**Informe Técnico - Etapa 6**

**Proyecto: Microservicios para Gestión de Usuarios, Inventario y Carrito de Compras**

**Asignatura:** DSY1103 - Desarrollo de Software

**Integrantes:** [Allan Salinas- David Fuentes]

**Fecha de Entrega:** Domingo 25 de mayo de 2025

**Introducción**

Este informe corresponde a la Etapa 6 del caso semestral, centrado en el desarrollo y configuración de tres microservicios independientes usando Spring Boot y arquitectura RESTful. Los microservicios desarrollados son:

* Microservicio de **usuarios**
* Microservicio de **inventario**
* Microservicio de **carrito**

Cada microservicio gestiona su propio recurso, base de datos y operaciones CRUD, y están diseñados para funcionar de forma desacoplada y escalable.

**Análisis de Requerimientos**

Este proyecto responde a los requerimientos identificados en la Evaluación Parcial N°1, abordando la gestión de usuarios, productos en inventario y operaciones de carrito de compras de clientes.

Los requerimientos principales considerados fueron:

* Mantener información de usuarios con roles y correos únicos.
* Gestionar productos con categoría, stock y precio.
* Permitir a los usuarios crear carritos con productos y cantidades.

**Diseño del Sistema**

**Arquitectura**

Se utilizó una arquitectura de microservicios con tres servicios desplegados de manera independiente:

* Cada microservicio es un proyecto independiente de Spring Boot.
* Cada uno se conecta a una base de datos propia.
* Comunicación a través de API REST usando JSON.

**Casos de Uso**

* CRUD de usuarios.
* CRUD de productos.
* CRUD de ítems en carrito de compras.

**Modelo de Datos**

Los modelos fueron definidos con JPA, y reflejan las tablas requeridas en el sistema. Se utilizaron anotaciones para definir entidades, claves primarias y relaciones.

**Implementación**

**Sub-sección: Base de Datos**

Cada microservicio posee su propia base de datos con las siguientes tablas:

**Usuarios**

* id (PK)
* nombre
* correo
* rol

**Productos (Inventario)**

* id (PK)
* nombre
* categoria
* stock
* precio

**CarritoItems**

* id (PK)
* usuario\_id (FK a Usuarios)
* producto\_id (FK a Productos)
* cantidad

Se garantiza la integridad referencial mediante claves foráneas. El archivo script\_creacion\_tablas.sql contiene los comandos SQL compatibles con los modelos JPA.

**Sub-sección: Microservicios**

Se desarrollaron tres microservicios utilizando Spring Boot, JPA y controladores REST. Cada uno incluye:

* Modelo de datos
* Repositorio (JPARepository)
* Servicio con lógica de negocio básica
* Controlador REST con operaciones CRUD y filtros por parámetros de consulta

El microservicio de carrito es el más complejo, ya que realiza validaciones y dependencias cruzadas con los endpoints de inventario y usuarios.

**Herramientas y Tecnologías**

* Spring Boot 3
* Spring Data JPA
* Maven
* PostgreSQL/MySQL (según configuración)
* Postman (pruebas de endpoints)
* GitHub (repositorio de código y control de versiones)

**Gestión del Proyecto**

* Se creó un repositorio GitHub con ramas main y develop.
* Cada miembro del equipo trabajó sobre la rama develop.
* Los commits fueron registrados progresivamente hasta el 25 de mayo de 2025.
* Se respetó la estructura de carpetas solicitada por la pauta de evaluación.

**Pruebas y Validación**

Se realizó una colección de Postman que prueba los tres microservicios, incluyendo al menos cinco ejemplos por cada endpoint. Esta colección fue compartida al correo del profesor tom.opazo@profesor.duoc.cl.

Además, se adjunta un archivo Excel con los siguientes datos por cada prueba:

* Método (GET, POST, PUT, DELETE)
* URL
* JSON de entrada
* JSON de salida esperado

**Conclusión**

El desarrollo de esta etapa permitió aplicar conocimientos de diseño modular, arquitectura basada en microservicios, buenas prácticas de desarrollo backend, uso de control de versiones con Git y pruebas con Postman. Cada microservicio es ejecutable de manera independiente y puede escalarse o integrarse fácilmente en un sistema más complejo.

**Anexos:**

* script\_creacion\_tablas.sql
* README.md
* Colección Postman
* Archivo Excel con ejemplos de endpoints

**Fin del Informe**