

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №2
по дисциплине «Веб-технологии»
Тема: REST-приложение управления библиотекой

Студент гр. 8383

Киреев К.А.

Преподаватель

Беляев С.А.

Санкт-Петербург

2020

Цель работы

Изучение взаимодействия клиентского приложения с серверной частью, освоение шаблонов web-страниц, формирование навыков разработки динамических HTML-страниц, освоение принципов построения приложений с насыщенным интерфейсом пользователя.

Основные теоретические сведения

CSS – язык описания внешнего вида документа, написанного с использованием языка разметки, используется как средство оформления внешнего вида HTML-страниц.

Express – это минималистичный и гибкий web-фреймворк для приложений Node.js, предоставляющий обширный набор функций для мобильных и web-приложений.

Pug – модуль, позволяющий использовать шаблоны для HTML-страниц.

REST – стиль взаимодействия компонентов распределенного приложения. В рамках лабораторной работы – браузера и сервера web-приложения. Для взаимодействия используются стандартные методы:

GET – получение записи (записей)

POST – добавление записи

PUT – обновление или добавление записи

DELETE – удаление записи

Общая формулировка задачи

Необходимо создать web-приложение управления домашней библиотекой, которое предоставляет список книг, их можно отфильтровать по признакам “в наличии”, “возврат просрочен”, есть возможность выдать книгу для чтения и вернуть книгу.

Основные требования следующие:

1. Начальное состояние библиотеки хранится в JSON-файле на сервере. Текущее состояние — в переменной в памяти сервера.
2. В качестве сервера используется Node.JS модулем express.
3. В качестве модуля управления шаблонами HTML-страниц используется pug, все web-страницы должны быть сделаны с использованием pug;
4. Предусмотрена страница для списка книг, в списке предусмотрена, фильтрация по дате возврата и признаку «в наличии», предусмотрена возможность добавления и удаления книг.
5. Предусмотрена страница для карточки книги, в которой ее можно от. редактировать (минимум: автор, звание, дата выпуска) и дать читателю вернуть в библиотеку. В карточке книги должно быть очевидно: находится ли книга в библиотеке, кто с взял (имя) и когда должен вернуть (дата).
6. Информация о читателе вводится с использованием всплывающего модульного окна
7. Оформление страниц выполнено с использованием CSS
8. Взаимодействие между браузером и web-сервером осуществляется с использованием REST.
9. Фильтрация списка книг осуществляется с использованием AJAX-запросов
10. Логика приложения реализована на языке JavaScript,

Ход работы

1. Используя среду разработки JetBrains WebStorm, были установлены все необходимые расширения.
2. Используя модуль express, был создан и настроен сервер.
3. Были созданы и настроены pug и css файлы.
4. Разработка интерфейса пользователя.
 - 1) Главная страница сайта представлена на рисунке 1

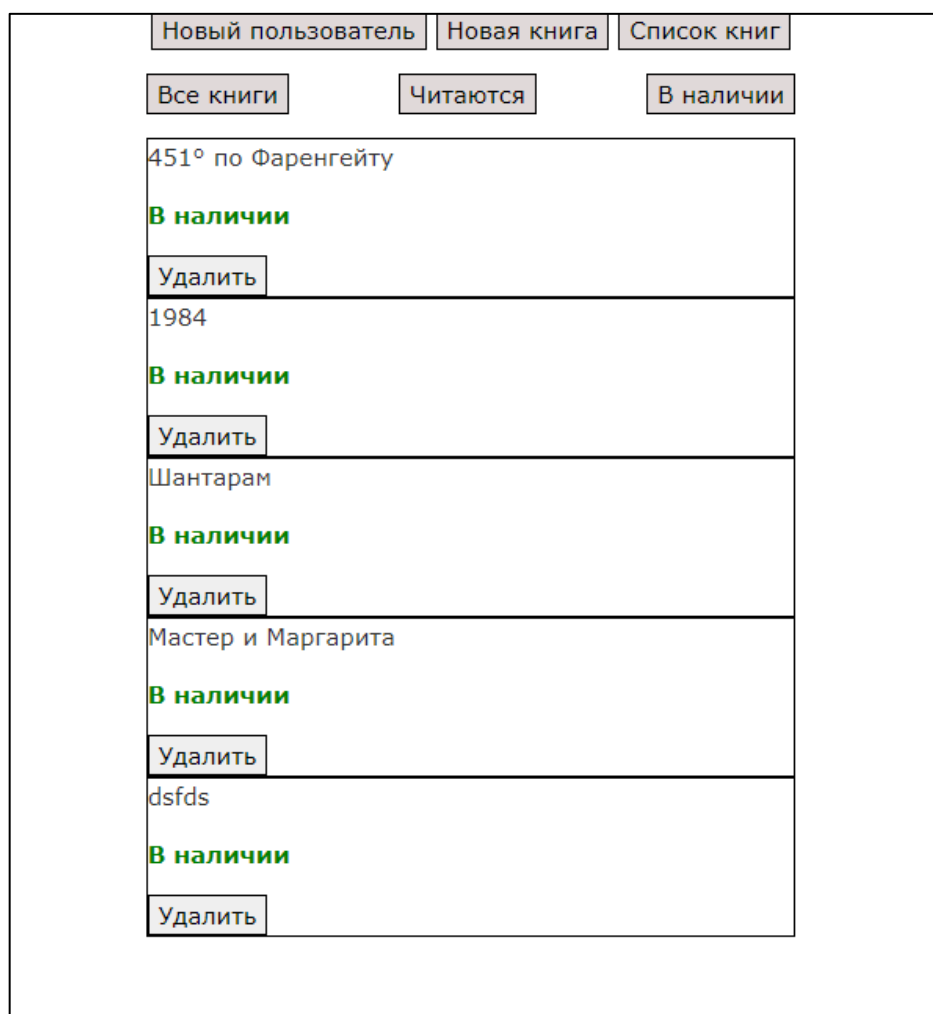
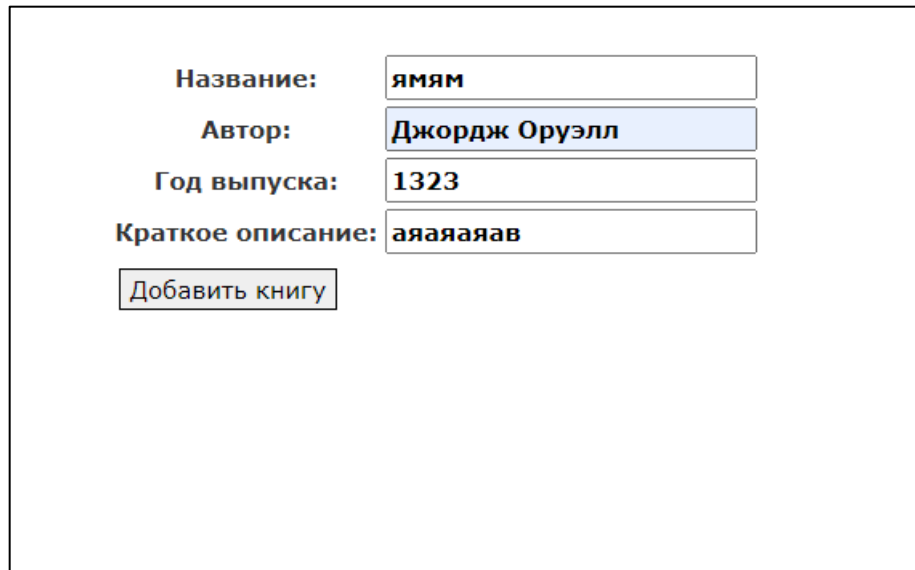


Рисунок 1 – Главная страница сайта

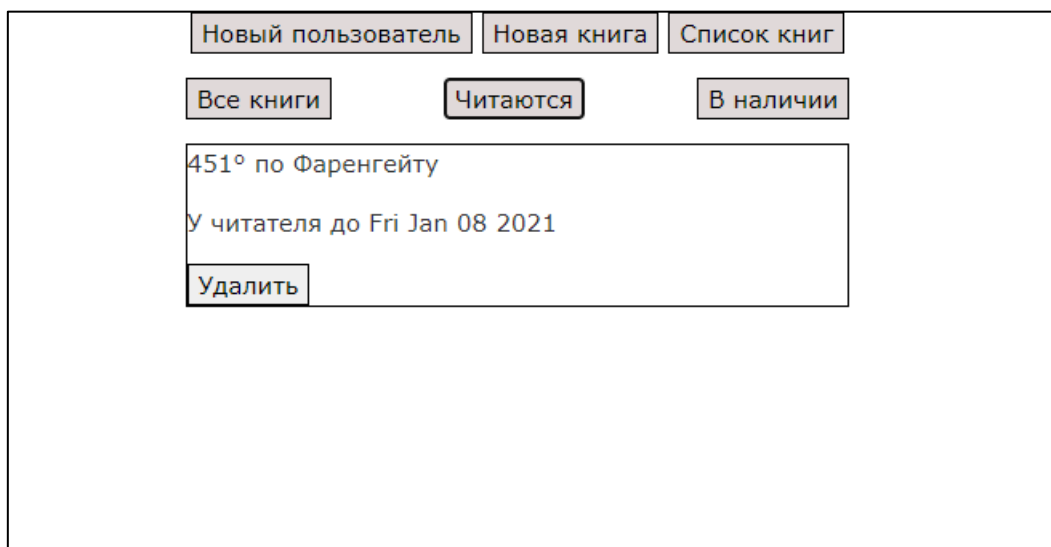
2) Окно создания книги представлено на рисунке 2.



The screenshot shows a form for creating a new book. It contains four labeled input fields: 'Название:' (Title) with the value 'ямям', 'Автор:' (Author) with the value 'Джордж Оруэлл', 'Год выпуска:' (Year of publication) with the value '1323', and 'Краткое описание:' (Brief description) with the value 'аяаяаяав'. Below these fields is a button labeled 'Добавить книгу' (Add book).

Рисунок 2 – Основная форма

3) Модальное окно представлено на рисунке 3.



The screenshot shows a modal window with a header containing three buttons: 'Новый пользователь' (New user), 'Новая книга' (New book), and 'Список книг' (List of books). Below the header are three buttons: 'Все книги' (All books), 'Читаются' (Being read), and 'В наличии' (In stock). The main content area displays the text '451° по Фаренгейту' (451° Fahrenheit), 'У читателя до Fri Jan 08 2021' (Available to reader until Fri Jan 08 2021), and a 'Удалить' (Delete) button.

Рисунок 3 – Модальное окно.

5. Для проекта были созданы следующие файлы.

- 1) form.css, Global.css, main.css, modal.css – набор стилей для оформления страниц.
- 2) editBook.pug, createBook.pug, main.pug – web-страницы хранящие в себе окна для редактирования книги, создания книги и списка книг, соответственно.

- 3) `booklist.json` – файл хранящий в себе начальные настройки книг
- 4) `server.js` – содержит функции, обеспечивающие работу сервера;
- 5) `sys.js` – содержит функции, генерирующие AJAX-запросы и обрабатывающие различные события.

Вывод

В ходе лабораторной работы был получен опыт работы с `rug`, `css` файлами, создании сервера на основе модуля `express` и генерации AJAX-запросов, на основе создания REST-приложения управления библиотекой