Honors Final Assessment

Code Snippet :

pipeline {

agent any

environment {

EC2\_HOST = 'ec2-13.232.27.81.ap-south-1.compute.amazonaws.com'

SSH\_KEY = 'github' // Jenkins SSH credentials ID

REMOTE\_PATH = '/home/ec2-user/app'

DOCKERHUB\_CREDENTIALS = 'dockerhub-credentials-id'

DOCKER\_IMAGE = 'jatin1310/my-springboot-app:latest'

}

stages {

stage('Checkout Code') {

steps {

git url: 'https://github.com/Jatin-35/Springboot-project.git', branch: 'master'

}

}

stage('Build with Maven') {

agent {

docker {

image 'maven:3.9.5-eclipse-temurin-21'

args '-v /root/.m2:/root/.m2'

}

}

steps {

dir('flightservice') {

sh 'mvn clean package -DskipTests'

}

}

}

stage('Run Tests') {

steps {

dir('flightservice') {

sh 'mvn test'

}

}

post {

always {

junit 'flightservice/target/surefire-reports/\*.xml'

}

}

}

stage('Build & Push Docker Image') {

steps {

withCredentials([usernamePassword(credentialsId: env.DOCKERHUB\_CREDENTIALS, usernameVariable: 'USERNAME', passwordVariable: 'PASSWORD')]) {

sh """

docker build -t ${DOCKER\_IMAGE} .

echo "$PASSWORD" | docker login -u "$USERNAME" --password-stdin

docker push ${DOCKER\_IMAGE}

"""

}

}

}

stage('Deploy Docker Container on EC2') {

steps {

sshagent([env.SSH\_KEY]) {

sh """

ssh -o StrictHostKeyChecking=no ec2-user@${EC2\_HOST} << EOF

docker pull ${DOCKER\_IMAGE}

docker stop my-springboot-container || true

docker rm my-springboot-container || true

docker run -d --name my-springboot-container -p 8080:8080 ${DOCKER\_IMAGE}

EOF

"""

}

}

}

}

post {

success {

echo "✅ Docker Deployment successful! App should be live on EC2: http://${EC2\_HOST}:8080"

}

failure {

echo "❌ Pipeline failed. Check logs for issues."

}

always {

cleanWs()

}

}

}