***4. Kubernetes deploying and exposing using Jenkins***

**Objective**

The objective of this setup is to configure a **Jenkins CI/CD pipeline** to deploy an application on **Minikube** running on an Ubuntu 24.04 system. This ensures an automated deployment process where code changes are fetched from a Git repository, deployed to Kubernetes, and made accessible as a service.

**Procedure**  
**1. Minikube Installation and Setup**   
**Installing Minikube**   
 1.Install Minikube:   
 2brew install minikube

3.minikube version

4.minikube start

1. **Configuring Kubernetes and Minikube**   
   **Set Up Kubeconfig for Minikube**   
    1.Ensure the Kubernetes context is switched to Minikube: 2.kubectl config use-context minikube   
    3.Verify Minikube status:   
    4.kubectl get nodes

**Fixing TLS Certificate Issues**

If you see a TLS certificate verification error:   
kubectl config set-cluster minikube --server=https://192.168.49.2:8443 \ --certificate-authority=$HOME/.minikube/ca.crt --embed-certs=true **3. Jenkins Integration with Minikube**   
**Installing and Configuring Jenkins**

Install Jenkins:   
sudo apt update && sudo apt install jenkins -y

Add Jenkins to Docker Group:   
sudo usermod -aG docker jenkins

Apply changes:   
sudo systemctl restart jenkins

Configure Kubernetes for Jenkins by setting environment variables in the Jenkins pipeline:   
environment {   
 KUBECONFIG = "/var/lib/jenkins/.kube/config"   
 MINIKUBE\_HOME = "/var/lib/jenkins/.minikube"   
}

Ensure Jenkins has access to the Kubeconfig file: sudo chown -R jenkins:jenkins /var/lib/jenkins/.kube sudo chmod -R 755 /var/lib/jenkins/.kube   
**4. Deploying Application via Jenkins Pipeline**   
**Jenkinsfile (Pipeline Configuration)**   
pipeline {   
 agent any

environment {   
 KUBECONFIG = "/var/lib/jenkins/.kube/config" MINIKUBE\_HOME = "/var/lib/jenkins/.minikube" }

stages {   
 stage('Setup Minikube Context') {   
 steps {   
 script {   
 sh '''   
 set -e

echo " Switching to Minikube context..." kubectl config use-context minikube

echo " Minikube context set successfully!" '''   
 }   
 }   
 }

stage('Deploy Application') {   
 steps {   
 script {   
 sh '''   
 set -e

echo " Deploying Application..."   
 kubectl apply -f deployment.yaml --validate=false '''   
 }   
 }   
 }   
 }   
}   
**Pipeline Execution Output (Successful Deployment)**

Minikube context set successfully!   
 Deploying Application...

deployment.apps/devopstask04 created   
Finished: SUCCESS







