового автомобиля |
| Резюме испытания | Проверка CRUD-функционала админ-панели |

Продолжение таблицы 3.3.5

|  |  |
| --- | --- |
| 1 | 2 |
| Этапы теста | 1 Запустить сервер и веб приложение;  2 ввод почты в поле для ввода «Email»;  3 ввод пароля в поле для ввода «Пароль»;  4 нажатие на кнопку «Войти»;  5 нажатие на кнопку «Админ-панель»;  6 нажатие на кнопку «Добавить автомобиль»;  7 выбор марки и модели авто, ввод года, цвета, VIN номера, описания, цены и пробега, url фотографии авто;  8 нажатие на кнопку «Добавить автомобиль». |
| Тестовые данные | Марка: «BMW»  Модель: «X5»  Год выпуска: «2023»  Цвет: «Красный»  VIN номер: «RJ2384F38FW8JESFJ»  Описание: «Семейное авто»  Цена: «3000000»  Пробег: «5000»  URL фотографии: «https://avatars.mds.yandex.net/i?id=c171ccb67eb9006a6befe27babdc0bbc7ccae701-4306623-images-thumbs&n=13» |
| Ожидаемый результат | Автомобиль добавлен и отображается в каталоге |
| Фактический результат | Соответствует ожидаемому. Автомобиль был добавлен и отображается в каталоге |

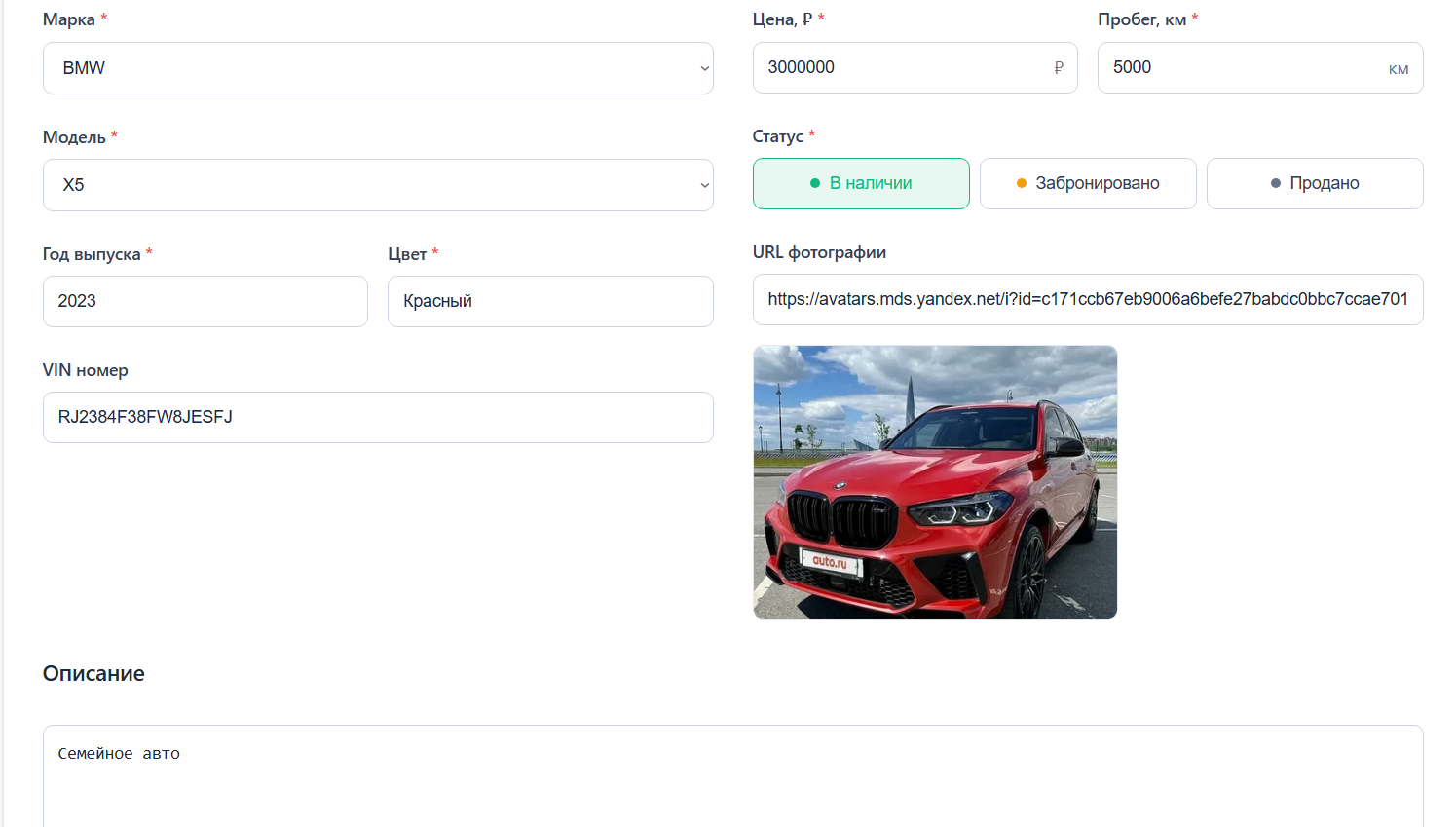
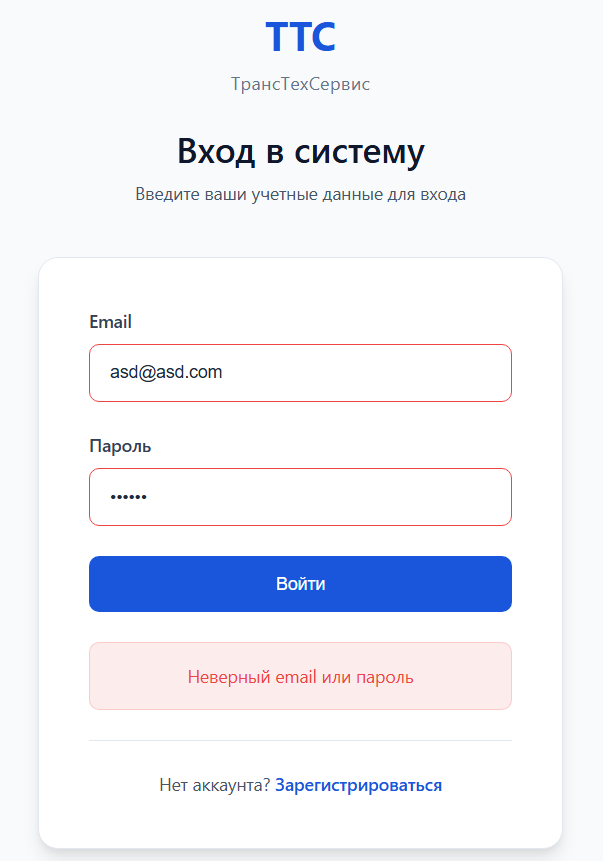
Рисунок 3.3.5 - Тестирование добавления автомобиля (администратор)

Таблица 3.3.6 – Тестирование входа в систему с некорректными данными

|  |  |
| --- | --- |
| Поле | Описание |
| Дата теста | 20.12.2025 |
| Приоритет | Высокий |
| Заголовок теста | Вход пользователя с некорректными данными |
| Резюме испытания | Проверка аутентификации пользователя |
| Этапы теста | 1 Запустить сервер и веб приложение;  2 ввод почты в поле для ввода «Email»;  3 ввод пароля в поле для ввода «Пароль»;  4 нажатие на кнопку «Войти». |
| Тестовые данные | 1 Email: asd@asd.com  2 Пароль: asdasd |
| Ожидаемый результат | Отображение уведомления «Неверный email или пароль». |
| Фактический результат | Соответствует ожидаемому. Отобразилось уведомление «Неверный email или пароль». |

Рисунок 3.3.6 - Тестирование входа в систему с некорректными данными

Вывод по тестированию: Все ключевые функции приложения работают корректно. Критических ошибок не обнаружено. Программный продукт соответствует требованиям технического задания.

3.4 Руководство пользователя

Основная цель руководства пользователя заключается в обеспечении пользователя необходимой информацией и пошаговыми инструкциями для корректной эксплуатации и самостоятельной работы с веб-приложением «Каталог автомобилей ТТС».

Программный продукт предназначен для пользователей, обладающих базовыми навыками работы с графическим интерфейсом и веб-браузером.

Начало работы с системой

Для начала работы необходимо запустить веб-приложение. На странице входа пользователь может ввести адрес электронной почты и пароль либо перейти к форме регистрации нового аккаунта (рисунок 3.4.1).

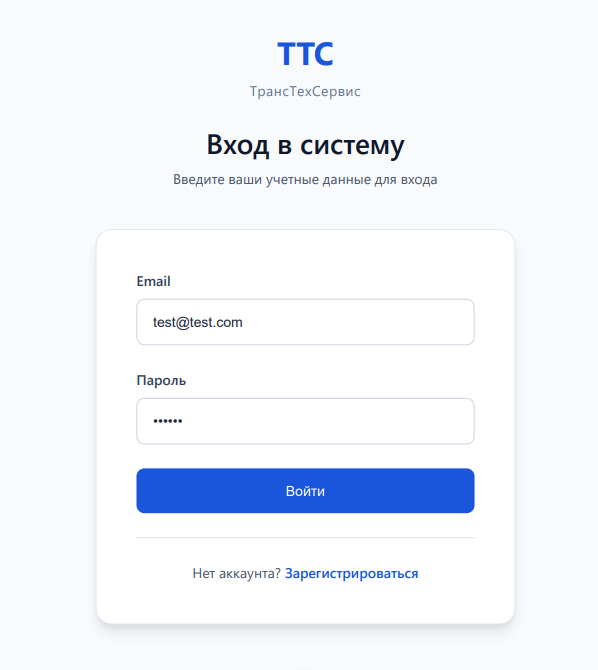


Рисунок 3.4.1 – Страница аутентификации пользователя

При отсутствии учетной записи необходимо нажать кнопку «Зарегистрироваться», после чего откроется форма создания нового пользователя (рисунок 3.4.2).

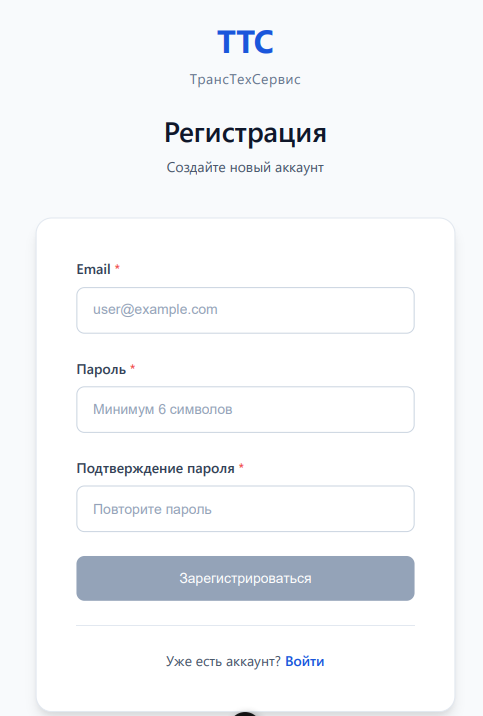


Рисунок 3.4.2 – Страница регистрации пользователя

Для успешной регистрации требуется корректно заполнить все обязательные поля формы. После регистрации пользователь может выполнить вход в систему, используя указанные учетные данные.

Вход в систему и навигация

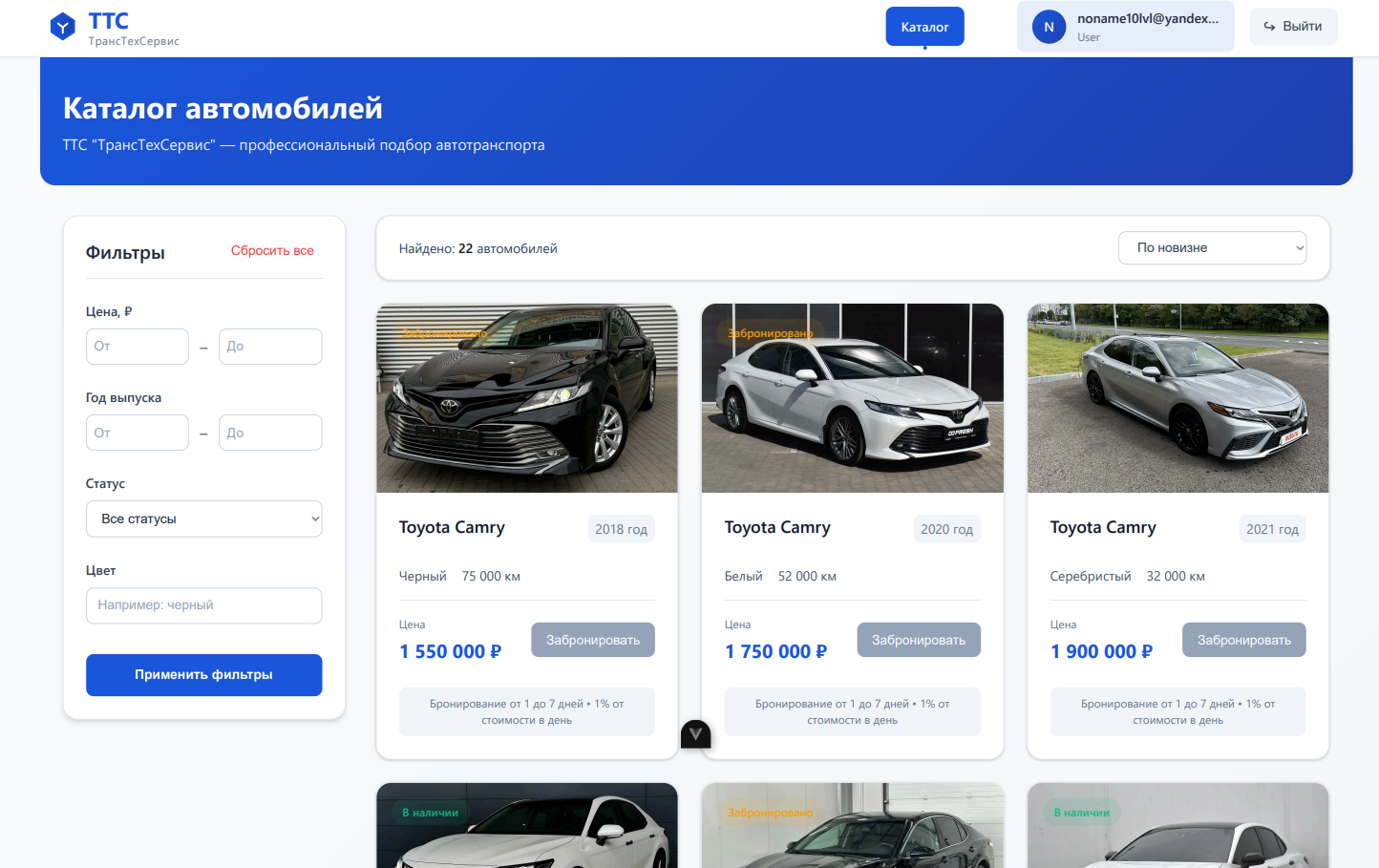
После успешной аутентификации пользователь автоматически перенаправляется на главную страницу каталога автомобилей (рисунок 3.4.3). В зависимости от роли пользователя (пользователь или администратор) доступен различный функционал системы.

Рисунок 3.4.3 – Главная страница каталога автомобилей

Навигация по системе осуществляется с помощью верхнего меню, в котором доступны следующие разделы:

* каталог автомобилей;
* административная панель (для пользователей с ролью Admin);
* профиль пользователя;
* выход из системы.

Работа с каталогом автомобилей

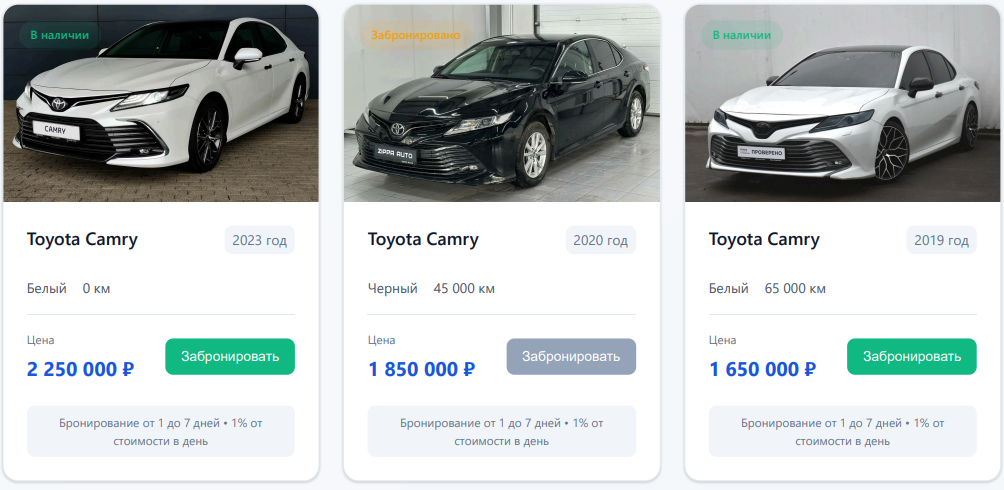
В разделе «Каталог» отображается список автомобилей, доступных в системе. Пользователь может просматривать карточки автомобилей с основной информацией: марка, модель, год выпуска, цена, статус, цвет, пробег и фотография (рисунок 3.4.4).

Рисунок 3.4.4 – Каталог автомобилей

Для удобства поиска предусмотрена система фильтрации и сортировки по различным параметрам (цена, год выпуска, статус, цвет). При выборе автомобиля пользователь может перейти на страницу с детальной информацией (рисунок 3.4.5).

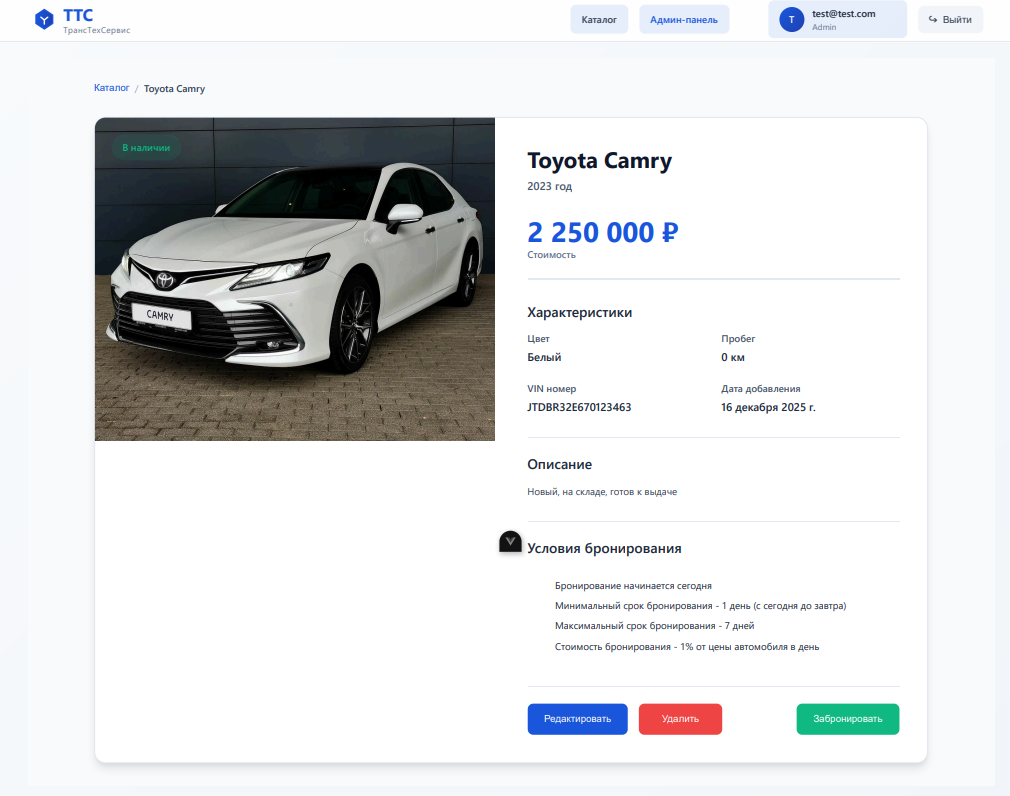


Рисунок 3.4.5 – Детальная страница автомобиля

Бронирование автомобиля

Для бронирования автомобиля пользователь нажимает кнопку «Забронировать» на карточке автомобиля или на его детальной странице. После этого открывается форма бронирования, в которой необходимо выбрать даты начала и окончания бронирования (от 1 до 7 дней) (рисунок 3.4.6).



Рисунок 3.4.6 – Форма бронирования автомобиля

Стоимость бронирования рассчитывается автоматически исходя из установленного тарифа. После подтверждения бронирования информация о нем сохраняется в системе, а статус автомобиля изменяется на «Забронирован».

Личный кабинет пользователя

В разделе «Профиль» пользователь может просматривать и редактировать свои персональные данные: электронную почту, имя, фамилию, номер телефона, а также выполнить смену пароля (рисунок 3.4.7).

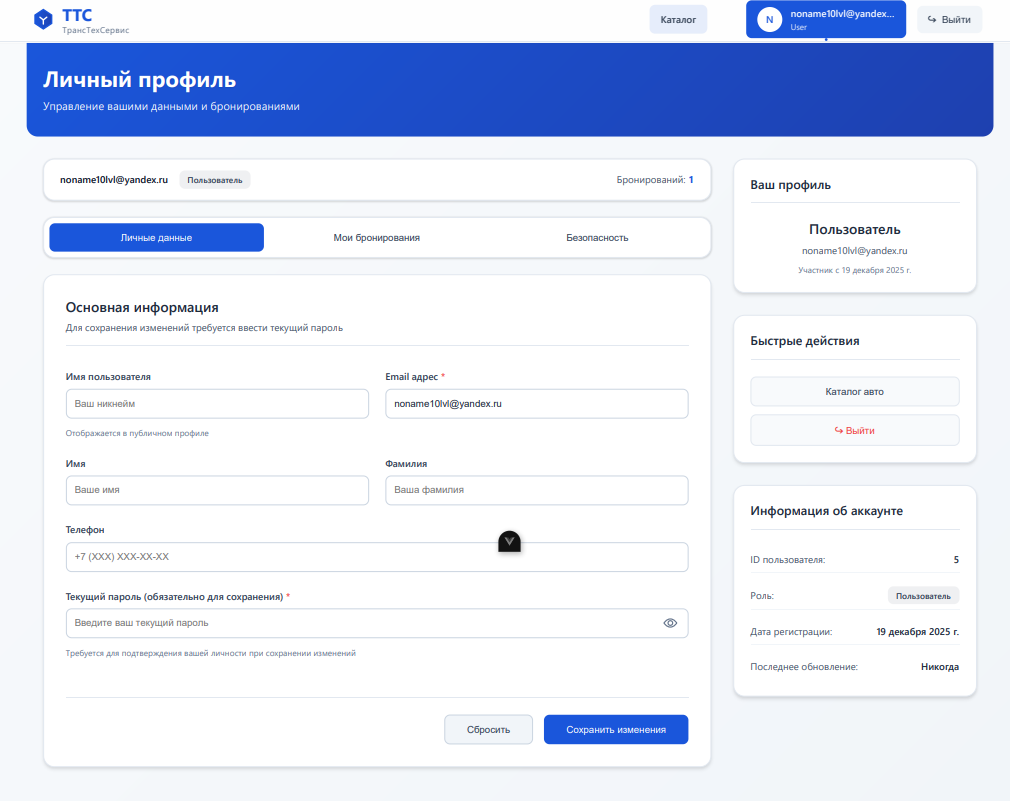


Рисунок 3.4.7 – Страница профиля пользователя

В разделе «История бронирований» отображается список всех бронирований пользователя с указанием автомобиля, дат бронирования (рисунок 3.4.8).

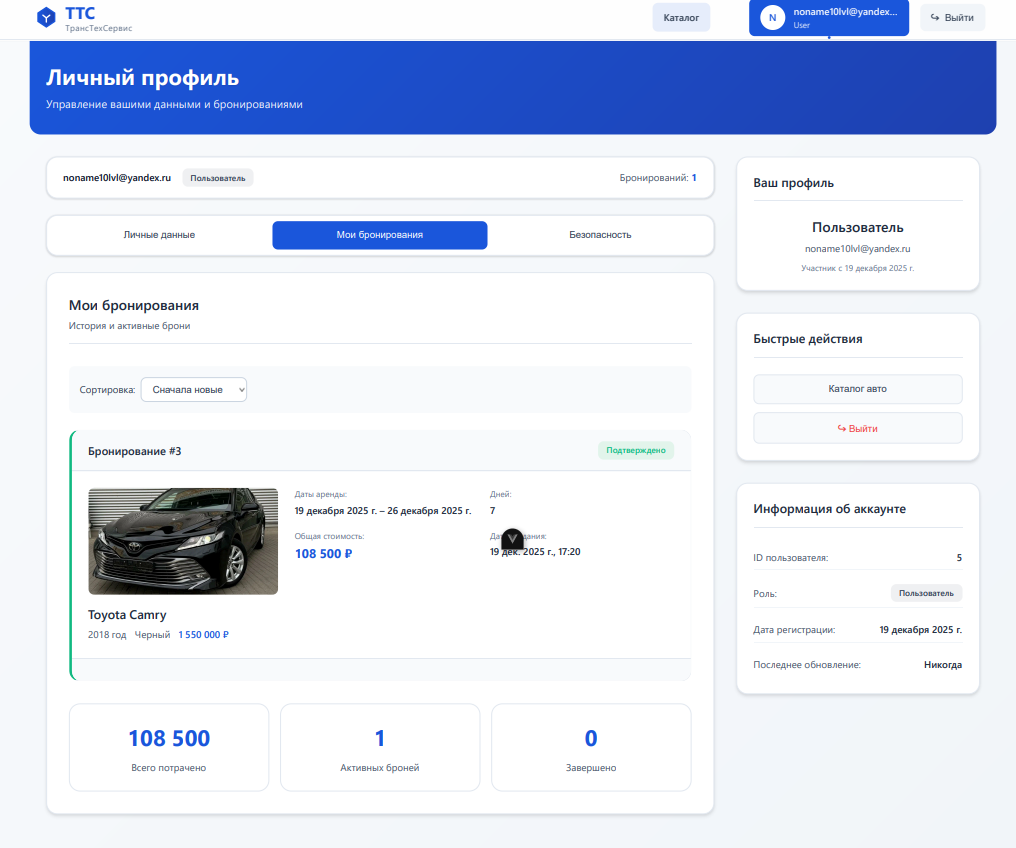


Рисунок 3.4.8 – История бронирований пользователя

Администрирование системы

Пользователи с ролью администратора имеют доступ к административной панели. В данном разделе администратор может:

* просматривать список всех автомобилей;
* добавлять новые автомобили;
* редактировать информацию об автомобилях;
* удалять автомобили из каталога.

Интерфейс административной панели представлен на рисунке 3.4.9

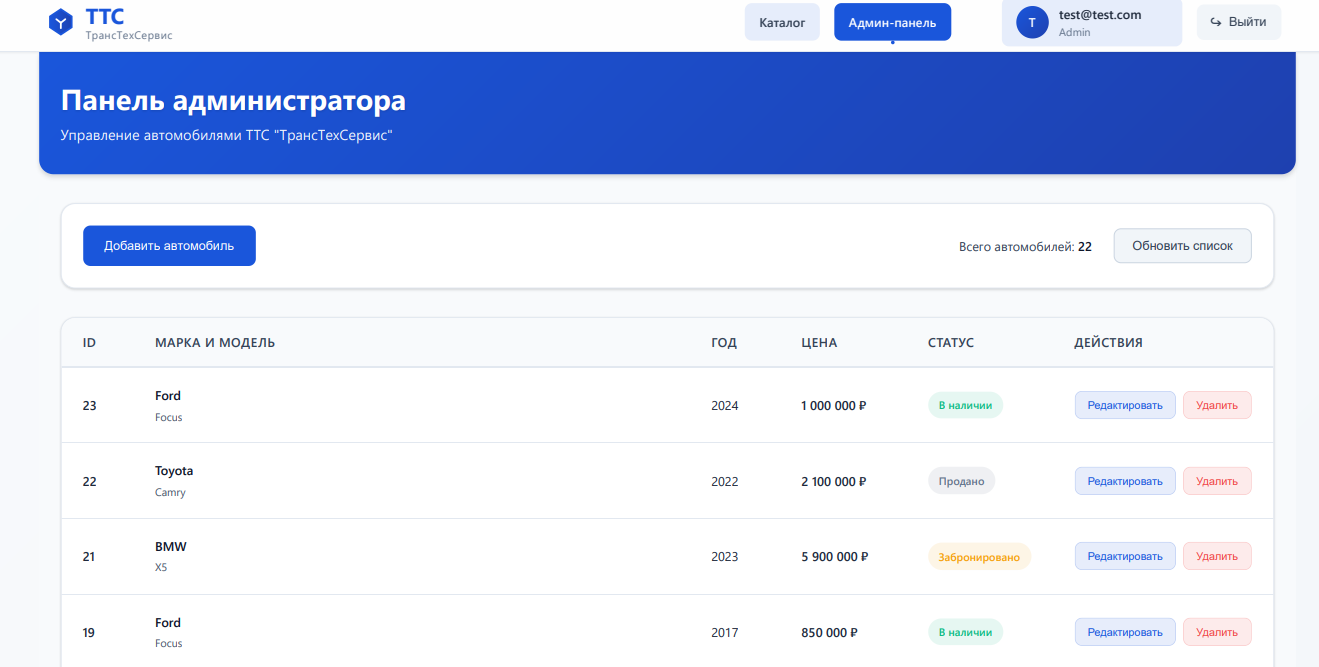


Рисунок 3.4.9 – Административная панель управления автомобилями

Завершение работы

Для завершения работы с системой необходимо нажать кнопку «Выйти» в навигационном меню. После выхода JWT-токен становится недействительным по истечении времени жизни, что обеспечивает безопасность пользовательских данных.

ЗАКЛЮЧЕНИЕ

Производственная практика по модулю ПМ.04 «сопровождение и обслуживание программного обеспечения компьютерных систем» была успешно пройдена в отделе информационных технологий ГК «ТрансТехСервис».

В ходе практики были выполнены все поставленные задачи:

* проведён анализ организационной структуры предприятия и его ит-инфраструктуры.
* изучены аппаратное и программное обеспечение рабочего места, политики информационной безопасности.
* практическим результатом стала полная разработка веб-приложения «Каталог автомобилей ТТС». Был реализован полный цикл: от анализа требований и проектирования базы данных до реализации клиентской части на vue.js, серверного api на c# asp.net core и интеграции jwt-аутентификации.

Разработанное приложение решает актуальную бизнес-задачу автоматизации учёта и бронирования автомобилей, предоставляя удобный интерфейс как для клиентов, так и для администраторов. В процессе работы были применены и закреплены профессиональные компетенции в области проектирования, разработки, тестирования и сопровождения программного обеспечения.

Цели практики достигнуты в полном объёме. Полученный опыт подтверждает готовность к профессиональной деятельности в сфере разработки информационных систем.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 19.701-90 ЕСПД. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения [Текст]. – Введ. 1992-01-01. – М.: Изд-во стандартов, 1991. – 28 с.
2. ГОСТ 34.602-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы [Текст]. – Введ. 1990-07-01. – М.: Изд-во стандартов, 1989.
3. ГОСТ 34.201-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем [Текст]. – Введ. 1990-07-01. – М.: Изд-во стандартов, 1989.
4. ГОСТ Р 57580.2-2017 Защита информации. Разработка безопасных приложений. Часть 2. Требования к безопасности веб-приложений [Текст]. – Введ. 2018-07-01. – М.: Стандартинформ, 2017.
5. Vue.js Official Documentation [Электронный ресурс]. – Режим доступа: https://vuejs.org/guide/introduction.html, свободный.
6. Microsoft Learn – Entity Framework Core Documentation [Электронный ресурс]. – Режим доступа: https://learn.microsoft.com/ef/core/, свободный.
7. Pomelo.EntityFrameworkCore.MySql – GitHub Repository [Электронный ресурс]. – Режим доступа: <https://github.com/PomeloFoundation/Pomelo.EntityFrameworkCore.MySql>, свободный.
8. MySQL 8.0 Reference Manual [Электронный ресурс]. – Режим доступа: <https://dev.mysql.com/doc/refman/8.0/en/>, свободный.
9. Pinia – Official Documentation [Электронный ресурс]. – Режим доступа: <https://pinia.vuejs.org/>, свободный.
10. Axios – HTTP Client Documentation [Электронный ресурс]. – Режим доступа: <https://axios-http.com/docs/intro>, свободный.
11. Metanit.com – Руководство по Vue.js [Электронный ресурс]. – Режим доступа: <https://metanit.com/web/vue3/>, свободный.
12. Metanit.com – ASP.NET Core и C# [Электронный ресурс]. – Режим доступа: <https://metanit.com/sharp/aspnet6/>, свободный.
13. Metanit.com – Entity Framework Core [Электронный ресурс]. – Режим доступа: <https://metanit.com/sharp/entityframeworkcore/>, свободный.
14. Metanit.com – Язык программирования C# [Электронный ресурс]. – Режим доступа: <https://metanit.com/sharp/tutorial/>, свободный.
15. Habr.com – Разработка SPA на Vue.js [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/hubs/vuejs/>, свободный.
16. Habr.com – ASP.NET Core и Web API [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/hubs/aspnet/>, свободный.
17. Habr.com – Entity Framework Core на практике [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/search/?q=Entity%20Framework%20Core>, свободный.
18. Habr.com – REST API и JWT-аутентификация [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/search/?q=JWT%20ASP.NET>, свободный.
19. Visual Studio 2022 Documentation [Электронный ресурс]. – Режим доступа: <https://learn.microsoft.com/visualstudio/>, свободный.
20. Visual Studio Code Documentation [Электронный ресурс]. – Режим доступа: <https://code.visualstudio.com/docs>, свободный.