Reporte Individual — Tarea 3: Diseño e Implementación de Arquitecturas de Microservicios

Nombre: Angel Janvier Gonzalez Delgado Materia: Arquitectura de Software
Profesor: Julieta G. Rodríguez Ruiz **Fecha de entrega:** Viernes 13 de Junio, 2025
Repositorio del equipo: Tarea de Microservicios\docker-microservicios

---

1) ¿Qué es un microservicio?

Un microservicio es una unidad de software pequeña, independiente y desplegable que realiza una única función de negocio dentro de un sistema más grande. Cada microservicio se comunica con otros mediante APIs bien definidas y permite escalar y mantener el sistema de forma más flexible.

---

2) Las 9 características según Martin Fowler y James Lewis

Característica	Descripción
1. Componentization via Services	Cada parte del sistema se construye como un servicio independiente.
2. Organised around Business Capabilities	Los equipos se organizan por funcionalidades de negocio, no por capas técnicas.
3. Products not Projects	Los equipos son responsables de un producto completo, no solo de su desarrollo.
4. Smart Endpoints and Dumb Pipes	Los microservicios son inteligentes; la comunicación entre ellos es simple.
5. Decentralized Governance	Se fomenta la autonomía tecnológica en cada equipo.
6. Decentralized Data Management	Cada servicio maneja su propia base de datos.
7. Infrastructure Automation	Se automatiza la creación y despliegue de infraestructura.
8. Design for Failure	El sistema se diseña para tolerar fallos de algunos servicios.
9. Evolutionary Design	Permite adaptar la arquitectura conforme cambian los requisitos.

3) Diferencia entre microservicio y API

Un microservicio es una aplicación pequeña que ejecuta una lógica de negocio específica. Una API es solo la interfaz de comunicación que expone ese microservicio. Es decir, la API es la puerta de entrada; el microservicio es la lógica y datos internos.

---

4) ¿Para qué se usa el API Gateway y cuál es su ventaja?

El API Gateway actúa como un único punto de entrada para todas las solicitudes externas hacia los microservicios. Simplifica la gestión de autenticación, balanceo de

carga, control de tráfico y seguridad. La ventaja principal es que desacopla a los clientes de la complejidad interna del sistema.

---

5) Tabla de responsabilidades de los elementos del sistema

Elemento	Responsabilidad
API Gateway (Tyk)	Orquesta todas las solicitudes de los clientes a los microservicios.
Cliente (Flutter)	Interfaz gráfica para interactuar con el sistema.
Gestor de Clientes	Gestiona información de los clientes asegurados.
Notificador	Envía notificaciones a clientes mediante Telegram.
Pagos	Procesa la lógica relacionada con pagos.
Reporteador	Genera reportes del sistema.
Simulador	Simula los pagos realizados para pruebas.
payment-validator-ms	Lee pagos, valida estado y envía póliza por Telegram.

6) Análisis del sistema: requerimientos y soporte

Requerimiento de arquitectura	¿Es soportado?	¿Cómo es soportado?	¿Contribuye al objetivo de negocio?
Gestión de clientes asegurados	Sí	Con el microservicio gestor de clientes.	Sí
Notificación de pago validado	Sí	Con payment-validator-ms y notificador.	Sí

Requerimiento de arquitectura	¿Es soportado?	¿Cómo es soportado?	¿Contribuye al objetivo de negocio?
Envío de póliza por mensajería	Sí	Con Telegram y payment-validator-ms.	Sí

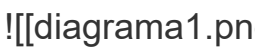
7) Atributos de calidad adicionales

Atributo	Definición	¿Es soportado?	¿Cómo se soporta?
Escalabilidad	Capacidad de crecer bajo demanda.	Sí	Se pueden añadir más instancias de microservicios.
Seguridad	Proteger datos y comunicaciones.	Sí	Uso de Tyk Gateway para control de acceso.
Auditabilidad	Registrar acciones del sistema.	Sí	Logs de servicios y registros de pagos.
Desempeño	Respuesta rápida y eficiente.	Sí	Servicios independientes optimizados y simulación local.

8) Justificación de la decisión de diseño

El diseño basado en microservicios permite dividir la complejidad del sistema en unidades pequeñas, que se pueden mantener, desplegar y escalar de forma independiente. Esto soporta el objetivo de negocio de brindar un servicio más rápido y flexible a medida que la cantidad de clientes crece.

9) Diagrama actualizado



![[diagrama2.png]]

10) Repositorio

URL del repositorio: Tarea de Microservicios\docker-microservicios

---