МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение   
высшего образования «Самарский национальный исследовательский университет   
имени академика С.П. Королева»

(Самарский университет)

Институт информатики и кибернетики

Кафедра технической кибернетики

**ОТЧЕТ**

по лабораторной работе №4

**Разработка пользовательского интерфейса и взаимодействие с API**

Выполнили студенты

группы 6304-010302D

Красных Д.П.

Накашидзе Е.С.

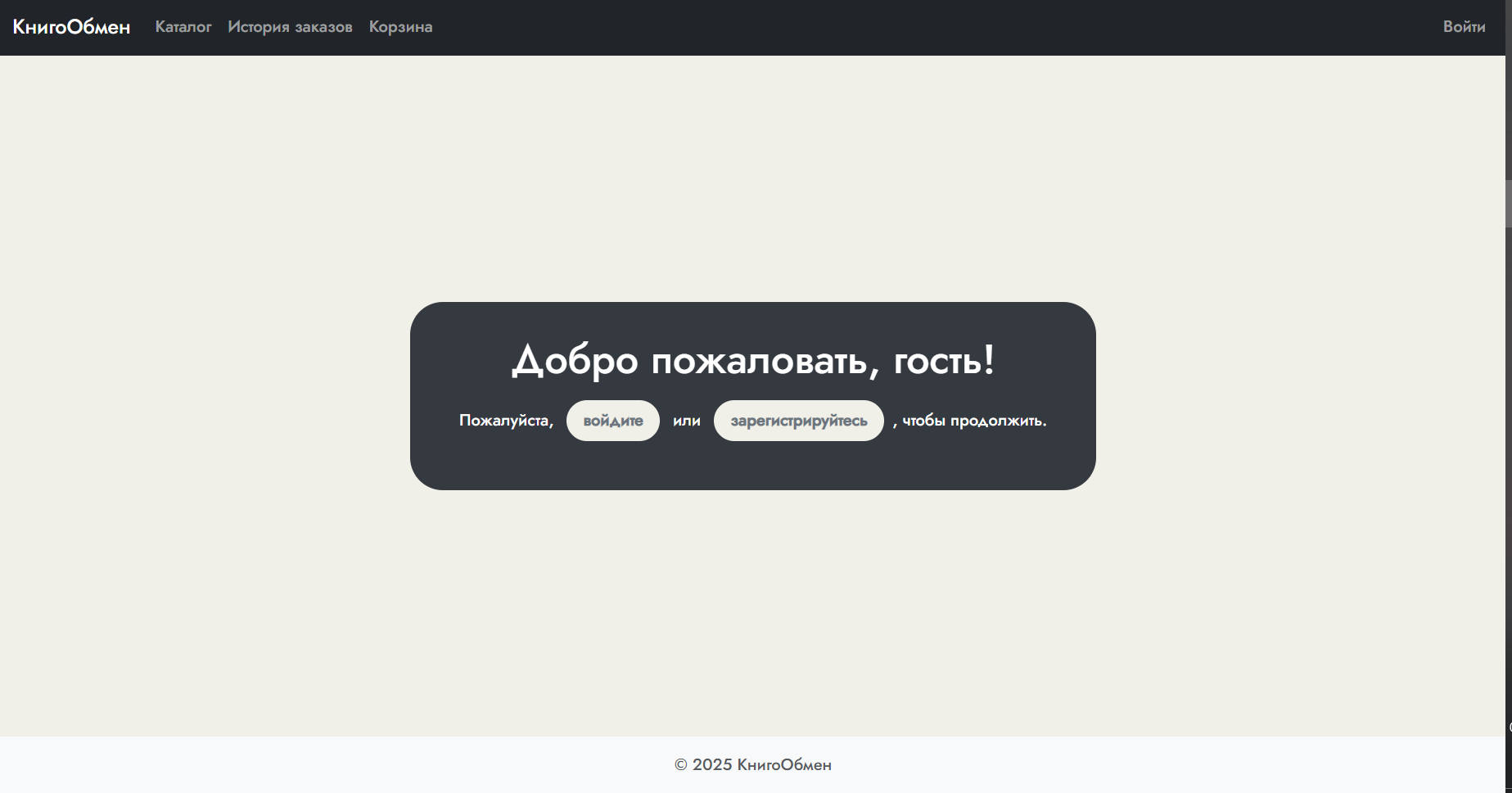
Самара 2025

**Цель работы:**

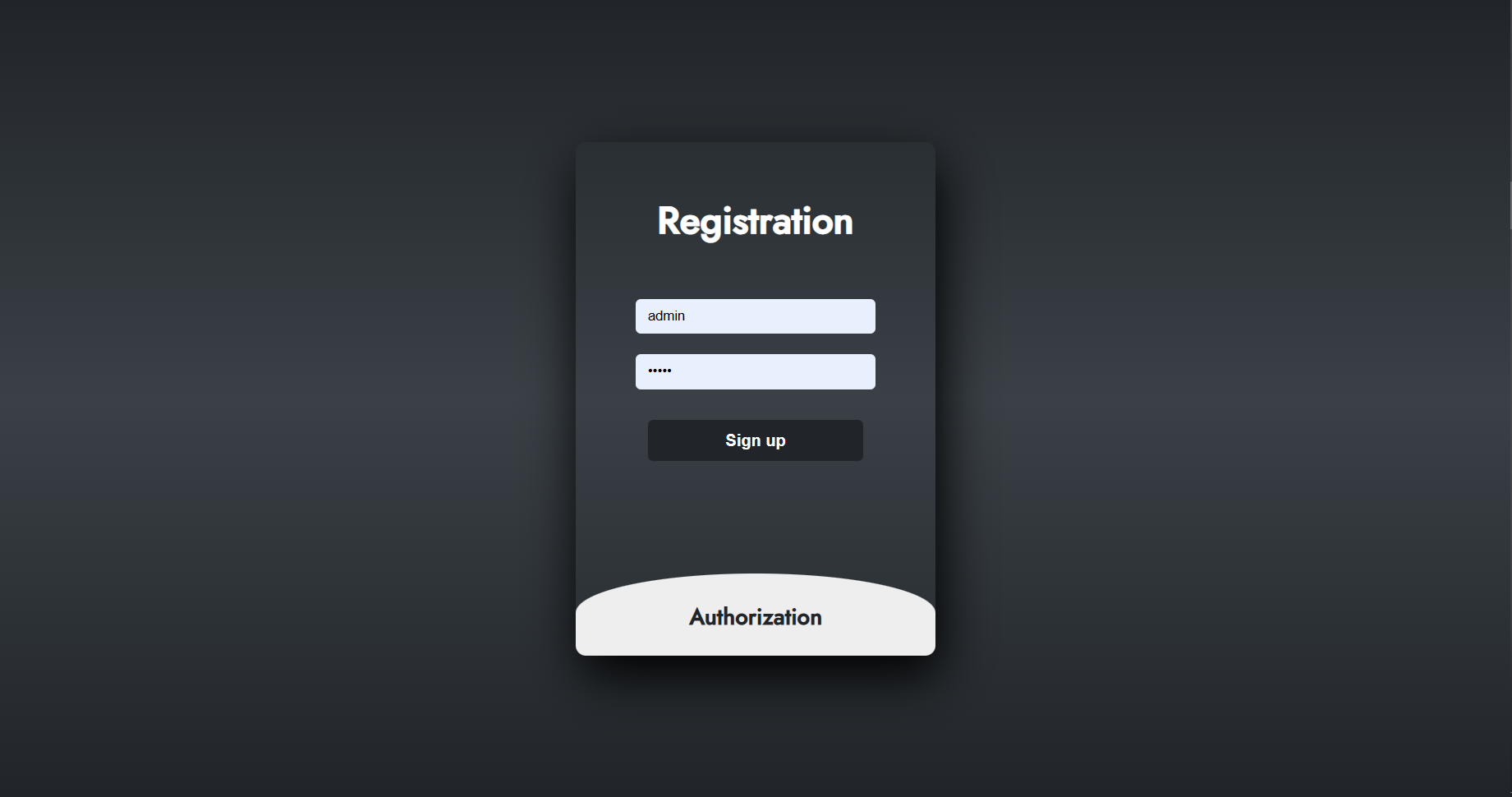
Создать клиентское веб-приложение, подключить его к серверу через API и реализовать основные пользовательские сценарии.

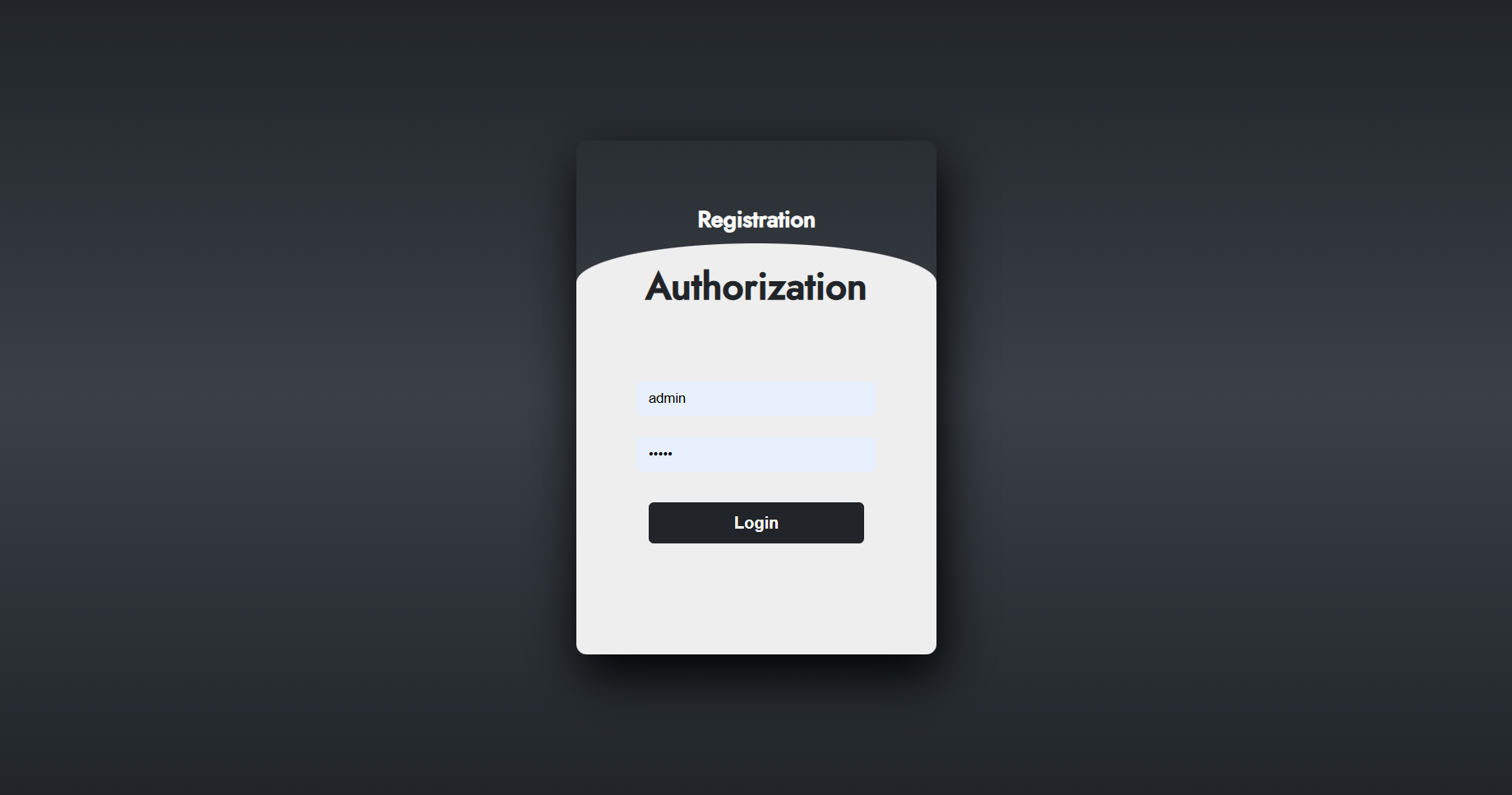
1. **Описание интерфейса**
   1. **Страница-приветствие;**

Так как гость только зашел на сайт, то единственное, что он может сделать, это зарегистрироваться или авторизоваться. Поэтому все ссылки с этой страницы ведут на страницу аутентификации.

****

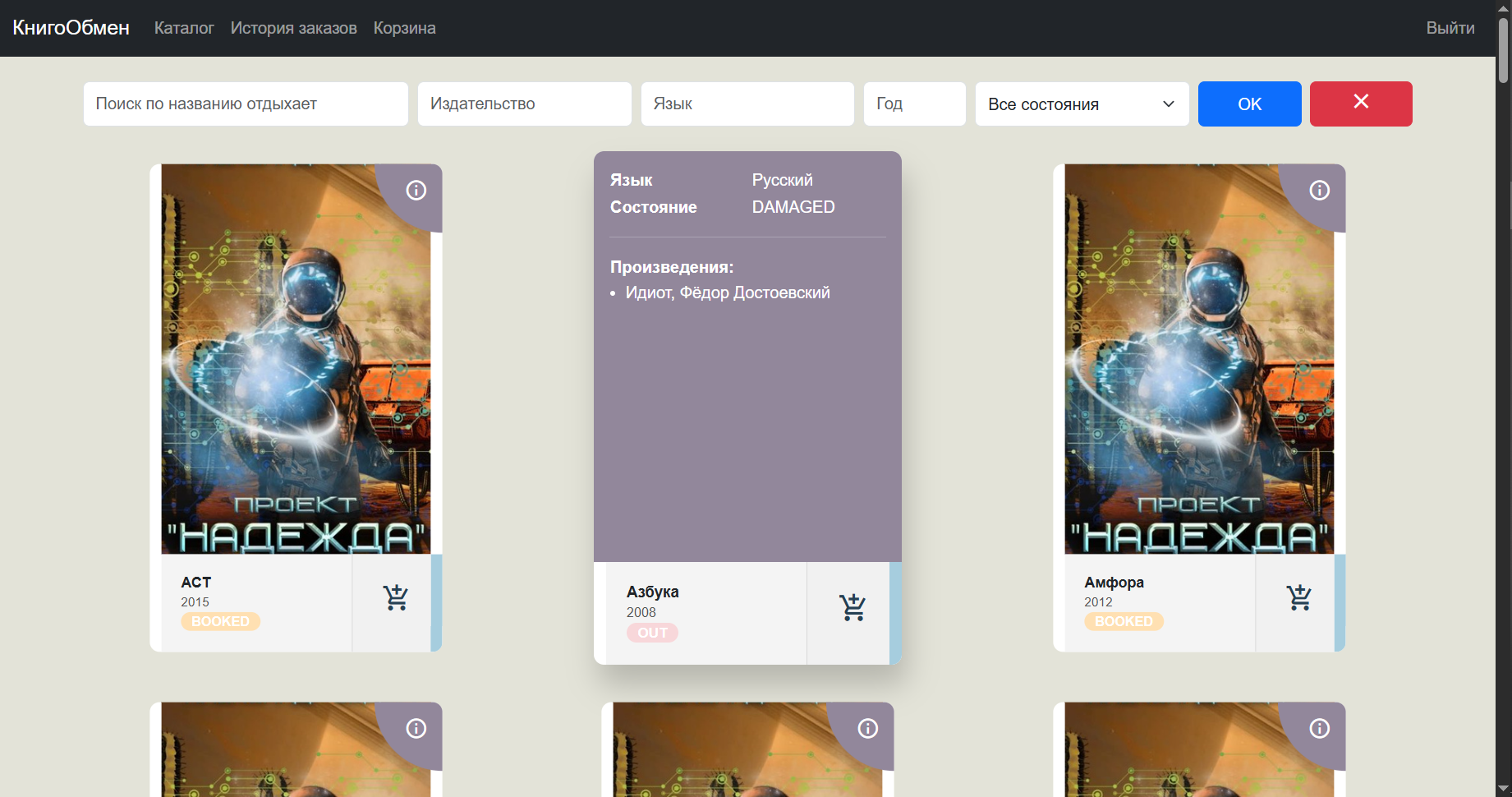
* 1. **Страница аутентификации;**

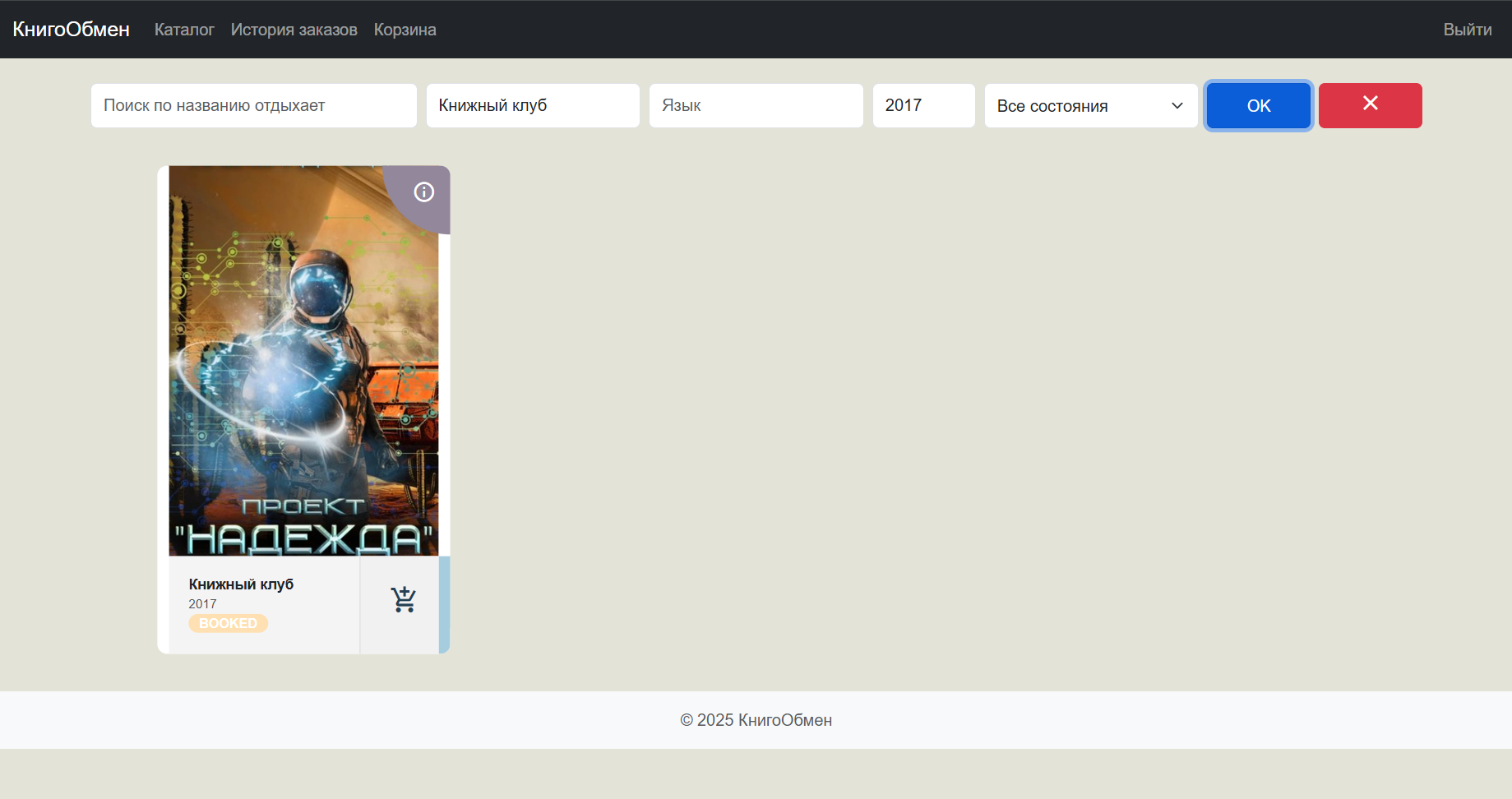


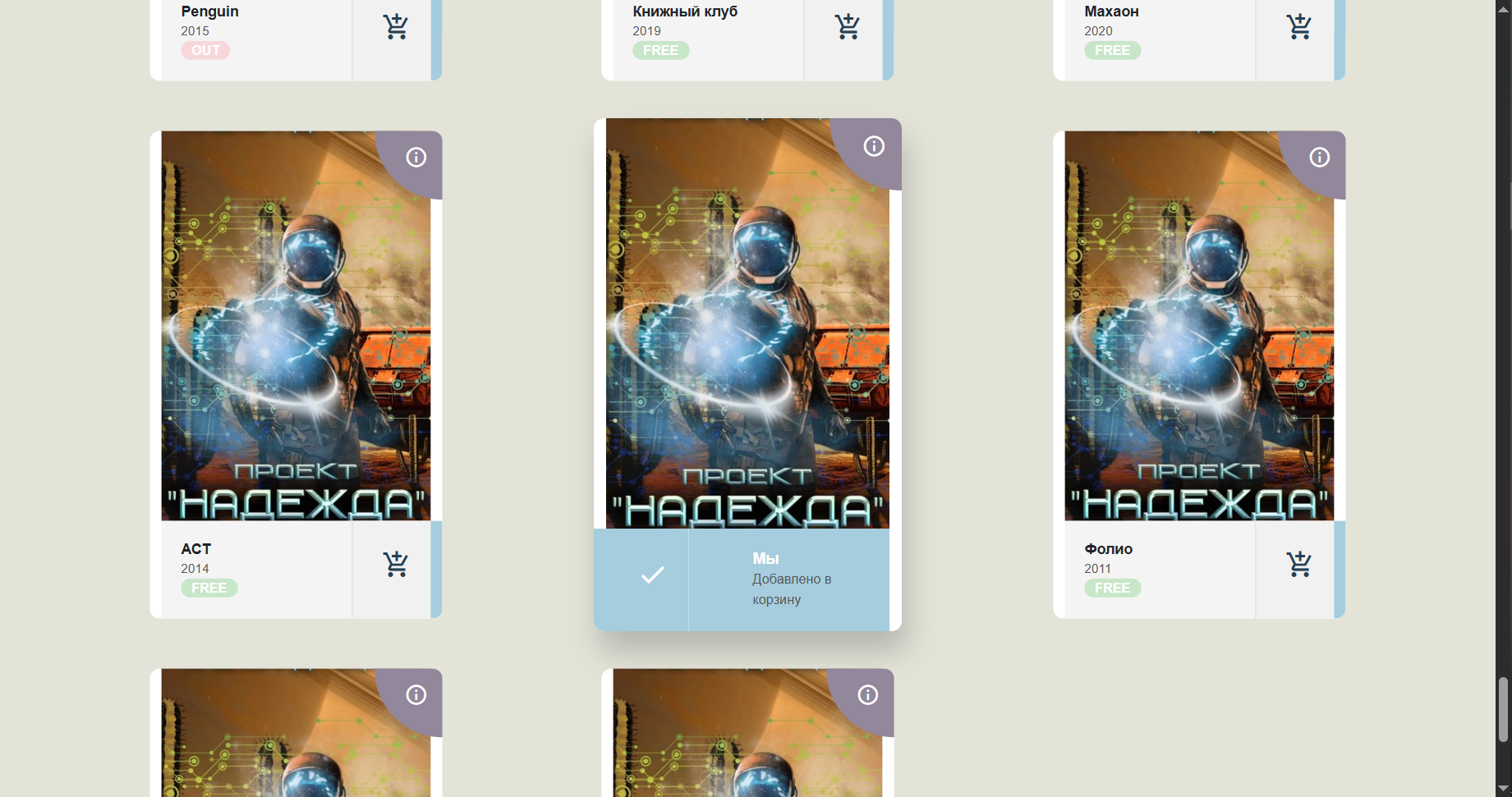


* 1. **Страница каталога книг;**

После авторизации и автоматического получения токена пользователю становится доступен весь функционал сайта, в том числе: выбор книг, добавление их в корзину, переход на страницы с корзинами и историей заказов и выход из аккаунта.

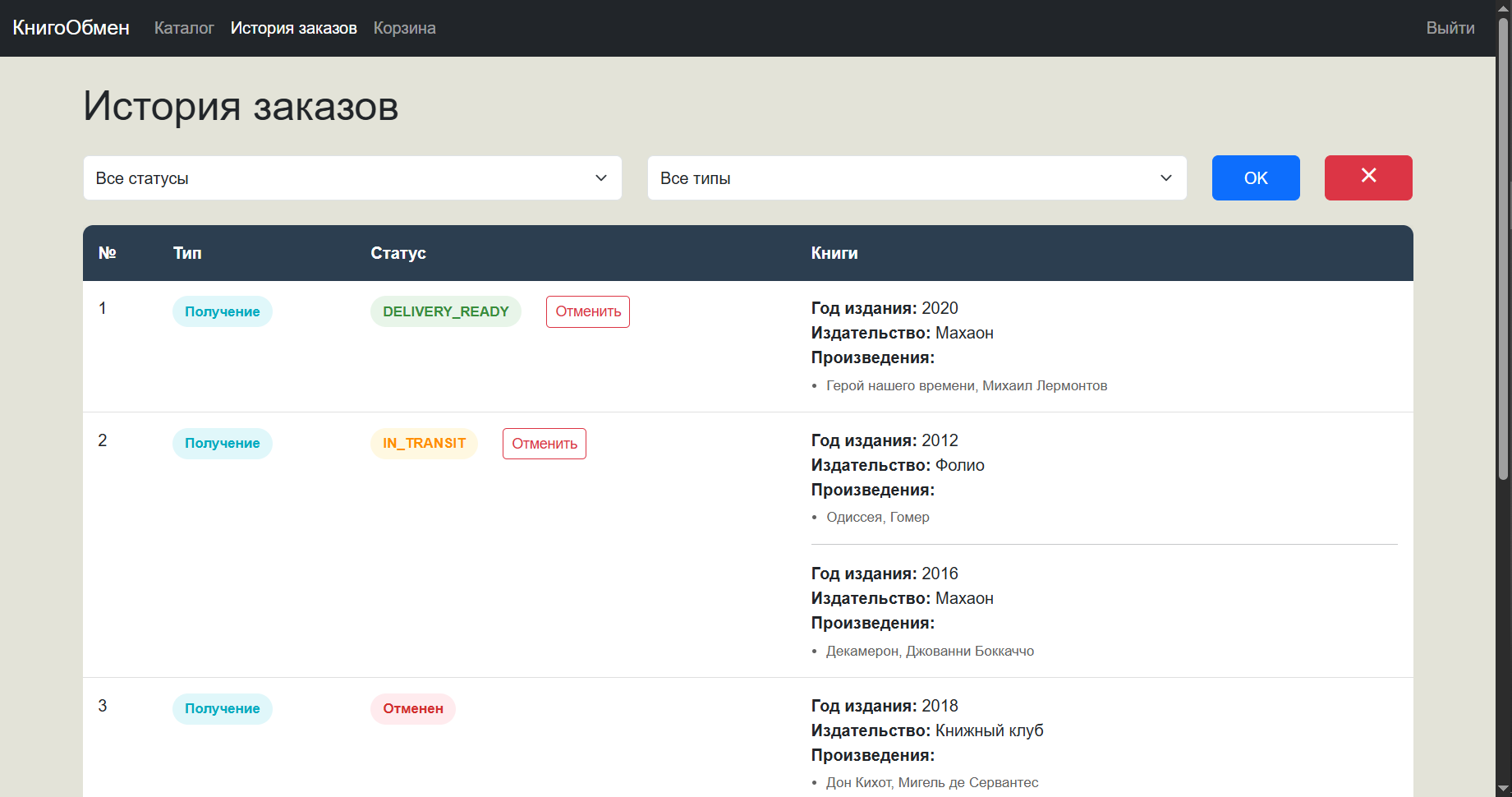
****

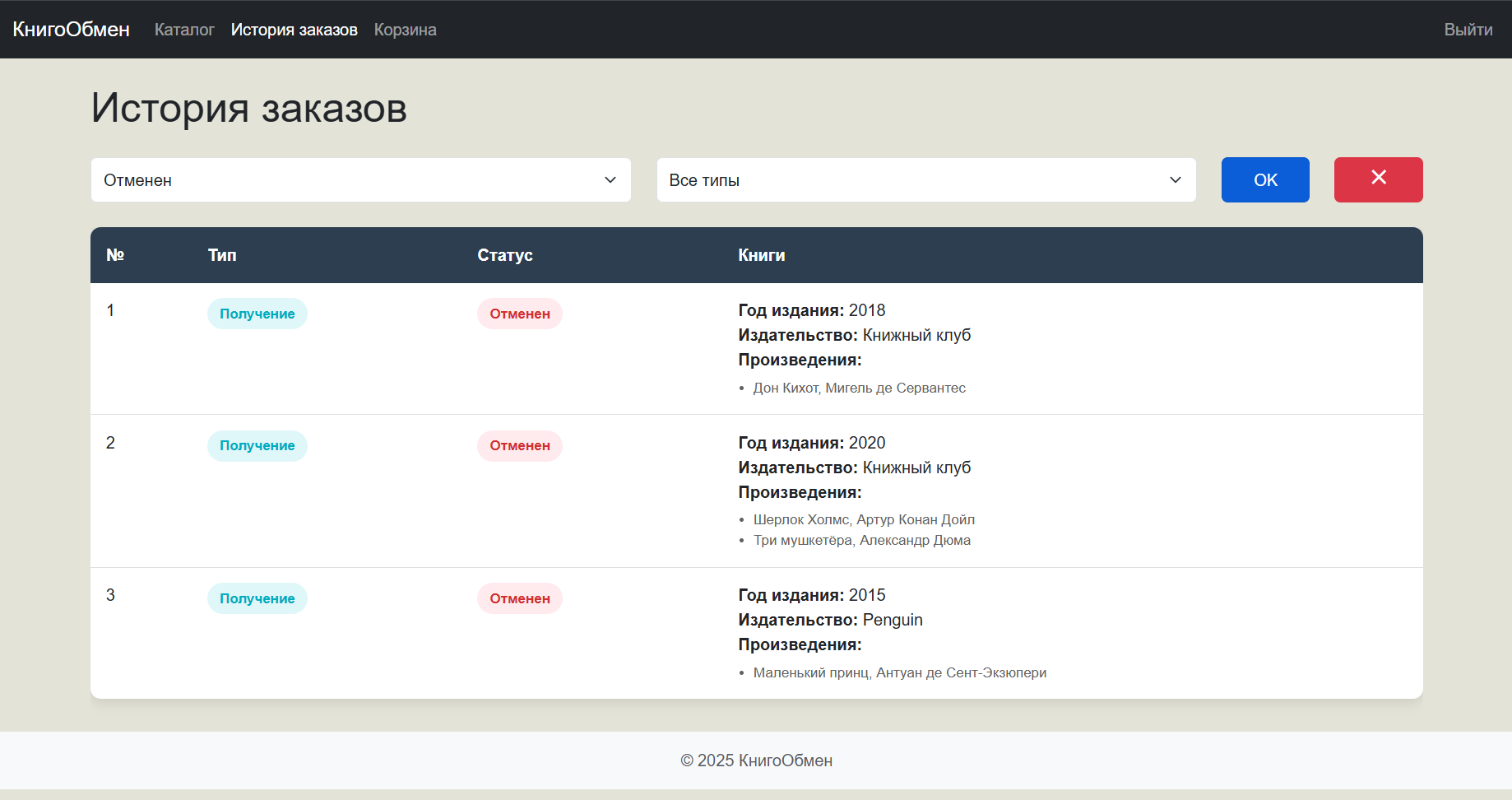
****

****

* 1. **Страница с предыдущими заказами;**

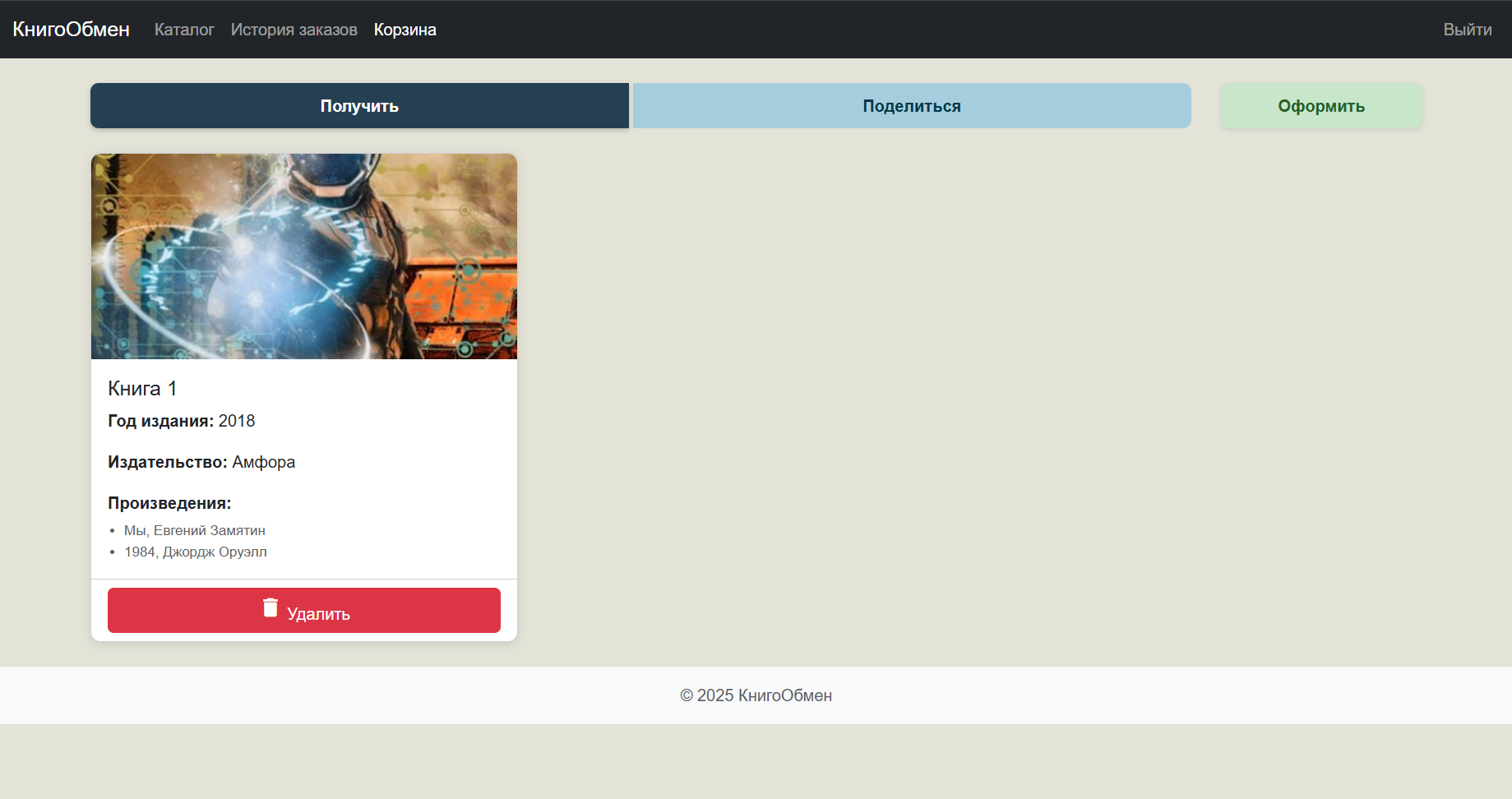
На данной странице пользователь может посмотреть и отфильтровать его предыдущие заказы, а также отменить незавершенные.

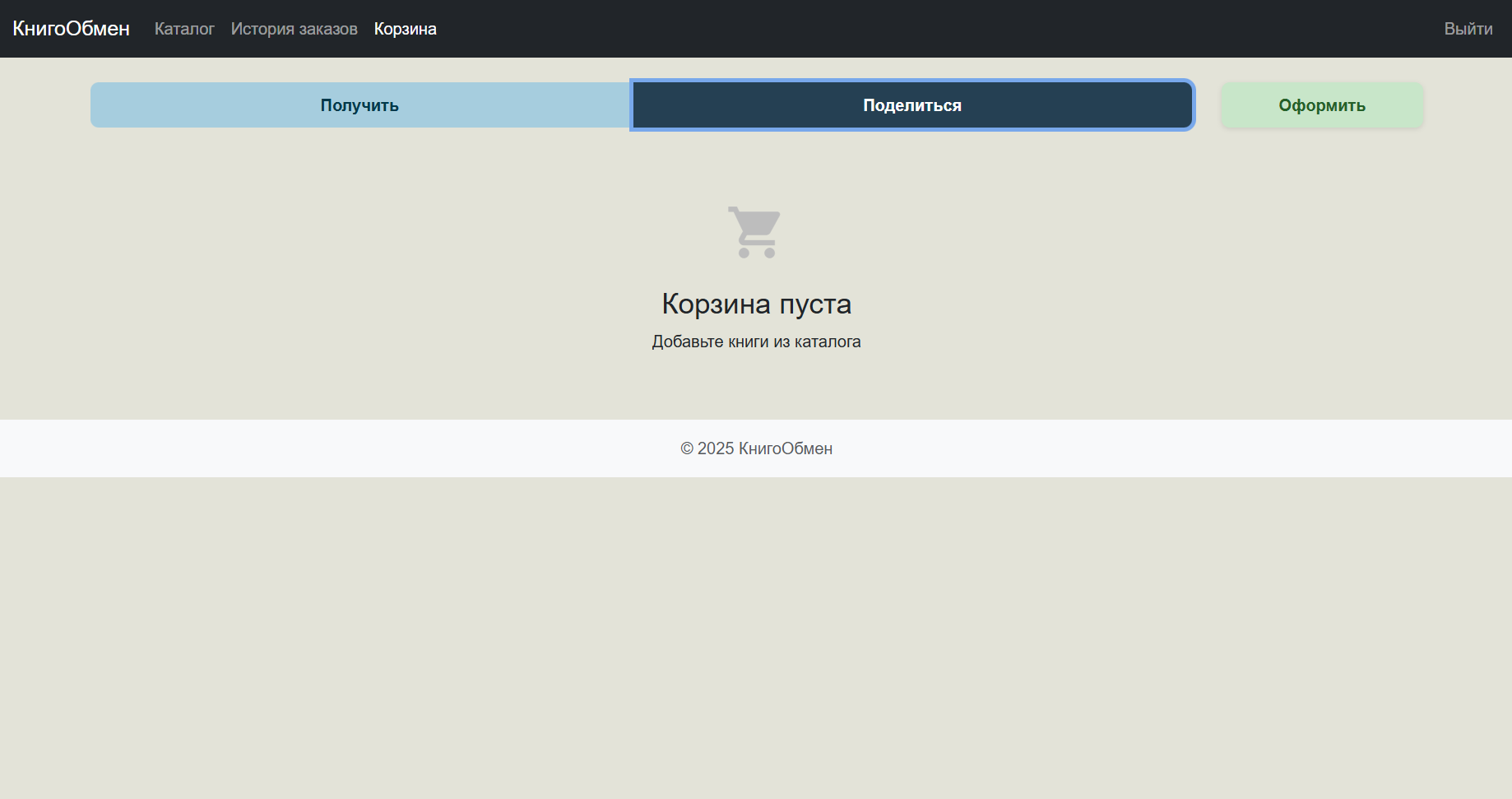
****

****

* 1. **Страница с действующей корзиной.**

Пользователь может просмотреть, какие книги у него лежат в корзинах (какие книги он хочет получить и какими он хочет поделиться), а также оформить заказ с просматриваемой в данный момент корзиной. Помимо этого он может удалить книги из корзины.



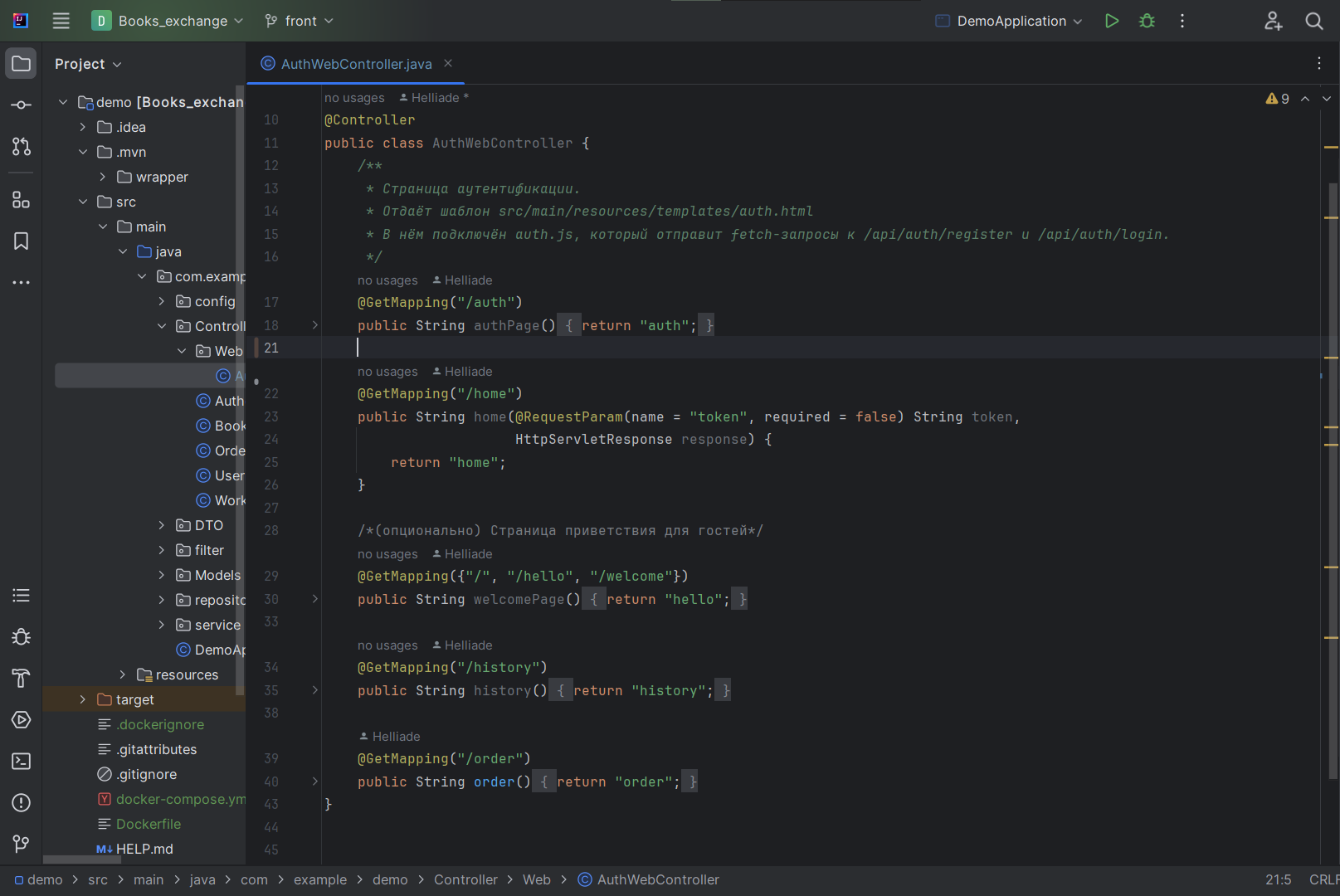


1. **Обработка данных**

Обработка всех данных, которыми обменивается бэкенд и фронтенд части проекта происходит в .js файлах. Переход на необходимые страницы по http запросам обрабатывается в AuthWebController.java.

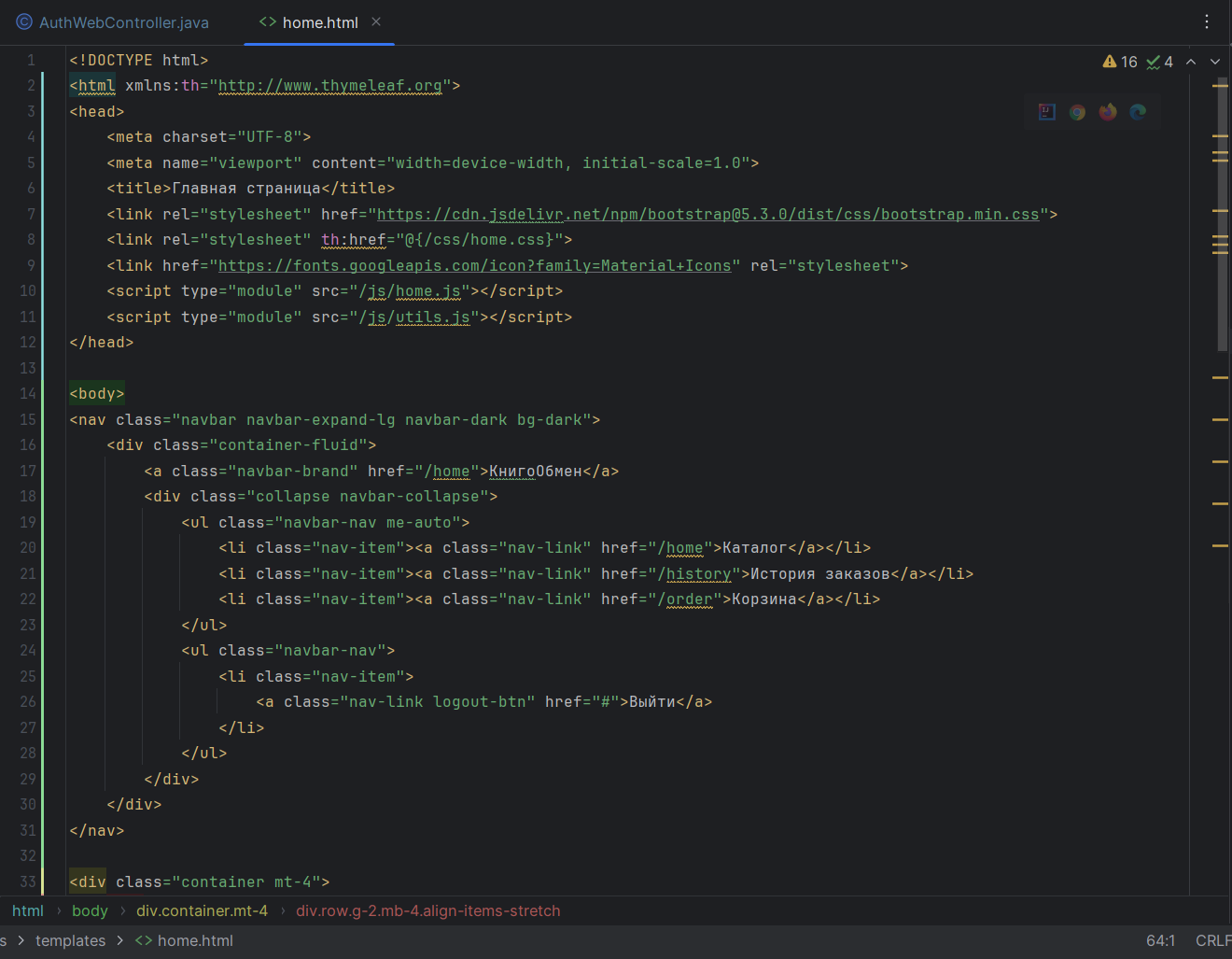
* 1. **Переход к страницам;**

AuthWebController.java:

****

* 1. **Шаблоны страницы;**

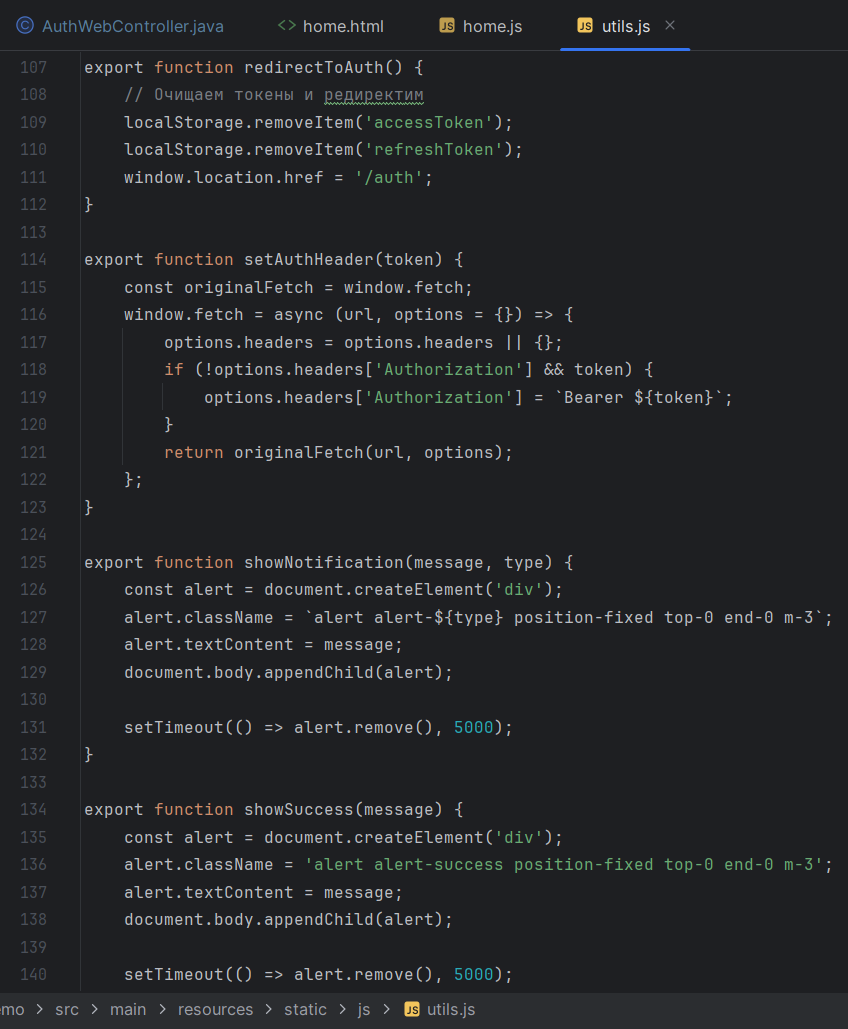
На каждой странице (кроме регистрации и авторизации) используются одинаковые заголовки и подвалы, а также в начале кода каждой страницы подключаются необходимые библиотеки и .css и .js файлы. Для составления страниц используется шаблонизатор Thymelife.



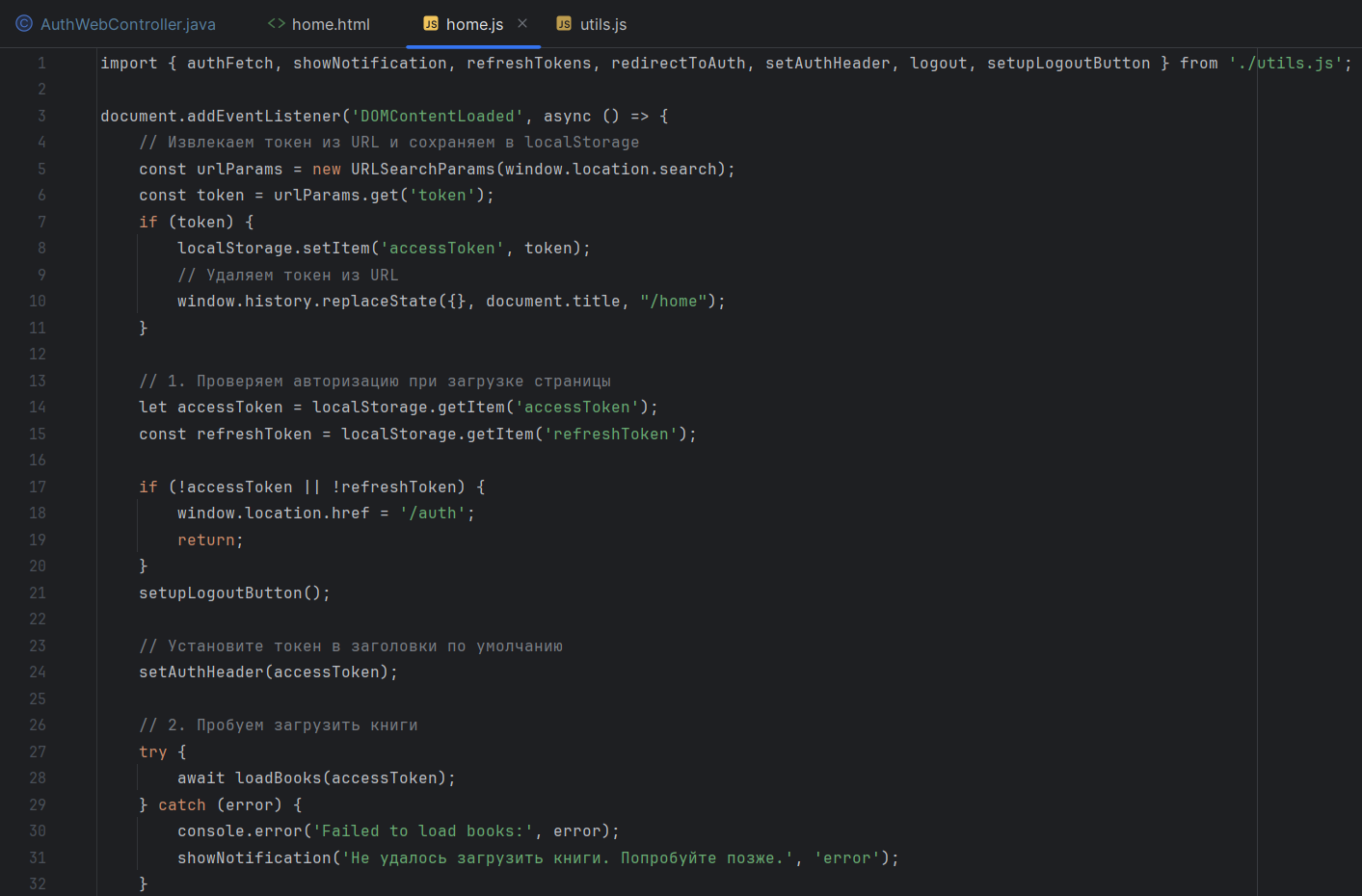
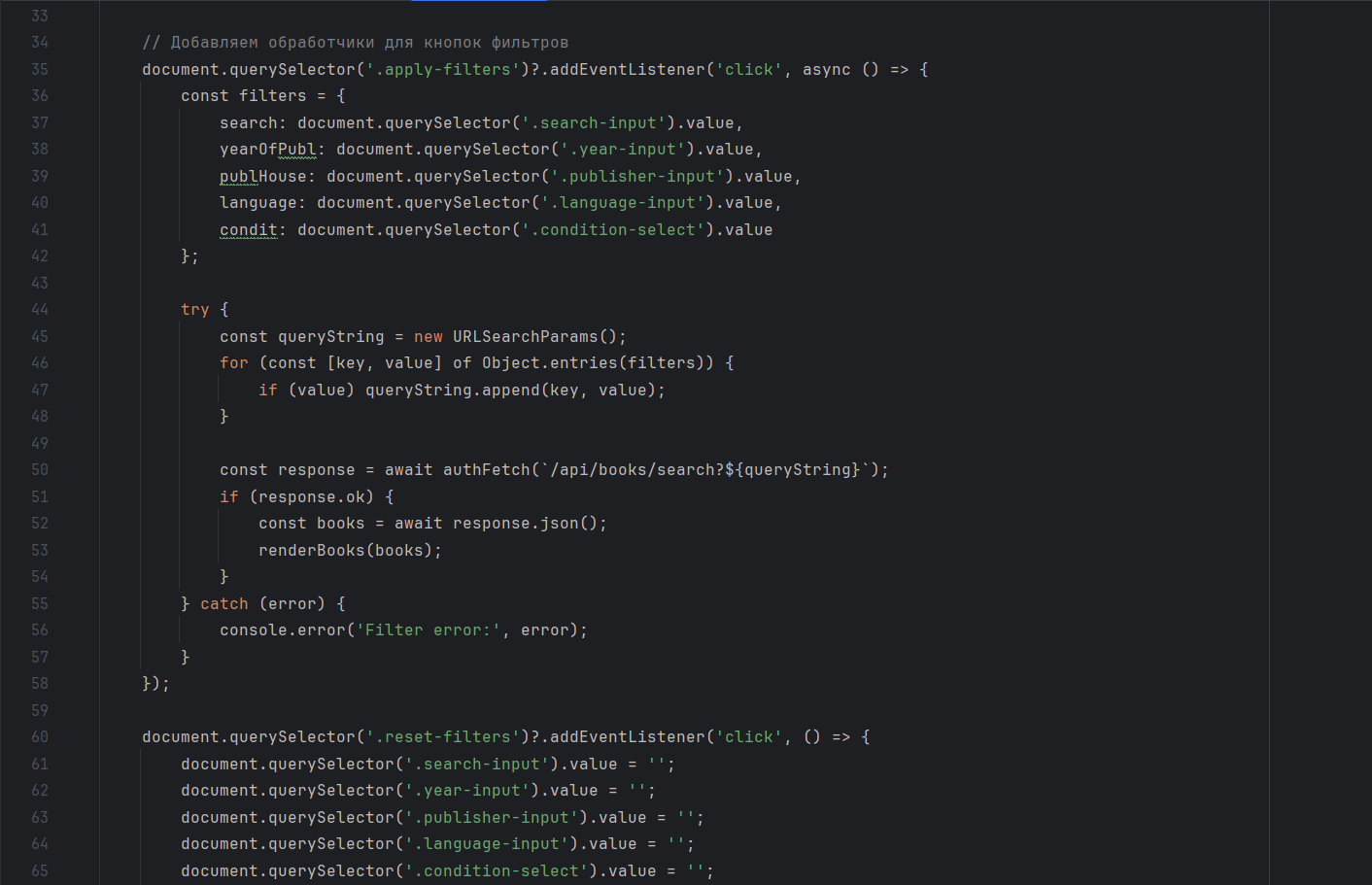
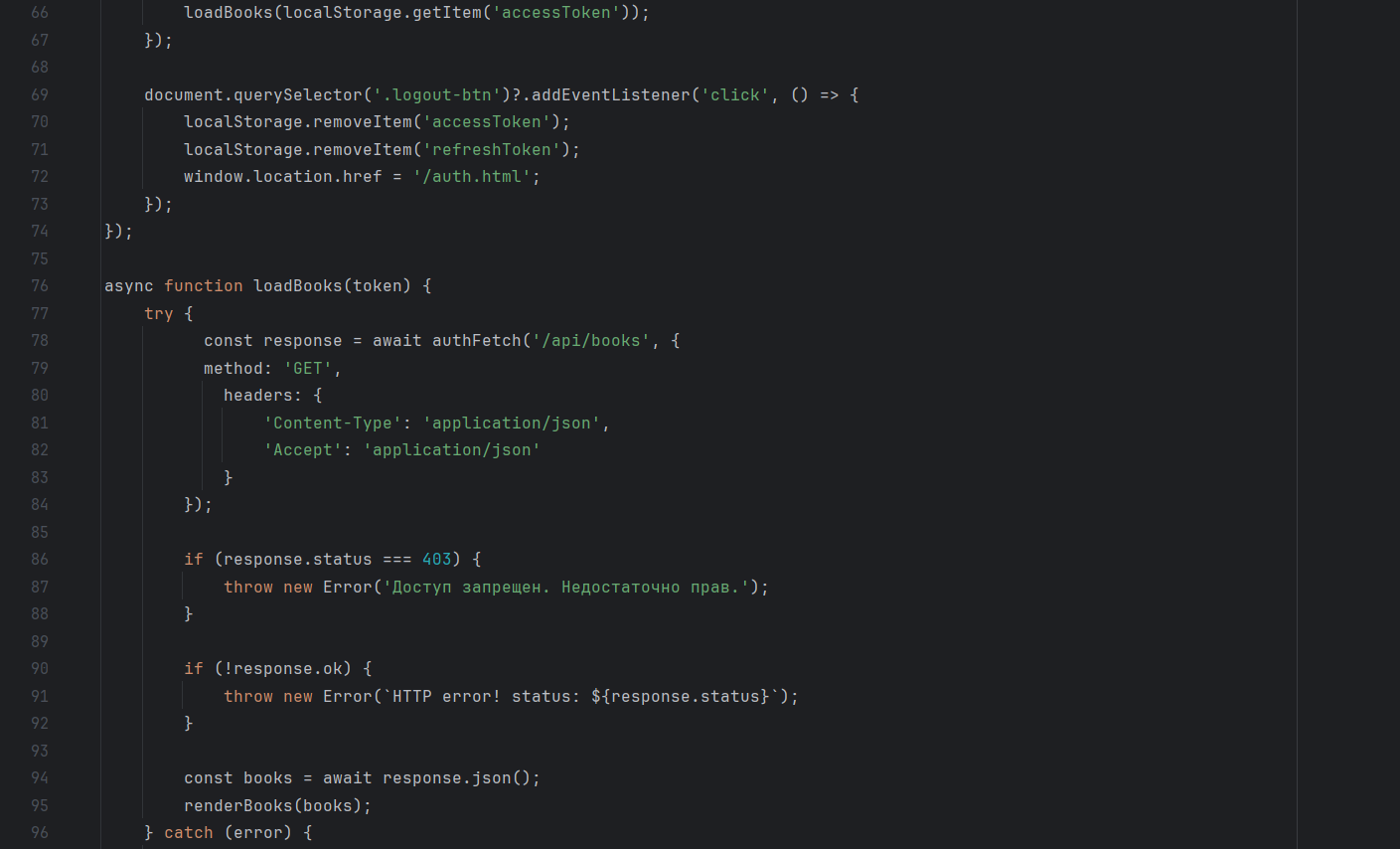


* 1. **Обработка данных.**

В каждом .js файле импортируется функции из utils.js – файла с повторяющимися для всех обобщенными функциями (такими как отправление API запроса на бэкенд, проверка и рефреш токенов и т.д.)



home.js:

…

Пример запроса на бэкенд в методе добавления книги в заказ на сайте:  


Карточки книг в каталоге и корзине, а так же список заказов динамически обрабатываются и добавляются на страницу так же в .js файлах:

