**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МОЭВМ**

отчет

**по лабораторной работе №2**

**по дисциплине «WEB-Технологии»**

Тема: **REST-ПРИЛОЖЕНИЕ УПРАВЛЕНИЯ БИБЛИОТЕКОЙ**

|  |  |  |
| --- | --- | --- |
| Студент гр. 6382 |  | Вайгачёв А.О. |
| Преподаватель |  | Беляев С.А. |

Санкт-Петербург

2018

**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.

**Постановка задачи**

Для достижения поставленной цели требуется решить следующие задачи:

1) Разработка интерфейса web-приложения.

2) Задание стилей для отображения web-приложения.

3) Создание web-сервера на основе express.

4) Создание шаблонов web-страниц.

5) Настройка маршрутов.

6) Создание json-хранилища.

7) Обработка REST-запросов.

*Основные требования:*

Основные требования:

1. Начальное состояние библиотеки хранится в JSON-файле на сервере. Текущее состояние – в переменной в памяти сервера.

2. В качестве сервера используется Node.JS с модулем express.

3. В качестве модуля управления шаблонами HTML-страниц используется 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 её можно отредактировать (минимум: автор, название, дата выпуска) и дать читателю или вернуть в библиотеку. В карточке книги должно быть очевидно: находится ли книга в библиотеке или кто её взял (имя) и когда должен вернуть (дата).

6. Информация о читателе вводится с использованием всплывающего модального окна.

7. Оформление страниц выполнено с использованием CSS (допустимо использование w3.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.

**Интерфейс программы**

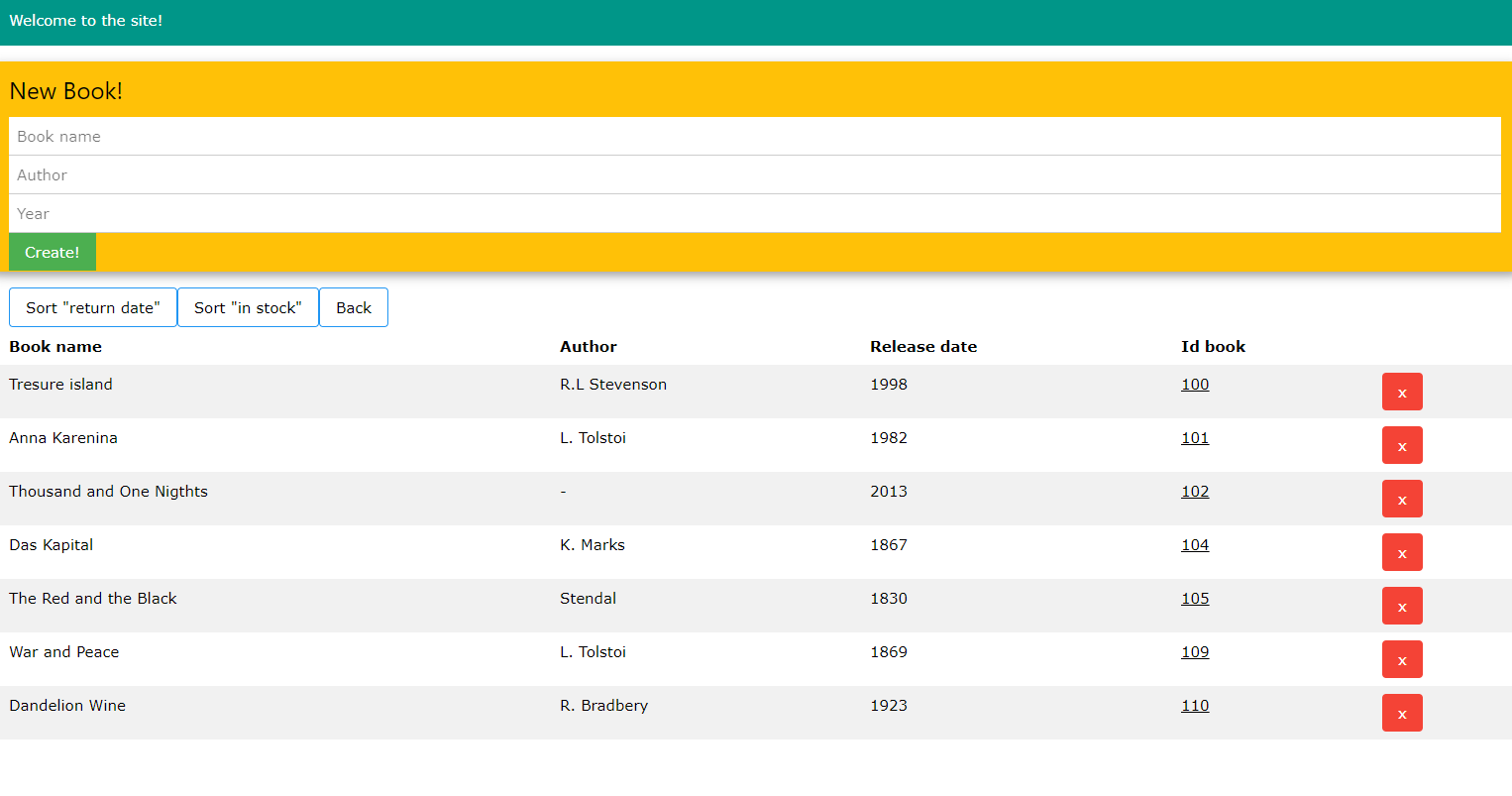


Рисунок 1 Главная страница

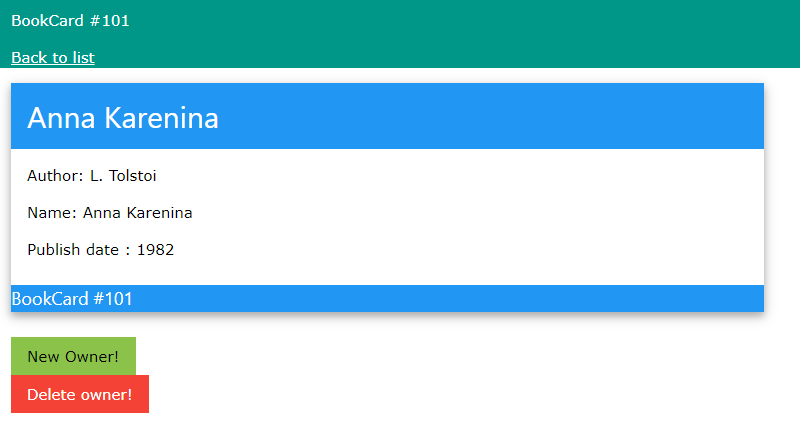


Рисунок 2 Страница книга

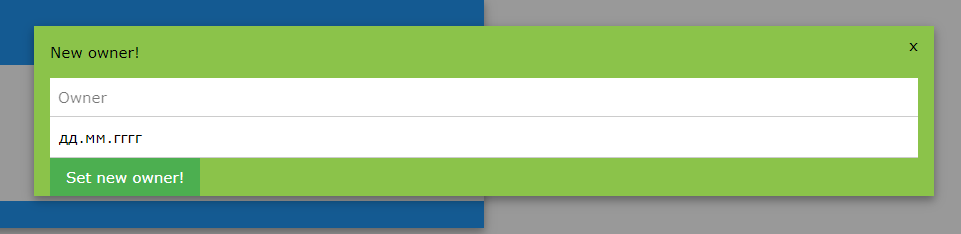


Рисунок 3 Добавление нового владельца



Рисунок 4 Информация о владельце

**Вывод**

Мы создали сайт, который обрабатывает rest-запросы с серверной частью на node.js