

Microservicios – Arquitectura y Desarrollo

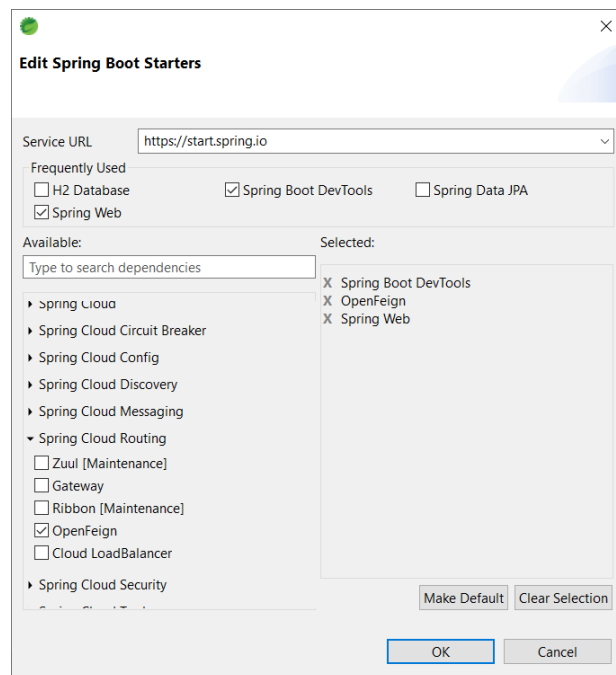
LAB Cliente REST con Netflix Feign

Objetivos

- Mostrar al participante el procedimiento para el desarrollo de cliente Rest de Microservicios basado en Netflix Feign ahora llamado OpenFeign

Procedimiento

1. Edita el proyecto **springboot-servicio-items** y agrega las dependencias de **OpenFeign**



2. Agrega la anotación `@EnableFeignClients` a la clase **SpringbootServicioItemsApplication**

```
package com.cognos.app.items;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.openfeign.EnableFeignClients;

@EnableFeignClients
```

```

@SpringBootApplication
public class SpringbootServicioItemsApplication {

    public static void main(String[] args) {
        SpringApplication.run(SpringbootServicioItemsApplication.class,
args);
    }

}

```

3. Crea el paquete **com.cognos.app.items.cliente** y la interface **ProductoClienteRest**

```

package com.cognos.app.items.cliente;

import java.util.List;

import org.springframework.cloud.openfeign.FeignClient;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;

import com.cognos.app.items.model.Producto;

@FeignClient(name="servicio-productos",url="http://localhost:8081")
public interface ProductoClienteRest {

    @GetMapping("/listar")
    public List<Producto> listar();

    @GetMapping("/mostrar/{id}")
    public Producto mostrar(@PathVariable Long id);

}

```

4. Crea la clase **ItemServiceFeign** que implemente la interface **ItemService**

```

package com.cognos.app.items.model.service;

import java.util.List;
import java.util.stream.Collectors;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.annotation.Primary;
import org.springframework.stereotype.Service;

import com.cognos.app.items.cliente.ProductoClienteRest;
import com.cognos.app.items.model.Item;

@Service
@Primary
public class ItemServiceFeign implements ItemService {

    @Autowired

```

```

        private ProductoClienteRest clienteFeigt;

        @Override
        public List<Item> findAll() {
            return clienteFeigt.listar().stream().map(p -> new
Item(p,1)).collect(Collectors.toList());
        }

        @Override
        public Item findById(Long id, Integer cantidad) {
            return new Item(clienteFeigt.mostrar(id),cantidad);
        }
    }
}

```

5. Modifica la clase **ItemController** para que use la implementación de cliente **ItemServiceFeign**

```

package com.cognos.app.items.controllers;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.beans.factory.annotation.Qualifier;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RestController;

import com.cognos.app.items.model.Item;
import com.cognos.app.items.model.service.ItemService;

@RestController

public class ItemController {

    @Autowired

    @Qualifier("itemServiceFeign")

    private ItemService itemService;

```

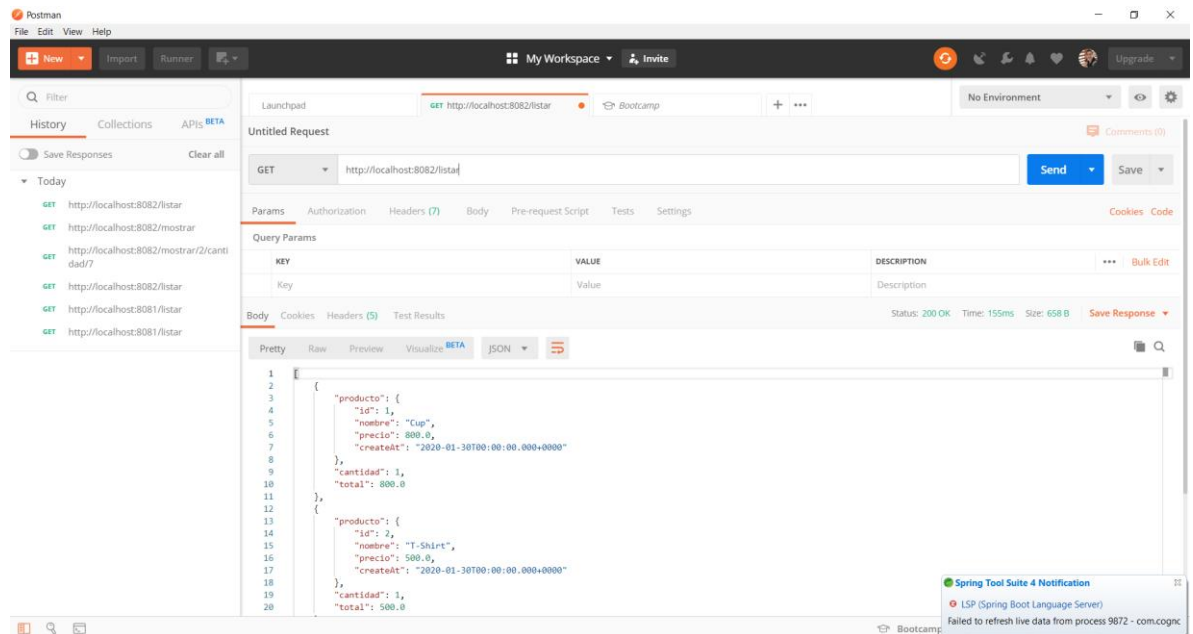
```
@GetMapping("/listar")
```

```
public List<Item> listar(){  
    return itemService.findAll();  
}
```

```
@GetMapping("/mostrar/{id}/cantidad/{cantidad}")
```

```
public Item detalle(@PathVariable Long id,@PathVariable Integer cantidad) {  
    return itemService.findbyId(id, cantidad);  
}  
}
```

6. Guarda los cambios, reinicia los dos servicios y prueba la aplicación.



7. (Opcional) Realiza los cambios necesarios para usar en el controlador la implementación **ItemServiceImp** (implementación REST)