

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САМАРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА С.П. КОРОЛЕВА  
(САМАРСКИЙ УНИВЕРСИТЕТ)»

Институт Информатики и кибернетики   
Кафедра Программных систем

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**  
  
к лабораторной работе №3 по дисциплине «Разработка микросервисных приложений»

Обучающийся группы 6231-020302D \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гижевская В.Д.

Обучающийся группы 6231-020302D \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Алкеев М.Г.

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Лобанков А.А.

Самара 2023

СОДЕРЖАНИЕ

[1 Постановка задачи 3](#_Toc146727791)

[2 Результаты работы 4](#_Toc146727792)

[ПРИЛОЖЕНИЕ А Листинг программы](#_Toc146727796) 6

1. Постановка задачи

В процессе выполнения заданий подключить службу обнаружения микросервисов и настроить api шлюз.

1. Создать новое Spring Boot приложение – Discovery server. В качестве службы обнаружения микросервисов можно использовать eureka.
2. Зарегистрируйте ваши микросервисы (user и company) в созданном Discovery server.
3. Создайте новое Spring boot приложение – Api Gateway. Зарегистрируйте его в Discovery server
4. Задайте маршруты в Api gateway таким образом, чтобы все запросы, которые идут по адресу …/user/… были перенаправлены в user-service, а все запросы по адресу …/company/… в company-service. Проверьте работоспособность приложения.
5. Поправьте url в своих feign clients, таким образом, чтобы вызов сервисов осуществлялся через api gateway. Проверьте работоспособность приложения.
6. Результаты работы

На рисунках 1- представлены результаты работы.

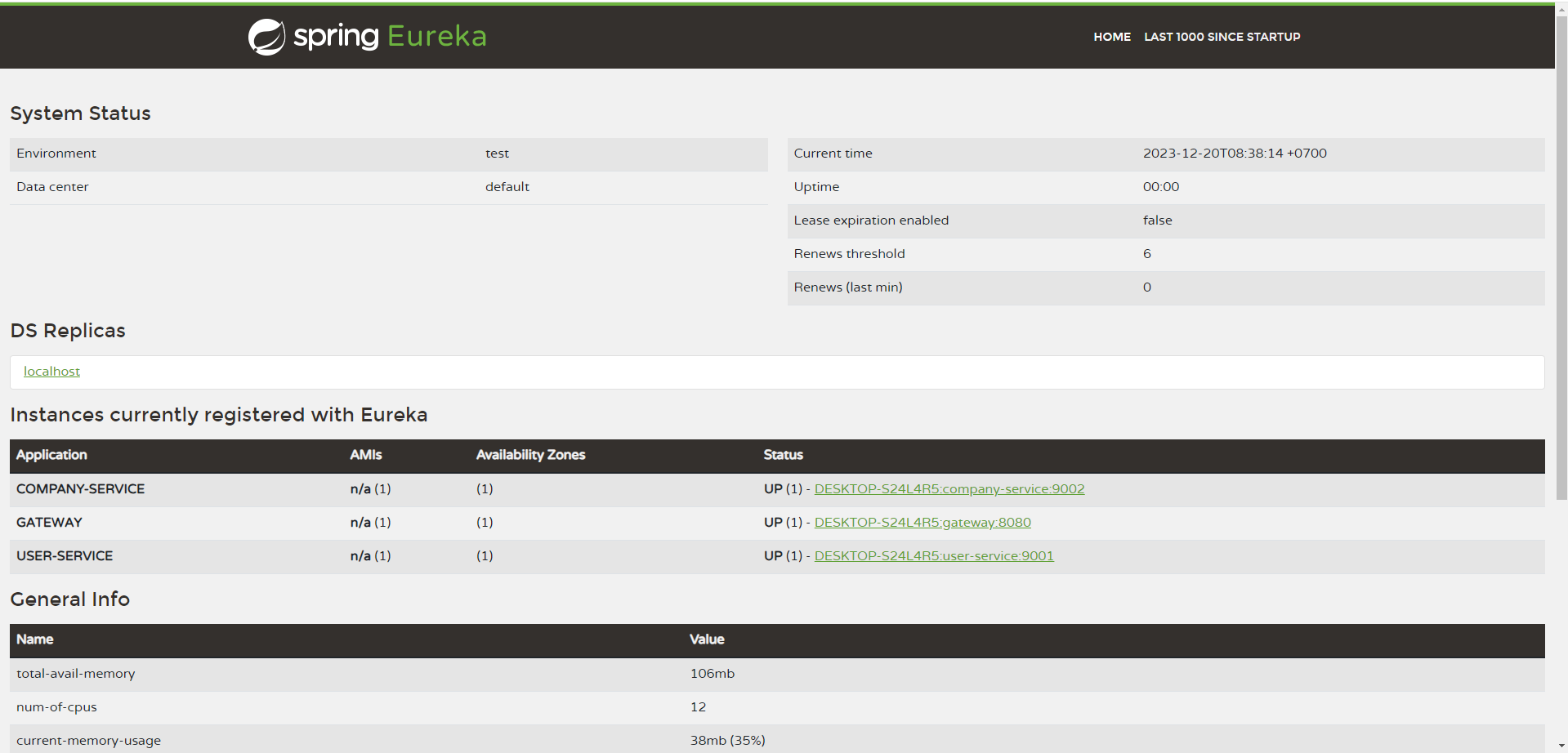


Рисунок 1 – Eureka со списком зарегистрированных сервисов

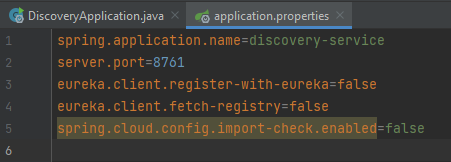


Рисунок 2 – конфиг discovery server



Рисунок 3 – конфиг gateway server

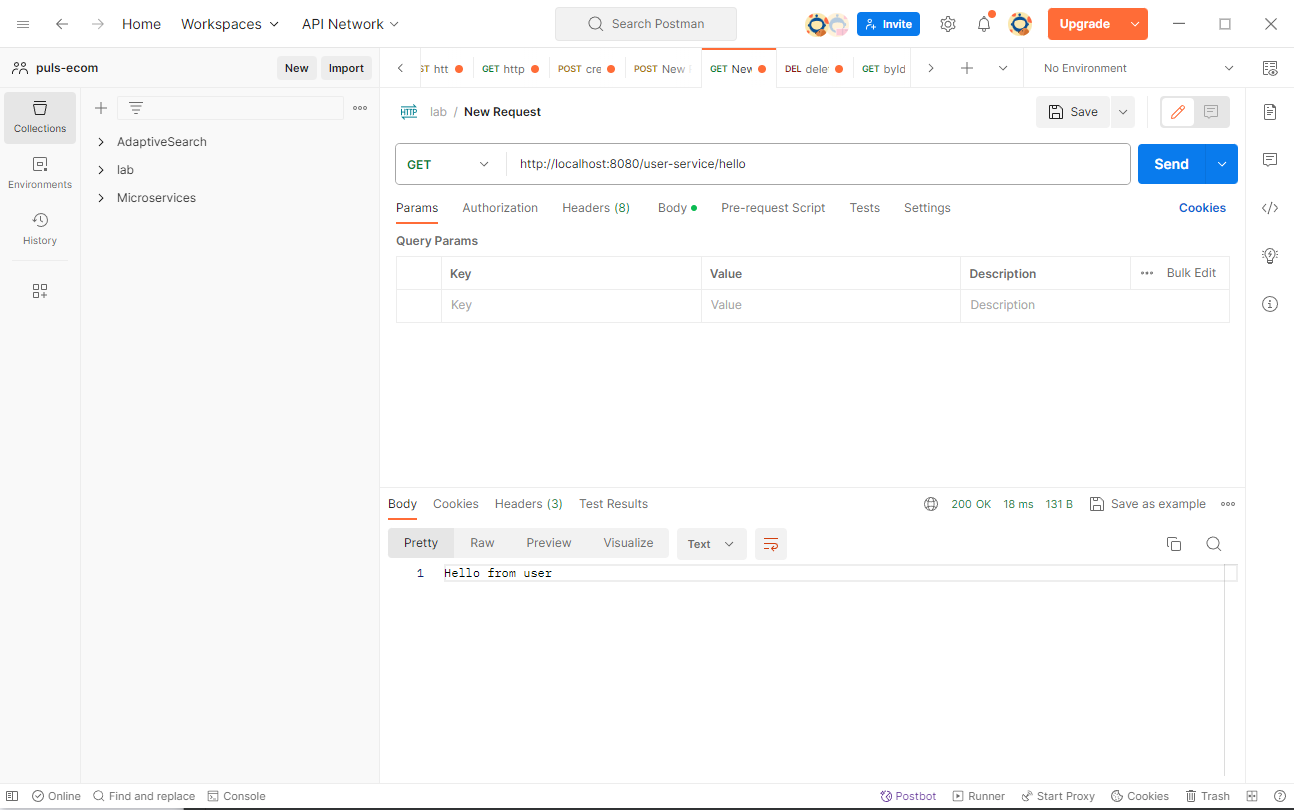


Рисунок 4 – запрос в user-service

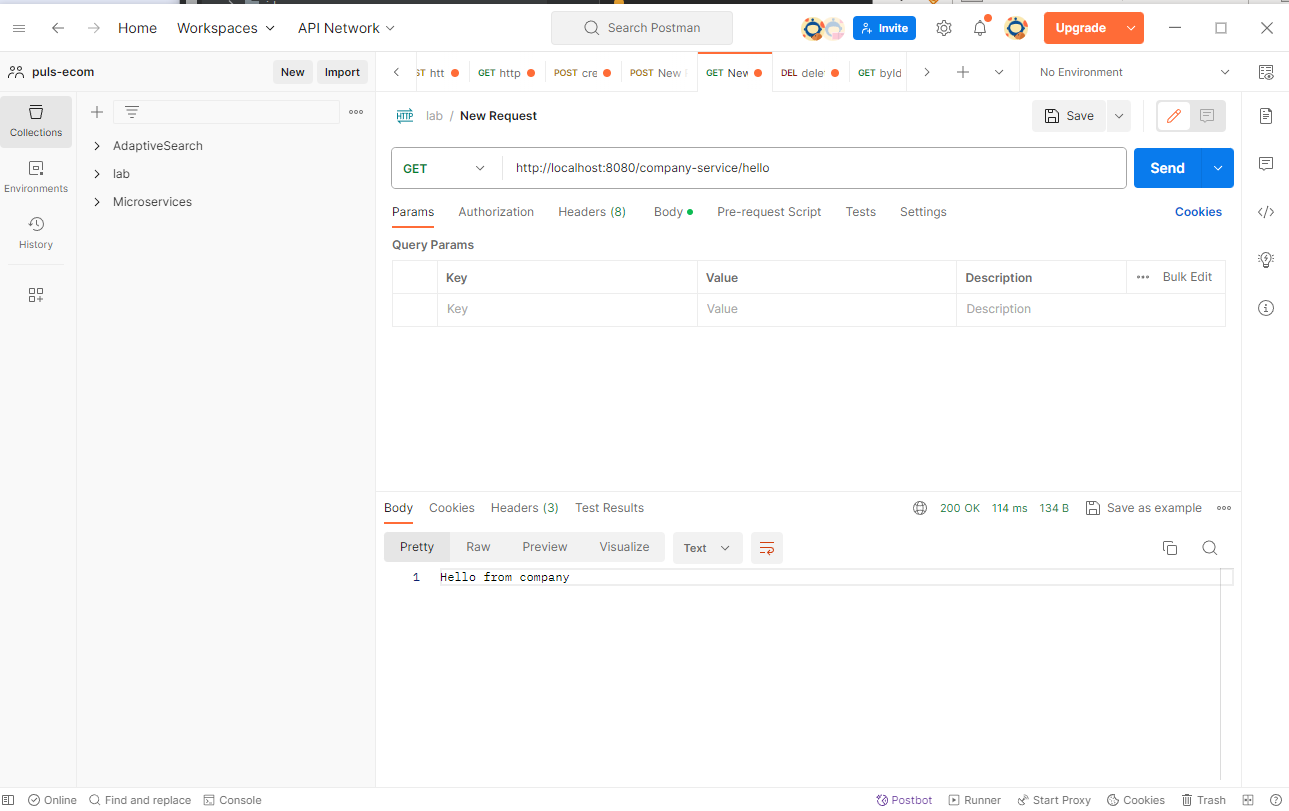


Рисунок 5 – запрос в company-service

ПРИЛОЖЕНИЕ А  
Листинг программы

spring:

application:

name: gateway

##GATEWAY CONFIGURATIONS

cloud:

gateway:

routes:

## user-service

- id: user-service

uri: lb://user-service

predicates:

- Path=/user-service/

filters:

- StripPrefix=1

## company-service

- id: company-service

uri: lb://company-service

predicates:

- Path=/company-service/

filters:

- StripPrefix=1

server:

port: 8080

eureka:

client:

service-url:

defaultZone: http://localhost:8761/eureka

package com.lab.gateway;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.cloud.netflix.eureka.EnableEurekaClient;

@SpringBootApplication

@EnableEurekaClient

public class GatewayApplication {

public static void main(String[] args) {

SpringApplication.run(GatewayApplication.class, args);

}

}

eureka:

client:

serviceUrl:

defaultZone: http://localhost:8761/eureka

package com.lab.company;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

@RequestMapping("/")

public class CompanyController {

@GetMapping(name = "/hello",value = "/hello")

public String sayHello() {

return "Hello from company";

}

}

package com.lab.company;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.cloud.netflix.eureka.EnableEurekaClient;

@SpringBootApplication

@EnableEurekaClient

public class CompanyApplication {

public static void main(String[] args) {

SpringApplication.run(CompanyApplication.class, args);

}

}

package com.lab.user;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class UserApplication {

public static void main(String[] args) {

SpringApplication.run(UserApplication.class, args);

}

}

package com.lab.user;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

@RequestMapping("/")

public class UserController {

@GetMapping(name = "/hello",value = "/hello")

public String sayHello() {

return "Hello from user";

}

}

eureka:

client:

serviceUrl:

defaultZone: http://localhost:8761/eureka/