DAY 5

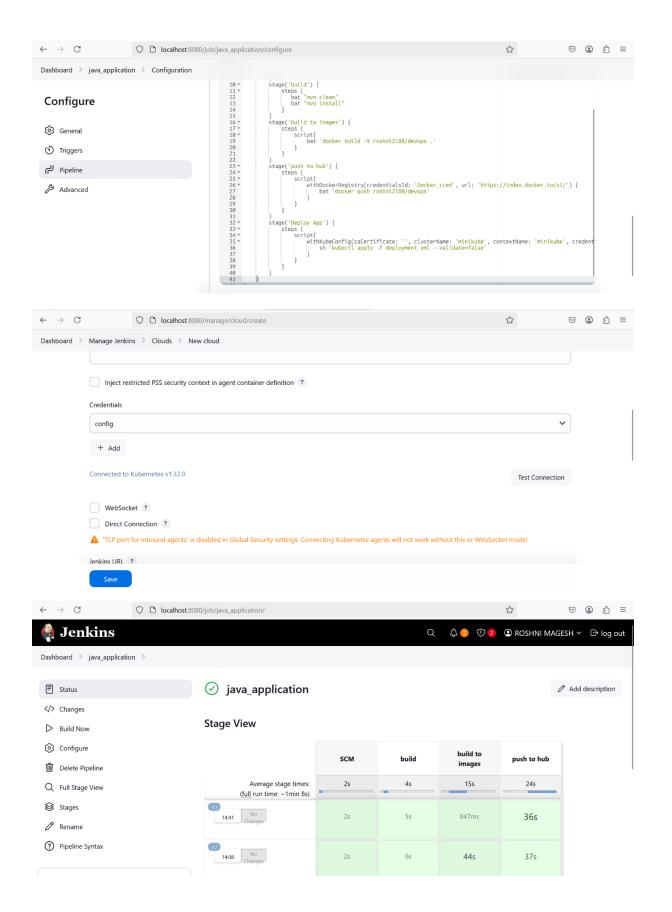
users:
- name: minikube
user:

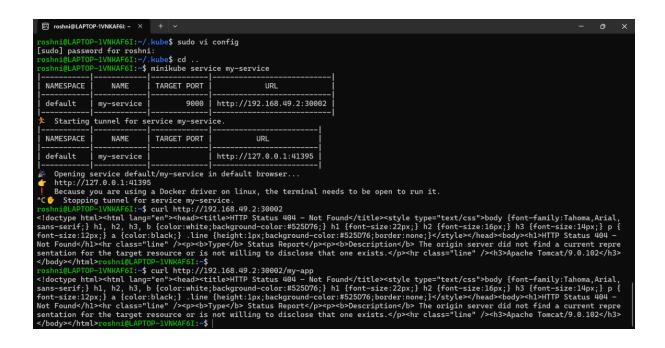
.

client-certificate: /home/roshni/.minikube/profiles/minikube/client.crt client-key: /home/roshni/.minikube/profiles/minikube/client.key

```
    roshni@LAPTOP-1VNKAF6I: ~ × + ∨

     oshni@LAPTOP-1VNKAF6I:~$ minikube status
rosnigLAPIOP-IVMRAFO:
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
roshni@LAPTOP-1VNKAF6I:~$ cd ~/.kube
roshni@LAPTOP-1VNKAF6I:~/.kube$ sudo vi config
[sudo] password for roshni:
roshni@LAPTOP-1VNKAF6I:~/.kube$ sudo vi config
[sudo] password for roshni:
roshni@LAPTOP-1VNKAF6I:~/.kube$ sudo vi config
[1]+ Stopped sudo vi config roshni@LAPTOP-1VNKAF6I:~/.kube$ sudo vi config roshni@LAPTOP-1VNKAF6I:~/.kube$ kubectl get node NAME STATUS ROLES AGE VERSION minikube Ready control-plane 2d v1.32.0 roshni@LAPTOP-1VNKAF6I:~/.kube$
  roshni@LAPTOP-1VNKAF6I:~$ cat ~/.kube/config apiVersion: v1
 clusters:
- cluster:
certificate-authority: /home/roshni/.minikube/ca.crt
      certificate-authority: /home/roshni/.minikube/cextensions:
- extension:
last-update: Fri, 21 Mar 2025 07:56:06 UTC
provider: minikube.sigs.k8s.io
version: v1.35.0
name: cluster_info
server: https://127.0.0.1:32769
name: minikube
  contexts:
        context:
cluster: minikube
cluster: minikube
  extensions:
    - extension:
    last-update: Fri, 21 Mar 2025 07:56:06 UTC
    provider: minikube.sigs.k8s.io
    version: v1.35.0
    name: context_info
    namespace: default
    user: minikube
    name: minikube
    current-context: minikube
kind: Config
    preferences: {}
    users:
```





sudo visudo

(below root => jenkins ALL=(ALL) NOPASSWD: ALL)

sudo systemctl restart ssh.service

sudo systemctl restart sshd.service

sudo apt update

sudo apt install openssh-server

sudo systemctl restart ssh.service

sudo systemctl status ssh

sudo systemctl daemon-reload

cd ~/.kube

sudo vi config

CODE

pipeline {

agent any

```
stages {
    stage('scm') {
      steps {
    git branch: "
      }
    }
    stage('builb-clean') {
      steps {
        sh "mvn clean"
}
}
    stage('build-validate') {
      steps {
        sh "mvn validate"
}
}
    stage('build-com') {
      steps {
        sh "mvn compile"
}
}
    stage('build-test') {
      steps {
        sh "mvn test"
}
}
```

```
stage('build-install') {
      steps {
        sh "mvn package"
}
}
stage('build to images') {
      steps {
        script{
          sh 'docker build -t roshni2108/devops .'
        }
  }
}
stage('push to hub') {
      steps {
        script{
         withDockerRegistry(credentialsId: 'Docker cred', url: 'https://index.docker.io/v1/') {
          sh 'docker push roshni2108/devops'
        }
      }
      }
}
    stage('Deploy App') {
      steps {
         withKubeConfig(caCertificate: ", clusterName: 'minikube', contextName: 'minikube',
credentialsId: 'mukubeconfig_011', namespace: '', restrictKubeConfigAccess: false, serverUrl:
'https://192.168.49.2:8443') {
         sh 'kubectl apply -f deployment.yml --validate=false'
      }
    }
```

```
stage('Test') {
    steps {
        withKubeConfig(caCertificate: ", clusterName: 'minikube', contextName: 'minikube',
        credentialsId: 'mukubeconfig_011', namespace: ", restrictKubeConfigAccess: false, serverUrl:
    'https://192.168.49.2:8443') {
        sh 'minikube service my-service --url | xargs curl'
     }
}
```