

Unidad 1 – Introducción a DevOps

En todas las fases del software, detrás de él (developers) o al final de él (end-users), lo que encontramos son personas. Personas que necesitan coordinarse, comunicarse de forma fluida, confiar en las capacidades y en la calidad del producto que poco a poco se va creando, avanzar de forma continua y en conjunto...

Durante muchos años, los procesos de software han ido evolucionando para mejorar y satisfacer estas necesidades que hemos comentado... para al final proporcionar un producto de calidad y un aprendizaje óptimo.

Esta evolución ha desembocado en lo que ahora conocemos como DevOps.

¿Y qué es DevOps?

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

El término es la unión de Dev, desarrollo, y Ops, operaciones. Por lo que el objetivo es crear equipos multidisciplinares que trabajen juntos con herramientas y prácticas eficientes y compartidas.

Es una filosofía, una forma de trabajo para equipos, es el derribo de las barreras entre personas que trabajan en operaciones y las que trabajan en desarrollo. Todo el equipo trabaja con un mismo objetivo y siguiendo la misma metodología.

¿Qué no es?

- Una herramienta
- Una aplicación
- Un rol
- Un equipo

Motivos para implementar DevOps

- Reducir tiempo de ciclos

Alcanzando la metodología 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?

- Integración continua, CI, Continuous Integration

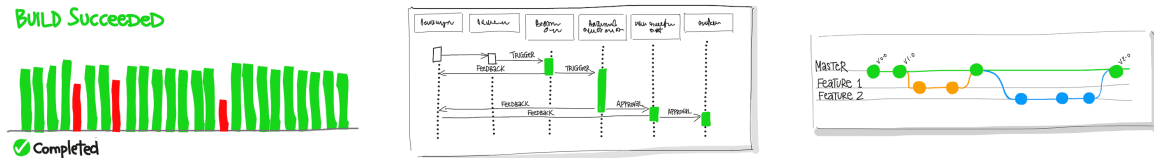
Consiste en el constante mergeo y testeo del código, lo que nos lleva a encontrar fallos y defectos en nuestra aplicación rápidamente. También obtendremos feedback de forma más rápida y eficiente, además de perder menos tiempo en mergeos difíciles y largos.

- Entrega continua, CD, Continuous Delivery

Consiste en la entrega de soluciones de software a producción, además del testeo de entornos que ayuda a los equipos de desarrollo a encontrar bugs de forma veloz, además de adaptar esos entornos a los siempre cambiantes requisitos de negocio.

- Control de versiones, VC, Version Control

Permite a los equipos trabajar de forma remota, además de integrar herramientas para monitorear la actividad como los despliegues. Todos los cambios realizados por todo el equipo están localizados, el código es la misma documentación de los cambios realizados...



- Planning agile y lean project management (gestión de proyectos sin pérdidas)

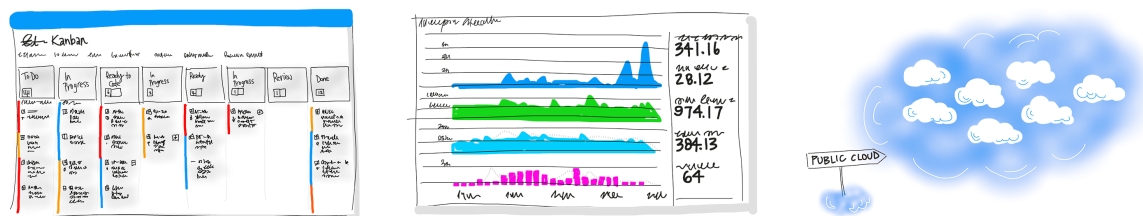
Permite planificar y aislar tareas en sprints, así como gestionar la capacidad del equipo... y ayudar a adaptar los cambios en las necesidades de negocio.

- Monitorización y logging

Consiste en el monitoreo de aplicaciones en entornos de producción, para controlar la salud de la aplicación así como el uso por parte de los usuarios. Permite a las organizaciones validar estrategias y descartar procesos de uso. Esta información se captura y se almacena en formatos de *logging*.

- Cloud públicas e híbridas

La Cloud ha derribado los tradicionales cuellos de botella permitiendo acomodar las infraestructuras. Ya sea que uses Infraestructura como Servicio (IaaS, Infrastructure as a Service) para levantar y modificar tus aplicaciones, o Plataforma como Servicio (PaaS, Platform as a Service) para ganar productividad sin precedentes... la nube te ofrece datacenters sin límites.



- Infraestructura como Código, IaC, Infrastructure as Code.

Es la práctica que permite la automatización y validación de la creación y destrucción de entornos, para ayudar a entregar aplicaciones de forma estable y segura.

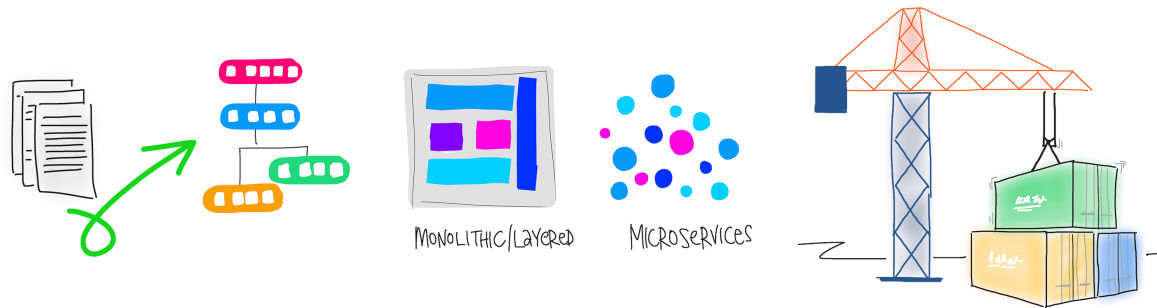
- Microservicios

Es una arquitectura que promueve el aislamiento de casos de uso de negocio en pequeños servicios reusables, comunicándolos a través de contratos de interfaz. Siguiendo esta

arquitectura conseguimos escalabilidad y eficiencia.

- Contenedores, containers

El futuro de la virtualización. Siendo mucho más ligeras que las máquinas virtuales, permiten una rápida hidratación y una configuración sencilla a través de ficheros.



Todos los inicios son complicados

"Si duele, hazlo más a menudo".

Implementar metodología DevOps no es un proceso de un día, ni siquiera de meses. No por implementar un par de estas metodologías podemos parar y decir "ya somos DevOps".

Es un trabajo continuo, diario, unas prácticas con las que hay que seguir evolucionando y buscando mejorar.

Y puesto que no es fácil ni algo rápido de implementar, debe ser un proceso lento, paulatino. Si lo haces a menudo, será cada vez más fácil. El objetivo es naturalizar estos procesos, los flujos de DevOps, y que sean fáciles de contagiar a los futuros miembros del equipo.