

**Содержание**

[Введение 3](#_Toc135255341)

[1.Системный анализ предметной области 4](#_Toc135255342)

[2.Описание программы 6](#_Toc135255343)

[3.Руководство пользователя 8](#_Toc135255344)

[3.1 Область применения 8](#_Toc135255345)

[3.2 Уровень подготовки пользователя 8](#_Toc135255346)

[3.3 Назначение программы 8](#_Toc135255347)

[3.4 Программные и аппаратные требования к системе 8](#_Toc135255348)

[3.5 Пуск программы 8](#_Toc135255349)

[3.6 Состав и структура программы 9](#_Toc135255350)

[3.7 Описание операций 13](#_Toc135255351)

[4.Руководство программиста 14](#_Toc135255352)

[4.1 Программные и аппаратные требования к системе 14](#_Toc135255353)

[4.2 Характеристики программы 14](#_Toc135255354)

[4.3 Код программы 15](#_Toc135255355)

[5.Отладка и тестирование программы 20](#_Toc135255356)

[Заключение 21](#_Toc135255357)

[Список литературы 22](#_Toc135255358)

ВВЕДЕНИЕ

Выбранная тема считается актуальной на сегодняшний день, так как сегодня миллионы людей ежедневно, не выходя из дому, покупают различные товары в электронных магазинах. В мире, а в частности в России, огромными темпами растет количество пользователей Internet и, как следствие, количество «электронных» покупателей и потенциальных «электронных» покупателей.   
 Электронные магазины существенно уменьшают издержки производителя, сэкономив на содержании обычного магазина, расширяют рынки сбыта, так же как и расширяет возможность покупателя – покупать любой товар в любое время в любой стране, в любом городе, в любое время суток, в любое время года. Это дает электронным магазинам неоспариваемое преимущество перед обычными магазинами. Этот момент является существенным при переходе производителей с «обычной» торговли на «электронную».   
 Высокое качество продукции, умение донести информацию о продукте до потребителя и эффективная система сбыта, делает предприятие успешным на рынке. Во многих компаниях встречаются проблемы сбыта, которые мешают эффективно работать отделу продаж, и не исчезают даже с подбором хороших продавцов. Решить их можно только путем автоматизации процесса продаж. В узком и технологическом смысле, под электронным бизнесом ранее понималось использование информационных технологий (в первую очередь связанных с Интернетом) для организации взаимодействия предприятия с внешней средой, включая поставщиков, потребителей, партнеров и т.д. При таком подходе электронный бизнес выступает, прежде всего, как достаточно сложная прикладная информационная система. Более широкий, или концептуальный, подход рассматривает электронный бизнес как способ предпринимательства, способствующий достижению стратегического успеха в новую информационную эпоху. При таком понимании электронный бизнес отнюдь не сводится к информационным технологиям или активности в Интернете. Электронная коммерция затрагивает все аспекты бизнеса, включая стратегию, процессы, организацию и технологию, и выводит его далеко за сложившиеся границы.

Цель курсовой работы – реализация интернет-магазина на одном из объектно-ориентированных языков программирования высокого уровня. В качестве такого языка в работе был выбран язык программирования C#.

Для достижения цели курсовой работы были поставлены следующие задачи:

* выявить ключевые особенности проектирования программ, позволяющие вводить информацию, хранить её в файле, а также осуществлять модификацию данных;
* выполнить реализацию интернет-магазина на языке C#, а также ее тестирование и отладку;
* разработать блок-схему программы.

1.Системный анализ предметной области

# 

Интернет-магазин – сайт, торгующий товарами посредством сети Интернет. Позволяет пользователям онлайн, в своём браузере или через мобильное приложение, сформировать заказ на покупку, выбрать способ оплаты и доставки заказа, оплатить заказ. При этом продажа товаров осуществляется дистанционным способом, и она накладывает ограничения на продаваемые товары.

Предметная область включает:

Описание книги:

1. Имя книги
2. Автор книги
3. Цена
4. Категория
5. Год издания
6. Вид переплёта
7. Картинку книги
8. Аннотацию

Описание авторов:

1. Имя автора и фамилия
2. Картинка автора
3. Описание автора

Описание избранного:

1. Айди книги
2. Количество книг

Описание корзины:

1. Айди книги
2. Количество книг

Место:

1. Наименование места, где живет пользователь

Описание пользователя:

1. Имя
2. Фамилия
3. Отчество
4. Емайл
5. Дата рождения
6. Пол
7. Пароль

Информация о книгах вводит разработчик в базу данных таблицу книги

Информацию о авторах вводит так же разработчик в таблицу авторы

Избранное формируется в зависимости от действий пользователя

Корзина формируется в зависимости от действий пользователя

Избранное формируется в зависимости от действий пользователя

2.Описание программы

Разработанный Интернет-магазин предназначен для персональных компьютеров, ноутбуков, а также для электронно-вычислительных машин с 32-разрядной и 64-разрядной версией Windows. В качестве технических средств выступают системный блок, монитор, мышь, клавиатура.

Способ вызова и загрузки разработанного программного средства осуществляется с помощью мыши или клавиатуры.

В качестве входных данных являются данные, которые вводятся либо с помощью клавиатуры, либо выбираются с помощью мыши, в зависимости от выполняемой функции.

В качестве выходных данных является информация о книгах в Web-странице.

Запуск Интернет-магазина

Закрытие Интернет-магазина

Добавление книг в корзину

Оформление заказа

Отображение списка имеющихся книг

Добавление книг в Web-сайт

Удаление книг с Web-сайта

Рисунок 1 - Функционирование программы «Интернет-магазин»

Начало

Вывод выбранных книг

Продолжить покупки

Добавление книг в корзину

Оформить заказ

Конец

Рисунок 2 - Добавление книг в корзину

Начало

Конец

Ввод контактных данных

Обработка заказа

Оформление заказа

Рисунок 3 - Оформление заказа

Начало

Конец

Удаление выбранной книги

Выбор книги для удаления

Рисунок 4 - Удаление информации о книги

Начало

Конец

Ввод названия, автора, аннотации, жанр, цены

Добавление книги

Рисунок 5 - Добавление книги

3.Руководство пользователя

3.1 Область применения

Интернет-магазин – форма торговли, которая доступна круглосуточно. В любое удобное время потенциальный клиент может зайти на сайт компании, ознакомиться с каталогами, внимательно изучить особые предложения, бонусные программы и другие плюсы данного ресурса. И тут же сделать заказ, выбрав удобный способ оплаты и доставки.

3.2 Уровень подготовки пользователя

Пользователь приложения «Интернет-магазин» должен обладать пользовательскими навыками в работе с ПК, должен ознакомиться с руководством пользователя, а также должен уметь делать заказ.

3.3 Назначение программы

Программное средство «Интернет-магазин» предназначено для купли-продажи различных товаров и услуг, а также их быстрого поиска.

3.4 Программные и аппаратные требования к системе

Данное программное средство требует 32-разрядные и 64-разрядные версии Windows, кабели, системный блок, монитор, клавиатуру, мышь, банковскую карту.

3.5 Пуск программы

При запуске программы «Интернет-магазин» появляется Web-страница, изображенное на рисунке 6.



Рисунок 6 - Web-страница программы «Интренет-магазин»

3.6 Состав и структура программы

Интерфейс программы состоит из:

1. Главная страница – на ней в шапке представлен основной функционал сайта, так же блок новинок и авторов.



Рисунок 7 - Главная страница

Страница поиска - на ней присутствует возможность выбора книг по жанрам, по автору, по году выхода, так же в строке «хедера» есть поле для поиска по имени книги.

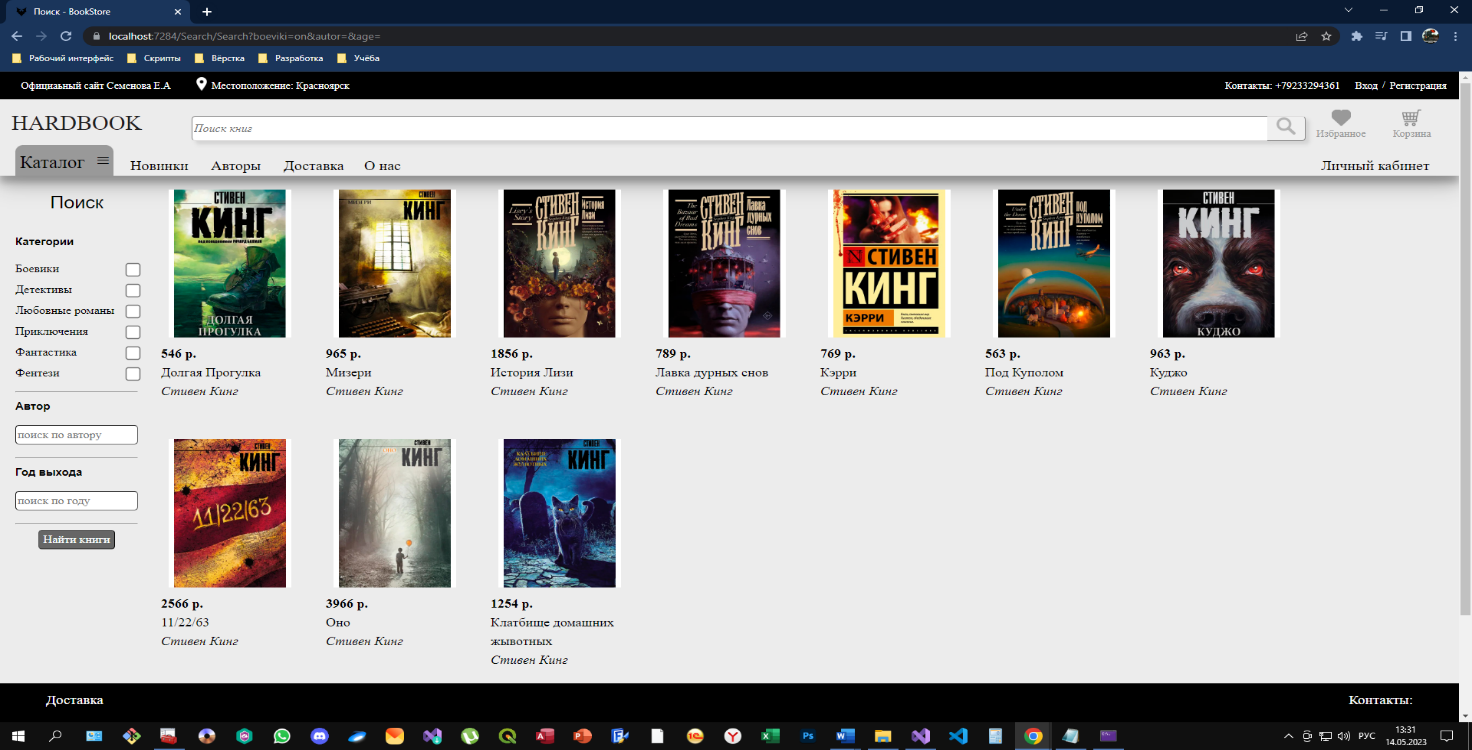


Рисунок 8 - Страница поиска

Страница личный кабинет позволяет изменять и удалять данные пользователя.

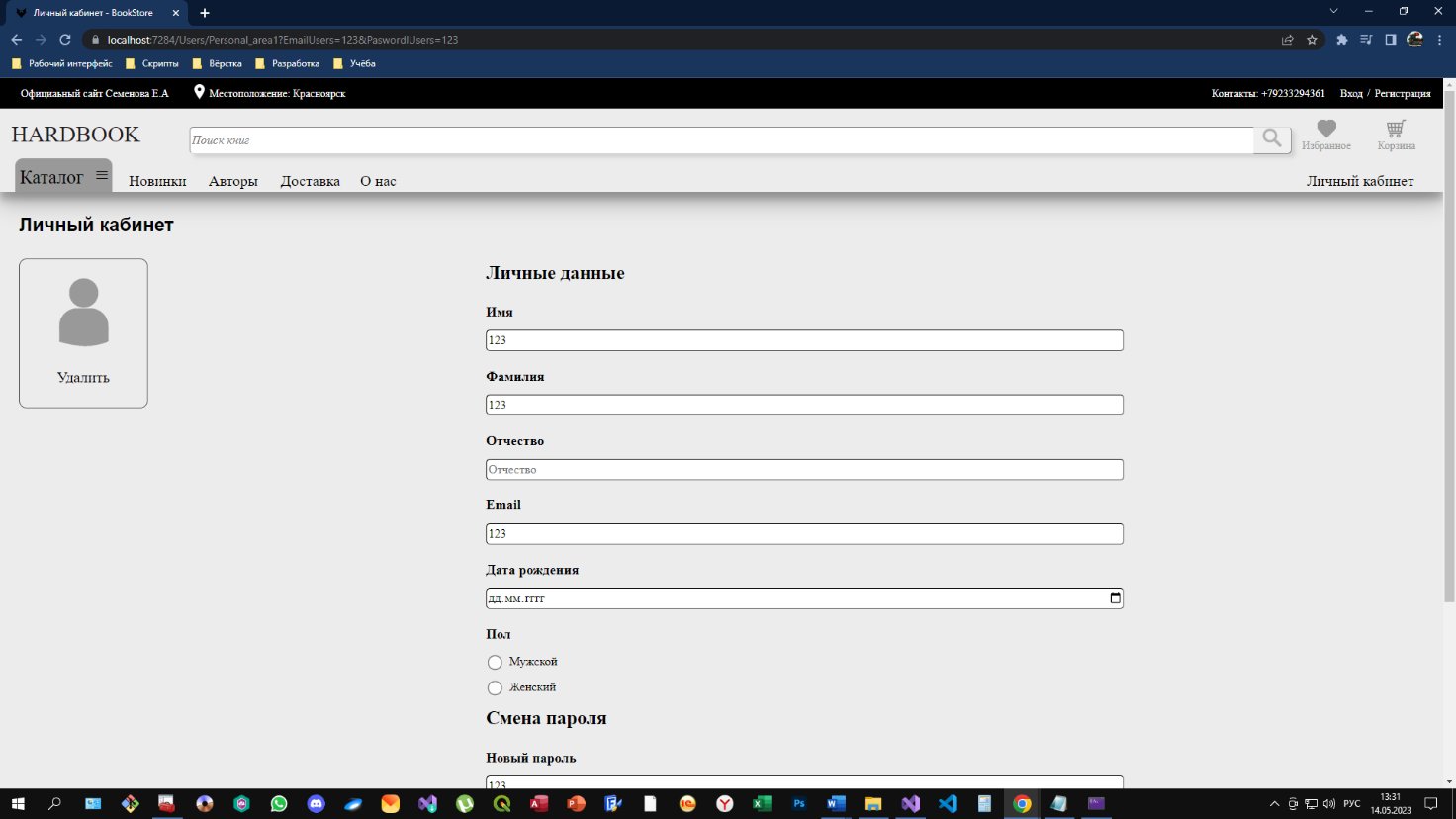


Рисунок 9 - Личный кабинет

Страница избранное хранит информацию о тех книгах, которые выбрал пользователь.

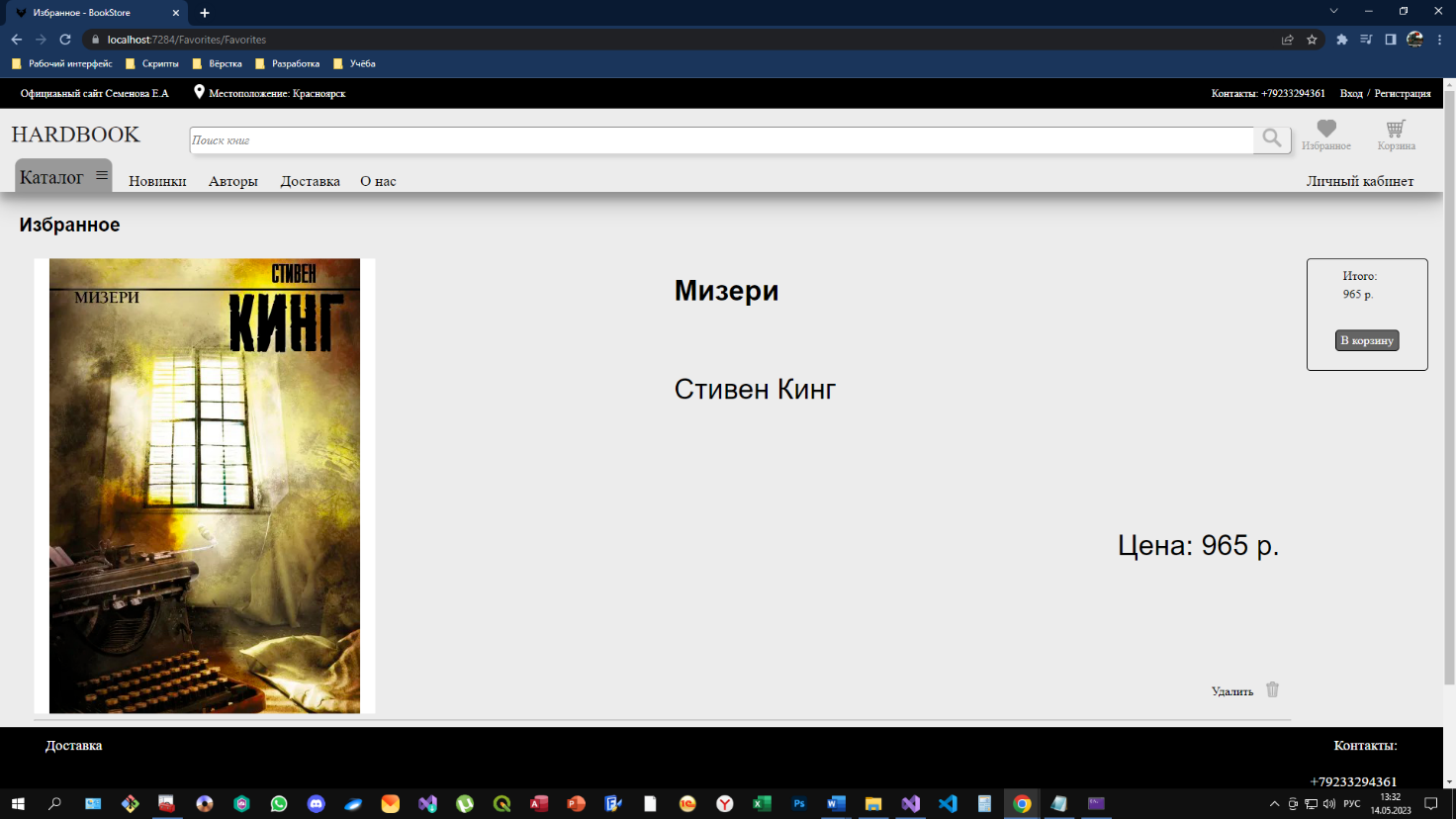


Рисунок 10 - Избранное

Страница корзина хранит информацию о тех книгах, которые выбрал пользователь.

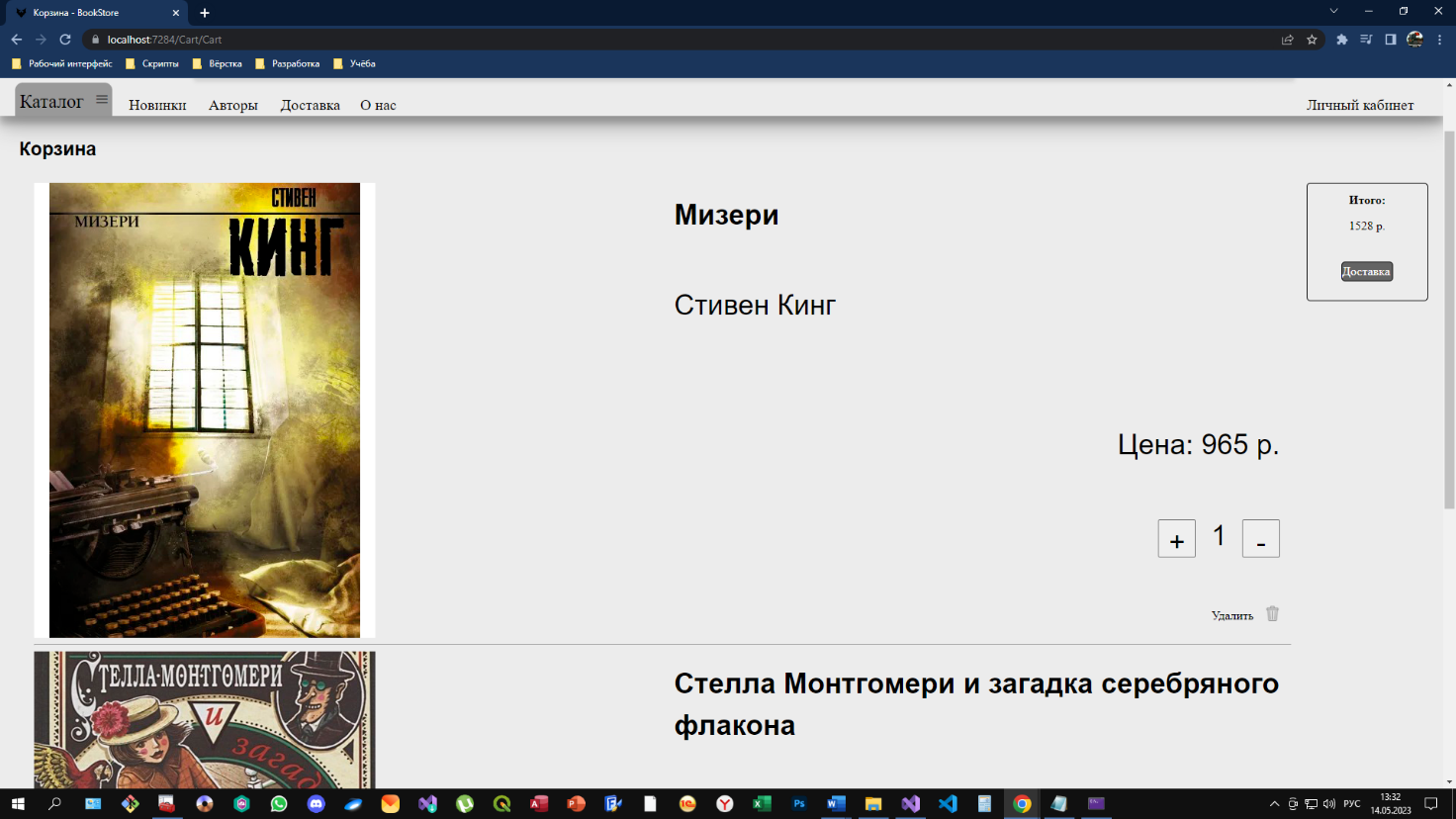


Рисунок 11 - Корзина

Страница оформление имеет поля для ввода информации для доставки.

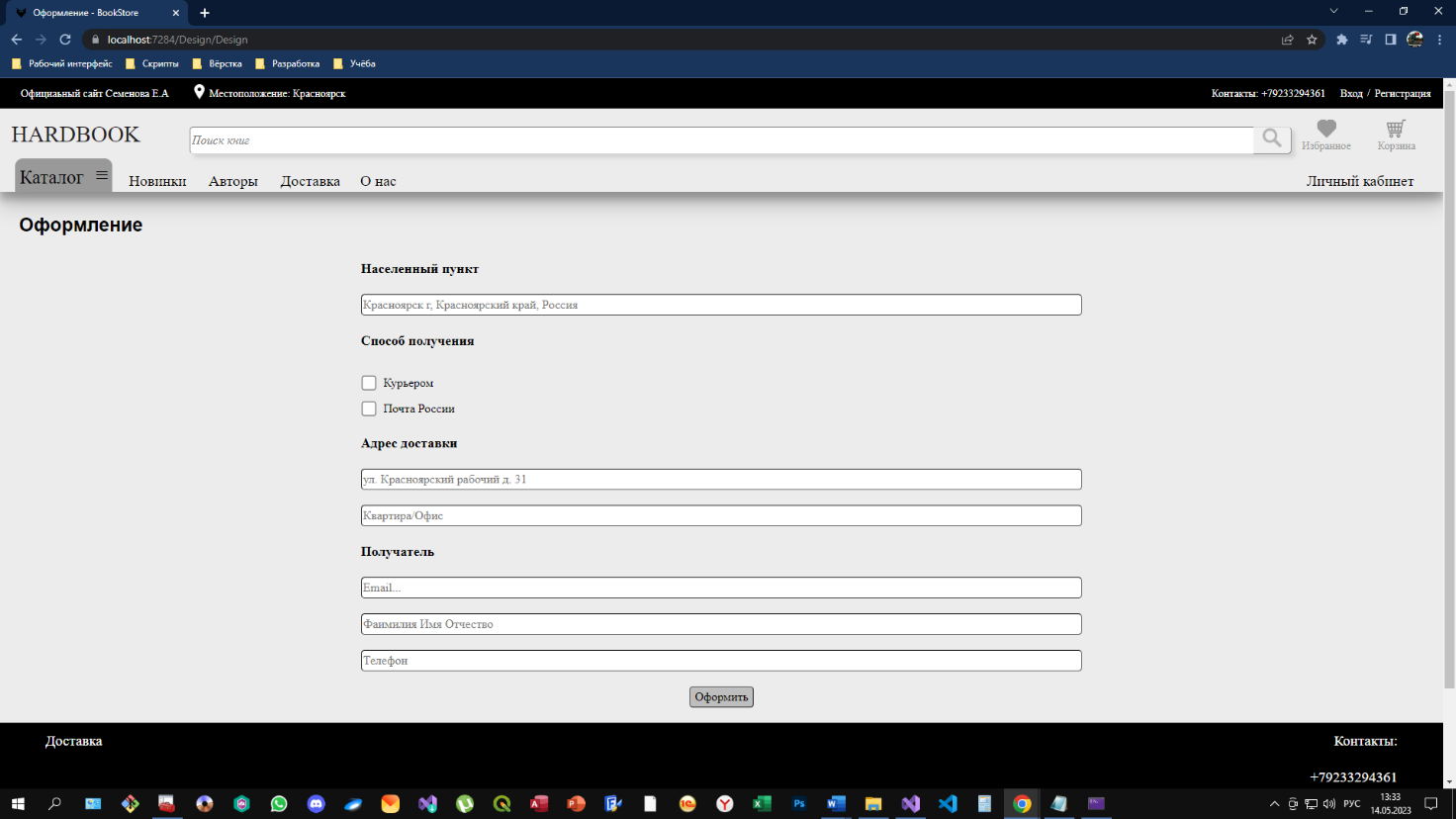


Рисунок 12 - Корзина

После заполнения оформления если в корзине не пусто и пользователь заполнил все поля формируется чек.

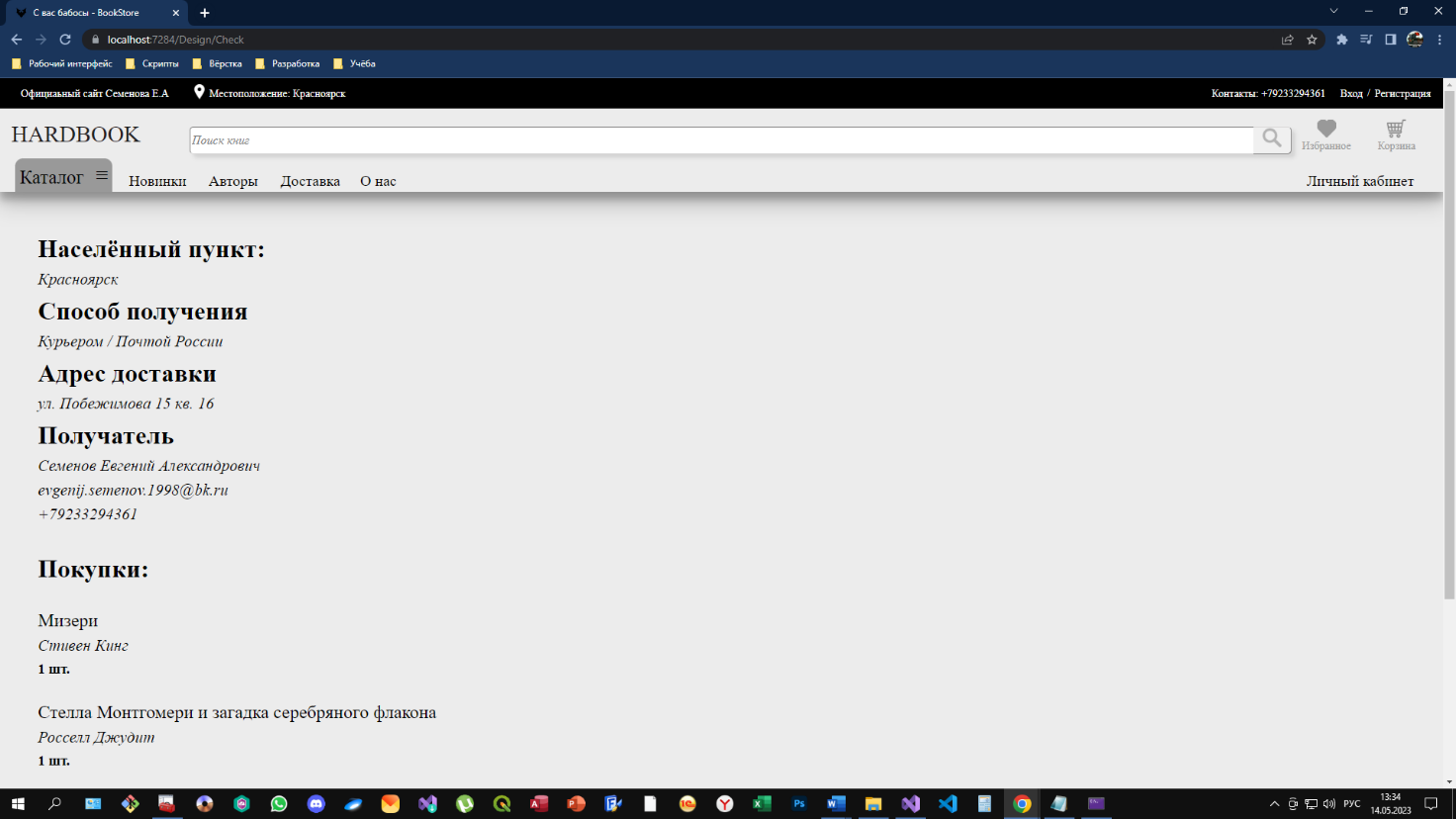


Рисунок 13 - Чек

Страница о нас представляет информацию о том как этот сайт был сделан.

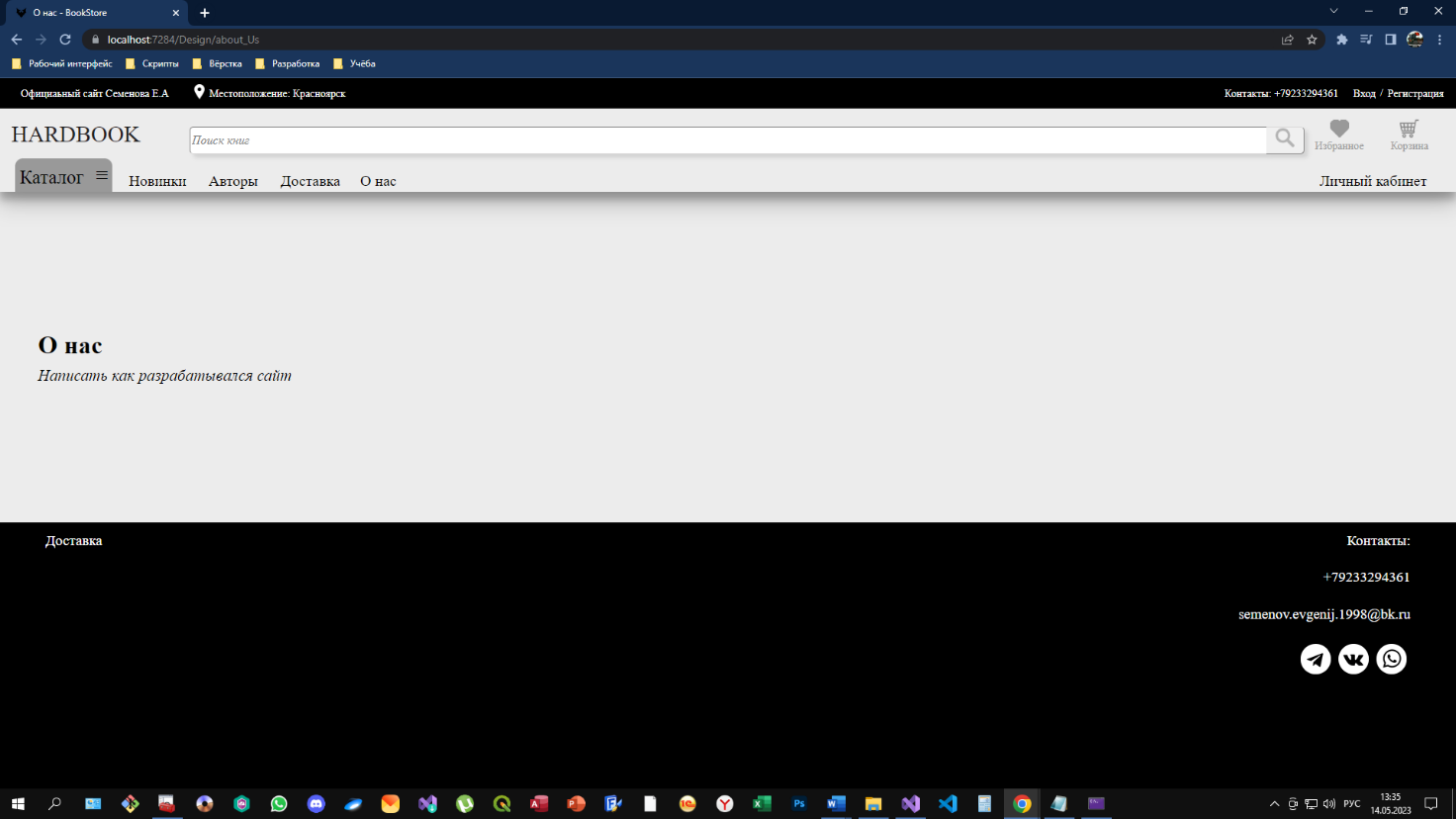


Рисунок 14 - О нас

3.7 Описание операций

1. На главной странице в шапке сайта можно ввести местоположение пользователя.
2. Во вкладке вход для того что бы перейти в личный кабинет нужно заполнить всплывающее меню вход.
3. Во вкладке регистрация заполняются поля для новых пользователей.
4. Во вкладку каталог всплывают все категории книг с ссылками на страницу поиска в заполненном виде с соответствующей категорией.
5. Новинки ссылаются на главную страницу.
6. Авторы ссылаются на главную страницу.
7. Доставка ссылается на страницу оформления.
8. О нас ссылается на страницу о нас.
9. Личный кабинет вызывает всплывающее меню вход.
10. Нажатие на любую книгу ссылается на страницу этой книги.
11. Нажатие на автора ссылается на страницу автора.
12. На странице поиска можно выбрать категории, ввести автора. Ввести год издания и получить в ответ представление книг в зависимости от запроса.
13. В избранном есть возможность удалить книгу из избранного или выгрузить всё избранное в корзину.
14. В корзине есть возможность добавить количество книг, удалить книгу из корзины, или же перейти в доставку.
15. Личный кабинет имеет возможность заполнить поля данных о пользователе или же их удалить.
16. Доставка имеет возможность заполнить поля для доставки книг.
17. Так же в подвале сайта есть ссылки на мессенджеры разработчика сайта.

4.Руководство программиста

4.1 Программные и аппаратные требования к системе

Требования к системе: наличие сервера IIS И базы данных SQLEXPRESS.

4.2 Характеристики программы

Характеристики программы:

1. Архитектурным решением было выбрано MVC – модель-представление-контроллер, что позволяет более гибко редактировать приложение, это связанно с тем, что проект разбит на 3 почти независимых блока это соответственно модели, представления и контролеры. Модели служат для описания предметной области сущностей, хранящихся в БД. Контроллеры — это логика программы, они отвечают за вывод представлений и обращаются к моделям для доступа к информации в базе данных. Все выше описанное позволяет удалять или же изменять отдельные блоки приложения без вреда для других блоков, а так же добавляет удобства в тестировании.
2. При разработке применялись следующие технологии:
3. На этапе макетирования интерфейса применялся:
4. Adobe Photoshop версии 3.2.24

2) На этапе верстки применялись:

* + - 1. HTML5 – это язык разметки, предназначенный для формирования
      2. CSS3 – таблицы стилей для придания визуальных эффектов
      3. JS – язык программирования клиентской части, применялся для анимации на сайте
      4. VS Code как редактор
      5. Git для сохранения состояния проекта

3)На этапе программирования применялись:

* 1. Язык программирования C# на платформе ASP.NET
  2. Библиотеки JQERY для использования фоновых запросов к серверу
  3. Файлы конфигурации json
  4. IDE Visual Studio
  5. База данных SQLEXPRESS
  6. Сервер IIS

Архитектура проекта:

Проект состоит из папки Properties, wwwroot, Controllers, Models, Views и фалов Program и appsettings.

В папке Properties храниться файл launchSettings который отвечает за базовую конфигурацию приложения.

В папке wwwroot хранятся статические файлы – картинки, стили, скрипты

В папке Controllers хранятся файлы контроллеров(файлы с кодом C#)

В папке Models хранятся модели(файлы с кодом C#)

В папке Views хранятся файлы представлений(файлы с кодом Razor)

Файл Program это входная точка программы там определяются основные маршруты и подключаются необходимые библиотеки для приложения

Файл appsettings предназначен для хранения конфигурации подключений к внешним данным таким как сервер IIS и базы данных.

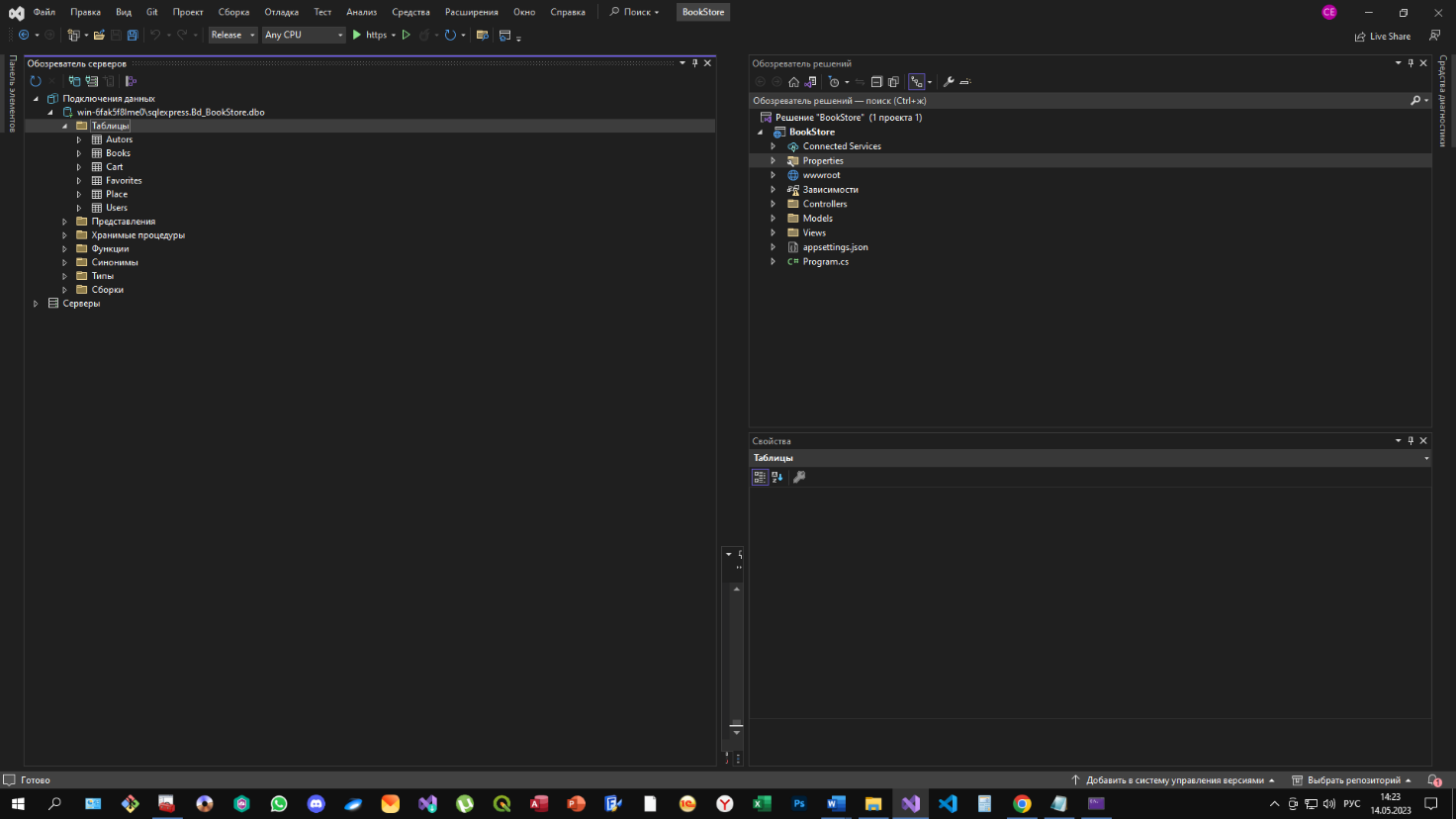


Рисунок 15 - Конфигурация приложения и БД

4.3 Код программы

Program

using BookStore.Models;

using Microsoft.AspNetCore.Builder;

using Microsoft.EntityFrameworkCore;

using System;

using Microsoft.AspNetCore.Authorization;

using Microsoft.AspNetCore.Authentication.Cookies;

using System.Security.Claims;

using Microsoft.AspNetCore.Authentication;

//НЕ трогать

var builder = WebApplication.CreateBuilder(args);

//НЕ трогать

/\*ДОБАВИЛ СЕССИЮ\*/

// добавляем сервисы сессии

builder.Services.AddDistributedMemoryCache();

builder.Services.AddSession(options =>

{

options.IdleTimeout = TimeSpan.FromSeconds(10);

options.Cookie.HttpOnly = true;

options.Cookie.IsEssential = true;

});

/\*ДОБАВИЛ СЕССИЮ\*/

/\*ДОБАВИЛ БД\*/

string? connection = builder.Configuration.GetConnectionString("DefaultConnection");

builder.Services.AddDbContext<ApplicationContextAutors>(options => options.UseSqlServer(connection));//добавил автора

builder.Services.AddDbContext<ApplicationContextBooks>(options => options.UseSqlServer(connection));//добавил книги

builder.Services.AddDbContext<ApplicationContextUsers>(options => options.UseSqlServer(connection));//добавил пользователя

builder.Services.AddDbContext<ApplicationContextFavorites>(options => options.UseSqlServer(connection));//добавил контекст избранного

builder.Services.AddDbContext<ApplicationContextCart>(options => options.UseSqlServer(connection));//добавил контекст корзины

builder.Services.AddDbContext<ApplicationContextPlace>(options => options.UseSqlServer(connection));//добавил контекст места

/\*ДОБАВИЛ БД\*/

builder.Services.AddControllersWithViews();

var app = builder.Build();

//НЕ трогать

//статические файлы

app.UseDefaultFiles();

app.UseStaticFiles();

if (!app.Environment.IsDevelopment())

{

app.UseExceptionHandler("/Home/Error");

// The default HSTS value is 30 days. You may want to change this for production scenarios, see https://aka.ms/aspnetcore-hsts.

app.UseHsts();

}

app.UseHttpsRedirection();

app.UseRouting();

//регистрация и вход

app.UseAuthentication();

app.UseAuthorization();

app.UseSession(); // добавляем middleware для работы с сессиями

//регистрация и вход

//стандартный маршрут

app.MapControllerRoute(

name: "default",

pattern: "{controller=Home}/{action=Main\_page}/{id?}");

//НЕ трогать

/\*Не трогать\*/

app.Run();

/\*Не трогать\*/

Appsettings

{

//это подключает к проекту БД

"ConnectionStrings": {

"DefaultConnection": "Data Source=WIN-6FAK5F8LME0\\SQLEXPRESS;Initial Catalog=Bd\_BookStore;Integrated Security=True;MultipleActiveResultSets=True;TrustServerCertificate=True;"

}, //это подключает к проекту БД

"Logging": {

"LogLevel": {

"Default": "Information",

"Microsoft.AspNetCore": "Warning"

}

},

"AllowedHosts": "\*"

}

AutorsController

using BookStore.Models;

using Microsoft.AspNetCore.Mvc;

using System.Diagnostics;

using System.Data.Entity.Infrastructure.DependencyResolution;

namespace BookStore.Controllers

{

public class AutorsController : Controller

{

private ApplicationContextAutors db1;

private ApplicationContextBooks db2;

private ApplicationContextPlace db4;

public AutorsController(ApplicationContextAutors context1, ApplicationContextBooks context2, ApplicationContextPlace context4)

{

db1 = context1;

db2 = context2;

db4 = context4;

}

/\*ПОДКЛЮЧАЮ БД\*/

/\*выгрузка тупо всей базы данных в нужное представление\*/

private object BDResult()

{

var ListAutors = db1.Autors.ToList();

var ListBooks = db2.Books.ToList();

var ListPlace = db4.Place.ToList();

var General = new GeneralModel { Autors = ListAutors, Books = ListBooks, Place = ListPlace };

return General;

}

public IActionResult Authors\_page(string id)

{

Place place = new Place();

foreach (var item in db4.Place)

{

place = item;

}

ViewData["locality"] = place.place;

ViewData["Message"] = id;

return View(BDResult());

}

//частичные представления

public IActionResult Header()

{

return PartialView(BDResult());

}

public IActionResult Footer()

{

return PartialView(BDResult());

}

//частичные представления

}

}

ApplicationContextAutors

using Microsoft.EntityFrameworkCore;

namespace BookStore.Models

{

public class ApplicationContextAutors : DbContext

{

public DbSet<Autors> Autors { get; set; } = null!;

public ApplicationContextAutors(DbContextOptions<ApplicationContextAutors> options)

: base(options)

{

Database.EnsureCreated();

}

}

}

Autors

namespace BookStore

{

public class Autors

{

public int ID { get; set; }

public string? NameAutors { get; set; }

public string? ImgUrlAutors { get; set;}

public string? DescriptionAutorsAutors { get;set;}

}

}

\_Layout

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>@ViewData["Title"] - BookStore</title>

<link rel="stylesheet" href="~/css/general/null.css">

<link rel="stylesheet" href="~/css/general/header.css">

<link rel="stylesheet" href="~/css/general/footer.css">

<link rel="stylesheet" href="~/css/main\_page.css">

<link rel="stylesheet" href="~/css/main\_Author's\_page.css">

<link rel="stylesheet" href="~/css/main\_Book\_page.css">

<link rel="stylesheet" href="~/css/main\_Cart.css">

<link rel="stylesheet" href="~/css/main\_Design.css">

<link rel="stylesheet" href="~/css/main\_Favorites.css">

<link rel="stylesheet" href="~/css/main\_personal\_area.css">

<link rel="stylesheet" href="~/css/main\_search.css">

<link rel="stylesheet" href="~/css/abaut.css">

</head>

<body>

<script src="~/js/jqery.js"></script>

@RenderBody()

</body>

</html>

Authors\_page

@{

ViewBag.Title = "Страница автора";

}

@model GeneralModel

@await Html.PartialAsync("Header")

<main class="main\_search1">

@foreach (var autor in Model.Autors)

{

if (autor.ID == Convert.ToInt32(ViewData["Message"]))

{

<!--Верхняя часть автора-->

<div class="main\_search\_info\_autor">

<div class="info\_autor\_img\_name">

<div class="info\_autor\_img\_box">

<img src="~/@autor.ImgUrlAutors" alt="" class="info\_autor\_img">

</div>

<div class="info\_autor\_name">

<p class="info\_autor\_name\_p">@autor.NameAutors</p>

</div>

</div>

<div class="info\_autor\_opisanie">

<p class="autor\_opisanie\_p">

@autor.DescriptionAutorsAutors

</p>

</div>

</div>

<!-- часть с книгами-->

<div class="main\_centr">

<div class="main\_centr-book-text">

Книги автора

</div>

<!-- часть с книгами-->

<div class="main\_centr-book">

@foreach (var book in Model.Books)

{

if (autor.NameAutors == book.AutorBook)

{

<div class="main\_new-box\_book">

<!--Сылка на популярную книгу-->

<div class="main\_new-box\_book-box1">

<a href="/Books/Book\_page/@book.ID" class="main\_new-box\_book-a">

<div class="box\_book\_img">

<img class="book\_img" src="~/@book.ImgBigBook" alt="">

</div>

<div class="box\_book\_price">

@book.PraiseBook

</div>

<div class="box\_book\_name">

@book.NameBook

</div>

<div class="box\_book\_autor">

@book.AutorBook

</div>

</a>

</div>

</div>

}

}

</div>

</div>

}

}

</main>

@await Html.PartialAsync("Footer")

5.Отладка и тестирование программы

Программное средство «Интернет-магазин» отлично функционирует на любом компьютере. Работает корректно, по щелчку любой кнопки, выводит правильную информацию.

Заключение

В ходе выполнения курсового проекта была изучена и проанализирована учебно-методическая документация, учебная литература и интернет по выбранной теме.

Были найдены всевозможные способы достижения поставленной цели, анализ которых позволил выбрать наиболее оптимальный способ, для достижения которого был разработан необходимый набор универсальных функций.

В рамках курсовой работы была разработана программа на языке программирования C# с использованием графического интерфейса, предоставляющая возможность отображения, добавления, изменения информации об адресах людей. Программа имеет понятный пользовательский интерфейс, выводит все необходимые пояснения и подсказки, является законченной и удобной для использования.

В ходе работы были выполнены все поставленные задачи:

* + выявлены ключевые особенности проектирования программ, позволяющие вводить информацию, хранить её в файле, а также осуществлять модификацию данных;
  + выполнена реализация и отладка веб-магазина на языке C#, а также ее тестирование. По результатам тестов получен вывод о том, что все необходимые функции работают корректно;
  + разработана блок-схема программы.

Список литературы

1. Общие сведения о Windows Forms / Microsoft. URL: https://docs.microsoft.com/ru-ru/dotnet/framework/winforms/windows-forms-overview (дата обращения 25.05.2019)
2. Введение в язык C# и .NET Framework / Microsoft. URL: https://docs.microsoft.com/ru-ru/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework (дата обращения 25.05.2019)
3. Руководство по языку программирования VB.NET URL: https://metanit.com/visualbasic/tutorial/ (дата обращения 25.05.2019)
4. Работа с базами данных на языке C#. Технология ADO.NET: Учебное пособие / сост. О. Н. Евсеева, А. Б. Шамшев. – Ульяновск: УлГТУ, 2009. – 170 с.
5. Компьютерные технологии в экономике. Методические указания по выполнению курсовой работы. / Сост.: Тихоненко Д.В. – Красноярск: СибГАУ, 2015 – 18 с.