

# ENTREGA CONTINUA



## INTRODUCCIÓN

Disciplina de desarrollo de software en la que el software se construye de tal manera que puede ser liberado en producción en cualquier momento. Suma la automatización de las pruebas de aceptación, por lo que el producto estará listo para desplegarlo a producción.

## MATERIALES Y MÉTODOS

### HERRAMIENTAS

- Automatizar la compilación e implementación (**Jenkins**)
- Crear tareas repetibles y manejables: (**ansible**)
- Colaborar en equipo (**Jira**)
- Planificar (**Jira/Trello**)
- Gestionar el código fuente (**Git**)
- Realizar pruebas: **Selenium** para interfaz y **JUnit** para unitarias



### INFRAESTRUCTURA

#### Entornos:

- Desarrollo: Local o en la nube
- Pruebas.
- Preproducción.

#### Contenedores y Orquestación:

- Docker
- Kubernetes



#### Automatización y Monitoreo:

- Power Bi.



#### Red y Seguridad:

- VPN y Firewalls
- Certificados SSL.



## RESULTADOS

Como resultado de aplicar estas herramientas e infraestructura tenemos:

- Reducir el tiempo
- Disminuir los costos
- Escalar el desarrollo de software según el tamaño del proyecto.
- Implementar código automáticamente .

## CONCLUSIONES

Optimiza el desarrollo de software mediante la automatización de pruebas y despliegues, obteniendo buenos resultados al reducir tiempos y costos. Facilita la colaboración en equipo y asegura que el producto esté siempre listo para producción, permitiendo a las empresas adaptarse rápidamente al mercado.

## BIBLIOGRAFÍA

Rossel Sander. (2017). *Continuous Integration, Delivery and Deployment*, Editorial Packt  
<https://www.ibm.com/es-es/topics/continuous-delivery>  
<https://www.atlassian.com/es/continuous-delivery>

