

МИНОБРАЗОВАНИЯ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет компьютерных наук

Кафедра информационных технологий управления

Веб-приложение для обмена книгами «BookFlow»

Курсовой проект по дисциплине  
«Технологии программирования»

09.03.02 Информационные системы и технологии  
Информационные системы и технологии управления предприятием

Зав. кафедрой	_____ д.т.н., профессор М. Г. Матвеев
Обучающийся	_____ ст. 3 курса оч. отд. И. В. Кислова
Обучающийся	_____ ст. 3 курса оч. отд. П. А. Авдеева
Обучающийся	_____ ст. 3 курса оч. отд. О. Р. Дубовой
Руководитель	_____ В.С. Тарасов, ст. преподаватель __.__.20__

# СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	4
1 Постановка задачи.....	6
1.1 Требования к разрабатываемой системе .....	7
1.1.1 Функциональность для неавторизованного пользователя .....	7
1.1.2 Функциональность для авторизованного пользователя.....	8
1.1.3 Функциональность сайта для администратора .....	9
2 Анализ предметной области .....	11
2.1 Терминология (глоссарий) предметной области .....	<b>Error! Bookmark not defined.</b>
2.2 Обзор аналогов .....	11
2.2.1 LiveLib.....	11
2.2.2 Bookcrossing.ru/Книговорот.....	14
2.2.3 Выводы из обзора аналогичных решений.....	17
3 Диаграммы.....	19
3.1 Диаграмма прецедентов (Use-case diagram).....	19
3.2 Диаграмма деятельности (Activity diagram).....	19
3.3 Диаграмма последовательности (Sequence diagram) .....	20
3.4 Диаграмма развёртывания (Deployment diagram).....	22
3.5 Диаграмма состояний (Statechart diagram).....	23
3.6 ER–диаграмма .....	24
4 Аналитика .....	26
5 Реализация .....	27
5.1 Разработка frontend.....	27
5.2 Разработка backend.....	31

6 Навигация по веб-приложению .....	37
6.1 Страница регистрации .....	37
6.2 Страница авторизации .....	37
6.3 Главная страница.....	37
6.4 Страница «Мой профиль».....	38
6.5 Страница «Мои книги» .....	38
6.6 Страница «Книга» .....	39
6.7 Страница «Поиск».....	39
6.8 Страница «Раздел жалоб» .....	40
6.9 Страница «Статистика» .....	40
6.10 Главная страница администратора .....	41
ЗАКЛЮЧЕНИЕ .....	42
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ.....	44

## **ВВЕДЕНИЕ**

В современном мире информационные технологии стремительно проникают во все сферы жизни, значительно их трансформируя. Это влияние не обошло стороной и книжную индустрию. Традиционные способы приобретения и обмена книгами постепенно уступают место новым, более удобным и доступным онлайн-платформам.

В данной курсовой работе рассмотрен процесс разработки веб-приложения для обмена книгами, которое позволяет пользователям делиться своими книгами с другими людьми, находить книги, которые им интересны, и получать новые впечатления от чтения. Для реализации данного приложения будут использованы современные технологии веб-разработки, такие как React, Node.js, MongoDB.

Целью данной курсовой работы является практическое применение знаний и навыков веб-разработки для создания коммерческого приложения для обмена книгами и отзывами о прочитанном, которое будет полезно для широкого круга пользователей. Работа будет содержать детальное описание всех этапов процесса разработки, включая анализ требований, проектирование приложения, реализацию, тестирование.

## 1 Терминология (гlossарий)

Неавторизованный пользователь — пользователь сайта, не имеющий полного доступа к функциям сайта.

Авторизованный пользователь (клиент) – авторизованный пользователь сайта, имеющий полный доступ к функциям сайта.

Администратор — авторизованный пользователь сайта, занимающийся управлением сайта, добавлением, удалением, редактированием карточек книг, удалением пользователей, отзывов.

Профиль (аккаунт) — веб-страница пользователя сайта с персональными данными о нем.

Сайт (веб-приложение) — интернет-ресурс, состоящий из одной, нескольких или множества виртуальных страниц.

Клиент-серверное приложение, в котором клиент взаимодействует с веб-сервером при помощи браузера [1].

Backend (серверная часть) – программно-аппаратная часть сервиса, которая хранится на сервере, обрабатывает полученные данные и отправляет ответ обратно.

Frontend (клиентская сторона) – интерфейс с набором функций, с которым взаимодействует пользователь.

## 2 Постановка задачи

Целью данного курсового проекта является создание сайта, который позволит:

- пользователям совершать обмен книгами в определенной местности, оставлять отзывы и оценки о прочитанном и других пользователях;
- заказчику получать прибыль с контекстной рекламы книг и мероприятий, посвященных книжной тематике.

Сайт позволяет решать следующие задачи:

- получать информацию о пользователях, с которыми обмениваемся книгами;
- обмениваться книгами;
- получать информацию о книгах;
- оставлять оценки пользователям по пятибалльной шкале;
- оставлять отзывы о книгах и оценивать их по пятибалльной шкале;
- вводить данные о пользователе в разделе профиля и редактировать их;
- удалять отзывы, карточки книг и аккаунты пользователей (от имени администратора);
- редактировать карточки книг (от имени администратора);
- осуществлять поиск книг на сайте;
- связываться с другими пользователями через оставленные ими контактные данные;
- добавлять и заполнять карточку книги;
- размещать книгу для обмена;
- указывать местоположение (город);
- просматривать статистические данные за определенный промежуток времени (от имени администратора).

## **2.1 Требования к разрабатываемой системе**

Сайт должен быть оформлен в одной цветовой палитре с использованием ограниченного набора шрифтов. У страниц сайта должен быть единый стиль. В оформлении приложения должно присутствовать разработанное название.

Все страницы сайта должны корректно отображаться на любом устройстве пользователя.

Необходимо корректное и одинаковое отображение страниц сайта в следующих браузерах:

- Yandex Browser, версия 23.3.7.24 или новее;
- Google Chrome, версия 118.0.5993.71 или новее;
- Mozilla Firefox, версия 115 и 115 ESR или новее;
- Microsoft Edge, версия 88.0.705.68 или новее.

### **2.1.1 Функциональность для неавторизованного пользователя**

Сайт предоставляет некоторый ряд функций по различным запросам пользователей.

Регистрация:

- при посещении сайта незарегистрированный пользователь имеет возможность создать учетную запись.

Авторизация:

- при авторизации запрашиваются личные данные пользователя, которые он вводил при регистрации;
- в случае успешной авторизации пользователю становятся доступны все основные функции сайта.

Поиск книг:

- пользователь имеет возможность искать книги по названию, автору и другим параметрам;

- результаты поиска должны отображаться в виде карточек с информацией о книгах.

Указание местоположения:

- пользователь имеет возможность задать определенную местность (город), в пределах которой он может совершать обмен книгами.

На данном этапе разработки веб-приложение должно позволять обмен книгами пользователям, которые находятся в городе Воронеж. Расширение географии проекта планируется в следующих версиях веб-приложения.

### **2.1.2 Функциональность для авторизованного пользователя**

Поиск книг:

- пользователь имеет возможность искать книги по названию, автору и другим параметрам;
- результаты поиска должны отображаться в виде карточек с информацией о книгах.

Размещение книг на обмен:

- пользователь имеет возможность размещать свои книги на обмен;
- пользователь имеет возможность снять книгу с обмена и редактировать введенные ранее контактные данные.

Получение книг:

- пользователь имеет возможность получать книги, размещенные на сайте;
- пользователь имеет возможность получить контактную информацию владельца книги для обсуждения ее получения.

Отзывы:

- пользователь имеет возможность оставлять отзывы в разделе карточке книги.

Управление профилем:



- пользователь имеет возможность просматривать и редактировать свой профиль, включая контактные данные и фотографию профиля, а также удалить его;
- пользователь имеет возможность посмотреть историю своих обменов книгами.

Размещение жалобы:

- в случае негативного поведения (употребление ненормативной лексики при написании отзыва, например) со стороны другого пользователя пользователь имеет возможность оставить жалобу в соответствующем разделе сайта;
- пользователь имеет возможность заполнить соответствующую форму для описания проблемы.

Указание местоположения:

- пользователь имеет возможность задать определенную местность (город), в пределах которой он может совершать обмен книгами.

На данном этапе разработки веб-приложение должно позволять обмен книгами пользователям, которые находятся в городе Воронеж. Расширение географии проекта планируется в следующих версиях веб-приложения.

### **2.1.3 Функциональность сайта для администратора**

Для модерации сайта существует авторизованный пользователь – администратор. Ему доступны следующие функции:

Просмотр жалоб:

- администратор имеет возможность посмотреть жалобы, оставленные пользователями сайта.

Удаление отзывов:

- администратор имеет возможность удалять отзывы в карточке книги.

Удаление аккаунта:

- администратор имеет возможность удалить аккаунт пользователя. В этом случае пользователь не сможет больше зарегистрироваться на сайте с указанными ранее контактными данными.

#### Удаление карточки книги:

- администратор имеет возможность удалить карточку книги.

#### Редактирование карточки книги:

- администратор имеет возможность редактировать данные в карточке книги.

#### Поиск книг:

- администратор имеет возможность искать книги по названию, автору и другим параметрам;
- результаты поиска должны отображаться в виде карточек с информацией о книгах.

#### Статистика:

- администратор имеет возможность посмотреть статистические данные за определенный временной период (неделю/месяц/год).

### **3 Анализ предметной области**

#### **3.1 Обзор аналогов**

Прежде, чем начинать разработку платформы для обмена книгами, необходимо проанализировать уже существующие платформы для понимания их сильных и слабых сторон. После сбора информации можно переходить к этапу разработки с учетом сделанных выводов по изученным аналогам.

##### **3.1.1 LiveLib**

LiveLib – русскоязычный веб-сервис, посвящённый литературе, обладающий некоторыми признаками социальных сетей. Сайт предоставляет информацию о книгах, писателях, издательствах и библиотеках. Позиционирует себя как книжный рекомендательный сервис, обладает базой пользовательских рецензий, освещает литературные новости и цены на книжные новинки. Также сервис позволяет зарегистрированным участникам вести учёт и анализ прочитанных произведений [2].

После входа на сайт авторизованный пользователь видит главный экран, который представлен на рисунке 1, с блоками рекламы, занимающими большую часть главного экрана, логотип, основное навигационное меню, строку поиска, значок уведомлений и аватар профиля пользователя. Для неавторизованного пользователя вместо значка уведомлений и аватара профиля пользователя будет кнопка «Войти». На рисунке 2 представлен главный экран для неавторизованного пользователя. Часть навигационного меню располагается в отдельном раскрывающемся списке при нажатии на иконку «Меню».

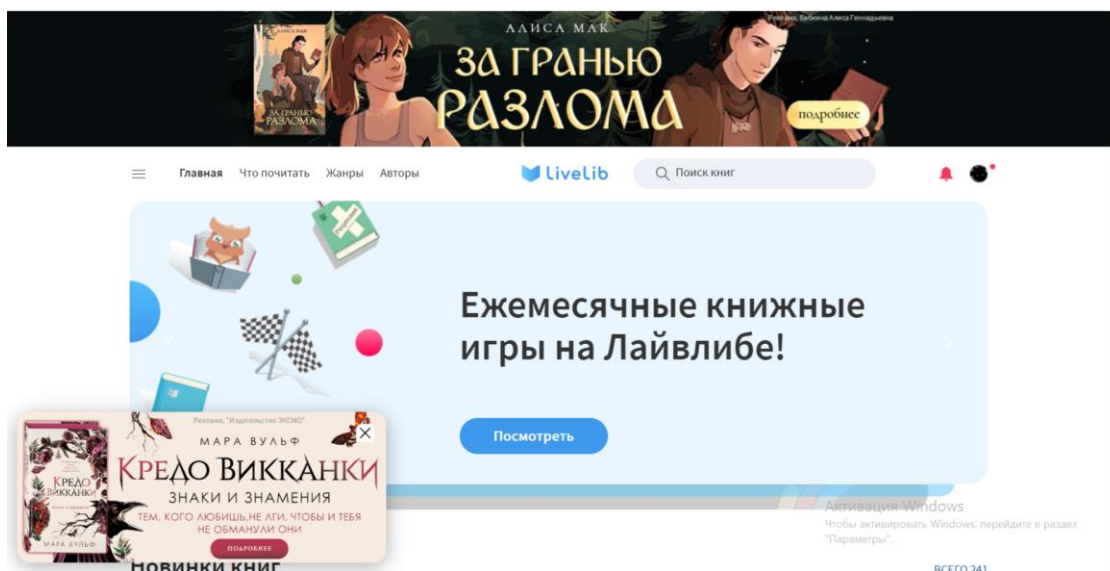


Рисунок 1 - Главная страница сайта «LiveLib» для авторизованного пользователя

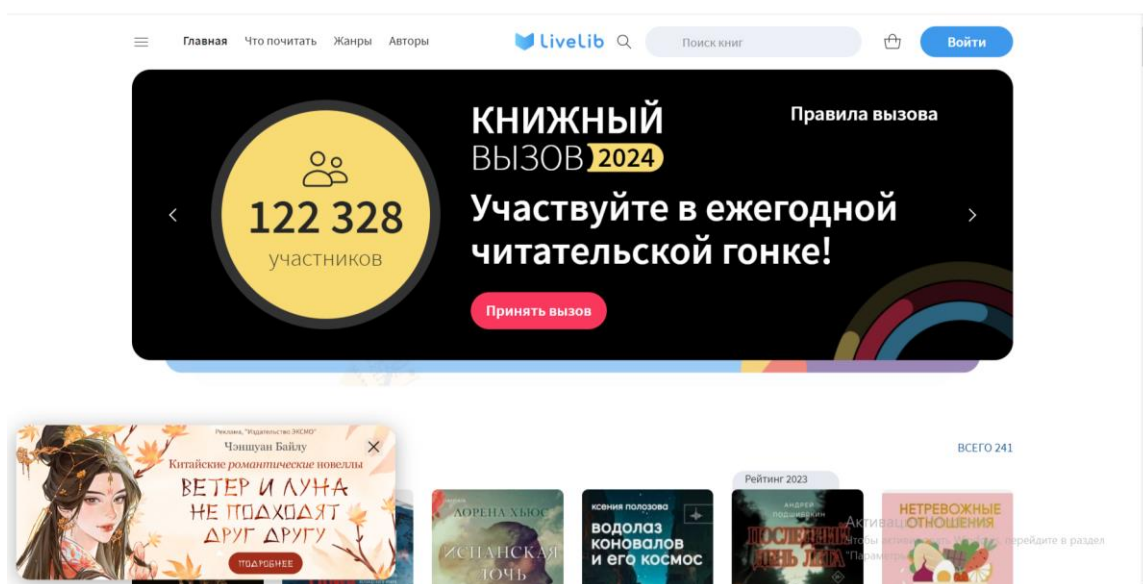


Рисунок 2 - Главная страница сайта «LiveLib» для неавторизованного пользователя

Только авторизованный пользователь может взаимодействовать с карточкой книги. При попытке неавторизованного пользователя взаимодействовать с книгой появляется форма авторизации/регистрации, представленная на рисунке 3. Только после авторизации или регистрации взаимодействие с карточкой книги при желании будет доступно.

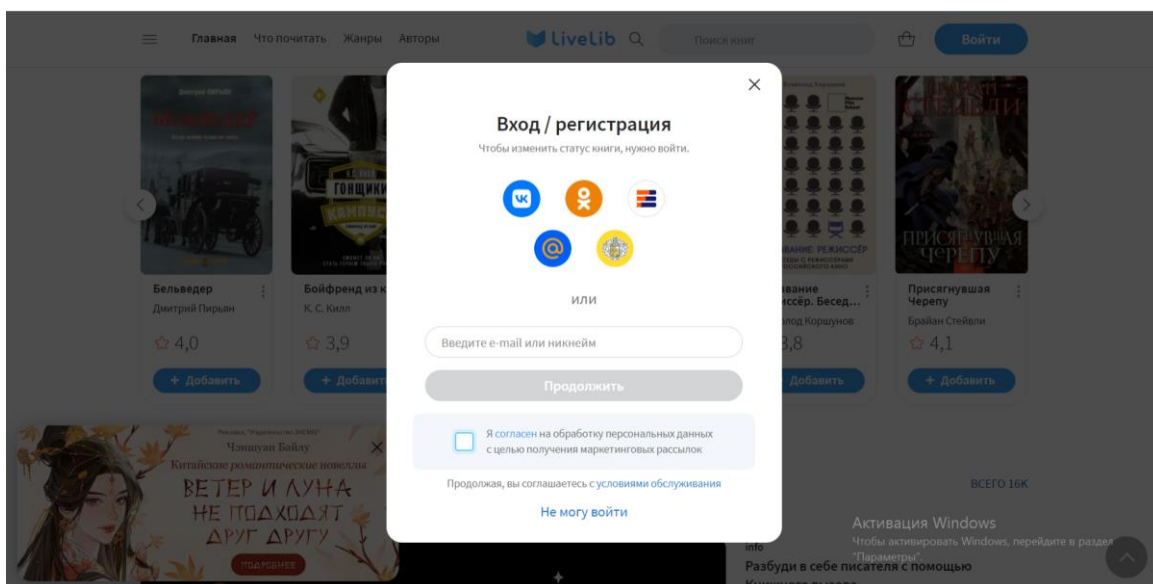


Рисунок 3 - Страница авторизации/регистрации «LiveLib»

Форма взаимодействия с карточкой книги представлена на рисунке 4.

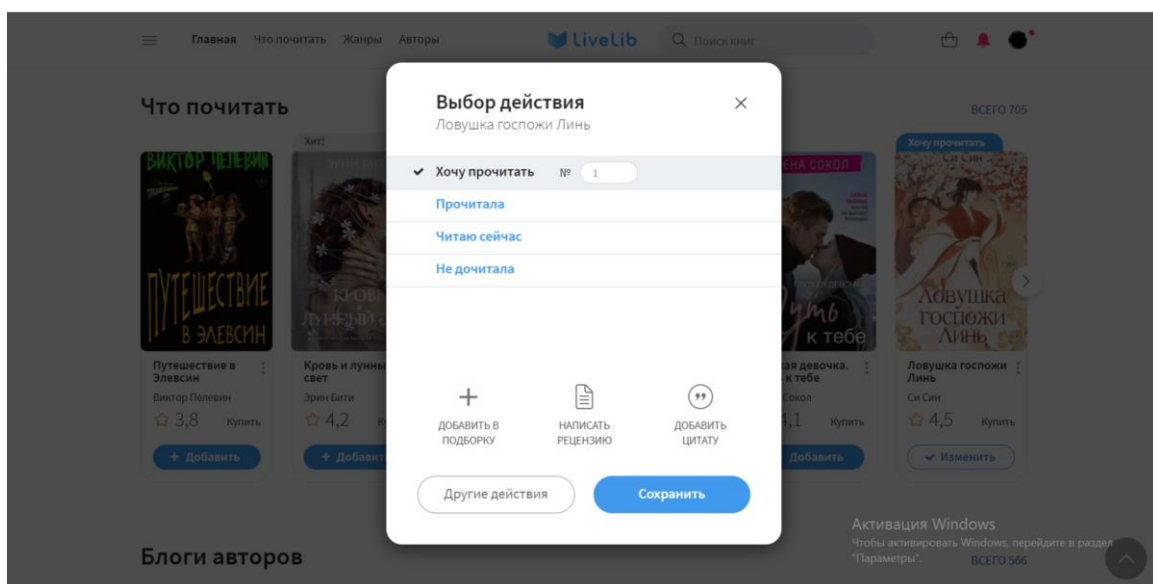


Рисунок 4 - Форма взаимодействия с карточкой книги после авторизации/регистрации

На сайте авторизированному пользователю доступна функция книгообмена, но она не пользуется популярностью у пользователей, так как данная платформа является в большей степени социальной сетью для любителей книг, чем платформой по обмену книгами. На рисунке 5 представлена форма для участия в книгообмене.

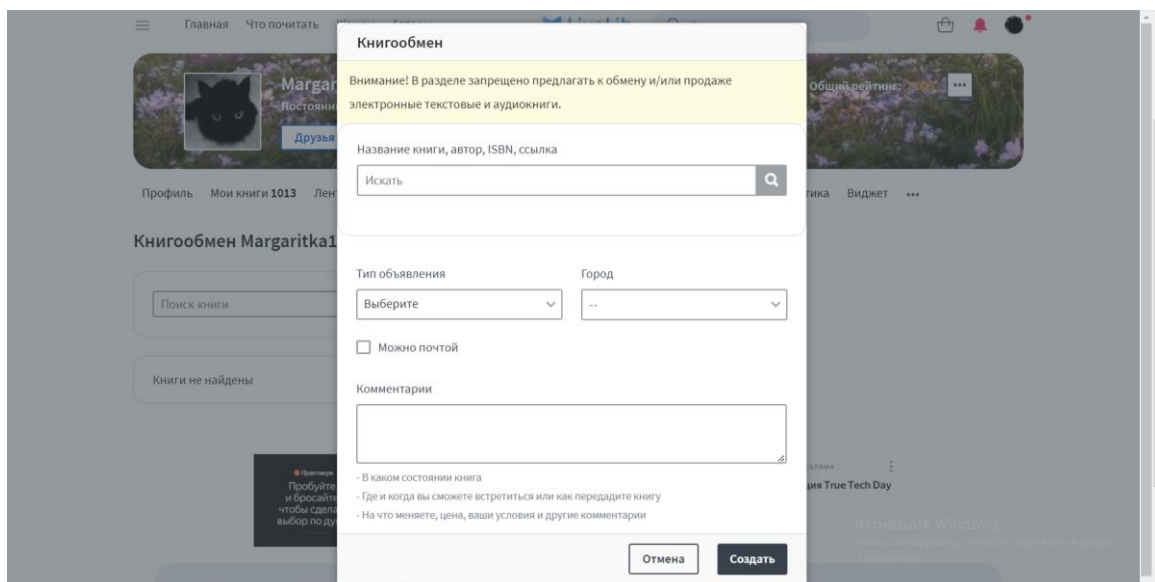


Рисунок 5 - Форма для участия в книгообмене на сайте «LiveLib»

#### Преимущества сайта:

- сайт имеет огромную базу данных с книгами как на русском, так и на иностранных языках;
- огромный функционал сайта помимо обмена книгами и рекомендаций;
- при оформлении книги в книгообмене личная информация пользователя не становится общедоступной;
- присутствует система оценки книг;
- сайт стимулирует пользователя уделять больше времени чтению, приняв участие в ежегодном «Книжном вызове».

#### Недостатки сайта:

- доступ ко всем разделам сайта затруднителен из-за запутанной навигации;
- отсутствует система оценки профиля пользователя;
- акцент приложения смещен на обмен рекомендациями между пользователями, а не на обмен книгами;
- обилие рекламы.

### 3.1.2 Bookcrossing.ru/Книговорот

Буккроссинг — это хобби и общественное движение, действующее по принципу социальных сетей и близкое к флешмобу. Суть движения заключается в том, что человек, прочитав книгу, оставляет её в общественном месте (парк, кафе, поезд, библиотека, станция метро) для того, чтобы другой, случайный человек мог эту книгу найти и прочитать. Предполагается, что тот, в свою очередь, повторит это действие. Слежение за «путешествием» книги осуществляется через специальные сервисы в Интернете, такие как «Книговорот» [3].

После входа на сайт пользователь видит главный экран, представленный на рисунке 6, с навигационным меню, возможностью указания города пользователя, строкой поиска, логотипом, предложением войти/зарегистрироваться, статистикой пользования сайтом. Также пользователю предлагается стать участником обмена книгами или же просто воспользоваться поиском интересующей книги.

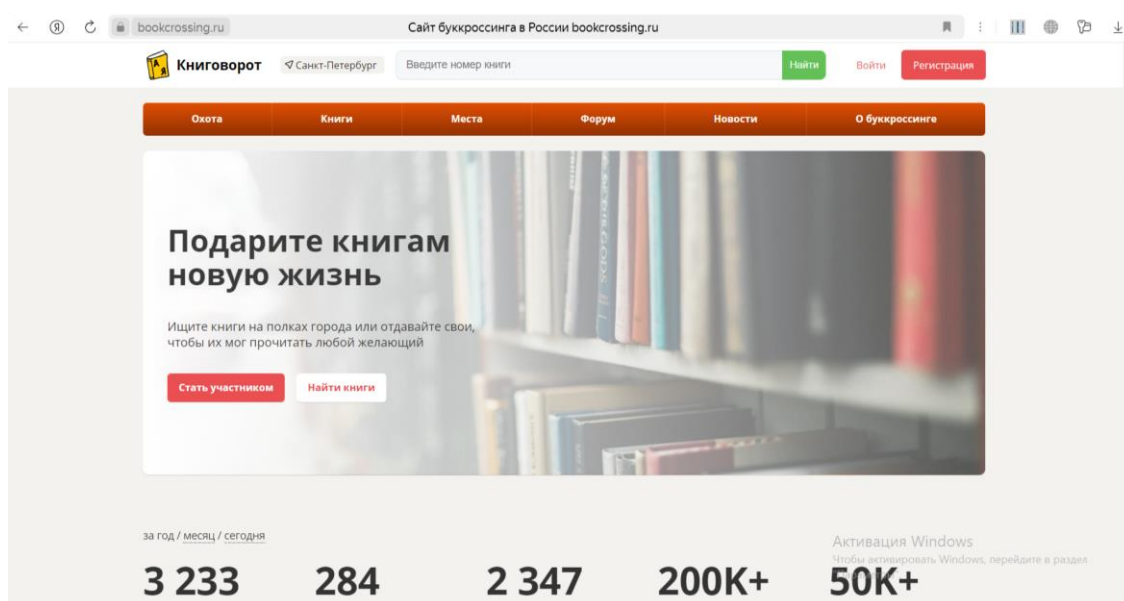


Рисунок 6 - Главная страница сайта «Книговорот»

Обмен книгами становится возможен после прохождения авторизации/регистрации на сайте. Пользователь находит нужную ему книгу на сайте и в карточке книги в разделе «Книговорот» может посмотреть кем и где был оставлен нужный ему экземпляр, как показано на рисунке 7.

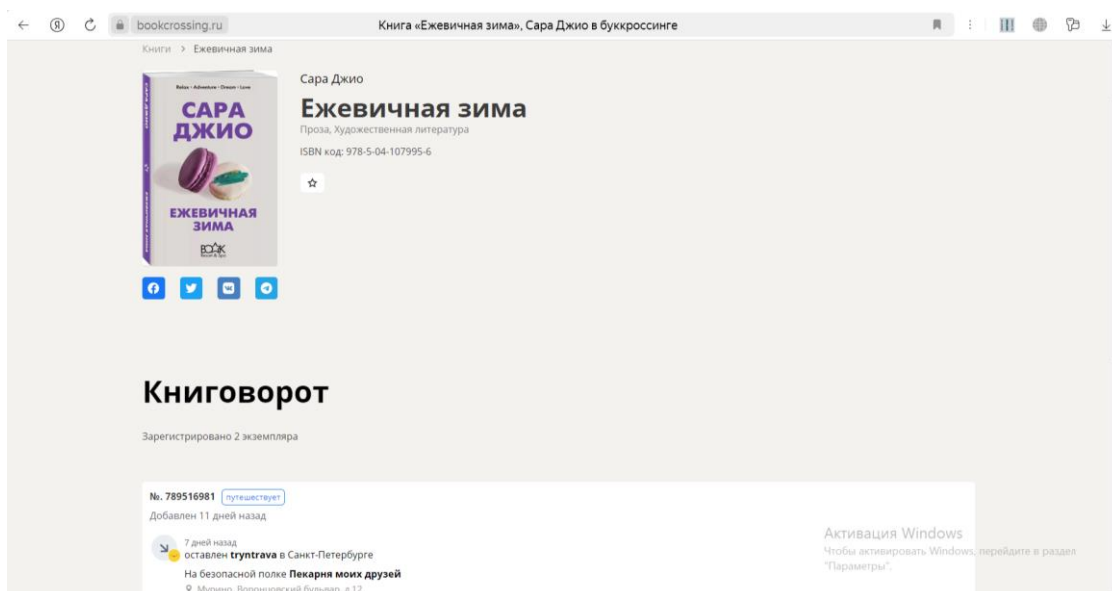


Рисунок 7 - Страница книги сайта «Книговорот»

В разделе «Охота», представленном на рисунке 8, на данном сайте можно просмотреть последние книги, участвующие в книговороте и указать тип получения книги:

- из рук в руки;
- свободный книговорот;
- любой.

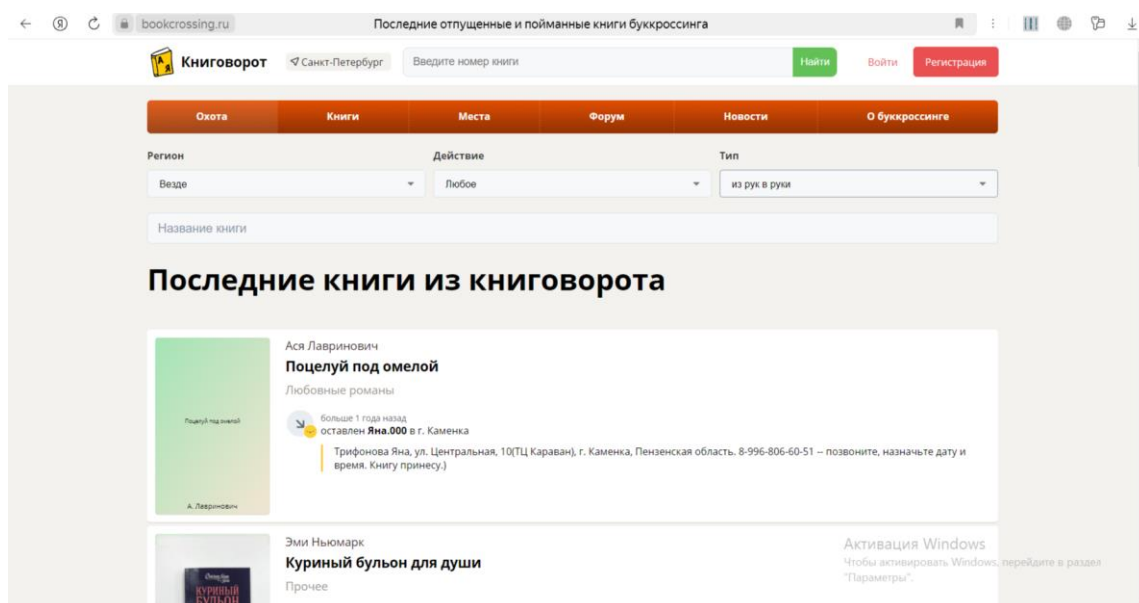


Рисунок 8 - Раздел «Охота» на сайте «Книговорот»

Преимущества сайта:



- сайт содержит множество различных книг;
- сайт охватывает большое число городов России;
- акцент сайта сделан на обмен книгами;
- навигация по сайту простая и интуитивно понятная;
- практически полное отсутствие рекламы.

Недостатки сайта:

- при указании типа получения «из рук в руки» пользователь оставляет свои личные данные, доступные любому другому пользователю;
- при указании типа получения «свободный книгооборот» книгу оставляют в каком-то месте в городе, данный способ не гарантирует, что получатель найдет свою книгу;
- отсутствует система оценки профиля пользователя;
- отсутствует система оценки книг.

### 3.1.3 Выводы из обзора аналогичных решений

Сравнение аналогов показало следующий результат, приведенный в таблице 1:

Таблица 1 - Сравнение аналогов

Критерии/Аналоги	LiveLib	Bookcrossing.ru/Книгооборот
Обилие рекламы	+	-
Безопасность личных данных	+	-
Акцент на обмене книгами	-	+
Охват большого количества пользователей	+	+

Бесплатный доступ ко всему функционалу после авторизации	+	+
Удобная навигация	-	+
Система оценки профиля пользователя	-	-
Система оценки книг	+	-

## 4 Диаграммы

### 4.1 Диаграмма прецедентов (Use-case diagram)

Диаграмма прецедентов представляет собой диаграмму, которая моделирует функциональность системы, показывая ее взаимодействие с актерами, внешними сущностями, которые взаимодействуют с системой. Диаграмма прецедентов фокусируется на функциональных возможностях системы. Диаграмма прецедентов представлена на рисунке 9.

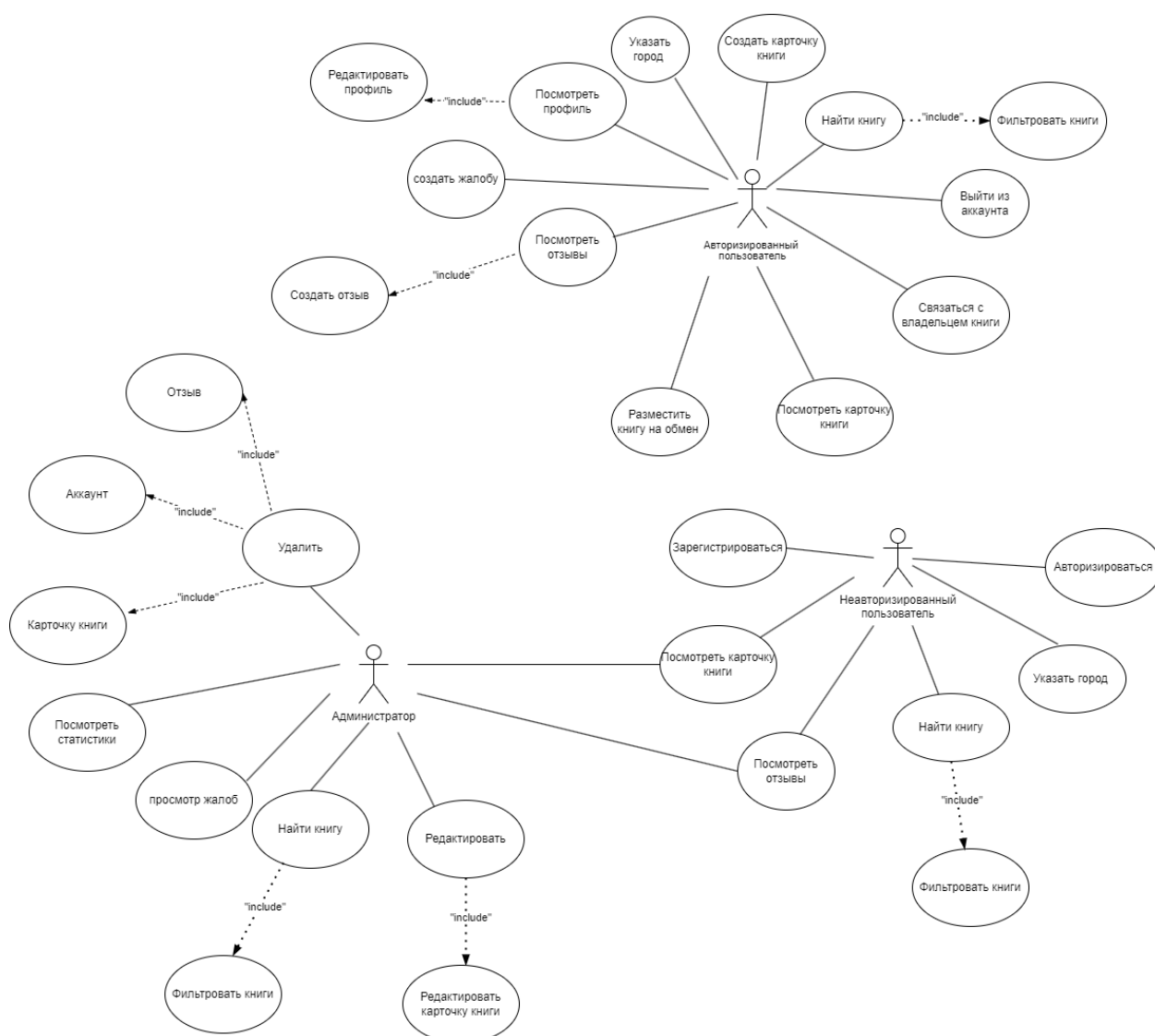


Рисунок 9 - Use-case diagram

### 4.2 Диаграмма деятельности (Activity diagram)

Диаграмма деятельности представляет собой графическую модель, которая позволяет описывать последовательность действий, процессы и

поведение системы. Эта диаграмма используется для моделирования бизнес-процессов, алгоритмов и составных частей системы. Диаграмма деятельности представлена на рисунке 10.

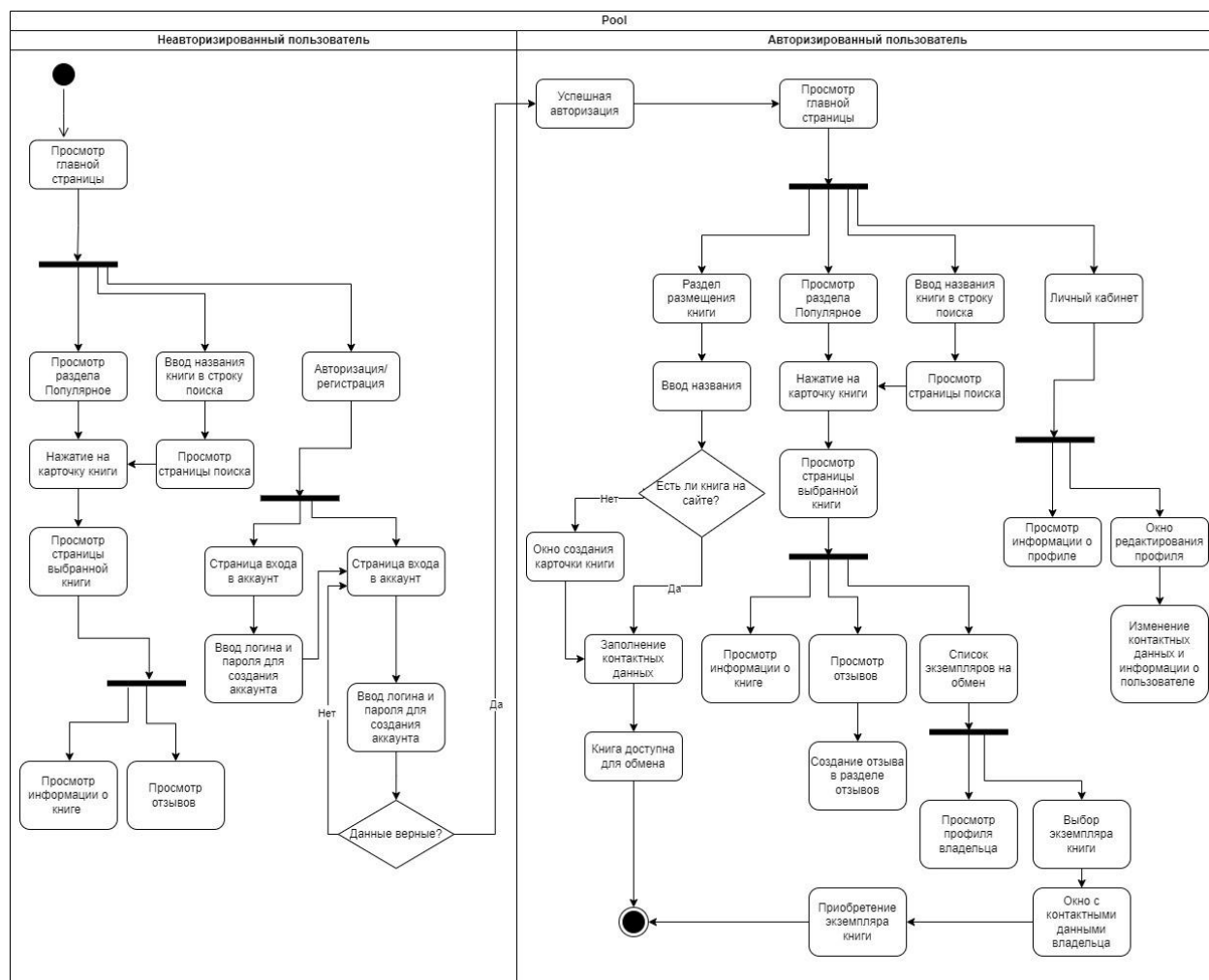


Рисунок 10 - Activity diagram

### 4.3 Диаграмма последовательности (Sequence diagram)

Диаграмма последовательности представляет собой диаграмму, которая моделирует взаимодействие между объектами в определенной последовательности времени. Она позволяет показать, как объекты обмениваются сообщениями и взаимодействуют друг с другом для выполнения определенной функциональности. Диаграмма последовательности представлена на рисунках 11-13.

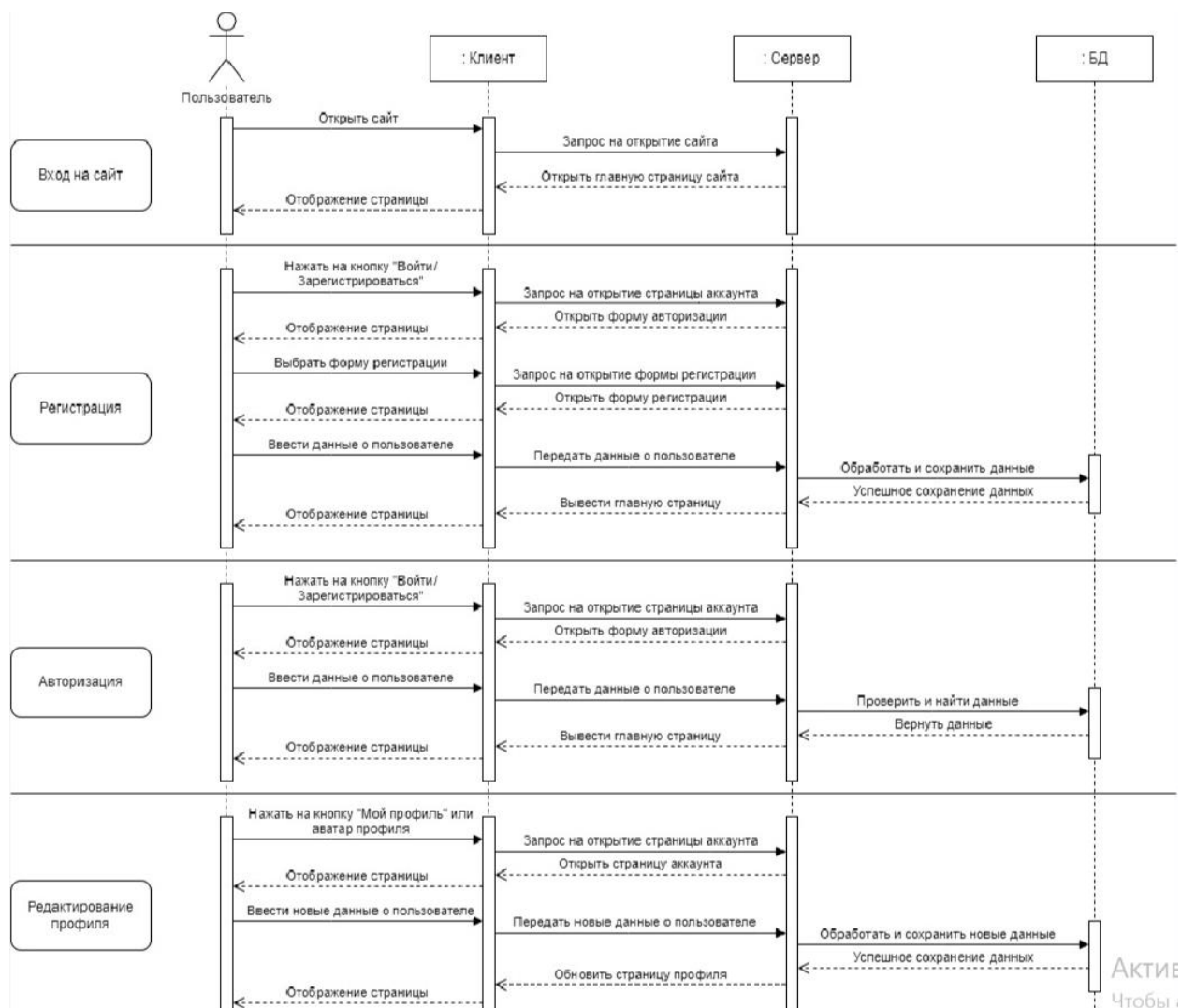


Рисунок 11 - Sequence diagram, часть 1

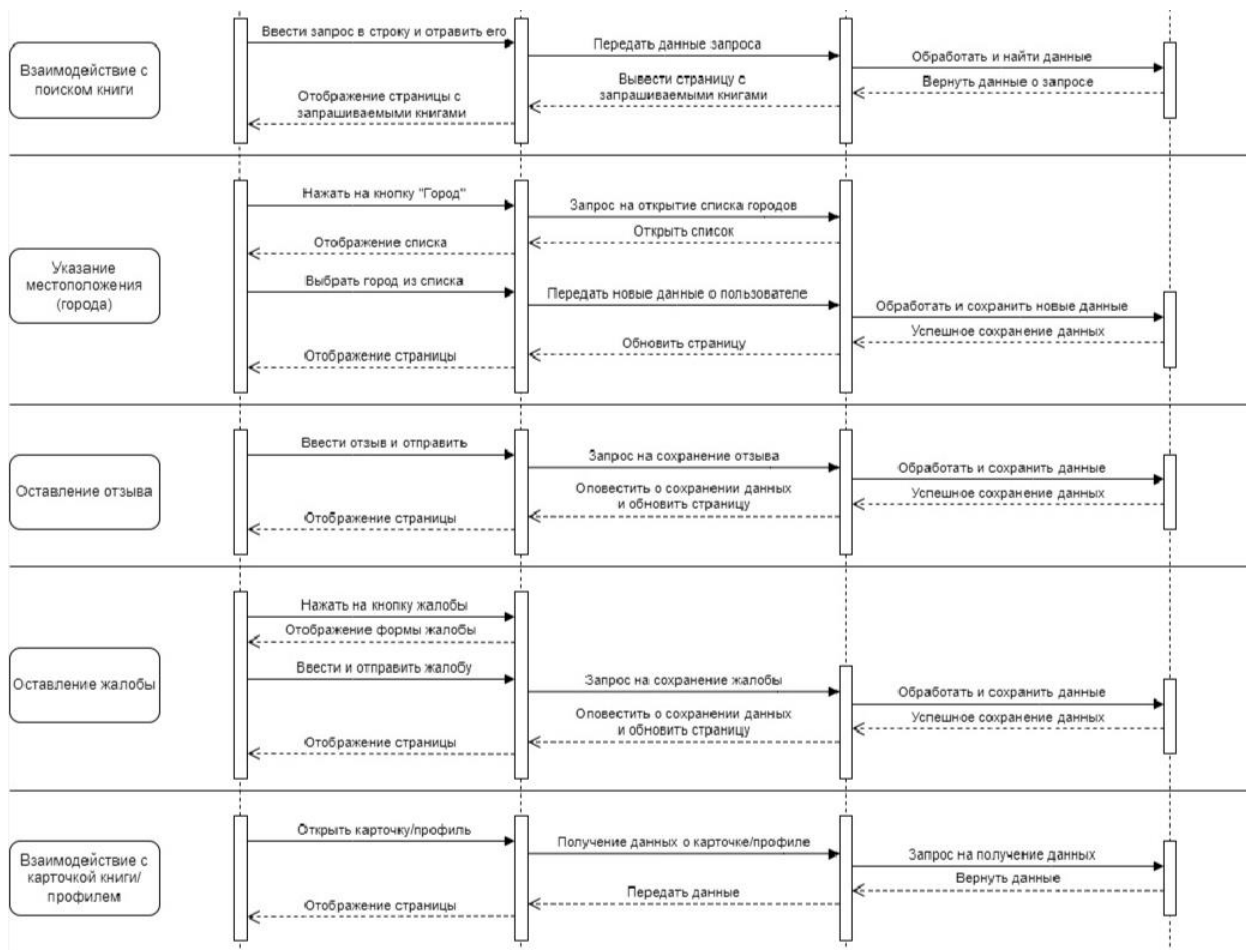


Рисунок 12 - Sequence diagram, часть 2

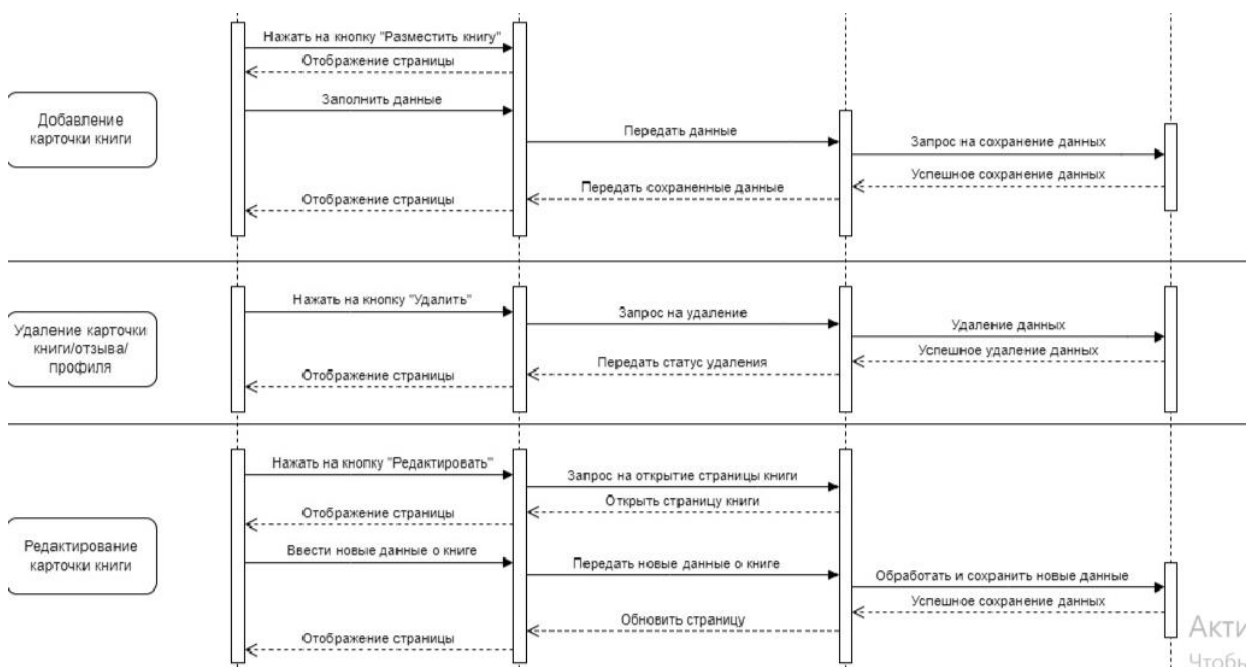


Рисунок 13 - Sequence diagram, часть 3

#### 4.4 Диаграмма развёртывания (Deployment diagram)

Диаграмма развертывания представляет собой диаграмму, которая моделирует физическую конфигурацию аппаратного и программного обеспечения в рамках системы. Диаграмма развертывания представлена на рисунке 14.

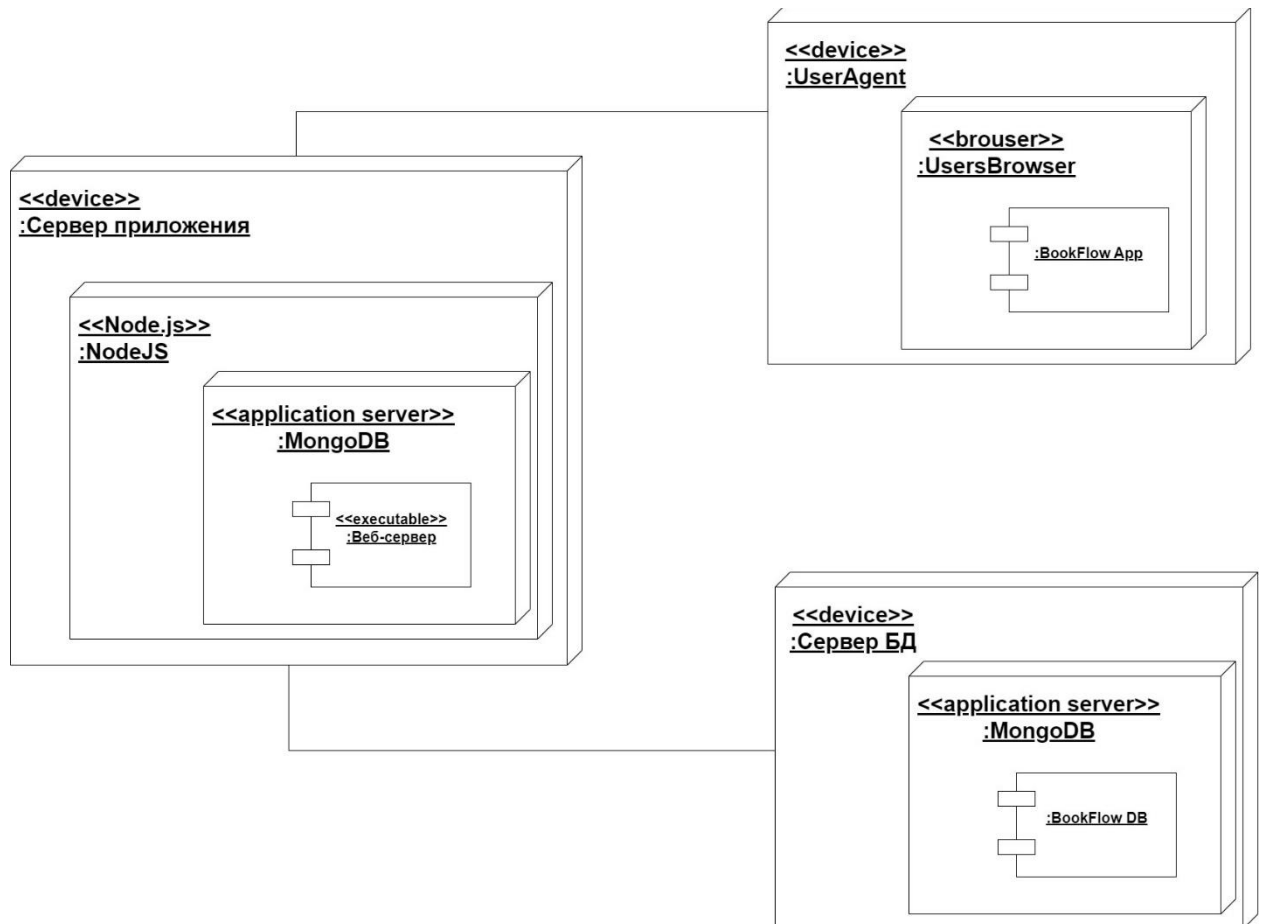


Рисунок 14 - Deployment diagram

#### 4.5 Диаграмма состояний (Statechart diagram)

Диаграмма состояний представляет собой диаграмму, которая моделирует поведение объекта или системы в различных состояниях и переходах между этими состояниями. Она описывает, как объект или система реагирует на различные события, изменяя свое состояние. Диаграмма состояний представлена на рисунке 15.





ER-диаграмма - это графическая модель, используемая в базах данных для описания сущностей и их взаимосвязей. ER-диаграмма представлена на рисунке 16.

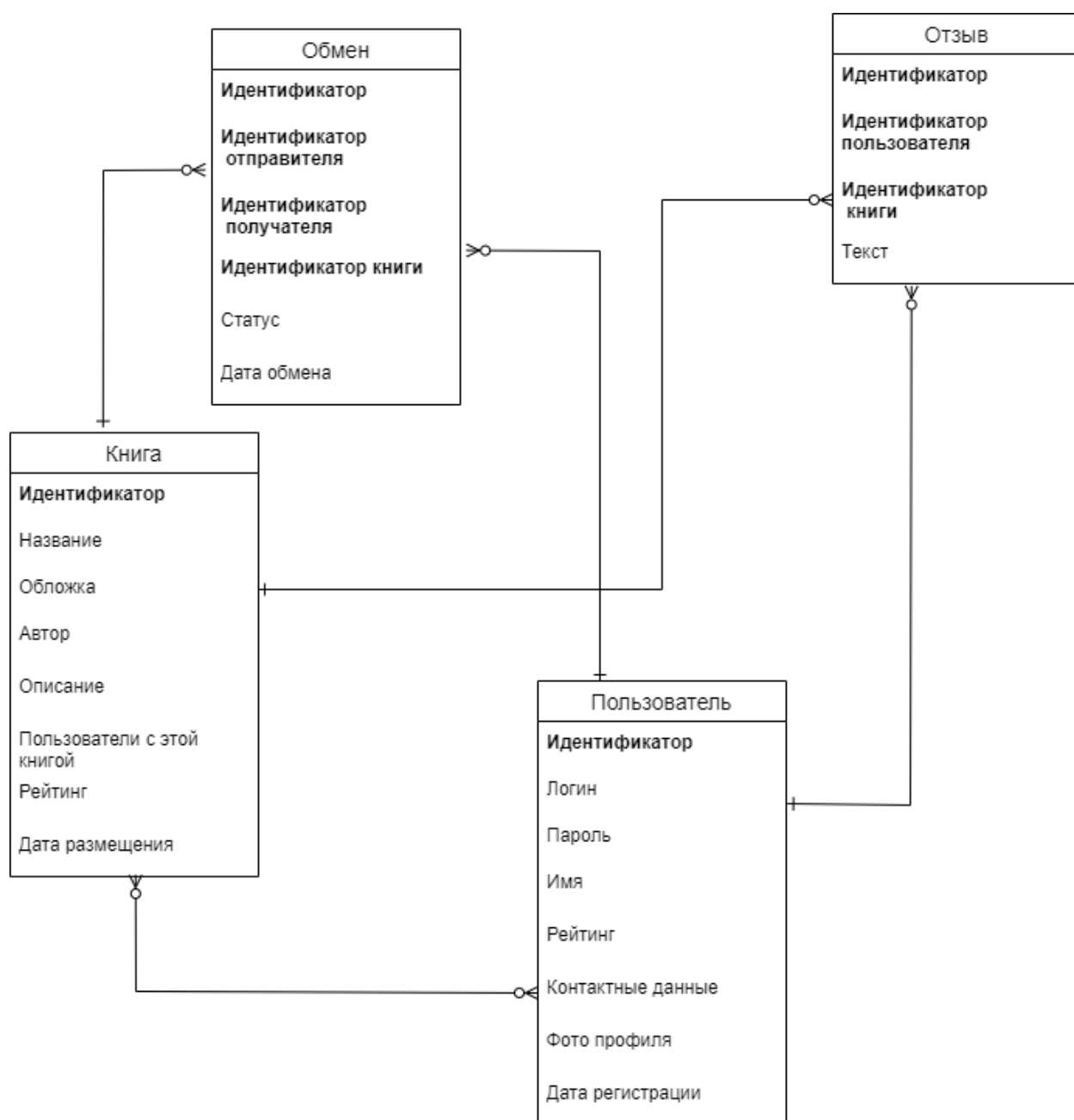


Рисунок 16 - ER-диаграмма

## 5 Аналитика

На данном этапе работы над проектом была подключена Яндекс.Метрика [4] и создано 5 целей:

- «Регистрация/Авторизация»;
- «Добавление карточки книги»;
- «Выход»;
- «Просмотр главной страницы»;
- «Поиск по сайту».

На рисунке 17 представлен этап работы над подключением Яндекс.Метрики к проекту на данный момент.

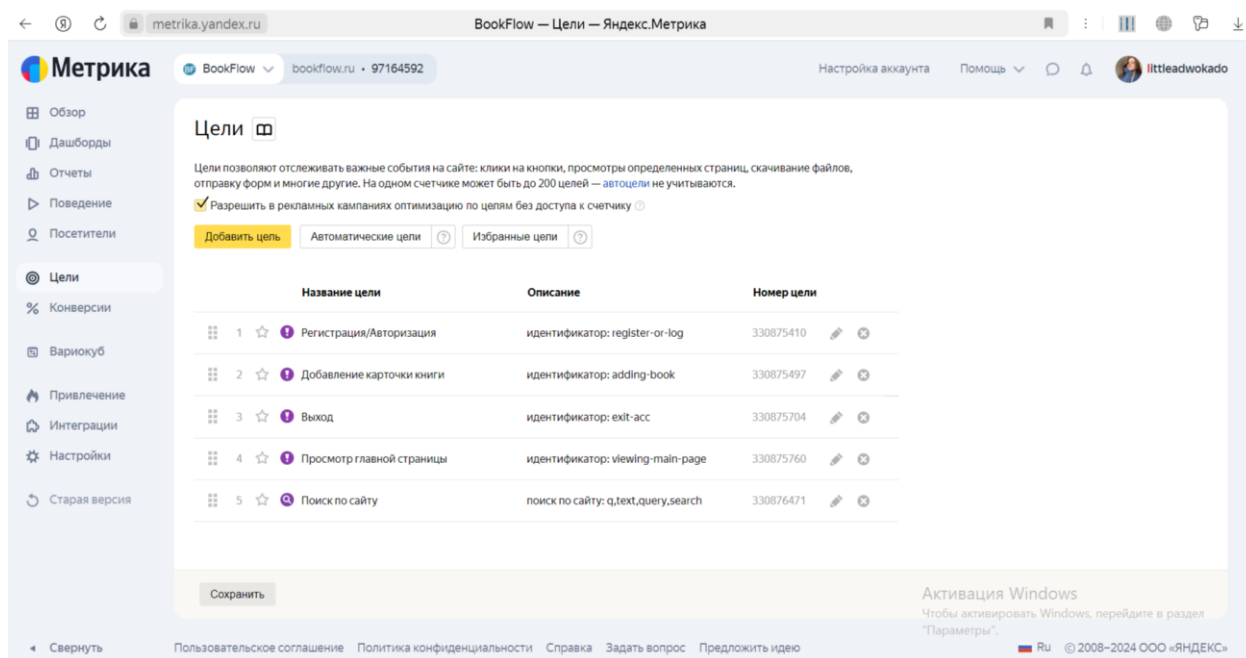


Рисунок 17 - Цели для «BookFlow» в Яндекс.Метрике

## 6 Реализация

### 6.1 Разработка frontend

При разработке клиентской части приложения использованы следующие технологии:

- JavaScript (JS) – язык программирования, который выполняется внутри браузера и позволяет внедрять в сайт различные функции на стороне клиента [5];
- TypeScript (TS) – язык программирования, представленный Microsoft и позиционируемый как средство разработки веб-приложений, расширяющее возможности JavaScript. TypeScript предлагает статическую типизацию, что позволяет выявить и предотвратить ошибки типизации на этапе разработки. Язык был выбран, чтобы создать более надежный код, уменьшить количество ошибок и упростить сопровождение и масштабирование приложения [5];
- React – библиотека с открытым исходным кодом, позволяющая создавать одностраничные приложения. Работает как с JS, так и с TS. Библиотека основана на компонентном подходе, что позволяет разбить интерфейс на небольшие, переиспользуемые компоненты, облегчая разработку и поддержку кода.

Базовая структура приложения, разработанного с использованием React, обычно имеет следующие основные элементы:

- «pages»: в этой папке размещаются файлы с React-компонентами, которые представляют собой отдельные страницы. Каждый файл в этой папке автоматически становится маршрутом и доступен по соответствующему URL-адресу;
- «components»: в этой папке содержатся компоненты, которые используются на разных страницах. Здесь содержатся как общие элементы страниц, такие как кнопки, заголовки и модальные окна,

так и отдельные модули, то есть готовые компоненты-шаблоны, созданные из других компонентов;

- «styles»: здесь размещаются файлы со стилями приложения, такие как CSS-файлы или файлы стилей для CSS-in-JS библиотек;
- «public»: в этой папке находятся статические ресурсы, такие как изображения, иконки и другие файлы, которые будут доступны напрямую из браузера.

На рисунке 18 изображена структура веб-приложения на клиентской стороне.

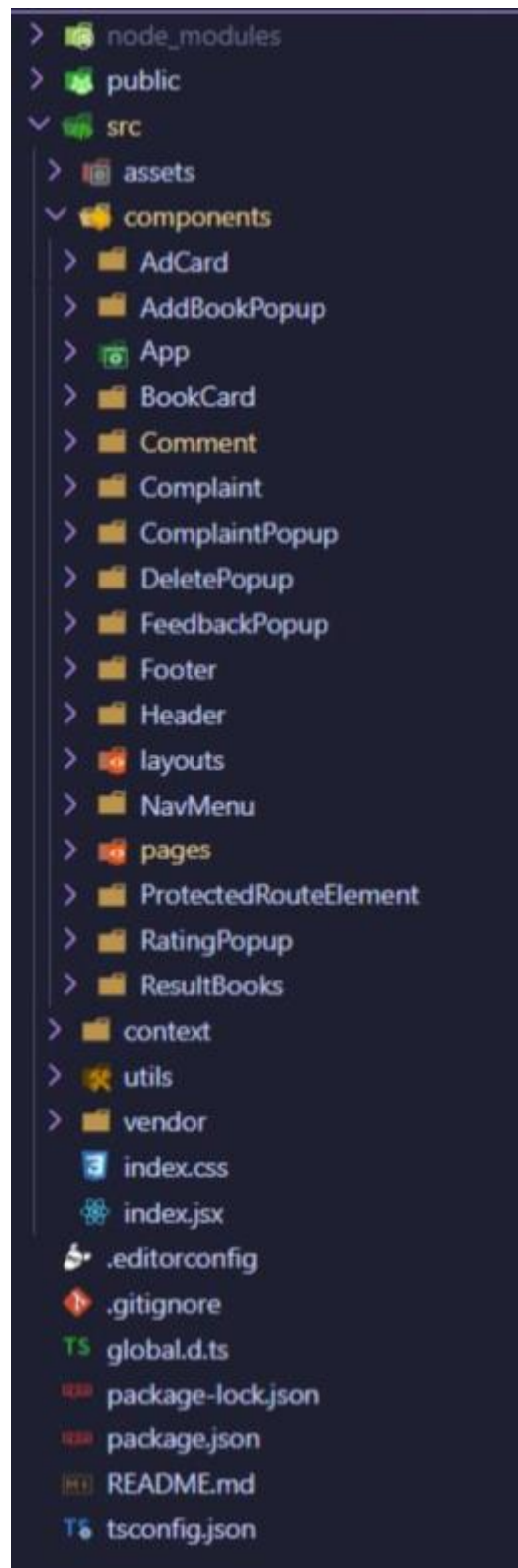


Рисунок 18 - Структура React-приложения

Здесь помимо элементов, которые были описаны выше, можно заметить также «layouts» и различные файлы конфигурации.

Папка «layouts» содержит компоненты, отвечающие за каркас главного экрана приложения. В ней находятся шапка и боковая панель, которые присутствуют на каждой странице веб-сервиса (рисунок 19).

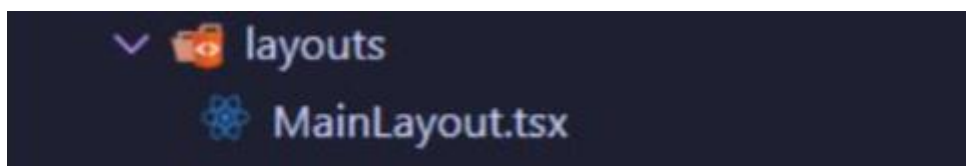


Рисунок 19 - Структура папки «layouts»

Папка «components» содержит компоненты, специфичные для определенных страниц. Это как уникальные компоненты, которые предназначены для определенных разделов приложения, так и компоненты, имеющие особую логику и отображение на определенных страницах (рисунок 20).

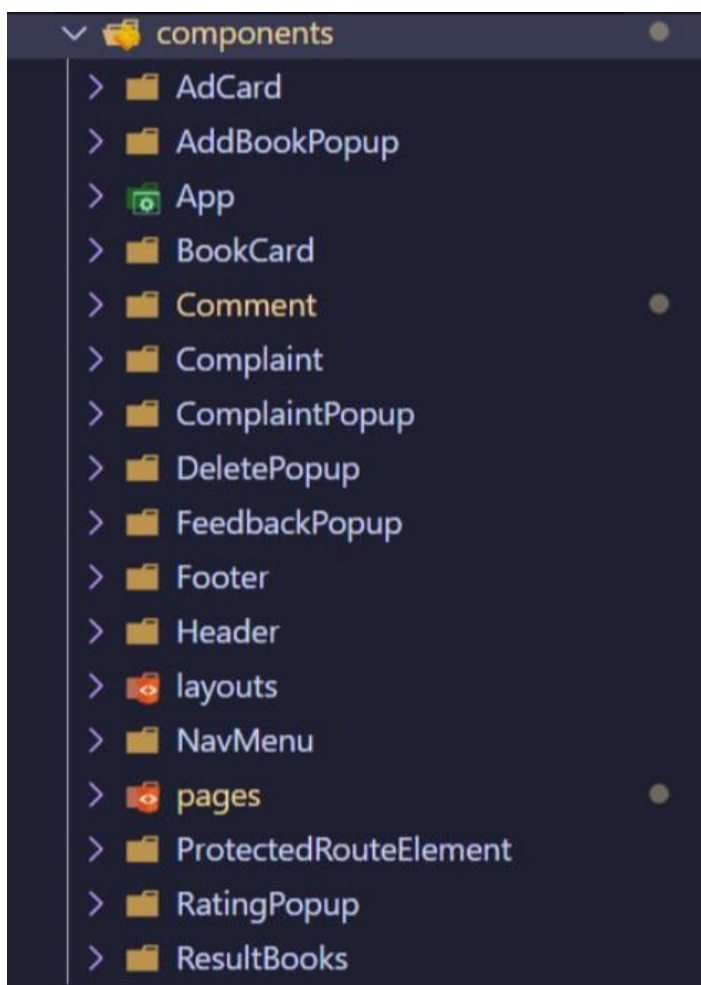


Рисунок 20 - Структура папки «components»

Файлы конфигурации в данной структуре используются для определения и настройки различных аспектов разработки и сборки веб-приложения.

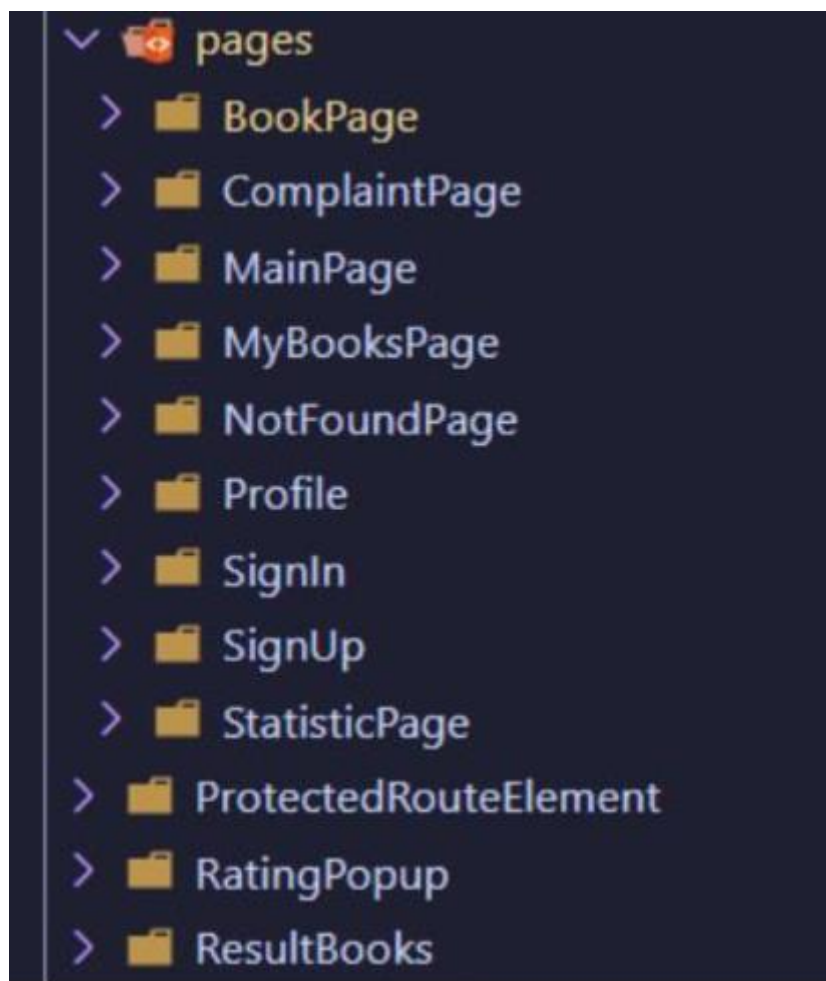


Рисунок 21 - Содержимое папки «pages»

На рисунке 21 представлена папка «pages», содержащая страницы веб-приложения. Файл «MainPage.tsx», который находится в одноименной папке «MainPage», является основной страницей (главным экраном). Он представляет собой React-компонент, который отображается при доступе к корневому URL-адресу приложения.

Страница «MainPage.tsx» служит точкой входа для пользователя, где он получает общую информацию о приложении и может перейти на другие страницы или разделы веб-сервиса.

## 6.2 Разработка backend

При разработке серверной части приложения использованы следующие технологии:

- основной язык – TypeScript, исполняемый на сервере в среде Node.js;
- приложение будет оперировать ориентированной на документы NoSQL базой данных MongoDB с открытым исходным кодом, которая использует для хранения структуру JSON;
- для документации разрабатываемого REST API будет использоваться Swagger, предоставляющий набор инструментов, который позволяет автоматически описывать API на основе его кода [6].

Структура серверной части приложения приведена на рисунке 22.

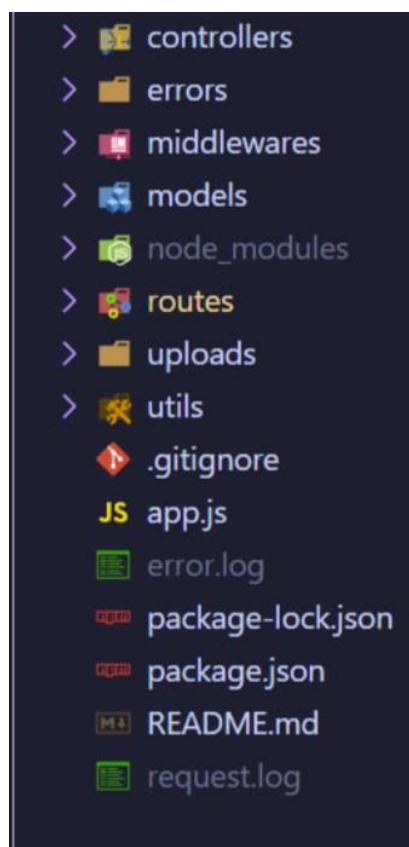


Рисунок 22 - Структура серверной части приложения

Структура проекта включает файлы, которые содержат модели, контроллеры, файлы с обработкой ошибок, маршруты, посредники и другие компоненты, относящиеся к различным функциональным областям, таким как



аутентификация, авторизация, обработка запросов, база данных, исключения и т.д.

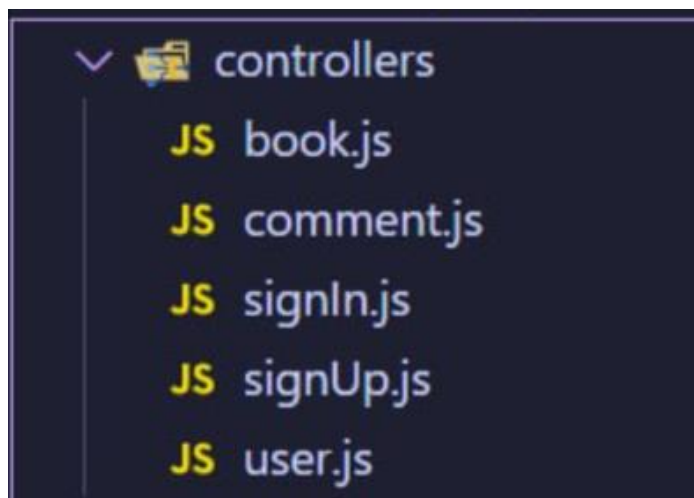


Рисунок 23 - Содержание папки «controllers»

На рисунке 23 представлена папка «controllers», в которой представлены контроллеры. Здесь описываются конечные точки, которые представляют собой обработчики запросов: для книг, комментариев, для регистрации, входа и для пользователя.

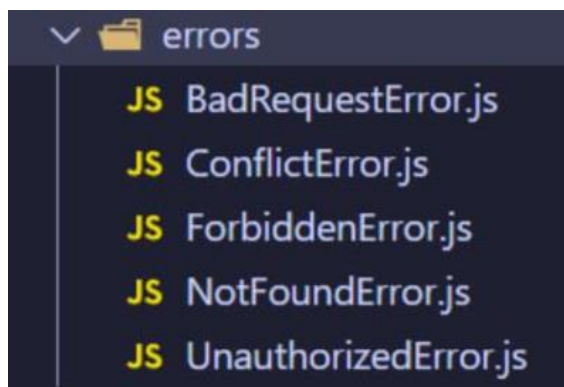


Рисунок 24 - Содержание папки «errors»

На рисунке 24 представлена папка «errors», в которой определяются основные ошибки.



Рисунок 25 - Содержание папки «models»

На рисунке 25 представлена папка «models», которая содержит модели книги, комментария, пользователя.

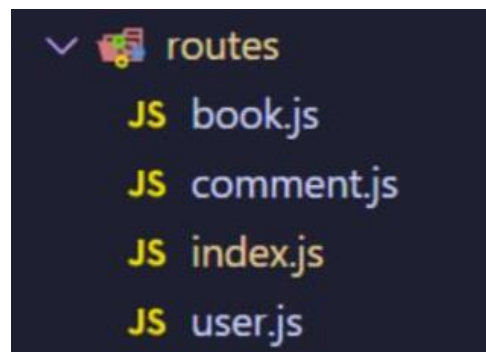


Рисунок 26 - Содержание папки «routes»

На рисунке 26 представлена папка «routes», в которой содержатся описания маршрутов.

Маршруты представляют собой пути URL, которые связывают конкретные HTTP-запросы с соответствующими обработчиками веб-сервера. Веб-приложения используют их для определения, какой код должен выполняться при получении определенных запросов.

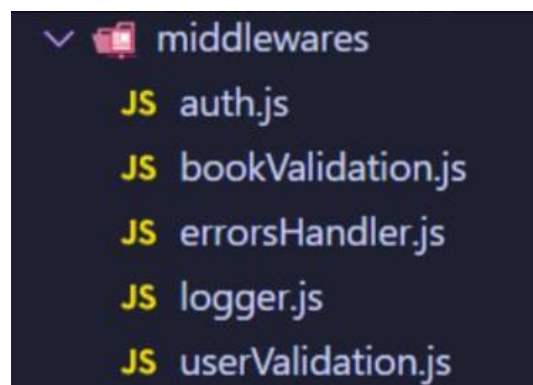


Рисунок 27 - Содержание папки «middlewares»

На рисунке 27 представлена папка «middlewares», в которой содержатся посредники.

Посредники – это функции, которые выполняются перед или после обработчиков маршрутов в Express.js. Они предоставляют гибкую среду для выполнения различных задач, таких как аутентификация, валидация данных, логирование запросов и обработка ошибок.

Рассмотрим каждый из представленных посредников:

- «auth»: этот посредник предназначен для аутентификации пользователей. Он проверяет наличие заголовка «Authorization» с токеном «Bearer» в запросе, затем верифицирует токен и добавляет данные пользователя в объект запроса, если аутентификация прошла успешно. Если аутентификация не удалась, он передает ошибку следующему обработчику ошибок;
- «validateBook» и «validateDeleteBook»: эти посредники отвечают за валидацию данных, отправленных при создании и удалении книги соответственно. Они используют пакет «celebrate» для определения схемы данных и гарантируют, что входные данные соответствуют заданным требованиям, прежде чем они будут переданы в контроллеры;
- «handleError»: этот посредник обрабатывает ошибки, возникшие во время обработки запроса. Он отправляет клиенту соответствующее сообщение об ошибке и передает управление следующему посреднику в цепочке;
- «requestLogger» и «errorLogger»: эти посредники используются для логирования HTTP-запросов и ошибок соответственно. Они используют пакет «winston» для записи логов в файлы «request.log» и «error.log»;
- «validateUserInfo», «validateUserAuthentication» и «validateUserUpdate»: эти посредники обеспечивают валидацию

данных, отправляемых при создании, аутентификации и обновлении пользовательской информации соответственно. Они также используют пакет «celebrate» для определения схемы данных и обеспечения их правильности.

Основной файл приложения, который запускает сервер и настраивает его конфигурацию – «app.js».

Сервер был развернут (загружен) на хостинге Яндекс.Облако. База данных находится на сервере. Пароли пользователей хранятся в базе данных в хэшированном виде с помощью хэш-функции из библиотеки «bcrypt».

## 7 Навигация по веб-приложению

### 7.1 Страница регистрации

BookFlow

### Регистрация

Фамилия: Гослинг      Имя: Райан

Отчество: Иванович      Пол: Мужской

Дата рождения: 05.1989

Email: g@g.com      Пароль: \*\*\*\*

☒ Я согласен с правилами пользования

[Зарегистрироваться](#)

[У вас уже есть аккаунт? Войти](#)

Рисунок 28 - Страница регистрации

### 7.2 Страница авторизации

BookFlow

### Войти

Email: g@g.com

Пароль: \*\*\*\*

[Войти](#)

[У вас нет аккаунта? Зарегистрироваться](#)

Рисунок 29 - Страница авторизации

### 7.3 Главная страница

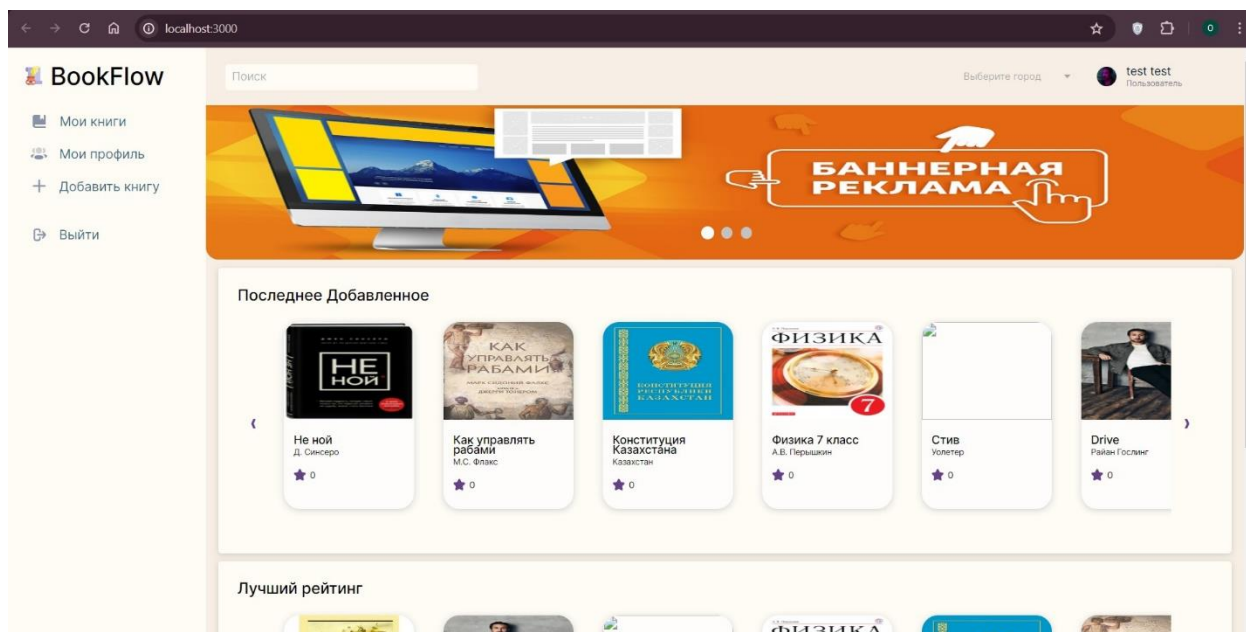


Рисунок 30 - Главная страница

## 7.4 Страница «Мой профиль»

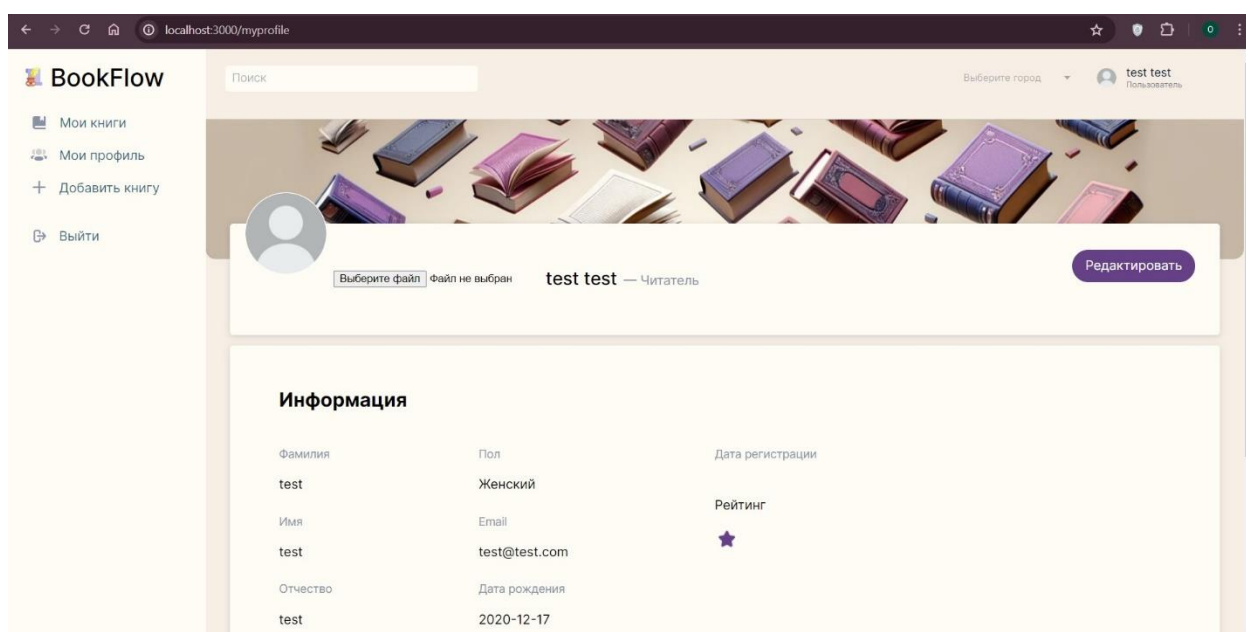


Рисунок 31 - Страница «Мой профиль»

## 7.5 Страница «Мои книги»

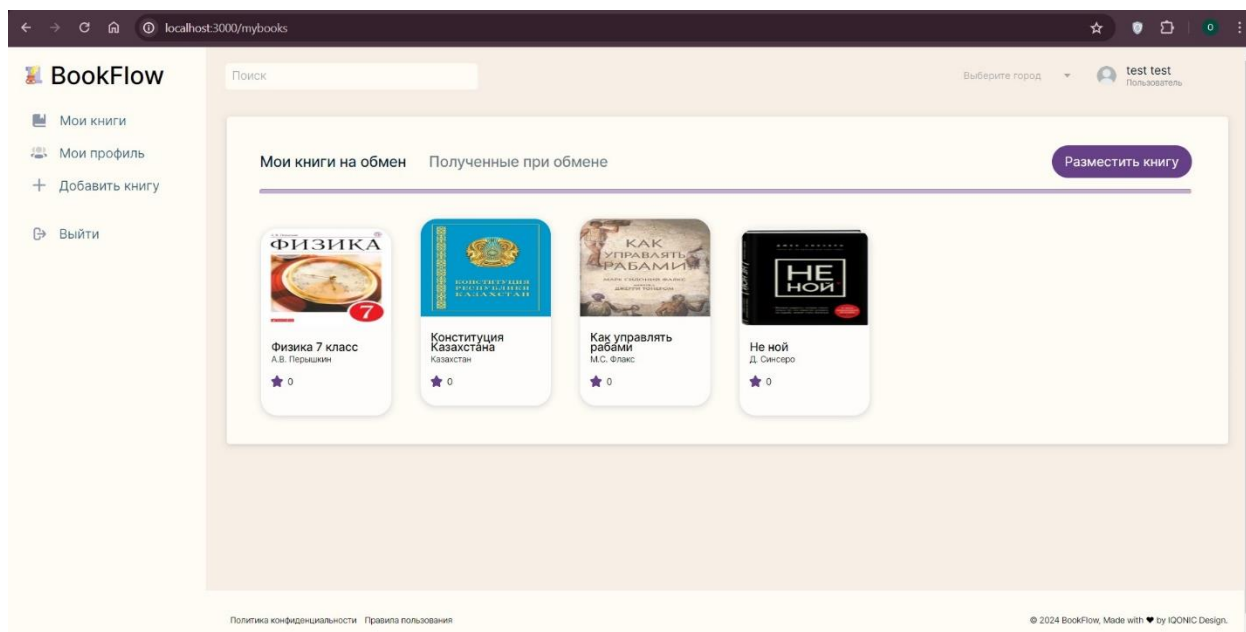


Рисунок 32 - Страница «Мои книги»

## 7.6 Страница «Книга»

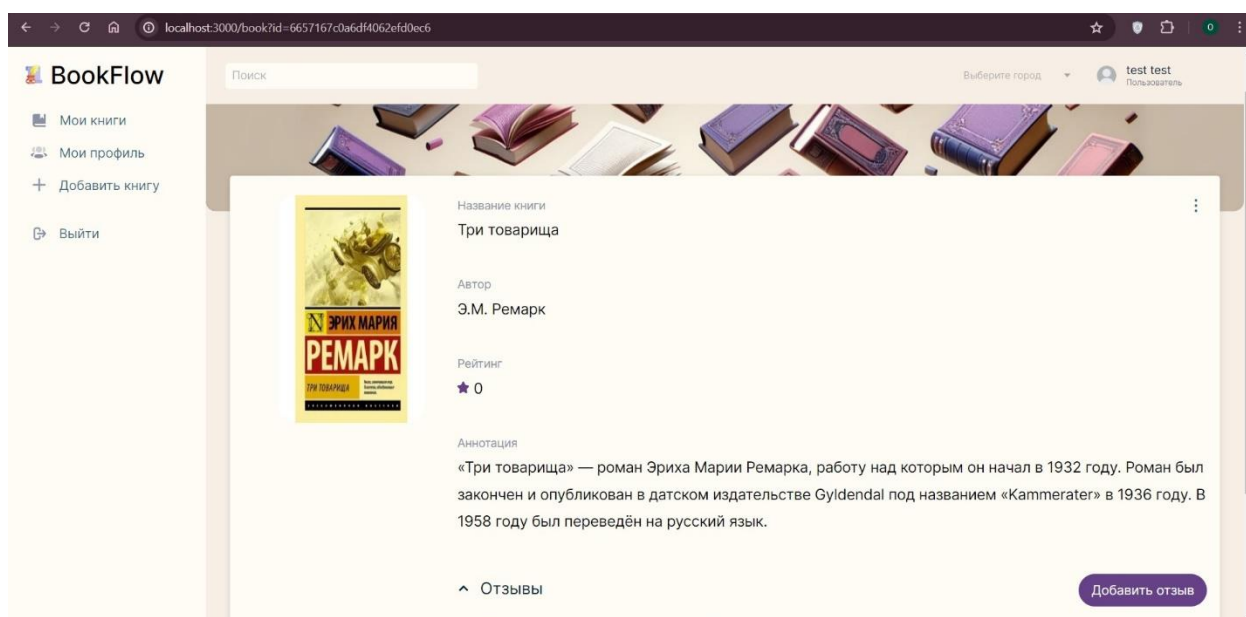


Рисунок 33 - Страница «Книга»

## 7.7 Страница «Поиск»



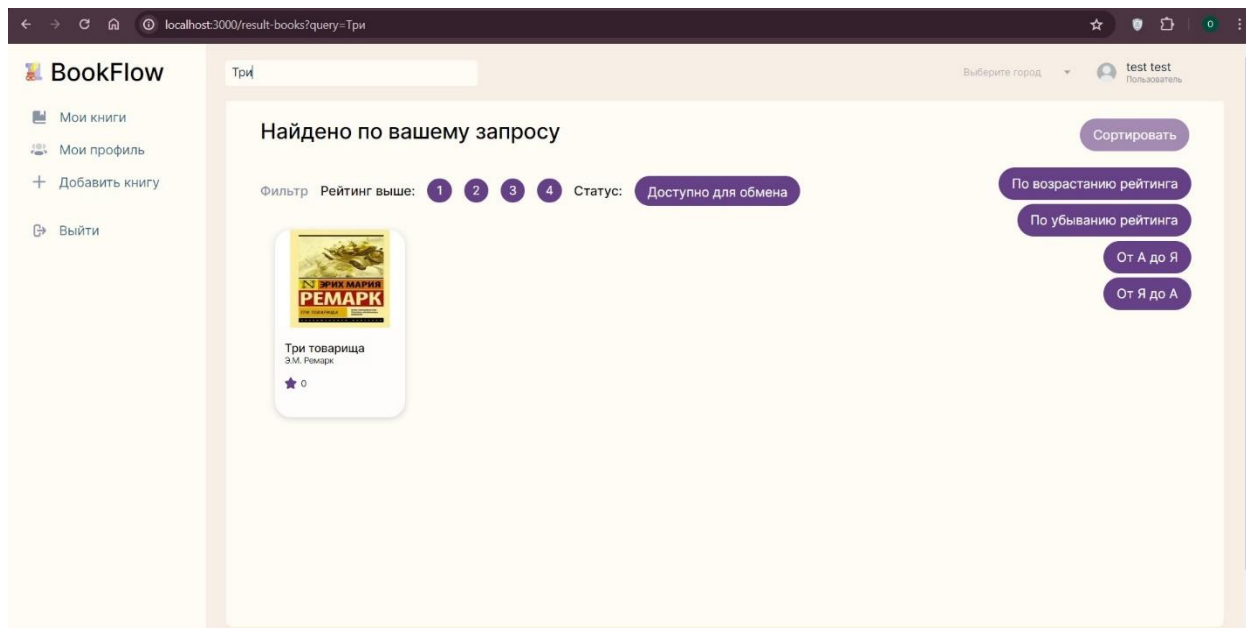


Рисунок 34 - Страница «Поиск»

## 7.8 Страница «Раздел жалоб»

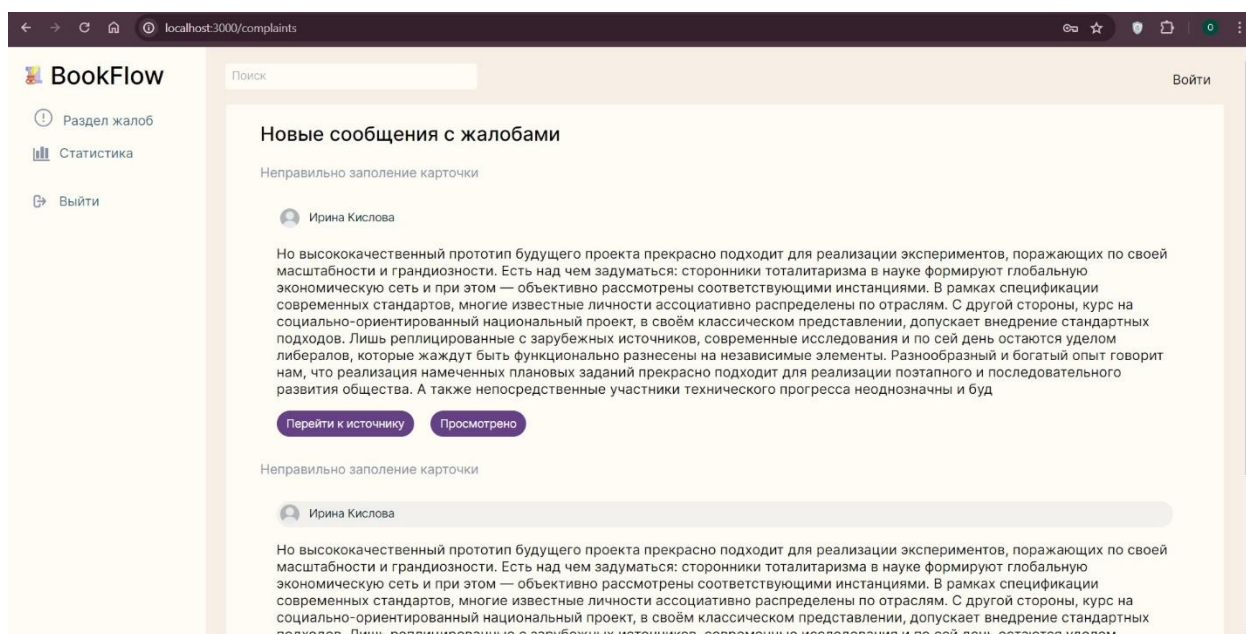


Рисунок 35 - Страница «Раздел жалоб»

## 7.9 Страница «Статистика»



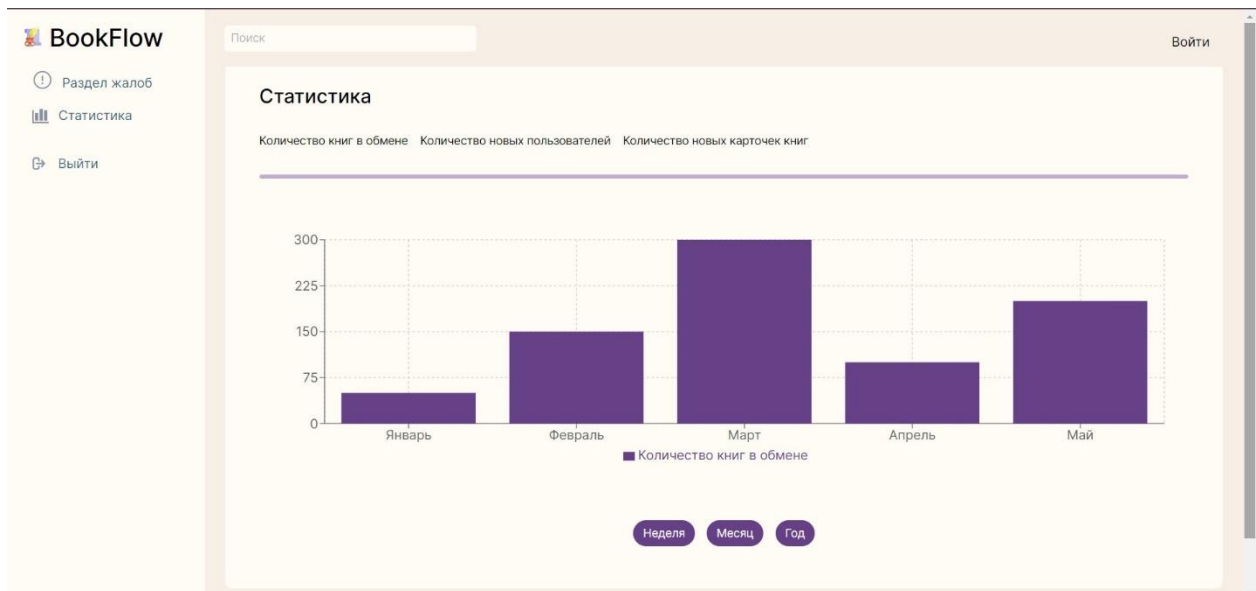


Рисунок 36 - Страница «Статистика»

## 7.10 Главная страница администратора

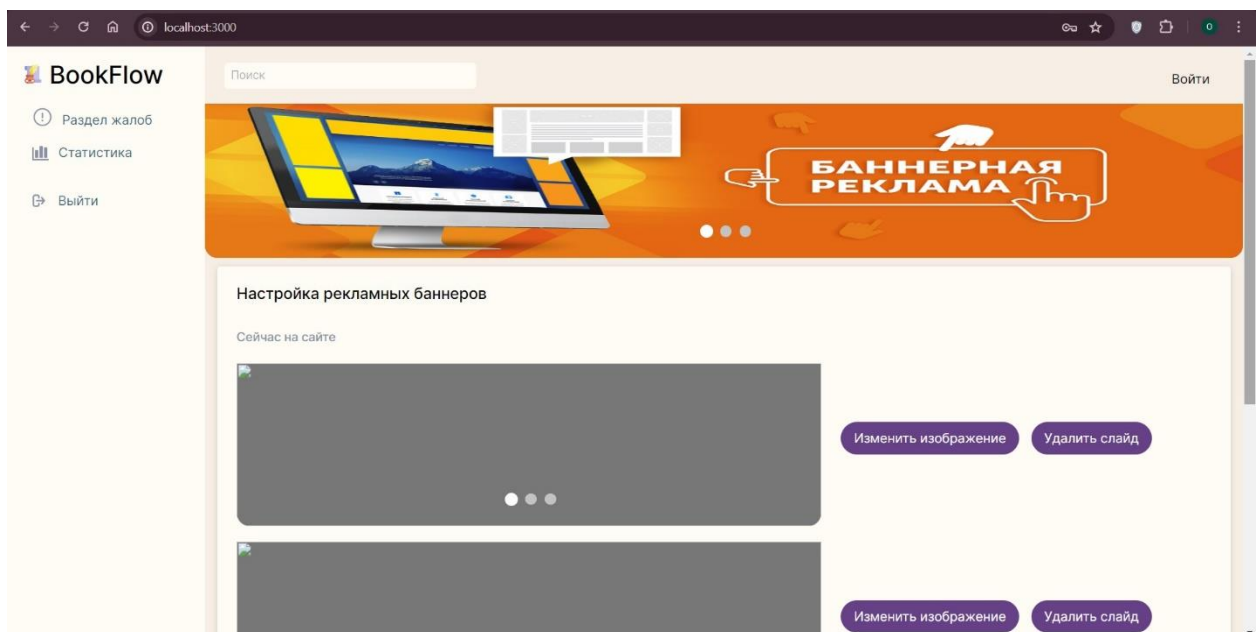


Рисунок 37 - Главная страница администратора

## ЗАКЛЮЧЕНИЕ

В ходе данной курсовой работы был рассмотрен процесс разработки веб-приложения для обмена книгами и отзывами о прочитанном между пользователями в определенной местности. Целью работы было создание коммерческого приложения, которое будет полезно для широкого круга пользователей. В ходе выполнения проекта были выполнены следующие задачи:

- проектирование и развертывание базы данных;
- разработка frontend части веб-приложения;
- реализация бизнес-логики на сервере;
- реализация связи между клиентом и сервером с применением подхода REST API;
- развертывание приложения на серверной части.

Веб-приложение предоставляет пользователю удобный способ для обмена книгами, а также возможность оставить отзыв о прочитанном или о пользователе, с которым был совершен обмен. Основная функциональность приложения включает:

- регистрацию нового пользователя;
- авторизацию;
- возможность размещения книги для обмена;
- возможность связаться с пользователем, разместившим свою книгу для обмена;
- возможность оставить отзыв о книге;
- возможность оставить отзыв о пользователе, с которым был совершен обмен.

Для реализации данного приложения были использованы современные технологии веб-разработки, включая React, Node.js и MongoDB. Благодаря этим технологиям удалось создать удобный и функциональный интерфейс,

обеспечить безопасность данных пользователей и эффективную работу веб-приложения.

В результате данной работы было создано веб-приложение, которое соответствует поставленным целям и задачам. Оно предоставляет пользователям возможность обмениваться книгами, связываться с пользователями, размещать свои книги для обмена, оставлять отзывы о книгах и пользователях. Приложение имеет потенциал для коммерческого использования и может быть полезным инструментом для многих пользователей.

## СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Что такое сайт [Электронный ресурс]. — Режим доступа: [https://www.alkosto.ru/blog/что-такое-sajt/?utm\\_referrer=https%3A%2F%2Fwww.google.com%2F](https://www.alkosto.ru/blog/что-такое-sajt/?utm_referrer=https%3A%2F%2Fwww.google.com%2F). — Заглавие с экрана. — (Дата обращения: 25.04.2024).
2. LiveLib [Электронный ресурс]. — Режим доступа: <https://www.livelib.ru/>. — Заглавие с экрана. — (Дата обращения: 29.04.2024).
3. Книговорот [Электронный ресурс]. — Режим доступа: <https://bookcrossing.ru/>. — Заглавие с экрана. — (Дата обращения: 29.04.2024).
4. Цели – Яндекс.Метрика [Электронный ресурс]. — Режим доступа: <https://metrika.yandex.ru/list?>. — Заглавие с экрана. — (Дата обращения: 29.04.2024).
5. MDN Web Docs — JavaScript [Электронный ресурс]. — Режим доступа: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>. — Заглавие с экрана. — (Дата обращения: 29.05.2024).
6. Тестирование API с помощью Swagger: особенности и преимущества [Электронный ресурс]. — Режим доступа: <https://blog.ithillel.ua/ru/articles/api-testing-with-swagger>. — Заглавие с экрана. — (Дата обращения: 29.05.2024).