



Servicio de Instrucción	Oficialización: Director Edwin Maraví
CURSO JAVA DevOPS	Año:2020
	Código : C09_202 0

Sílabo

JAVA DevOPS

I. DATOS GENERALES

Año de vigencia	:	2020
Sesiones	:	10
N° de Horas académicas:		50
Requisitos	:	
	-	Conocimientos básicos de java 8.
	-	Conocimientos de desarrollo de servicios REST con Java.
	-	Conocimientos básicos desarrollo de microservicios con JAVA.
	-	Conocimientos básicos de Linux.

II. FUNDAMENTACIÓN

El desarrollo en la NUBE necesita de tener claro la infraestructura sobre la cual vana a correr las aplicaciones y microservicios. El curso comprende la administración de los siguientes productos: OpenShift, Docker y Kubernetes.

III. UNIDAD DE COMPETENCIA

Definir y configurar los contenedores escalables y de alta disponibilidad para microservicios.

IV. CAPACIDADES TERMINALES Y CRITERIOS DE EVALUACIÓN

CAPACIDAD TERMINAL	CRITERIOS DE EVALUACIÓN
Configurar contenedores de alta disponibilidad.	Administración y configuración de OpenShift
	Administración y configuración de Docker
	Administración y configuración de Kubernetes

V. CONTENIDOS

5.1. Administración en OpenShift:

- a) **Introducción a Red Hat OpenShift Container Platform**
 - ✓ Revisión de las características y la arquitectura de OpenShift
- b) **Instalación de OpenShift Container Platform**
 - ✓ Instalación de OpenShift Container Platform y configuración de un maestro y un nodo.



Servicio de Instrucción	Oficialización: Director Edwin Maraví
CURSO JAVA DevOPS	Año:2020
	Código : C09_202 0

- c) **Ejecución de comandos**
 - ✓ Ejecución de comandos desde la interfaz de línea de comandos.
- d) **Creación de aplicaciones**
 - ✓ Creación, construcción e implementación de aplicaciones en una instancia de OpenShift Container Platform.
- e) **Almacenamiento persistente**
 - ✓ Provisión de almacenamiento persistente y uso del mismo para el registro interno.
- f) **Administración del sistema**
 - ✓ Uso de componentes de OpenShift Container Platform para gestionar aplicaciones implementadas.

5.2. **Administración en Docker:**

- a) **Introducción a Docker**
 - ✓ Introducción a docker, ventajas y mitos
 - ✓ Que son los contenedores
 - ✓ Que es Docker
- b) **Instalando Docker**
 - ✓ Docker en Linux
 - ✓ Instalando Docker
 - ✓ Creando el primer contenedor
- c) **Primeros pasos con Docker**
 - ✓ Estructura de comandos
 - ✓ Imágenes y contenedores
 - ✓ Docker Store
- d) **Rutinas con imágenes y contenedores**
 - ✓ Iniciar y listar contenedores
 - ✓ Mostrar logs
 - ✓ Eliminando imágenes y contenedores
 - ✓ Guardar imágenes
- e) **Construyendo imágenes**
 - ✓ Introducción al módulo
 - ✓ Introducción a Dockerfile
 - ✓ Construir la primera imagen
 - ✓ Instrucciones en Dockerfile
 - ✓ Buenas prácticas en Dockerfile
- f) **Volúmenes**
 - ✓ ¿Qué son los volúmenes?
 - ✓ Utilizar volúmenes de datos
 - ✓ Utilizar contenedores como volúmenes de datos
 - ✓ Salvar la información



Servicio de Instrucción	Oficialización: Director Edwin Maraví
CURSO JAVA DevOPS	Año:2020
	Código : C09_202 0

- g) Publicar imágenes desde GitHub**
- ✓ Filosofía de trabajo con GitHub
 - ✓ Crear imágenes desde GitHub
 - ✓ ¿Y si se modifica el código?

5.3.Administración en Kubernetes:

- a) Introducción**
- ✓ Introduction to Kubernetes
 - ✓ Introduction to Docker
 - ✓ Containers vs. Virtual Machines
 - ✓ Docker Architecture
 - ✓ The Docker Hub
 - ✓ Kubernetes Architecture
 - ✓ Introduction to YAML
- b) Configuración de kubernetes**
- ✓ Packages and Dependencies
 - ✓ Install and Configure Master Controller
 - ✓ Install and Configure the Minions
 - ✓ Kubectl: Exploring our Environment
- c) Fundamentos de Docker**
- ✓ Pulling an Image
 - ✓ Running Containers
 - ✓ The Dockerfile
 - ✓ Managing Ports with Container Deployments
- d) Pods, Tags and Services**
- ✓ Create and Deploy Pod Definitions
 - ✓ Tags, Labels and Selectors
 - ✓ Deployment State
 - ✓ Multi-Pod (Container) Replication Controller
 - ✓ Create and Deploy Service Definitions
- e) Logs, Scaling and Recovery**
- ✓ Creating Temporary Pods at the Command Line
 - ✓ Interacting with Pod Containers
 - ✓ Logs
 - ✓ Autoscaling and Scaling our Pods
 - ✓ Failure and Recovery



Servicio de Instrucción	Oficialización: Director Edwin Maraví
CURSO JAVA DevOPS	Año:2020
	Código : C09_202 0

VI. METODOLOGÍA

- Curso teórico-práctico.
- Exposición del profesor, ayudas audiovisuales, diálogo abierto y debate.



Servicio de Instrucción	Oficialización: Director Edwin Maraví
CURSO JAVA DevOPS	Año:2020
	Código : C09_202 0

- Realización de casos prácticos en laboratorio.
- Realización de tareas en aulavirtual.
- Generación de conocimiento mediante investigación y foros de discusión en aulavirtual.

VII. EVALUACIÓN

La evaluación, como constatación del logro de los objetivos que es alcanzar las capacidades mencionadas se dará mediante la siguiente ponderación:

03 Tareas	20 %
03 Evaluaciones Virtuales	20 %
03 Evaluaciones prácticas	60 %
TOTAL	100 %