

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение высшего образования «Уральский федеральный университет имени первого Президента России Б.Н. Ельцина» (УрФУ)

Институт фундаментального образования

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к Лабораторной работе №3

по теме: \_\_Настройка связи между REST сервисами\_

по дисциплине: \_\_\_\_\_Технологии программирования\_\_\_\_\_

Студент: \_\_\_Пряминин Т. В.\_\_\_

(ФИО)

Группа:\_\_РИЗ-120938у

Преподаватель: \_\_\_\_Архипов Н. А.\_\_\_\_\_\_\_

(ФИО)

Екатеринбург

2023

**ПОСТАНОВКА ЗАДАЧИ**

Доработать простой REST сервис и настроить связь между двумя сервисами. Необходимо создать новый сервис, скопировав сервис разработанный в ЛР №1 и ЛР №2 и поменяв его имя, новый сервис будет принимать request от сервиса разработанного в ЛР 1 и ЛР 2, у нового сервиса нужно поменять порт на котором он будет работать, т.к. одновременно два сервиса на порту 8080 работать не смогут. Для этого в новом проекте необходимо в файле application.properties добавить настройку порта, записав **server.port**=**8082**

**РЕЗУЛЬТАТ РАБОТЫ ПРОГРАММЫ**

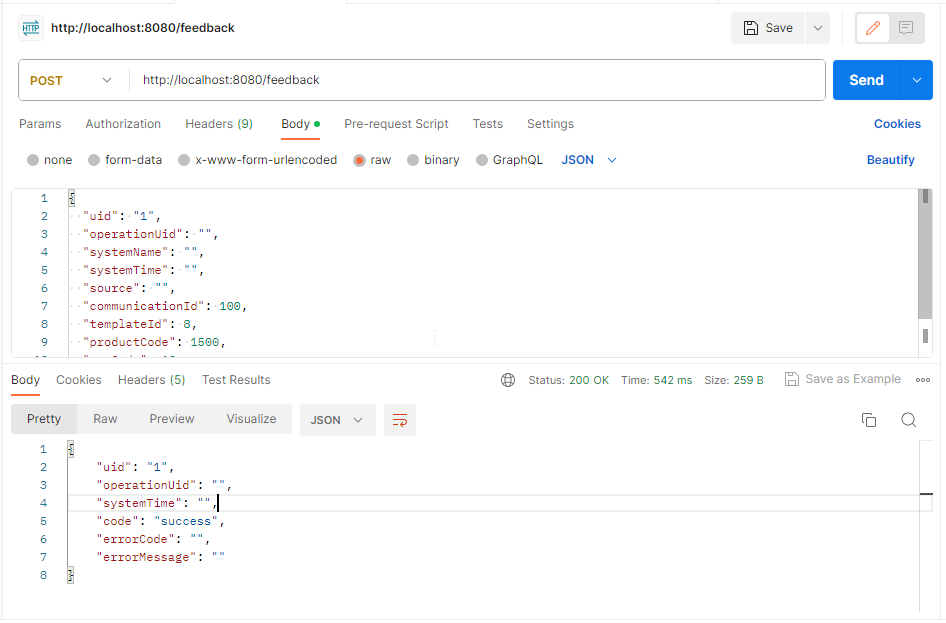


Рисунок 1 – Результат работы сервисов

**Первое приложение**

ПОЛУЧЕН ЗАПРОС: Request(uid=1, operationUid=, systemName=, systemTime=, source=, communicationId=100, templateId=8, productCode=1500, smsCode=10)

МОДИФИЦИРОВАННЫЙ: Responce(uid=1, operationUid=, systemTime=, code=success, errorCode=, errorMessage=ERROR!!!)

**Второе приложение**

ПОЛУЧЕН ЗАПРОС: Request(uid=1, operationUid=, systemName=, systemTime=testing time, source=, communicationId=100, templateId=8, productCode=1500, smsCode=10)

МОДИФИЦИРОВАННЫЙ: Responce(uid=1, operationUid=, systemTime=, code=success, errorCode=, errorMessage=ERROR!!!)

**ОПИСАНИЕ ПРОГРАММЫ**

**Интерфейс ModifyRequestService**

public interface ModifyRequestService {  
 void modifyRequest(Request request);  
}

**Класс ModifyRequestSystemTime**

@Service  
public class ModifyRequestSystemTime implements ModifyRequestService {  
 @Override  
 public void modifyRequest(Request request){  
 request.setSystemTime("testing time");  
  
 HttpEntity<Request> httpEntity = new HttpEntity<>(request);  
  
 new RestTemplate().exchange("http://localhost:8080/feedback",  
 HttpMethod.*POST*,  
 httpEntity,  
 new ParameterizedTypeReference<> (){});  
  
 }  
}

**Класс MyController**

@Slf4j  
@RestController  
public class MyController {  
 private final MyModifyService myModifyService;  
 private final ModifyRequestService modifyRequestService;  
  
 @Autowired  
 public MyController(@Qualifier("ModifyErrorMessage") MyModifyService myModifyService, ModifyRequestService modifyRequestService) {  
 this.myModifyService = myModifyService;  
 this.modifyRequestService = modifyRequestService;  
 }  
  
 @PostMapping(value = "/feedback")  
 public ResponseEntity<Responce> feedback(@RequestBody Request request) {  
 *log*.info("ПОЛУЧЕН ЗАПРОС: " + String.*valueOf*(request));  
  
 Responce response = Responce.*builder*()  
 .uid(request.getUid())  
 .operationUid(request.getOperationUid())  
 .systemTime(request.getSystemName())  
 .code("success")  
 .errorCode("")  
 .errorMessage("")  
 .build();  
  
 modifyRequestService.modifyRequest(request);  
  
 Responce modifyedResponce = myModifyService.modify(response);  
 *log*.info("МОДИФИЦИРОВАННЫЙ: " + String.*valueOf*(response));  
  
 return new ResponseEntity<>(response, HttpStatus.*OK*);  
 }  
}

**ВЫВОД**

В ходе выполнения данной лабораторной работы, был добавлен новый REST сервис и настроено взаимодействие между приложениями.

Ссылка на github с выполненными заданиями: *https://github.com/Prodiks/spring\_labs*