Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий
институт
Кафедра «Информатика»
кафедра

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №7

Поддержка архитектуры REST в Spring
тема

Преподаватель _____ А.С. Черниговский
подпись, дата инициалы, фамилия

Студент КИ18-17/1б 031830504 _____ Е.В. Железкин
номер группы, зачетной книжки подпись, дата инициалы, фамилия

Красноярск 2020

1 Цель работы

Познакомиться с механизмами поддержки архитектуры REST в Spring.

2 Задача работы

Взять практическое задание №6 (или №5, на усмотрение студента, при работе с защищенным приложением могут возникнуть трудности) и добавить следующий функционал:

1) Преобразовать веб-приложение таким образом, чтобы оно поддерживало архитектуру REST. Должны поддерживаться следующие типы запросов: GET (показ (html) и извлечение (json) всех/одной записей/сущностей), PUT (добавление), DELETE (удаление).

2) Разработать REST-клиент для вашего приложения, который, используя RestTemplate позволяет выполнять базовые операции по извлечению (GET), добавлению (PUT), удалению (DELETE) ресурсов. REST-клиент не обязан иметь пользовательский интерфейс, необходим тестовый пример, который можно запускать из консоли.

3) Обязательным условием является сохранение всего предшествующего функционала приложения. Для удовлетворения всем характеристикам RESTархитектуры приложение может быть реорганизовано (убраны GET-запросы с параметрами) или добавлен новый функционал.

4*) PUT и DELETE запросы не обязательно делать через запросы из браузера. Достаточно реализации для клиентов-приложений.

3 Ход работы

1) Веб-приложение доделано до функционала, необходимого в 7 работе (рисунки 1-):

Рисунок 1 – Обновлённый маппинг контроллера BooksController.java

```java
package ru.xorsiphus;

import org.springframework.web.client.RestTemplate;

public class RestClient
{
    public static void main( String[] args )
    {
        var param = "0";
        if (args.length != 0)
            param = args[0];
        String ret = retrieve(Integer.parseInt(param));
        System.out.println(ret);
    }

    public static String retrieve(int id)
    {
        if (id == 0)
            return new RestTemplate().getForObject(
                    url: "http://localhost:8081/books",
                    String.class);
        else
            return new RestTemplate().getForObject(
                    url: "http://localhost:8081/books/{id}",
                    String.class, id);
    }
}
```

Рисунок 2 – Использование RestTemplate

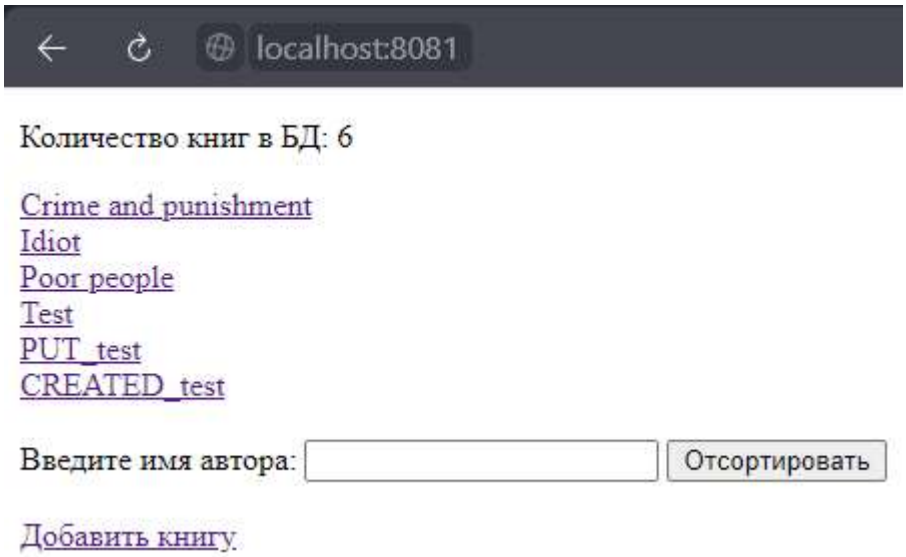

Рисунок 3 – Получение данных с помощью браузера

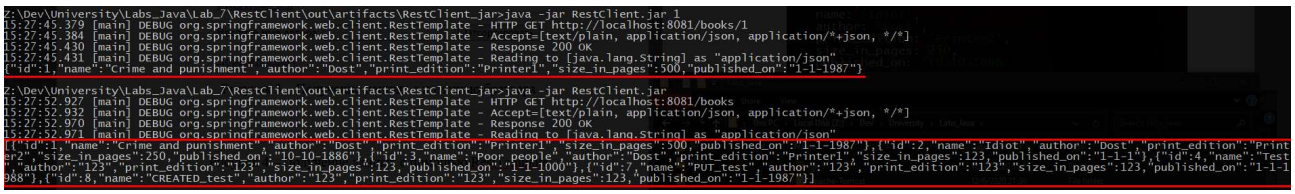

Рисунок 4 – Получение данных с помощью RestTemplate

Рисунок 5 – Получение данных с помощью кастомного Get-запроса

4 Вывод

В ходе данной лабораторной работы было проведено ознакомление с механизмами поддержки архитектуры REST в Spring.