



códigofacilito



# Microservicios

Bootcamp – Backend Avanzado

Eduardo I. García Pérez.



# 📋 Objetivos.

>\_ ¿Qué son los microservicios?

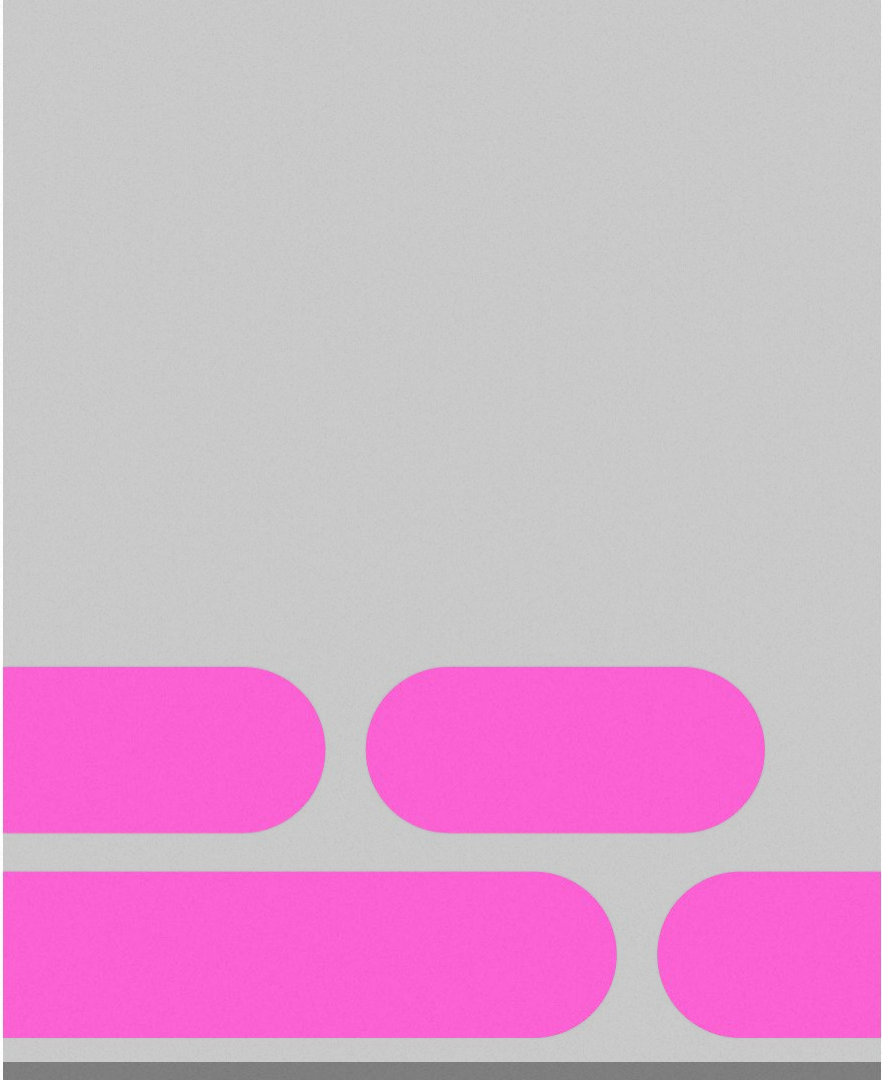
>\_ ¿Vale la pena, o no, utilizar microservicios?

>\_ Microservicios o Monolitos.





**Hablemos de monolitos.**

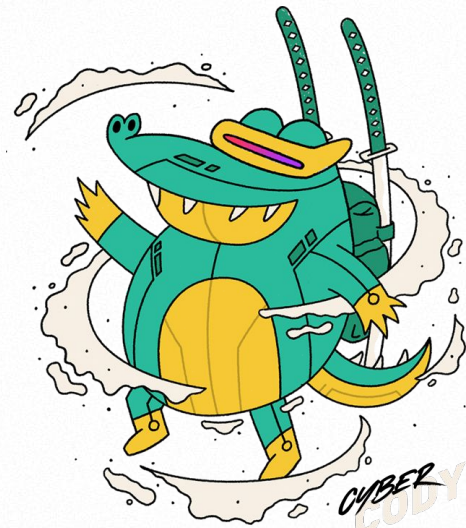




## >\_ Monolitos.

Un monolito es un sistema (tradicional) desarrollado y desplegado como una sola entidad.

Todos los componente se encuentran ejecutándose en una sola base de código y en una misma instancia.







## >\_ Pros Monolitos.

- El stack de tecnologías suele ser limitado.
- Fácil de desplegar.
- Fácil de monitorear.
- Sin preocupación de la latencia.
- Todo se encuentra en una misma base de código.\*





## >\_ Cons Monolitos. ❌

- La escalabilidad es costosa para ciertas áreas.
- Tecnologías limitadas.
- Continuous Delivery (Pequeños cambios pueden romper todo).
- Difícil de modificar.\*





## >\_ Ejemplo. 🗿

- Django App para listado de artículos.
- Nueva pasarela de pagos.
- Envío de libros en formato físicos.
- Creación de un API.



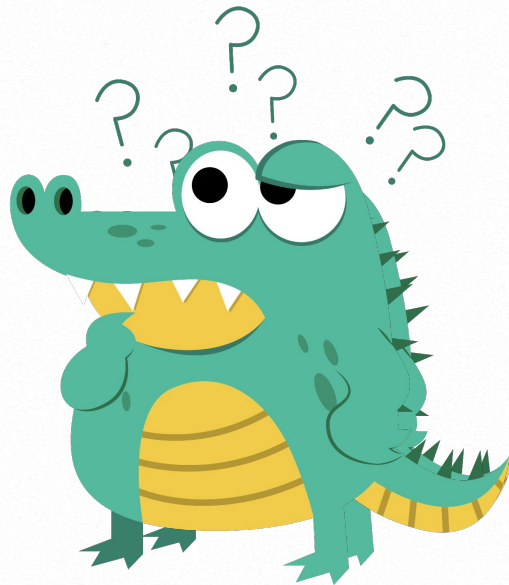




## >\_ ¿Cuando usar molitos? 🐸

*Todas las aplicaciones comienzan siendo monolitos.*

- Cuando los recursos de la empresa son pocos.
  - Tiempo de desarrollo limitado.
  - Equipo de desarrollo, Q&A y Devops pequeño.
  - Cuando se tiene poco dinero \$.



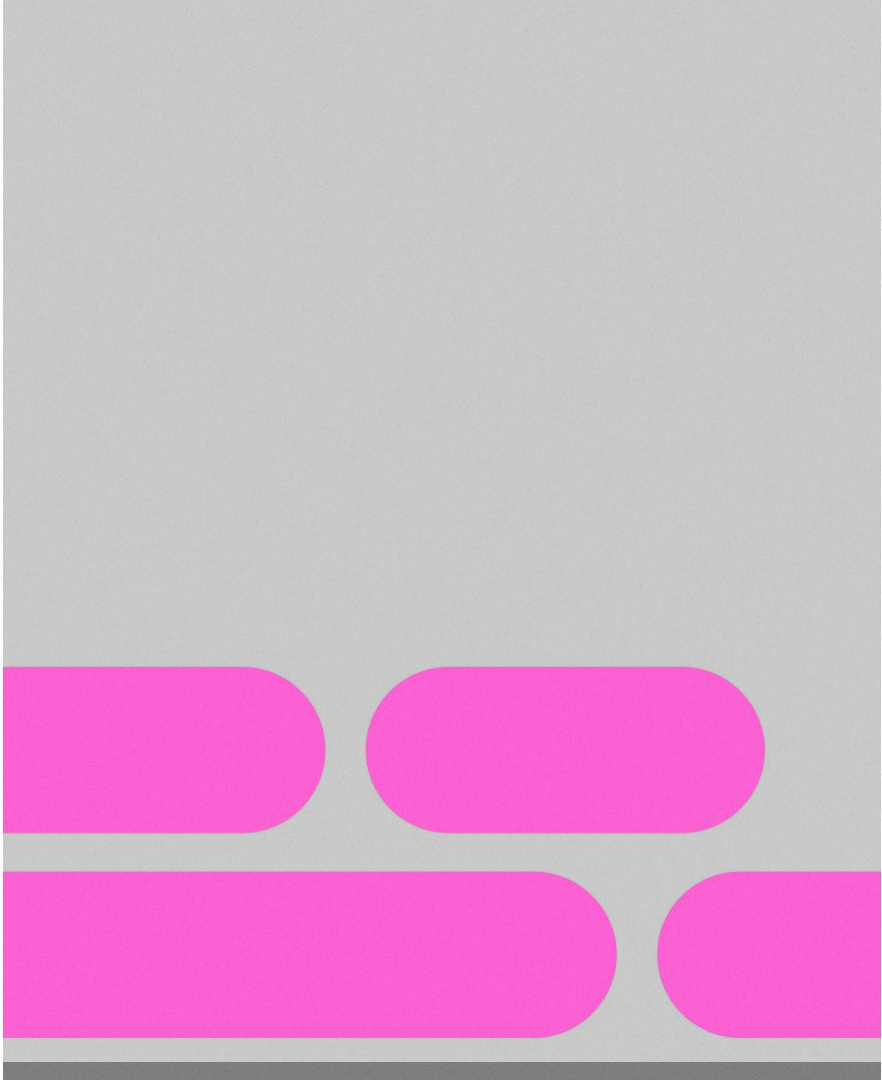




**Analogía.**



**¿Qué son los  
microservicios?**







## >\_ **Microservices.** 🤠

- Los microservicios son una arquitectura de Software en la que se desarrollan un conjunto de pequeñas aplicaciones (servicios) independientes.
- Estos servicios son ejecutados en sus propios procesos así como con sus propios recursos.
- Comúnmente son aplicaciones que se ejecutan sobre HTTP a través de APIs.







**Analogía.**



## >\_ Pros Microservices.

- Cada aplicación es independiente de otras.
- Escalabilidad.
- Flexibilidad en cuanto a las tecnologías.
- Información descentralizada.







## >\_ **Cons Microservices.** ❌

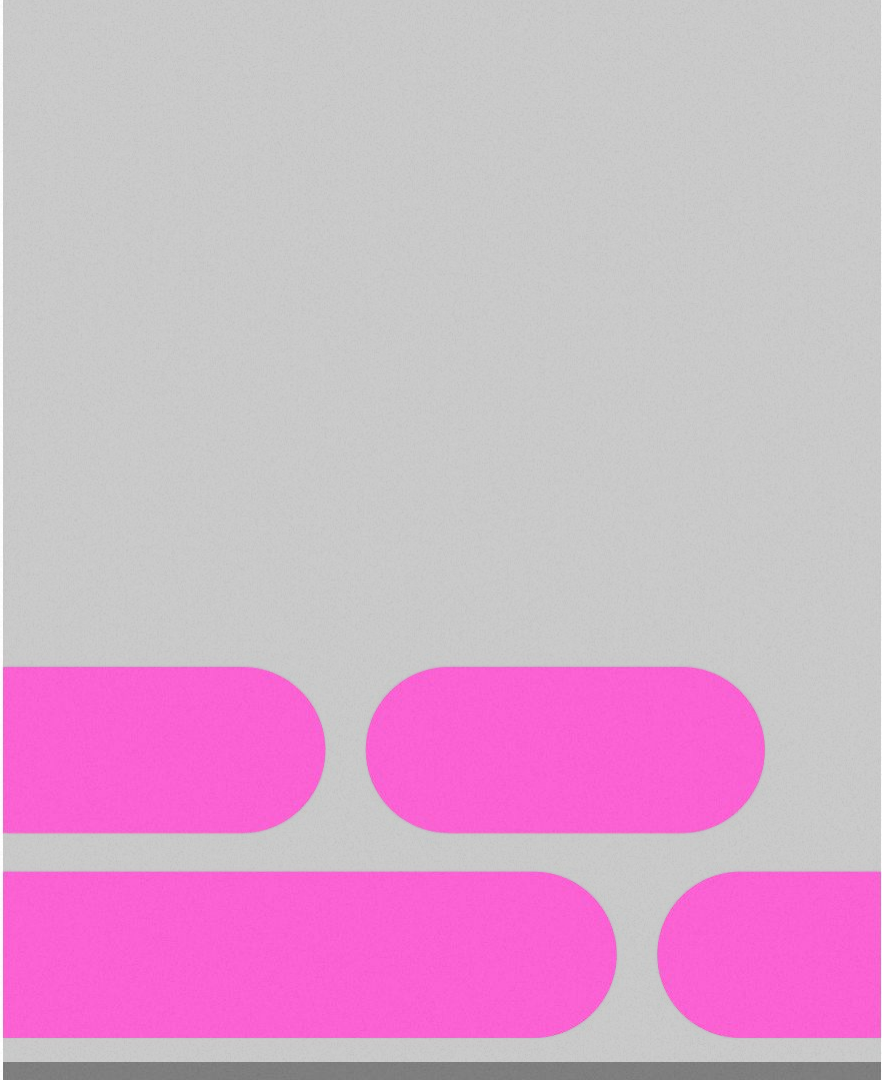
- Complejidad en cuanto a la comunicación.
- Complejidad para sincronizar los datos.
- Latencia en la red.
- Robusto monitoreo de los servicios.
- Versionamiento.
- Costos de mantenimiento.
- Mayor cantidad de personas en desarrollo.
- Seguridad.
  - SSH.







## ¿Monolitos o microservicios?





## >\_ Ambos.

- El uso de microservicios en conjunto con monolitos no están peleados.







**Analogía.**





## >\_ Ejemplo.

- React client.
- Django server.

# django





códigofacilito



# Microservicios

Bootcamp – Backend Avanzado

Eduardo I. García Pérez.

