МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных Технологий

Кафедра Программной инженерии

Специальность 1-40 01 01 Программное обеспечение информационных технологий

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**КУРСОВОГО ПРОЕКТА:**

по дисциплине «Объектно-ориентированные технологии программирования и стандарты проектирования»

Тема «Каталогизатор книг»

Исполнитель

студент 2 курса 4 группы Губарь Анна Александровна \_

Руководитель работы Северинчик Никита Александрович\_

Курсовой проект защищен с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель Cеверинчик Н. А. \_

Минск 2021

**Оглавление**

[**Введение** 3](#_Toc73082379)

[**1. Аналитический обзор прототипов** 4](#_Toc73082380)

[**1.1 «BOOKSHELF»** 4](#_Toc73082381)

[**1.2 «HANDY LIBRARY»** 5](#_Toc73082382)

[**1.3 «Reading Planner»** 5](#_Toc73082383)

[**1.4 «Bookmate»** 6](#_Toc73082384)

[**1.5 «Read More: Reading Habit Tracker»** 7](#_Toc73082385)

[**1.6 «GoodReads»** 8](#_Toc73082386)

[**2 Разработка архитектуры проекта** 10](#_Toc73082387)

[**2.1 Архитектура системы** 10](#_Toc73082388)

[**2.2 Проектирование логической структуры базы данных** 10](#_Toc73082389)

[**2.3 Проектирование структуры окон** 11](#_Toc73082390)

[**2.4 Пример диаграммы последовательностей** 13](#_Toc73082391)

[**2.5 Пример вариантов использования** 13](#_Toc73082392)

[**2.6 Структура проекта** 15](#_Toc73082393)

[**2.7 Алгоритм решения задач** 15](#_Toc73082394)

[**2.7.1 Преобразование изображения в массив байтов** 15](#_Toc73082395)

[**2.7.2 Открытие pdf-файла в браузере** 16](#_Toc73082396)

[**3. Архитектура проекта** 16](#_Toc73082397)

[**3.1 Использование администратором** 16](#_Toc73082398)

[**3.2 Использование стандартным пользователем** 19](#_Toc73082399)

[**Заключение** 23](#_Toc73082400)

[**Приложение А** 24](#_Toc73082401)

[**Приложение Б** 24](#_Toc73082402)

[**Приложение В** 25](#_Toc73082403)

[**Приложение Г** 27](#_Toc73082404)

# **Введение**

Целью данной работы является создание приложения для сохранения каталогизации прочитанных книг и отслеживания своего прогресса. Это программное средство ставит перед собой задачу по упрощению и повышению удобства:

* отслеживания статистики по прочитанным книгам;
* трекинга привычки читать;
* использования программы за счёт графического интерфейса;

Достоинства программы:

* удобный графический интерфейс в сравнении с конкурентами;
* бесплатный доступ;
* вам не потребуется ключ аппаратной защиты для активации программы;

Таким образом, приложение будет полезно для тех, кто хочет привить себе привычку регулярно и много читать, отслеживая при этом прогресс своего навыка и объем изученного.

Структура приложения: приложение, написанное на С#. Оно устанавливается на ПК и представляет собой графический интерфейс.

ПК приложение для оптимальной работы требует:

* ОС Windows 10;
* установленный Microsoft SQL Server 2012;
* установленный .Net Framework 4.7.2 или более поздней версии.

Функционал приложения позволяет:

* добавлять/изменять/удалять данные из базы данных;
* получить информацию о книге;
* получить информацию о пользователях;
* получить информацию о статистике по количеству прочитанных книг;
* получить информацию о статистике по жанрам прочитанных книг;
* вести свой каталог литературы;
* просматривать системные предустановленные в приложении книги и добавлять их в список предпочтений.

# **1. Аналитический обзор прототипов**

На сегодняшний день существует множество программ для просмотра и каталогизации книг. Системы программирования дают возможность удобно работать с базами данных.

Использование современных технологий – естественный выбор для создания приложений на современной технологической базе.

Рассмотрим несколько приложений, реализующих вышеупомянутые функции и использованных в качестве прототипов при разработке курсового проекта.

### **1.1 «BOOKSHELF»**

Данная программа была разработана под мобильные операционные системы IOS и Android. Здесь можно записывать прочитанные книги, писать собственные рецензии, создавать виртуальные книжные полки по жанрам или придумать собственное деление, и добавлять книги в список желаний.

Также приложение позволяет вывести статистику: вы можете узнать, сколько прочитали за год или месяц, сколько в среднем страниц в день читаете и какова средняя толщина ваших книг.

Интерфейс программы представлен на рисунке 1.1.

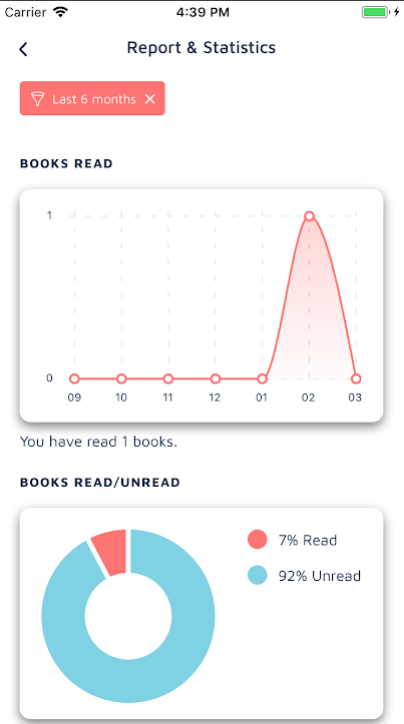


Рисунок 1.1 – «BOOKSHELF»

### **1.2 «HANDY LIBRARY»**

Данное приложение было разработана под мобильную операционную систему Android. Приложение позволяет управлять своей домашней библиотекой, фильтровать книги на любимые, прочитанные и непрочитанные, создавать список желаний, искать книги онлайн и писать собственные отзывы или любимые цитаты. Помимо этого, присутствуют следующие функции:

* если пользователь одалживает или занимает книги, приложение напомнит вовремя вернуть;
* есть возможность поставить напоминание, чтобы выработать в себе регулярную привычку читать;
* можно создавать собственные подборки книг и делиться ими;

Интерфейс программы представлен на рисунке 1.2.

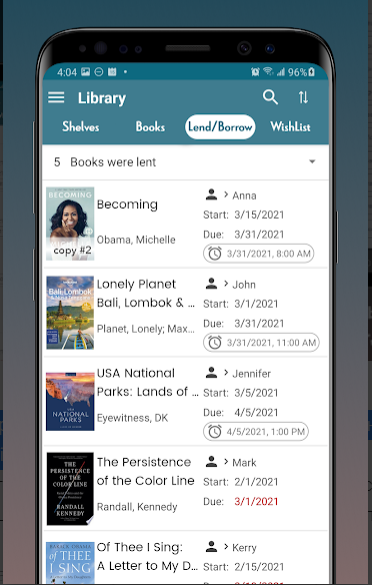


Рисунок 1.2 – «HANDY LIBRARY»

### **1.3 «Reading Planner»**

Данная программа была разработана под мобильные операционные системы IOS и Android.

Приложение позволяет добавлять книгу, устанавливать дату для прочтения и оптимальный режим — по одной странице в день, по разделу на неделю и так далее.

Приложение напоминает пользователю о книге и показывает, насколько он отстаёт от своего плана или опережает его. Данная функция полезна при обучении, когда нет мотивации осваивать материал, но необходимо сделать это до указанного срока.

Интерфейс программы представлен на рисунке 1.3.

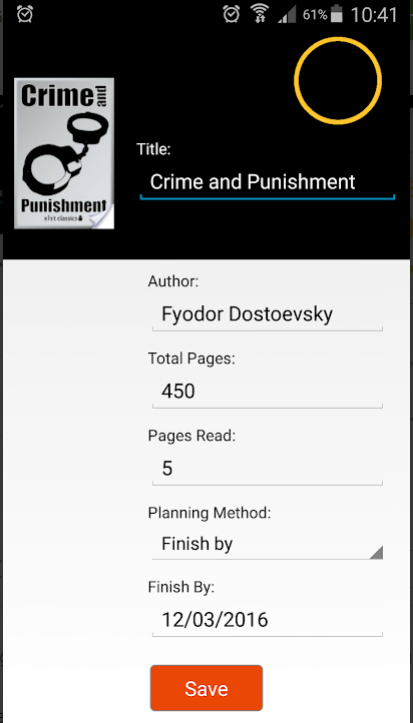


Рисунок 1.3 – «Reading Planner»

### **1.4 «Bookmate»**

Данная программа была разработана под мобильные операционные системы IOS и Android.

Приложение представляет собой социальную сеть. «Bookmate» предоставляет возможность добавлять пользователей в раздел «Друзья», следить за тем, что они читают и слушают, делиться отзывами и цитатами, получать лайки.

Помимо этого, приложение является онлайн-библиотекой с книгами в формате EPUB и FB2 и аудиокнигами на украинском, русском и других языках.

Приложение можно синхронизировать со всеми устройствами и читать и слушать в режиме офлайн.

Интерфейс программы представлен на рисунке 1.4.

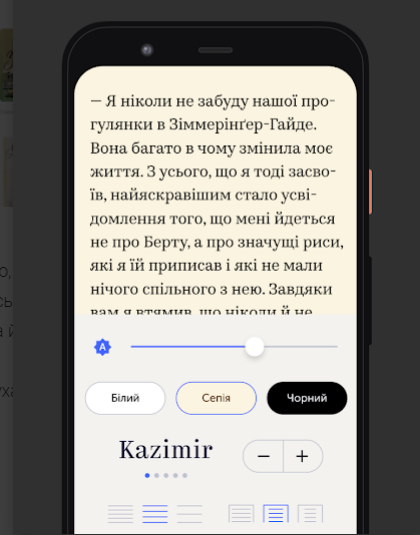


Рисунок 1.4 – «Bookmate»

## **1.5 «Read More: Reading Habit Tracker»**

Данное приложение было разработана под мобильную операционную систему Android.

Приложение позволяет хранить книги, которые читаете или уже прочитали, составлять списки желаний, искать новые книги и вести статистику прочитанного.

Можно также добавлять цитаты из книг — записывать в приложении или делать фото страниц и подчеркивать маркером непосредственно в телефоне.

За достижение новых целей в приложении предусмотрено награждение призами — электронными книжными закладками.

Интерфейс программы представлен на рисунке 1.5.

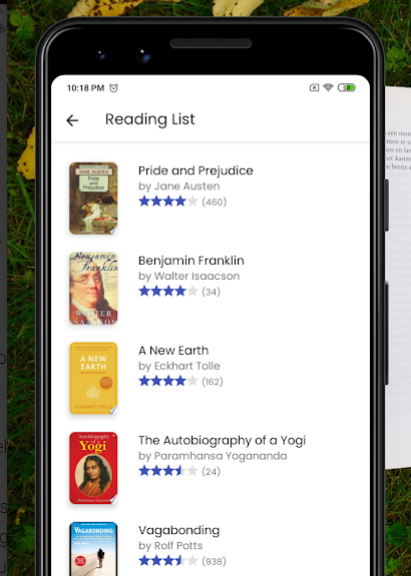


Рисунок 1.5 – «Read More: Reading Habit Tracker»

## **1.6 «GoodReads»**

Данная программа была разработана под мобильные операционные системы IOS и Android.

Приложение позволяет добавлять книги в электронную библиотеку пользователя, писать собственные рецензии, делиться отзывами с другими пользователями, а также вести собственную статистику и создавать списки пожеланий.

Интерфейс приложения представлен на рисунке 1.6.

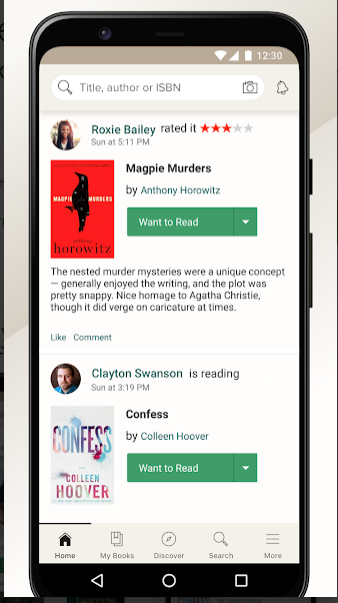


Рисунок 1.6 – «GoodReads»

# **2 Разработка архитектуры проекта**

## **2.1 Архитектура системы**

Для курсового проекта был использован способ разработки программных модулей, основанный на архитектуре клиентского приложения. Приложение состоит из двух частей:

* Клиентская часть. В этом модуле осуществляется взаимодействие пользователя с электронным каталогом с помощью графического интерфейса. В клиентской части пользователь сможет получить полную информацию об объектах приложения, в зависимости от присвоенной ему роли (пользователь или администратор).
* Базы данных. В базе данных описываемого приложения хранится информация о пользователе, пользовательском каталоге, о системном каталоге, книгах и прогрессе пользователя.

## **2.2** **Проектирование логической структуры базы данных**

Для создания базы данных описываемого приложения использовалась система управления реляционными базами данных Microsoft SQL Server 2012.

База данных – это совокупность данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и манипулирования данными, независимо от прикладных программ.

Система управления базами данных – совокупность программ и языковых средств, предназначенных для управления данными в базе данных, ведения базы данных и обеспечения взаимодействия ее с прикладными программами. База данных данного курсового проекта состоит из 4 таблиц. Схема базы данных изображена на рисунке 2.1.

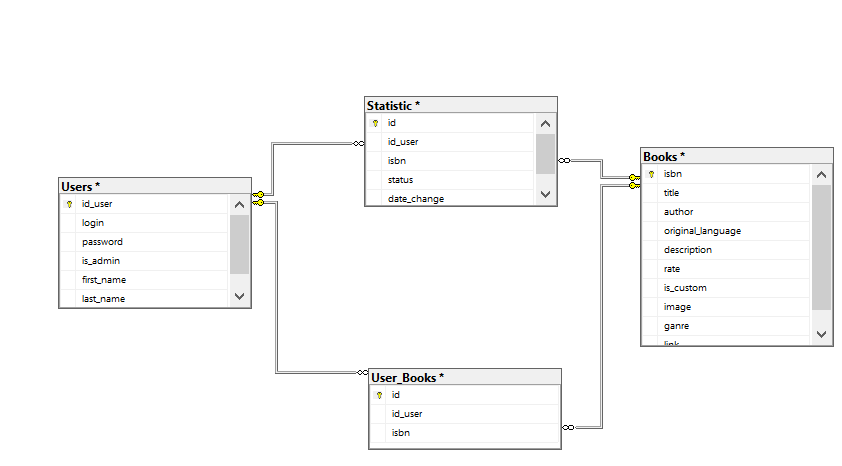


Рисунок 2.1 – Схема базы данных «READER»

Таблица Users используется для хранения информации о пользователе. Включает 6 столбцов: id\_user, login, password, is\_admin, first\_name, last\_name. Столбец id\_user является ключевым и уникальным, в нём хранится идентификатор пользователя. Столбец login содержит логин пользователя. Столбец password содержит пароль пользователя. Столбец is\_admin содержит информацию о том, является ли пользователь администратором. Столбцы first\_name и last\_name содержать имя и фамилию пользователя соответственно.

Таблица Book используется для хранения информации о книгах. Таблица состоит из 10 столбцов: isbn, title, author, original\_language, description, rate, is\_custom, image, ganre, link. Столбец isbn является ключевым и уникальным, выступает в качестве идентификатора книги. Столбец title содержит название книги. Столбец author содержит имя автора книги. Столбец original\_language содержит язык оригинала произведения. Столбец description содержит краткое описание книги. Столбец rate содержит рейтинг книги. Столбец is\_custom показывает, кем книга была добавлена в базу данных: если значение равно 1 - книга добавлена пользователем, если равно 0 – администратором. Столбец image содержит изображение, используемое в качестве обложки. Столбец ganre сожержит название жанра, в котором написана книга. Столбец link содержит ссылку на файл с содержанием книги.

Таблица Statistics используется для хранения данных, используемых для формирования статистики по прочитанным пользователем книгам. Таблица состоит из 5 столбцов: id, id\_user, isbn, status, date\_change. Столбец id является ключевым и уникальным, используется в качестве идернификатора строки таблицы. Столбец id\_user хранит идентификатор пользователя и используется для связи с таблицей Users. Столбец isbn содержит идентификатор книги и используется для связи с таблицей Books. Столбец status хранит инфрмацию о статусе книги: может принимать значения «Читаю» и «Прочитано». Столбец date\_change хранит дату внесенных изменений.

Таблица User\_Books используется для хранения информации о том, какие книги были добавлены пользователями в их личные каталоги. Таблица содержит 3 столбца: id, id\_user, isbn. Столбец id является ключевым и уникальным, используется в качестве идентификатора строки таблицы. Столбец id\_user хранит идентификатор пользователя, который добавил таблицу, и используется для связи с таблицей Users. Столбец isbn хранит идентификатор книги и используется для связи с таблицей Books.

## **2.3 Проектирование структуры окон**

Приложение включает в себя 6 окон и 8 страниц. Связи между ними представлены на рисунке 2.2.

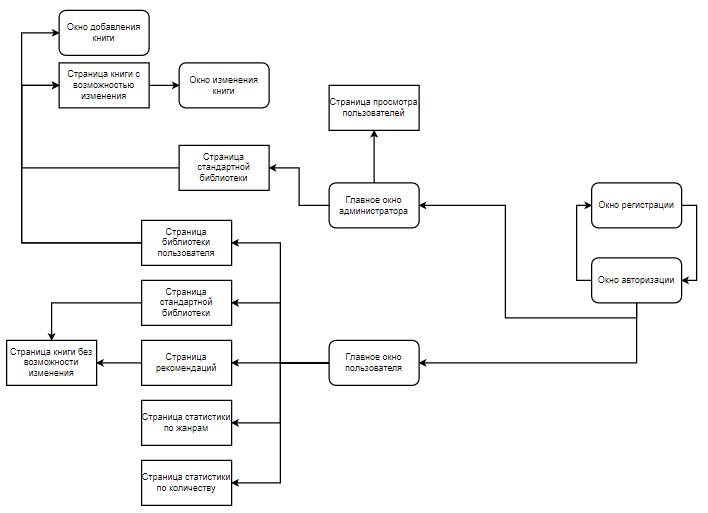


Рисунок 2.2 – Структура окон

Стартовым окном является окно авторизации. Пользователь либо вводит данные и получает доступ к приложению, либо переходит на окно регистрации.

В зависимости от роли, пользователь переходит либо на окно администратора, либо стандартного пользователя.

Окно стандартного пользователя дает доступ к странице стандартного каталога приложения и странице с рекомендованными книгами из стандартной библиотеки, которое, в свою очередь, позволяет перейти на страницу конкретной книги. Также из окна пользователя можно попасть на страницу со статистикой пользователя по количеству прочитанного в текущем году и на страницу со статистикой пользователя по прочитанному по жанрам. Помимо этого, стандартный пользователь имеет доступ к странице с его личным каталогом книг, которая связана с окном добавления нового элемента. Также можно перейти на страницу конкретной книги. Если книга добавлена пользователем, то ее страница связана с окном редактирования. Таким образом, пользователь имеет возможность добавлять книги только в личный каталог и редактировать информацию только о тех книгах, которые добавил сам.

Из окна администратора можно попасть на страницу просмотра пользователей, а также на станицу стандартной библиотеки. При этом открыт доступ к окну добавления книги, а также к странице с возможностью изменения и окном редактирования книги.

## **2.4 Пример диаграммы последовательностей**

Рассмотрим диаграмму последовательностей для авторизации пользователя. Диаграмма представлена на рисунке 2.3.

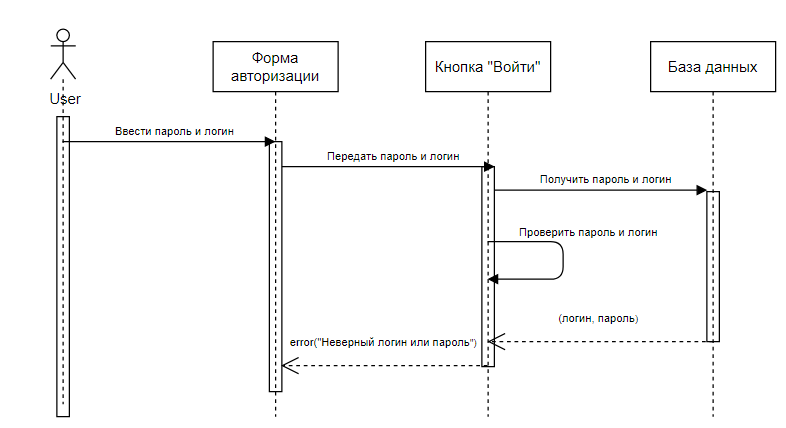


Рисунок 2.3 – Диаграмма последовательностей

## **2.5 Пример вариантов использования**

Рассмотрим диаграммы вариантов использования для стандартного пользователя и администратора. Диаграмма для пользователя представлена на рисунке 2.4, диаграмма администратора – на рисунке 2.5.

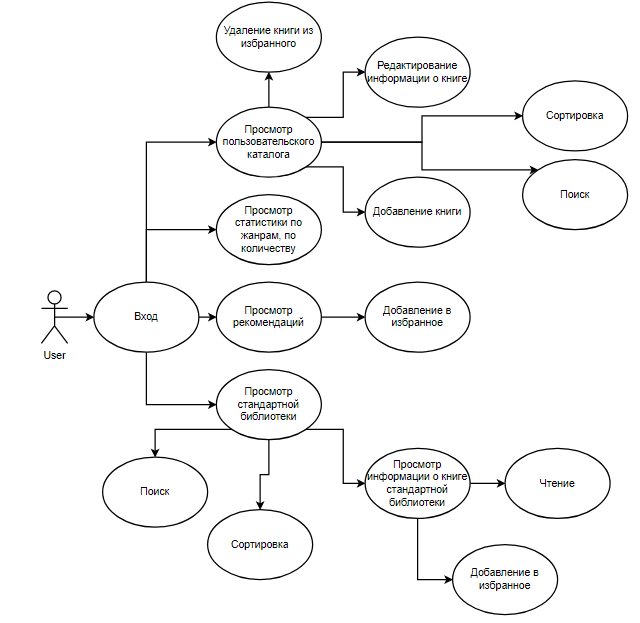


Рисунок 2.4 – Диаграмма вариантов использования для пользователя

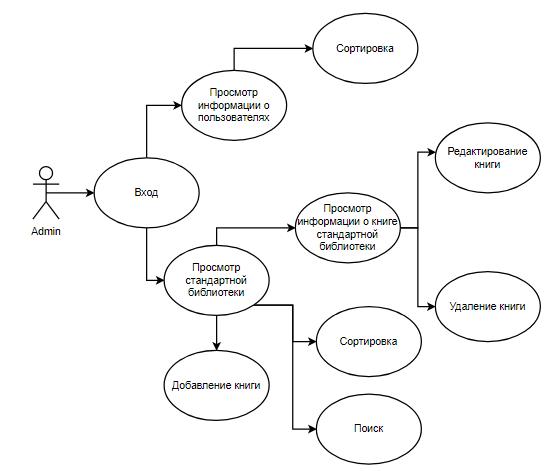


Рисунок 2.5 – Диаграмма вариантов использования для администратора

## **2.6 Структура проекта**

Структура проекта представлена диаграммой классов в Приложении А.

## **2.7 Алгоритм решения задач**

### **2.7.1 Преобразование изображения в массив байтов**

В базе данных файлы хранятся в виде двоичных данных, в то время как UIElement Image принимает изображения в виде Source. Для корректной записи изображения в базу данных необходимо преобразовать его в массив байтов. Для этого, после открытия файла с помощью класса OpenFileDialog , необходимо побайтно считать его и записать в переменную типа byte[] с помощью метода File.ReadAllBytes(). Пример кода приведен на рисунке 2.6.

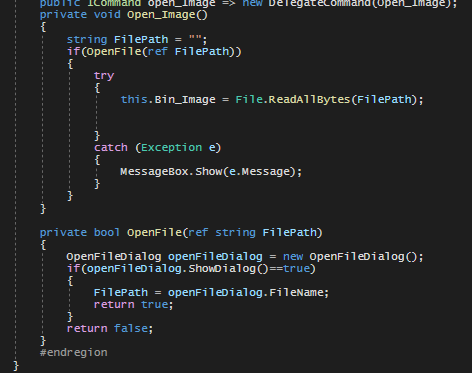


Рисунок 2.6 – Запись изображения в объект

### **2.7.2 Открытие pdf-файла в браузере**

Для того, чтобы открыть pdf-файл, находящийся на Google Drive, используем класс Process и передадим в конструктор в качестве параметра переменную, содержащую ссылку на файл. Пример кода представлен на рисунке 2.7.

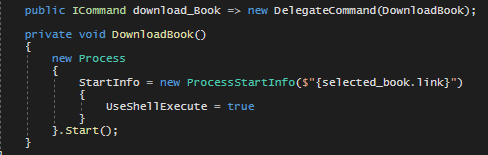


Рисунок 2.7 – Открытие pdf-файла

# **3. Архитектура проекта**

## **3.1 Использование администратором**

После запуска приложения пользователь видит окно авторизации приложения, представленное на рисунке 3.1.

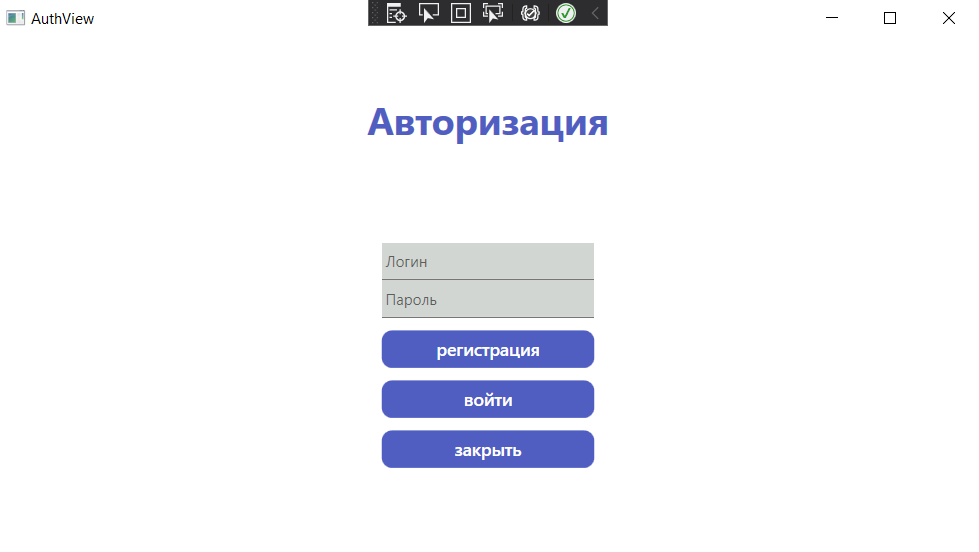


Рисунок 3.1 – Окно авторизации

По клику открывается страница с информацией о книге. Администратор может изменять и удалять книги системного каталога. Страница представлена на рисунке 3.2

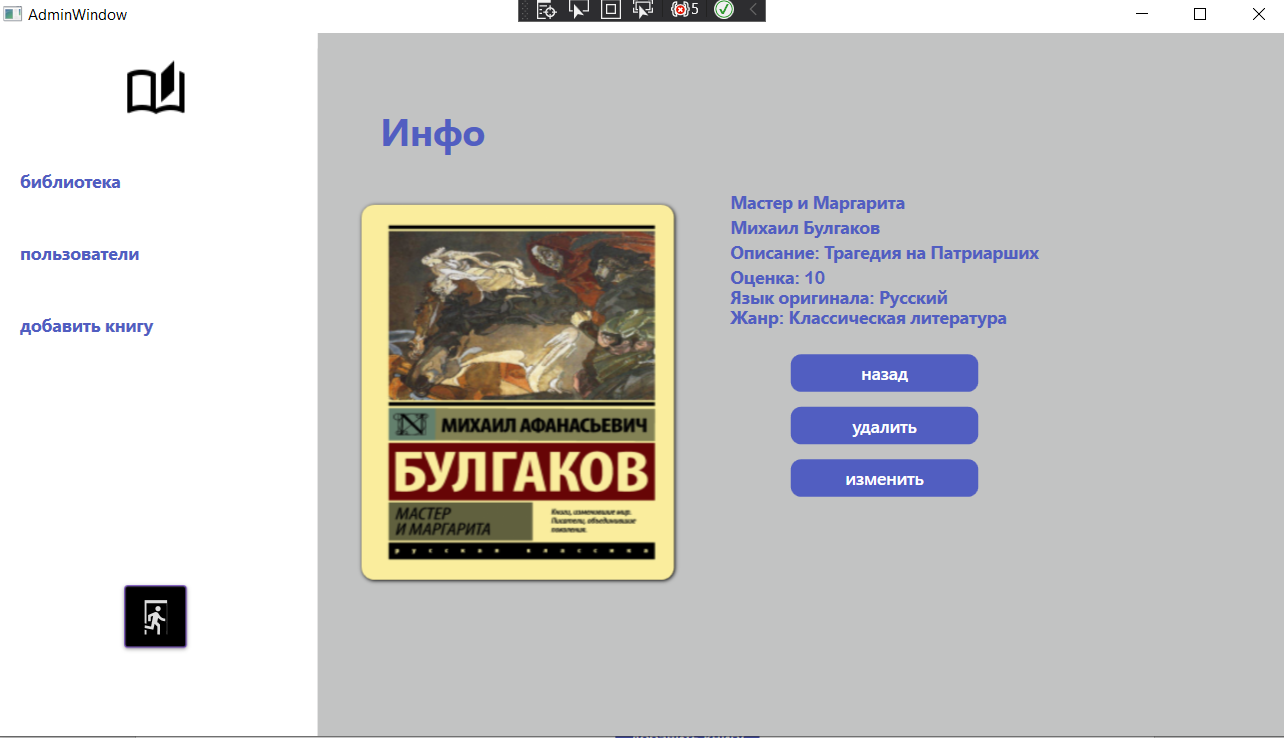


Рисунок 3.2 – Информация о книге

Администратор имеет право добавлять книги в системный каталог. Окно добавления представлено на рисунке 3.3.

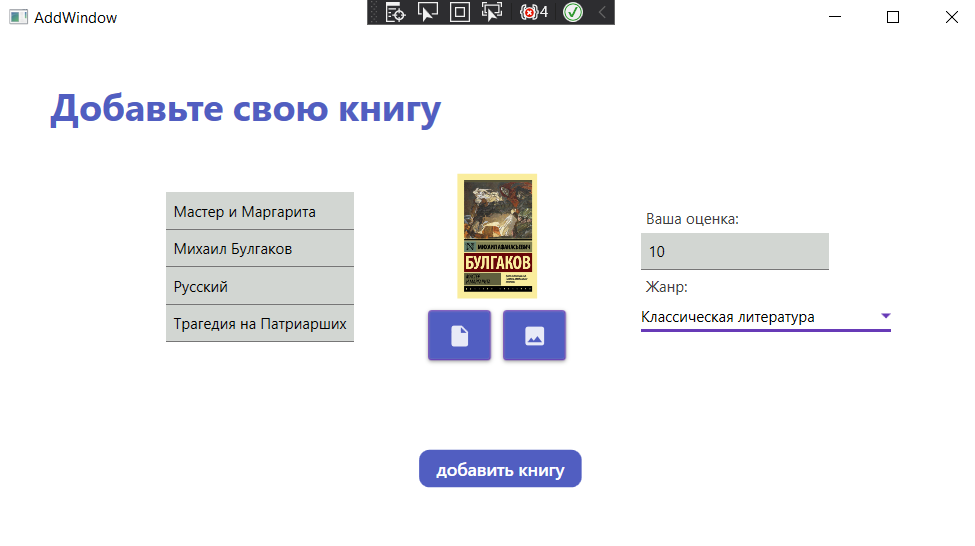


Рисунок 3.3 – Окно добавления администратора

При нажатии на кнопку «Изменить» откроется соответствующее окно, представленное на рисунке 3.4.

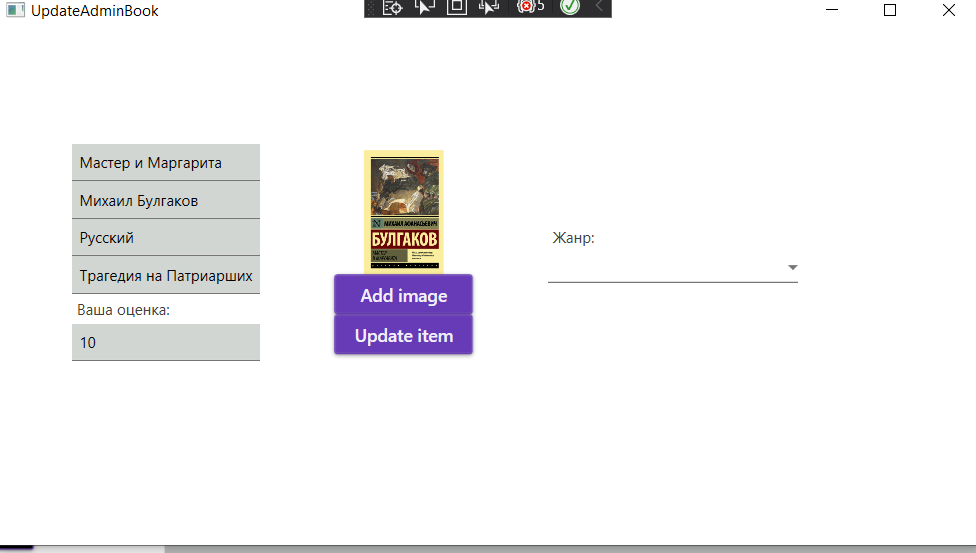


Рисунок 3.4 – Окно редактирования

Также администратор может просматривать информацию о пользователях. Соответствующая страница представлена на рисунке 3.5.

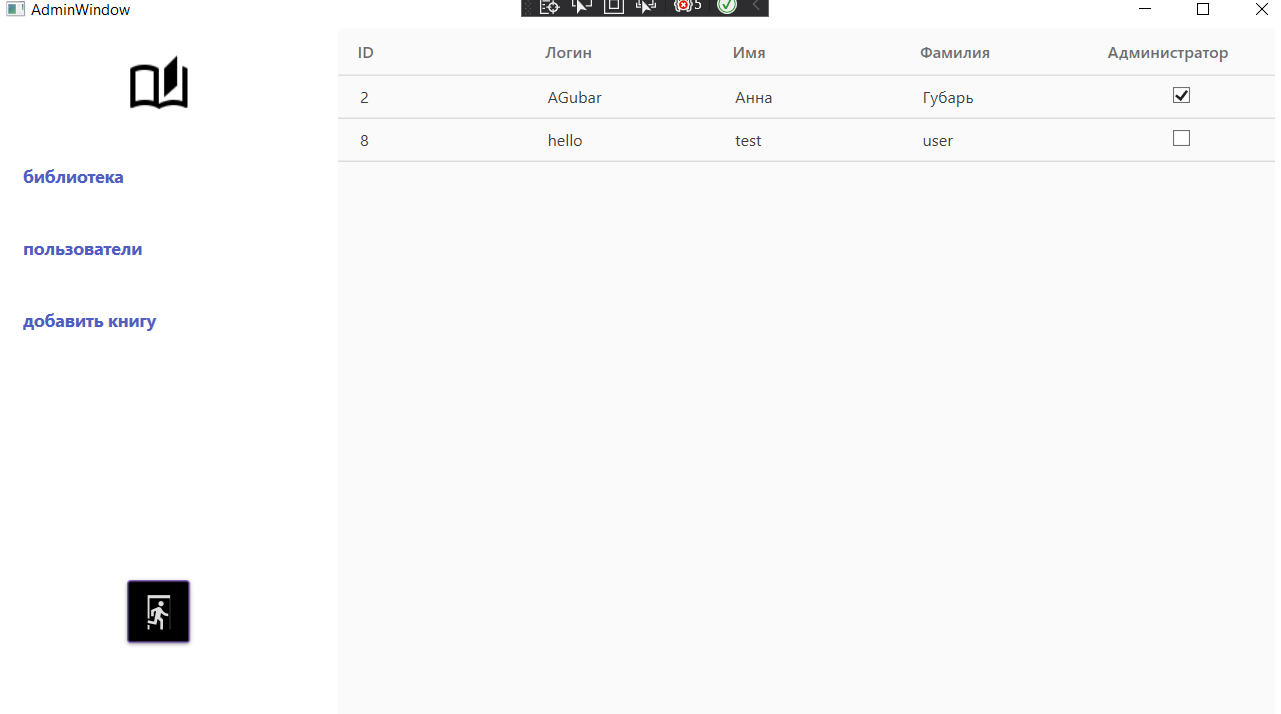


Рисунок 3.5 – Информация о пользователях

## **3.2 Использование стандартным пользователем**

После запуска приложения пользователь видит окно авторизации приложения, представленное на рисунке 3.6.

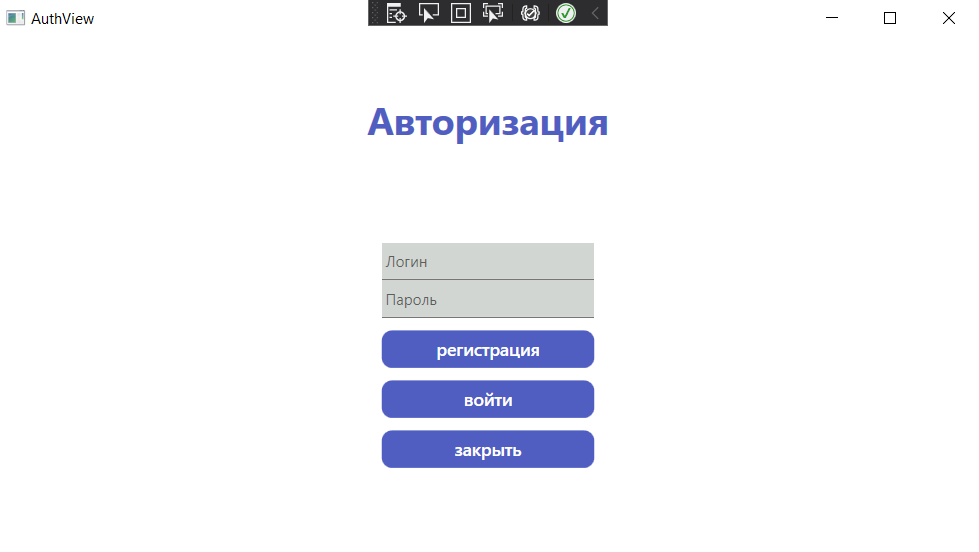


Рисунок 3.6 – Окно авторизации

В случае, если пользователь ранее не использовал приложение. Он может перейти в окно регистрации. Окно регистрации представлено на рисунке 3.7.

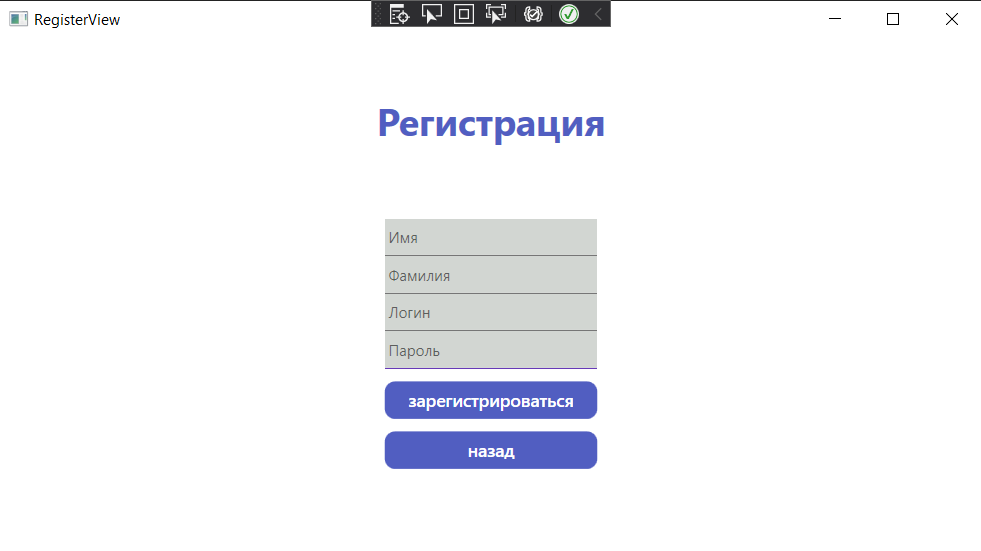


Рисунок 3.7 – Окно регистрации

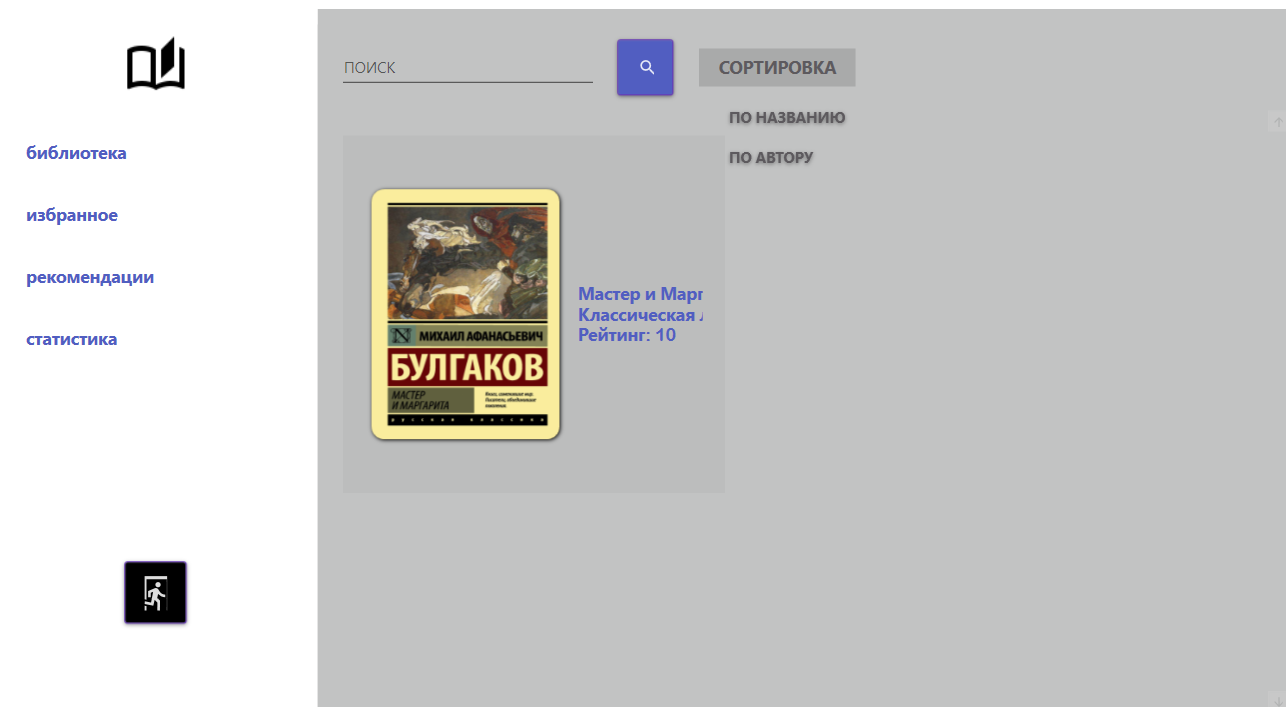
После авторизации пользователь попадает на главное окно, на котором по умолчанию отображается стандартная библиотека. Окно представлено на рисунке 3.8 

Рисунок 3.8 – Главное окно

По клику открывается информация о книге. Страница представлена на рисунке 3.9.

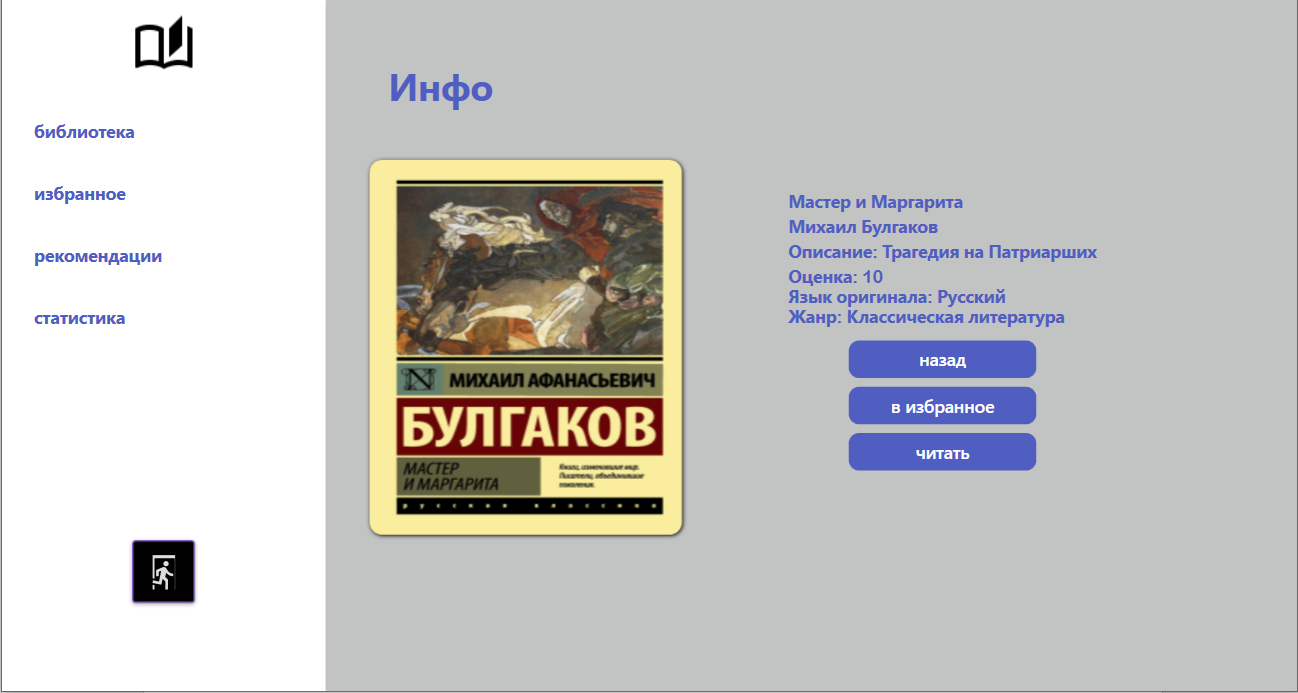


Рисунок 3.9 – Информация о книге из стандартной библиотеки

Пользователь может добавить книгу в свой локальный каталог. Локальный каталог пользователя и страница добавления книги представлены на рисунке 3.10 и рисунке 3.11 соответственно.

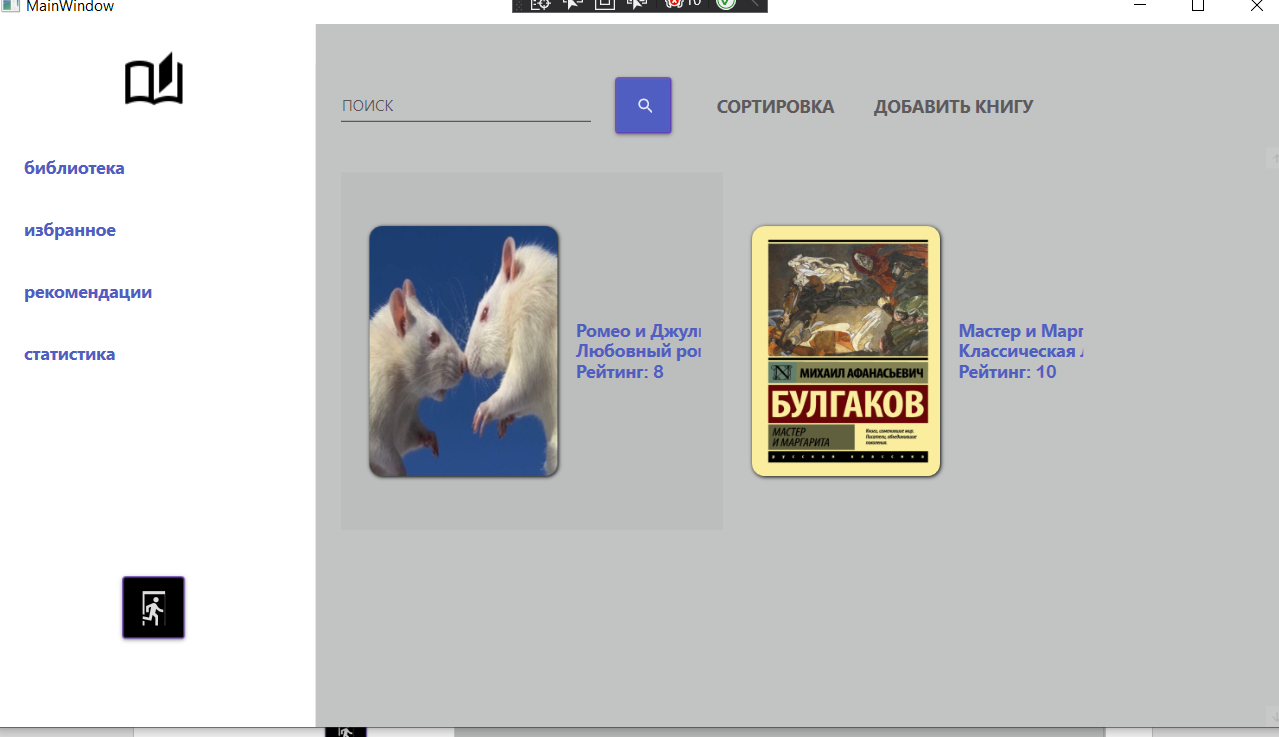


Рисунок 3.10 – Локальный каталог пользователя

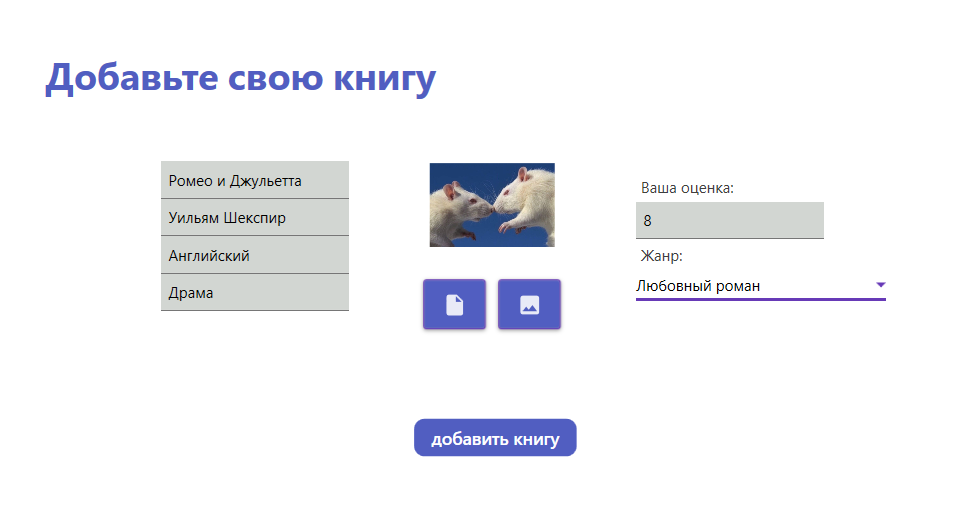


Рисунок 3.11 – Добавление книги в каталог пользователя

Пользователь также имеет право удалять и редактировать объекты из каталога. Окно редактирования представлено на рисунке 3.12.



Рисунок 3.12 – Редактирование книги.

# **Заключение**

Данный проект был разработан для тех, кто любит читать и хочет отслеживать свой прогресс, а также для тех, кто хочет привить себе привычку читать больше. Главной задачей было сделать простое и удобное приложение. Для этого была использована технология WPF и SQL-Server, а также ADO.Net, в качестве способа взаимодействия с базами данных.

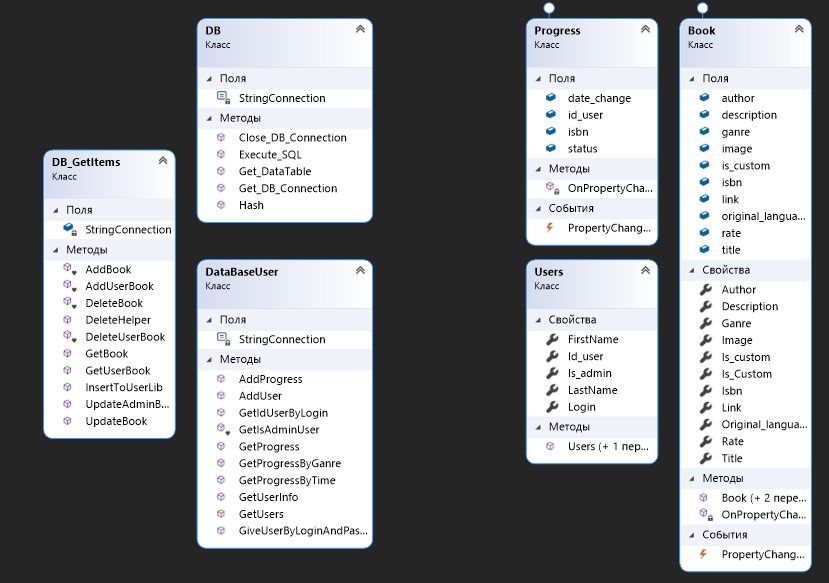
Само приложение – первый опыт в данной области. Естественно, существует ряд приложений, имеющих функционал и дизайн в разы лучше. Подводя итог всей курсовой работы, можно сделать выводы, что в разработке приложений мне предстоит еще долгий путь.

# **Список использованной литературы**

1. Цыганенко, Н. П. Курс лекций по предмету «Программирование серверных кроссплатформенных приложений» - 2019.
2. N-layer архитектура [Электронный ресурс] / Github – Режим доступа: <https://accetone.github.io/cwp/10/#/> - Дата доступа: 16.04.2019.
3. Дизайн API [Электронный ресурс] – режим доступа https://docs.microsoft.com/en-us/azure/architecture/b.. Дата доступа 05.04.2019
4. Архитектуры WEB приложений [Электронный ресурс] – режим доступа https://docs.microsoft.com/en-us/dotnet/standard/mode.. Дата доступа 12.04.2019
5. ADO.Net [Электронный ресурс] – режим доступа https://docs.microsoft.com/en-us/ef/core/ Дата доступа 01.04.2019

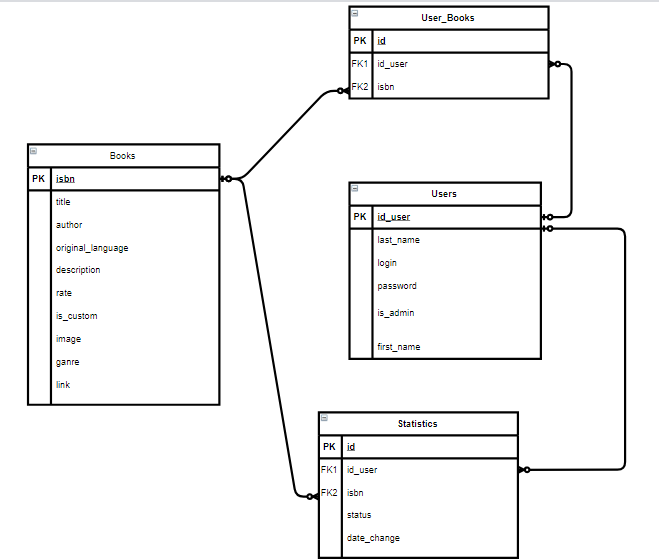
# **Приложение А**

UML-диаграмма классов.



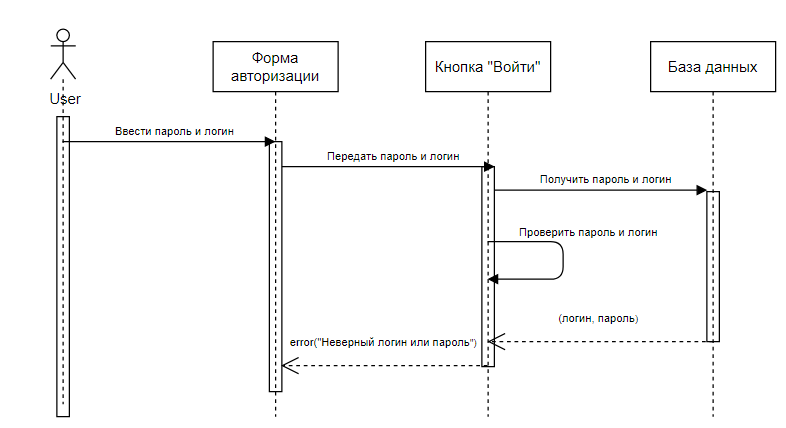
# **Приложение Б**

Структурная схема базы данных



# **Приложение В**

Диаграмма последовательности



# **Приложение Г**

Диаграммы вариантов использования