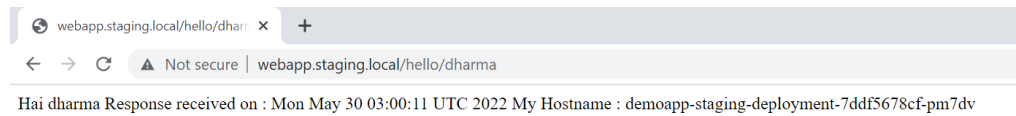
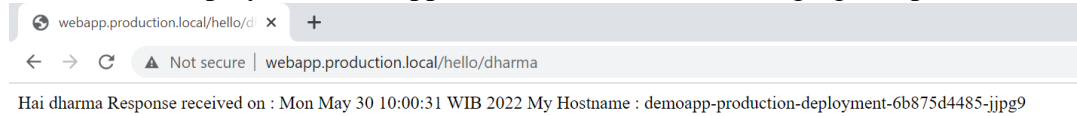


Technical Assessment Devops

A. How to use CI/CD Web with Jenkins

- Check status before deployment

- First check status deployment web application on environment staging and production



- Check all service on Kubernetes cluster and check version image deployment file

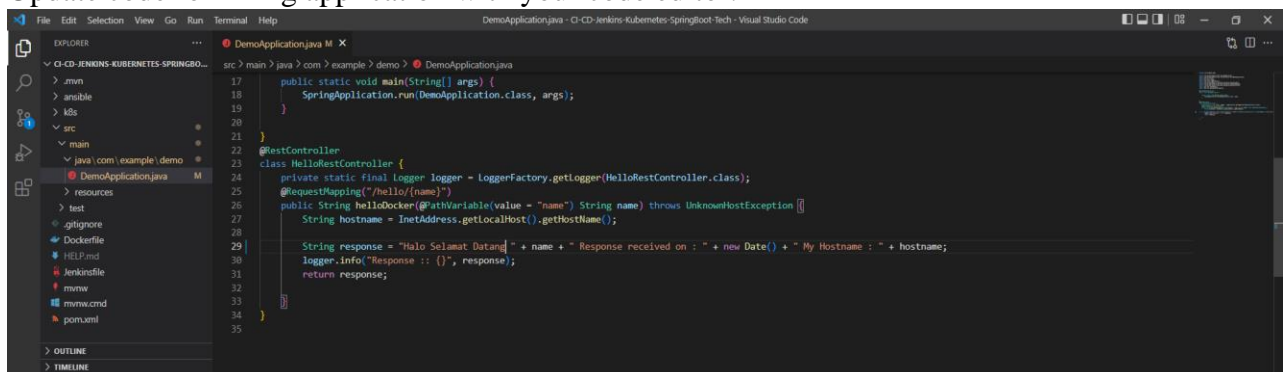
```
vagrant@ubuntu-focal: ~  
vagrant@ubuntu-focal:~$ kubectl get all -n staging  
NAME                                READY   STATUS    RESTARTS   AGE  
pod/demoapp-staging-deployment-7ddf5678cf-pm7dv  1/1     Running   1 (6m33s ago)  19h  
  
NAME                                TYPE          CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE  
service/demoapp-staging-service      ClusterIP     10.107.10.201 <none>        80/TCP      19h  
  
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE  
deployment.apps/demoapp-staging-deployment  1/1     1             1           19h  
  
NAME                                DESIRED   CURRENT   READY   AGE  
replicaset.apps/demoapp-staging-deployment-765b7cb799  0         0         0       19h  
replicaset.apps/demoapp-staging-deployment-7ddf5678cf  1         1         1       19h  
vagrant@ubuntu-focal:~$  
vagrant@ubuntu-focal:~$ kubectl get all -n production  
NAME                                READY   STATUS    RESTARTS   AGE  
pod/demoapp-production-deployment-6b875d4485-jjg9  1/1     Running   1 (6m54s ago)  20h  
  
NAME                                TYPE          CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE  
service/demoapp-production-service      ClusterIP     10.104.219.146 <none>        80/TCP      21h  
  
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE  
deployment.apps/demoapp-production-deployment  1/1     1             1           21h  
  
NAME                                DESIRED   CURRENT   READY   AGE  
replicaset.apps/demoapp-production-deployment-67c58cc899  0         0         0       21h  
replicaset.apps/demoapp-production-deployment-6b875d4485  1         1         1       20h  
vagrant@ubuntu-focal:~$  
vagrant@ubuntu-focal:~$  
vagrant@ubuntu-focal:~$
```

```
vagrant@ubuntu-focal: ~/ci-cd-jenkins-kubernetes-springboot-tech/k8s
vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$ cat prod/deployment.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    kubernetes.io/object: deployment
    app: demoapp
  name: demoapp-production-deployment
  namespace: production
spec:
  replicas: 1
  selector:
    matchLabels:
      app: demoapp
      env: production
  template:
    metadata:
      labels:
        app: demoapp
        env: production
    spec:
      imagePullSecrets:
        - name: dockerhubpull
      containers:
        - image: dharmatkj/demoapp:v1.0.0-production
          imagePullPolicy: "Always"
          name: demoapp
          ports:
            - containerPort: 8080vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$ |

vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$ cat staging/deployment.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    kubernetes.io/object: deployment
    app: demoapp
  name: demoapp-staging-deployment
  namespace: staging
spec:
  replicas: 1
  selector:
    matchLabels:
      app: demoapp
      env: staging
  template:
    metadata:
      labels:
        app: demoapp
        env: staging
    spec:
      imagePullSecrets:
        - name: dockerhubpull
      containers:
        - image: dharmatkj/demoapp:v1.0.0-staging
          imagePullPolicy: "Always"
          name: demoapp
          ports:
            - containerPort: 8080vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$
```

- **Developer create branch to update feature**

- Create new branch for develop application for example name branch **fix/fixing-endpoint-response**.
- Update code for fixing application with your code editor.



```
src > main > java > com > example > demo > DemoApplication.java
17 public static void main(String[] args) {
18     SpringApplication.run(DemoApplication.class, args);
19 }
20
21
22 @RestController
23 class HelloRestController {
24     private static final Logger logger = LoggerFactory.getLogger(HelloRestController.class);
25     @RequestMapping("/hello/{name}")
26     public String helloDocker(@PathVariable(value = "name") String name) throws UnknownHostException {
27         String hostname = InetAddress.getLocalHost().getHostName();
28
29         String response = "Halo Selamat Datang " + name + " Response received on : " + new Date() + " My Hostname : " + hostname;
30         logger.info("Response :: {}", response);
31         return response;
32     }
33 }
34
35
```

- Push branch to repository.

- Create pull request to merge from new source code fixing to branch environment **staging**

- Approve pull request for the next step.

The screenshot displays a GitHub Pull Request interface. At the top, the navigation bar includes links for Code, Issues, Pull requests (selected), Actions, Projects, Wiki, Security, Insights, and Settings. The main heading is 'Update endpoint response #1'. Below this, a green 'Open' button is followed by the text 'igstbagusdharma... wants to merge 1 commit into staging from fix/fixing-endpoint-response'. A toolbar shows 'Conversation' (0), 'Commits' (1), 'Checks' (0), and 'Files changed' (1). The conversation area shows a comment from 'igstbagusdharma...' stating 'fixing endpoint response'. Below the comment, a green box contains a build service warning and a green checkmark indicating no conflicts with the base branch. A red box highlights the 'Merge pull request' button. The right sidebar shows various settings like Reviews, Assignees, Labels, Projects, Milestone, and Development.

- Check changes source on branch staging.

staging ci-cd-jenkins-kubernetes-springboot-tech / src / main / java / com / example / demo / DemoApplication.java / <> Jump to ~ Go to file ...

igstbagusdharma Putra Update endpoint response Latest commit ffee98b 6 minutes ago History

1 contributor

34 lines (28 sloc) 1.13 KB

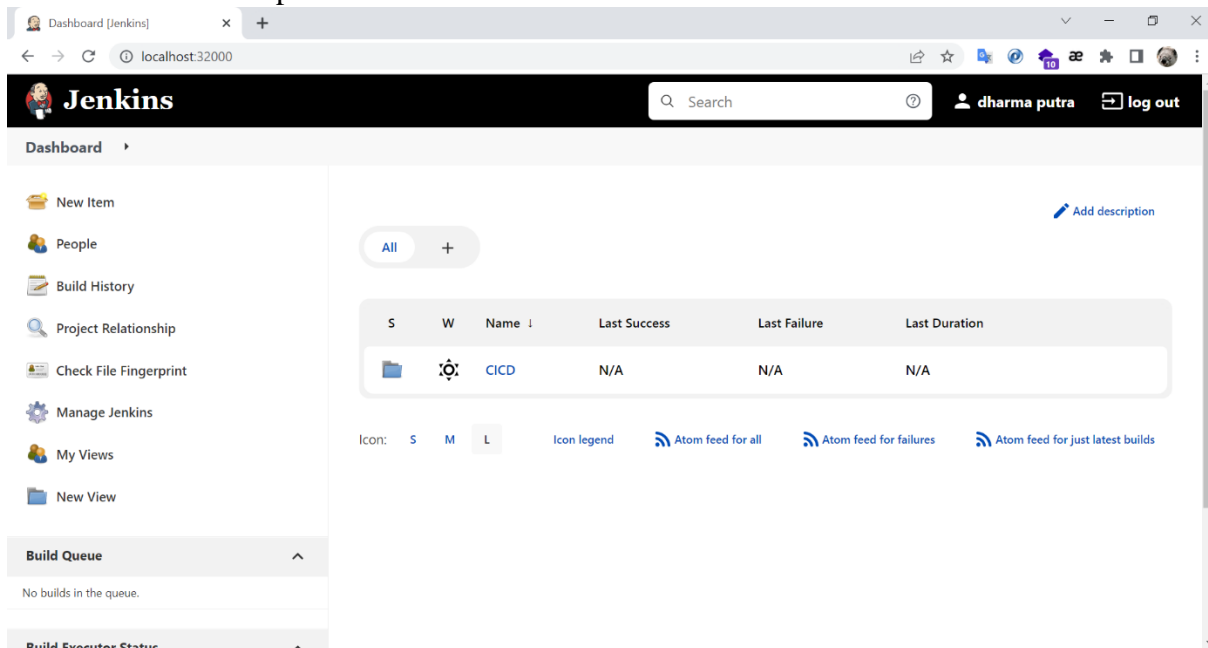
```

1 package com.example.demo;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5 import java.util.Date;
6 import org.slf4j.Logger;
7 import org.slf4j.LoggerFactory;
8 import org.springframework.web.bind.annotation.PathVariable;
9 import org.springframework.web.bind.annotation.RequestMapping;
10 import org.springframework.web.bind.annotation.RestController;
11 import java.net.InetAddress;
12 import java.net.UnknownHostException;
13
14 @SpringBootApplication
15 public class DemoApplication {
16
17     public static void main(String[] args) {
18         SpringApplication.run(DemoApplication.class, args);
19     }
20
21 }
22 @RestController
23 class HelloRestController {
24     private static final Logger logger = LoggerFactory.getLogger(HelloRestController.class);
25     @RequestMapping("/hello/{name}")
26     public String helloDocker(@PathVariable(value = "name") String name) throws UnknownHostException {
27         String hostname = InetAddress.getLocalHost().getHostName();
28
29         String response = "Halo Selamat Datang " + name + " Response received on : " + new Date() + " My Hostname : " + hostname;
30         logger.info("Response received on : " + new Date() + " My Hostname : " + hostname);
31         return response;
32     }
33 }

```

- **Run Jenkins to Deploy Web Applications Staging Environment**

- First Access Jenkins platform.



- Select job Jenkins to use deployment.

The screenshot shows the Jenkins Dashboard for the 'CICD' job. The left sidebar contains navigation links: Up, Status, Configure, New Item, Delete Folder, People, Build History, Project Relationship, Check File Fingerprint, Rename, and Credentials. The main content area displays the 'CICD' job details. A table lists the jobs with columns: S, W, Name, Last Success, Last Failure, and Last Duration. The 'cicd-application-springboot' job is highlighted with a red box.

S	W	Name	Last Success	Last Failure	Last Duration
✓	🔄	cicd-application-android	10 hr #15	10 hr #11	21 sec
✓	🔄	cicd-application-springboot	20 hr #67	20 hr #65	10 min

- Choice menu “Build with parameters”.

The screenshot shows the Jenkins Pipeline view for the 'cicd-application-springboot' job. The left sidebar contains navigation links: Up, Status, Changes, Build with Parameters, Configure, Delete Pipeline, Move, Full Stage View, Rename, and Pipeline Syntax. The 'Build with Parameters' option is highlighted with a red box. The main content area displays the 'Pipeline cicd-application-springboot' details, including the full project name and a 'Disable Project' button. Below the pipeline name, there is a 'Recent Changes' section and a 'Stage View' table.

Declarative: Checkout SCM	Fetch Source Code	Scanning Source Code	Build Staging	Build Production	Deploy to Staging	Deploy to Production
3s	3s	0ms	9min 48s	0ms	29s	0ms

- Please input fields according to the environment and click build to next process.

The screenshot shows the Jenkins Pipeline view for the 'cicd-application-springboot' job. The left sidebar contains navigation links: Up, Status, Changes, Build with Parameters, Configure, Delete Pipeline, Move, Full Stage View, Rename, and Pipeline Syntax. The 'Build with Parameters' option is highlighted with a red box. The main content area displays the 'Pipeline cicd-application-springboot' details, including the full project name and a 'Disable Project' button. Below the pipeline name, there is a 'Recent Changes' section and a 'Stage View' table. The 'Build' button is highlighted with a red box.

This build requires parameters:

GITHUB_BRANCH
 staging
 deploy to environment

SONARSCANNER
 ON
 scanning source code

TAG_VERSION
 v1.0.1-staging
 environment staging : v1.0.0-staging , environment production : v1.0.0-production

- Pipeline runs according to cicd flow design.

Dashboard > CICD > cicd-application-springboot

Build with Parameters
Configure
Delete Pipeline
Move
Full Stage View
Rename
Pipeline Syntax

Build History trend

Filter builds...

#68 May 30, 2022, 3:27 AM

Stage View

Recent Changes

Declarative: Checkout SCM	Fetch Source Code	Scanning Source Code	Build Staging	Build Production	Deploy to Staging	Deploy to Production
3s	3s	0ms	9min 48s	0ms	29s	0ms
Average stage times: (Average full run time: ~10min 40s)						
10min 35s						

May 29 1

- Check job log in pipeline process (Scanning Source Code, Build, and Deploy), code review can check on sonarqube

Stage Logs (Scanning Source Code)

Print Message -- Scanning (self time 189ms)

Checks if running on a Unix-like node (self time 110ms)

Shell Script -- docker inspect -f. "\$JD_TO_RUN" (self time 1s)

Shell Script -- pwd (self time 1s)

Shell Script -- mvn -v (self time 1s)

Shell Script -- mvn clean verify sonar:sonar -Dmaven.test.skip=true -Dsonar.projectKey=\${SONAR_PROJECT_KEY} -Dsonar.host.url=\${SONAR_URL} -Dsonar.login=\${SONAR_TOKEN} (self time 8min 5s)

```
[INFO] Sensor Java CPD Block Indexer (done) | time=51ms
[INFO] SCM Publisher SCM provider for this project is: git
[INFO] SCM Publisher 1 source file to be analyzed
[INFO] SCM Publisher 1/1 source file have been analyzed (done) | time=7979ms
[INFO] CPD Executor Calculating CPD for 1 file
[INFO] CPD Executor CPD calculation finished (done) | time=20ms
[INFO] Analysis report generated in 298ms, dir size=123.0 kB
[INFO] Analysis report compressed in 1149ms, zip size=19.3 kB
[INFO] Analysis report uploaded in 131ms
[INFO] ANALYSIS SUCCESSFUL, you can find the results at: http://192.168.1.8:31000/dashboard?id=****
[INFO] Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
[INFO] More about the report processing at http://192.168.1.8:31000/api/ce/task?id=AYETBvTWJ6hguOpLkoT
[INFO] Analysis total time: 30.840 s
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 05:50 min
```

Not secure | 192.168.1.8:31000/dashboard?id=cicd-sonar-springboot

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration

demo master

May 30, 2022, 10:33 AM Version 0.0.1-SNAPSHOT

Project Settings Project Information

QUALITY GATE STATUS

Some Quality Gate conditions on New Code were ignored because of the small number of New Lines

Passed
All conditions passed

MEASURES

New Code
Since May 28, 2022
Started 1 day ago

Overall Code

0 Bugs Reliability A

0 Vulnerabilities Security A

1 Security Hotspots E 0.0% Reviewed Security Review

10min Debt 1 Code Smells Maintainability A

0.0% 0.0% 0

Type here to search

11:27 AM 5/30/2022

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration

demo master May 30, 2022, 10:33 AM Version 0.0.1-SNAPSHOT

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

Search for files

demo src main/java/com/example/demo

	Lines of Code	Bugs	Vulnerabilities	Code Smells	Security Hotspots	Coverage	Duplications
main/java/com/example/demo	28	0	0	0	1	0.0%	0.0%
DemoApplication.java	28	0	0	0	1	0.0%	0.0%

1 of 1 shown

SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 9.4 (build 54424) - LGPL v3 - Community - Documentation - Plugins - Web API

Stage Logs (Build Staging)

Shell Script -- docker login -u dharmatkj -p \$(password) docker.io (self time 4s)

Shell Script -- docker build -t dharmatkj/demoapp:v1.0.1-staging . (self time 5min 52s)

Progress (1): 2.6/2.6 MB

Progress (1): 2.6/2.6 MB

Progress (1): 2.6 MB

Downloaded from central: <https://repo.maven.apache.org/maven2/com/google/guava/guava/28.2-android/guava-28.2-android.jar> (2.6 MB at 331 kB/s)

#12 246.1 [INFO] Replacing main artifact with repackaged archive

#12 246.1 [INFO] -----

#12 246.1 [INFO] BUILD SUCCESS

#12 246.1 [INFO] -----

#12 246.2 [INFO] Total time: 04:02 min

#12 246.2 [INFO] Finished at: 2022-05-30T03:40:40Z

#12 246.2 [INFO] -----

#12 DONE 246.3s

#7 [stage-1 1/2] FROM docker.io/library/openjdk:11-jdk-slim-buster@sha256:b4889a6ece62068b2a5111bbb3e98f76498e7a0d813405557d6795357dfa01a8

#7 sha256:ac19e05dc84348f7867cec9cc6a314ccf96b11a594153e6a7050a49986418b01

#7 CACHED

#13 [stage-1 2/2] COPY --from=build /app/target/*.jar app.jar

#13 sha256:8956ea5c8ff41c746487c62482714c41cadd23e9771dc5c8104df3e29ea51fc

Shell Script -- docker push dharmatkj/demoapp:v1.0.1-staging (self time 1min 50s)

Shell Script -- docker rmi dharmatkj/demoapp:v1.0.1-staging (self time 757ms)

Stage Logs (Deploy to Staging)

Print Message -- Deploy to staging (self time 92ms)

Checks if running on a Unix-like node (self time 119ms)

Shell Script -- docker inspect -f. "\$JD_TO_RUN" (self time 988ms)

Shell Script -- cp \$(myprivatekey) /root/id_rsa && chmod 400 /root/id_rsa (self time 1s)

Shell Script -- sed -i "s/([VERSION])/\$(TAG_VERSION)/g" ./k8s/staging/deployment.yml (self time 1s)

Shell Script -- ansible-playbook -i ansible/hosts ansible/playbook-staging.yml --extra-vars "SOURCE_PATH=\$(WORKSPACE)/k8s/staging" (self time 19s)

+ ansible-playbook -i ansible/hosts ansible/playbook-staging.yml --extra-vars 'SOURCE_PATH=/var/jenkins_home/workspace/CICD/cicd-application-springboot/k8s/staging'

PLAY [vagrant] *****

TASK [Gathering Facts] *****

ok: [192.168.1.99]

TASK [Copy Deployment File] *****

changed: [192.168.1.99]

TASK [Apply Kubeclt1] *****

changed: [192.168.1.99]

PLAY RECAP *****

192.168.1.99 : ok=3 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

- Check status update service application on Kubernetes after deploy

```
vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$ kubectl get all -n staging
NAME                                READY    STATUS    RESTARTS   AGE
pod/demoapp-staging-deployment-6944697df8-g8prc  1/1      Running   0           85s

NAME                                TYPE               CLUSTER-IP    EXTERNAL-IP   PORT(S)    AGE
service/demoapp-staging-service  ClusterIP          10.107.10.201 <none>        80/TCP      20h

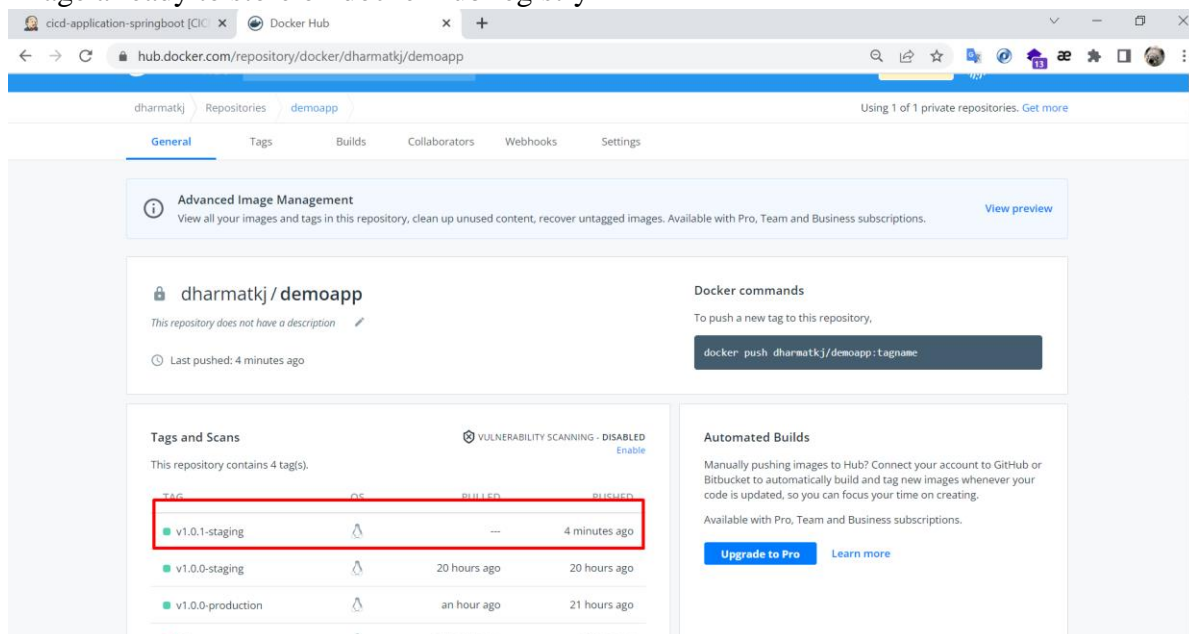
NAME                                READY    UP-TO-DATE   AVAILABLE   AGE
deployment.apps/demoapp-staging-deployment  1/1      1             1           20h

NAME                                DESIRED   CURRENT   READY   AGE
replicaset.apps/demoapp-staging-deployment-6944697df8  1         1         1       85s
replicaset.apps/demoapp-staging-deployment-765b7cb799  0         0         0       20h
replicaset.apps/demoapp-staging-deployment-7ddf5678cf  0         0         0       20h
vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$
```

- Check deployment file application to make sure tag version image updated

```
vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$ cat staging/deployment.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    kubernetes.io/object: deployment
    app: demoapp
    env: staging
  name: demoapp-staging-deployment
  namespace: staging
spec:
  replicas: 1
  selector:
    matchLabels:
      app: demoapp
      env: staging
  template:
    metadata:
      labels:
        app: demoapp
        env: staging
    spec:
      imagePullSecrets:
        - name: dockerhubpull
      containers:
        - image: dharmatkj/demoapp:v1.0.1-staging
          imagePullPolicy: Always
          name: demoapp
          ports:
            - containerPort: 8080
vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$
```

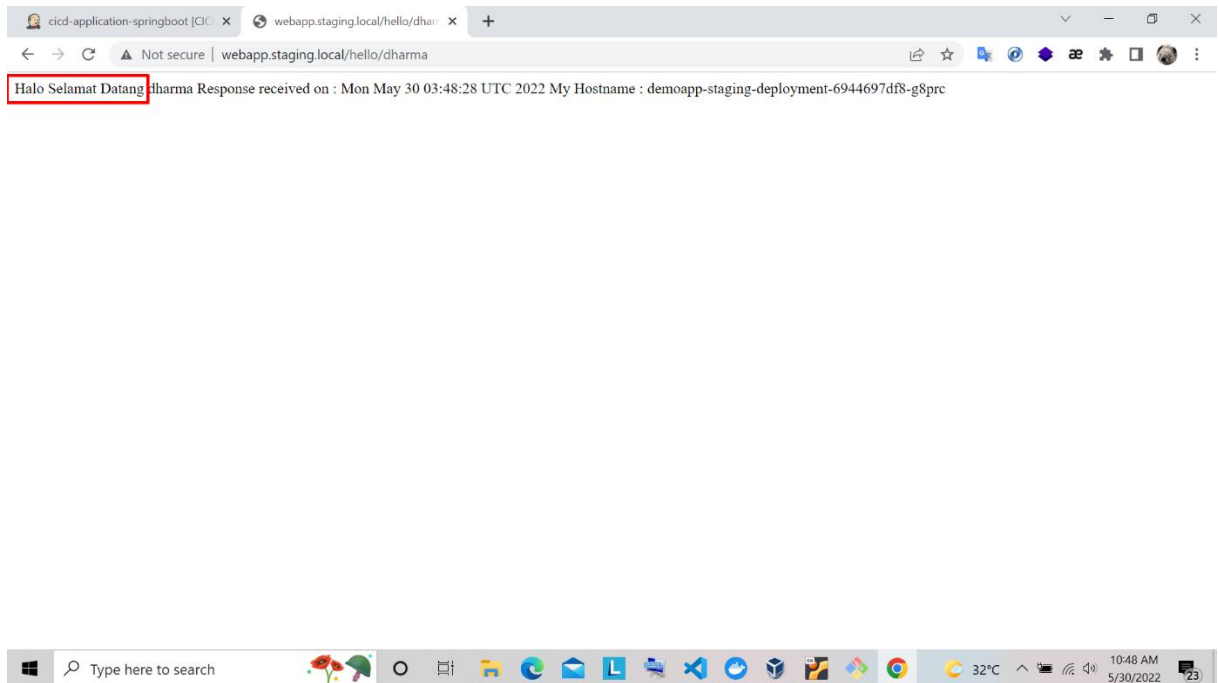
- Image already to store on docker hub registry



The screenshot shows the Docker Hub repository page for `dharmatkj/demoapp`. The page includes a header with the repository name and a description. Below the header, there is a section for "Tags and Scans" which lists the available tags and their build status. The tag `v1.0.1-staging` is highlighted with a red box, indicating it is the current version. The page also shows the Docker commands for pushing a new tag and the automated builds section.

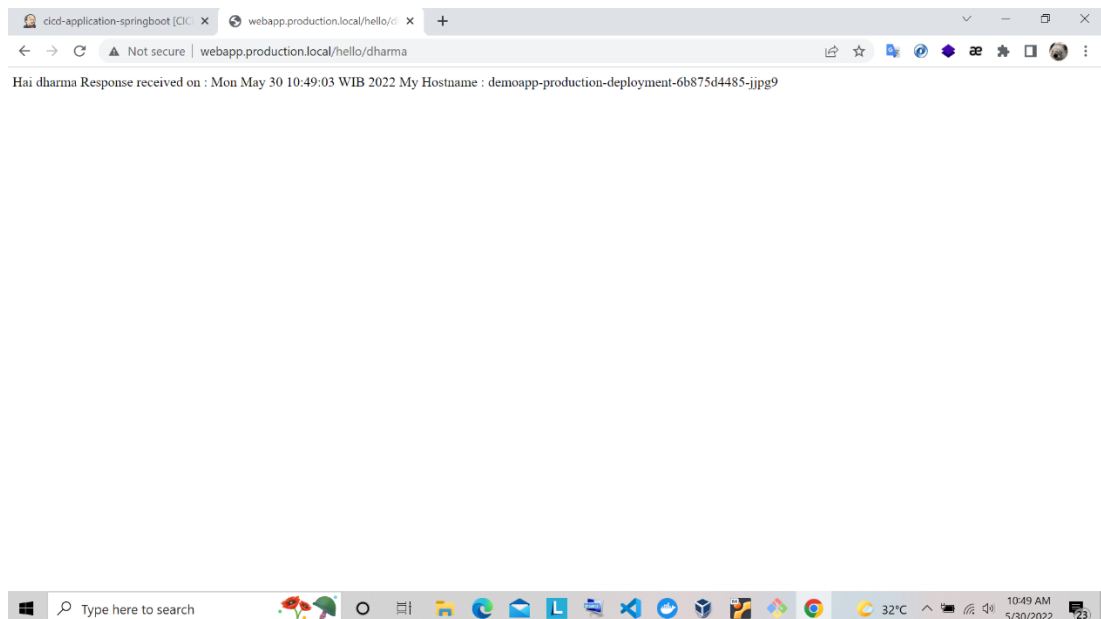
TAG	OS	BUILT	PUSHED
v1.0.1-staging	linux/amd64	---	4 minutes ago
v1.0.0-staging	linux/amd64	20 hours ago	20 hours ago
v1.0.0-production	linux/amd64	an hour ago	21 hours ago

- Try access endpoint web application after deploy.



- **Run Jenkins to Deploy Web Applications Production Environment**

- After staging already update, next to deploy environment production. Check endpoint url make sure still not yet update.



- Create merge request with branch after test on staging environment.

igstbagusdharmaputra / ci-cd-jenkins-kubernetes-springboot-tech Public

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

base: main ← compare: fix/fixing-endpoint-response ✓ Able to merge. These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

Create pull request

1 commit 1 file changed 1 contributor

Commits on May 30, 2022

Update endpoint response
igstbagusdharmaputra committed 38 minutes ago

Showing 1 changed file with 1 addition and 1 deletion.

Split Unified

```
src/main/java/com/example/demo/DemoApplication.java
@@ -26,7 +26,7 @@ class HelloRestController {
    public String helloDocker(@PathVariable(value = "name") String name) throws UnknownHostException {
        String hostname = InetAddress.getLocalHost().getHostName();
-       String response = "Hai " + name + " Response received on : " + new Date() + " My Hostname : " + hostname;
+       String response = "Halo Selamat Datang " + name + " Response received on : " + new Date() + " My Hostname : " + hostname;
        logger.info("Response :: {}", response);
        return response;
    }
}
```

- Approve merge request.

<> Code Issues Pull requests 1 Actions Projects Wiki Security Insights Settings

Update endpoint response #2

igstbagusdharmaputra wants to merge 1 commit into main from fix/fixing-endpoint-response

Conversation 0 Commits 1 Checks 0 Files changed 1

igstbagusdharmaputra commented now

No description provided.

Update endpoint response ffee98b

Add more commits by pushing to the fix/fixing-endpoint-response branch on igstbagusdharmaputra/ci-cd-jenkins-kubernetes-springboot-tech.

A build service has not been set up
We have detected a top-level @dockerfile. Pick from apps that can perform automatic builds.

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request You can also open this in GitHub Desktop or view command line instructions.

Write Preview H B I E <> @

Development
Successfully merging this pull request may close these issues.
None yet

- Check file changes on branch **main** for environment production.

main
ci-cd-jenkins-kubernetes-springboot-tech / src / main / java / com / example / demo / DemoApplication.java / <> Jump to
Go to file
...

igstbagudharmaputra Update endpoint response Latest commit #fee98b 39 minutes ago History

1 contributor

34 lines (28 sloc) | 1.13 KB
Raw
Blame
Copy
Download
Edit

```

1 package com.example.demo;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5 import java.util.Date;
6 import org.slf4j.Logger;
7 import org.slf4j.LoggerFactory;
8 import org.springframework.web.bind.annotation.PathVariable;
9 import org.springframework.web.bind.annotation.RequestMapping;
10 import org.springframework.web.bind.annotation.RestController;
11 import java.net.InetAddress;
12 import java.net.UnknownHostException;
13
14 @SpringBootApplication
15 public class DemoApplication {
16
17     public static void main(String[] args) {
18         SpringApplication.run(DemoApplication.class, args);
19     }
20
21 }
22 @RestController
23 class HelloRestController {
24     private static final Logger logger = LoggerFactory.getLogger(HelloRestController.class);
25     @RequestMapping("/hello/{name}")
26     public String helloDocker(@PathVariable(value = "name") String name) throws UnknownHostException {
27         String hostname = InetAddress.getLocalHost().getHostName();
28
29         String response = "Halo Selamat Datang " + name + " Response received on : " + new Date() + " My Hostname : " + hostname;
30         logger.info("Response : " + response);
31         return response;
32     }
33 }

```

- Go to Jenkins with run pipeline and input fields to deploy production.

Search
dharmaputra
log out

Dashboard
> CICD
> cicd-application-springboot

Up
Status
Changes
Build with Parameters
Configure
Delete Pipeline
Move
Full Stage View
Rename
Pipeline Syntax

Pipeline cicd-application-springboot

This build requires parameters:

GITHUB_BRANCH

main
deploy to environment

SONARSCANNER

OFF
scanning source code

TAG_VERSION

v1.0.1-production
environment staging : v1.0.0-staging , environment production : v1.0.0-production

Build

- Pipeline already to process jobs to update image applications and deploy.

Dashboard
> CICD
> cicd-application-springboot

Pipeline cicd-application-springboot

Full project name: CICD/cicd-application-springboot

[Add description](#)

[Disable Project](#)

Recent Changes

Stage View

	Declarative: Checkout SCM	Fetch Source Code	Scanning Source Code	Build Staging	Build Production	Deploy to Staging	Deploy to Production
Average stage times: (Average full run time: ~14min)	5s	4s	8min 25s	8min 51s	18s	33s	0ms
#60 May 30 10:57 3 commits	3s	4s			18s		
13min 27s							
#68 May 30 1	10s	5s	8min 25s	7min 55s		38s	

- Show log on job build

```
Stage Logs (Build Production)

Shell Script -- docker login -u dharmatkj -p $(password) docker.io (self time 4s)

Shell Script -- docker build -t dharmatkj/demoapp:v1.0.1-production . (self time 6min 46s)

Progress (2): 2.6 MB | 475/500 kB
Progress (2): 2.6 MB | 479/500 kB
Progress (2): 2.6 MB | 483/500 kB
Progress (2): 2.6 MB | 487/500 kB
Progress (2): 2.6 MB | 491/500 kB
Progress (2): 2.6 MB | 495/500 kB
Progress (2): 2.6 MB | 499/500 kB
Progress (2): 2.6 MB | 500 kB

Downloaded from central: https://repo.maven.apache.org/maven2/com/google/guava/guava/28.2-android/guava-28.2-android.jar (2.6 MB at 369 kB/s)
#12 275.6 Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-lang3/3.7/commons-lang3-3.7.jar (500 kB at 70 kB/s)
#12 276.4 [INFO] Replacing main artifact with repackaged archive
#12 276.4 [INFO] -----
#12 276.4 [INFO] BUILD SUCCESS
#12 276.4 [INFO] -----
#12 276.5 [INFO] Total time: 04:33 min
#12 276.5 [INFO] Finished at: 2022-05-30T04:02:21Z
#12 276.5 [INFO] -----
#12 DONE 276.6s

Shell Script -- docker push dharmatkj/demoapp:v1.0.1-production (self time 45s)

Shell Script -- docker rmi dharmatkj/demoapp:v1.0.1-production (self time 931ms)
```

- Show logs deploy to make sure job success.

```
Stage Logs (Deploy to Production)

Print Message -- Deploy to production (self time 82ms)

Checks if running on a Unix-like node (self time 113ms)

Shell Script -- docker inspect -f "${JD_TO_RUN}" (self time 1s)

Shell Script -- cp $(myprivatekey) /root/id_rsa && chmod 400 /root/id_rsa (self time 1s)

Shell Script -- sed -i 's/([VERSION])/TAG_VERSION/g' ./k8s/prod/deployment.yml (self time 1s)

Shell Script -- ansible-playbook -i ansible/hosts ansible/playbook-production.yml --extra-vars "SOURCE_PATH=${WORKSPACE}/k8s/prod" (self time 15s)

+ ansible-playbook -i ansible/hosts ansible/playbook-production.yml --extra-vars 'SOURCE_PATH=/var/jenkins_home/workspace/CICD/cicd-application-springboot/k8s/prod'

PLAY [vagrant] *****

TASK [Gathering Facts] *****
ok: [192.168.1.99]

TASK [copy Deployment File] *****
changed: [192.168.1.99]

TASK [Apply Kubectl] *****
changed: [192.168.1.99]

PLAY RECAP *****
192.168.1.99      : ok=3   changed=2   unreachable=0   failed=0   skipped=0   rescued=0   ignored=0

Shell Script (self time 675ms)
```

- Check service on Kubernetes after deployment update.

```
vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$ kubectl get all -n production
NAME                                READY   STATUS    RESTARTS   AGE
pod/demoapp-production-deployment-6c6c56fdb6-hglz4  1/1     Running   0           4m24s

NAME                                TYPE           CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
service/demoapp-production-service  ClusterIP      10.104.219.146 <none>        80/TCP      22h

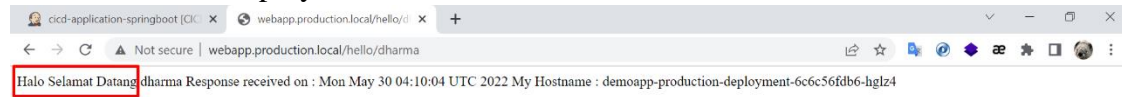
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/demoapp-production-deployment  1/1     1             1           22h

NAME                                DESIRED   CURRENT   READY   AGE
replicaset.apps/demoapp-production-deployment-67c58cc899  0         0         0       22h
replicaset.apps/demoapp-production-deployment-6b875d4485  0         0         0       21h
replicaset.apps/demoapp-production-deployment-6c6c56fdb6  1         1         1       4m24s
vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$
```

- Make sure to changes tag version image on deployment file.

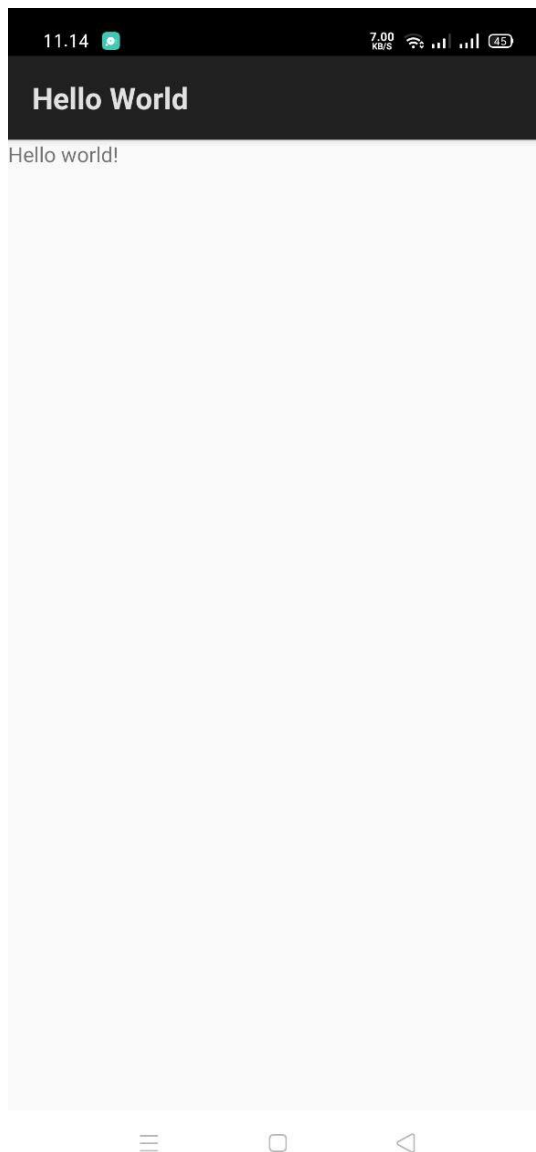
```
vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$ cat prod/deployment.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    kubernetes.io/object: deployment
    app: demoapp
    env: production
  name: demoapp-production-deployment
  namespace: production
spec:
  replicas: 1
  selector:
    matchLabels:
      app: demoapp
      env: production
  template:
    metadata:
      labels:
        app: demoapp
        env: production
    spec:
      imagePullSecrets:
        - name: dockerhubpull
      containers:
        - image: dharmatkj/demoapp:v1.0.1-production
          imagePullPolicy: "Always"
          name: demoapp
          ports:
            - containerPort: 8080vagrant@ubuntu-focal:~/ci-cd-jenkins-kubernetes-springboot-tech/k8s$
```

- Final result after deployment.



B. How to use CI/CD Android with Jenkins

- **Check status before update deployment application android**
 - Please open application on your mobile phone.



- **Developer create branch and update page mobile**

- Go to your code editor and update source on new branch for deploy

```
app > cicd-jenkins-android-tech > app > src > main > res > values > strings.xml
1  <resources>
2  <string name="app_name">Android App</string>
3  <string name="hello_world">Selamat Datang Dharma!</string>
4  </resources>
```

- Push branch to update on repository.

```

Klik Digital Sinergi@LAPTOP-60SCBK75 MINGW64 /d/Test Mandiri/app/cicd-jenkins-android-tech (main)
$ git branch
* main

Klik Digital Sinergi@LAPTOP-60SCBK75 MINGW64 /d/Test Mandiri/app/cicd-jenkins-android-tech (main)
$ git checkout -b fix/edit-string-value
Switched to a new branch 'fix/edit-string-value'

Klik Digital Sinergi@LAPTOP-60SCBK75 MINGW64 /d/Test Mandiri/app/cicd-jenkins-android-tech (fix/edit-string-value)
$ git add .

Klik Digital Sinergi@LAPTOP-60SCBK75 MINGW64 /d/Test Mandiri/app/cicd-jenkins-android-tech (fix/edit-string-value)
$ git status
On branch fix/edit-string-value
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   app/src/main/res/values/strings.xml

Klik Digital Sinergi@LAPTOP-60SCBK75 MINGW64 /d/Test Mandiri/app/cicd-jenkins-android-tech (fix/edit-string-value)
$ git commit -m "edit string value app android"
[fix/edit-string-value 92cb8c7] edit string value app android
1 file changed, 2 insertions(+), 2 deletions(-)

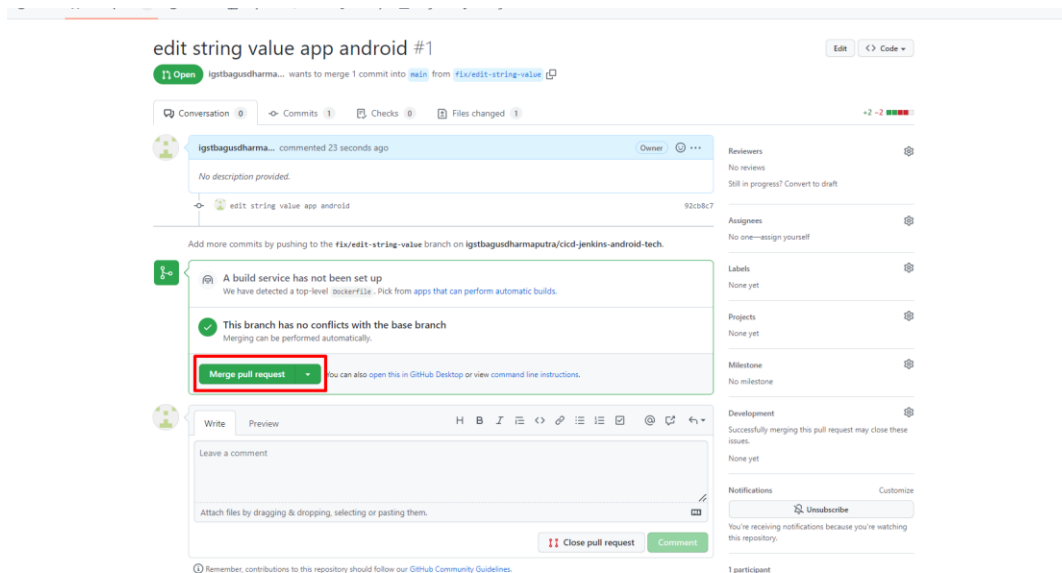
Klik Digital Sinergi@LAPTOP-60SCBK75 MINGW64 /d/Test Mandiri/app/cicd-jenkins-android-tech (fix/edit-string-value)
$ git push origin fix/edit-string-value
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (8/8), 668 bytes | 668.00 KiB/s, done.
Total 8 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'fix/edit-string-value' on GitHub by visiting:
remote:   https://github.com/igstbagusdharma Putra/cicd-jenkins-android-tech/pull/new/fix/edit-string-value
remote:
To https://github.com/igstbagusdharma Putra/cicd-jenkins-android-tech
 * [new branch]      fix/edit-string-value -> fix/edit-string-value

```

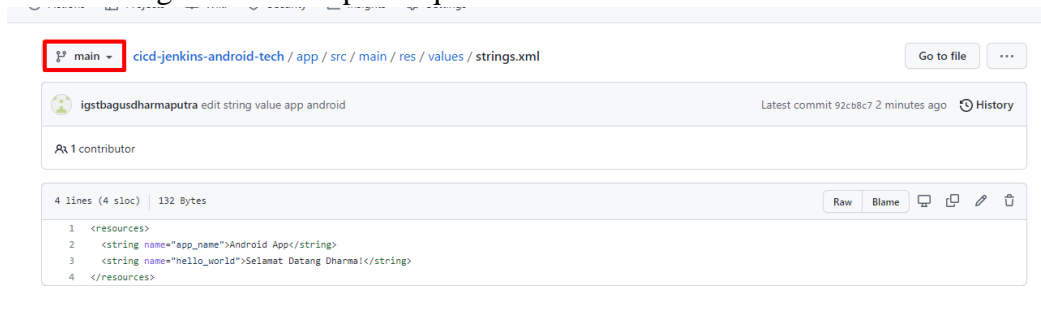
- Create pull request to update on branch **main** for deployment.

- Show changes code when pull request.

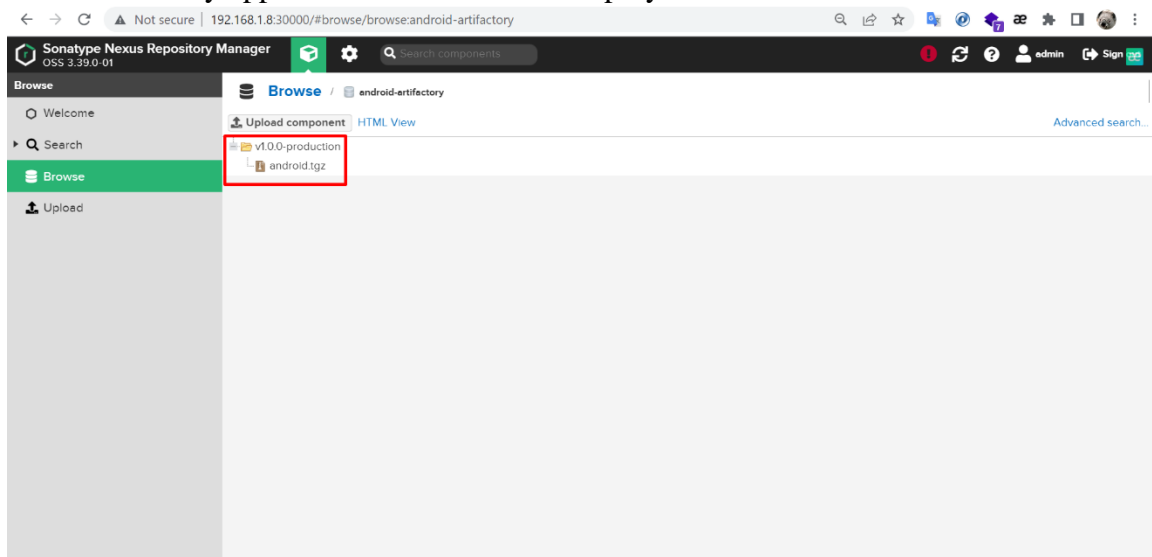
- Approve pull request to update on branch **main**.



- Check changes code after pull request on branch **main**.



- Show artifactory application android before deploy.



• Run Jenkins to Deploy Application Mobile

- Go to Jenkins and select job to use deployment application android.
- Input fields for running pipeline process.

Jenkins Search dharmaputra log out

Dashboard > CICD > cicc-application-android >

Pipeline cicc-application-android

This build requires parameters:

GITHUB_BRANCH
main
deploy to environment

SONARSCANNER
ON
scanning source code

TAG_VERSION
v1.0.1-production
environment staging : v1.0.0-staging , environment production : v1.0.0-production

Build

Build History trend

Filter builds...

- Show log scanning for check code on sonarqube.

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration

Search for projects...

cicc-application-android master May 30, 2022, 11:29 AM Version unspecified

Overview Issues Security Hotspots Measures Code Activity

QUALITY GATE STATUS

Passed
All conditions passed

MEASURES

New Code
Since May 29, 2022
Started 14 hours ago

Overall Code

0 Bugs Reliability **A**

0 Vulnerabilities Security **A**

2 Security Hotspots 0.0% Reviewed Security Review **E**

0 Debt 0 Code Smells Maintainability **A**

0.0% Coverage on 3 Lines to cover Unit Tests

0.0% Duplications on 36 Lines Duplicated Blocks

0

Type here to search

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration

Search for projects...

cicc-application-android master May 30, 2022, 11:29 AM Version unspecified

Overview Issues Security Hotspots Measures Code Activity

Search for files...

cicc-application-android app/src/main

	Lines of Code	Bugs	Vulnerabilities	Code Smells	Security Hotspots	Coverage	Duplications
app/src/main	36	0	0	0	2	0.0%	0.0%
java/com/sqsland/android/hello	10	0	0	0	0	0.0%	0.0%
res	11	0	0	0	0	—	0.0%
AndroidManifest.xml	15	0	0	0	2	—	0.0%

3 of 3 shown

SonarQube™ technology is powered by SonarSource SA
Community Edition - Version 9.4 (build 54424) - LGPL, v3 - Community - Documentation - Plugins - Web API

- Show log build to check process build success.

```

Stage Logs (Build Production)
[+] Checks if running on a Unix-like node (self time 75ms)
[+] Shell Script -- docker inspect -f. "${ID_TO_RUN}" (self time 613ms)
[+] Shell Script -- pwd (self time 650ms)
[+] Shell Script -- chmod +x ./gradlew (self time 657ms)
[+] Shell Script -- ./gradlew assembleDebug (self time 4min 20s)
  > Task :app:mergeDebugJavaResource UP-TO-DATE
  > Task :app:checkDebugDuplicateClasses UP-TO-DATE
  > Task :app:mergeExtDexDebug UP-TO-DATE
  > Task :app:transformClassesWithDexBuilderForDebug UP-TO-DATE
  > Task :app:mergeDexDebug UP-TO-DATE
  > Task :app:validateSigningDebug
  > Task :app:signingConfigWriterDebug
  > Task :app:mergeDebugJniLibFolders UP-TO-DATE
  > Task :app:mergeDebugNativeLibs UP-TO-DATE
  > Task :app:stripDebugDebugSymbols UP-TO-DATE
  > Task :app:packageDebug
  > Task :app:assembleDebug

Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.0.1/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 4m 18s
24 actionable tasks: 3 executed, 21 up-to-date

[+] Shell Script -- find -type f -name *${BUILD_FORMAT} -exec cp -rf {} result/app-debug.${BUILD_FORMAT} \; (self time 2s)

```

- Check log deploy to make sure artifact android store on nexus repository.

```

Stage Logs (Upload to Nexus Apk Production)
[+] Shell Script -- ls -l result (self time 781ms)
[+] Shell Script -- cd result && tar -C $(pwd) -czf ${SONAR_PROJECT_KEY_ANDROID}.tgz * (self time 795ms)
[+] Shell Script -- cd result && curl -X 'POST' 'http://192.168.1.8:30000/service/rest/v1/components?repository=${SONAR_PROJECT_KEY_ANDROID}-artifactory' -H 'accept: application/json' -H 'Content-Type: multipart/form-data' -H 'Authorization: Basic $(NEXUSCRED)' -F 'raw.directory=v1.0.1-production' -F 'raw.asset1=@${SONAR_PROJECT_KEY_ANDROID}.tgz;type=application/x-tgz' -F 'raw.asset1.filename=${SONAR_PROJECT_KEY_ANDROID}.tgz' (self time 3s)

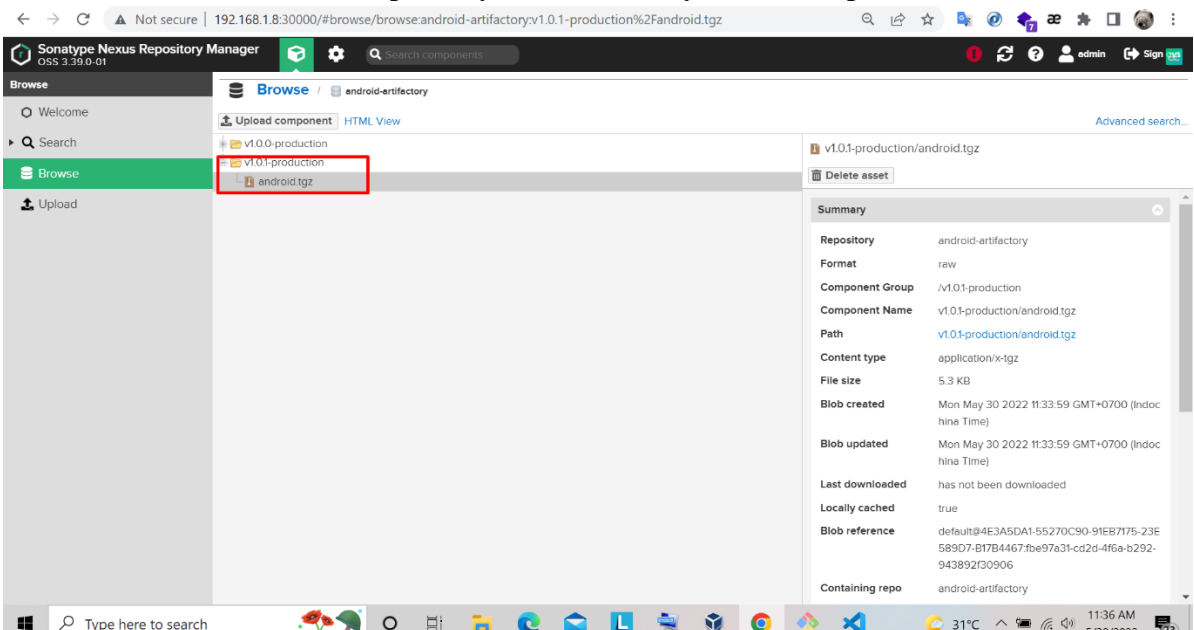
Warning: A secret was passed to "sh" using Groovy string interpolation, which is insecure.
Affected argument(s) used the following variable(s): [NEXUSCRED, SONAR_PROJECT_KEY_ANDROID]
See https://jenkins.io/redirect/groovy-string-interpolation for details.

+ cd result
+ curl -X POST http://192.168.1.8:30000/service/rest/v1/components?repository=****-artifactory -H accept: application/json -H Content-Type: multipart/form-data -H Authorization: Basic ****
-F raw.directory=v1.0.1-production -F raw.asset1=****.tgz;type=application/x-tgz -F raw.asset1.filename=****.tgz
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left  Speed

  0     0    0     0     0     0      0  0:00:00 --:--:-- --:--:--    0
100 5910    0     0 100 5910    0 4920 0:00:01 0:00:01 --:--:-- 4920
100 5910    0     0 100 5910    0 2662 0:00:02 0:00:02 --:--:-- 2663
100 5910    0     0 100 5910    0 2029 0:00:02 0:00:02 --:--:-- 2028
100 5910    0     0 100 5910    0 2010 0:00:02 0:00:02 --:--:-- 2010

```

- Download artifact on nexus repository and install on your mobile phone.



- Final result after install new mobile application

