



DuocUC[®] INFORMÁTICA Y
TELECOMUNICACIONES

■ INTRODUCCIÓN A MICROSERVICIOS

DESARROLLO FULLSTACK I
DSY1103

• Contenidos

- ¿Qué es la arquitectura de microservicios?.
- Ventajas y desventajas de la arquitectura de microservicios.
- ¿Qué es la arquitectura monolítica?.
- Ventajas y desventajas de la arquitectura monolítica.
- El boom de los microservicios.
- Realizar actividad.
- Reflexión.

Desarrollamos rápido para probar los límites de lo que podemos hacer.
– *Mark Zuckerberg*

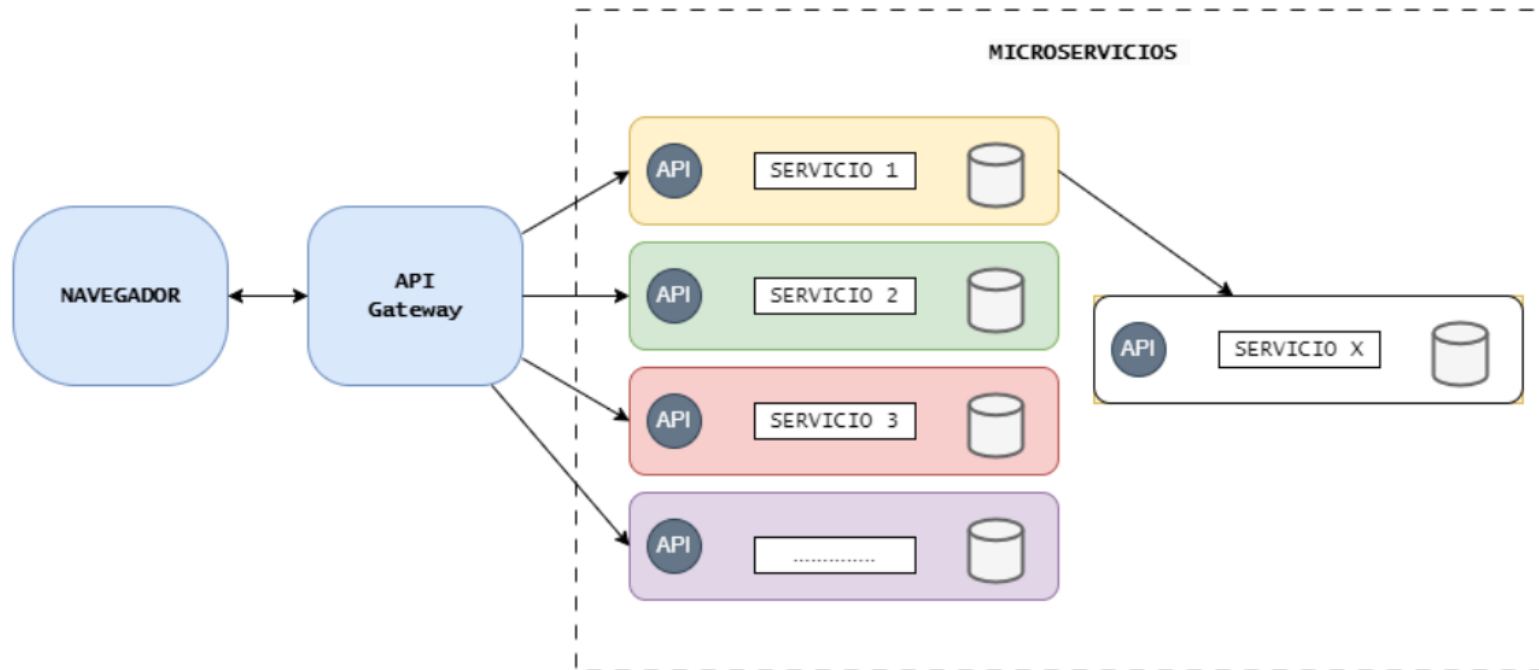
A black and white photograph of a man in a suit standing in a modern office, holding a tablet and smiling. The office has glass partitions and modern furniture. A blue square is in the top right corner.

01

Introducción a los Microservicios

• Microservicios

Los **microservicios** son un estilo arquitectónico que estructura una aplicación como un conjunto de servicios pequeños y autónomos, que se comunican a través de **APIs** bien definidas. Cada **servicio** está diseñado para realizar una función específica y puede ser desarrollado, desplegado y escalado de manera independiente de los demás servicios.





• Beneficios

ESCALABILIDAD HORIZONTAL

Cada microservicio se puede escalar de manera independiente según la demanda. Esto permite una mejor utilización de recursos y una escalabilidad más eficiente.

INDEPENDENCIA DE COMPONENTES

Los microservicios pueden desplegarse en diferentes servidores o en la nube, distribuyendo la carga de trabajo de manera más efectiva.

CICLO DE VIDA INDEPENDIENTE

Cada microservicio puede desarrollarse, probarse y desplegarse de forma independiente, lo que facilita la implementación continua y reduce el riesgo de afectar otras partes del sistema.



• Beneficios

ACTUALIZACIONES Y MANTENIMIENTO

Las actualizaciones y correcciones pueden realizarse en servicios específicos sin afectar a toda la aplicación.

TECNOLOGÍA APROPIADA PARA CADA SERVICIO

Se puede elegir la mejor tecnología o lenguaje de programación para cada microservicio, permitiendo una optimización específica para diferentes partes del sistema.

INNOVACIÓN Y EXPERIMENTACIÓN

Facilita la adopción de nuevas tecnologías y la experimentación con diferentes enfoques sin afectar a toda la aplicación.

AISLAMIENTO DE FALLOS

Si un microservicio falla, no necesariamente afecta a los demás, lo que mejora la resiliencia y la disponibilidad del sistema en su conjunto.



- **Desventajas**

GESTIÓN DE SERVICIOS

Administrar múltiples servicios puede ser complejo y requiere una infraestructura avanzada para orquestación, monitoreo y registro.

DESARROLLO Y PRUEBAS

El desarrollo y las pruebas de un sistema distribuido pueden ser más complicados debido a la interacción entre servicios y la necesidad de gestionar la consistencia de los datos.

RECURSOS ADICIONALES

Solo necesitas desplegar un solo archivo o conjunto de archivos, lo que simplifica la gestión de versiones y la implementación.



• Desventajas

OVERHEAD DE COMUNICACIÓN

La comunicación entre microservicios a través de la red puede introducir latencia y requerir un manejo eficiente de las solicitudes y respuestas.

SUPERFICIE DE ATAQUE MAYOR

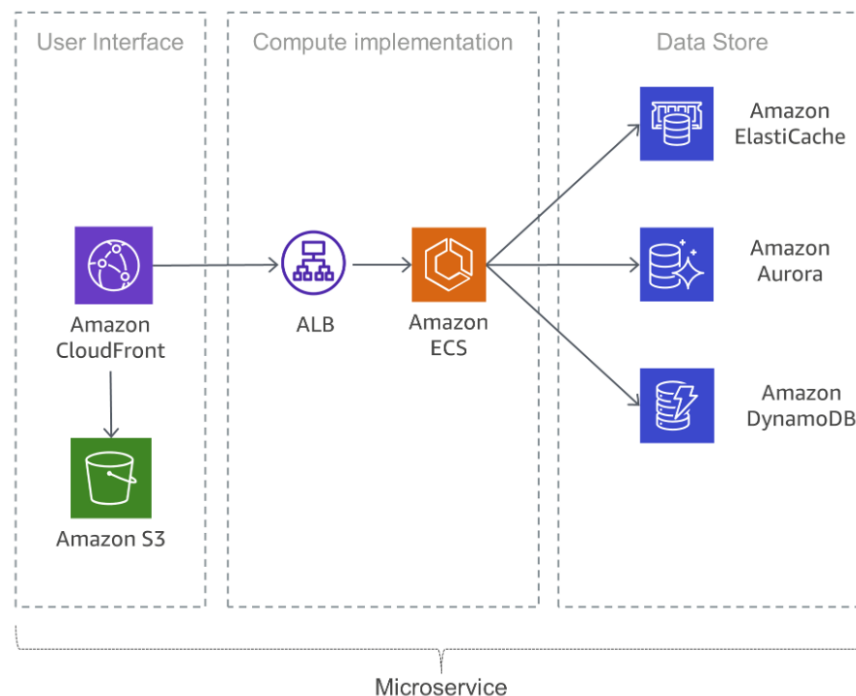
Con múltiples puntos de entrada y comunicación entre servicios, la seguridad se vuelve más compleja y puede requerir medidas adicionales para proteger cada microservicio y sus interacciones.

GESTIÓN DE TRANSACCIONES

Mantener la consistencia de datos en un sistema distribuido puede ser un desafío, especialmente en transacciones que abarcan múltiples servicios.

• Microservicios

La arquitectura de **microservicios** es ideal para aplicaciones grandes y complejas que requieren escalabilidad, flexibilidad y resiliencia. Sin embargo, introduce una mayor complejidad en el desarrollo y la gestión, así como posibles costos operacionales adicionales. Es crucial evaluar si los beneficios superan las desventajas según las necesidades específicas del proyecto.





02

Arquitectura Monolítica

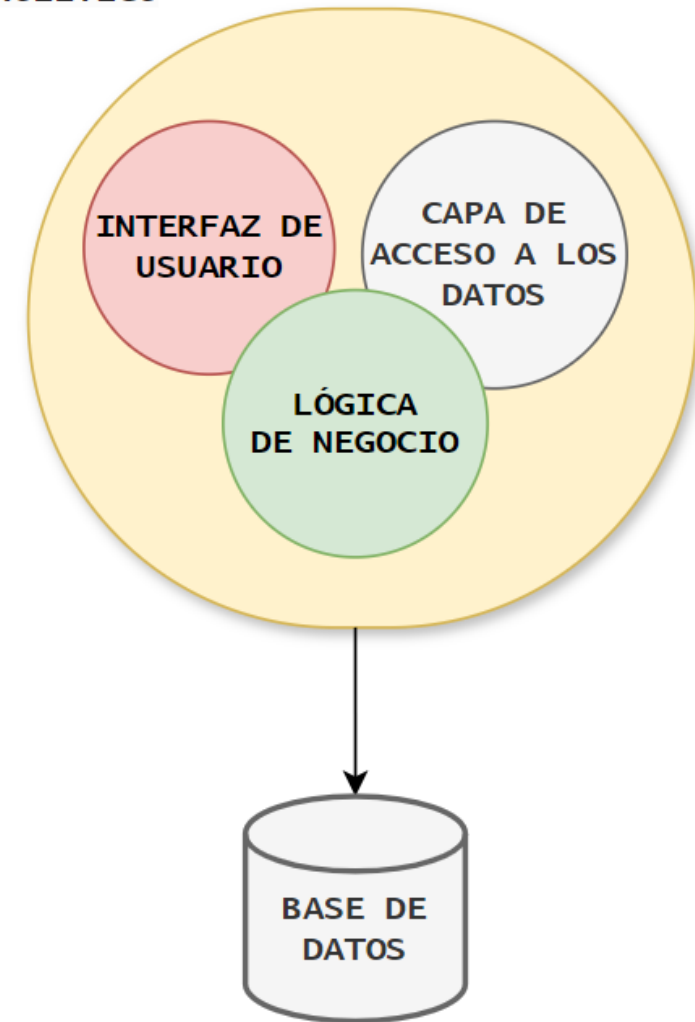
• Arquitectura Monolítica

Una aplicación monolítica se caracteriza por tener todos sus **componentes** integrados en **un solo bloque de código**.

Estas aplicaciones son fáciles de desarrollar, con un despliegue y ejecución sencilla, y su coste de desarrollo es más bajo en comparación con otras arquitecturas.

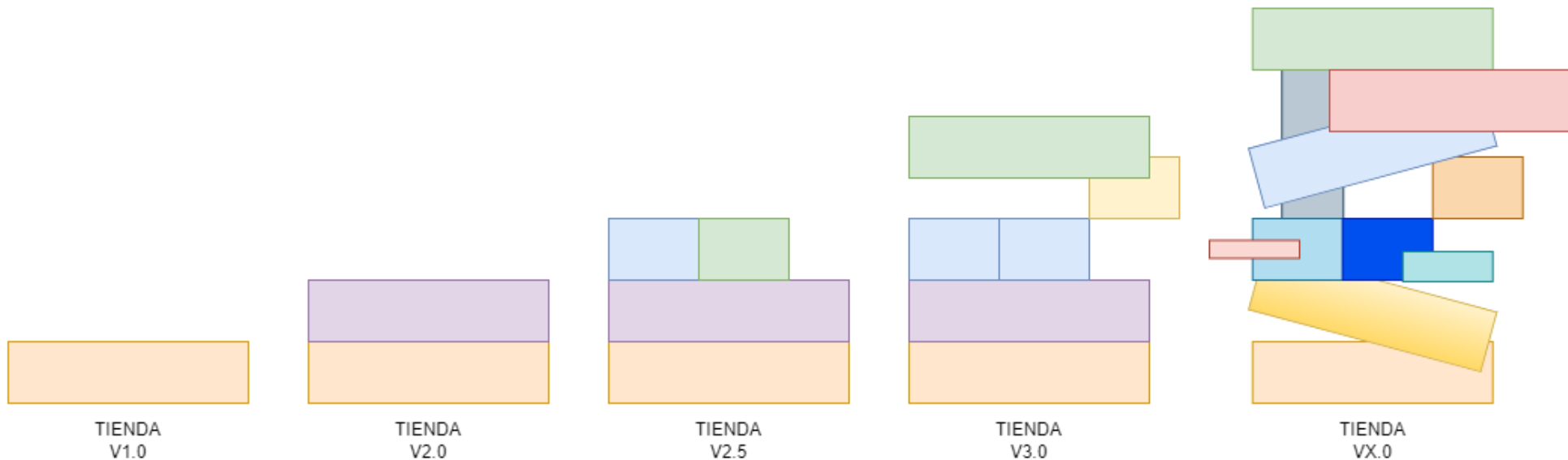
No obstante, presentan desventajas significativas que las hacen una opción cada vez menos preferida en el desarrollo de software moderno.

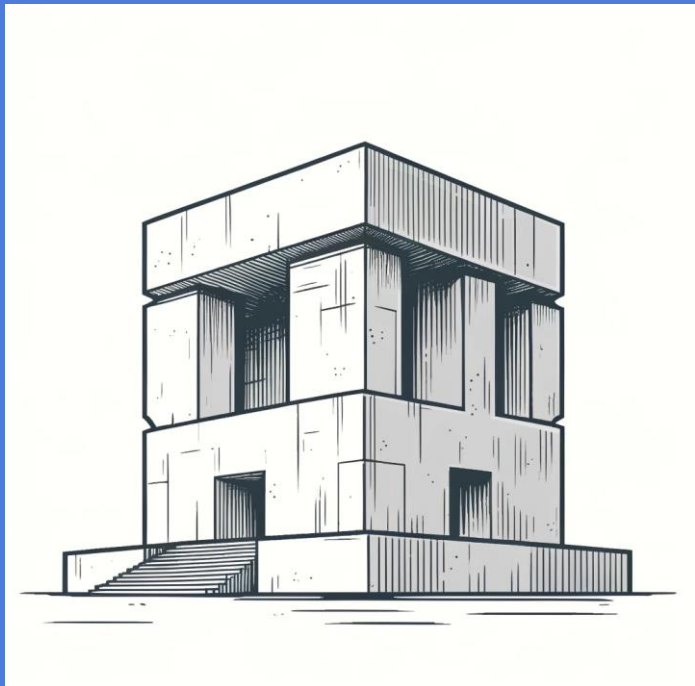
MONOLÍTICO



• Arquitectura Monolítica

Si un **monolito** no está bien diseñado en módulos y lo suficientemente desacoplado, según va creciendo el tamaño de la aplicación puede llegar a representar un alto coste por la complejidad del mantenimiento del código (deuda técnica).





- **Beneficios**

FÁCIL DE ENTENDER Y DESARROLLAR

Las aplicaciones monolíticas son más fáciles de desarrollar, especialmente para equipos pequeños. Todo el código se encuentra en un solo lugar, lo que simplifica la comprensión y el desarrollo.

FACILIDAD DE PRUEBAS Y DEPURACIÓN

Las pruebas y la depuración suelen ser más sencillas porque todo está en un solo proyecto.

UN SOLO DESPLIEGUE

Solo necesitas desplegar un solo archivo o conjunto de archivos, lo que simplifica la gestión de versiones y la implementación.

RENDIMIENTO DE MENOR LATENCIA DE COMUNICACIÓN

Dado que todos los componentes están en el mismo proceso, las llamadas entre módulos son simples invocaciones de funciones, lo que puede resultar en una menor latencia.



- **Desventajas**

ESCALABILIDAD HORIZONTAL DIFÍCIL

Es más difícil escalar una aplicación monolítica horizontalmente, ya que no se puede dividir fácilmente en componentes más pequeños que puedan ser distribuidos entre múltiples servidores.

CÓDIGO EXTENSO Y COMPLEJOS

A medida que la aplicación crece, el código puede volverse muy grande y complejo, dificultando el mantenimiento y la evolución.

DEPENDENCIA FUERTEMENTE ACOPLADAS

Los componentes están estrechamente acoplados, lo que puede complicar los cambios y actualizaciones en partes específicas del sistema.

TIEMPO DE DESPLIEGUE

Las aplicaciones grandes pueden tomar mucho tiempo para desplegarse, lo que puede causar tiempos de inactividad significativos.

RESTRICCIONES TECNOLÓGICAS

Dado que todos los módulos están en un solo proyecto, todos deben usar la misma tecnología, lo que puede limitar la capacidad de utilizar herramientas o lenguajes más adecuados para ciertos componentes.

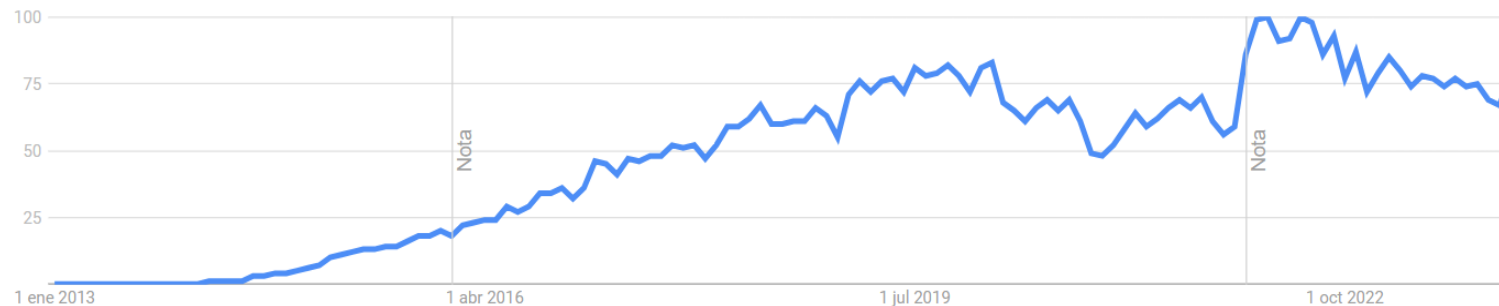
A black and white photograph of a woman in a server room. She is standing in a narrow aisle between rows of server racks. She is holding a tablet in her left hand and pointing with her right index finger at a small screen or panel on one of the server racks. The server racks are filled with equipment, and some fans are visible. The room is dimly lit, with bright light coming from the ceiling. The overall mood is professional and technical.

03

Transición de Monolítico a Microservicios

• El boom de los Microservicios

El modelo de **Arquitectura de Microservicios** (Microservices Architecture) es un nuevo término que se ha hecho conocido poco a poco en los últimos años en la comunidad del desarrollo de software. El término fue creado por un grupo de arquitectos de software en 2012, pero recién en 2014 empezó hacerse conocido, cuando **Martín Fowler**, desarrollador de software y autor conocido de la comunidad, comenzó a usar el término en algunas de sus publicaciones de su página web. En la siguiente figura podemos notar cómo fue el crecimiento del término en Google.



Búsqueda de microservicios en el mundo a través del tiempo 2013–2024. Fuente: Google Trends, 2024

• El inicio de una era de Microservicios

*¿Qué eventos crees que pasaron en **Chile** y en el **mundo**?*



Búsqueda de microservicios en Chile a través del tiempo 2013–2024. Fuente: Google Trends, 2024

• La Pandemia y el Auge de los Microservicios

La pandemia del **COVID-19** marcó un punto de inflexión en el desarrollo de aplicaciones, especialmente en los sectores de delivery y mapas de zonas. La necesidad de adaptarse rápidamente a las nuevas demandas del mercado impulsó a muchas empresas a adoptar arquitecturas de microservicios para mejorar la integración y la experiencia del usuario.



• La Pandemia y el Auge de los Microservicios

Impacto en el Desarrollo de Aplicaciones

Durante la pandemia, las aplicaciones móviles se convirtieron en un pilar fundamental para la continuidad de miles de negocios. La rápida adopción de microservicios permitió a estas aplicaciones.



ESCALAR RÁPIDAMENTE

Las aplicaciones de delivery y servicios relacionados pudieron manejar un aumento significativo en la demanda, escalando sus servicios de manera eficiente.



INTEGRARSE CON OTROS SERVICIOS

Los microservicios facilitaron la integración con servicios de terceros, como sistemas de pago, seguimiento de pedidos y notificaciones en tiempo real.



MEJORAR LA RESILENCIA Y LA FLEXIBILIDAD

La arquitectura de microservicios permitió a las empresas realizar actualizaciones y mejoras sin interrumpir el servicio, adaptándose rápidamente a los cambios del mercado.

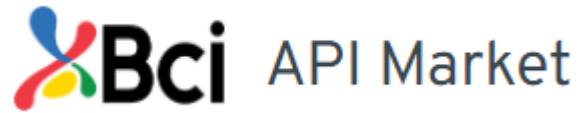
• La Pandemia y el Auge de los Microservicios

Empresas de todo el mundo, desde pequeñas **startups** hasta gigantes tecnológicos, adoptaron **microservicios** para responder a las nuevas necesidades. Aplicaciones de delivery, mapas interactivos de zonas de riesgo y servicios de telemedicina son solo algunos ejemplos de cómo los microservicios han sido fundamentales durante la pandemia.



La pandemia aceleró la adopción de nuevas tecnologías en las PyMEs, como la automatización de procesos (35%) y la capacitación en nuevas habilidades (36%)

• Empresas y Startup que utilizan microservicios



Developers



• ACTIVIDAD

Desarrolla la actividad sobre diferentes arquitecturas de software, donde deberás investigar y escribir sus definiciones.

Comparte tus resultados con tus compañeros.



Reflexionemos



- **¿Cuál es el principal beneficio de utilizar una arquitectura de microservicios en lugar de un monolito?**
- **¿Cómo se manejan los datos en un monolito?**
- **¿Qué estrategias se utilizan para garantizar la consistencia y la integridad de los datos?**
- **¿Qué tipo de monolito es más adecuado para proyectos pequeños y qué tipo de microservicios es más adecuado para proyectos grandes?**

04

¡Muchas gracias!

DuocUC[®]

CERCANÍA. LIDERAZGO. FUTURO.

duoc.cl

7 AÑOS
ACREDITADO



DESDE AGOSTO 2017 HASTA AGOSTO 2024.
DOCENCIA DE PREGRADO. GESTIÓN
INSTITUCIONAL. VINCULACIÓN CON EL MEDIO.