**Аннотация**

Программный продукт "BiblioLand" — это десктоп-приложение для просмотра электронных книг, разработанное на платформе .NET Framework с использованием Windows Forms и языка C#. Приложение предоставляет удобный интерфейс для поиска, фильтрации и чтения книг в формате PDF.

В пояснительной записке описаны:

1. Назначение и функциональность программы
2. Используемые технологии и структура проекта
3. Руководство пользователя
4. Методика тестирования

**Содержание**

[Введение 4](#_Toc199361133)

[1 Назначение и область применения 5](#_Toc199361134)

[2 Постановка задачи 6](#_Toc199361135)

[3 Описание программы 7](#_Toc199361136)

[3.1 Общие сведения 7](#_Toc199361137)

[3.2 Функциональное назначение 7](#_Toc199361138)

[3.3 Описание логической структуры 7](#_Toc199361139)

[3.4 Используемые технические средства 8](#_Toc199361140)

[3.5 Вызов и загрузка 8](#_Toc199361141)

[3.6 Входные и выходные данные 9](#_Toc199361142)

[4 Программа и методика испытаний 10](#_Toc199361143)

[4.1 Объекты испытаний 10](#_Toc199361144)

[4.2 Цель испытаний 10](#_Toc199361145)

[4.3 Требования к программе 10](#_Toc199361146)

[4.4 Методы испытаний 10](#_Toc199361147)

[4.5 Тестовый пример 10](#_Toc199361148)

[5 Руководство оператора 12](#_Toc199361149)

[5.1 Выполнение программы 12](#_Toc199361150)

[5.2 Сообщение оператору 13](#_Toc199361151)

[Заключение 14](#_Toc199361152)

[Приложение А 15](#_Toc199361153)

[Приложение Б 15](#_Toc199361154)

[Источники, использованные при разработке 16](#_Toc199361155)

# Введение

Разработка выполнена в рамках курсового проекта по профессиональному модулю ПМ.01. Программа "KitchenBoss" автоматизирует ключевые процессы управления рестораном: учет заказов, управление персоналом, ведение меню и контроль столиков. В качестве СУБД используется Microsoft SQL Server 18.

Проект "BiblioLand" разработан как десктоп-приложение для удобного чтения управления электронными книгами.

Основные цели:

1. Предоставить пользователю простой интерфейс для просмотра книг.
2. Реализовать базовые функции (поиск, фильтрация, закладки).
3. Обеспечить поддержку популярных форматов (PDF, EPUB).

Исходный код проекта доступен на GitHub:

🔗 https://github.com/L2MBO/BiblioLand.git

# 1 Назначение и область применения

Назначение:

* Просмотр электронных книг в удобном интерфейсе.
* Поиск и сортировка книг по названию, автору, жанру.
* Сохранение закладок и последнего места чтения.

Область применения:

* Домашнее использование.
* Библиотеки и образовательные учреждения.
* Любители чтения, предпочитающие локальное хранение книг.

Аналоги:

* Calibre (кроссплатформенная библиотека).
* FBReader (читалка для ПК и мобильных устройств).
* Adobe Digital Editions (просмотр EPUB и PDF).

# 2 Постановка задачи

Разработать desktop-приложение со следующим функционалом:

* + 1. Просмотр книг PDF.
    2. Управление библиотекой (добавление книги в категорию, удаление, оценка).
    3. Поиск и фильтрация (по названию, автору).
    4. Закладки и история чтения.
    5. Настройки интерфейса (темы, шрифты, масштаб).

# 3 Описание программы

## 3.1 Общие сведения

* Технологии: язык программирования C#, платформа пользовательского интерфейса Windows Forms, .NET Framework 4.7.2, Microsoft SQL Server 18.
* Архитектура: клиент-серверное приложение с монолитной архитектурой.
* Сущности БД: 11 таблиц (Book, Users, Rating и др.)

## 3.2 Функциональное назначение

* Основные модули:

1. Авторизация (AuthorizationForm):
   * + Вход по email и паролю.
     + Хеширование паролей с солью (BCrypt) (см. Приложение А)

Главная форма (MainForm):

* + Навигация между модулями.
  + Отображение книг по категориям.
  + Отслеживание пользовательских книг

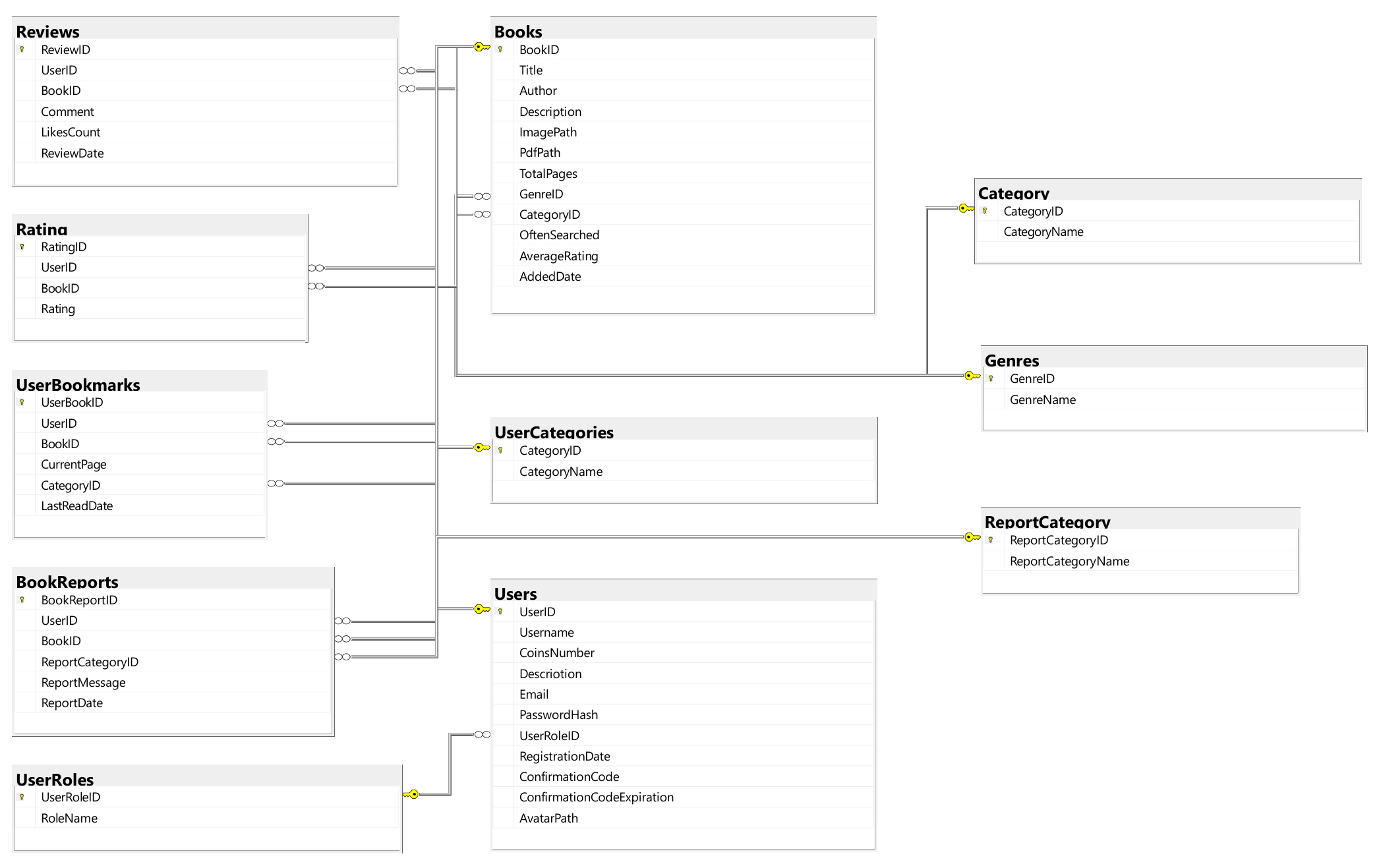
1. Бизнес-логика:
   * Агент и триггеры базы данных для автоматического расчета оценки книги.
   * Валидация данных при сохранении.

## 3.3 Описание логической структуры

Используемые методы см. Приложение Б:

* HashHelper() – класс для хеширования паролей.
* DataAcces – папка с классами для доступа к данным.
* ValidationHelper – класс включающий в себя всю логику валидации данных.
* Триггеры базы данных для автоматического расчета средней оценки.
* Паттерн “Facade” для более удобной работы с приложением.
* В качестве оптимизации панель навигации сделана при помощи UserControl’а.

Схема базы данных:



Структура программы:

1. Слой данных:

* Entity Framework для работы с Microsoft SQL Server.

1. Бизнес-логика:

* Валидация данных.
* Расчет оценки книги.

1. Интерфейс:

* MainForm – главная форма.
* BookInfoForm – страница книги.
* AuthorizationForm – авторизация.

## 3.4 Используемые технические средства

* Операционная система: Microsoft Windows 10/11 или выше.
* .NET Framework 4.7.2.
* Microsoft SQL Server 2018 или выше.

## 3.5 Вызов и загрузка

Способ запуска:

* Двойной клик по BiblioLand.exe.
* Запуск через добавленный пользователем ярлык в меню «Пуск».

Технические параметры:

* Объем оперативной памяти: минимально 8 ГБ.
* Дисковое пространство: 30 ГБ.

Точки входа:

* Program.Main() – инициализация приложения.
* AuthorizationForm.loginButton\_Click() – обработка входа.

## 3.6 Входные и выходные данные

Входные данные:

|  |  |  |
| --- | --- | --- |
| Тип данных | Формат | Пример |
| Учётные данные | Текстовый, числовой | Логин: ivanov  Пароль: qwerty123 |
| Данные книги | Текстовый, числовой | Название: Государь  Автор: Никколо Макиавелли  Описание: это книга великого мыслителя и политика…  Кол-во страниц: 300  Жанр: политика  Категория: популярное  Часто ищут: да  Общая оценка: 10.00 |

# 4 Программа и методика испытаний

## 4.1 Объекты испытаний

* Модуль авторизации.
* Расчет оценки книги.

## 4.2 Цель испытаний

Проверка:

* + - Корректности работы алгоритмов расчета.
    - Соответствия интерфейса ТЗ.
    - Устойчивости к ошибочному вводу.
    - Корректный расчет оценки несмотря на их кол-ва.

## 4.3 Требования к программе

Проверяемые параметры:

* Время отклика ≤ 1 сек.
* Корректное отображение на разрешениях 1280x720 или выше
* Защита от SQL-инъекций.

## 4.4 Методы испытаний

1. Модульное тестирование:

* NUnit-тесты для классов.
* Проверка граничных значений.

## 4.5 Тестовый пример

Таблица 1 – Тест – кейс 01 «Авторизация»

|  |  |  |  |
| --- | --- | --- | --- |
| Номер тест-кейса | 01 | | |
| Название: | Авторизация | | |
| Действие | | Ожидаемый результат | Результат теста:  Пройден |
| Предусловие: | | | |
| Запустить сервер с БД, открыть приложение | | Приложение запущено без ошибок | Пройден |
| Шаги теста: | | | |
| Ввод корректных данных | | Открытие главной панели | Пройден |
| Ввод некорректных данных | | Сообщение об ошибке | Пройден |
| Постусловие: | | | |
| Закрыть приложение, выключить БД | | Приложение закрыто, БД выкл. | Пройден |

Таблица 2 – Тест – кейс 02 «Просмотр страницы книги»

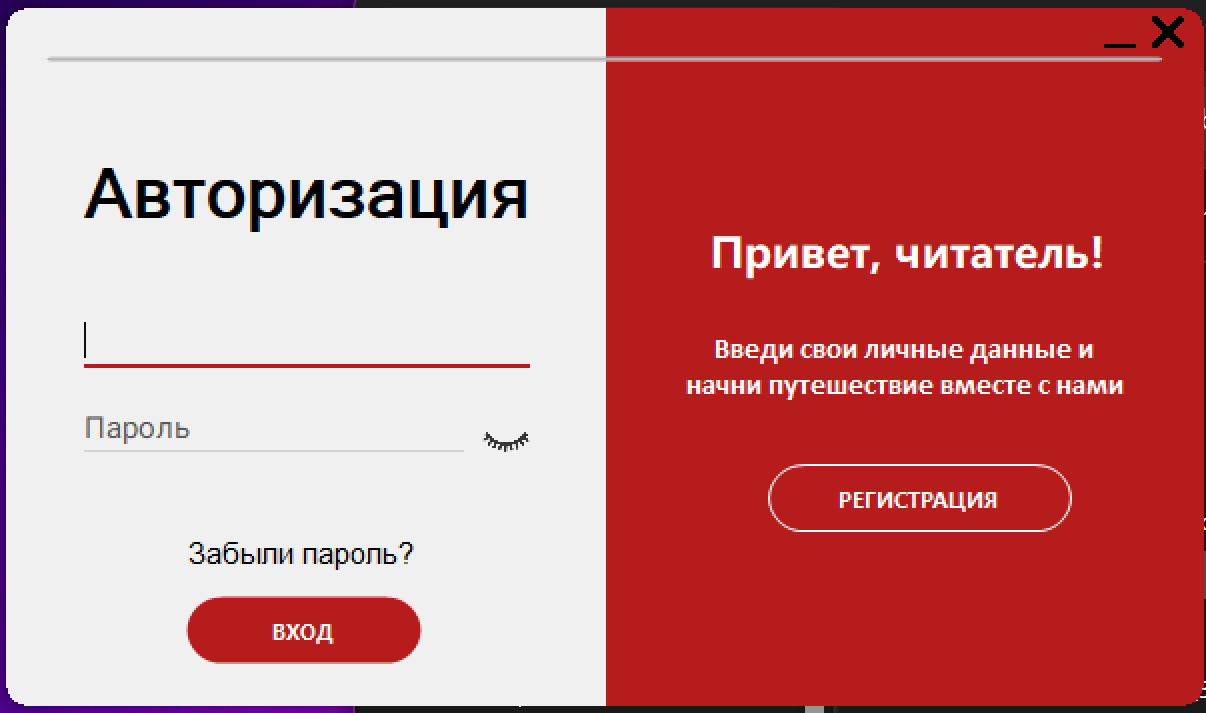
|  |  |  |  |
| --- | --- | --- | --- |
| Номер тест-кейса | 02 | | |
| Название: | Просмотр страницы книги | | |
| Действие | | Ожидаемый результат | Результат теста:  Пройден |
| Предусловие: | | | |
| Запустить сервер с БД, открыть приложение | | Приложение запущено без ошибок | Пройден |
| Шаги теста: | | | |
| Ввод корректных данных | | Открытие главной панели | Пройден |
| Выбрать книгу | | Загружается нужная книга | Пройден |
| Посмотреть на панель рейтинга | | Подгружена средняя оценка книги | Пройден |
| Сравнить оценку книги с оценкой в БД | | Оценка совпадает | Пройден |
| Постусловие: | | | |
| Закрыть приложение, выключить БД | | Приложение закрыто, БД выкл. | Пройден |

Таблица 3 – Тест – кейс 03 «Сохранение книги в закладки»

|  |  |  |  |
| --- | --- | --- | --- |
| Номер тест-кейса | 03 | | |
| Название: | Сохранение книги в закладки | | |
| Действие | | Ожидаемый результат | Результат теста:  Пройден |
| Предусловие: | | | |
| Запустить сервер с БД, открыть приложение | | Приложение запущено без ошибок | Пройден |
| Шаги теста: | | | |
| Выбрать книгу | | Загружается нужная книга | Пройден |
| Нажать на кнопку “В закладки” и выбрать нужную категорию | | Категория успешно выбрана | Пройден |
| Сравнить категорию закладки с категорией в БД | | Категория совпадает | Пройден |
| Постусловие: | | | |
| Закрыть приложение, выключить БД | | Приложение закрыто, БД выкл. | Пройден |

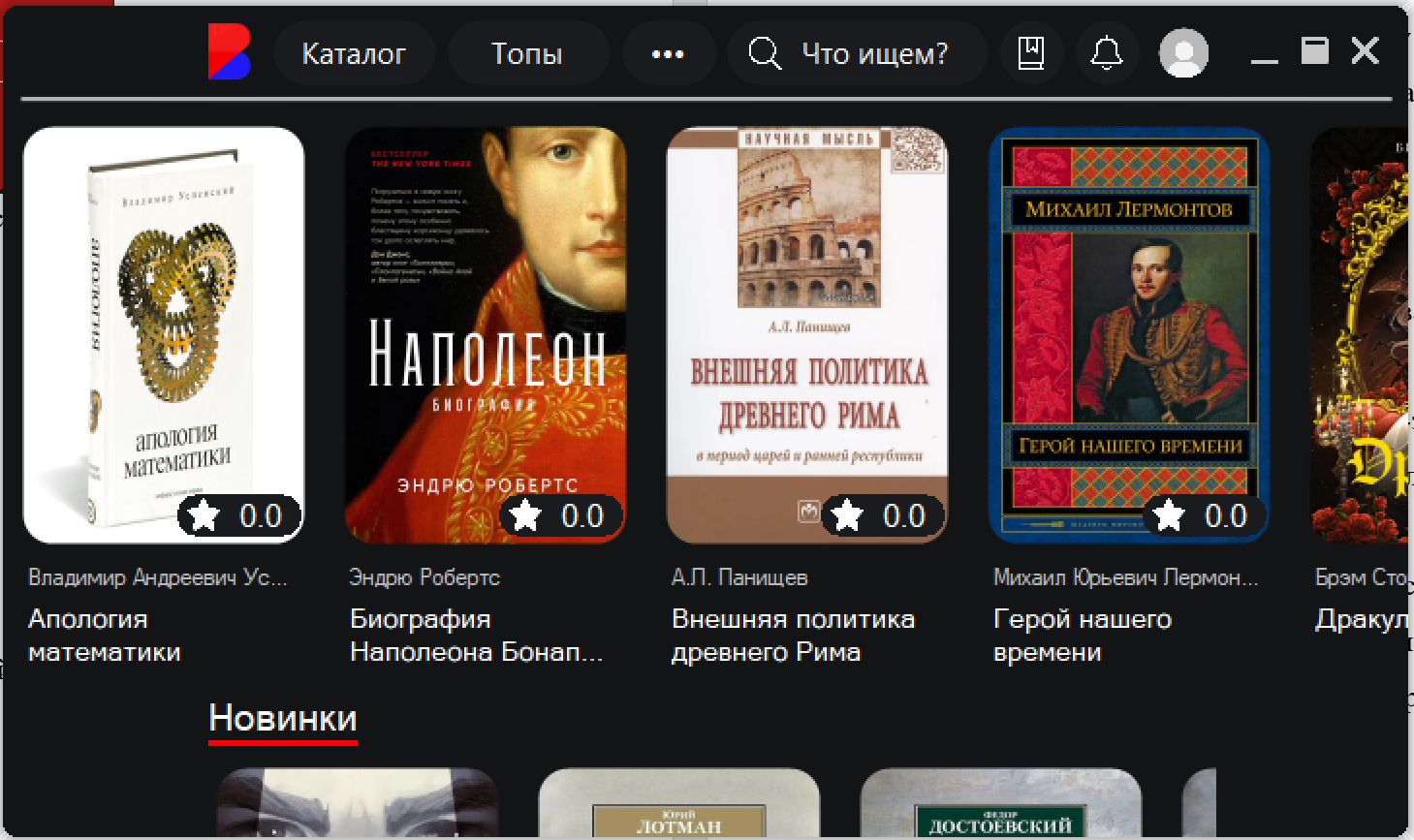
# 5 Руководство оператора

## 5.1 **Выполнение** программы



1. Запуск приложения через exe файл / ярлык (сервер баз данных должен работать).

2. Ввод учётных данных.



1. Главная Панель.

Расположены книги по различным категориям.



1. Страница книги, отображенная после нажатия на книгу.

Отображается информация о книге, а также ее статистика.

Завершение программы – нажать на крестик на панеле навигации.

## 5.2 Сообщение оператору

Ошибки, которые могут возникнуть:

* "Ошибка подключения к БД" → проверить сервер.

# Заключение

В ходе разработки приложения "BiblioLand" для онлайн чтения книг, был реализован комплексный программный продукт, решающий ключевые задачи приятного пользовательского опыта. Система демонстрирует следующие ключевые характеристики:

1. Гибкость архитектуры:

* Модульная структура позволяет легко расширять функционал.
* Разделение на слой данных, бизнес логику и пользовательский интерфейс обеспечивает поддержку кода.
* Использование Entity Framework упрощает поддержку миграцию на другие СУБД.

1. Безопасность и надежность:

* Многоуровневая система авторизации с хешированием паролей.
* Встроенные механизмы валидации входных данных.

1. Производительность:

* Оптимизированные SQL-запросы к базе данных.
* Кэширование часто используемых данных.

1. Удобство использования:

* Единый интерфейс для всех бизнес-процессов.
* Контекстные подсказки и валидация ввода.
* Адаптивный дизайн под разные разрешения экранов.

Перспективы развития системы:

* Внедрение мобильного приложения.
* Разработка модуля аналитики и прогнозирования.
* Поддержка мультиязычного интерфейса.

Рекомендации по внедрению:

* Провести обучение персонала по работе с системой.
* Разработать регламент резервного копирования данных.
* Обеспечить регулярное обновление ПО.

# Приложение А

**Функция хеширования пароля**

/// <summary>

/// Хэширование и верификация пароля.

/// </summary>

public class HashHelper

{

/// <summary>

/// Хэширование пароля.

/// </summary>

/// <param name="password"></param>

/// <returns>хэшированный пароль</returns>

public static string HashPassword(string password)

{

return BCrypt.Net.BCrypt.HashPassword(password);

}

/// <summary>

/// Верификация пароля.

/// </summary>

/// <param name="password"></param>

/// <param name="storedHash"></param>

/// <returns>сравнение пароля</returns>

public static bool VerifyPassword(string password, string storedHash)

{

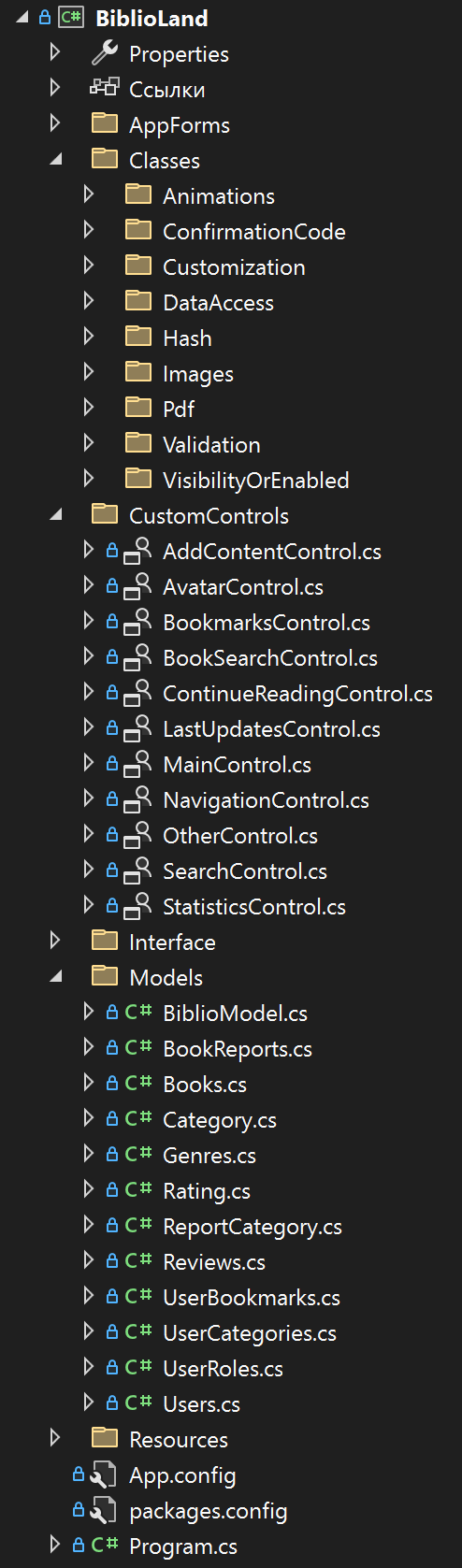
return BCrypt.Net.BCrypt.Verify(password, storedHash);

}

}

# Приложение Б

**Структура проекта**



# Источники, использованные при разработке

1. ГОСТ Р ИСО\_МЭК 25051-2017 Требования к качеству готового к использованию программного продукта (RUSP) и инструкции по тестированию.
2. ЕСПД Единая система программной документации.
3. Изучаем PHP 7. Руководство по созданию интерактивных веб-сайтов, Дэвид Скляр, 2017
4. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5, Никсон Робин, 2019
5. Изучаем JavaScript. Руководство по созданию современных веб-сайтов, Этан Браун, 2017

**Internet – ресурсы**

1. PHP: Hypertext preprocessor [Электронный ресурс] – [http://php.net](http://php.net/)
2. JavaScript Tutorial [Электронный ресурс] – <http://w3schools.com/js>
3. MySQL [Электронный ресурс] - <https://www.mysql.com/>
4. htmlbook.ru | Для тех кто делает сайты [Электронный ресурс] -http://htmlbook.ru/