|  |  |
| --- | --- |
|  | Министерство науки и высшего образования Российской Федерации  Федеральное государственное бюджетное образовательное учреждение высшего образования  «Новгородский государственный университет имени Ярослава Мудрого»  ИНСТИТУТ ЭЛЕКТРОННЫХ И ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ  ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ |

РАЗРАБОТКА СЕРВИСА ДЛЯ ЧТЕНИЯ КНИГ И

ПРОСЛУШИВАНИЯ АУДИОКНИГ

Курсовой проект

ПМ.08 ПРОЕКТИРОВАНИЕ ДИЗАЙНА ИНФОРМАЦИОННЫХ СИСТЕМ

Пояснительная записка курсового проекта по специальности

09.02.07 Информационные системы и программирование

|  |
| --- |
| Студент группы 2994  \_\_\_\_\_\_\_\_/ Н.И. Нефёдова /  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20\_\_ г |
| Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Руководитель  \_\_\_\_\_\_\_\_\_/ В. Р. Ильин/  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20\_\_ г |

Содержание

[Введение 3](#_Toc164940188)

[1. Общая часть 4](#_Toc164940189)

[1.1 Постановка задачи 4](#_Toc164940190)

[1.2 Обоснование проектных решений 7](#_Toc164940191)

[1.3 Обзор и анализ существующих программных систем 9](#_Toc164940192)

[2. Практическая составляющая работы 11](#_Toc164940193)

[2.1 Анализ задачи 11](#_Toc164940194)

[2.2 Реализация 11](#_Toc164940195)

[2.3 Тестирование 14](#_Toc164940198)

[2.4 Руководство оператора 14](#_Toc164940199)

[Заключение 17](#_Toc164940200)

[Список литературы 18](#_Toc164940201)

[Приложение А 19](#_Toc164940202)

[Приложение Б 20](#_Toc164940203)

[Приложение Г 23](#_Toc164940204)

[Приложение Д 29](#_Toc164940205)

# Введение

В соответствие с заданием на курсовое проектирование требуется разработать веб-приложение для чтения книг и прослушивания аудиокниг.

Книги всегда хранили в себе нужную и полезную для человека информацию. Люди до сих пор ищут в них советы, инструкции, рецепты и многое другое.

Часто на интернет ресурсах не найти необходимую информацию и даже если она есть, довольно трудно проверить её правдивость. В таком случае, люди прибегают к помощи книг, но печатные издания бывают дорогостоящими, затрачивается много времени на их поиск и получение, поэтому появились онлайн книги. Онлайн книги легко найти в приложениях и на сайтах, всё, что нужно для поиска информации – стабильное интернет-соединение, а если нет времени на чтение, есть возможность прослушать книгу в электронном формате. Поэтому платформы для чтения книг и прослушивания аудиокниг очень важны в наше время, что подтверждает актуальность курсового проекта.

Цель курсового проекта – разработать веб-приложение для чтения книг и прослушивания аудиокниг.

Задачи курсового проекта:

* знакомство с предметной областью исследуемой задачи;
* разработка IDEF1X диаграммы;
* разработка диаграммы классов;
* проектирование функциональных элементов сайта;
* создание интерфейсов пользователя в приложении Figma;
* разработка клиентской части приложения;
* разработка серверной части приложения;
* тестирование приложения;

# Общая часть

# Постановка задачи

В соответствии с заданием требуется разработать веб-приложение для чтения книг и прослушивания аудиокниг. Для успешного выполнения этой задачи потребуется разработать следующий функционал:

* + Регистрация
  + Авторизация
  + Получение подписки
  + Изменение личных данных в профиле
  + Поиск книг и аудиокниг по автору и названию
  + Чтение отрывка книги в размере 3 страниц
  + Чтение книги целиком
  + Прослушивание аудиокниги

1.1.1 Характеристики бизнес-процессов

Бизнес-процесс – это многократно повторяющаяся последовательность действий, которая направлена на создание продукта, имеющего ценность для заказчика.

В приложении А.1 можно увидеть IDEF0 модель бизнес-процесса чтения книг.

* + 1. Технико-математическое описание задачи

При разработке приложения были использованы:

HTML – язык разметки для описания структуры веб-страниц. Страницы, созданные с его помощью, могут быть просмотрены только при помощи браузера. Элементы HTML являются строительными блоками HTML страниц. С помощью HTML разные конструкции, изображения и другие объекты, такие как [интерактивная веб-форма](https://ru.wikipedia.org/wiki/%D0%A4%D0%BE%D1%80%D0%BC%D0%B0_(HTML)), могут быть встроены в отображаемую страницу. HTML предоставляет средства для создания заголовков, абзацев, списков, ссылок, цитат и других элементов.

TailwindCSS – это постепенно набирающий популярность CSS-фреймворк, позволяющий вносить изменения в оформление сайтов и приложений, не покидая HTML-разметку (причем как в соответствующих файлах, так и в компонентах типа React или Svelte) и не используя тег <style>. Идея заключается в том, чтобы прописывать стили напрямую в директиву class, а не под селекторами в CSS-файле.

JS - это мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией спецификации ECMAScript. Он позволяет создать динамически обновляемый контент, управляет мультимедиа, анимирует изображения.

Vue.js - это прогрессивный фреймворк для создания пользовательских интерфейсов. В отличие от фреймворков-монолитов, Vue создан пригодным для постепенного внедрения. Его ядро в первую очередь решает задачи уровня представления, что упрощает интеграцию с другими библиотеками и существующими проектами. С другой стороны, Vue полностью подходит и для создания сложных одностраничных приложений, если использовать его совместно с [современными инструментами](https://ru.vuejs.org/v2/guide/single-file-components.html) и [дополнительными библиотеками](https://github.com/vuejs/awesome-vue#components--libraries).

PHP - это интерпретируемый скриптовый язык программирования общего назначения с открытым исходным кодом. PHP специально сконструирован для веб-разработок и его код может внедряться непосредственно в HTML.

Laravel - это бесплатный PHP-фреймворк с открытым исходным кодом. Laravel разработали в качестве помощника при создании сложных веб-ресурсов и приложений. С его помощью специалисты упрощают процесс аутентификации, а также работу с БД, кэширование, сессии, структуру приложения, маршрутизацию и другие не менее важные процессы.

* + 1. Требования к программе

1. Требования к применению

Веб-приложение позволяет новым пользователям заводить личные аккаунты, оформлять один из трёх видов подписки, читать книги и слушать аудиокниги.

1. Требования к дизайну

* Дизайн сайта должен придерживаться идее минимализма, не загружать страницы нефункциональными элементами
* Основными цветами сайта являются светло-коричневый и зелёный, акцентный цвет – тёмно-коричневый
* Шрифт на сайте должен быть в стиле Open Sans
* Обычный текст на странице не должен превышать 24 пунктов, подзаголовки – 36 пунктов, а заголовки – 50 пунктов

1. Требования к интерфейсу

* Интерфейс должен быть понятен для пользователя, придерживаться правилу не более 3-х кликов для поиска нужной информации
* Кнопки должны реагировать на наведение, например, менять цвет заднего фона и шрифта

1. Требования к организации входных и выходных данных

Личные данные пользователя хранятся в базе данных, их может увидеть только он в своём профиле. Тексты книг хранятся в базе данных в виде текста, а аудиокниги в формате .mp3 в storage.

1. Требования к безопасности

Перед отправкой в базу данных пароли хешируются средствами bcrypt. Доступ к панели администрации есть только у администратора, пароль и логин которого хранятся в программе. Есть разделение функционала для авторизованного и неавторизованного пользователя.

1. Ограничения

* Размер аудиокниги не должен превышать 30 МБ
* Размер изображения профиля не должен превышать 2 МБ
* Изображения книг и аудиокниг не должны превышать 2 МБ

# Обоснование проектных решений

1.2.1 Обоснование выбора языков программирования

Клиентская часть приложения будет написана средствами языка JavaScript, его преимуществами является полная интеграция с вёрсткой страниц и серверной частью приложения, простой синтаксис и высокая скорость работы. К тому же, для JS есть множество полезных фреймворков, один из которых я использую в разработке приложения.

Для вёрстки страниц будет использоваться язык гипертекстовой разметки HTML. Он поддерживается всеми браузерами, прост в использовании, легко интегрируется с JavaScript и СSS, позволяет использовать шаблоны, что упрощает создание веб-страниц.

Стили приложения написаны с помощью TailwindCSS, это CSS фреймворк для стилизации веб-приложений. Он обладает большим набором готовых классов, что сокращает время на создание пользовательского интерфейса, даёт возможность создавать адаптивные интерфейсы, настраивать и переопределять готовые классы для уникальной стилизации, содержит в себе инструменты для упрощения разработки, например, автодополнение классов в редакторе кода, визуальные инструменты для настройки цветовой схемы и многое другое.

Серверная часть приложения построена с помощью языка программирования PHP. Преимуществами этого языка является процедурный и объектно-ориентированный подход, что значит, что он подойдёт и для новичков, и для опытных программистов, PHP – бесплатный язык программирования с открытым исходным кодом, любой желающий может начать на нём писать и он поддерживается широким сообществом, к которому можно обратиться в случае возникновения проблем или вопросов. К тому же PHP поддерживает все главные базы данных, такие как MySQL и SQLite, имеет высокую скорость и стабильность.

* + 1. Инструментальные средства

Для создания дизайна пользовательского интерфейса я использовала графический редактор Figma. Это онлайн-сервис для разработки интерфейсов и прототипирования с возможностью организации совместной работы в режиме реального времени. Преимуществами данного редактора является бесплатный тариф, неограниченное файловое хранилище, возможность командной работы и библиотека компонентов.

Для написания клиентской части использовался Vue.js - JavaScript-фреймворк с открытым исходным кодом для создания пользовательских интерфейсов. Легко интегрируется в проекты с использованием других JavaScript-библиотек. Может функционировать как веб-фреймворк для разработки одностраничных приложений в реактивном стиле.

Для написания серверной части кода использовался PHP фреймворк – Laravel. Он имеет большую экосистему с мгновенным разворачиванием своей платформе, позволяет создавать как монолитные приложения с использованием blade шаблонов, так и использовать принцип REST API, этот фреймворк имеет много встроенных функций, упрощающих разработку приложения, например, работу с cookie, session, регистрацией, авторизацией и другое. Преимуществом работы с Larevel является наличие сайта с обширной документацией большим количеством мануалов и информации для ознакомления.

1.2.3 Обоснование выбора среды программирования

Visual Studio Code - Текстовый редактор, разработанный Microsoft для Windows, Linux и macOS. Позиционируется как «лёгкий» редактор кода для кроссплатформенной разработки веб- и облачных приложений. Он легковесный, имеет интуитивно понятный интерфейс, благодаря чему Visual Studio Code может освоить даже новичок.

* + 1. Информационное обеспечение

В качестве хранилища данных будут использоваться базы данных, так как благодаря им можно сохранить избыточность, целостность и непротиворечивость данных, поддерживать безопасность, стандарты и эффективность и совместно использовать данные.

Для написания пояснительной записки использовался Microsoft Word - официальное приложение системы Windows, основным заданием которого является просмотр, создание и редактирование документов в формате текста, оно имеет удобный интерфейс и занимает мало места в памяти устройства.

Для создания презентации по курсовому проекту была использована программа Microsoft PowerPoint - это универсальный инструмент для подготовки визуальных цифровых презентаций, получивший широкое распространение в таких отраслях, как бизнес и учебный процесс

Для упрощения разработки использовалась система контроля версий – GIT, которая позволяет отслеживать историю изменений файлов в проекте.

# Обзор и анализ существующих программных систем

1. Литрес

Достоинства:

* Есть бесплатные книги
* Пользователь платит за книгу, которая останется у него навсегда, а не за подписку, которую необходимо продлевать, то есть можно читать книги в своём темпе
* Перед покупкой можно прочитать ознакомительный фрагмент, чтобы поближе познакомиться с произведением
* Большинство книжных новинок выходит в Литрес вместе с бумажной версией
* Есть промокоды, дающие скидки на покупки или бесплатные книги

Недостатки

* Часто высокие цены на книги
* Большая часть аудиокниг озвучена любителями, а не профессионалами с лишними шумами, щелчками и прочим

1. Mybook

Достоинства:

* Приложение составляет статистику по прочитанному пользователем материалу
* Синхронизация библиотеки с Литрес
* Большой выбор книг и аудиокниг
* Есть бесплатные книги

Минусы:

* Дорогостоящая подписка

1. Wattpad

* Есть возможность добавления авторских книг
* В приложении часто ведутся голосования на разные номинации, что позволяет выигрывать призы
* Есть прямая связь между автором книги и пользователем
* Можно оставлять заметки/комментарии и обсуждать их с другими пользователями во время чтения
* Все книги бесплатные

Недостатки:

* Много некачественной литературы, не соблюдены пунктуационные, орфографические и другие ошибки
* Нет аудиокниг
* Маленькая вероятность найти книгу, которая вышла недавно

# Практическая составляющая работы

# 2.1 Анализ задачи

2.1.1 Информационное моделирование предметной области

При моделировании предметной области стоит обратить внимание на такие особенности системы:

* Одна книга может состоять в нескольких подписках
* Одна подписка может быть у многих пользователей
* Одна книга может быть в избранном у многих пользователей
* На одну книгу может быть много комментариев

Учитывая эти особенности выделим сущности предметной области в таблице 1.

Таблица 1 – Анализ предметной области

|  |  |  |
| --- | --- | --- |
| Предметная область | Объекты | Характеристики объектов |
| Веб-приложение для чтения книг и прослушивания аудиокниг | User | id, path\_to\_photo, nickname, description, id\_subscription, email, password, isAdmin |
| Book | id, author, title, content, description, path\_to\_image, path\_to\_audio, type |
| Subscription | id, name, cost |
| AppReview | id, content, count\_star, user\_id |
| BookReview | id, content, count\_star, book\_id, user\_id |

В приложении Б.1 – Б.2 можно увидеть диаграмму классов и IDEF1X диаграмму, построенную в результате анализа предметной области

# Реализация

# 2.2.1 Практическая реализация алгоритмов

Алгоритм программы - это набор инструкций, описывающих порядок действий исполнителя для решения определённой задачи.

Рассмотрим алгоритм чтения книги пользователем. В нём используются пользовательские функции splitTextIntoPages, close на стороне клиента, book, setLastPage на стороне сервера. Также в нём используются стандартные функции onMounted, ref, useRoute, useRouter. При монтировании страницы на вход поступают данные о книге, такие как content, description, path\_to\_image, author, title и last\_page, если пользователь уже открывал эту книгу. В методе splitTextIntoPages книга разбивается на страницы, блок-схему данного алгоритма можно увидеть в приложении В.1. В цикле for книга разбивается на страницы по 930 символов для удобочитаемости с помощью стандартного метода substring. Когда пользователь хочет выйти со страницы с текстом книги, он нажимает на символ Х в верхнем правом углу страницы, в этот момент срабатывать метод close, который передаёт данные о last\_page на сервер, где методом setLastPage в базу данных в таблицу user\_book заносится последняя страница книги.

2.2.2 Используемые методы

2.2.2.1 Стандартные методы

onMounted – метод, который регистрирует обратный вызов, который будет вызван после того, как компонент будет смонтирован

reactive – метод, в котором возвращаемое состояние будет являться реактивным объектом

ref – метод, возвращающий реактивный объект, который можно изменять

computed - это функция в Vue Composition API, которая позволяет определить вычисленное свойство в компоненте Vue

2.2.2.2 Пользовательские методы

setLike – метод для добавления книги в избранное

setLastPage – метод, устанавливающий последнюю страницу книги, на которой пользователь остановился

setLastTime – метод, устанавливающий последнее время в секундах, где остановился пользователь при прослушивании аудиокниги

createReview – создание отзыва на книгу

updateReview – изменение отзыва на книгу

# 2.2.3 Составные части программы и связи между ними

Программа построена на архитектуре “Клиент-сервер”, она предусматривает разделение процессов предоставление услуг и отправки запросов на них на разных компьютерах в сети, каждый из которых выполняют свои задачи независимо от других.

Клиенты и серверы обмениваются сообщениями в шаблоне запрос-ответ. Клиент отправляет запрос, а сервер возвращает ответ. Этот обмен сообщениями является примером межпроцессного взаимодействия.

Преимуществами архитектуры «клиент-сервер» являются централизованность, информационная безопасность, производительность и масштабируемость.

* + 1. Реализация интерфейса

Основным элементом пользовательского интерфейса является шапка, в которой находятся ссылки для навигации по сайту, которую можно увидеть в приложении.

Неавторизованный пользователь в шапке профиля может увидеть ссылки на каталог с книгами, аудиокнигами, перейти на страницу «О нас» и попасть на главную страницу, нажав на логотип, также есть кнопки для авторизации и регистрации что можно увидеть в приложении Г.1. В приложении Г.2 видно, что у авторизованного пользователя исчезают кнопки для авторизации и регистрации, но появляется ссылка на личный кабинет и кнопка, позволяющая выйти из личного кабинета.

В каталоге с книгами и аудиокнигами, который находится в приложении Г.3 располагается список книг, при клике на книгу можно попасть на страницу с её описанием, например, как можно увидеть в приложении Г.4, на этой странице, если вы перешли на книгу, есть возможность прочесть отрывок книги – приложение Г.5, или прочесть книгу в подписке – приложение Г.6, если человек перешёл на аудиокнигу, у него есть возможность нажать на кнопку для прослушивания. Внизу страницы появится плеер с возможностью перемотки аудио на 15 секунд назад и вперёд, что видно в приложении Г.7.

Чтобы пользователю попасть в свой профиль, ему следует пройти регистрацию, а после авторизоваться в своём профиле, эти страницы есть в приложении Г.8 – Г.9. После регистрации пользователь попадает в свой профиль, который можно увидеть в приложении Г.10, где располагаются избранные книги и аудиокниги пользователя, его отзывы на книги и приложение. Кнопка редактирование профиля ведёт на отдельную страницу – приложение Г.11, где есть функционал изменения nickname, описания, пароля и смена подписки.

Если пользователь является администратором, у него есть доступ к панели администрирования сайта, где он может удалять и добавлять необходимые данные, одну из страниц админ-панели находится в приложении Г.12

# Тестирование

Тестирование — это контроль качества любого продукта разработки: мобильного приложения, сайта или компьютерной программы.

Его задача — сделать конечную версию максимально удобной, надёжной и безопасной для пользователя.

В приложении Д.1 можно увидеть функции для тестирования получения книги для чтения её фрагмента с существующим и несуществующим id. А в приложении Д.2 можно увидеть результат тестирования данных функций.

Так же в приложении Д.3 – Д.5 представлены тест-кейсы для проверки изменения nickname и описания в профиле пользователя, добавления книги и аудиокниги в избранное.

# Руководство оператора

2.4.1 Назначение программы

Программа предназначена для русскоязычных пользователей, имеющих доступ в интернет и желающих прочитать книгу или послушать аудиокнигу в удобном формате с возможностью оставить отзыв о книге/приложении.

2.4.2 Условия выполнения

Чтобы пользователь мог попасть в систему, он должен иметь стабильный выход в интернет и один из популярных браузеров последней версии, например Yandex Browser, Google Chrome или их аналоги.

2.4.3 Техника безопасности при работе за компьютером

При занятиях с компьютером важно помнить об опасностях, которые вас подстерегают:

* если вы близко сидите у компьютера, то идет сильная нагрузка на глаза от монитора и может ухудшиться зрение;
* если вы долго сидите у компьютера, то возникает нагрузка на позвоночник;
* повышенный уровень шума при работе компьютера и принтера создает нагрузку на мозг;
* электромагнитное излучение от монитора может также привести к заболеваниям;
* пыль, возникающая при работе с компьютером, приводит к снижению иммунитета;
* компьютер работает под высоким напряжением, а при неисправной проводке может возникнуть возгорание монитора и поражение пользователя электротоком;
* а если вы играете длительное время в игры, то страдает ваше эмоциональное состояние.

Учитывая эти опасности, следует знать правила работы с компьютером:

* правильно установите компьютер: задняя часть монитора не должна быть направлена на людей и место их отдыха; диваны и кровати должны находиться на расстоянии не менее 2,5 - 3 метров от компьютера; на экране должен быть защитный фильтр.
* компьютер необходимо подключить через заземленную розетку и пилот.
* обязательно в помещении должен быть включен свет и должна гореть настольная лампа.
* при работе с компьютером надо соблюдать правильную осанку.
* расстояние от глаз монитора должно быть не менее полметра.
* помощником при работе с компьютером станут физические упражнения.
* после работы с компьютером необходимо включать ионизаторы воздуха.
* После завершения работы с компьютером следует:
* закрыть все активные программы и задачи;
* извлечь все накопители;
* отключить компьютер, а затем питание системника;
* выключить вспомогательную технику, бесперебойник, стабилизатор;
* вынуть все кабели из розетки;

# Заключение

В курсовой работе успешно были выполнены поставленные задачи, улучшились знания в области использования фреймворков, написания серверной части кода, баз данных и анализа предметной области. Улучшились навыки программирования и построения диаграмм.

В ходе разработки приложения были спроектированы IDEF0, IDEF1X диаграммы и диаграмма классов, блок-схема одной из функций программы. Средствами языка PHP и фреймворка Larevel были созданы миграции для базы данных приложения.  С помощью фреймворка Vue.js была написана клиентская часть приложения. Средствами онлайн-сервиса Figma был разработан дизайн интерфейса пользователя. Тестирование средствами PHPUnit и построение тест-кейсов подтвердило работоспособность приложения.

Были реализованы такие функции, как регистрация, авторизация, чтение книг, прослушивание аудиокниг, добавление книг в избранное, чтение фрагмента книги, редактирование профиля возможность оставить отзыв на приложение и на книгу.

# Список литературы

1. ГОСТ 19.701-90. Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения. Москва, Издательство стандартов, 2010. – 24с.

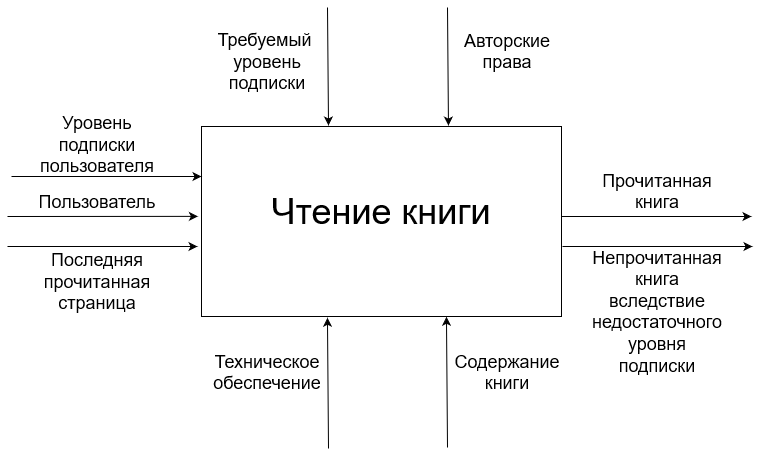
2. ГОСТ 7.32-2017. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления. – М.: Стандартинформ, 2017;

3. Официальная документация Vue.js. [Электронный ресурс] URL: [https://v3.ru.vuejs.org/ru/guide/introduction.html](https://www.google.com/url?q=https://www.google.com/url?q%3Dhttps://v3.ru.vuejs.org/ru/guide/introduction.html%26amp;sa%3DD%26amp;source%3Deditors%26amp;ust%3D1714077204006307%26amp;usg%3DAOvVaw2G3o1sVgoXFDCNTIB3xZcn&sa=D&source=docs&ust=1714077204229073&usg=AOvVaw1j2avCYg5UrG4xgaTKUv5U) (Дата обращения 25.03.2024).

4. Официальная документация Laravel [Электронный ресурс]

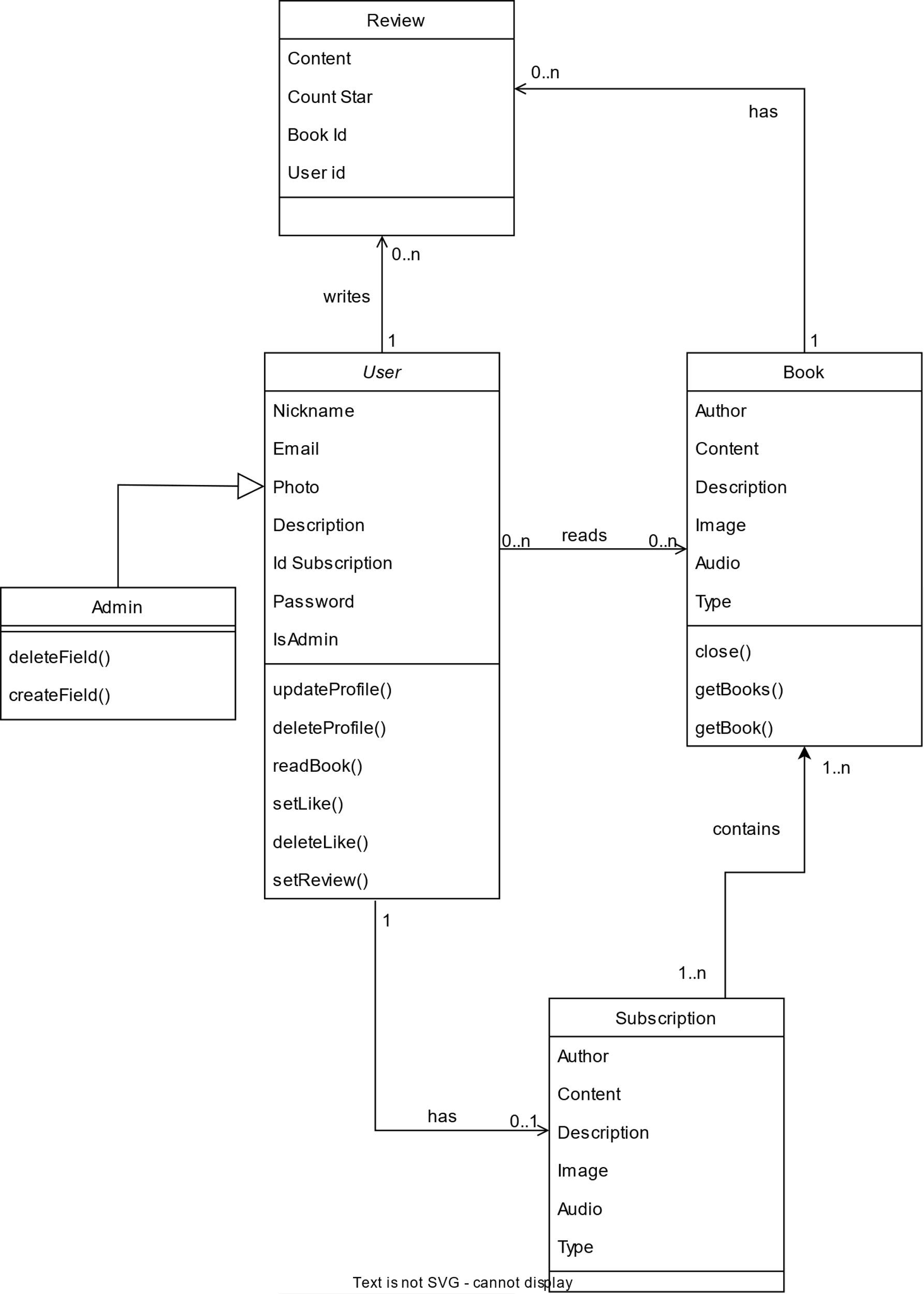
URL: <https://laravel.com/docs/10.x/documentation> (Дата обращения 13.04.2024).

# Приложение А

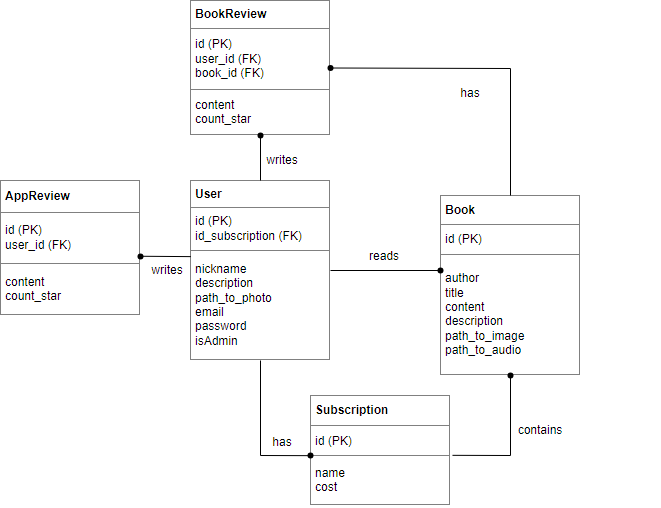


А.1 – IDEF0 диаграмма

# Приложение Б

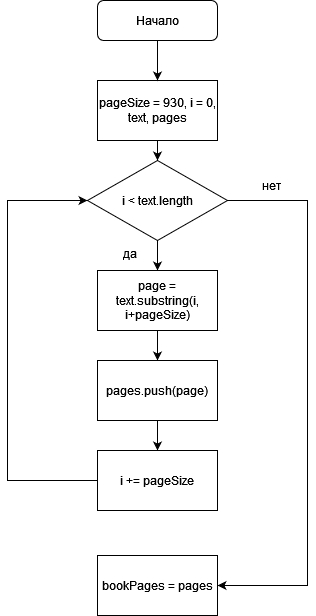


Б.1 – Диаграмма классов



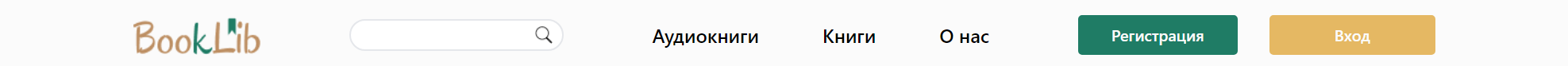
Б.2 – IDEF1X диаграмма

Приложение В

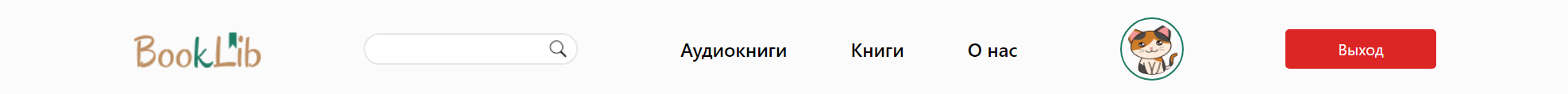


В.1 – Блок-схема функции splitTextIntoPages

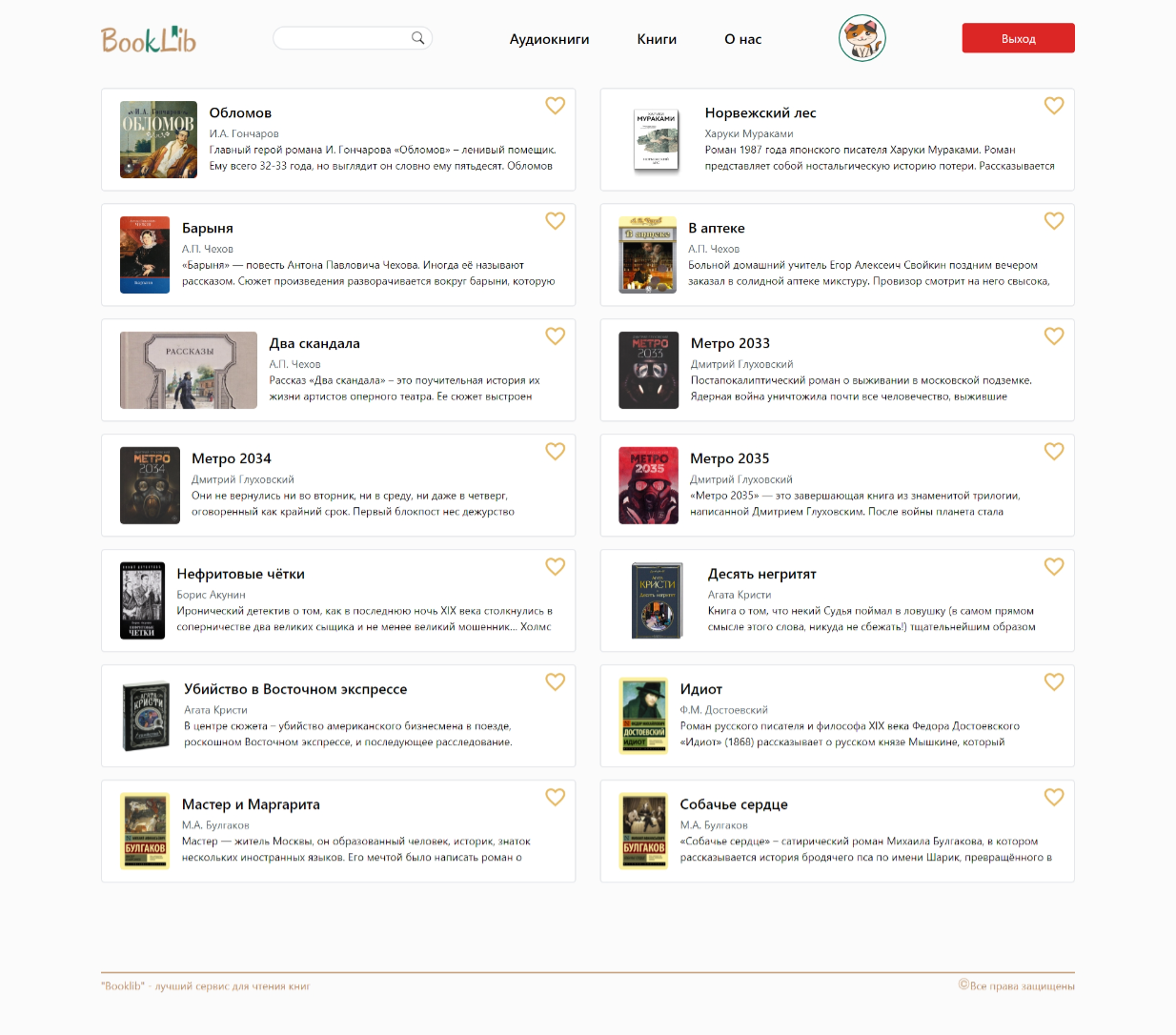
# Приложение Г

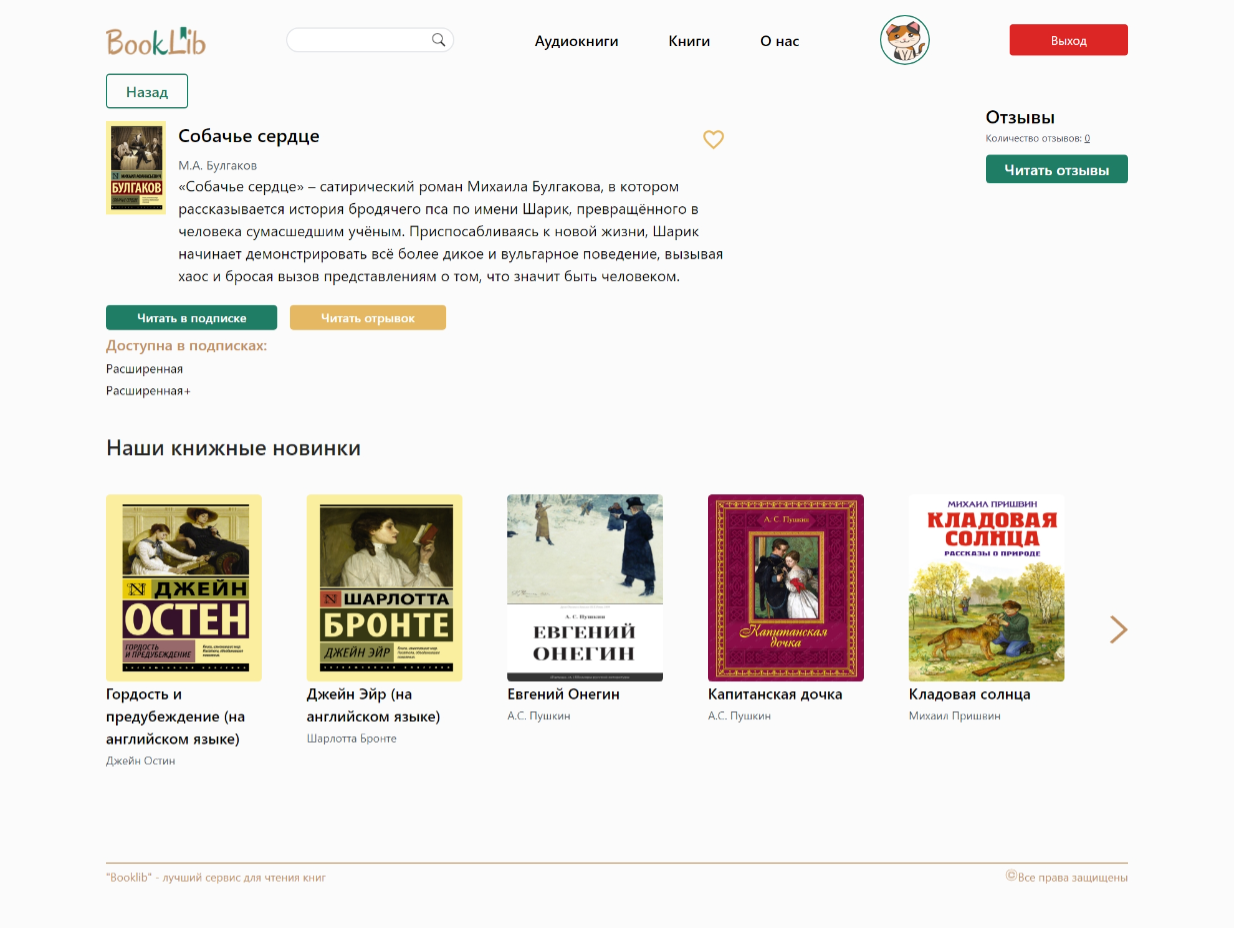


Г.1 – Шапка сайта неавторизованного пользователя

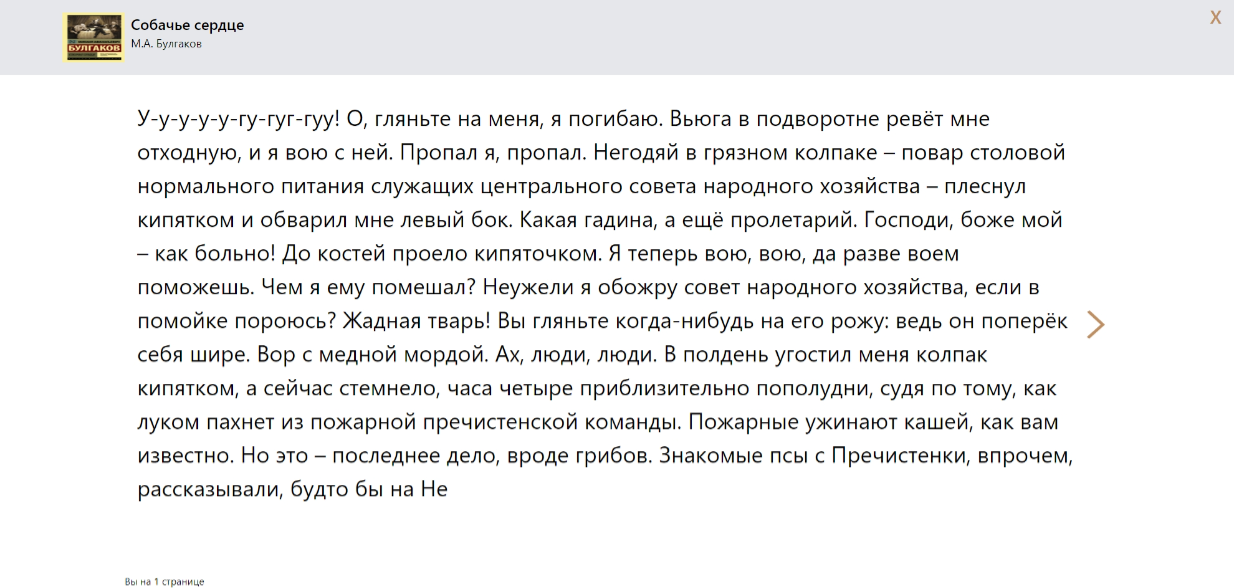


Г.2 – Шапка сайта авторизованного пользователя

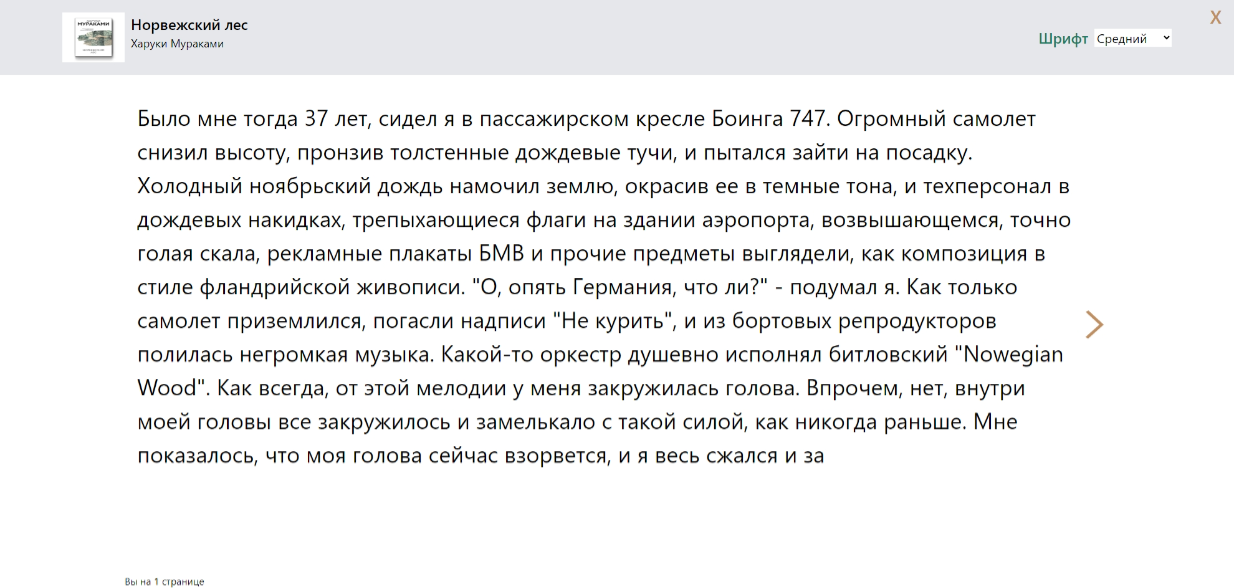
Г.3 – Каталог книг



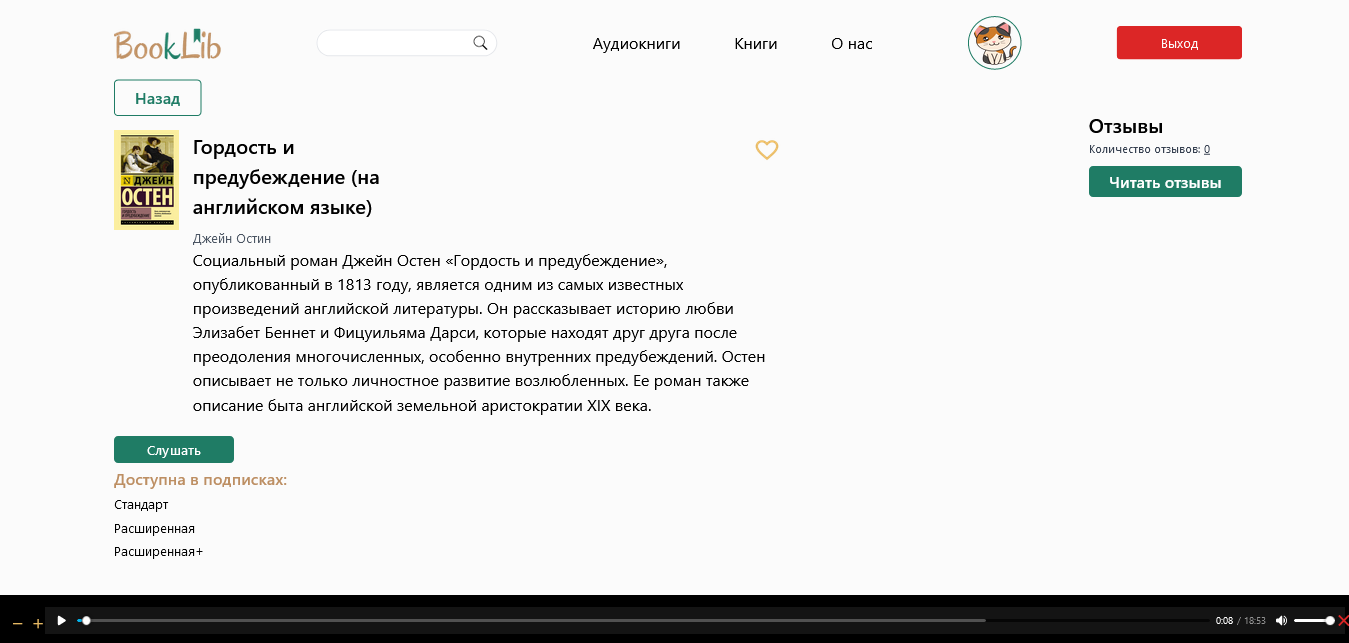
Г.4 – Описание книги



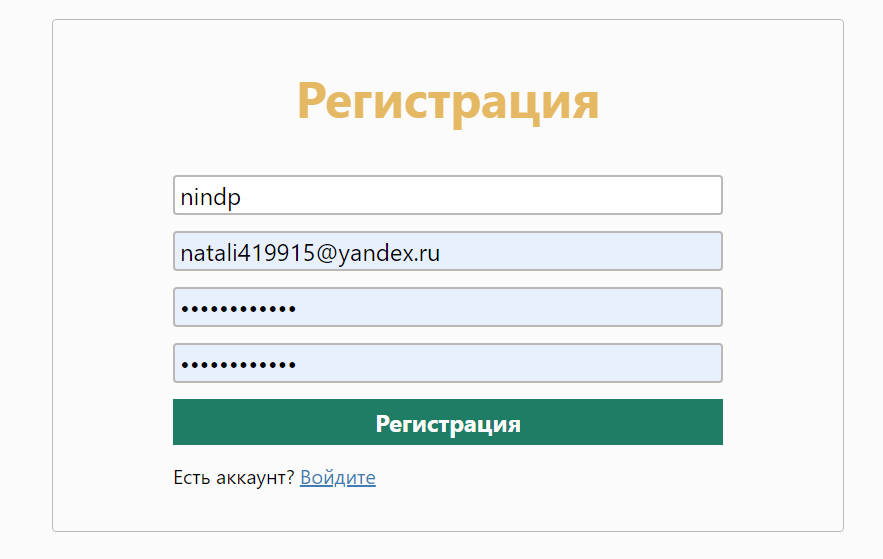
Г.5 – Страница с фрагментом книги



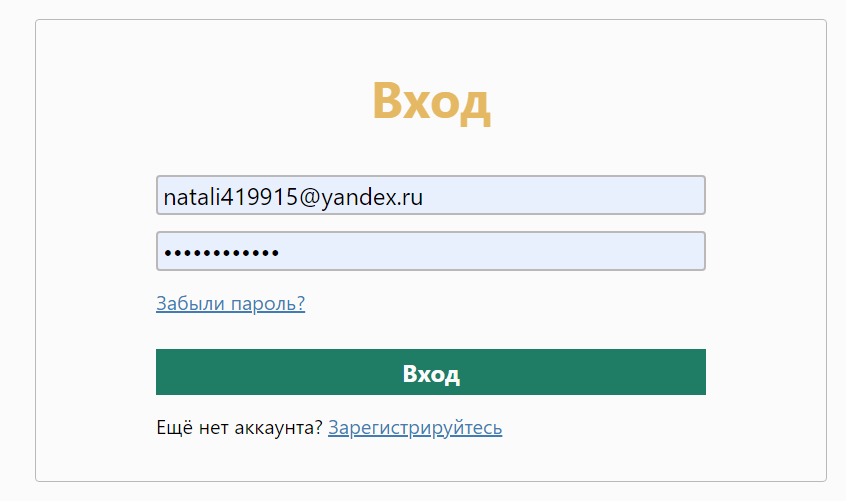
Г.6 – Страница для чтения книги



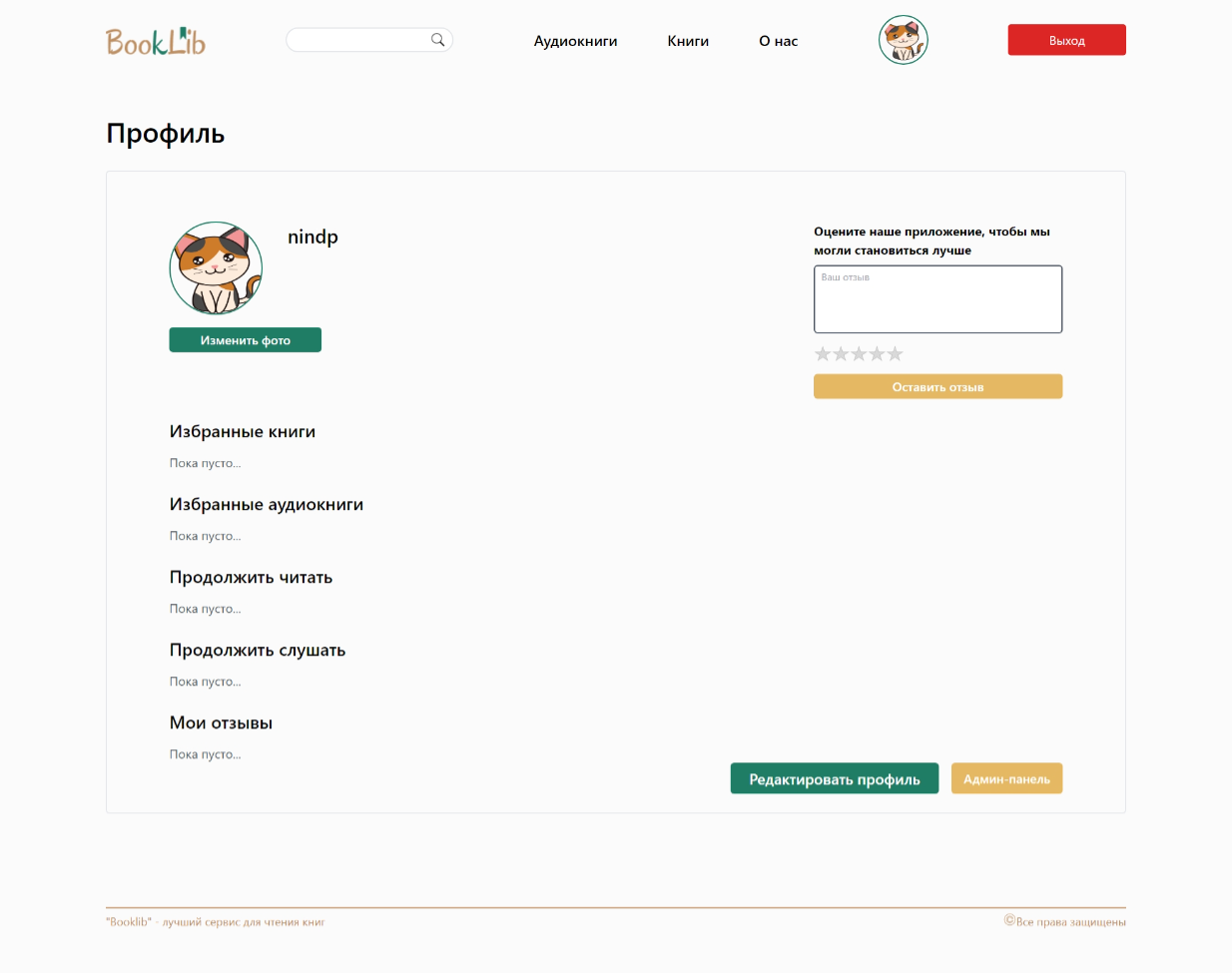
Г.7 – Страница для прослушивания аудиокниги



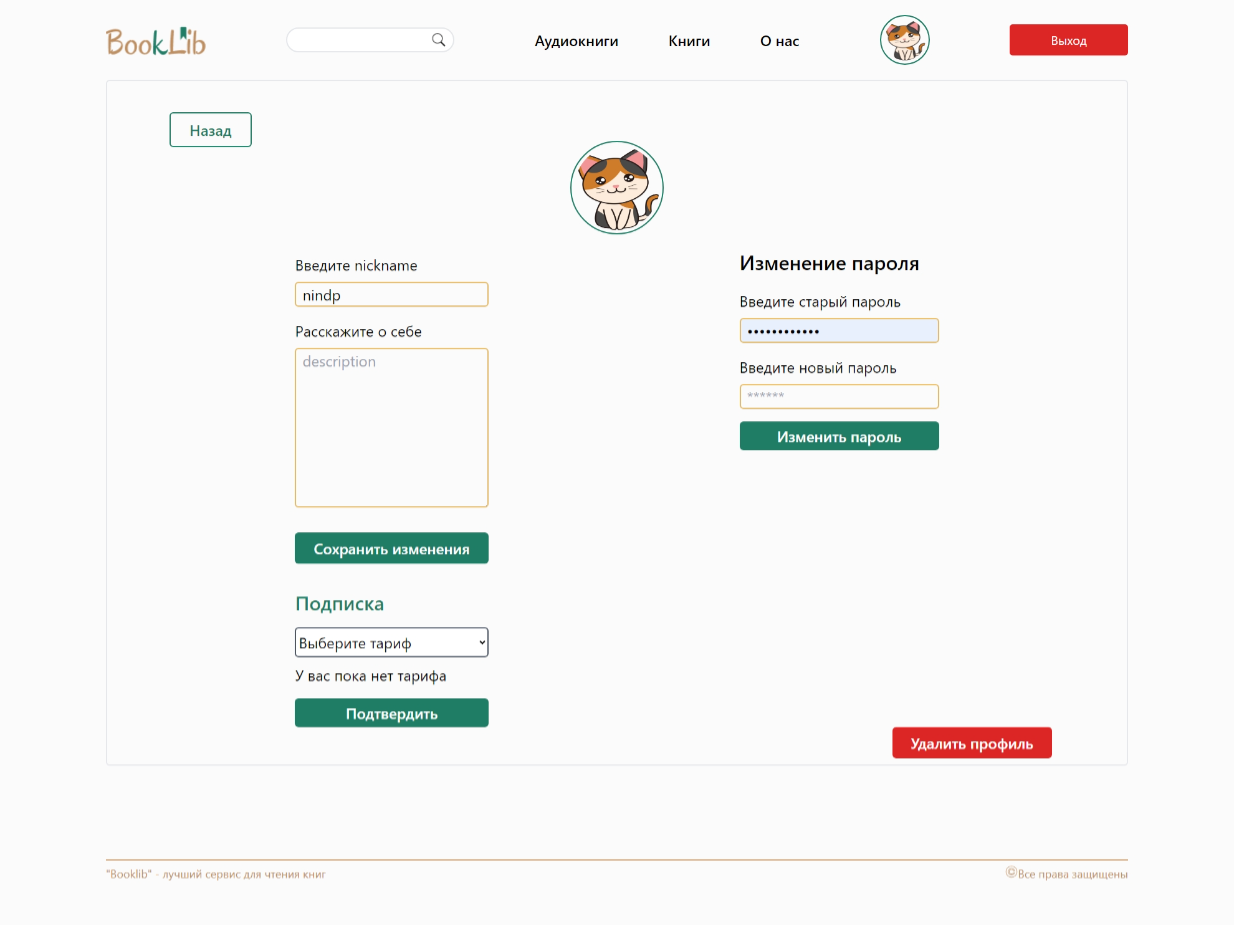
Г.8 – Страница регистрации



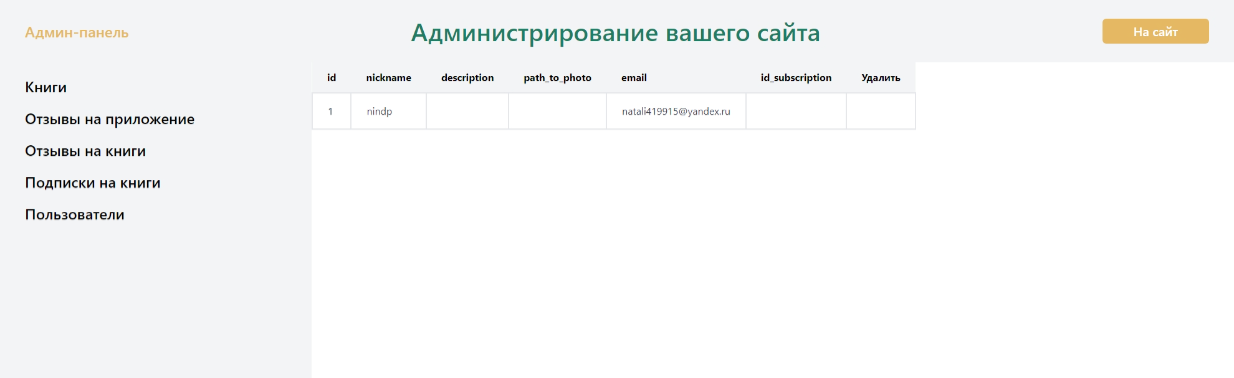
Г.9 – Страница авторизации



Г.10 – Профиль пользователя



Г.11 – Страница редактирования профиля

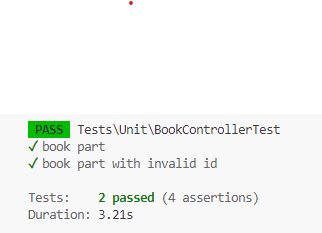


Г.12 – Админ-панель

# Приложение Д



Д.1 – Функции тестирования получения книги



Д.2 – Результат выполнения тестов

|  |  |
| --- | --- |
| Поле | Описание |
| Название проекта | BookLib |
| Рабочая версия | 1.0 |
| Имя тестирующего | nefedovanatalia |
| Дата(ы) теста | 25.04.2024 |
| Тестовый пример # | TC\_UI\_1 |
| Приоритет тестирования (Низкий/Средний/Высокий) | Средний |
| Заголовок/название теста | Смена описания в профиле пользователя |
| Краткое изложение теста | Когда пользователь на странице редактирования профиля меняет описание, при переходе на профиль пользователя, должно быть выведено изменённое описание |
| Этапы теста | 1.  Войти в систему под своим логином и паролем  2.    Перейти на страницу «Профиль»  3.    Нажать на кнопку «Редактировать профиль»  4.     Ввести в поле description «Привет, мир»  5.     Нажать на кнопку «Сохранить изменения» |
| Тестовые данные | Email пользователя: [natali419915@yandex.ru](mailto:natali419915@yandex.ru)  Пароль пользователя: Natasha199 |
| Ожидаемый результат | Появится alert с сообщением «Данные успешно изменены», пользователя перенаправит на его профиль, появится описание «Привет, мир» |
| Фактический результат | Появляется alert с сообщением «Данные успешно изменены», пользователь перенаправляется на свой профиль, появляется описание «Привет, мир» |
| Предварительное условие | Пользователь должен существовать в базе данных и иметь стабильное интернет-соединение |
| Постусловие |  |
| Статус (Зачет/Незачет) | Зачёт |
| Примечания/комментарии |  |

Д.3 – Тест-кейс для проверки изменения описания в профиле пользователя

|  |  |
| --- | --- |
| Поле | Описание |
| Название проекта | BookLib |
| Рабочая версия | 1.0 |
| Имя тестирующего | nefedovanatalia |
| Дата(ы) теста | 25.04.2024 |
| Тестовый пример # | TC\_UI\_2 |
| Приоритет тестирования (Низкий/Средний/Высокий) | Средний |
| Заголовок/название теста | Смена никнейма в профиле пользователя |
| Краткое изложение теста | Когда пользователь на странице редактирования профиля меняет никнейм, при переходе на профиль пользователя, должен быть выведен изменённый никнейм |
| Этапы теста | 1. Войти в систему под своим логином и паролем 2. Перейти на страницу «Профиль» 3. Нажать на кнопку «Редактировать профиль» 4. Ввести в поле nickname значение "nindp2" |
| Тестовые данные | Email пользователя: [natali419915@yandex.ru](mailto:natali419915@yandex.ru)  Пароль пользователя: Natasha199 |
| Ожидаемый результат | Появится alert с сообщением «Данные успешно изменены», пользователь будет перенаправлен на его профиль, появится никнейм "nindp2" |
| Фактический результат | Появляется alert с сообщением «Данные успешно изменены», пользователь перенаправляется на свой профиль, появляется никнейм "nindp2" |
| Предварительное условие | Пользователь должен существовать в базе данных и иметь стабильное интернет-соединение |
| Постусловие |  |
| Статус (Зачет/Незачет) | Зачёт |
| Примечания/комментарии |  |

Д.4 – Тест-кейс для проверки изменения nickname в профиле пользователя

|  |  |
| --- | --- |
| Поле | Описание |
| Название проекта | BookLib |
| Рабочая версия | 1.0 |
| Имя тестирующего | nefedovanatalia |
| Дата(ы) теста | 26.04.2024 |
| Тестовый пример # | TC\_UI\_3 |
| Приоритет тестирования (Низкий/Средний/Высокий) | Высокий |
| Заголовок/название теста | Добавление книги в избранное |
| Краткое изложение теста | Когда пользователь на странице книги или в каталоге нажимает на иконку сердца в верхнем правом углу, в его профиле в разделе «Избранные книги» должна появиться эта книга |
| Этапы теста | 1. Войти в систему под своим логином и паролем 2. Перейти на страницу «Книги» 3. Нажать на инонку сердца в верхнем правом углу у книги с названием «Метро 2035» 4. Перейти в профиль пользователя |
| Тестовые данные | Email пользователя: [natali419915@yandex.ru](mailto:natali419915@yandex.ru)  Пароль пользователя: Natasha199 |
| Ожидаемый результат | В профиле пользователя в разделе «Избранные книги» появится книга «Метро 2035» |
| Фактический результат | В профиле пользователя в разделе «Избранные книги» появилась книга «Метро 2035» |
| Предварительное условие | Пользователь и книга «Метро 2035» должен существовать в базе данных и иметь стабильное интернет-соединение |
| Постусловие |  |
| Статус (Зачет/Незачет) | Зачёт |
| Примечания/комментарии |  |

Д.5 – Тест-кейс для проверки добавления книги в избранное

|  |  |
| --- | --- |
| Поле | Описание |
| Название проекта | BookLib |
| Рабочая версия | 1.0 |
| Имя тестирующего | nefedovanatalia |
| Дата(ы) теста | 26.04.2024 |
| Тестовый пример # | TC\_UI\_4 |
| Приоритет тестирования (Низкий/Средний/Высокий) | Высокий |
| Заголовок/название теста | Добавление аудиокниги в избранное |
| Краткое изложение теста | Когда пользователь на странице аудиокниги или в каталоге нажимает на иконку сердца в верхнем правом углу, в его профиле в разделе «Избранные аудиокниги» должна появиться эта книга |
| Этапы теста | 1. Войти в систему под своим логином и паролем 2. Перейти на страницу «Аудиокниги» 3. Нажать на инонку сердца в верхнем правом углу у книги с названием «Сказка о рыбаке и рыбке» 4. Перейти в профиль пользователя |
| Тестовые данные | Email пользователя: [natali419915@yandex.ru](mailto:natali419915@yandex.ru)  Пароль пользователя: Natasha199 |
| Ожидаемый результат | В профиле пользователя в разделе «Избранные аудиокниги» появится аудиокнига «Сказка о рыбаке и рыбке» |
| Фактический результат | В профиле пользователя в разделе «Избранные аудиокниги» появилась аудиокнига «Сказка о рыбаке и рыбке» |
| Предварительное условие | Пользователь и аудиокнига «Сказка о рыбаке и рыбке» должен существовать в базе данных и иметь стабильное интернет-соединение |
| Постусловие |  |
| Статус (Зачет/Незачет) | Зачёт |
| Примечания/комментарии |  |

Д.3 – Тест-кейс для проверки добавления аудиокниги в избранное