Task 5-Spring Boot with Maven & Tomcat in Docker

https://github.com/kumarbharath2004/spring-boot-app
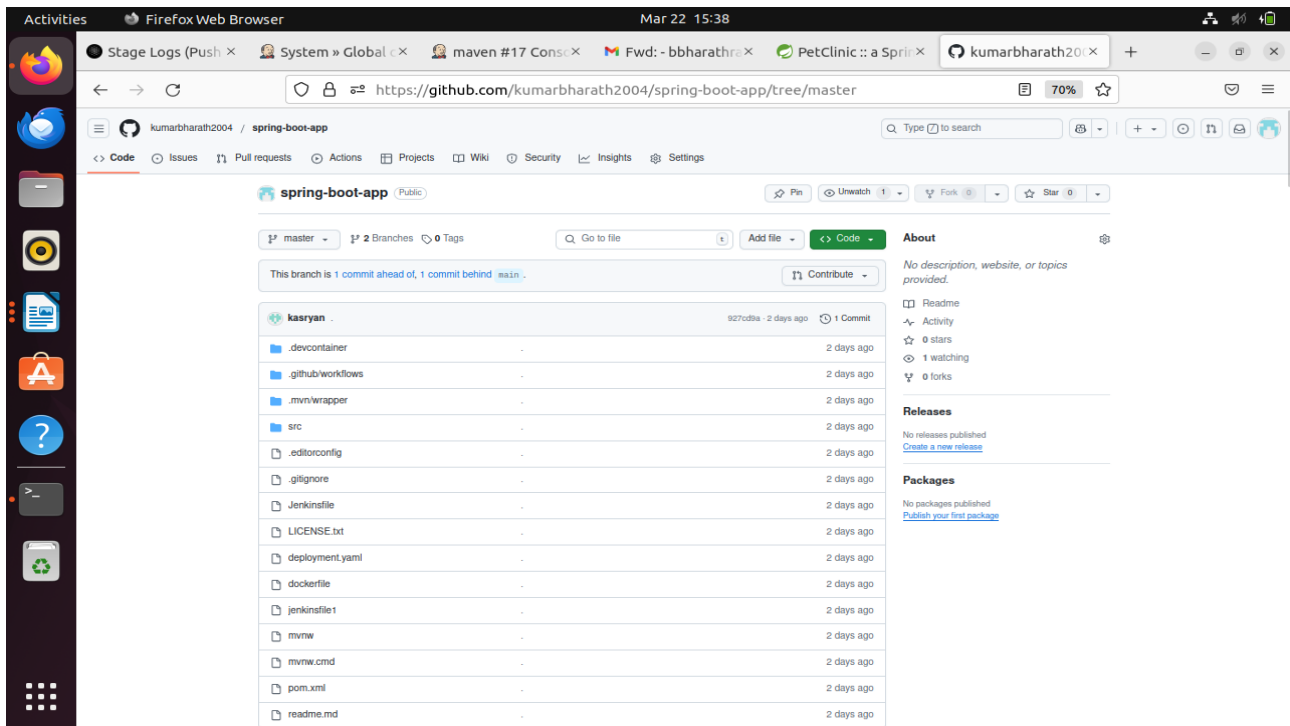
Plugins:

Maven Integration plugin

Pipeline:

Maven plugin

Tools:

Use maven 3.9.8 and Maven Home as /opt/home

Configure – pipeline:

```
pipeline {
    agent any
    environment {
        DOCKER_IMAGE = "bharathrajk/spring-boot-app"
        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 {
```

```groovy
                git credentialsId: GITHUB_CREDENTIALS_ID, url:
'https://github.com/kumarbharath2004/spring-boot-app.git', branch: 'master'
            }
        }
        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!"
```

```
        }
    }
}
```

Jenkins

Dashboard > maven

✓ **maven**   🖉 Add description

Stage View

		Checkout Code	Build Application (Success / Logs)	Run Maven Tests	Build Docker Image	Push Docker Image	Declarative: Post Actions
Average stage times: (Full run time: ~7min 50s)		3s		1min 31s	15s	3min 17s	1s
#17 15:16 No Changes		2s	18s	53s	5s	6min 25s	849ms
#16 15:01 No Changes		4s	54s	2min 10s	26s	9s failed	1s

Permalinks

- Last build (#17), 36 min ago
- Last stable build (#17), 36 min ago
- Last successful build (#17), 36 min ago
- Last failed build (#16), 51 min ago
- Last unsuccessful build (#16), 51 min ago
- Last completed build (#17), 36 min ago

REST API   Jenkins 2.492.2

Jenkins

Dashboard > maven > #17

✓ **Console Output**   ⬇ Download   Copy   View as plain text

```
Started by user bharathraj
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/maven
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Checkout Code)
[Pipeline] git
The recommended git tool is: NONE
using credential github_seccred
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/maven/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/kumarbharath2004/spring-boot-app.git # timeout=10
Fetching upstream changes from https://github.com/kumarbharath2004/spring-boot-app.git
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_ASKPASS to set credentials
 > git fetch --tags --force --progress -- https://github.com/kumarbharath2004/spring-boot-app.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 927cd9a1caf4126f56a6235016d333c8e1199d17 (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 927cd9a1caf4126f56a6235016d333c8e1199d17 # timeout=10
 > git branch -a -v --no-abbrev # timeout=10
 > git branch -D master # timeout=10
 > git checkout -b master 927cd9a1caf4126f56a6235016d333c8e1199d17 # timeout=10
Commit message: "."
 > git rev-list --no-walk 927cd9a1caf4126f56a6235016d333c8e1199d17 # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build Application)
[Pipeline] script
```
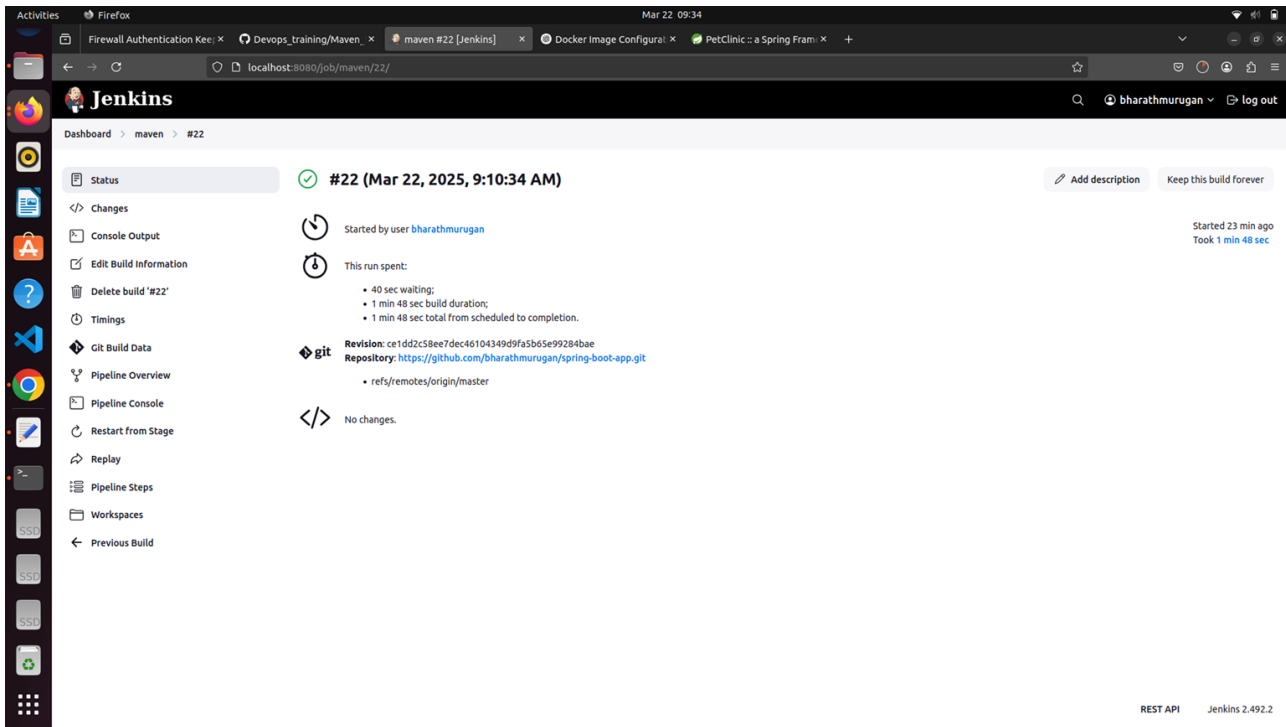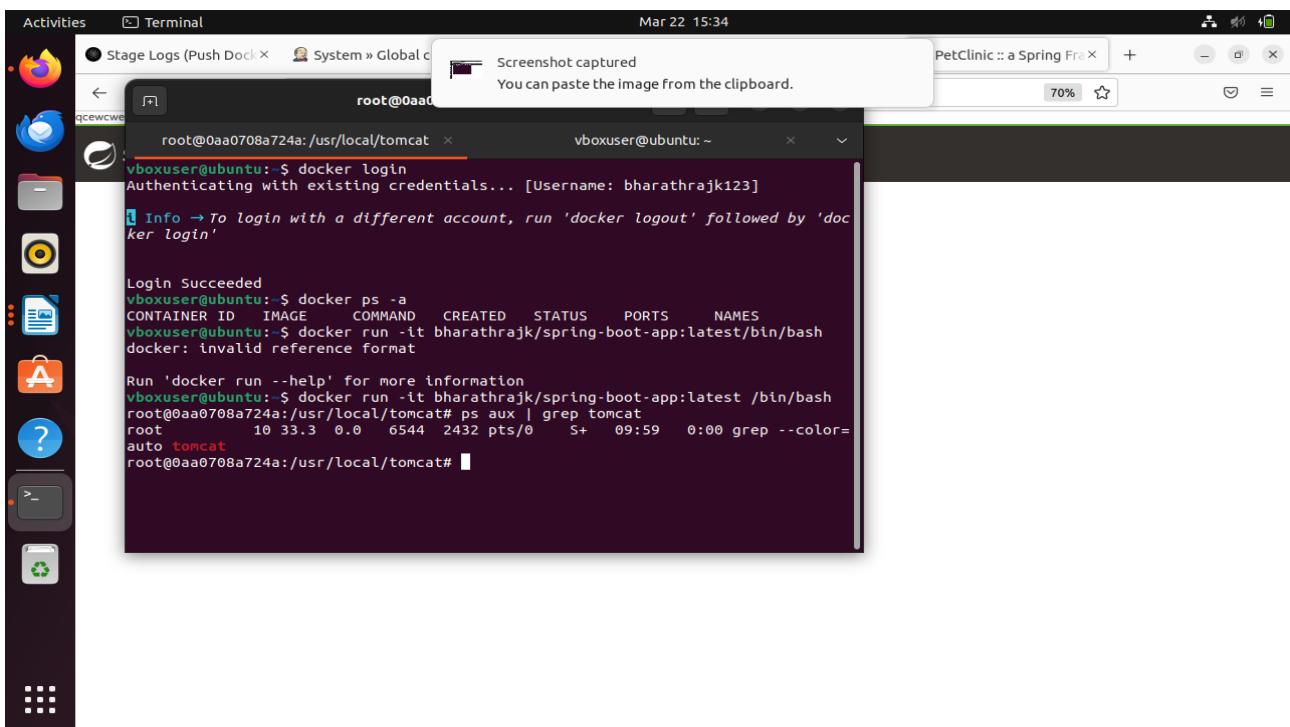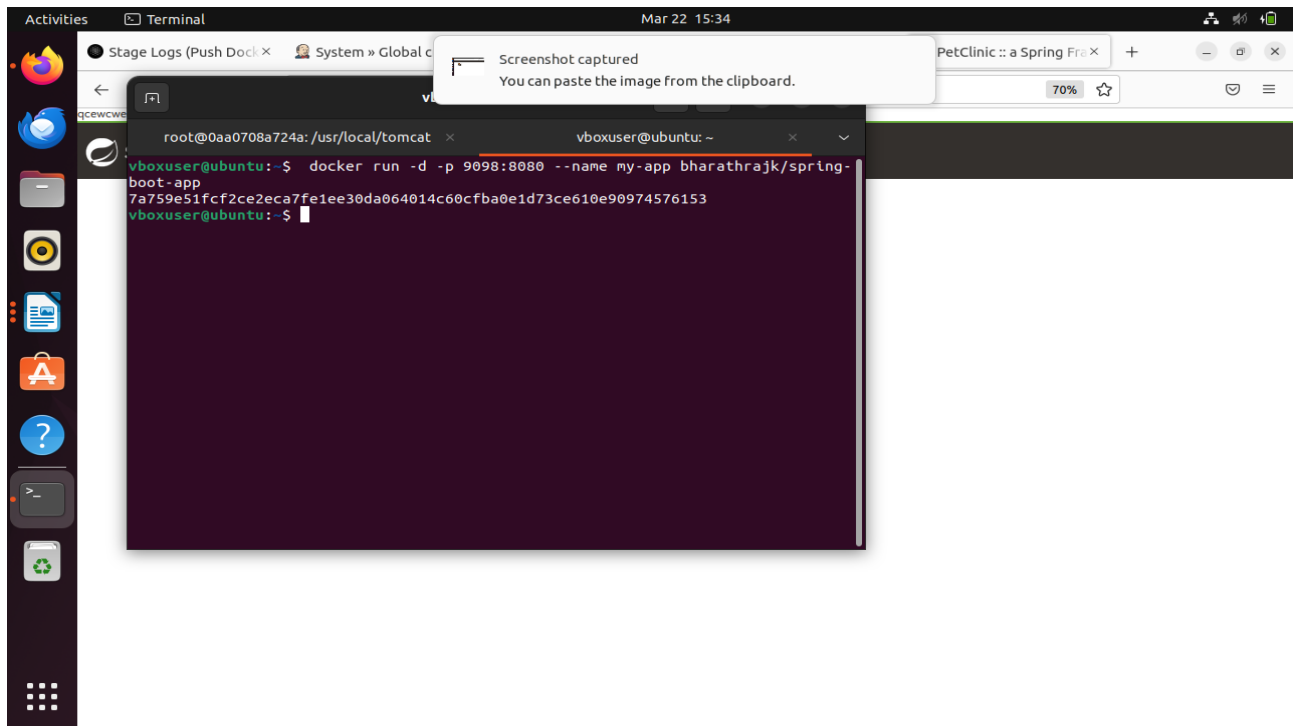
After build:

docker ps -a docker run -it bharathrajk/spring-boot-app:latest /bin/bash

root@f5c4f8921dbf:/usr/local/tomcat# ps aux | grep tomcat

root@f5c4f8921dbf:/usr/local/tomcat# /usr/local/tomcat/bin/catalina.sh run

In new terminal -

docker run -d -p 9091:8080 bharathrajk/spring-boot-app:latest