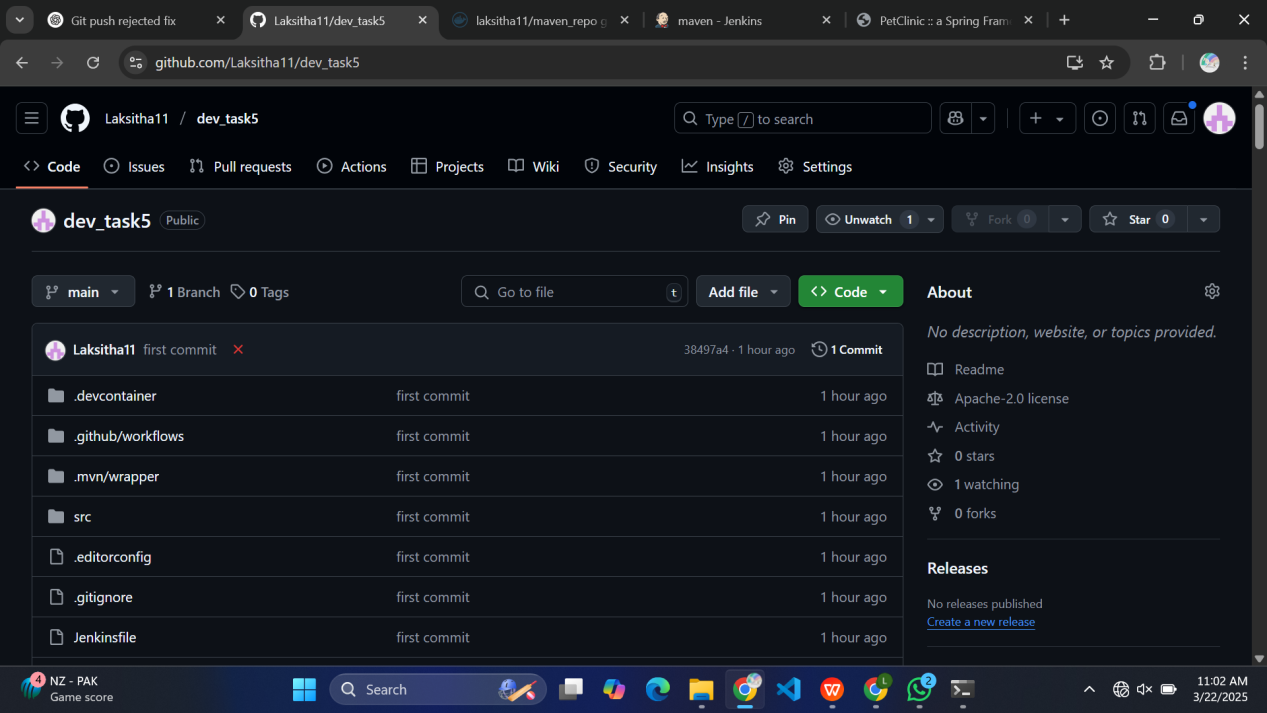
**PUSH THE GIVEN REPO GIVEN INTO YOUR GITHUB:-**

****

**THEN USE THE BELOW PIPELINESCRIPT ACCORDING TO YOUR OWN CREDENTIALS:-**

pipeline {

agent any

environment {

DOCKER\_IMAGE = "laksitha11/maven\_repo"

DOCKER\_TAG = "latest"

DOCKER\_CREDENTIALS\_ID = "docker"

GITHUB\_CREDENTIALS\_ID = "github\_seccred"

KUBECONFIG = "/var/lib/jenkins/.kube/config"

MAVEN\_HOME = "/opt/maven" // Set the correct Maven path

}

stages {

stage('Checkout Code') {

steps {

git credentialsId: GITHUB\_CREDENTIALS\_ID, url: 'https://github.com/Laksitha11/dev\_task5.git', branch: 'main'

}

}

stage('Build Application') {

steps {

script {

sh "${MAVEN\_HOME}/bin/mvn clean package -DskipTests"

}

}

}

stage('Run Maven Tests') {

steps {

script {

catchError(buildResult: 'SUCCESS', stageResult: 'FAILURE') {

sh "${MAVEN\_HOME}/bin/mvn test"

}

}

}

}

stage('Build Docker Image') {

steps {

script {

sh "docker build -t ${DOCKER\_IMAGE}:${DOCKER\_TAG} ."

}

}

}

stage('Push Docker Image') {

steps {

withDockerRegistry([credentialsId: DOCKER\_CREDENTIALS\_ID, url: '']) {

sh "docker push ${DOCKER\_IMAGE}:${DOCKER\_TAG}"

}

}

}

// Uncomment this stage when Kubernetes deployment is ready

// stage('Deploy to Kubernetes') {

// steps {

// script {

// sh '''

// chmod +x scripts/deploy.sh

// ./scripts/deploy.sh

// '''

// }

// }

// }

}

post {

success {

echo "✅ Deployment Successful!"

}

failure {

echo "❌ Deployment Failed!"

}

}

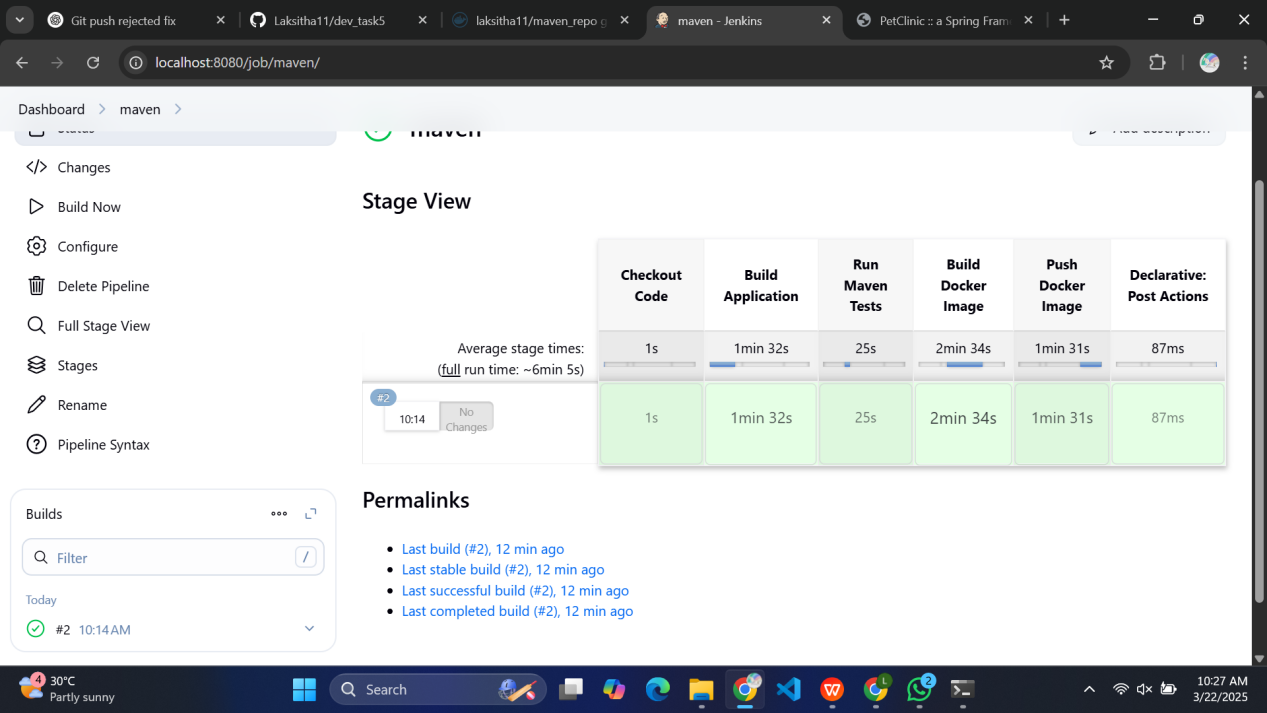
}

**BUILD THE PIPELINE:-**

**In Tools:-**

Use Maven 3.8.8

and Maven Home as /opt/home

****

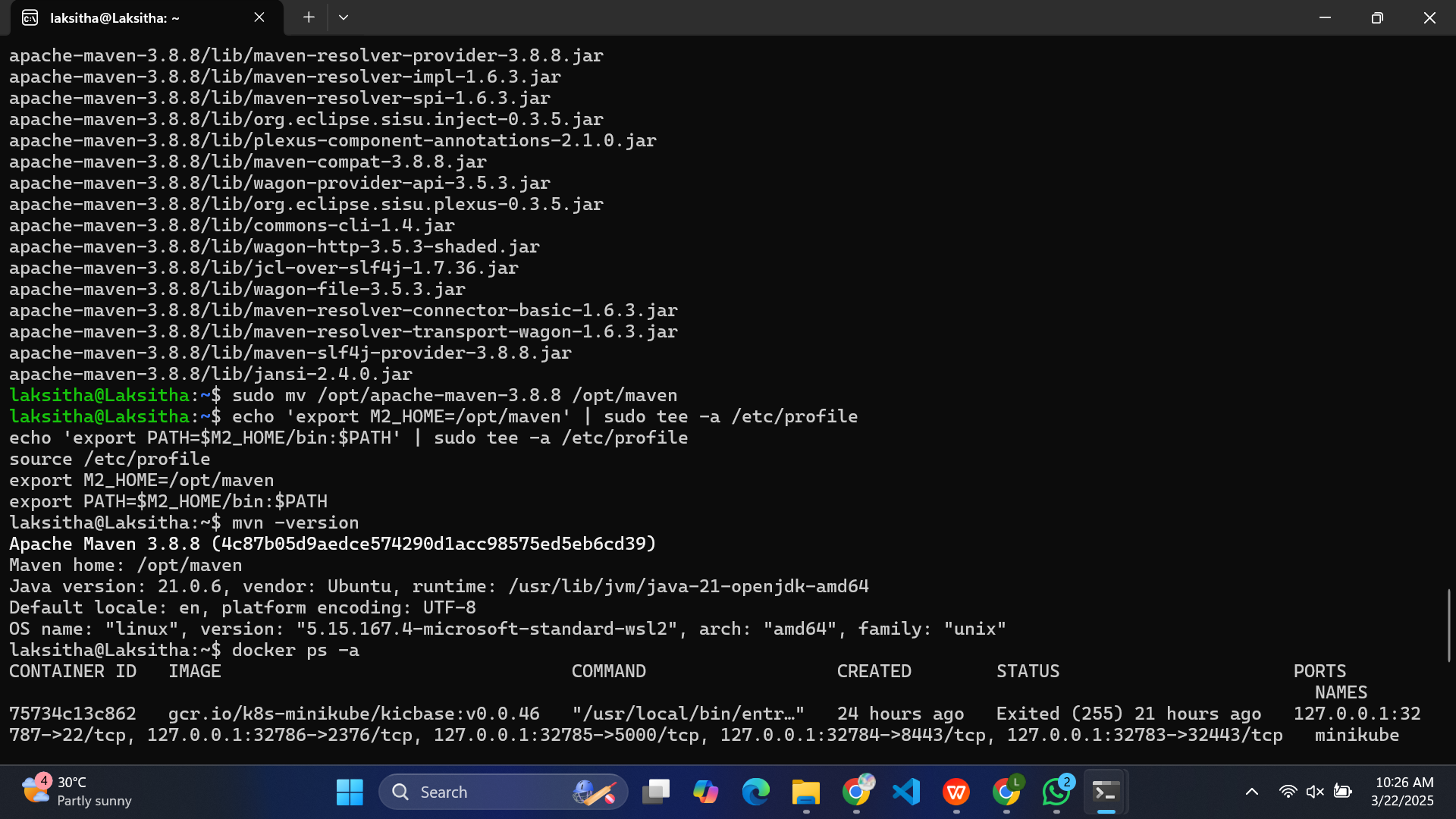
**AFTER BUILD:-**

docker ps -a

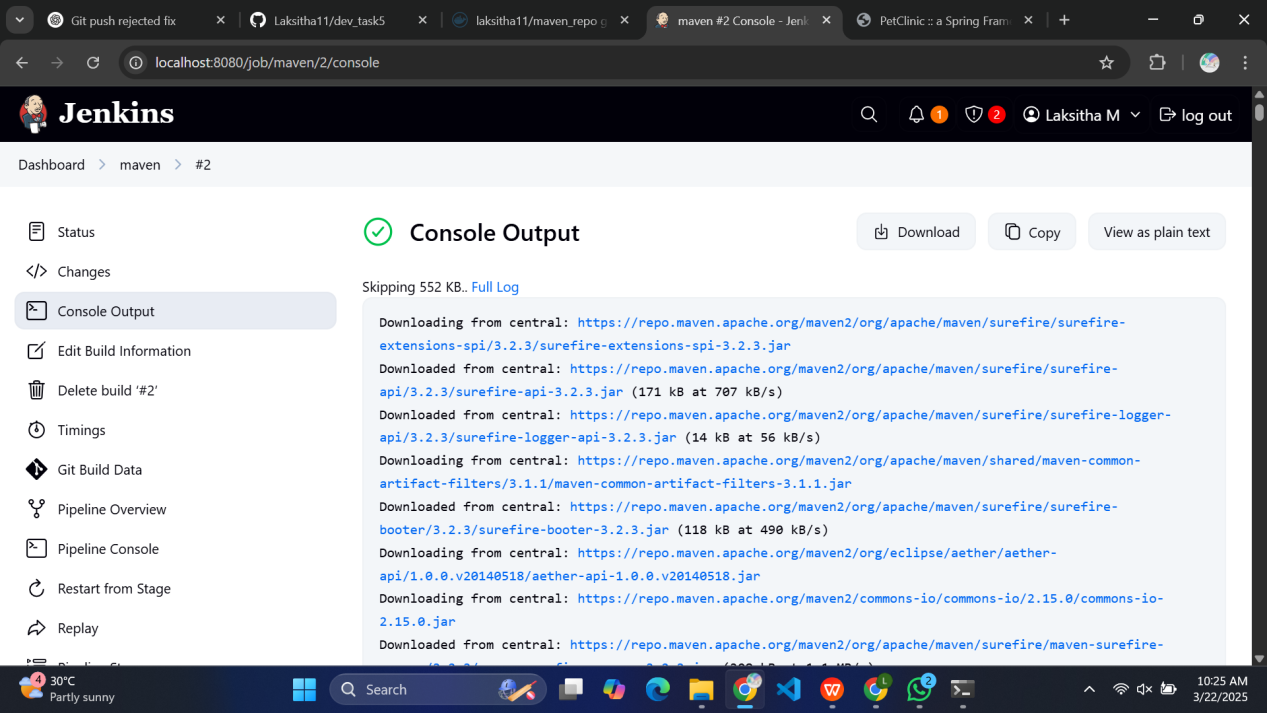
docker run -it laksitha11/maven\_repo:latest /bin/bash

/usr/local/tomcat# ps aux | grep tomcat

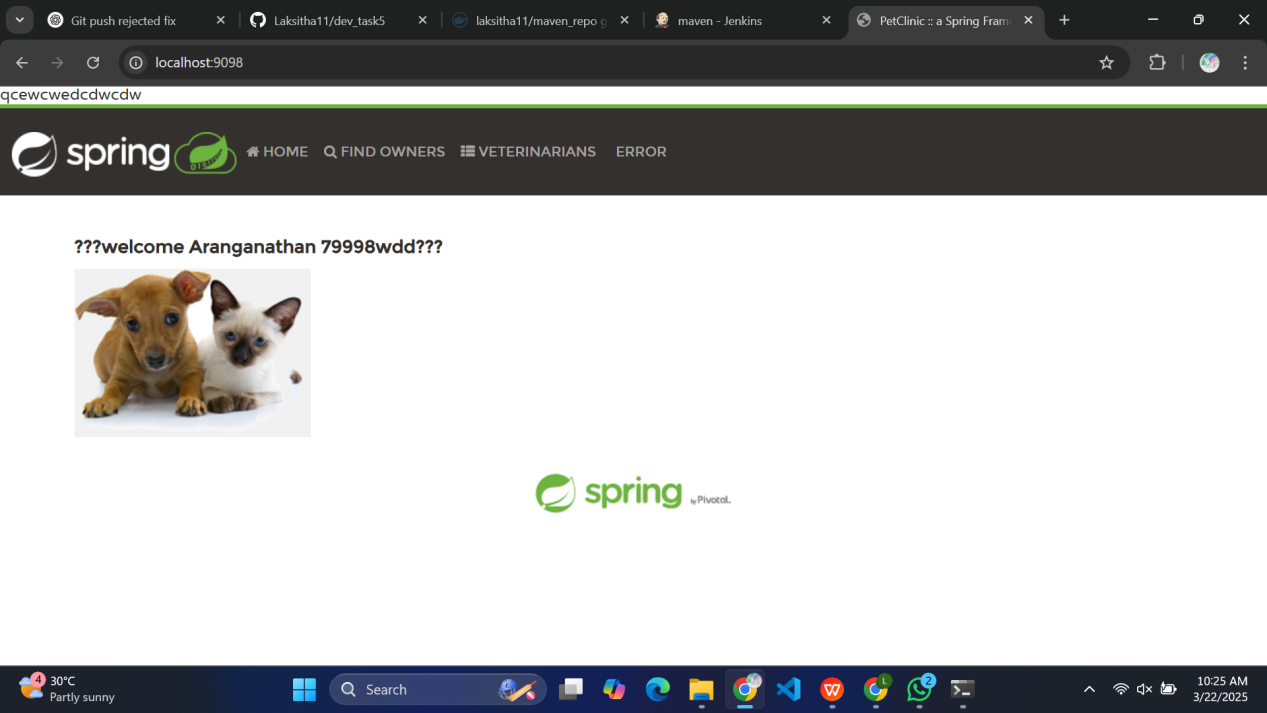
docker run -d -p 9098:8080 --name my-app laksitha11/maven\_repo

****

**BUILD SUCCESS OUTPUT:-**

****

**FINAL OUTPUT IN localhost:9098**

****