**Imagen que contiene Logotipo

Descripción generada automáticamente**

**Docente Hector Manuel Vanegas Solis**

**Materia: Diseño de servicios computacionales**

**Parcial uno**

**12 de marzo de 2024**

**==================================================================**

Una de las soluciones más usadas en el desarrollo de aplicaciones empresariales, es usar microservicios. Esta decisión trae consigo múltiples desafíos. Para contrarrestar estos desafíos, apareció recientemente un concepto muy importante , conocido con el nombre de Patrón Saga.

El patrón saga consiste básicamente en una serie de eventos emitidos por varios servicios que se requieren para completar una transacción.

El patrón saga se subdivide en dos tipos de patrones, uno conocido como Coreografía y el otro como Orquestación.

En la coreografía cada servicio emite una respuesta ante un evento externo, actuando de forma autónoma. Entre tanto una orquestación consiste en tener un servicio centralizado que sea el encargado de llamar a los otros servicios para responder a una transacción.

Hasta ahora hemos trabajado con servicios donde cada uno responde de forma autónoma (coreografía).

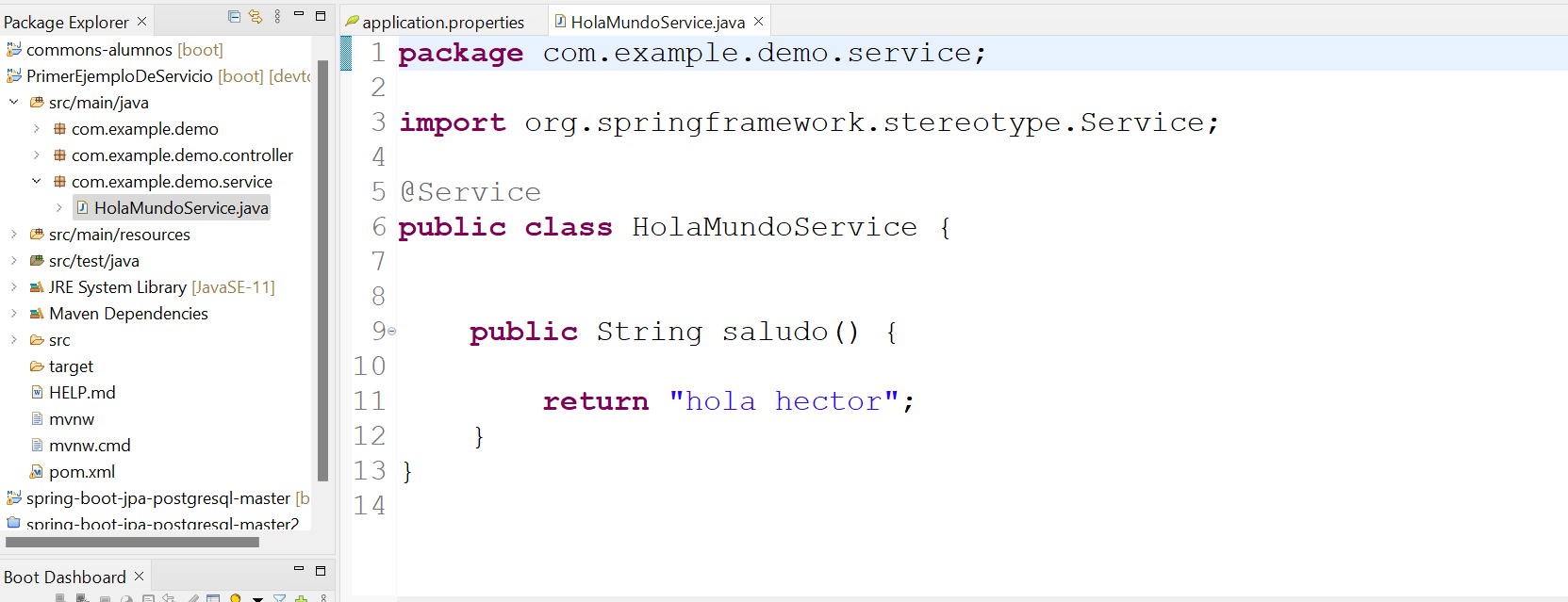
Ahora el reto es crear un orquestador que gestione la respuesta de cada uno de los servicios y los una para devolver una sola respuesta a toda la transacción.

1. Para ello usted deberá construir otro servicio más, tal y como ha construido los anteriores servicios.
2. En la capa de servicio del servicio construido en el punto anterior, deberá colocar dentro de un método la lógica necesaria para llamar a cada uno de los servicios, haciendo uso objetos de la clase RestTemplate , tal y como se muestra en el siguiente fragmento de código:

**RestTemplate** restTemplate=**new RestTemplate**(); **String** fooResourceUrl="http://localhost:8080/spring-rest/foos"; ResponseEntity<String> response =

restTemplate.getForEntity(fooResourceUrl + "/1", String.class);

Este sería el ejemplo de lo que debería llevar la clase servicio donde se llamara a los servicios de la Saga



Tener muy en cuenta la estructura del paquete que usted debe crear y que se muestra en la parte izquierda de la imagen anterior.

1. Le toca crear un objeto ResTemplate por cada llamado a cada uno de los servicios y luego cuando los 3 objetos hayan recibido la respuesta de cada servicio, deberá usted unir las respuestas en una sola, para que, esta sea obtenida desde un método de tipo GET expuesto en el controlador, para que, cuando este método sea invocado desde postman, devuelva la frase completa y en orden, que resuelva el enigma puesto en la clase anterior.
2. Deberá subir el código de todos los servicios construidos (4 en total), a su repositorio de git hub, deberá adjuntar en la entrega del parcial, la evidencia de la imagen en postman con la respuesta del servicio orquestador, donde se vea la frase como respuesta y además adjuntará el link del repositorio, para validar el código.
3. Puede usar la implementación que usted considere y además puede buscar en internet la implementación que usted crea cumpla con lo solicitado en los puntos atrás

**¡ EXITOS ¡**