

# DEVOPS FOR JAVA DEVELOPERS: CI/CD, Jenkins & Cloud

Java, Jenkins, Git, Maven, SonarQube,  
Docker, Kubernetes y AWS



*maven*

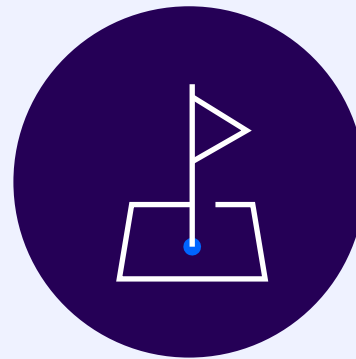


sonarqube

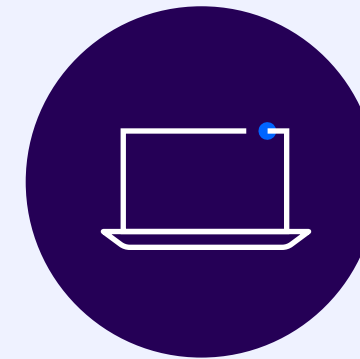




Inicio:  
**22 DE SETIEMBRE**



Finalización:  
**20 DE OCTUBRE**



**32 HORAS**  
académicas



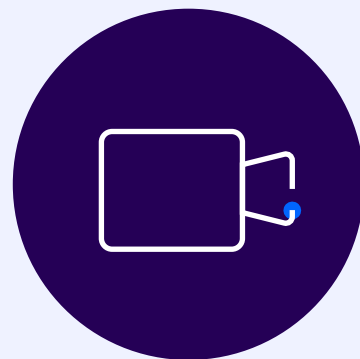
Miércoles y Viernes  
**DE 07:30H - 10:30H.**



Soporte  
**TÉCNICO**



Plataforma  
**DIGITAL**



Aplicativo  
**ZOOM**



**CERTIFICACIÓN  
DIGITAL**  
Previa aprobación  
de examen

## RESUMEN

DevOps es una metodología de desarrollo, prueba y despliegue de software de calidad de forma automatizada en periodos muy cortos para responder a las necesidades de los usuarios; se basa en la integración de desarrolladores de software y administradores de infraestructura. En este contexto Jenkins, juega un rol muy importante ya que es un servidor de automatización de código abierto que permite a los desarrolladores construir, probar e implementar de manera confiable software de calidad. Para cumplir con dicho objetivo, Jenkins se integra con frameworks, herramientas y plugins como es el caso de Git, GitFlow, Maven, SonarQube y Docker; los cuales permiten desarrollar soluciones flexibles y escalables.

## OBJETIVO

El profesional al finalizar el curso adquirirá las siguientes competencias: Interiorizar la cultura DevOps y aplicar los conceptos y tecnologías relacionadas. Usar DevOps para entregar soluciones de calidad y de manera continua. Uso de framework, herramientas y tecnologías relacionadas con DevOps .

## METODOLOGÍA

- ➔ Desarrollo de casos prácticos progresivos.
- ➔ Compartir experiencias empresariales del instructor y alumnos relacionados con proyectos e iniciativas de DevOps.
- ➔ Desarrollo de talleres complementarios.
- ➔ La evaluación es continua en el desarrollo de cada taller por sesión.

## REQUISITOS

- ➔ Entendimiento básico del manifiesto Ágil y SCRUM.
- ➔ Entendimiento básico del ciclo de desarrollo de software.
- ➔ Conocimiento de desarrollo y despliegue de aplicaciones Java.
- ➔ De preferencia experiencia como rol de analista, programador o arquitecto de software.

## TECNOLOGÍAS

- ➔ Agile y DevOps, CI y CD.
- ➔ Git y GitFlow.
- ➔ Jenkins, Maven y SonarQube.
- ➔ Docker y Docker Swarn.
- ➔ SAML, JDK (Java).
- ➔ Kubernetes.
- ➔ AWS.

# — • SESIÓN 01

## DEVOPS: TECNOLOGÍAS Y CAMBIO CULTURAL

- ¿Qué es DevOps?
- Ciclo de vida DevOps.
- Agile, Continuous Integration(CI)/ Continuous Delivery(CD) y DevOps.
- Principales herramientas DevOps, roles y certificaciones
- Taller: Casos de éxito de principales empresas que lo implementan.

## AMBIENTE DE CONTROL DE VERSIONES CON GIT

- Definición de Control de versiones.
- Diferencias entre GIT, GitHub, GitLab y Bitbucket.
- Recomendaciones y buenas prácticas basado en GITFLOW.
- Asegurando el repositorio Git (SSH VS HTTPS, ACCESOS Y PERMISOS).
- Taller: Configuración y principales comandos de Git.

# — • SESIÓN 02

## AMBIENTE DE DESARROLLO Y DESPLIEGUE CON DOCKER

- ¿Qué es Docker?
- Arquitectura, Containers vs Virtual Machine.
- Definición de Imágenes, DockerFile, DockerCompose y DockerHub.
- Ejecutando build, push, pull y otros comandos.
- Laboratorio: Migración de una aplicación monolítica.

## AMBIENTE CI/CD CON JENKINS

- ¿Qué es Jenkins?, Jenkins como servidor de automatización.
- Integración Continua Vs Entrega Continua Vs Despliegue Continuo.
- Gestión de la consola Jenkins, principales funcionalidades.
- Gestión de Jobs (crear, copiar, renombrar y eliminar).
- Taller: Configuración y principales funcionalidades.

# — • SESIÓN 03

## INTEGRANDO JENKINS CON TOOLS

- Gestión de alertas (correos).
- Gestión de dependencias (Maven).
- Gestión de dependencias (Git).
- Gestión de calidad del código (SonarQube) .
- Taller: Taller de integración de Correos, Maven, Git y SonarQube.

## CASO INTEGRADO – PARTE I

- Creando del ambiente de desarrollo con Docker/Git.
- Configurando la aplicación a utilizar.
- Configurar el Pipeline (Build Project).
- Jenkins Test con SonarQube.
- Notificación de resultados del testing y alertas con Jenkins y Slack.

# — • SESIÓN 04

## ASEGURANDO EL AMBIENTE CI/CD

- Authentication and authorization.
- Que es Role-based Authorization Strategy.
- Definición de SAML.
- Que es OKTA.
- Taller: Integrando SAML Auth con Jenkins.

## CASO INTEGRADO – PARTE II

- Integrando Jenkins con Docker.
- Jenkins Build and DockerHub.
- Jenkins Deploy y monitoring docker image.
- Trabajando con master, hotfix, reléase, develop, feature.
- Lecciones aprendidas y próximos temas a investigar.

CRONOGRAMA

SESIÓN	01		02		03		04	
ACTIVIDAD	DevOps: Tecnologías y Cambio Cultural		Ambiente de Desarrollo y Despliegue con Docker		Integrando Jenkins con Tools		Asegurando el Ambiente CI/CD	
	Ambiente de Control de Versiones con Git		Ambiente CI/CD con Jenkins		Caso Integrado – Parte I		Caso Integrado – Parte II	
FECHA	22 SET	24 SET	29 SET	01 OCT	06 OCT	13 OCT	15 OCT	20 OCT

Este cronograma puede estar sujeto a cambios por parte de Galaxy Training.



## TALLERES

### 01 DEVOPS CON KUBERNETES

Integraremos Jenkins y Kubernetes para escalar e implementar despliegues automáticos en entornos cloud.

**03 HORAS**

### 02 DEVOPS EN CLOUD (AWS)

Automatizaremos despliegues en cloud (AWS) utilizando Jenkins, AWS S3 Bucket y AWS CodeDeploy.

**03 HORAS**