

Lab 4 – Spring 2: Modelado de un sistema de microservicios

Ejercicio sobre una plataforma de comercio electrónico

Nombre de la Aplicación: E-commerce Platform

1. Microservicios:

1. **Frontend Service:** Proporciona la interfaz de usuario para la plataforma de comercio electrónico.
2. **User Service:** Maneja la gestión de usuarios.
3. **Product Service:** Maneja la gestión de productos.
4. **Order Service:** Maneja la gestión de pedidos.
5. **Payment Service:** Maneja la gestión de pagos.
6. **Recommendation Service:** Genera recomendaciones de productos.
7. **API Gateway Service:** Actúa como intermediario para enrutar solicitudes y manejar la autenticación y autorización.

2. Endpoints de la API:

• User Service:

- **POST /users/register:** Registrar un nuevo usuario.
- **POST /users/login:** Autenticar un usuario.
- **GET /users/{id}:** Obtener detalles de un usuario.
- **PUT /users/{id}:** Actualizar un usuario.

• Product Service:

- **GET /products:** Listar todos los productos.
- **POST /products:** Crear un nuevo producto.
- **GET /products/{id}:** Obtener detalles de un producto.
- **PUT /products/{id}:** Actualizar un producto.
- **DELETE /products/{id}:** Eliminar un producto.

• Order Service:

- **POST /orders:** Crear un nuevo pedido.
- **GET /orders/<id>:** Obtener detalles de un pedido.
- **GET /orders/user/<userId>:** Obtener todos los pedidos de un usuario.

- **Payment Service:**

- **POST /pay:** Realizar un pago.
- **GET /payments/<orderId>:** Obtener el estado de un pago.

- **Recommendation Service:**

- **GET /recommendations/<userId>:** Obtener recomendaciones para un usuario.

3. Persistencia de Datos:

- **User Service:** Base de datos SQLite para almacenar información de usuarios.
- **Product Service:** Base de datos SQLite para almacenar información de productos.
- **Order Service:** Base de datos SQLite para almacenar información de pedidos.
- **Payment Service:** Base de datos SQLite para almacenar información de pagos.
- **Recommendation Service:** Base de datos SQLite para almacenar información de recomendaciones.

4. Dockerización:

- **Dockerfile:** Cada microservicio tiene su propio Dockerfile para crear contenedores independientes.
- **docker-compose.yml:** Archivo de composición para orquestar los contenedores y definir la red de comunicación entre ellos.