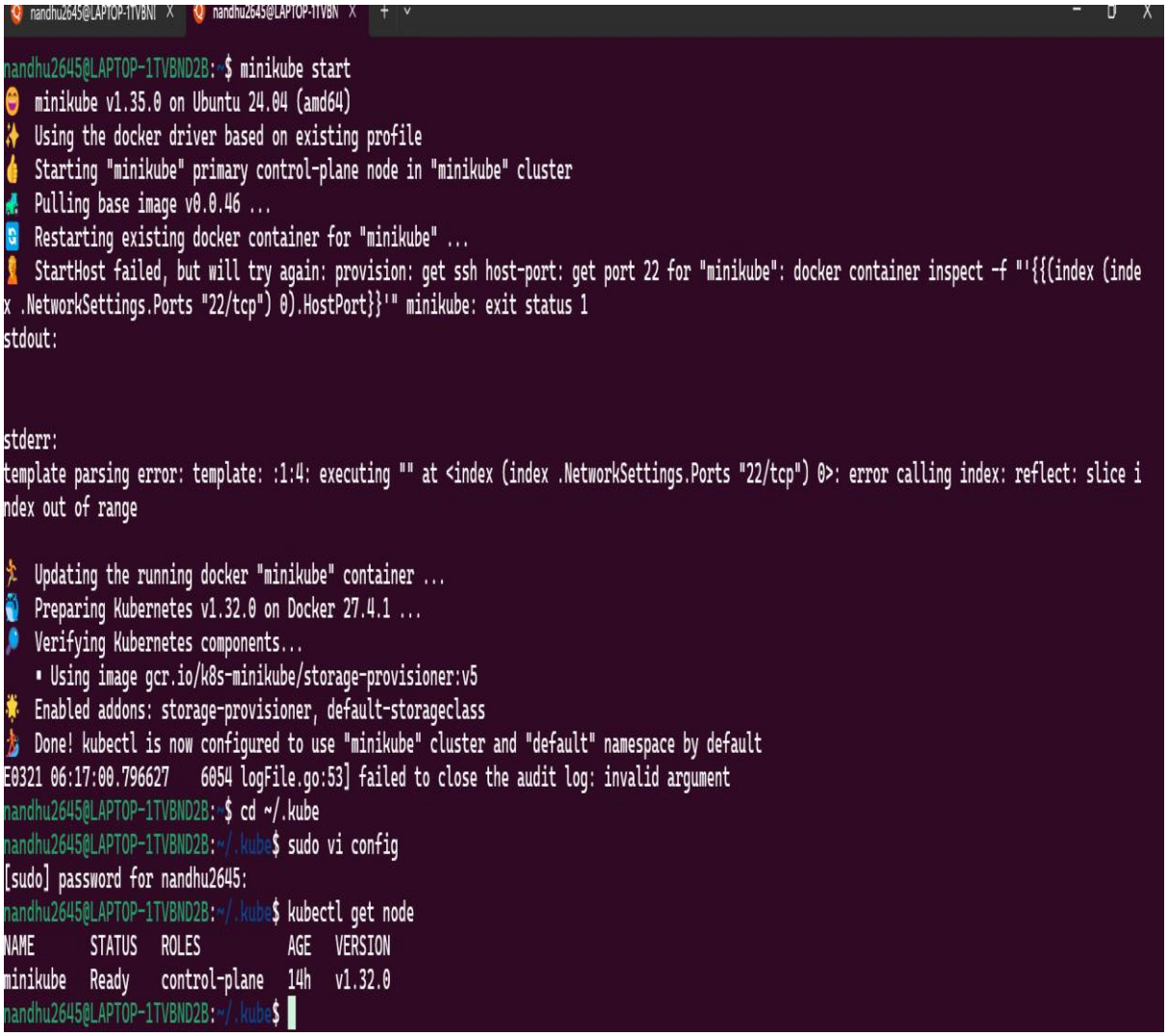
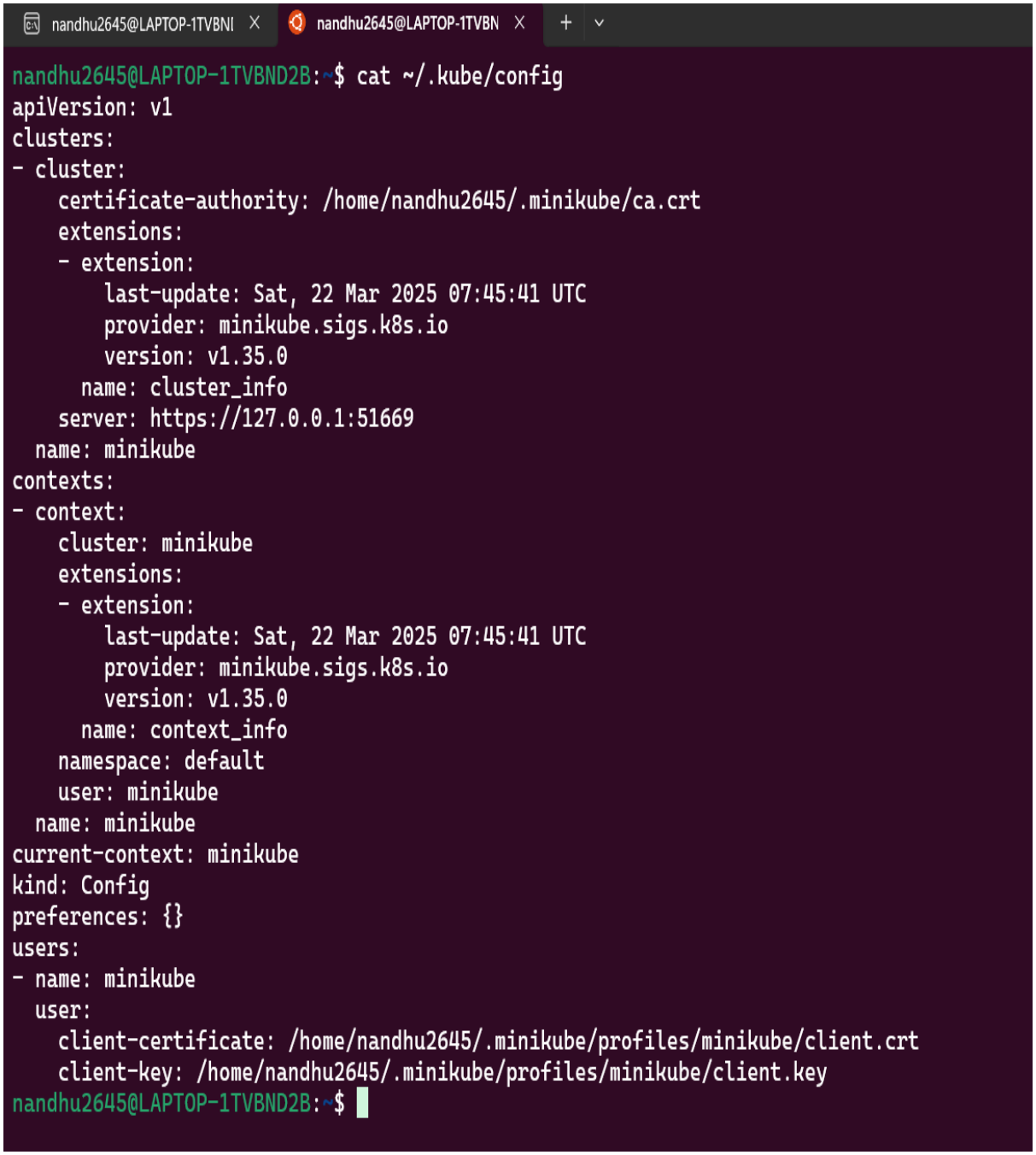
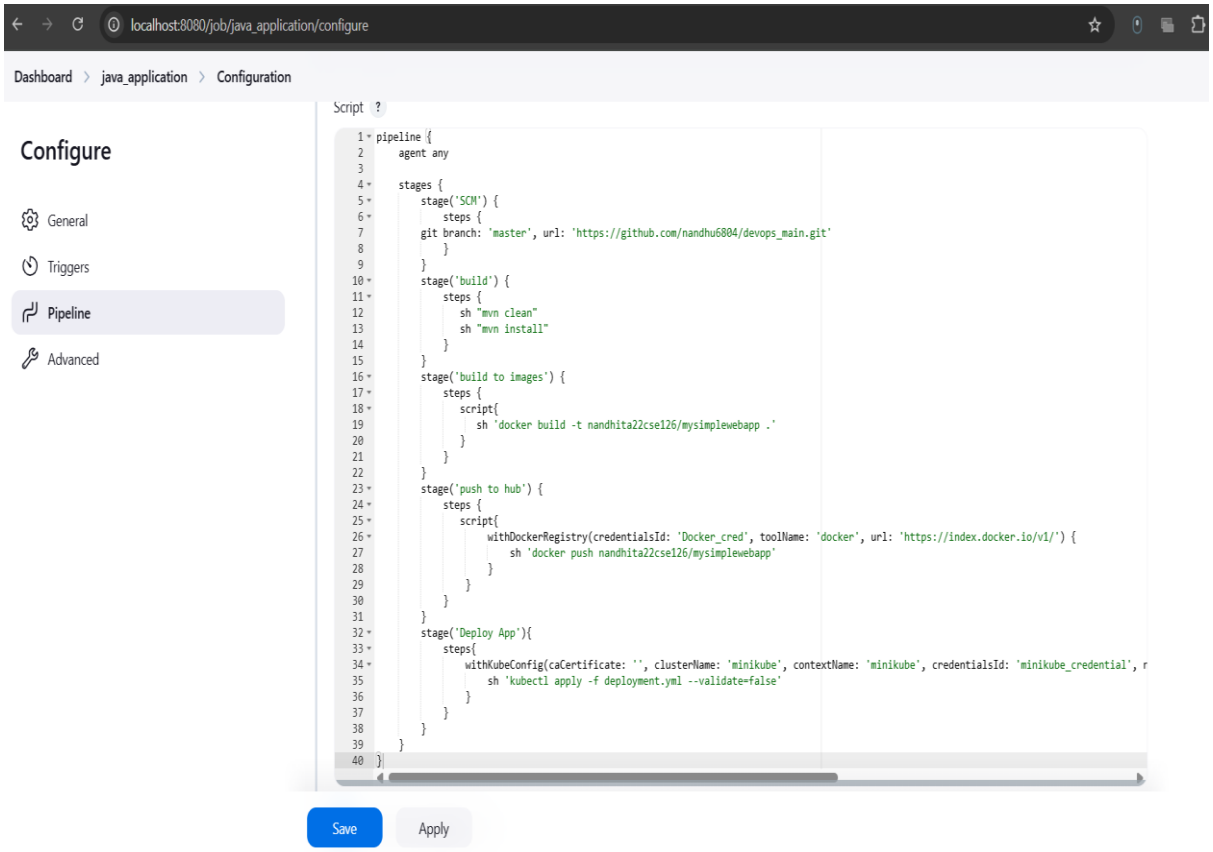
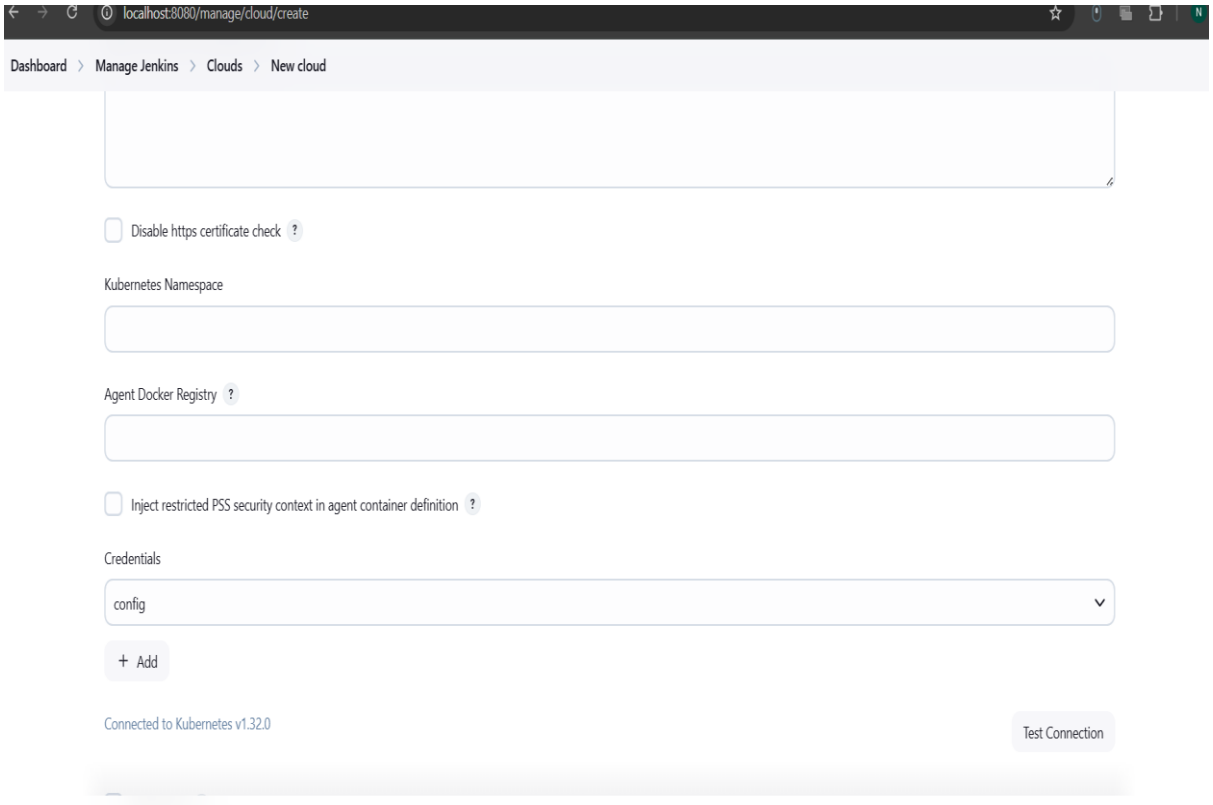
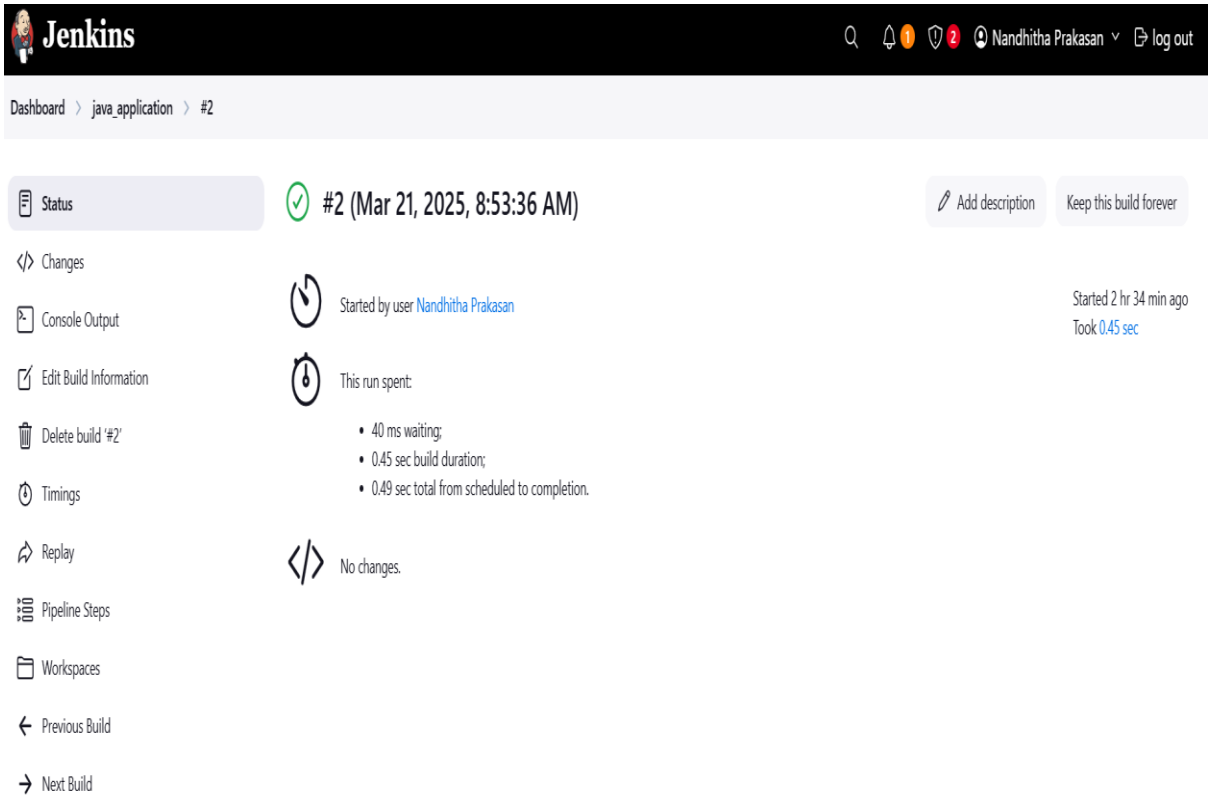
JAVA APPLICATION MINIKUBE DEPLOYMENT

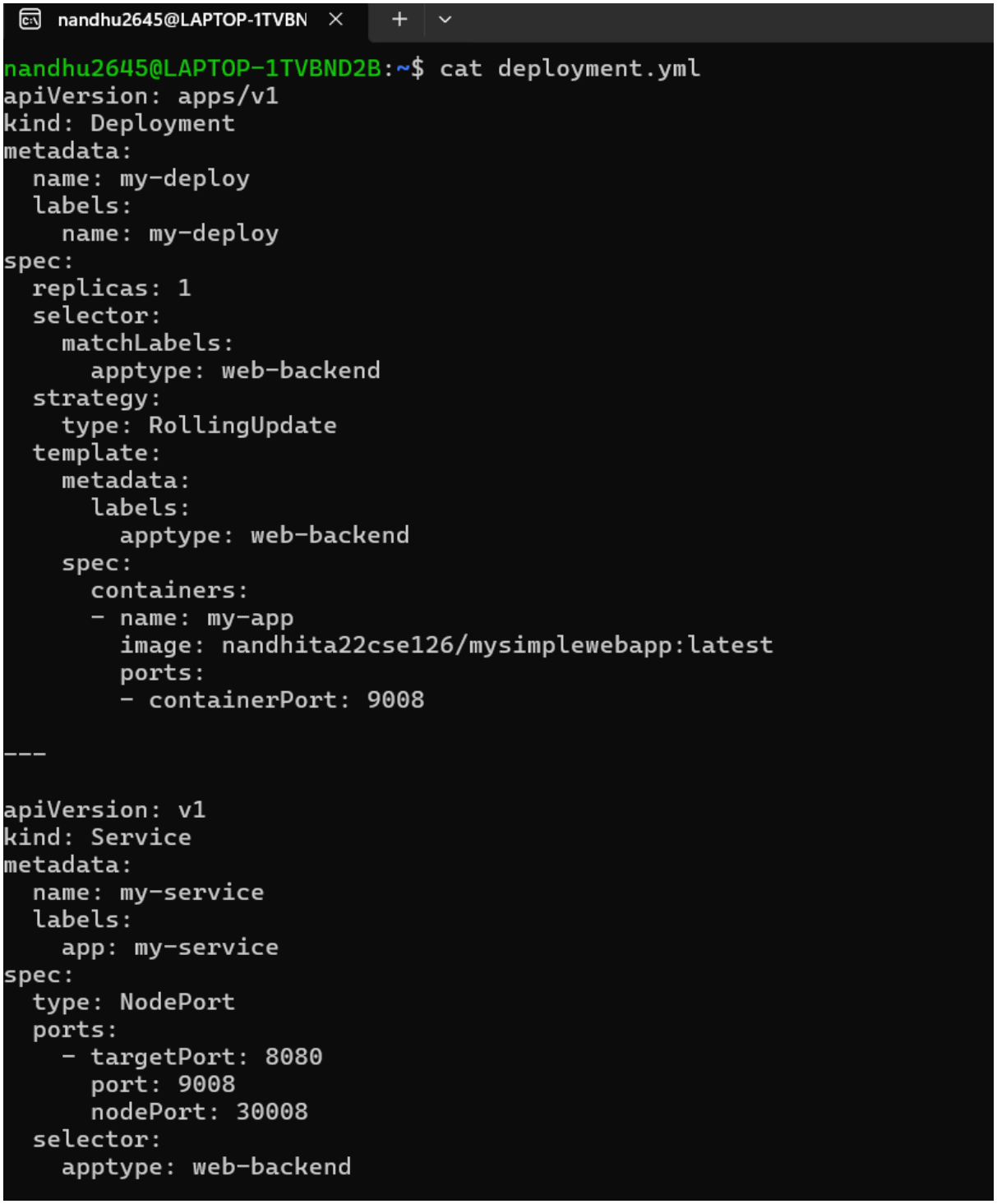



nandhu2645@LAPTOP-1TVBNI
nandhu2645@LAPTOP-1TVBN
nandhu2645@LAPTOP-1TVBND2B:~$ cat ~/.kube/config
apiVersion: v1
clusters:
- cluster:
    certificate-authority: /home/nandhu2645/.minikube/ca.crt
    extensions:
    - extension:
        last-update: Sat, 22 Mar 2025 07:45:41 UTC
        provider: minikube.sigs.k8s.io
        version: v1.35.0
      name: cluster_info
    server: https://127.0.0.1:51669
  name: minikube
contexts:
- context:
    cluster: minikube
    extensions:
    - extension:
        last-update: Sat, 22 Mar 2025 07:45:41 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:
- name: minikube
  user:
    client-certificate: /home/nandhu2645/.minikube/profiles/minikube/client.crt
    client-key: /home/nandhu2645/.minikube/profiles/minikube/client.key
nandhu2645@LAPTOP-1TVBND2B:~$


localhost:8080/job/java_application/configure
Dashboard › java_application › Configuration
Configure
General
Triggers
Pipeline
Advanced
Script ?
pipeline {
agent any
stages {
stage('SCM') {
steps {
git branch: 'master', url: 'https://github.com/nandhu6804/devops_main.git'
}
}
stage('build') {
steps {
sh "mvn clean"
sh "mvn install"
}
}
stage('build to images') {
steps {
script{
sh 'docker build -t nandhita22cse126/mysimplewebapp .'
}
}
}
stage('push to hub') {
steps {
script{
withDockerRegistry(credentialsId: 'Docker_cred', toolName: 'docker', url: 'https://index.docker.io/v1/') {
sh 'docker push nandhita22cse126/mysimplewebapp'
}
}
}
}
stage('Deploy App'){
steps{
withKubeConfig(caCertificate: '', clusterName: 'minikube', contextName: 'minikube', credentialsId: 'minikube_credential',
sh 'kubectl apply -f deployment.yml --validate=false'
}
}
}
}
}
Save
Apply


localhost:8080/manage/cloud/create
Dashboard › Manage Jenkins › Clouds › New cloud
Disable https certificate check ?
Kubernetes Namespace
Agent Docker Registry ?
Inject restricted PSS security context in agent container definition ?
Credentials
config
+ Add
Connected to Kubernetes v1.32.0
Test Connection


Jenkins
Nandhitha Prakasan
log out
Dashboard
java_application
#2
Status
Changes
Console Output
Edit Build Information
Delete build '#2'
Timings
Replay
Pipeline Steps
Workspaces
Previous Build
Next Build
#2 (Mar 21, 2025, 8:53:36 AM)
Add description
Keep this build forever
Started by user Nandhitha Prakasan
Started 2 hr 34 min ago
Took 0.45 sec
This run spent:
40 ms waiting;
0.45 sec build duration;
0.49 sec total from scheduled to completion.
No changes.


nandhu2645@LAPTOP-1TVBN
nandhu2645@LAPTOP-1TVBND2B:~$ cat deployment.yml
apiVersion: apps/v1
kind: Deployment
metadata:
name: my-deploy
labels:
name: my-deploy
spec:
replicas: 1
selector:
matchLabels:
apptype: web-backend
strategy:
type: RollingUpdate
template:
metadata:
labels:
apptype: web-backend
spec:
containers:
- name: my-app
image: nandhita22cse126/mysimplewebapp:latest
ports:
- containerPort: 9008
---
apiVersion: v1
kind: Service
metadata:
name: my-service
labels:
app: my-service
spec:
type: NodePort
ports:
- targetPort: 8080
port: 9008
nodePort: 30008
selector:
apptype: web-backend

```
nandhu2645@LAPTOP-1TVBN
nandhu2645@LAPTOP-1TVBND2B:~$ minikube start
😄  minikube v1.35.0 on Ubuntu 24.04 (amd64)
✨  Using the docker driver based on existing profile
👍  Starting "minikube" primary control-plane node in "minikube" cluster
🚜  Pulling base image v0.0.46 ...
🔄  Restarting existing docker container for "minikube" ...
🐳  Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟  Enabled addons: default-storageclass, storage-provisioner
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
E0322 06:05:57.597826    1133 logFile.go:53] failed to close the audit log: invalid argument
nandhu2645@LAPTOP-1TVBND2B:~$ ls
Jenkinsfile  config  deployment.yml  devops_main  pod.yml  rs-test.yml
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl get pod
No resources found in default namespace.
nandhu2645@LAPTOP-1TVBND2B:~$ sudo nano deployment.yml
[sudo] password for nandhu2645:
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl apply -f deployment.yml
deployment.apps/my-deploy created
service/my-service created
nandhu2645@LAPTOP-1TVBND2B:~$ minikube service my-service
|-----------|------------|-------------|---------------------------|
| NAMESPACE |    NAME    | TARGET PORT |            URL            |
|-----------|------------|-------------|---------------------------|
| default   | my-service |        9008 | http://192.168.58.2:30008 |
|-----------|------------|-------------|---------------------------|
🏃  Starting tunnel for service my-service.
|-----------|------------|-------------|------------------------|
| NAMESPACE |    NAME    | TARGET PORT |          URL           |
|-----------|------------|-------------|------------------------|
| default   | my-service |             | http://127.0.0.1:34969 |
|-----------|------------|-------------|------------------------|
🎉  Opening service default/my-service in default browser...
👉  http://127.0.0.1:34969
❗  Because you are using a Docker driver on linux, the terminal needs to be open to run it.
^C✋  Stopping tunnel for service my-service.
E0322 06:11:30.563315    1835 logFile.go:53] failed to close the audit log: invalid argument
nandhu2645@LAPTOP-1TVBND2B:~$ kubectl port-forward svc/my-service 9008:9008
Forwarding from 127.0.0.1:9008 -> 8080
Forwarding from [::1]:9008 -> 8080
```

Hello World!

Deployment.yml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-deploy
  labels:
    name: my-deploy
spec:
  replicas: 1
  selector:
    matchLabels:
      apptype: web-backend
  strategy:
    type: RollingUpdate
  template:
    metadata:
      labels:
        apptype: web-backend
    spec:
      containers:
      - name: my-app
        image: nandhita22cse126/mysimplewebapp:latest
        ports:
        - containerPort: 9008

---

apiVersion: v1
kind: Service
metadata:
```

```
  name: my-service
  labels:
    app: my-service
spec:
  type: NodePort
```