**Departamento de Ciencias de la Computación(DCCO)**

**Carrera de Ingeniería de Software**

**Curso de Aplicaciones Distribuidas**

Enfoques Arquitectónicos

Presentado por: Marley Almeida, Sebastian Bolaños, Nicole Lara, Axel Pullaguari

Tutor: Morales, Dario.

Ciudad: Sangolquí, Ecuador

Fecha: 01/02/2025

# **Enfoques Arquitectónicos**

## **Arquitectura Basada en API Gateway**

### **Descripción**

Un API Gateway actúa como un punto de entrada único para todas las solicitudes dirigidas a los microservicios. El Gateway enruta, autentica y gestiona las peticiones antes de que lleguen a los microservicios individuales.

### **Ventajas y Desventajas**

|  |  |
| --- | --- |
| **Ventajas** | **Desventajas** |
| Mejora la seguridad centralizando la autenticación y autorización. | Agrega un punto único de fallo si no se gestiona adecuadamente. |
| Reduce la carga de los microservicios al gestionar peticiones, caching y balanceo de carga. | Puede generar latencias adicionales en cada solicitud. |
| Simplifica el consumo de APIs al ofrecer una única interfaz para clientes. |  |

### **Viabilidad**

Este enfoque es recomendado porque facilita la escalabilidad y la seguridad, permitiendo administrar autenticación y balanceo de carga desde un solo punto. Dado que el sistema tiene varios módulos, un API Gateway ayuda a administrar mejor el tráfico.

## **Arquitectura Basada en Comunicación Directa entre Microservicios**

### **Descripción**

Cada microservicio se comunica directamente con otros microservicios sin un punto central de control.

### **Ventajas y Desventajas**

|  |  |
| --- | --- |
| **Ventajas** | **Desventajas** |
| Reduce la sobrecarga de administración de un Gateway. | Aumenta la complejidad en la gestión de dependencias entre servicios. |
| Puede ser más rápido en sistemas con pocas dependencias entre servicios. | Difícil de manejar en proyectos grandes debido al acoplamiento entre microservicios. |

### **Viabilidad**

No se recomienda este tipo de arquitectura, porque la aplicación tiene múltiples módulos que interactúan con frecuencia. La comunicación directa puede generar una red de interdependencias difícil de mantener.

## Arquitectura Basada en Mensajería Asíncrona

### **Descripción**

Los microservicios se comunican mediante un bus de eventos, lo que permite una comunicación asíncrona sin dependencias directas.

### **Ventajas y Desventajas**

|  |  |
| --- | --- |
| **Ventajas** | **Desventajas** |
| Reduce la latencia y permite el escalado independiente de los servicios. | Introduce complejidad en la trazabilidad de eventos y depuración. |
| Evita bloqueos en la comunicación, mejorando la resiliencia del sistema. | Puede generar problemas de consistencia eventual en algunas operaciones. |

### **Viabilidad**

Este enfoque podría ser útil para ciertas operaciones, como el registro de asistencia o generación de reportes, pero no es necesario para toda la aplicación.

# **Conclusión**

Basándonos en los análisis anteriores, la mejor opción para este proyecto es una **arquitectura basada en API Gateway**, debido a:

* + API Gateway gestionará la autenticación, enrutamiento y seguridad.
  + Los microservicios estarán desacoplados y se comunicarán a través del Gateway.