



El futuro digital  
es de todos

MinTIC



‘Mision  
TIC 2022’

01

# Arquitectura de Microservicios

Ciclo 4a:

Desarrollo de aplicaciones web



El futuro digital  
es de todos

MinTIC

# Objetivo de Aprendizaje

**Identificar** las principales **características** de una **arquitectura de microservicios**.



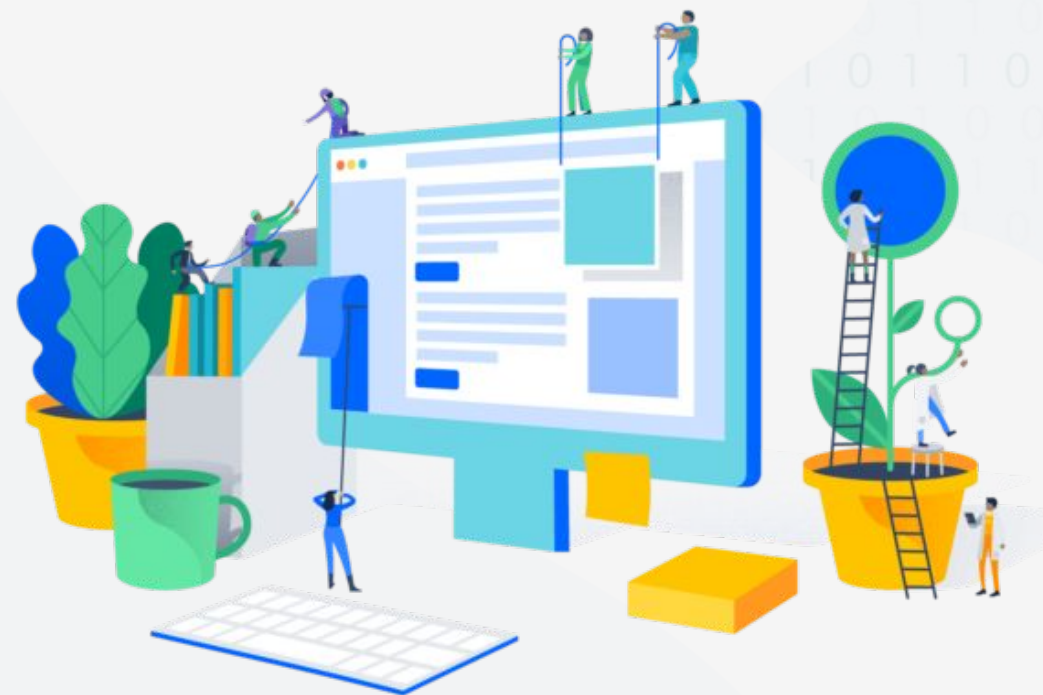
El futuro digital  
es de todos

MinTIC

# Arquitectura: Definición

La **arquitectura** de un sistema de software es el conjunto de **elementos, relaciones y propiedades** que constituyen la **estructura** del sistema, y la cual es la base para **razonar** acerca del mismo.

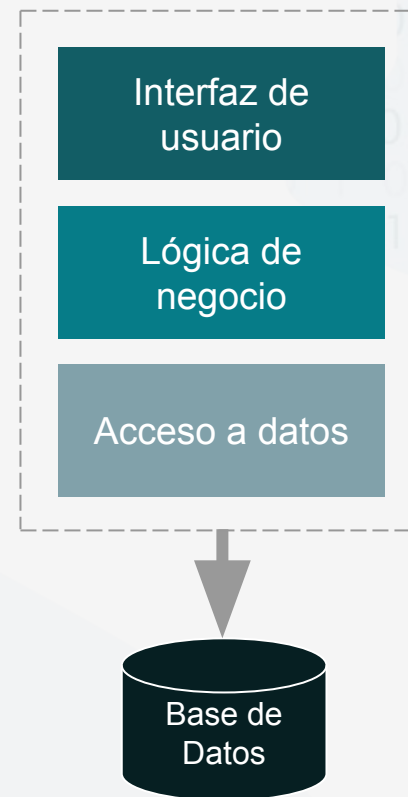
En la arquitectura se incluyen aspectos como la **estructura** de los **componentes**, la **funcionalidad** de cada uno de ellos, y sus formas de **comunicación**.





# Arquitectura Monolítica

La arquitectura **monolítica** es aquella en la que el software se estructura de forma tal que sus **funcionalidades** quedan **acopladas** y sujetas a un **mismo programa** (un único **componente**). Esto implica que toda funcionalidad es **dependiente** de las demás y que toda la información necesaria para el funcionamiento del sistema es alojada en un **único servidor**.



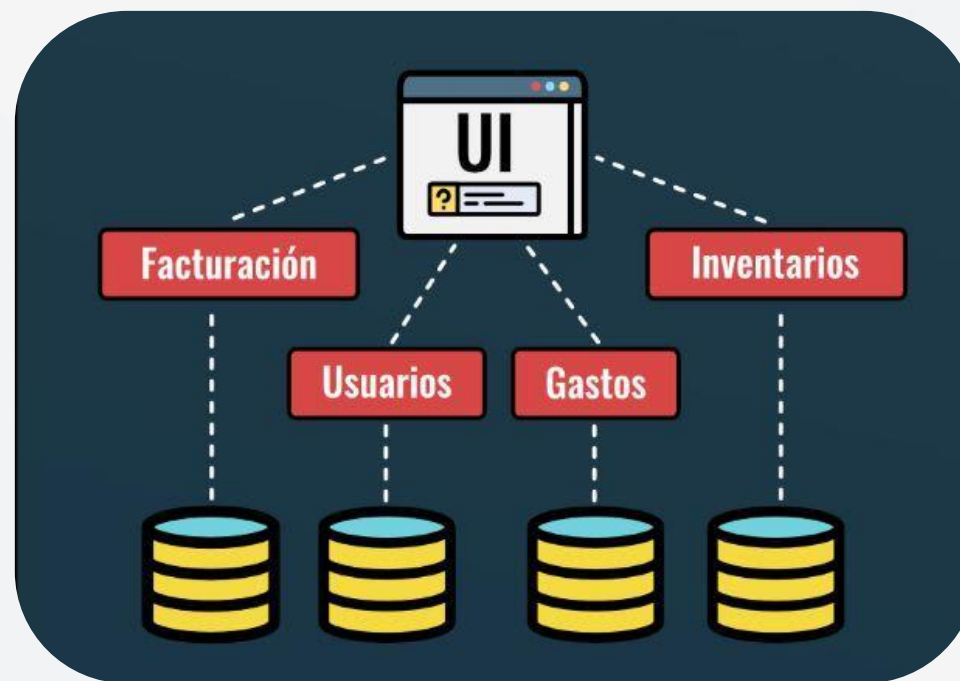


El futuro digital  
es de todos

MinTIC

# Arquitectura de Microservicios

La arquitectura de **microservicios** es un tipo de **arquitectura distribuida** en la que una aplicación se construye como un conjunto de pequeñas **aplicaciones autónomas e independientes** (microservicios), que se **comunican** con mecanismos ligeros para proporcionar las **funcionalidades** del sistema.



[Imagen] ¿Qué es la arquitectura de microservicios? (s. f.). [JPG]. Ed team.

[https://www.google.com/url?sa=i&url=https%3A%2F%2Fed.team%2Fcomunidad%2Fque-es-la-arquitectura-de-microservicios&psig=AQvVaw1FvSazTZFHMyty\\_n2DD-Y1&ust=1626280090724000&source=images&cd=vfe&ved=0CAcQIRxgFwo](https://www.google.com/url?sa=i&url=https%3A%2F%2Fed.team%2Fcomunidad%2Fque-es-la-arquitectura-de-microservicios&psig=AQvVaw1FvSazTZFHMyty_n2DD-Y1&ust=1626280090724000&source=images&cd=vfe&ved=0CAcQIRxgFwo)

TCLCn1Mm74PECFQAAAAAAdAAAAABAD





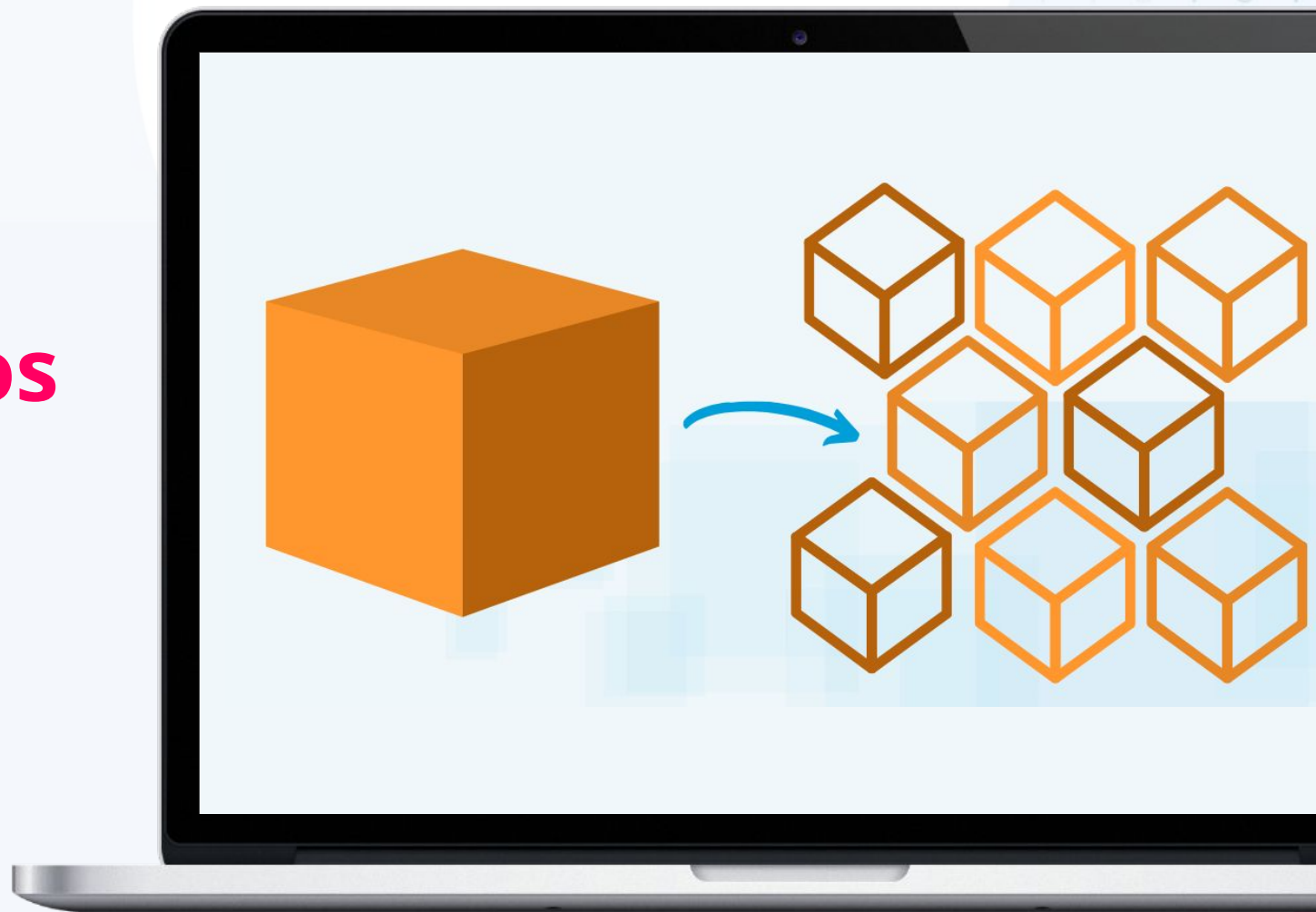


El futuro digital  
es de todos

MinTIC

# Características de los Microservicios

Para entender **por qué** la arquitectura de microservicios es una de las más utilizadas actualmente, es importante entender sus particularidades.





El futuro digital  
es de todos

MinTIC

# Microservicios: Enfocados

Cada microservicio es un **componente** del sistema que desempeña una **función específica**, y cuyo código fuente puede estar escrito en un **lenguaje de programación** diferente a los utilizados en los demás componentes de la aplicación.





# Microservicios: Autónomos

Todos los microservicios son **independientes**, lo que significa que cada uno de ellos puede ser modificado y desplegado **sin afectar** a las demás funciones o servicios del sistema.

Esto permite la construcción de un sistema con una **infraestructura adaptable y flexible**, donde no es necesario alterar toda la infraestructura para modificar una función o un servicio.







El futuro digital  
es de todos

MinTIC

# Microservicios: Almacenamiento

Para evitar la sobrecarga y caída del sistema, todo microservicio debe contar con un **sistema de almacenamiento propio**. Este se puede presentar en bases de datos **relacionales** o **NoSQL**, de acuerdo con los requisitos del software y con la lógica de negocio.



[Imagen] Database table icon. (s. f.). [PNG]. Pngwing. <https://w7.pngwing.com/pngs/13/363/png-transparent-computer-icons-database-table-sql-others-blue-data-big-data.png>

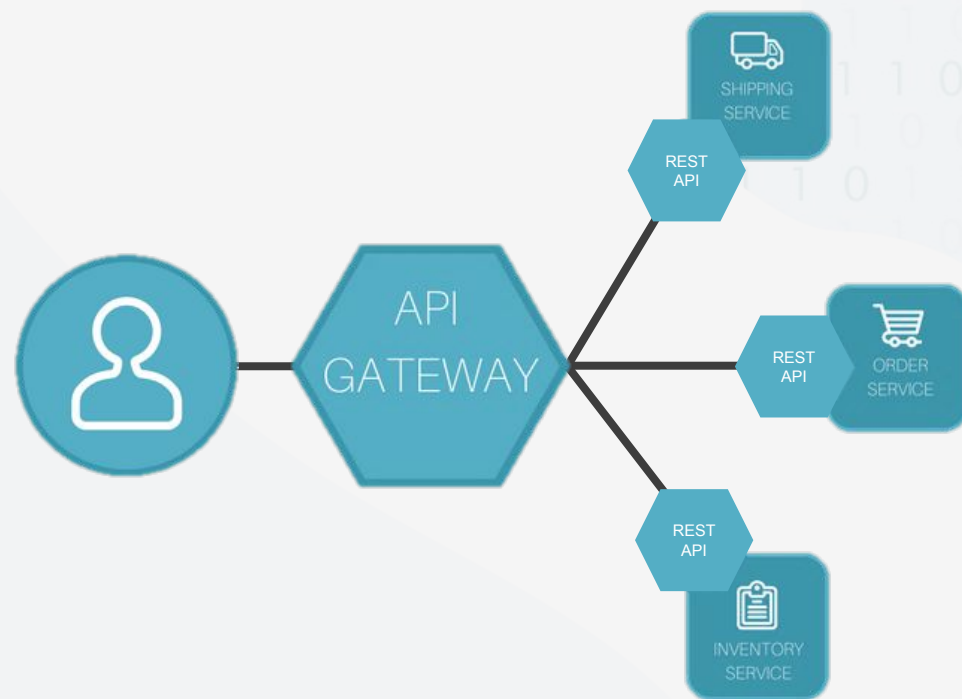




# Microservicios: Comunicados

Debido a que los microservicios deben comunicarse utilizando mecanismos ligeros, comúnmente se **exponen** sus funcionalidades utilizando **conectores REST**.

A su vez, para mantener la autonomía de cada microservicio, dichas funcionalidades son consumidas a través de un **API Gateway**. Este componente se encarga de recibir y responder las peticiones del usuario, y de **comunicar** los microservicios entre sí, según sea necesario para ejecutar la petición.

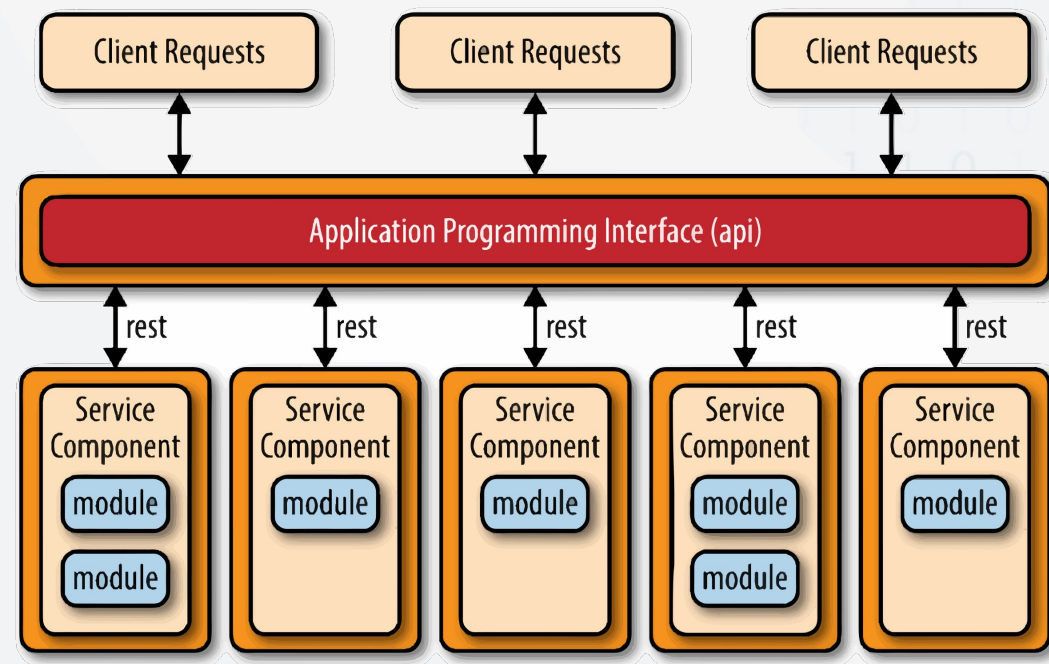




# Microservicios: Peticiones

El flujo de una petición a una arquitectura de microservicios, es el siguiente:

- 1) El cliente realiza la **petición** a la API.
- 2) La API se **comunica** con los **microservicios** necesarios para ejecutar la **funcionalidad** solicitada.
- 3) Cada microservicio ejecuta su **función** y **responde** a la API con la información adecuada.
- 4) La API **organiza** las respuestas obtenidas de los microservicios y **responde** la petición al cliente.





El futuro digital  
es de todos

MinTIC

# VENTAJAS

El uso de la arquitectura de microservicios en una aplicación, trae algunos **beneficios** que facilitan el trabajo en el sistema.





El futuro digital  
es de todos

MinTIC

# Ventajas: Heterogeneidad

La **autonomía** que poseen por defecto los microservicios, permite que cada uno sea **tecnológicamente independiente** de los demás. Esto posibilita a su vez que cada microservicio pueda funcionar bajo la tecnología que mejor resultados le brinda, **sin perjudicar** al sistema.

Esta **heterogeneidad** es posible gracias a los **mecanismos de conexión**, que se establecen en el diseño de una arquitectura de microservicios.

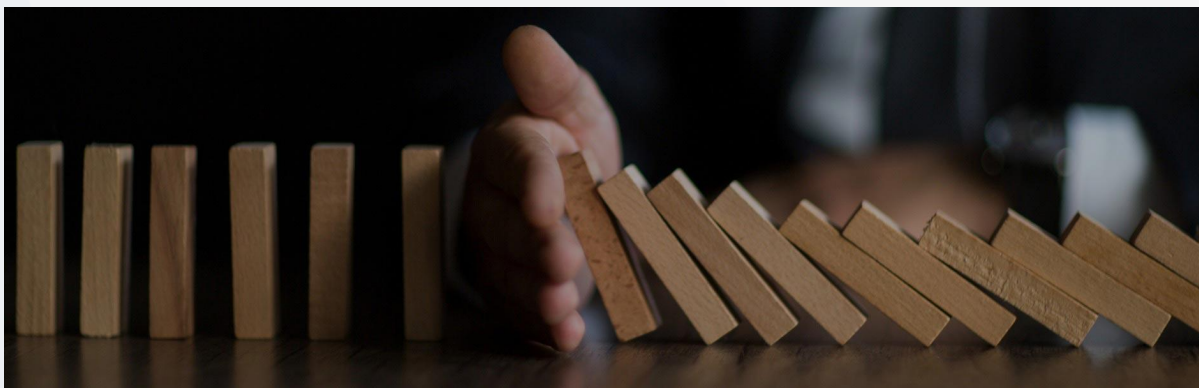






# Ventajas: Tolerancia a Fallos

Normalmente cuando se trabaja con **arquitecturas monolíticas** (un único **componente**), si una de las **funcionalidades** de la aplicación **falla**, todo el sistema fallará.



En cambio, en una **arquitectura de microservicios**, si un **componente falla**, los **demás** podrán seguir **funcionando** ya que existe una separación tanto **lógica**, como **física**.



El futuro digital  
es de todos

MinTIC

# Ventajas: Escalabilidad

Cuando las **funcionalidades** de una aplicación **reciben más peticiones** de las esperadas, se aplican estrategias de **escalabilidad**: duplicar los servicios o desplegarlos en mejores máquinas.

En una arquitectura **monolítica** la escalabilidad puede ser **compleja**, pues al estar todas las **funcionalidades integradas**, de manera implícita se asignan recursos a funcionalidades que **no los necesitan**.





El futuro digital  
es de todos

MinTIC

# Ventajas: Escalabilidad

En una arquitectura de **microservicios** se puede detectar **cuáles** funcionalidades necesitan ser escaladas y **cuáles no**, lo cual permite **asignar recursos** únicamente a los microservicios que lo **necesitan**.

Esto **facilita** las estrategias de escalabilidad y **reduce los costos** de ejecución del sistema.





El futuro digital  
es de todos

MinTIC

# DESVENTAJAS

El uso de la arquitectura de microservicios también puede traer algunas **complicaciones**.





# Desventajas: Separación

La arquitectura de microservicios provee una **separación física** de las **funcionalidades** (cada una se implementa en un microservicio), pero en algunas ocasiones pueden ser **complicado separar** las funcionalidades. En estos casos se es muy propenso a generar una **dependencia lógica** (a pesar de estar separada una funcionalidad, esta depende totalmente de otra para funcionar).

Por esta razón, para aprovechar las ventajas de los microservicios se debe **evitar** cualquier **dependencia lógica** entre componentes.







El futuro digital  
es de todos

MinTIC

# Desventajas: Trabajo Extra



Pasar de trabajar con **una sola tecnología** en un monolito, a trabajar con **muchas tecnologías**, requiere un **esfuerzo** extra por parte del equipo para adoptar estas tecnologías.

Se debe **planificar** ese gasto extra de tiempo. De no hacerlo, se puede **saturar** al equipo de trabajo y generar **retrasos** en la entrega del producto.

[Imagen] Trabajo Extra. (s. f.). [Fotografía]. [https://www.in-mind.org/sites/default/files/pictures/work\\_picin\\_mind\\_blog\\_0.jpeg](https://www.in-mind.org/sites/default/files/pictures/work_picin_mind_blog_0.jpeg)





# Desventajas: Comunicación

Para que un **sistema** diseñado bajo una arquitectura de microservicios **funcione** de manera **correcta**, debe existir una buena **comunicación** entre los **equipos** o personas encargadas de diseñar cada uno de los microservicios. De lo contrario, se tendrán servicios que **funcionan perfectamente** de manera independiente, pero en conjunto **no cumplen el objetivo** del sistema.





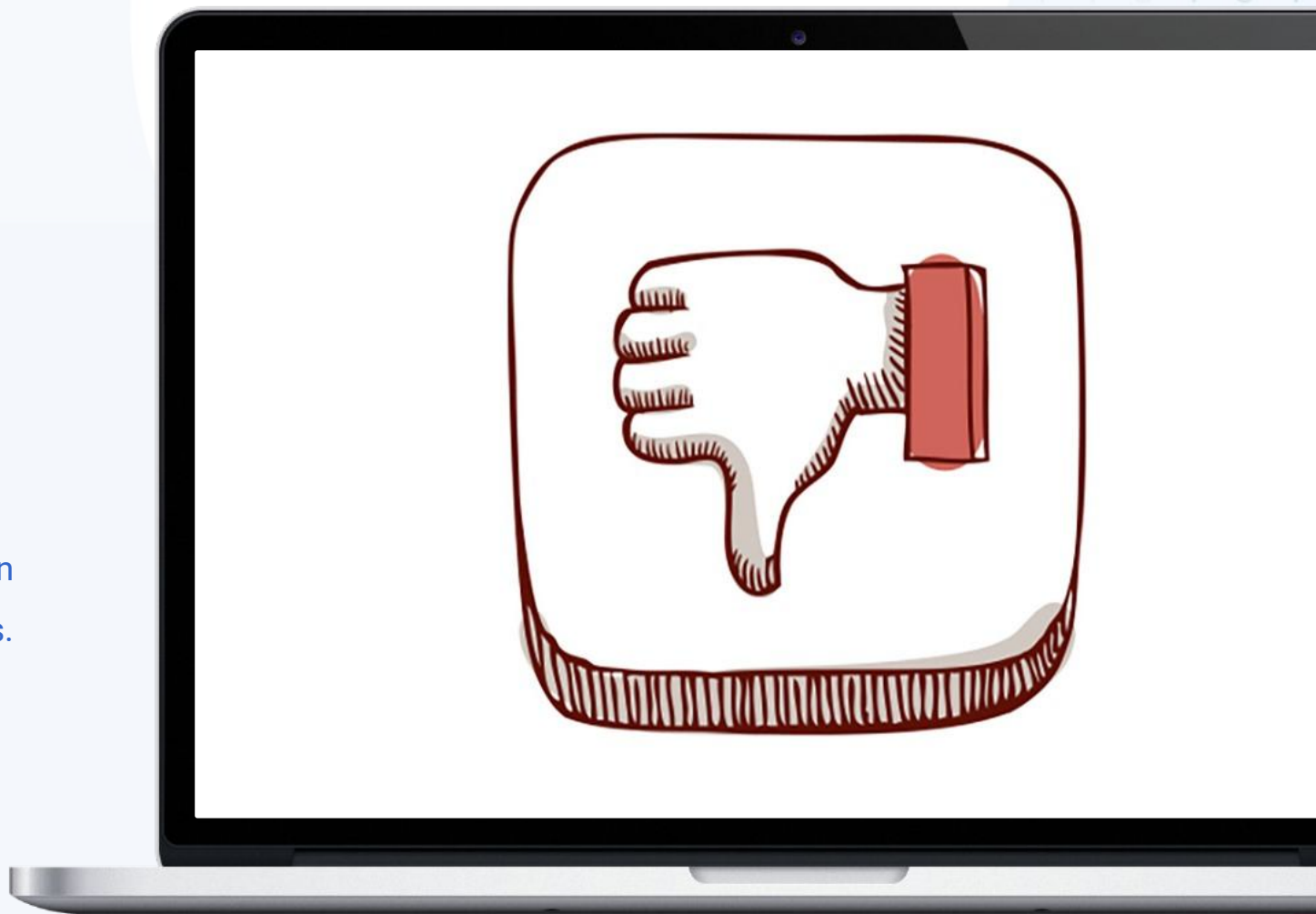
El futuro digital  
es de todos

MinTIC

# MALAS PRÁCTICAS

Existen **malas prácticas** que se presentan comúnmente en la implementación de una arquitectura de microservicios.

Estas se **pueden y deben** evitar.





# Mala Práctica: Mega-Servicios

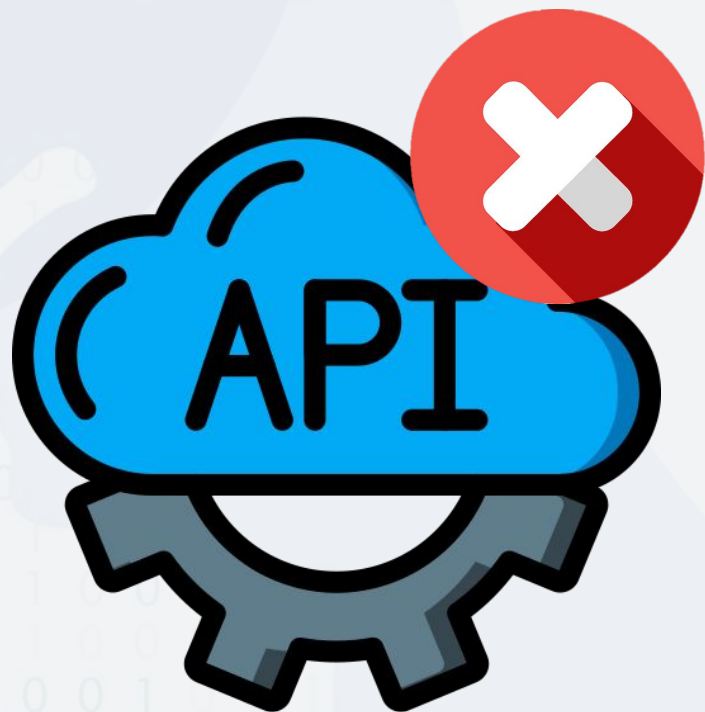
Como todo sistema, una arquitectura basada en microservicios **crece** y requiere **implementar nuevas funcionalidades**. En un primer momento resulta exagerado crear nuevos microservicios, por lo cual las nuevas funcionalidades **se agregan a microservicios existentes**.

Con el tiempo, los microservicios se convierten en **mega-servicios** con **decenas de funcionalidades**, desaprovechando las ventajas de la arquitectura.





# Mala Práctica: Sin API Gateway



En algunos casos, puede resultar interesante la idea de **eliminar el API Gateway**, ya que los microservicios fueron separados correctamente y el **cliente** puede **consumir** cada uno de manera **independiente**.

Esto no es una buena idea, ya que obliga al cliente a **estar al tanto del cambio** en cada uno de los microservicios. En otras palabras, **delega** al cliente las **responsabilidades** del API Gateway, lo cual puede **generar problemas** con funcionalidades que dependen de dos o más microservicios.





# Mala Práctica: Datos Compartidos

Uno de los **beneficios** de los microservicios es su **independencia** de los demás. Esto permite que cada uno maneje los datos a su manera, haciendo el uso de la tecnología que desee y del esquema que mejor se adapte.

Esta independencia **se pierde** cuando dos o más microservicios **se conectan a la misma base de datos**. Por lo tanto, se debe **evitar** esta mala práctica.





El futuro digital  
es de todos

MinTIC

# CASOS REALES

Tanto compañías pequeñas como compañías de talla mundial utilizan los **microservicios** en sus negocios.





El futuro digital  
es de todos

MinTIC

# Casos Reales: Empresas

A día de hoy **muchas empresas** diseñan sus sistemas usando arquitecturas **basadas en microservicios**, algunas de las compañías más relevantes son: Netflix, Uber, Spotify, Amazon, eBay y SoundCloud.

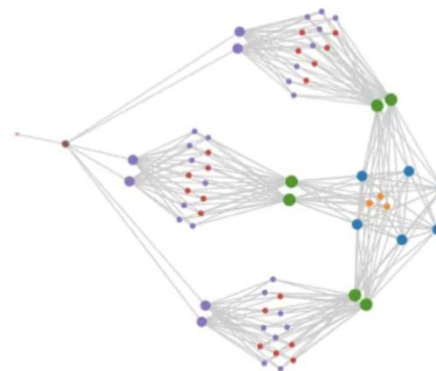




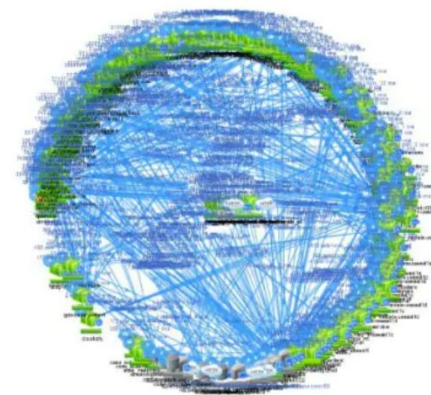
# Casos Reales: Netflix

Uno de los casos más sorprendentes es **Netflix**, su arquitectura **basada en microservicios** es muy **compleja** y puede ser **intimidante** incluso para arquitectos experimentados.

Sin embargo, en el fondo de esa arquitectura compleja, se sigue un **principio básico**: el de tener **microservicios** que **funcionan** de manera **conjunta**.



Simplified Architecture



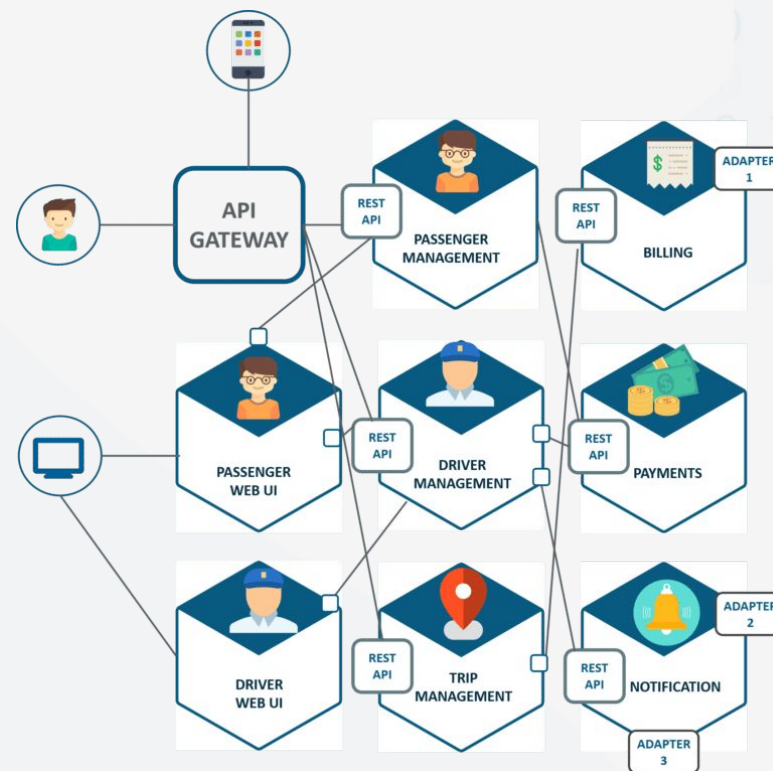
Actual Architecture



# Casos Reales: Uber

Otro caso que aplica de manera **interesante** los conceptos de microservicios es el de **Uber**.

En el esquema de la derecha se pueden evidenciar algunos conceptos vistos anteriormente, como el **API Gateway**, los distintos **microservicios** con sus respectivas **funcionalidades** y las **conexiones REST**.







# Casos Reales: Uber

Previamente **Uber** usaba una **arquitectura basada** en un **monolito**. Esta resultaba **tediosa** de **manejar** y **evolucionar**. A la derecha se puede evidenciar el esquema de la anterior arquitectura de Uber.

