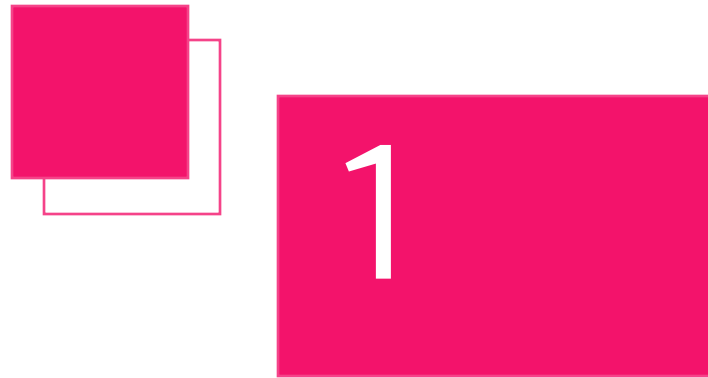


codehouse  
{ ACADEMY }



¿Qué es DevOps?

# Agenda

1. ¿Qué es DevOps?
2. ¿Qué no es DevOps?
3. ¿Por qué es importante?
4. ¿Cómo podemos aplicarlo?
5. Elementos en DevOps

¿Qué es DevOps?

## 1. ¿Qué es DevOps?

Su definición es: **la unión de personas, procesos y productos para hacer posible la entrega continua de valor a nuestros usuarios finales.**

Es la creación de equipos multidisciplinares para el trabajo con herramientas comunes y prácticas eficientes.

¿Qué es DevOps?

## 2. ¿Qué no es DevOps?

Existen ciertas confusiones con el término, por eso es preciso aclarar qué **no** es DevOps.

DevOps no es:

- Una herramienta
- Una aplicación
- Un equipo
- Un rol que una persona ejerce

¿Qué es DevOps?

### 3. ¿Por qué es importante?

En nuestra industria, todo evoluciona muy deprisa. Es fácil quedarnos atrás, dejar de innovar, atascarnos en nuestros procesos internos.

DevOps trata de conseguir la **entrega continua de valor al cliente**.

Conseguiremos ser competitivos y mejorar como productores de software.

¿Qué es DevOps?

## 4. ¿Cómo podemos aplicarlo?

La aplicación de prácticas, metodología o cultura DevOps **no es sencilla**.

Es un cambio notable en la forma en que nuestros equipos trabajan, y debe ser integrado poco a poco para tener éxito.

Por lo que, siguiendo estas prácticas, conseguiremos:

- Flujo constante de mejora
- Mínimos de calidad cumplidos
- Agilidad y eficiencia de desarrollo
- Entregas constantes

¿Qué es DevOps?

## 5. Elementos en DevOps

Puesto que ya sabemos que DevOps es una metodología creada para ser compartida para todo el equipo, ¿qué elementos necesito para empezar a implementarla?

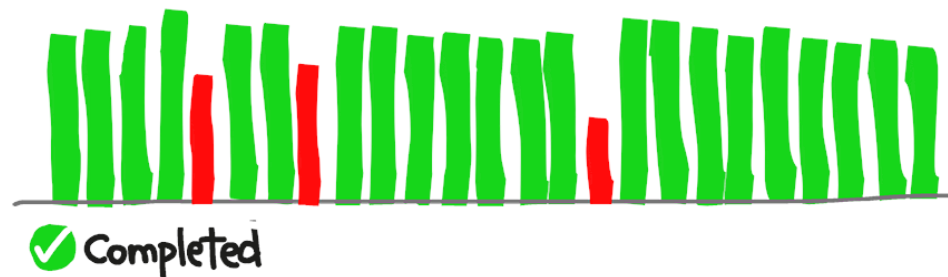


¿Qué es DevOps?

## 5.1. CI, Continuous Integration

Es la práctica de incorporar y *testear* el código nuevo en nuestra aplicación, para localizar fallos y errores de forma más rápida y eficiente.

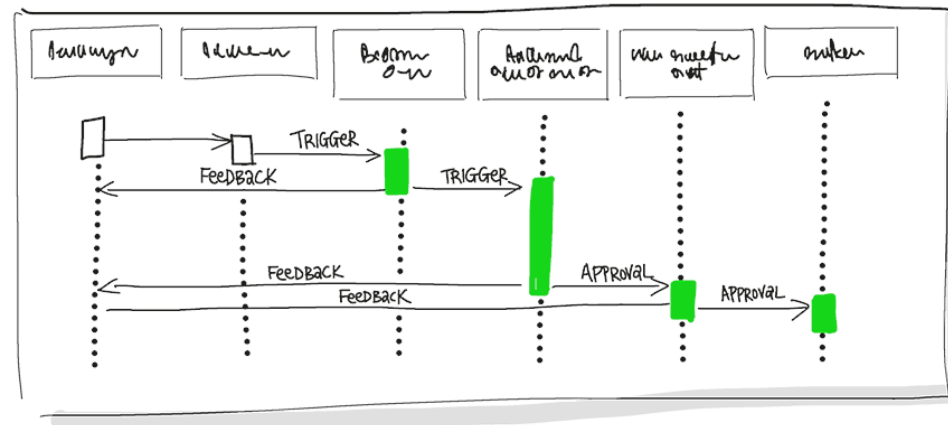
BUILD Succeeded



¿Qué es DevOps?

## 5.2. CD, Continuous Delivery

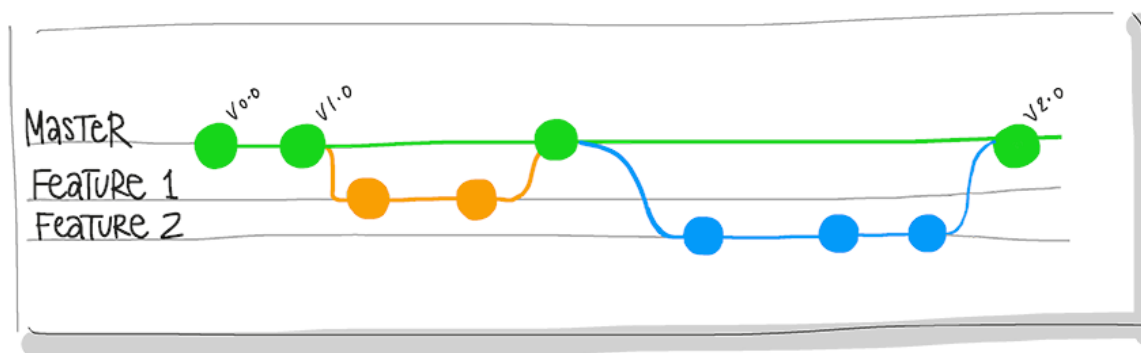
Consiste en la entrega continua de software a producción, además del *testing* de los entornos que manejan los usuarios finales, para mejorarlo de forma constante y encontrar posibles fallos en los mismos.



¿Qué es DevOps?

## 5.3. Control de versiones

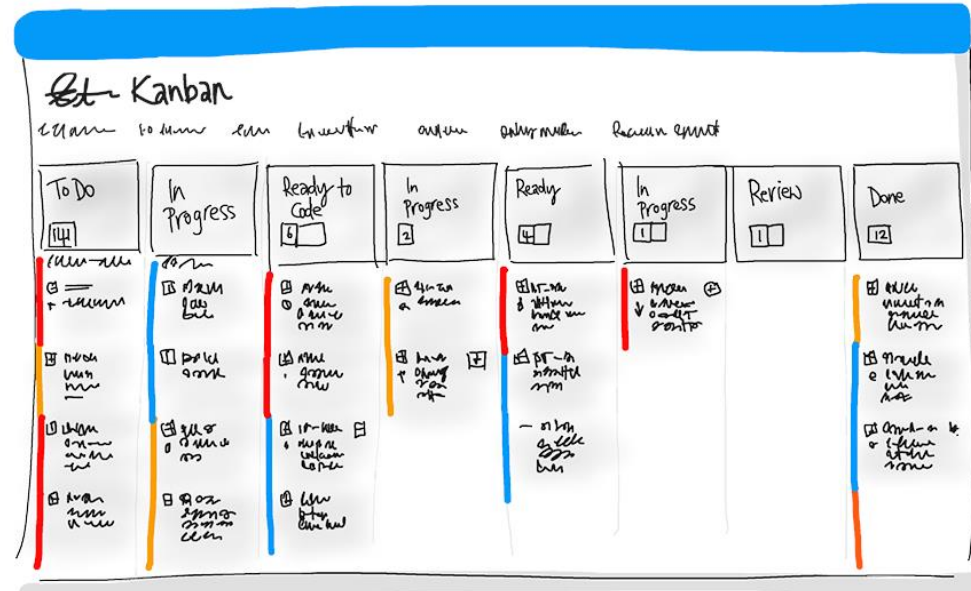
Herramientas para controlar e introducir nuevo código en nuestra aplicación, trabajando con múltiples versiones sin afectar a otras. Ideal para trabajar con equipos a distancia.



¿Qué es DevOps?

## 5.4. Agile y lean project management

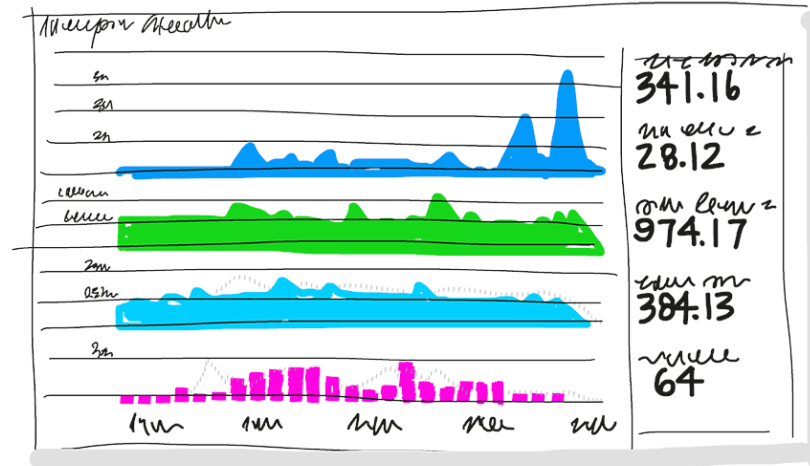
Planificación y organización de tareas en el tiempo, gestionar las capacidades del equipo y adaptarse a los cambios de requisitos en el negocio.



¿Qué es DevOps?

## 5.5. Monitorización de proyectos

Control de salud de los entornos en los que los usuarios finales usan las aplicaciones que construimos, controlando los procesos de uso entre otros parámetros.



## 5.6. Servicios *Cloud*

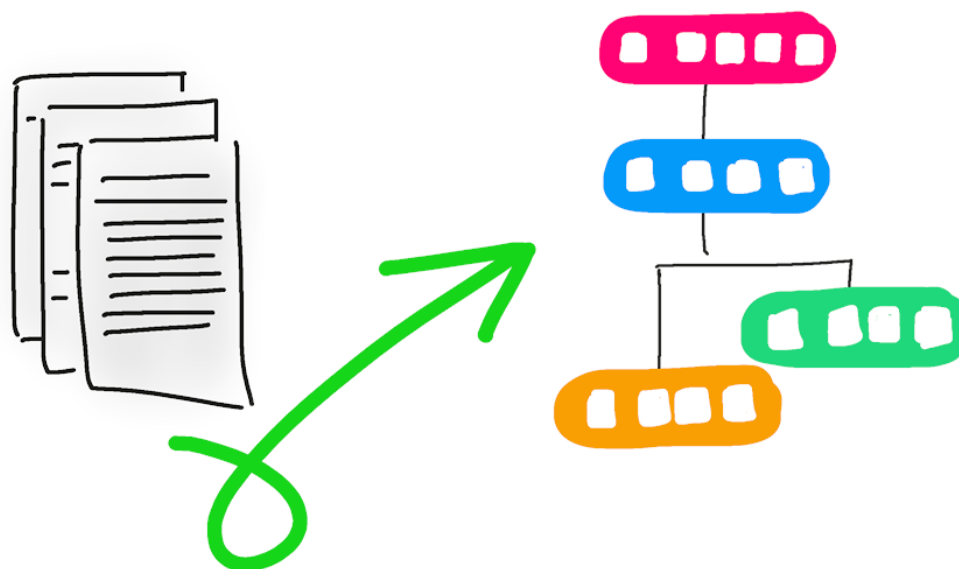
Servicios de almacenamiento, bases de datos, análisis... a través de Internet, ¡en la nube!



¿Qué es DevOps?

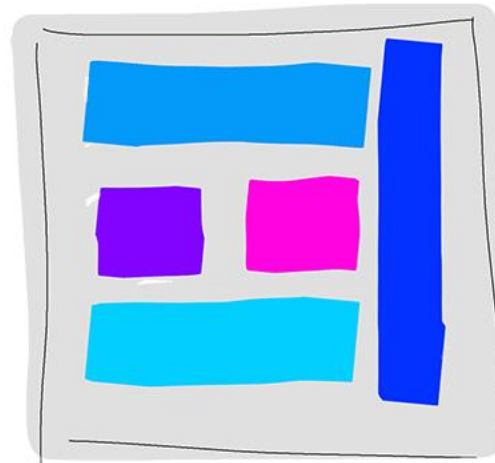
## 5.7. IaC, Infrastructure as Code

Automatización para la gestión de entornos en forma de código, convirtiendo un fichero de especificaciones en una arquitectura donde desplegar.

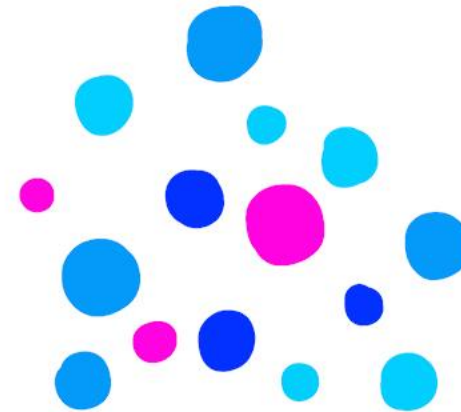


## 5.8. Microservicios

Modelo de arquitectura para aislar el código en servicios reusables.



MONOLITHIC/LAYERED

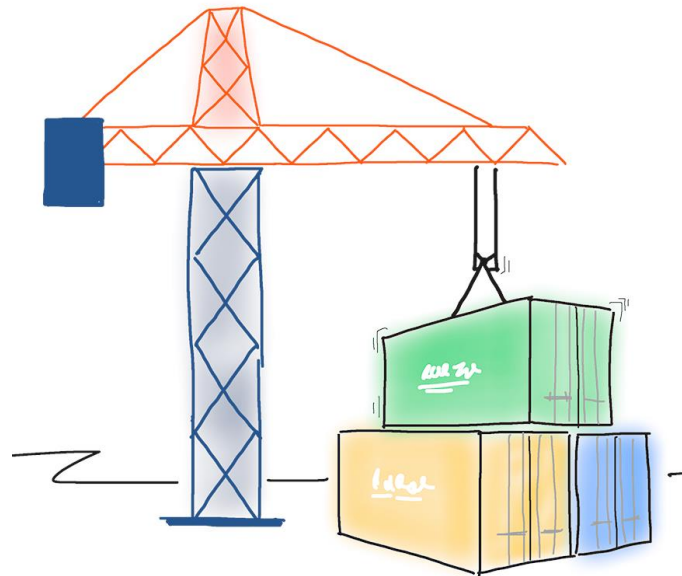


MICROSERVICES



## 5.9. Contenedores

Son un modelo de virtualización de sistemas operativos, para ejecutar en ellos desde microservicios a grandes aplicaciones.



codehouse  
{ ACADEMY }