

Guía de Usuario para el Proyecto Automated Deployment

Descripción del Proyecto

Este proyecto, Automated Deployment with Docker and Kubernetes, está diseñado para automatizar el despliegue de una aplicación web. Utiliza tecnologías modernas como Docker, Kubernetes, Jenkins y AWS para crear un flujo de trabajo CI/CD eficiente, desde la construcción de la imagen hasta el despliegue en un entorno de producción.

Requisitos Previos

Antes de comenzar, asegúrate de tener los siguientes componentes instalados en tu máquina o en tu entorno de trabajo:

- **Git:** Para clonar el repositorio.
 - **Docker:** Para construir y ejecutar contenedores.
 - **Kubernetes (kubectl y minikube o un clúster en la nube):** Para gestionar y orquestar los despliegues.
 - **Jenkins (opcional):** Para configurar y ejecutar pipelines de CI/CD.
 - **AWS CLI (opcional):** Si deseas utilizar servicios de AWS como EKS o EC2.
-

Paso 1: Clonar el Repositorio

Primero, clona el repositorio en tu máquina local:

```
bash
git clone https://github.com/<tu-usuario>/automated-deployment.git
cd automated-deployment
```

Paso 2: Construir la Imagen Docker

Construye la imagen Docker de la aplicación usando el script proporcionado:

```
bash
./scripts/build.sh
```

Este script ejecutará el Dockerfile, construyendo una imagen con el nombre myapp:latest.

Paso 3: Ejecutar la Aplicación en Local

Opción 1: Docker

Si deseas ejecutar la aplicación localmente utilizando Docker, simplemente ejecuta:

```
bash
docker run -p 8080:8080 myapp:latest
```

Esto levantará un contenedor Docker y expondrá la aplicación en <http://localhost:8080>.

Opción 2: Kubernetes (Minikube)

Si prefieres utilizar Kubernetes localmente, puedes usar Minikube para crear un clúster local:

1. Inicia Minikube:

```
bash
minikube start
```

2. Despliega la aplicación en Kubernetes usando los archivos YAML en el directorio kubernetes:

```
bash
kubectl apply -f kubernetes/deployment.yaml
kubectl apply -f kubernetes/service.yaml
kubectl apply -f kubernetes/ingress.yaml
```

3. Obtén la URL del Ingress para acceder a la aplicación:

```
bash
minikube service myapp-service --url
```

Esto te proporcionará una URL a la cual puedes acceder desde tu navegador.

Paso 4: Configuración y Ejecución del Pipeline CI/CD con Jenkins

Para automatizar el flujo de trabajo usando Jenkins:

1. Instala Jenkins en tu máquina local o en una instancia de AWS utilizando el script `jenkins_setup.sh`:

```
bash Copiar código
./scripts/jenkins_setup.sh
```

2. Configura un nuevo pipeline en Jenkins y apunta al Jenkinsfile del repositorio:
 - Abre Jenkins en tu navegador.
 - Crea un nuevo proyecto de tipo *Pipeline*.
 - Configura el repositorio Git y selecciona la rama principal.
 - En la sección de *Pipeline*, selecciona "Pipeline script from SCM" y especifica el path al Jenkinsfile.
3. Ejecuta el Pipeline:
 - El pipeline compilará la imagen Docker, realizará pruebas, y desplegará la aplicación en el clúster de Kubernetes configurado.

Paso 5: Despliegue en un Entorno en la Nube

Opción 1: Kubernetes en AWS EKS

1. Configura AWS CLI y autentícate en tu cuenta.
2. Crea un clúster EKS:

```
bash Copiar código
aws eks create-cluster --name my-cluster --region <tu-región>
```

3. Despliega la aplicación en EKS siguiendo los mismos pasos que para Minikube, pero apuntando a tu clúster en AWS.

Opción 2: Jenkins en AWS EC2

1. Lanza una instancia EC2 y sigue el script `jenkins_setup.sh` para configurar Jenkins.

2. **Sigue los pasos del Paso 4 para ejecutar el pipeline y desplegar la aplicación.**
-

Solución de Problemas Comunes

- **Problemas de conexión en Kubernetes:** Verifica que los puertos estén correctamente configurados y que Minikube esté ejecutándose.
- **Fallos en Jenkins:** Revisa los logs del pipeline para identificar errores en la construcción de la imagen o en el despliegue.
- **Acceso a AWS:** Asegúrate de que tus credenciales de AWS estén correctamente configuradas y que tienes los permisos necesarios para crear recursos.

Conclusión

Este proyecto proporciona una solución completa para automatizar el despliegue de aplicaciones web, desde la construcción con Docker hasta el despliegue en Kubernetes. Puedes personalizar el pipeline y adaptarlo a diferentes entornos según tus necesidades. ¡Anímate a explorar y expandir este proyecto con nuevas funcionalidades y mejoras!

Si tienes alguna pregunta o necesitas asistencia adicional, no dudes en abrir un issue en el repositorio.