

DEVOPS

UTOMATIZACIÓN DE LA CONSTRUCCIÓN Y PRUEBAS:

- Utiliza herramientas como Apache Maven o Gradle para gestionar las dependencias y construir tu proyecto Java de manera automatizada.
- Configura una suite de pruebas automatizadas (unitarias, de integración, etc.) que se ejecuten cada vez que se realiza una nueva confirmación en el repositorio de código.

CONTROL DE VERSIONES Y COLABORACIÓN:

- Utiliza un sistema de control de versiones como Git para colaborar con otros desarrolladores de manera eficiente y rastrear los cambios en el código.
- Utiliza ramas (branches) para separar las características en desarrollo y realizar fusiones (merges) cuando las funcionalidades estén listas.

ENTREGA CONTINUA:

- Configura un servidor de integración continua como Jenkins o Travis CI que automatice la construcción, las pruebas y la entrega de tu aplicación en un entorno de pruebas o de preproducción cada vez que se confirme nuevo código en el repositorio.

DEVOPS

UTOMATIZACIÓN DE LA IMPLEMENTACIÓN:

- Utiliza herramientas de automatización de implementación como Ansible o Docker para definir y automatizar la configuración de la infraestructura necesaria para ejecutar tu aplicación.

MONITOREO Y RETROALIMENTACIÓN:

- Implementa herramientas de monitoreo y registro (logging) en tu aplicación para recopilar información sobre su rendimiento y salud en tiempo real.
- Utiliza métricas y datos de monitoreo para identificar cuellos de botella, errores o problemas de rendimiento.

ERACIÓN Y MEJORA CONTINUA:

- Utiliza la retroalimentación de los usuarios y los datos de monitoreo para iterar y mejorar constantemente la aplicación.
- Realiza análisis post-implementación para evaluar el impacto de las nuevas características o cambios en la aplicación.