Stage View

	Checkout Code	Build Application	Run Maven Tests	Build Docker Image	Push Docker Image	Declarative: Post Actions
Average stage times: (full run time: ~12min 2s)	4s	17s	49s	2min 50s	31s	461ms
#8 13:45 No Changes	3s	26s	1min 15s	8min 28s	1min 33s	360ms
#7 13:36 No Changes	3s	24s	1min 11s	1s failed	1s failed	562ms
#6 13:28 No Changes	7s aborted	1s aborted	732ms aborted	891ms aborted	961ms aborted	

Permalinks

- Last build (#8), 10 min ago
- Last failed build (#7), 18 min ago

Sidebar

- Build Now
- Configure
- Delete Pipeline
- Full Stage View
- Stages
- Rename
- Pipeline Syntax

Builds

Filter

Today

- #8 08:15
- #7 08:06

```
39cf0ac89a5a: Mounted from library/tomcat
f844dcf94898: Mounted from library/tomcat
3359bc3d7a6a: Mounted from library/tomcat
4b7c01ed0534: Mounted from library/tomcat
d75543229a68: Pushed
latest: digest: sha256:3574ed4bcf651a41c3b842374569b417888a9328386dc9a22af4aa164e5341d0 size: 2413
[Pipeline] }
[Pipeline] // withDockerRegistry
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] echo
✅ Deployment Successful!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

dharani@DHARANI: /mnt/c/Windows/system32

Processing triggers for dbus (1.14.10-4ubuntu4.1) ...
Processing triggers for libc-bin (2.39-0ubuntu8.4) ...
dharani@DHARANI:/mnt/c/Windows/system32$ sudo usermod -aG docker jenkins
restart jenkins
sudo systemctl restart jenkins
dharani@DHARANI:/mnt/c/Windows/system32$ sudo usermod -aG docker jenkins
dharani@DHARANI:/mnt/c/Windows/system32$ sudo systemctl restart jenkins
dharani@DHARANI:/mnt/c/Windows/system32$ docker ps -a
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.45/containers/json?all=1": dial unix /var/run/docker.
sock: connect: permission denied
dharani@DHARANI:/mnt/c/Windows/system32$ sudo usermod -aG docker $USER
[sudo] password for dharani:
dharani@DHARANI:/mnt/c/Windows/system32$ sudo usermod -aG docker dharani
dharani@DHARANI:/mnt/c/Windows/system32$ docker ps
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.45/containers/json": dial unix /var/run/docker.sock:
connect: permission denied
dharani@DHARANI:/mnt/c/Windows/system32$ sudo systemctl restart docker
dharani@DHARANI:/mnt/c/Windows/system32$ sudo chmod 666 /var/run/docker.sock
dharani@DHARANI:/mnt/c/Windows/system32$ docker ps -a
CONTAINER ID   IMAGE   COMMAND   CREATED   STATUS   PORTS   NAMES
dharani@DHARANI:/mnt/c/Windows/system32$ docker ps
CONTAINER ID   IMAGE   COMMAND   CREATED   STATUS   PORTS   NAMES
dharani@DHARANI:/mnt/c/Windows/system32$ docker images
REPOSITORY   TAG   IMAGE ID   CREATED   SIZE
dharani400/maven   latest   757d85db007a   6 minutes ago   563MB
tomcat   latest   88b0f1cee84c   2 weeks ago   519MB
dharani@DHARANI:/mnt/c/Windows/system32$ docker ps -a
CONTAINER ID   IMAGE   COMMAND   CREATED   STATUS   PORTS   NAMES
dharani@DHARANI:/mnt/c/Windows/system32$ docker run -it kowsika12/maven:latest /bin/bash
ocal/tomcat# ps aux | grep tomcat
Unable to find image 'kowsika12/maven:latest' locally
latest: Pulling from kowsika12/maven
5a7813e071bf: Already exists
8dbbbc6af9dc: Already exists
a10b6847b9f1: Already exists
dcc1c5ea3c7d: Already exists
91e6cc55403a: Already exists
5d4660d0a9e9: Already exists
4f4fb700ef54: Already exists
e231914ca483: Already exists
abe8e163538c: Downloading 16.02MB/44.35MB
^C
dharani@DHARANI:/mnt/c/Windows/system32$ docker run -it dharani400/maven:latest /bin/bash
cat# ps aux | grep tomcat/usr/local/tomcat# ps aux | grep tomcatroot@5a48e3c58194:/usr/local/tomcat# /usr/local/tomcat# ps aux | grep tomcat exit
bash: /usr/local/tomcat#: No such file or directory
grep: exit: No such file or directory
root@5a48e3c58194:/usr/local/tomcat# exit
exit
dharani@DHARANI:/mnt/c/Windows/system32$ docker run -d -p 9098:8080 --name my-app dharani400/maven
90f26c5cc0722f18fe01c566f7b305f67bc02b8e4a414a93ad12cdd496cdf427
dharani@DHARANI:/mnt/c/Windows/system32$

qcewcwedcdwcdw

spring

HOME   Q FIND OWNERS   VETERINARIANS   ERROR
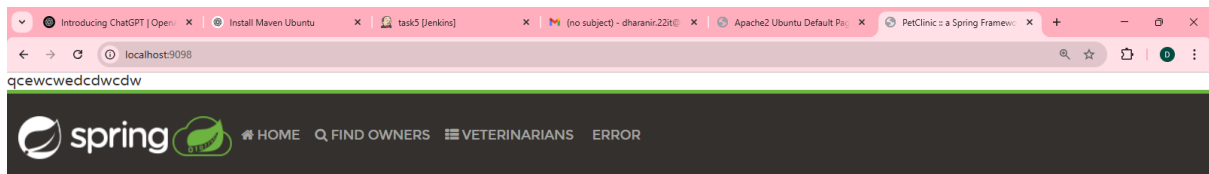
???welcome Aranganathan 79998wdd???

spring by Pivotal.

pipeline {

  agent any

  environment {

```
        DOCKER_IMAGE = "dharani400/maven"

        DOCKER_TAG = "latest"

        DOCKER_CREDENTIALS_ID = "docker"

        GITHUB_CREDENTIALS_ID = "github_seccred"

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

        MAVEN_HOME = "/usr/share/maven"  // Set the correct Maven path

    }


    stages {
        stage('Checkout Code') {

            steps {

                git credentialsId: GITHUB_CREDENTIALS_ID, url:
'https://github.com/dharaniramakrishnan/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!"
    }
}
}
```