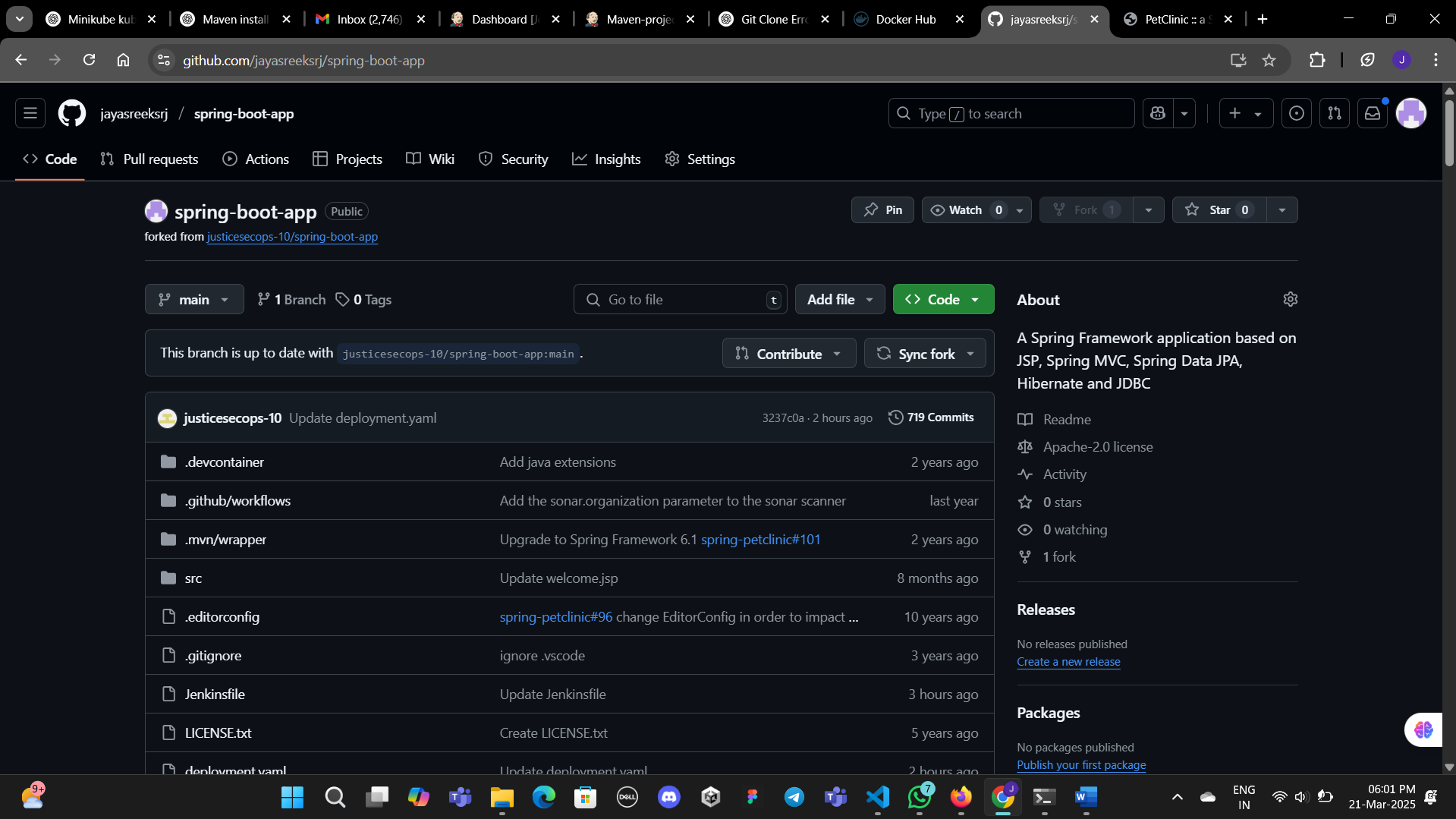
Github Clone:



Add the tool:

Name: Maven 3.9.6

Maven path :/opt/maven

After Build add the scripted Pipeline:

pipeline {

agent any

environment {

DOCKER\_IMAGE = "jayasree37/task-5"

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/jayasreeksrj/spring-boot-app.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 {

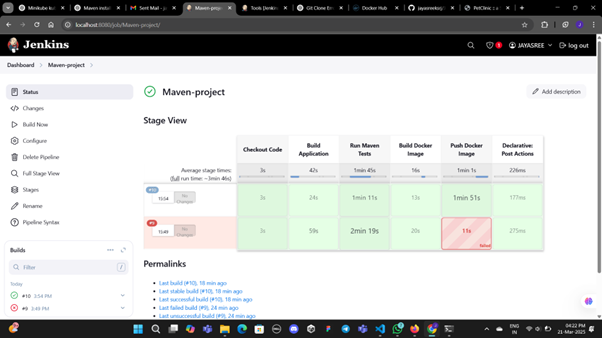
echo "❌ Deployment Failed!"

}

}

}

BUILD SUCCESSFUL:



After build:

docker ps -a

docker run -it jayasree37/task-5:latest /bin/bash

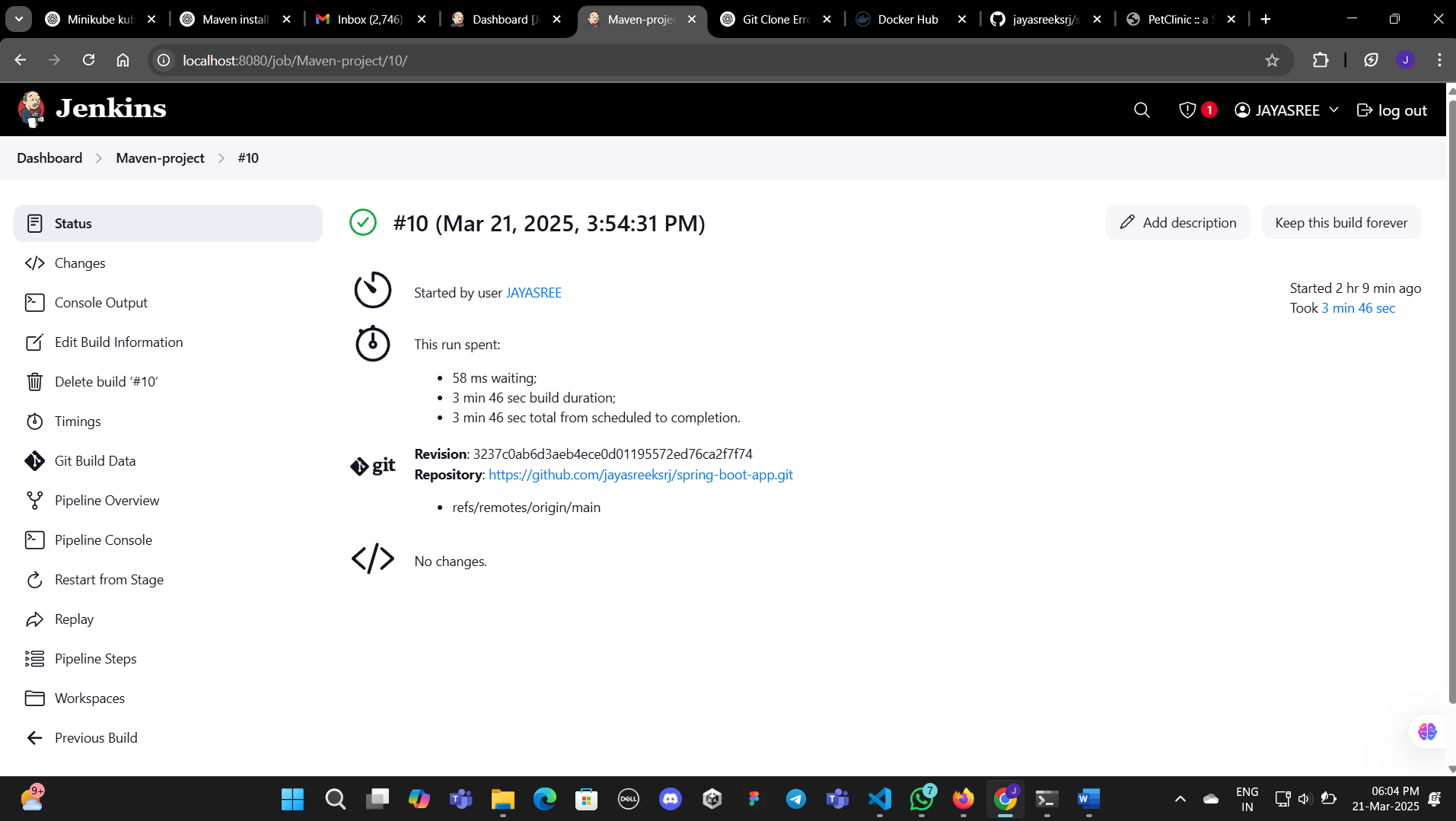
docker run -d -p 9090:8080 --name my-app jayasree37/task-5

MAVEN VERSION:

A screenshot of a computer screen

AI-generated content may be incorrect.

BUILD SUCCESSFUL:



FINAL OUTPUT IN LOCALHOST 9098:

A screenshot of a computer

AI-generated content may be incorrect.