

Taller 2

Integrantes

Rodrigo Puente

Tema: Diferencias entre Servicios y Microservicios

Objetivos:

- Proporcionar una comprensión clara y concisa de lo que son los servicios y los microservicios en el contexto de la programación distribuida.
- Analizar en profundidad las diferencias esenciales entre los enfoques de servicios y microservicios.
- Evaluar situaciones en las que sería más adecuado utilizar un enfoque basado en servicios o en microservicios.

Desarrollo

Servicios

Los servicios, en el contexto de la programación distribuida, son componentes independientes y autónomos de una aplicación que ofrecen funcionalidades específicas. Estos servicios se comunican entre sí a través de protocolos estándar, como HTTP, y pueden implementarse en diferentes lenguajes de programación. La arquitectura de servicios suele ser monolítica, lo que significa que la aplicación se divide en módulos separados, pero sigue siendo una entidad única. Esto puede generar acoplamiento y dificultades para escalar componentes individuales.

Microservicios

Los microservicios, por otro lado, son una arquitectura en la que una aplicación se construye como una colección de servicios pequeños y autónomos. Cada microservicio se enfoca en una única funcionalidad y se comunica con otros microservicios a través de APIs. Esta arquitectura promueve la independencia, la escalabilidad y la mantenibilidad. Los microservicios pueden implementarse en diferentes lenguajes y tecnologías, lo que facilita la elección de la mejor herramienta para cada servicio específico.

Conclusiones

En resumen, los servicios y los microservicios son dos enfoques arquitectónicos que abordan la programación distribuida de manera diferente. Los servicios tienden a ser más monolíticos y pueden resultar adecuados para aplicaciones más simples o equipos con menos recursos. Por otro lado, los microservicios ofrecen mayor flexibilidad y escalabilidad, lo que los hace ideales para aplicaciones complejas en entornos dinámicos.

La elección entre servicios y microservicios depende de múltiples factores, como la complejidad del sistema, los requisitos de tiempo de respuesta, las necesidades de escalabilidad y el equipo de desarrollo. Cada enfoque tiene sus ventajas y desventajas, y es esencial evaluar cuidadosamente las características del proyecto antes de tomar una decisión.

En última instancia, tanto los servicios como los microservicios tienen un lugar en la programación distribuida, y la elección entre ellos debe basarse en una comprensión sólida de las necesidades del proyecto y las capacidades de desarrollo disponibles.

Bibliografía

Newman, S. (2015). "Building Microservices: Designing Fine-Grained Systems." O'Reilly Media.

Fowler, M. (2014). "Microservices: a definition of this new architectural term." MartinFowler.com.

Lewis, J., & Fowler, M. (2018). "Microservices." Sam Newman.