Transforma tu arquitectura con Microservicios en el desarrollo de software.

Iván Villa López
Gerencia Solution Center CDMX

Iniciemos hablando de los **microservicios** como una arquitectura que vino a revolucionar el desarrollo de software dentro del mundo de TI, son más clientes que requieren que sus proyectos informáticos sean creados con esta arquitectura y más aún, los que solicitan que sus sistemas actuales sean migrados por la eficiencia a corto plazo que pueden encontrar con esta arquitectura. Un caso de éxito es Netflix, que maneja más de 200 millones de usuarios mediante un sistema basado en microservicios.

¿Qué son y para qué sirven los microservicios?

Son soluciones pensadas en el negocio, que dan mayor agilidad a la respuesta esperada por el cliente, ya que un equipo de trabajo puede trabajar de manera independiente con cada de uno de los servicios requeridos, por lo que es más fácil diseñar, probar, implementar y actualizar, en comparación con una arquitectura monolítica. Algunas de las tendencias que han evolucionado o son más populares para la integración de microservicios:

- La arquitectura basada en eventos: Esta permite una comunicación más asíncrona entre los microservicios, tecnologías como *Kafka, RabbitMQ y Amazon EventBridge* son utilizadas para implementar patrones de eventos.
- API *Gateways* avanzados: Se incluyen características como gestión de tráfico basada en políticas y seguridad avanzada.
- *Service Mesh:* Proporciona balaceo de carga, trazabilidad y observabilidad avanzadas, los service Meshes somo Istio y Consul, son utilizados para la gestión de la comunicación.

Aplicaciones que utilizan microservicios son *Amazon, Ebay, Netflix y Facebook.*

Vamos a imaginar con un ejemplo práctico, como podemos aplicar los microservicios para una aplicación de e-commerce, esta podría estar distribuida de la siguiente manera:

- **Servicio de usuarios**: Maneja la autenticación y la gestión de cuentas.
- **Servicio de catálogo**: Administra la lista de productos.
- **Servicio de pagos**: Procesa las transacciones.
- **Servicio de notificaciones**: Envía correos y notificaciones al usuario.

¿Cuáles son las principales características de trabajar con microservicios?

- La independencia de poder desarrollar, desplegar y escalar de manera independiente cada uno de los servicios.
- El diseño de cada servicio es desarrollado de acuerdo con una funcionalidad en específico, tomando nuestro ejemplo de referencia, un servicio de usuario, de pagos y notificaciones.

Insurgentes Sur No. 64, Torre B, Oficina 501, 502 y 503, Col Juárez, Alcaldía Cuauhtémoc, CDMX, C.P. 06600
praxisglobe.com
55 5080-0048

- Como lo mencionamos en el primer punto, el despliegue lo podemos realizar de manera independiente, por lo que, muy importante, disminuye la afectación del funcionamiento del resto del sistema.
- Escalabilidad solo de los servicios requeridos, no de todo el aplicativo.
- Se pueden utilizar diferentes tecnologías, lenguajes de programación, bases de datos, de acuerdo con lo mejor de cada servicio.
- La resistencia a errores en la aplicación aumenta, ya que, en una arquitectura monolítica, un error de componente puede provocar error en toda la aplicación.

Diferencia entre arquitectura monolítica y arquitectura de microservicios.

En las arquitecturas monolíticas todos los procesos están asociados y se ejecutan como un solo servicio, esto quiere decir, que, si un proceso de la aplicación llega a tener alguna deficiencia, se debe de atender a toda la aplicación. Por lo que aumenta el riesgo de la disponibilidad que se tenga de la aplicación.

Si contamos con una arquitectura de microservicios, entendemos que la aplicación se crea con componentes independientes, se comunican mediante una interfaz API, en contraste con la arquitectura monolítica, el riesgo de no tener disponibilidad de la aplicación disminuye.

Dentro de las empresas, un impacto positivo al utilizar microservicios es la agilidad a la adaptación de cambios, al no tener que actualizar todo el aplicativo, se vuelve más rápida la entrega al usuario, ya que el desarrollo y pruebas, se centraliza en un solo microservicio, a esto agregamos el aumento de la disponibilidad de la aplicación.

Dentro de las tecnologías más populares para la integración de microservicios, se encuentra los lenguajes de programación Java, JavaScript, Python, por mencionar algunos.

Ventajas:

- **Modularidad:** Al ser servicios autónomos, se pueden trabajar de manera independiente hasta su despliegue. Dentro de este despliegue, el resto de los servicios pueden continuar trabajando.
- **Escalabilidad:** Al ser una aplicación modular, esta se puede escalar de manera horizontal de acuerdo con las necesidades requeridas.
- **Versátil:** Se pueden utilizar diferentes tecnologías y lenguajes de programación, dando posibilidad de poder implementar la que mejores adapte a las necesidades requeridas.
- **Mantenimiento más ágil:** Al realizar mejoras solo a lo que el aplicativo necesite, disminuye el tiempo y costo de atención.

Insurgentes Sur No. 64, Torre B, Oficina 501, 502 y 503, Col Juárez, Alcaldía Cuauhtémoc, CDMX, C.P. 06600
praxisglobe.com
55 5080-0048

Desventajas:

- **Alto consumo de memoria:** Cada microservicio trabaja con sus propios recursos y bases de datos, por lo que consumen más memoria y CPU.
- **Inversión inicial:** El definir como estará fragmentado cada uno de los microservicios, el consumo de tiempo en esta definición es mayor.
- **Gestión:** Se debe de contar con una gestión clara, ya que, en aplicaciones de gran tamaño, se puede volver compleja la administración de cada uno de los microservicios.
- **Perfil del desarrollador:** Se requiere experiencia en soluciones y versiona miento, así como para la solución de latencia y carga de la red.

PRAXIS y la especialidad de Solution Center considera las ventajas y desventajas de la implementación de una arquitectura de microservicios; lo más importante antes de tomar la elección de con qué arquitectura deseamos nuestra aplicación, validamos, que solución es la que se está buscando y se adapta a las necesidades del cliente y proyecto con el objetivo de lograr la satisfacción del cliente.

El desarrollo o migración a una arquitectura de microservicio, es fundamental para mejorar la escalabilidad, agilidad y resiliencia de los sistemas. Aunque, no menos importante, la decisión de migrar a debe basarse en una evaluación de costo / Beneficio.

CARLOS GARCÍA
Gerente de la Especialidad Solution Center CDMX
55 5080 0048
solutioncenter@praxisglobe.com
Carlos García

Referencias:

1. https://aws.amazon.com/es/microservices/#:~:text=Los%20microservicios%20fomentan%20una%20organizaci%C3%B3n,tiempos%20del%20ciclo%20de%20desarrollo.

2. https://www.redhat.com/es/topics/microservices

3. https://decidesoluciones.es/arquitectura-de-microservicios/

4. https://jorgegarciadesign.wordpress.com/2020/04/06/microservicios-para-que-y-cuando-usarlos/

5. https://www.mordorintelligence.com/es/industry-reports/cloud-microservices-market

6. https://www.paradigmadigital.com/dev/netflix-un-gigante-software/

Insurgentes Sur No. 64, Torre B, Oficina 501, 502 y 503, Col Juárez, Alcaldía Cuauhtémoc, CDMX, C.P. 06600

praxisglobe.com
55 5080-0048