Министерство науки и высшего образования Российской Федерации

ФГБОУ ВО «Марийский государственный университет»

Институт цифровых технологий

Физико-математический факультет

Кафедра прикладной математики и информатики

Направление подготовки Информационные системы и технологии

Профиль: Программирование и контроль качества программного обеспечения

Курсовая работа

по дисциплине: «Разработка Web приложений»

на тему: «Разработка Web-приложения «Сервис грузоперевозок»»

Научный руководитель:

к. т. н, доцент кафедры ПМиИ

Р. В. Ерусланов

Выполнил:

Студент ФМФ СИ-35

Камаев А. Л.

Йошкар-Ола

2023г.

# Содержание

[Содержание 2](#_Toc135916551)

[Перечень сокращений и обозначений 3](#_Toc135916552)

[Введение 4](#_Toc135916553)

[1. Теоретическая часть 5](#_Toc135916554)

[1.1 Анализ предметной области 5](#_Toc135916555)

[1.2 Формирование требований к системе 5](#_Toc135916556)

[1.3 Выбор инструментария реализации 6](#_Toc135916557)

[2. Проектная часть 8](#_Toc135916558)

[2.1 Постановка задачи 8](#_Toc135916559)

[2.2 Проектирование базы данных 8](#_Toc135916560)

[2.3. Проектирование страницы регистрации и логинации пользователя 10](#_Toc135916561)

[2.4 Структура сайта: Your Data Base 12](#_Toc135916562)

[2.5 Проектирование главного раздела 12](#_Toc135916563)

[2.6 Проектирование раздела контакты 13](#_Toc135916564)

[2.7 Проектирование раздела оформления заказа 13](#_Toc135916565)

[Заключение 14](#_Toc135916566)

[Список литературы 15](#_Toc135916567)

# Перечень сокращений и обозначений

HTML – Hypertext Markup Language

**CSS** – Cascading Style Sheets

**ID** – Identifier

# Введение

В современном мире все интернет-ресурсы содержат базы данных, без баз данных они просто не могут функционировать, так как сайт должен хранить информацию пользователей. Если веб ресурс является маркетплейсом, то структура базы данных будет более обширной и будет содержать в себе несколько больших таблиц, например: таблицы товаров, заказов покупателей чеков и много другое.

Целью данной курсовой работы является разработка web-приложения «Your Data Base», который будет соответствовать современным высоким требованиям пользователей сети интернет.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Произвести анализ предметной области;
2. Формирование требований к модулю;
3. Выбор прикладных инструментов;
4. Проектирование базы данных;
5. Тестирование и отладка на стороне сервера.

Этапы выполнения курсовой работы:

* Анализ информационных потребностей пользователей объектов предметной области;
* Разработка web-приложения, на основе предметных областей задач;
* Разработка базы данных, основанной на физической модели системы;
* Подготовка отчетов по результатам проделанной работы.

Все этапы выполнения курсовой работы реализуются в практической части.

# 1. Теоретическая часть

## 1.1 Анализ предметной области

База данных – это организованная коллекция структурированных данных, которая обычно хранится в электронном формате и управляется базами данные (СУБД). Вместе с СУБД и связанными приложениями, они образуют систему баз данных.

Базы данных делятся на несколько основных типов:

1. Реляционные – в данных БД информация хранится в виде таблиц (MySQL, Oracle DB, PostgreSQL);
2. Резидентные – сведения хранятся в оперативной памяти (Redis, Apache Ignite);
3. Столбчатые – в данных БД информация хранится в столбцах. При использовании такой БД используется пакетная вставка для быстрого чтения по столбцам (Clickhouse, Vertica);
4. Графовые – хранение осуществляется в виде графов, то есть моделями с узлами и связями (OrientDB, Neo4j).

Сейчас почти в любом бизнесе необходима база данных для ведения статистики, учета и т.д. Чтобы успешно вести свой бизнес жизненно необходима правильная структура баз данных предприятия, чтобы получать стабильны доход.

На сегодняшний день существует очень мало сайтов, которые могут предоставить услугу разработки базы данных, ее настройку и сопровождение.

## 1.2 Формирование требований к системе

При создании сайта были выделены следующие основные этапы и требования к разработке сайта:

* Сформировать задачи и цели;
* Разработать дизайн web-приложения;
* Протестировать web-приложение.

С помощью административного блока совершается управление типами доступа сайта. В административном блоке происходит исполнение следующих функций управление типами доступа:

* Добавление типа доступа;
* Удаление типа доступа;
* Изменение типа доступа у пользователей.

## 1.3 Выбор инструментария реализации

Для выполнения данного курсового проекта были использованы программы:

* Microsoft Visual Studio 2022;
* Visual Studio Code;

Среды разработки:

* ASP.NET Core 5.0
* HTML
* CSS
* JavaScript

ASP.NET Core – это открытая платформа с открытым исходным кодом, предназначенная для разработки высокопроизводительных облачных приложений, которые могут работать на различных операционных системах. Она обеспечивает возможность создания современных подключенных к интернету приложений, которые могут быть развернуты в облаке и локально.

HTML – это стандартизированный язык разметки гипертекстовых документов, который используется для отображения web-страниц в браузере. Web-браузеры получают HTML-документы от сервера по протоколам HTTP/HTTPS или открывают их с локального диска и интерпретируют код, чтобы отобразить его в виде пользовательского интерфейса на экране монитора. В данном проекте используется блочная верстка.

Блочная верстка – это методика верстки web-страниц, при которой для структурирования страницы используется свойство CSS "float" и элементы "div" или другие элементы в качестве строительных блоков. С использованием свойства "float" и блочных элементов, можно создавать страницы с несколькими столбцами и задавать их расположение.

CSS – это формальный язык, используемый для стилизации и описания внешнего вида web-страниц, написанных с использованием языка разметки. CSS позволяет задавать цвета, шрифты, стили, расположение блоков и другие аспекты визуального представления web-страниц. Основная цель CSS - отделение логической структуры страницы от описания ее внешнего вида.

JavaScript – это мультипарадигменный язык программирования, который обычно используется в качестве встроенного языка для доступа к объектам приложений. Он широко применяется в браузерах как язык сценариев для создания интерактивных web-страниц.

ASP.NET Core позволяет выполнять следующие задачи:

* Создавать web-приложения, службы и серверные компоненты для мобильных приложений;
* Использовать предпочитаемые инструменты разработки на платформах Windows, macOS и Linux;
* Развертывать приложения как в облачных средах, так и на локальных серверах;
* Запускать приложения с использованием .NET Core.

# 2. Проектная часть

## 2.1 Постановка задачи

Для разработки сервиса для оформления заказа были поставлены следующие задачи:

* Разработка и создание базы данных;
* Разработка интерфейса заказа для пользователя.

## 2.2 Проектирование базы данных

ASP.NET Core включает в себя возможность использования локальной базы данных, при этом все необходимые таблицы автоматически создаются при первоначальной инициализации базы данных.

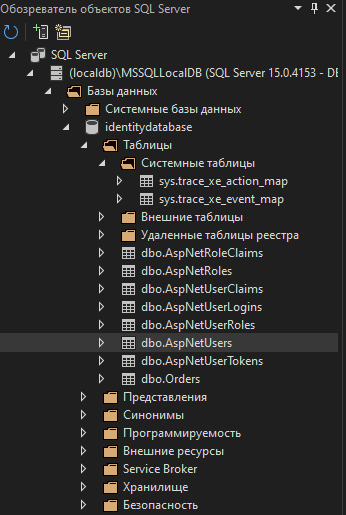


Рисунок 1 – Таблицы БД

В рамках данного проекта используются таблицы «dbo.AspNetUsers», «dbo.Orders».

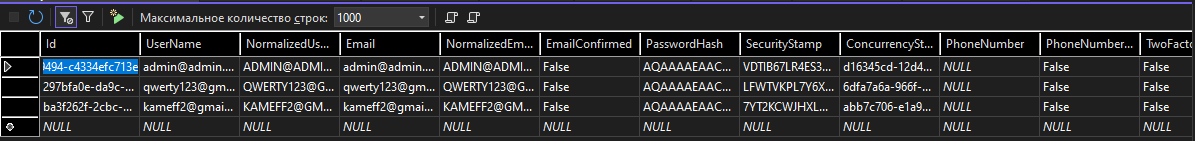
В базе данных используется таблица dbo.AspNetUsers для хранения информации о зарегистрированных пользователях. Однако, в рамках данной курсовой работы, используются только два поля этой таблицы - "Email" и "Password". 

Рисунок 2 – Таблица dbo.AspNetUsers

Таблица dbo.Orders используется для хранения заказов пользователей. В данной таблице заполняются поля “Id”, “Email” и “Comment”

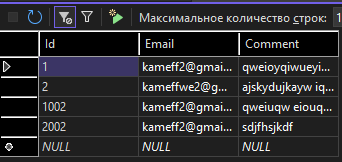


Рисунок 3 – Таблица dbo.Orders

## 2.3. Проектирование страницы регистрации и логинации пользователя

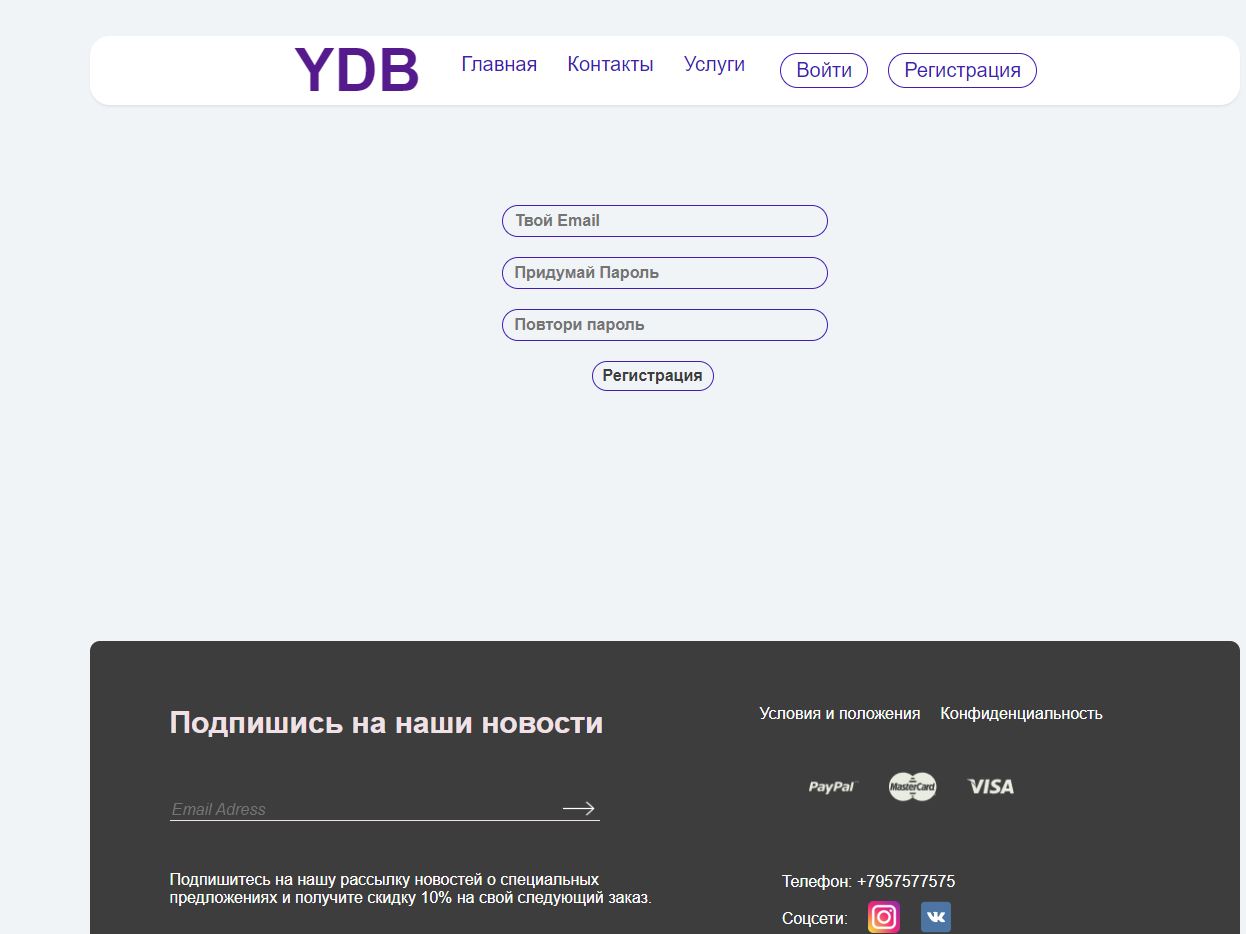
Для реализации страницы регистрации, в первую очередь был создан дизайн. Пользователь переходит на страницу при клике на кнопку «Регистрация». Далее вводит нужные параметры, а именно: свой Email, пароль и подтверждает сой пароль повторным вводом ниже в текстовое поле.

Рисунок 4 – Страница регистрации пользователя

Далее производится проверка на наличие пользователи и если такого пользователя не существует он записывается в таблицу «dbo.AspNetUsers».

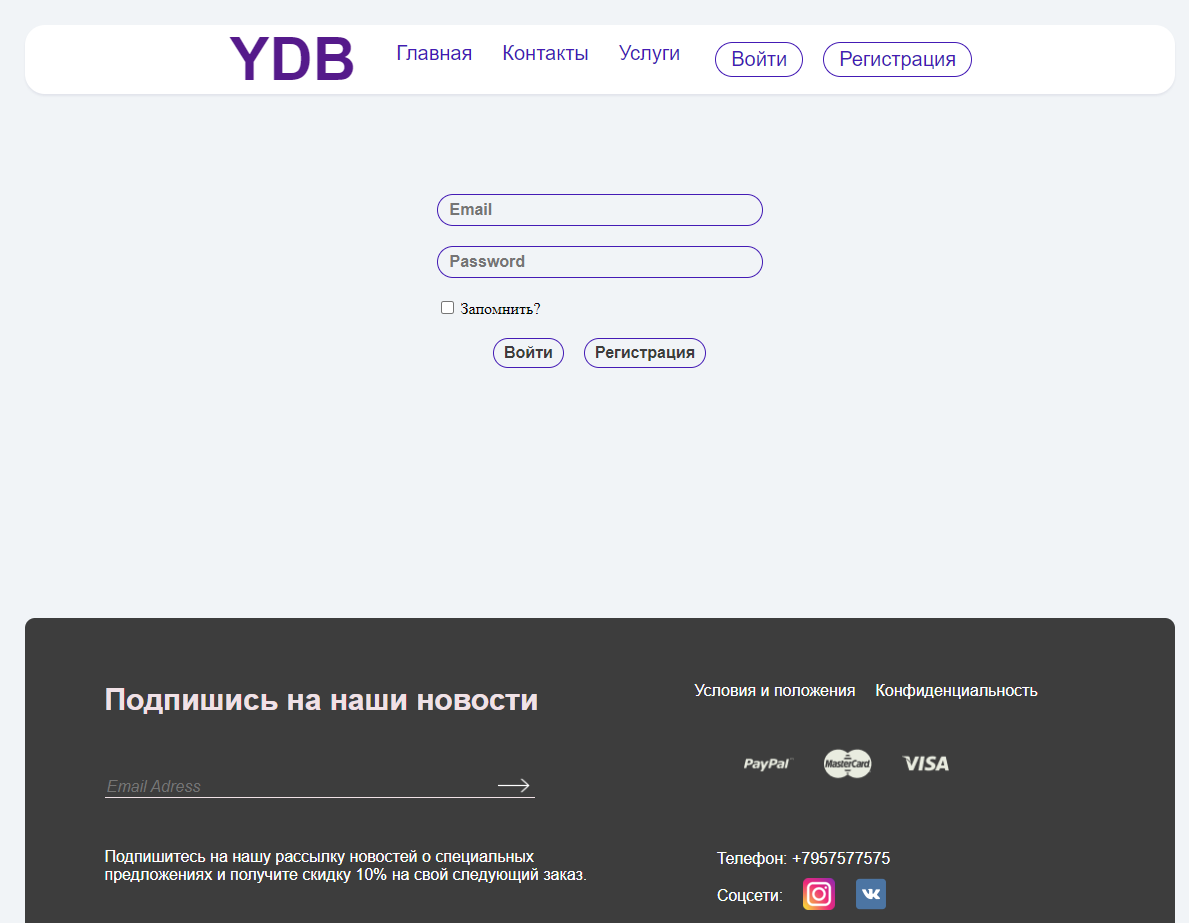
Если пользователь уже зарегистрирован на сайте, то он может нажать на кнопку «Войти» и так же пройти логинацию с запылением следующих полей: «Email» и «Пароль». Если пользователь не зарегистрирован, то он может перейти в раздел регистрации нажав на кнопку «Регистрация».

Рисунок 5 – Страница логинации пользователя

Добавление, удаление ролей, изменение типов доступа у пользователей происходит в контроллере RolesController. В данном контроллере созданы менеджеры пользователей и ролей с типами RoleManager и UserManager и добавлены функции создания, удаления ролей и изменения типов доступа у зарегистрированных пользователей.

## 2.4 Структура сайта: Your Data Base

Сайты различных сервисов обычно имеют разнообразные структуры, но существуют также общие структурные элементы, которые встречаются на большинстве из них.

Структура данного сайта выглядит следующим образом:

* Главная страница – содержит представление о сайте в целом, есть несколько картинок, которые дают представление для чего разработан данный сайт и просит авторизироваться пользователя, чтобы совершить заказ;
* После авторизации пользователю дается возможность выполнить заказ;
* Страница заказа содержит в себе поле ввода почты пользователя и комментарий к заказу, который он выбрал на главной странице;
* Страницы регистрации и авторизации – регистрация и авторизация пользователя;

## 2.5 Проектирование главного раздела

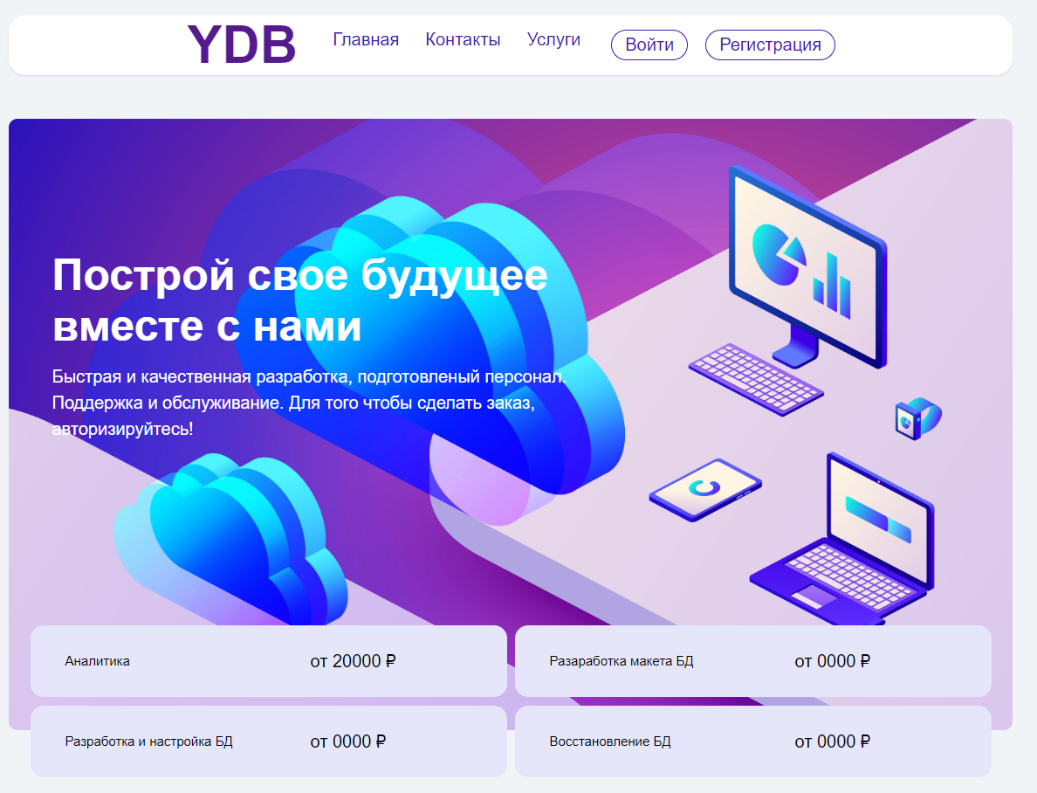
В данном разделе выводится краткое описание сайта.

Рисунок 6 – Главная страница

## 2.6 Проектирование раздела контакты

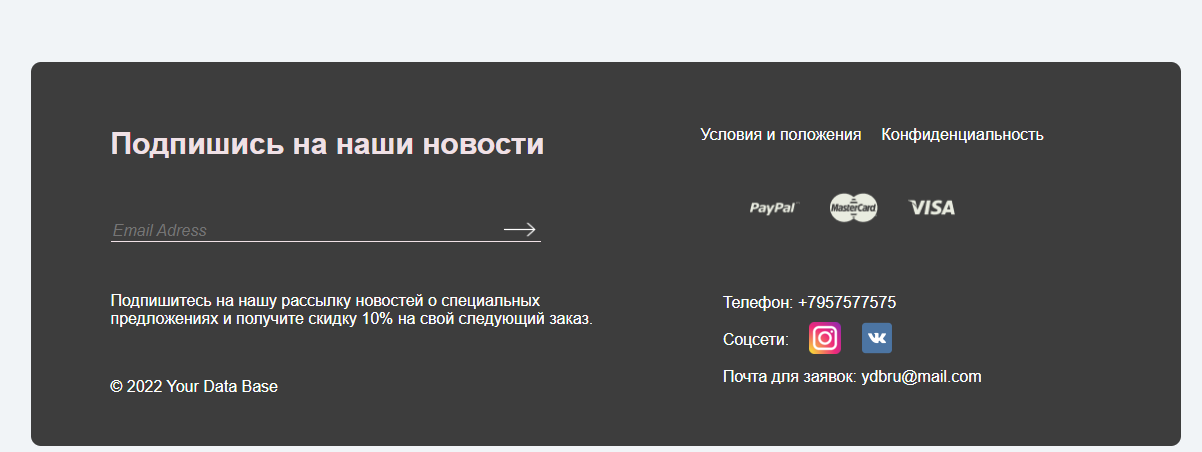
В данном разделе выводятся вся информации о связи пользователя с руководством компании, а также возможность подписки на рассылку о акциях на сайте.

Рисунок 10 – Раздел «Контакты»

## 2.7 Проектирование раздела оформления заказа

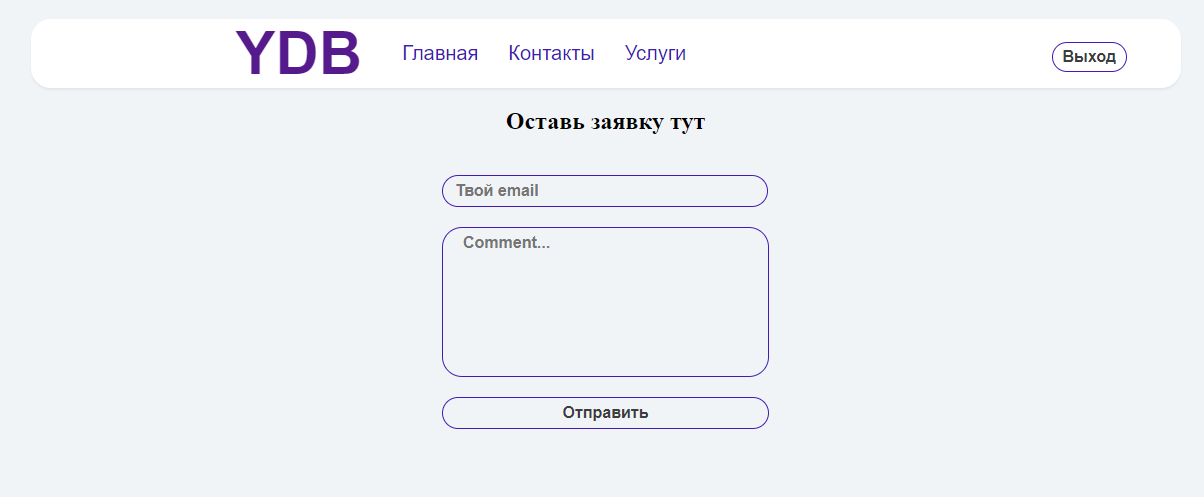
 В данной секции отображается форма для оформления заказа. После нажатия кнопки «Отправить»., информация о заказе отправляется в базу данных «dbo.Orders».

Рисунок 11 – Раздел оформления заказа

# Заключение

Цель курсового проекта успешно достигнута, все задачи выполнены, и web-приложение реализовано.

В процессе разработки данного проекта были изучены различные программные продукты и среды разработки, включая Asp.Net, а также пакеты AspNet.Identity и EntityFrameworkCore.

В целом, проект получился продуктивным и имеет потенциал для дальнейшей разработки.

# Список литературы

1. Что такое базы данных [сайт] – URL: Что такое база дынных? ([https://www.oracle.com/cis/database/what-is-database](https://www.oracle.com/cis/database/what-is-database/)) (дата обращения 15.05.2023).

2. Общие сведения об ASP.NET Core [сайт] – URL: Общие сведения об ASP.NET Core (<https://learn.microsoft.com/ru-ru/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-7.0>) (дата обращения 12.05.2023).

3. Основные типы баз данных [сайт] – URL: 7 основных типов баз данных (<https://jino.ru/journal/articles/7-baz-dannyh/>) (дата обращения 15.05.2023)

3. Добавление Identity в проект с нуля [сайт] – URL: Добавление Identity в проект с нуля (<https://metanit.com/sharp/aspnet5/16.2.php>) (дата обращения 27.04.2023).

4. Регистрация и создание пользователей в Identity [сайт] – URL: Регистрация и создание пользователей в Identity (<https://metanit.com/sharp/aspnet5/16.3.php>) (дата обращения 24.04.2023).

5. Авторизация пользователей в Identity [сайт] – URL: Авторизация пользователей в Identity (<https://metanit.com/sharp/aspnet5/16.4.php>) (дата обращения 27.04.2023).

6. Подключение и создание базы данных в Entity Framework Core [сайт] – URL: Подключение и создание базы данных в Entity Framework Core (<https://metanit.com/sharp/aspnet5/12.1.php>) (дата обращения 01.05.2023).

7. Основные операции с моделями. Редактирование и удаление [сайт] – URL: Основные операции с моделями. Редактирование и удаление (<https://metanit.com/sharp/aspnet5/12.3.php>) (дата обращения 01.05.2023).

8. HTML [сайт] – URL: HTML (<https://ru.wikipedia.org/wiki/HTML>) (дата обращения 15.05.2023).

9. CSS [сайт] – URL: CSS (<https://ru.wikipedia.org/wiki/CSS>) (дата обращения 15.05.2023).

10. JavaScript [сайт] – URL: JavaScript (<https://ru.wikipedia.org/wiki/JavaScript>) (дата обращения 15.05.2023).