**Caso de Negocio**

Un ecommerce en las que están envueltos clientes, productos y el carrito de compra del cliente.

**Microservicio: ecommerce-customers-mongodb-ms (clientes)**

Subdominio: Gestión de información relacionada con clientes.

Bounded Context: Aquí se maneja toda la lógica empresarial relacionada con los clientes, como la gestión de cuentas, información personal, autenticación, autorización, y cualquier otra lógica centrada en la gestión de clientes.

SGBD: MongoDB

BD: ecommerce

Tablas:

|  |  |
| --- | --- |
| Customer | |
| id | (PK) Identificador único del cliente. No es autogenerado. |
| first\_name | Nombre del cliente. |
| last\_name | Apellido del cliente. |
| email | Dirección de correo electrónico del cliente. |
| address | Dirección del cliente. |
| phoneNumber | Número de teléfono del cliente. |
| address | Dirección del cliente. |
| dateOfBirth | Fecha de nacimiento del cliente. |
| gender | Sexo del cliente. |
| registrationDateTime | Fecha y hora del registro del cliente. |

**Microservicio: ecommerce-products-h2-ms (productos)**

Subdominio: Gestión de información relacionada con productos.

Bounded Context: Enfocado en la gestión de productos, incluyendo la información de los productos, categorías, inventario, precios, y cualquier lógica específica relacionada con la gestión y mantenimiento de productos.

SGBD: H2

BD: ecommerce

Tablas:

|  |  |
| --- | --- |
| Product | |
| id | (PK) Identificador único del producto. No es autogenerado. |
| name | Nombre del producto. |
| description | Descripción del producto. |
| price | Precio actual del producto. |
| stock | Cantidad disponible en el inventario. |

**Microservicio: ecommerce-shoppingcart-mysql-ms (carrito de compra)**

Subdominio: Gestión de carritos de compra y artículos en el carrito

Bounded Context: Este microservicio se centra en la gestión de carritos de compra, artículos dentro de los carritos, procesos de compra, cálculos de precios, descuentos, gestión de pedidos, y todo lo relacionado con las interacciones del usuario para la compra de productos.

SGBD: MySql

BD: ecommerce\_core

Tablas:

|  |  |
| --- | --- |
| Cart | |
| cart\_id | (PK) Identificador único del carrito. Es autogenerado. |
| customer\_id | Referencia al cliente que posee este carrito. No es FK pues las tablas están en distintas BD. |
| created\_at | Fecha y hora de creación del carrito. |
| confirmedAt | Fecha y hora de confirmación o checkout del carrito. |
| status | Estado del carrito (IN\_PROCESS, CANCELLED, COMPLETED). |
| total\_amount | Monto total de la compra en el carrito. |

|  |  |
| --- | --- |
| CartItem | |
| item\_id | (PK) Identificador único del item en el carrito. Es autogenerado. |
| cart\_id | (FK) Referencia al carrito al que pertenece el item. |
| product\_id | Referencia al producto que está en el item. No es FK pues las tablas estan en distintas BD. |
| quantity | Cantidad del producto en el item. |
| price | Precio del producto en el momento de agregarlo al carrito. |

**Diagrama de Microservicios y Bases de Datos**

**Diagrama

Descripción generada automáticamente**

**Procedimiento para armar el ambiente**

1. BD

1.1. MongoDB:

Crear la BD: ecommerce y el Collection: customer.

1.2. H2: El proyecto creará la base de datos de manera automática.

1.3. MySql:

Crear la BD: ecommerce\_core

1. Microservicios

2.1. Levantar los 4 proyectos con IntelliJ IDEA:

* + - ecommerce-customers-mongodb-ms
    - ecommerce-eureka-server
    - ecommerce-products-h2-ms
    - ecommerce-shoppingcart-mysql-ms

2.2. Ejecutar el proyecto: ecommerce-eureka-server

2.3. Ejecutar el proyecto: ecommerce-customers-mongodb-ms

2.4. Ejecutar el proyecto: ecommerce-products-h2-ms

2.5. Ir a la opción Run –> Edit Configurations.

Volver e ejecutar el proyecto ecommerce-products-h2-ms, agregando la opción “Allow multiple instances” y agregando en “Add VM options” lo siguiente:

**-Dserver.port=8082**

2.6. Ejecutar el proyecto: ecommerce-shoppingcart-mysql-ms

2.7. En una de las BD de H2, **ecommerce**, ejecutar los siguientes updates:

Por ejemplo, entramos a una de ellas con la URL: <http://localhost:8082/h2-console/>

(usuario: root, contraseña en blanco)

update product set description='laptop', price=100.10, stock=90 where id=100;

update product set description='mouse', price=200.10, stock=80 where id=101;

update product set description='teclado', price=300.10, stock=70 where id=102;

update product set description='televisor', price=400.10, stock=60 where id=103;

update product set description='refrigeradora', price=500.10, stock=50 where id=104;

update product set description='celular', price=600.10, stock=40 where id=105;

update product set description='iphone', price=700.10, stock=30 where id=106;

update product set description='nintendo', price=800.10, stock=20 where id=107;

update product set description='playstation', price=900.10, stock=10 where id=108;

Esto lo hacemos para efectos de ver como se hace el balanceo de peticiones sobre las 2 instancias del microservicio ecommerce-products-h2-ms.

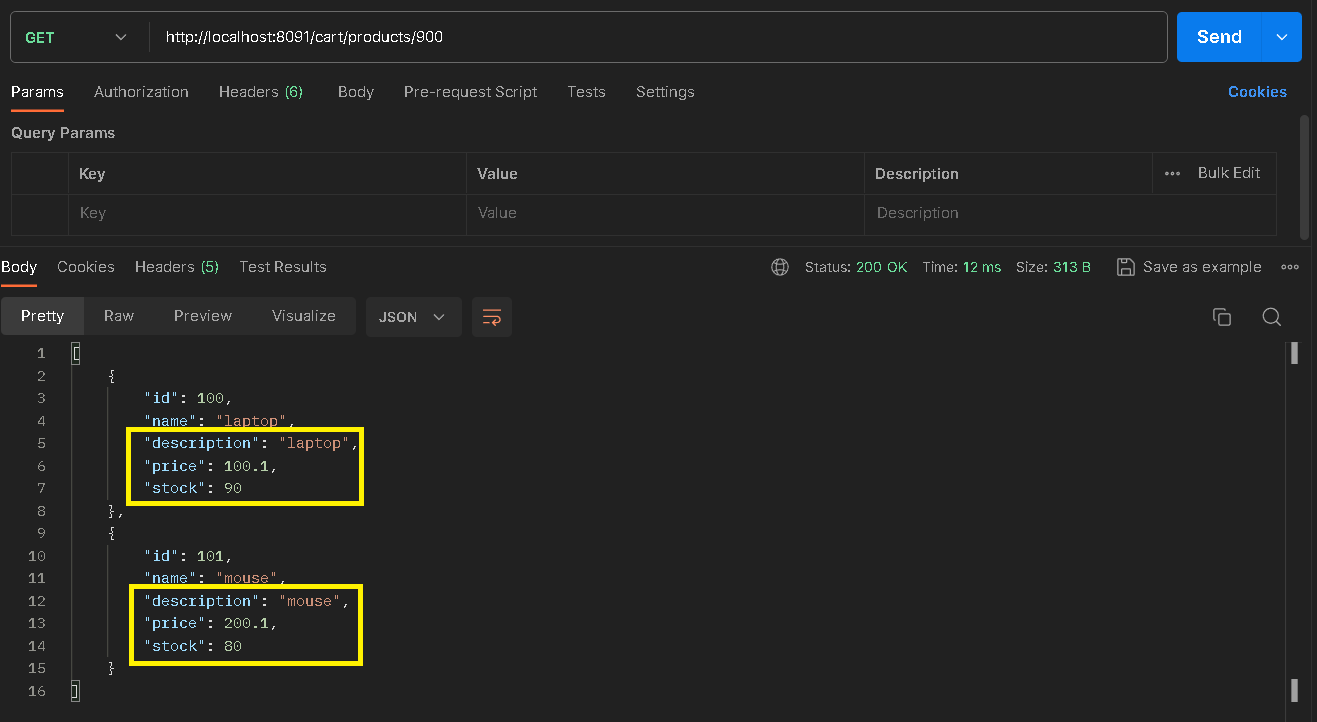
**Validación de microservicios registrados en Eureka Server**

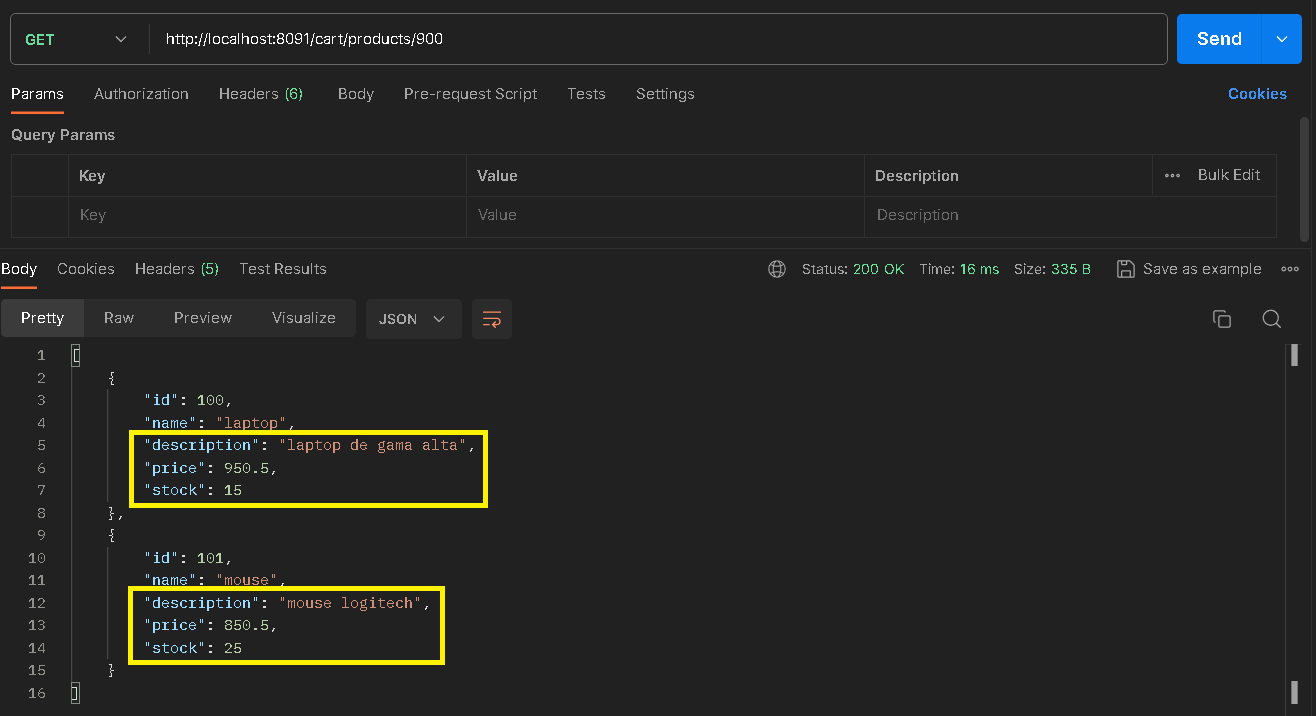
**Tabla

Descripción generada automáticamente**

**Pruebas con Postman**

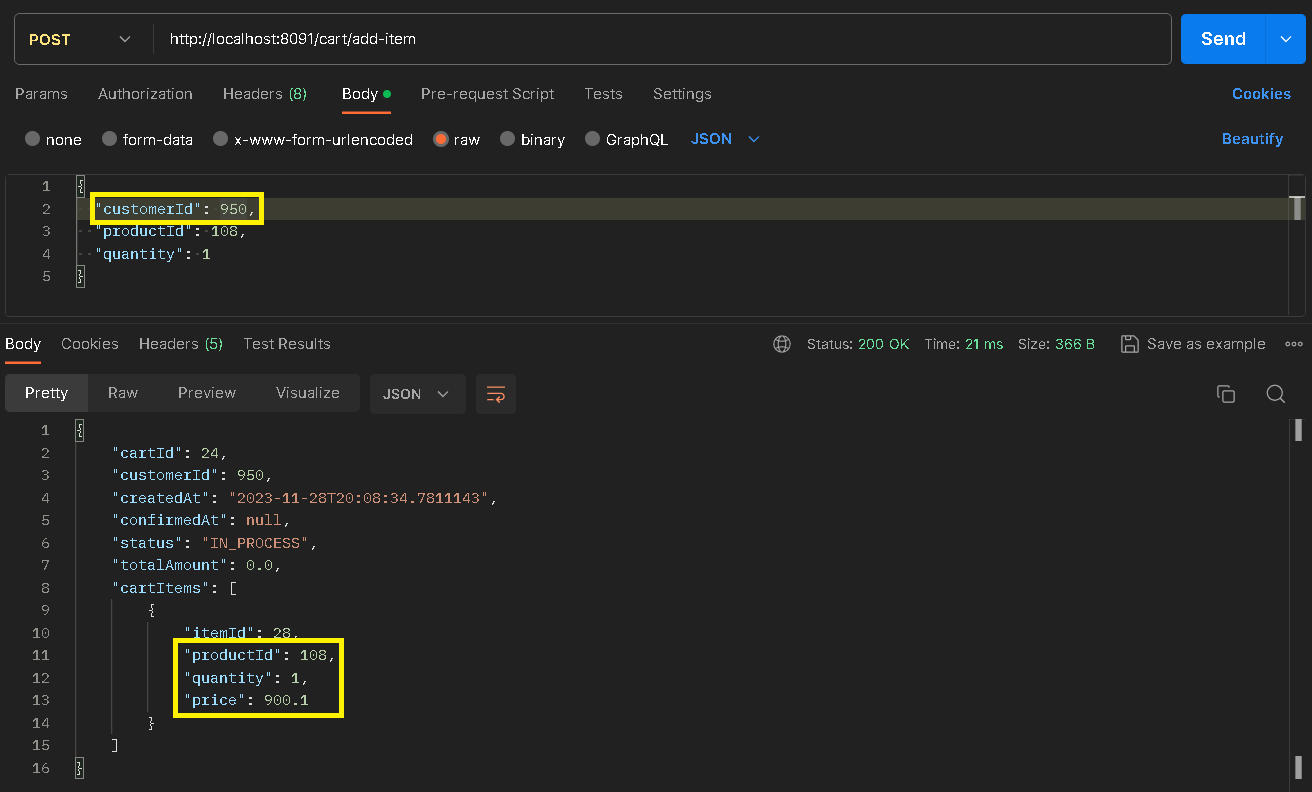
http://localhost:8091/cart/products/900

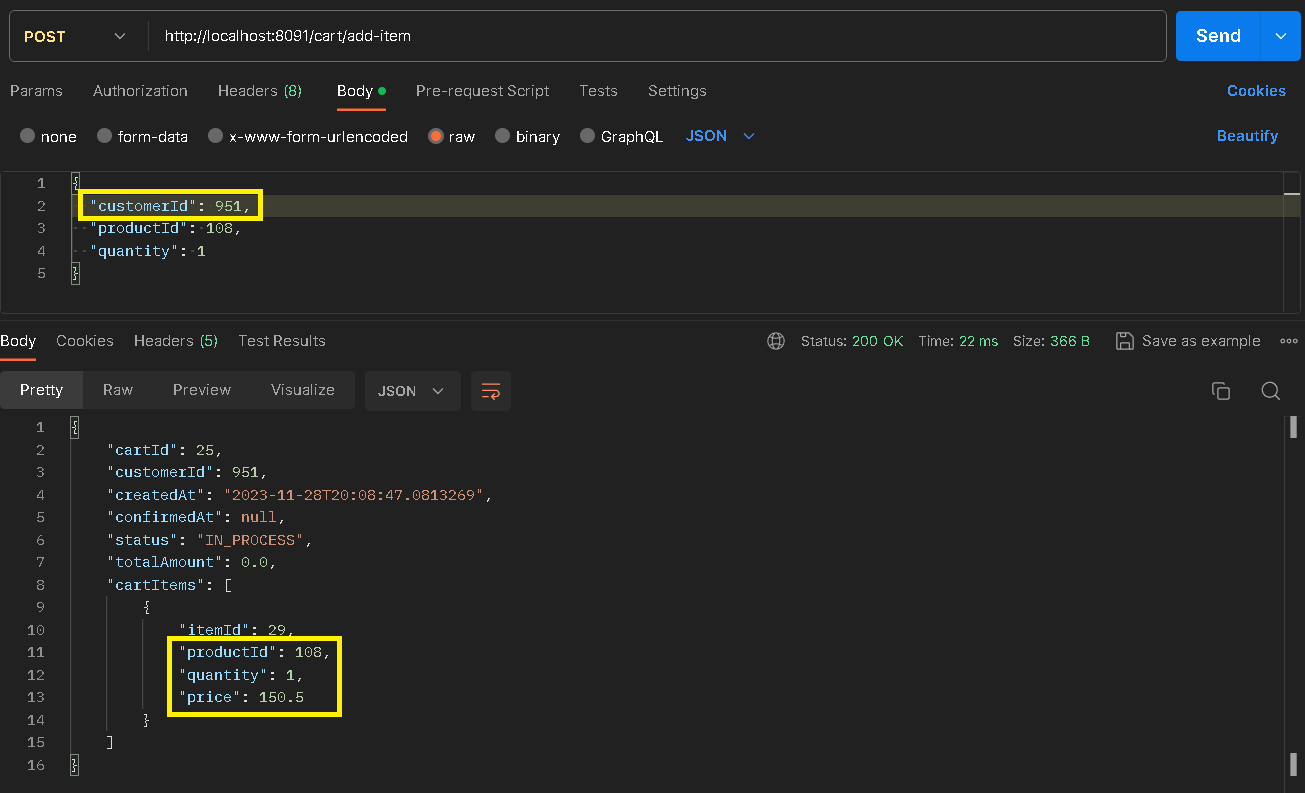
****

****

A medida que vamos dando click a “Send”, el detalle de los productos va cambiando pues el microservicio **ecommerce-shoppingcart-mysql-ms** internamente está llamando al microservicio **ecommerce-products-h2-ms**, balanceando dicha llamada entre los 02 microservicios levantados, cada una con su propia base de datos.

Se pueden hacer otras pruebas sobre el microservicio **ecommerce-shoppingcart-mysql-ms** donde se puede visualizar el balanceo de peticiones al microservicio **ecommerce-products-h2-ms**:

****

****

A continuación, variaremos en la implementación del microservicio de productos.

No usaremos H2 sino la misma base de datos MySql.

**Microservicio: ecommerce-products-mysql-ms (productos)**

SGBD: MySql

BD: ecommerce, ecommerce\_replica

**Diagrama de Microservicios y Bases de Datos**

Diagrama

Descripción generada automáticamente

**Procedimiento para armar el ambiente**

1. BD

1.1. MongoDB:

Crear la BD: ecommerce y el Collection: customer.

1.3. MySql:

Crear las 3 BD: ecommerce, **ecommerce\_replica**, ecommerce\_core

1. Microservicios

2.1. Levantar los 4 proyectos con IntelliJ IDEA

2.2. Ejecutar el proyecto: ecommerce-eureka-server

2.3. Ejecutar el proyecto: ecommerce-customers-mongodb-ms

2.4. Ejecutar el proyecto: ecommerce-products-mysql-ms

2.5. Ir a la opción Run –> Edit Configurations.

Volver e ejecutar el proyecto ecommerce-products-mysql-ms, agregando la opción “Allow multiple instances” y agregando en “Add VM options” lo siguiente:

**-Dserver.port=8082 -Dspring.datasource.url=jdbc:mysql://localhost:3306/ecommerce\_replica?useSSL=false&serverTimezone=UTC**

2.6. Ejecutar el proyecto: ecommerce-shoppingcart-mysql-ms

2.7. En la BD **ecommerce-replica**, ejecutar los siguientes updates:

update product set description='laptop', price=100.10, stock=90 where id=100;

update product set description='mouse', price=200.10, stock=80 where id=101;

update product set description='teclado', price=300.10, stock=70 where id=102;

update product set description='televisor', price=400.10, stock=60 where id=103;

update product set description='refrigeradora', price=500.10, stock=50 where id=104;

update product set description='celular', price=600.10, stock=40 where id=105;

update product set description='iphone', price=700.10, stock=30 where id=106;

update product set description='nintendo', price=800.10, stock=20 where id=107;

update product set description='playstation', price=900.10, stock=10 where id=108;

**Validación de microservicios registrados en Eureka Server**

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

**Pruebas con Postman**

http://localhost:8091/cart/products/900

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

A medida que vamos dando click a “Send”, el detalle de los productos va cambiando pues el microservicio **ecommerce-shoppingcart-mysql-ms** internamente está llamando al microservicio **ecommerce-products-mysql-ms**, balanceando dicha llamada entre los 02 microservicios levantados, cada una con su propia base de datos.

Se pueden hacer otras pruebas en donde se puede visualizar el balanceo de peticiones al microservicio ecommerce-products-mysql-ms:

Captura de pantalla de un celular

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Sitio web

Descripción generada automáticamente