JAVA APPLICATION DEPLOYMENT IN MINIKUBE

Linux System Setup and Java Installation

sudo apt install fontconfig openjdk-17-jre

java -version

Jenkins Installation and Management

sudo service jenkins restart

sudo service jenkins status

For installation instructions: Jenkins Installation Guide

Docker Installation and Commands

sudo apt install docker.io -y

sudo service docker restart

sudo service docker status

sudo usermod -aG docker $USER

Checking Docker Images and Containers

docker images

docker ps

Fixing permission issues

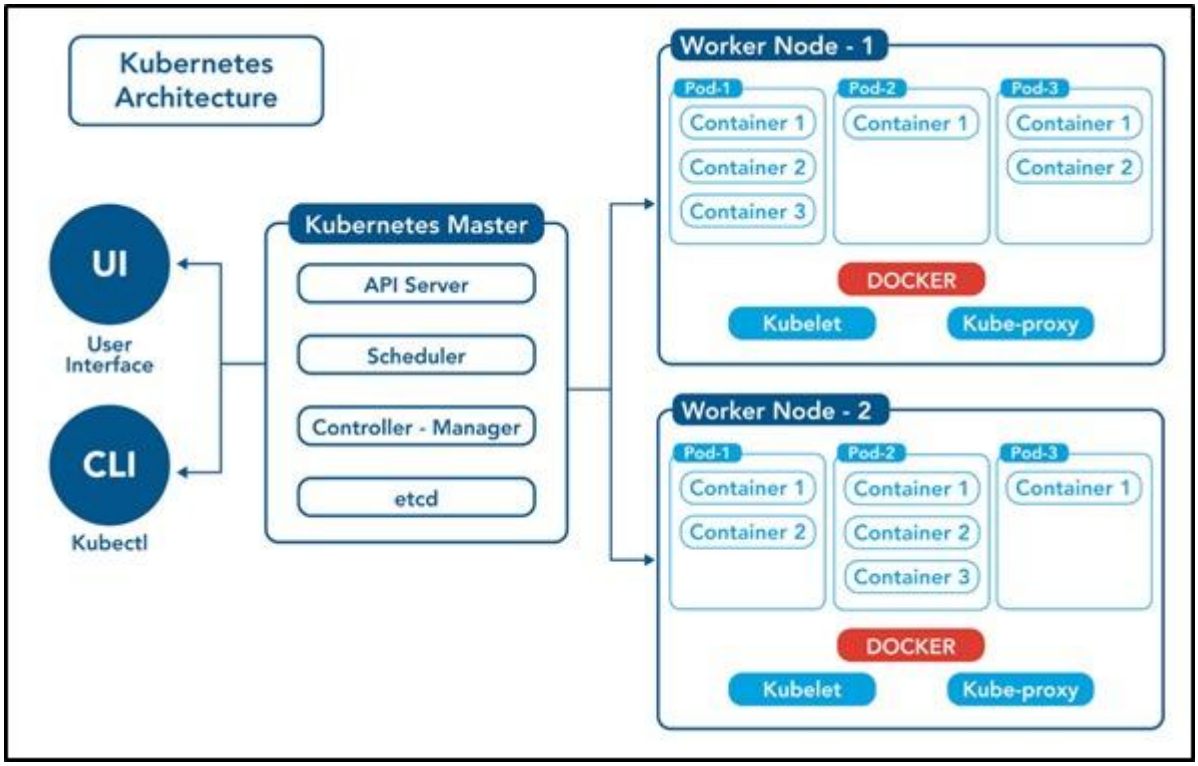
sudo chmod 666 /var/run/docker.sock

Docker Compose Installation

sudo apt install docker-compose -y

sudo curl -L "https://github.com/docker/compose/releases/latest/download/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

---



Kubernetes (K8s) Installation and Commands

Installing kubectl

curl -LO https://dl.k8s.io/release/v1.32.0/bin/linux/amd64/kubectl

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

chmod +x kubectl

mkdir -p ~/.local/bin

mv ./kubectl ~/.local/bin/kubectl

kubectl version --client

More details: Install kubectl

Installing Minikube

curl -LO https://github.com/kubernetes/minikube/releases/latest/download/minikube-linux-amd64

sudo install minikube-linux-amd64 /usr/local/bin/minikube && rm minikube-linux-amd64

minikube start

minikube status

Kubernetes Commands

Pod Management

```
# Create a pod
kubectl run my-pod --image=nginx --port=80

# View all pods
kubectl get pods
kubectl get pods -A
kubectl get pods -n kube-system

# View pod details
describe pod <pod-name>
kubectl logs <pod-name>
kubectl exec <pod-name> -- <command>
```

YAML Configuration for a Pod

```
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
  labels:
    app: my-web-app
    type: backend
spec:
  containers:
    - name: nginx-container
      image: nginx
      ports:
        - containerPort: 80
```

ReplicaSet Management

```
# Create a ReplicaSet
kubectl create -f rs-test.yml
kubectl apply -f rs-test.yml

# View ReplicaSets
kubectl get replicasets
kubectl get rs -o wide

# Scale a ReplicaSet
kubectl scale replicaset <replicaset-name> --replicas=<desired-replica-count>

# Delete a ReplicaSet
kubectl delete rs <replicaset-name>
kubectl delete -f rs-test.yml
```

ReplicaSet YAML Configuration

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: my-rs
  labels:
    name: my-rs
spec:
  replicas: 4
  selector:
    matchLabels:
      apptype: web-backend
  template:
    metadata:
```

```
    labels:
      apptype: web-backend
  spec:
    containers:
    - name: my-app
      image: nginx
      ports:
        - containerPort: 8080
```

Deployment Management

```
# Create a deployment
kubectl create deployment webnginx2 --image=nginx:latest --replicas=1

# View deployments
kubectl get deployments
kubectl describe deploy <deployment-name>

# Scale a deployment
kubectl scale deploy <deployment-name> --replicas=<desired-replica-count>

# Delete a deployment
kubectl delete deploy <deployment-name>
kubectl delete -f web-deploy.yml
```

Deployment YAML Configuration

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-deploy
  labels:
    name: my-deploy
```

```
spec:
  replicas: 4
  selector:
    matchLabels:
      apptype: web-backend
  strategy:
    type: RollingUpdate
  template:
    metadata:
      labels:
        apptype: web-backend
    spec:
      containers:
      - name: my-app
        image: nginx
        ports:
          - containerPort: 7070
```

Service Management

```
# View services
kubectl get svc

# Create a service from YAML
kubectl create -f service.yml

# Delete a service
kubectl delete svc <service-name>
```

Service YAML Configuration

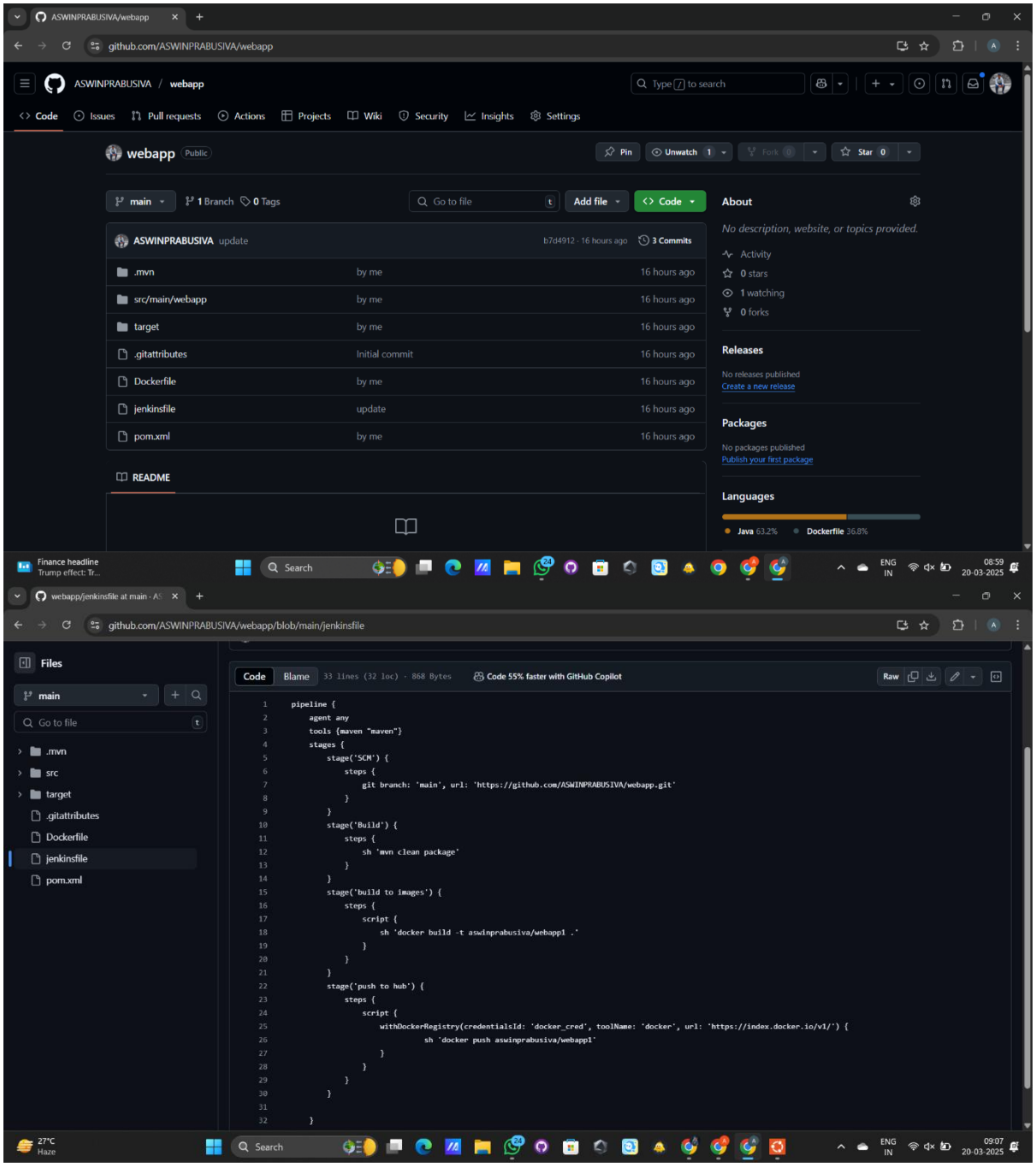
```
apiVersion: v1
kind: Service
```

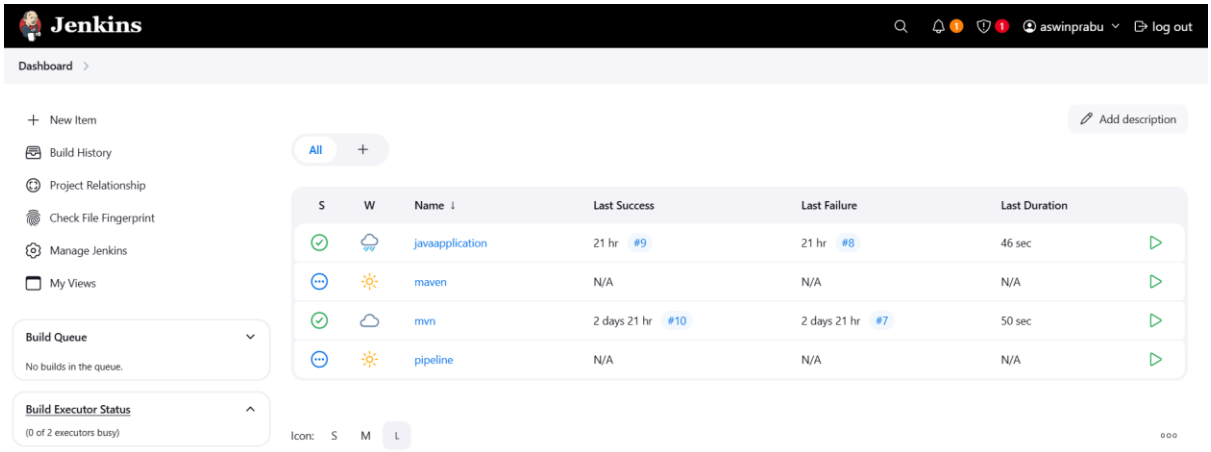
```
metadata:
  name: my-service
  labels:
    app: my-service
spec:
  type: NodePort
  ports:
    - port: 9000
      targetPort: 8080
      nodePort: 30002
  selector:
    apptype: web-backend
```

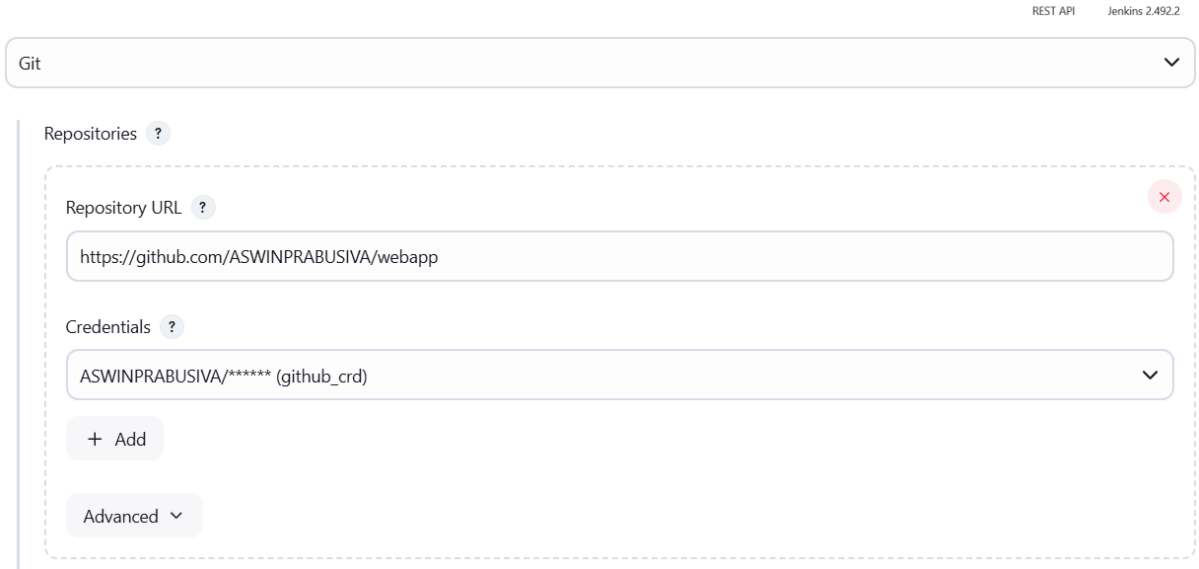
Namespace Management

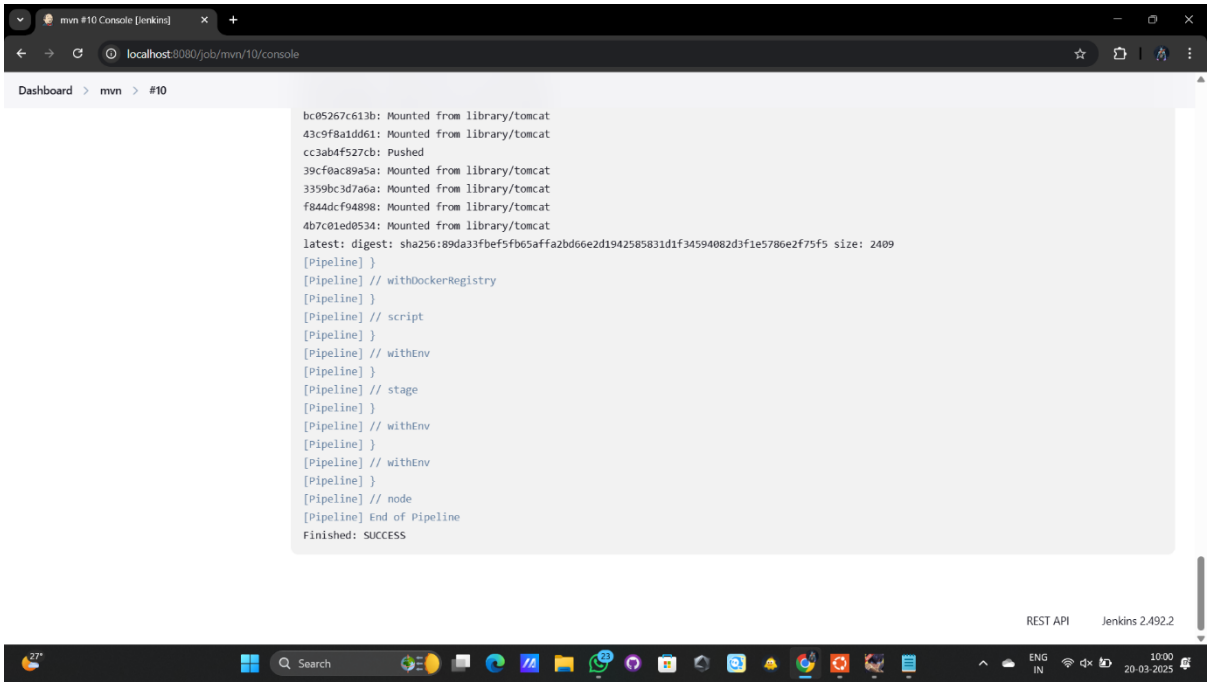
```
# Create a namespace
kubectl create namespace <namespace-name>
kubectl create ns my-bank

# View namespaces
kubectl get ns

# Switch to a namespace
kubectl config set-context --current --namespace=<namespace-name>
```

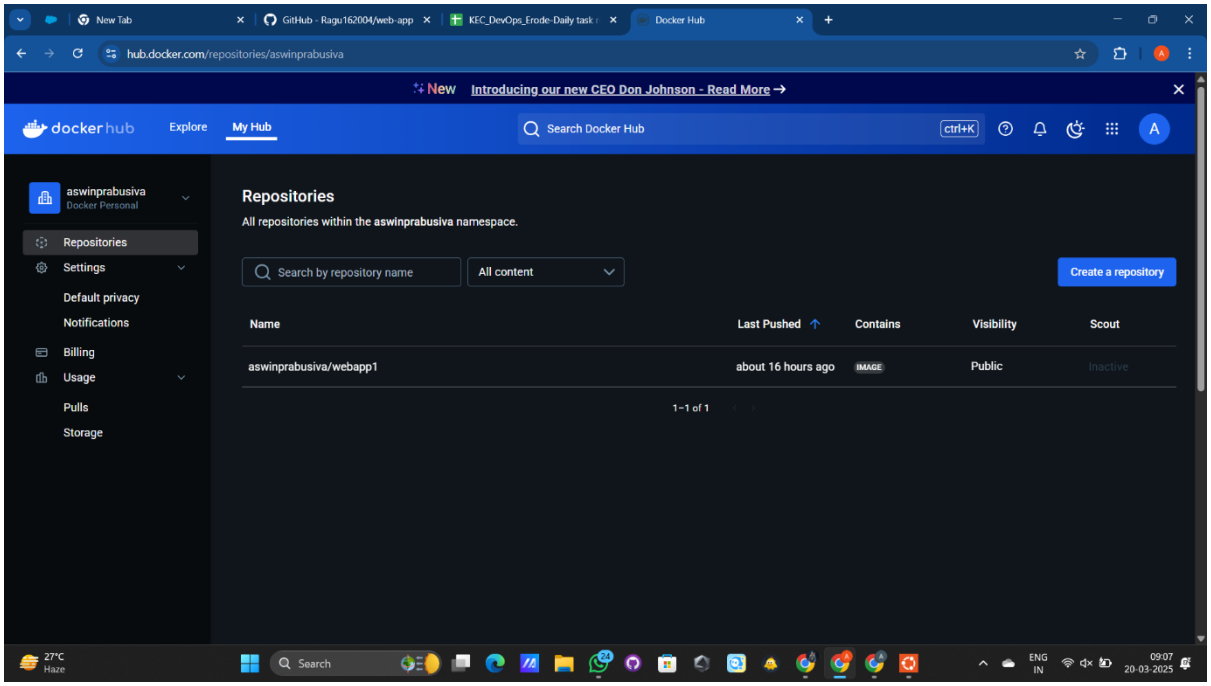
Automation using Jenkins push image from github to docker hub using jenkins automation:




Jenkins
aswinprabu
log out
Dashboard
New Item
Build History
Project Relationship
Check File Fingerprint
Manage Jenkins
My Views
Add description
All
S
W
Name
Last Success
Last Failure
Last Duration
javaapplication
21 hr #9
21 hr #8
46 sec
maven
N/A
N/A
N/A
mvn
2 days 21 hr #10
2 days 21 hr #7
50 sec
pipeline
N/A
N/A
N/A
Build Queue
No builds in the queue.
Build Executor Status
(0 of 2 executors busy)
Icon: S M L


REST API
Jenkins 2.492.2
Git
Repositories
Repository URL
https://github.com/ASWINPRABUSIVA/webapp
Credentials
ASWINPRABUSIVA/****** (github_crd)
Add
Advanced


mvn #10 Console [Jenkins]
localhost:8080/job/mvn/10/console
Dashboard > mvn > #10
bc05267c613b: Mounted from library/tomcat
43c9f8a1dd61: Mounted from library/tomcat
cc3ab4f527cb: Pushed
39cf0ac89a5a: Mounted from library/tomcat
3359bc3d7a6a: Mounted from library/tomcat
f844dcf94898: Mounted from library/tomcat
4b7c01ed0534: Mounted from library/tomcat
latest: digest: sha256:89da33fbef5fb65affa2bd66e2d1942585831d1f34594082d3f1e5786e2f75f5 size: 2409
[Pipeline] }
[Pipeline] // withDockerRegistry
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
REST API
Jenkins 2.492.2
Search
ENG IN
10:00 20-03-2025

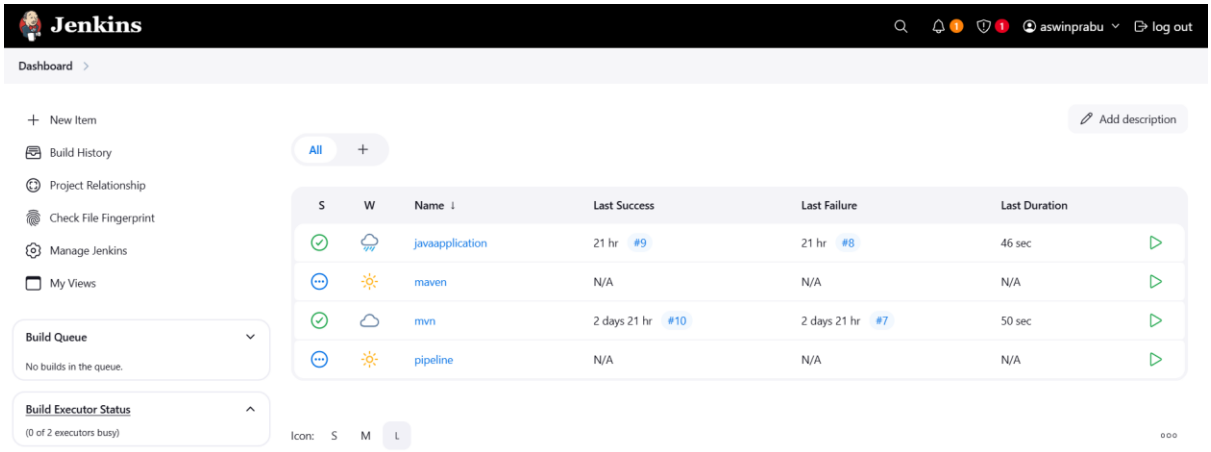

New Tab
GitHub - Ragu162004/web-app
KEC_DevOps_Erode-Daily task
Docker Hub
hub.docker.com/repositories/aswinprabusiva
New Introducing our new CEO Don Johnson - Read More
docker hub
Explore
My Hub
Search Docker Hub
ctrl+K
aswinprabusiva
Docker Personal
Repositories
Settings
Default privacy
Notifications
Billing
Usage
Pulls
Storage
Repositories
All repositories within the aswinprabusiva namespace.
Search by repository name
All content
Create a repository
Name
Last Pushed
Contains
Visibility
Scout
aswinprabusiva/webapp1
about 16 hours ago
IMAGE
Public
Inactive
1–1 of 1
27°C Haze
Search
ENG IN
09:07 20-03-2025

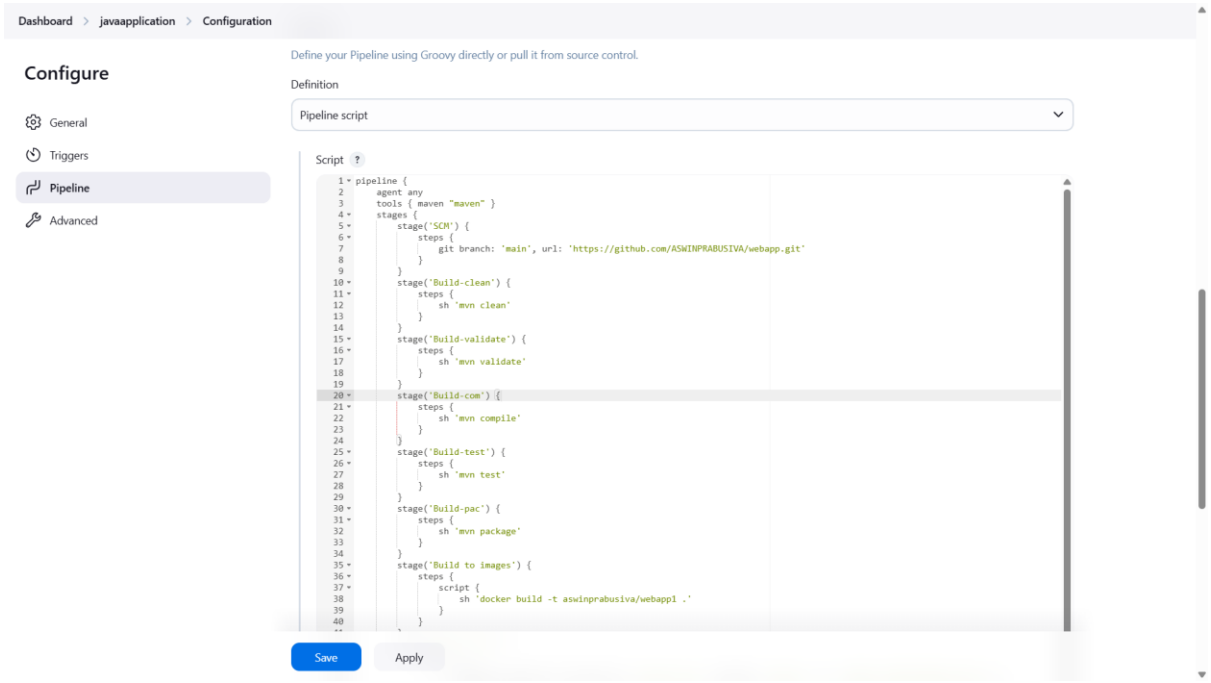
Minikube installation and mysql:

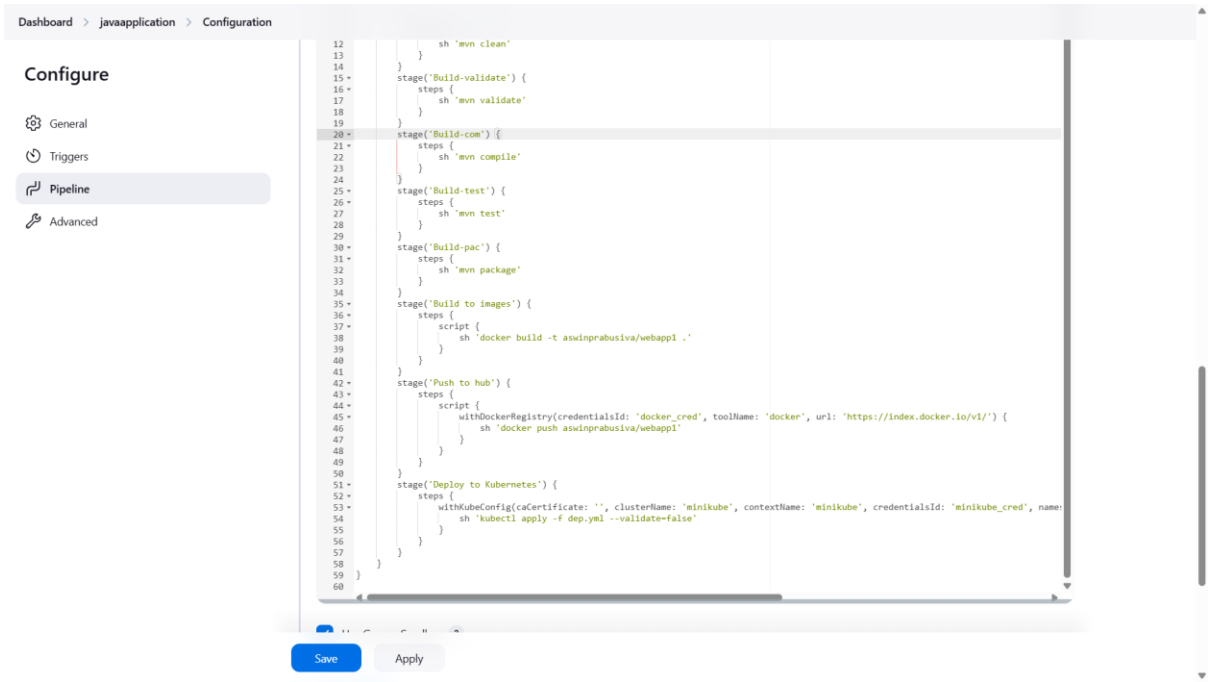
```
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Sat Mar 22 04:14:27 UTC 2025

  System load:  0.98                Processes:             38
  Usage of /:   0.9% of 1006.85GB   Users logged in:       0
  Memory usage: 10%                 IPv4 address for eth0: 172.25.205.8
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

This message is shown once a day. To disable it please create the
/home/anpu/.hushlogin file.
anpu@ASWINPRABU:~$ 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
```

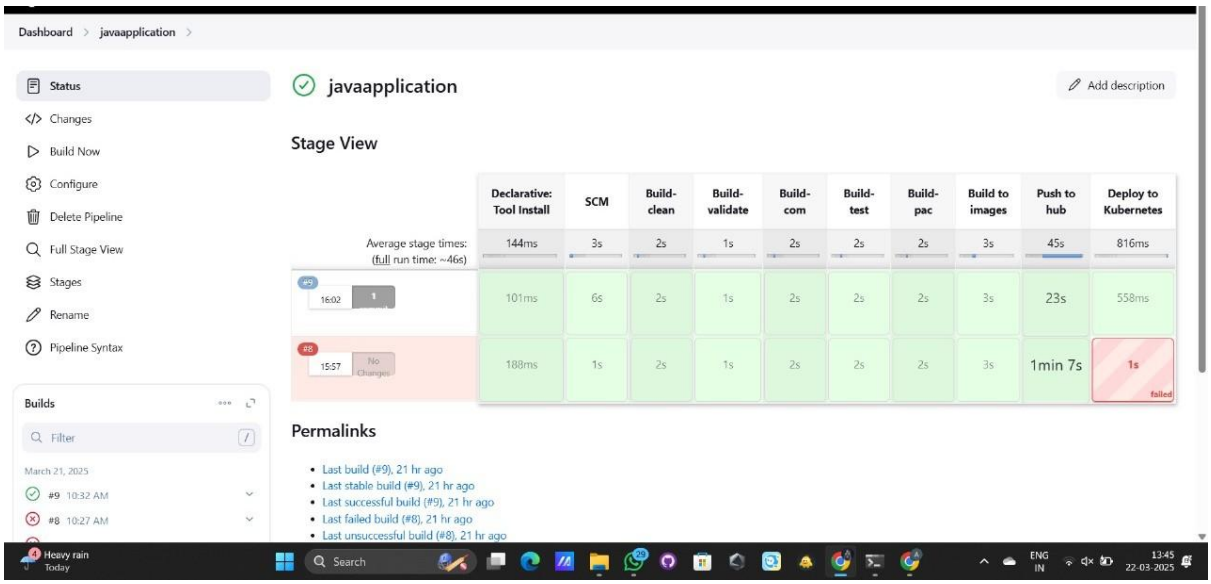
Kubernetes, Namespace:

```
 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

This message is shown once a day. To disable it please create the
/home/anpu/.hushlogin file.
anpu@ASWINPRABU:~$ 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
anpu@ASWINPRABU:~$ minikube service my-service
|-----------|------------|-------------|--------------------------|
| NAMESPACE |    NAME    | TARGET PORT |            URL           |
|-----------|------------|-------------|--------------------------|
| default   | my-service |        8000 | http://192.168.49.2:30009 |
|-----------|------------|-------------|--------------------------|
🏃  Starting tunnel for service my-service.
|-----------|------------|-------------|------------------------|
| NAMESPACE |    NAME    | TARGET PORT |          URL           |
|-----------|------------|-------------|------------------------|
| default   | my-service |             | http://127.0.0.1:39057 |
|-----------|------------|-------------|------------------------|
🎉  Opening service default/my-service in default browser...
👉  http://127.0.0.1:39057
❗  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.
anpu@ASWINPRABU:~$ curl http://192.168.49.2:30099/maven-web-app/
curl: (7) Failed to connect to 192.168.49.2 port 30099 after 0 ms: Couldn't connect to server
anpu@ASWINPRABU:~$ curl http://192.168.49.2:30009/maven-web-app/
<html>
<body>
<h2>Hello World!</h2>
</body>
</html>
anpu@ASWINPRABU:~$ kubectl port-forward svc/my-service 9010:8000
Forwarding from 127.0.0.1:9010 -> 8080
Forwarding from [::1]:9010 -> 8080
```


Jenkins
aswinprabu
log out
Dashboard
New Item
Build History
Project Relationship
Check File Fingerprint
Manage Jenkins
My Views
Build Queue
No builds in the queue.
Build Executor Status
(0 of 2 executors busy)
Add description
All
S
W
Name ↓
Last Success
Last Failure
Last Duration
javaapplication
21 hr #9
21 hr #8
46 sec
maven
N/A
N/A
N/A
mvn
2 days 21 hr #10
2 days 21 hr #7
50 sec
pipeline
N/A
N/A
N/A
Icon: S M L
REST API
Jenkins 2.492.2


Dashboard
javaapplication
Configuration
Configure
General
Triggers
Pipeline
Advanced
Define your Pipeline using Groovy directly or pull it from source control.
Definition
Pipeline script
Script
pipeline {
agent any
tools { maven "maven" }
stages {
stage('SCM') {
steps {
git branch: 'main', url: 'https://github.com/ASWINPRABUSIVA/webapp.git'
}
}
stage('Build-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-pac') {
steps {
sh 'mvn package'
}
}
stage('Build to images') {
steps {
script {
sh 'docker build -t aswinprabusiva/webapp1 .'
}
}
Save
Apply


Dashboard > javaapplication > Configuration
Configure
General
Triggers
Pipeline
Advanced
sh 'mvn clean'
stage('Build-validate')
sh 'mvn validate'
stage('Build-com')
sh 'mvn compile'
stage('Build-test')
sh 'mvn test'
stage('Build-pac')
sh 'mvn package'
stage('Build to images')
sh 'docker build -t aswinprabusiva/webapp1 .'
stage('Push to hub')
withDockerRegistry(credentialsId: 'docker_cred', toolName: 'docker', url: 'https://index.docker.io/v1/')
sh 'docker push aswinprabusiva/webapp1'
stage('Deploy to Kubernetes')
withKubeConfig(caCertificate: '', clusterName: 'minikube', contextName: 'minikube', credentialsId: 'minikube_cred'
sh 'kubectl apply -f dep.yml --validate=false'
Save
Apply


Dashboard > javaapplication >
Status
Changes
Build Now
Configure
Delete Pipeline
Full Stage View
Stages
Rename
Pipeline Syntax
Builds
Filter
March 21, 2025
#9 10:32 AM
#8 10:27 AM
javaapplication
Add description
Stage View
Declarative: Tool Install
SCM
Build-clean
Build-validate
Build-com
Build-test
Build-pac
Build to images
Push to hub
Deploy to Kubernetes
Average stage times:
(full run time: ~46s)
144ms
3s
2s
1s
2s
2s
2s
3s
45s
816ms
#9
16:02
101ms
6s
2s
1s
2s
2s
2s
3s
23s
558ms
#8
15:57
No Changes
188ms
1s
2s
1s
2s
2s
2s
3s
1min 7s
1s
failed
Permalinks
Last build (#9), 21 hr ago
Last stable build (#9), 21 hr ago
Last successful build (#9), 21 hr ago
Last failed build (#8), 21 hr ago
Last unsuccessful build (#8), 21 hr ago
Heavy rain
Today
Search
ENG
IN
13:45
22-03-2025

Output:

Hello World!