

DEVOPS CI/CD PIPELINE

As steps given in the question the screenshots are placed accordingly

1.Implement a CI/CD pipeline using Jenkins multi-branch pipelines

N.Jithendra

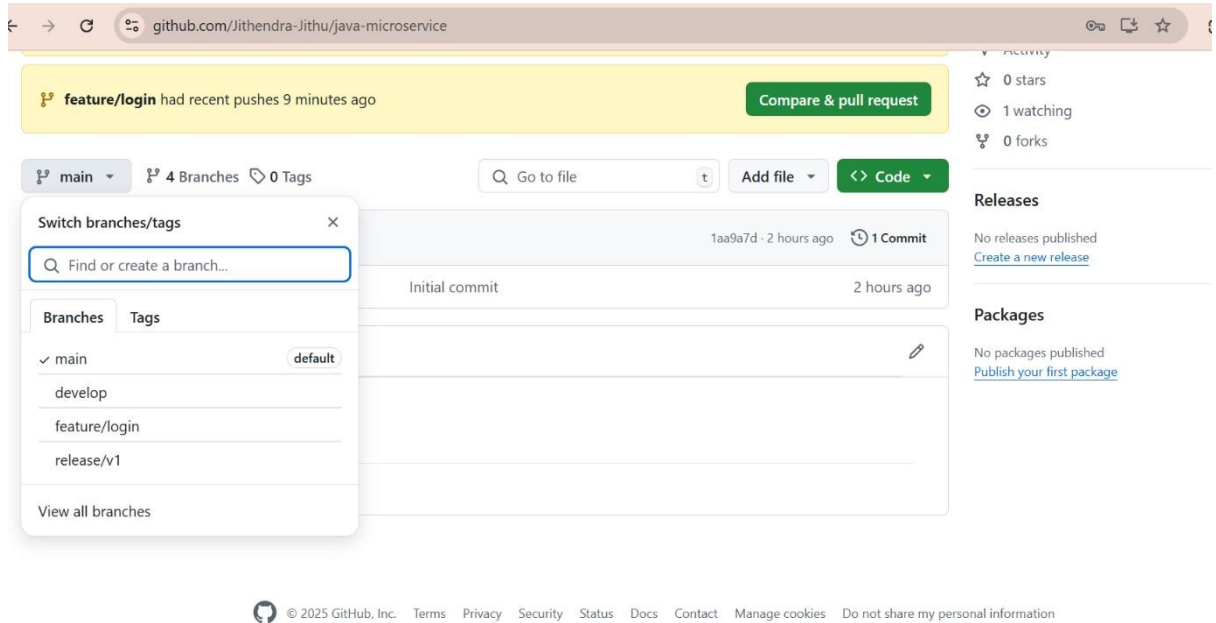
This github contains three branches:

289238

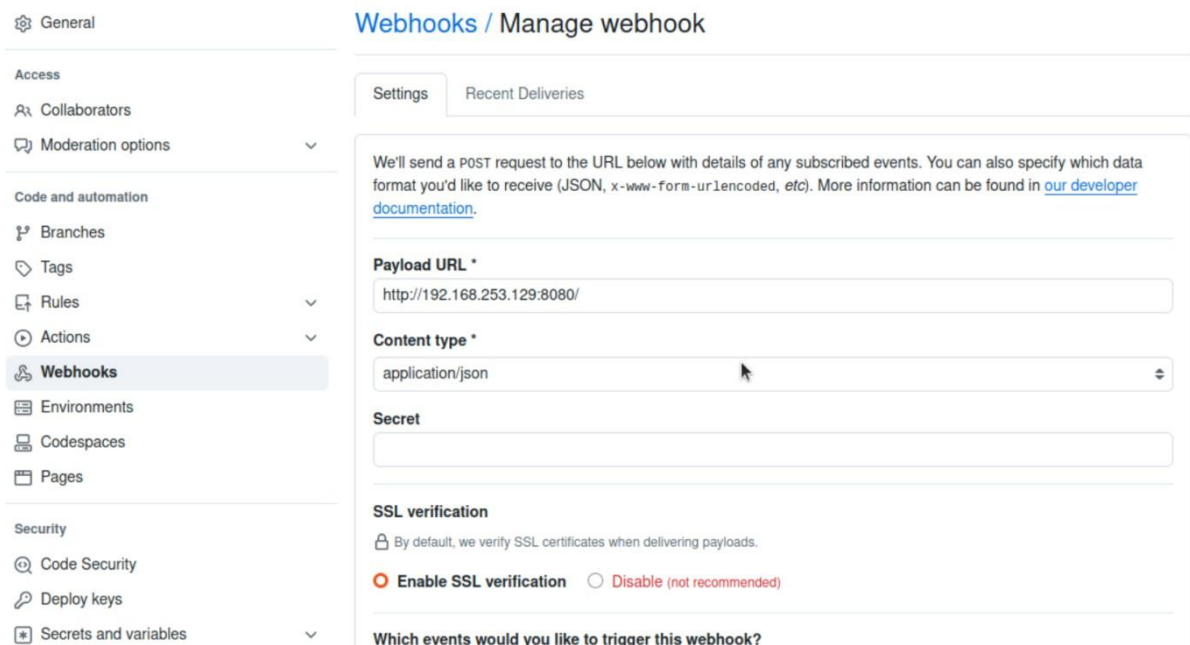
1.Develop

2.Feature/login

3.Release/v1



2.Triggers automatically using a GitHub webhook.



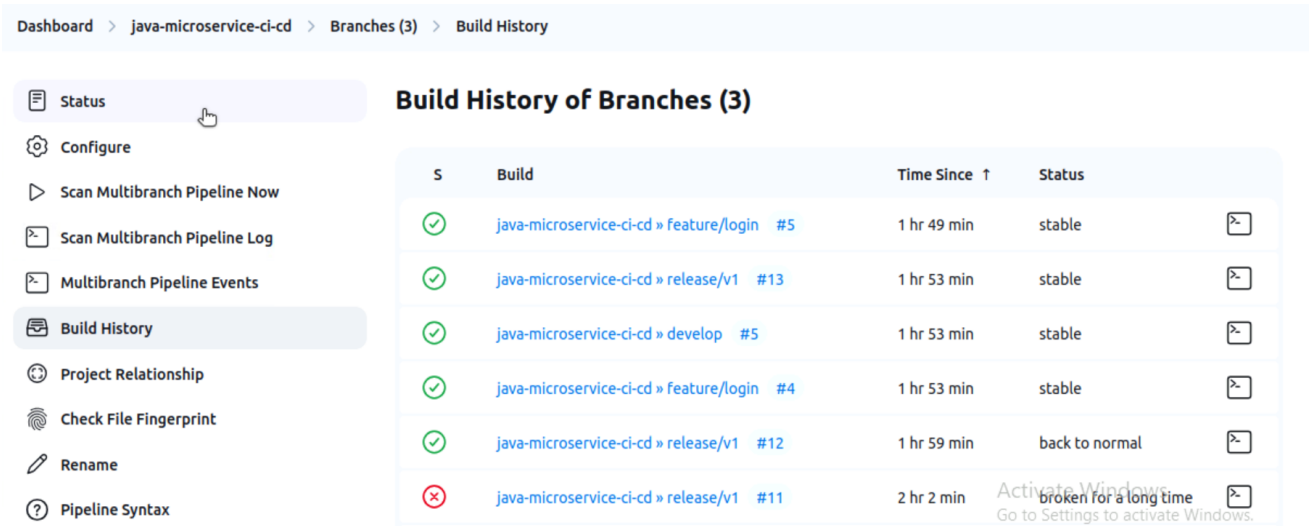
3. Builds and tests the Java application using maven.



The screenshot shows the Jenkinsfile for the 'java-microservice' project. The file is located at 'java-microservice / Jenkinsfile' and has 80 lines (72 loc) and 2.24 KB. The code defines a pipeline with four stages: 'Checkout', 'Build', 'Test', and 'Deploy'. The 'Build' stage uses 'mvn clean package' and the 'Test' stage uses 'mvn test'.

```
1  pipeline {
2      agent {
3          kubernetes {
4              namespace = 'default'
5              serviceAccount = 'jenkins'
6          }
7      }
8      stages {
9          stage('Checkout') {
10             steps {
11                 checkout scm
12             }
13         }
14         stage('Build') {
15             steps {
16                 sh 'mvn clean package'
17             }
18         }
19         stage('Test') {
20             steps {
21                 sh 'mvn test'
22             }
23         }
24         stage('Deploy') {
25             steps {
26                 sh 'mvn deploy'
27             }
28         }
29     }
30 }
```

Build history:



The screenshot shows the Jenkins Build History page for the 'java-microservice-ci-cd' pipeline. The page displays a table of build history for the 'feature/login' branch. The table has columns for 'S' (Status), 'Build', 'Time Since', and 'Status'. The builds are listed in descending order of time since the last build.

S	Build	Time Since	Status
✓	java-microservice-ci-cd » feature/login #5	1 hr 49 min	stable
✓	java-microservice-ci-cd » release/v1 #13	1 hr 53 min	stable
✓	java-microservice-ci-cd » develop #5	1 hr 53 min	stable
✓	java-microservice-ci-cd » feature/login #4	1 hr 53 min	stable
✓	java-microservice-ci-cd » release/v1 #12	1 hr 59 min	back to normal
✗	java-microservice-ci-cd » release/v1 #11	2 hr 2 min	broken for a long time

Activate Windows. Go to Settings to activate Windows.

Output of maven:

```
Dashboard > java-microservice-ci-cd > Branches (3) > develop > #5
[[1;34mINFO[m] [1m--- [0;32mmaven-compiler-plugin:3.1:testCompile[m [1m(default-testCompile)
[m @ [36mjava-microservice[0;1m ---[m
[[1;34mINFO[m] No sources to compile
[[1;34mINFO[m]
[[1;34mINFO[m] [1m--- [0;32mmaven-surefire-plugin:2.12.4:test[m [1m(default-test)[m @
[36mjava-microservice[0;1m ---[m
[[1;34mINFO[m] No tests to run.
[[1;34mINFO[m]
[[1;34mINFO[m] [1m--- [0;32mmaven-jar-plugin:2.4:jar[m [1m(default-jar)[m @ [36mjava-
microservice[0;1m ---[m
[[1;34mINFO[m] Building jar: /var/lib/jenkins/workspace/java-microservice-ci-cd_develop/
target/java-microservice-1.0-SNAPSHOT.jar
[[1;34mINFO[m] [1m-----[m
[[1;34mINFO[m] [1;32mBUILD SUCCESS[m
[[1;34mINFO[m] [1m-----[m
[[1;34mINFO[m] Total time: 4.987 s
[[1;34mINFO[m] Finished at: 2025-04-10T15:41:52+05:30
[[1;34mINFO[m] [1m-----[m
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Test)
[Pipeline] sh
+ mvn test
```

4. Packages the application into a Docker container and pushes it to an image registry.


```
master@master-vm:~$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
jithu145/java-microservice   develop            cfc25ecfba55       11 minutes ago     326MB
jithu145/java-microservice   release-v1        de30310dbe7e       11 minutes ago     326MB
jithu145/java-microservice   <none>            a811920edab4       18 minutes ago     326MB
jithu145/java-microservice   <none>            761f1de2637f       24 minutes ago     326MB
jithu145/java-microservice   <none>            e626c4202288       46 minutes ago     326MB
jithu145/java-microservice   <none>            50bc25066993       52 minutes ago     326MB
jithu145/java-microservice   <none>            27c6884f350a       About an hour ago   326MB
```

5. Docker Containerization and Image Push

The screenshot shows the Docker Hub interface for the user 'jithu145'. The left sidebar contains navigation links: Repositories (selected), Settings, Default privacy, Notifications, Billing, and Usage. The main area is titled 'Repositories' and shows 'All repositories within the jithu145 namespace.' Below this is a search bar and a dropdown menu set to 'All content'. A 'Create a repository' button is in the top right. A table lists the repositories:

Name	Last Pushed	Contains	Visibility	Scout
jithu145/java-microservice	11 minutes ago	IMAGE	Public	Inactive

Docker file:

 **Jithendra-Jithu** Initial commit for multi-branch Java microservice


Code **Blame** 3 lines (3 loc) • 93 Bytes


```
1 FROM openjdk:17-jdk-alpine
2 COPY target/*.jar app.jar
3 ENTRYPOINT ["java", "-jar", "/app.jar"]
```


And exposed to 8080


Kubernetes Deployment


- deployment.yaml (Defines the app deployment and replicas).
 - service.yaml (Exposes the application internally).
- b. Add Kubernetes apply command to Jenkinsfile to enable deployment.





▼  k8s


 deployment.yaml


 service.yaml

>  src

 Dockerfile

 Jenkinsfile

 pom.xml

 sonar-project.properties

Code **Blame** 19 lines (19 loc) • 360 Bytes

```
1 apiVersion: apps/v1
2 kind: Deployment
3 metadata:
4   name: java-microservice
5 spec:
6   replicas: 1
7   selector:
8     matchLabels:
9       app: java-microservice
10  template:
11    metadata:
12      labels:
13        app: java-microservice
14  spec:
15    containers:
16    - name: java-microservice
17      image: IMAGE_PLACEHOLDER
18      ports:
19    - containerPort: 8080
```

[java-microservice](#) / [k8s](#) / [service.yaml](#) 



Jithendra-Jithu Initial commit for multi-branch Java microservice

Code

Blame

12 lines (12 loc) • 194 Bytes

```
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: java-microservice
5  spec:
6    selector:
7      app: java-microservice
8    ports:
9      - protocol: TCP
10      port: 80
11      targetPort: 8080
12  type: NodePort
```

Jenkins file that is written in the code:

```
1  pipeline {
2      agent any
3      environment {
4          IMAGE = "docker.io/jithu145/java-microservice:${env.BRANCH_NAME.replaceAll('/', '-')}}"
5          KUBECONFIG_PATH = "/var/lib/jenkins/.kube/config"
6      }
7
8
9      stages {
10         stage('Checkout') {
11             steps {
12                 checkout scm
13             }
14         }
15
16         stage('Build') {
17             steps {
18                 sh 'mvn clean package'
19             }
20         }
21
22         stage('Test') {
23             steps {
24                 sh 'mvn test'
25             }
26         }
27     }
28
29     stage('Docker Build & Push') {
30         when {
31             anyOf {
32                 branch "develop"
33                 branch pattern: "release/*.+", comparator: "REGEXP"
34                 branch "main"
35             }
36         }
37         steps {
38             withCredentials([usernamePassword(credentialsId: 'fb16b1ba-d2e9-41bb-8654-d00d3b5b61e6', usernameVariable:
39                 sh '''
40                     docker build -t ${IMAGE} .
41                     echo "${PASS}" | docker login -u "${USER}" --password-stdin
42                     docker push ${IMAGE}
43                 ''')
44             ]
45         }
46     }
47 }
```

```

stage('Deploy to Staging') {
    when {
        branch pattern: "release/.*", comparator: "REGEXP"
    }
    steps {
        sh '''
            sed -i 's|IMAGE_PLACEHOLDER|${IMAGE}|' k8s/deployment.yaml
            kubectl --kubeconfig=${KUBECONFIG_PATH} apply -f k8s/deployment.yaml
            kubectl --kubeconfig=${KUBECONFIG_PATH} apply -f k8s/service.yaml
            ...
        '''
    }
}

```

```

stage('Approval & Deploy to Prod') {
    when {
        branch 'main'
    }
    steps {
        input message: "Deploy to Production?"
        sh '''
            sed -i 's|IMAGE_PLACEHOLDER|${IMAGE}|' k8s/deployment.yaml
            kubectl --kubeconfig=${KUBECONFIG_PATH} apply -f k8s/deployment.yaml
            kubectl --kubeconfig=${KUBECONFIG_PATH} apply -f k8s/service.yaml
            ...
        '''
    }
}

```

After executing the Jenkins multibranch pipeline:

Output:

Total pipeline output:

The screenshot displays the Jenkins web interface for a multibranch pipeline named 'java-microservice-ci-cd'. The top navigation bar includes the Jenkins logo, a search icon, a notification bell with 3 alerts, a shield icon with 2 warnings, the user 'admin', and a 'log out' button. The breadcrumb trail shows 'Dashboard > java-microservice-ci-cd > Scan Multibranch Pipeline Now'. The left sidebar contains a list of actions: Status, Configure, Scan Multibranch Pipeline Now (highlighted), Scan Multibranch Pipeline Log, Multibranch Pipeline Events, Delete Multibranch Pipeline, Build History, Project Relationship, Check File Fingerprint, Rename, and Pipeline Syntax. The main content area shows the 'Branches (3)' section with a table of pipeline runs.

S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☁	develop	56 min #4	57 min #3	1 min 33 sec
✓	☁	feature/login	1 hr 6 min #3	1 hr 22 min #1	1 min 2 sec
✓	☁	release/v1	6 min 2 sec #12	8 min 50 sec #11	1 min 4 sec

Below the table, there is a link to 'Scan Multibranch Pipeline Log' and a 'View as plain text' option. The log content is displayed in a code block, showing the execution of git commands to index branches and fetch updates. At the bottom, a summary bar for the 'develop' branch shows a duration of '6 min 12 sec', a status of 'log', and a '2 sec' duration for the current run.

```
Started by user admin
[Thu Apr 10 15:41:01 IST 2025] Starting branch indexing...
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git ls-remote --symref -- https://github.com/Jithendra-Jithu/java-microservice.git #
timeout=10
> git rev-parse --resolve-git-dir /var/lib/jenkins/caches/
git-3a14c6c2a63d382eee485c46a4508359/.git # timeout=10
Setting origin to https://github.com/Jithendra-Jithu/java-microservice.git
> git config remote.origin.url https://github.com/Jithendra-Jithu/java-microservice.git #
timeout=10
Fetching & pruning origin...
Listing remote references...
> git config --get remote.origin.url # timeout=10
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git ls-remote -h -- https://github.com/Jithendra-Jithu/java-microservice.git # timeout=10
Fetching upstream changes from origin
```


Develop branch:

develop 4 Branches 0 Tags

Go to file

Add file

Code

This branch is 8 commits ahead of, 1 commit behind main.

Jithendra-Jithu Update Jenkinsfile 35fc792 · 10 minutes ago 8 Commits

k8s	Initial commit for multi-branch Java microservice	2 hours ago
src/main/java/com/example	Initial commit for multi-branch Java microservice	2 hours ago
Dockerfile	Initial commit for multi-branch Java microservice	2 hours ago
Jenkinsfile	Update Jenkinsfile	10 minutes ago
pom.xml	Initial commit for multi-branch Java microservice	2 hours ago

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Output:

Dashboard > java-microservice-ci-cd > develop > #5

</> Changes

Console Output

Edit Build Information

Delete build '#5'

Timings

Git Build Data

Git Build Data

Pipeline Overview

Pipeline Console

Restart from Stage

Replay

Pipeline Steps

Workspaces

Branch indexing
> git rev-parse --resolve-git-dir /var/lib/jenkins/caches/git-3a14c6c2a63d382eee485c46a4508359/.git # timeout=10
Setting origin to https://github.com/Jithendra-Jithu/java-microservice.git
> git config remote.origin.url https://github.com/Jithendra-Jithu/java-microservice.git # timeout=10
Fetching origin...
Fetching upstream changes from origin
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git config --get remote.origin.url # timeout=10
> git fetch --tags --force --progress -- origin +refs/heads/*:refs/remotes/origin/* # timeout=10
Seen branch in repository origin/develop
Seen branch in repository origin/feature/login
Seen branch in repository origin/main
Seen branch in repository origin/release/v1
Seen 4 remote branches
Obtained Jenkinsfile from 35fc792ff94490122c8e2bec4800b005abb85361
Loading library jenkins-shared-library@main
Attempting to resolve main from remote references...
> git --version # timeout=10

Release branch:

release/v1

4 Branches

0 Tags

Go to file

Add file

Code

This branch is 12 commits ahead of, 1 commit behind main.

Contribute

Jithendra-Jithu Update Jenkinsfile

de4f8ff · 16 minutes ago

12 Commits

k8s	Initial commit for multi-branch Java microservice	2 hours ago
src/main/java/com/example	Initial commit for multi-branch Java microservice	2 hours ago
Dockerfile	Initial commit for multi-branch Java microservice	2 hours ago
Jenkinsfile	Update Jenkinsfile	16 minutes ago
pom.xml	Initial commit for multi-branch Java microservice	2 hours ago

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Output:

Dashboard > java-microservice-ci-cd > release/v1 > #13

Status

Changes

Console Output

Edit Build Information

Delete build '#13'

Timings

Git Build Data

Git Build Data

Pipeline Overview

Pipeline Console

Restart from Stage

Build

Console Output

Download

Copy

View as plain text

Branch indexing
> git rev-parse --resolve-git-dir /var/lib/jenkins/caches/
git-3a14c6c2a63d382eee485c46a4508359/.git # timeout=10
Setting origin to https://github.com/Jithendra-Jithu/java-microservice.git
> git config remote.origin.url https://github.com/Jithendra-Jithu/java-microservice.git #
timeout=10
Fetching origin...
Fetching upstream changes from origin
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git config --get remote.origin.url # timeout=10
> git fetch --tags --force --progress -- origin +refs/heads/*:refs/remotes/origin/* #
timeout=10
Seen branch in repository origin/develop
Seen branch in repository origin/feature/login
Seen branch in repository origin/main
Seen branch in repository origin/release/v1
Seen 4 remote branches

Activate Windows
Go to Settings to activate Windows.

Feature/login:

feature/login

4 Branches 0 Tags

Go to file

Add file

<> Code

This branch is 6 commits ahead of, 1 commit behind main.

Contribute

Jithendra-Jithu Update Jenkinsfile

c5e4f5c · 40 minutes ago 6 Commits

k8s	Initial commit for multi-branch Java microservice	2 hours ago
src/main/java/com/example	Initial commit for multi-branch Java microservice	2 hours ago
Dockerfile	Initial commit for multi-branch Java microservice	2 hours ago
Jenkinsfile	Update Jenkinsfile	40 minutes ago
pom.xml	Initial commit for multi-branch Java microservice	2 hours ago
sonar-project.properties	Initial commit for multi-branch Java microservice	2 hours ago

README

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Output:

Dashboard > java-microservice-ci-cd > feature/login > #5

Status

</> Changes

Console Output

Edit Build Information

Delete build '#5'

Timings

Git Build Data

Git Build Data

Pipeline Overview

Pipeline Console

Restart from Stage

Console Output

Download Copy View as plain text

```
Started by user admin
> git rev-parse --resolve-git-dir /var/lib/jenkins/caches/
git-3a14c6c2a63d382eee485c46a4508359/.git # timeout=10
Setting origin to https://github.com/Jithendra-Jithu/java-microservice.git
> git config remote.origin.url https://github.com/Jithendra-Jithu/java-microservice.git #
timeout=10
Fetching origin...
Fetching upstream changes from origin
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git config --get remote.origin.url # timeout=10
> git fetch --tags --force --progress -- origin +refs/heads/*:refs/remotes/origin/* #
timeout=10
Seen branch in repository origin/develop
Seen branch in repository origin/feature/login
Seen branch in repository origin/main
Seen branch in repository origin/release/v1
```

Activate Windows
Go to Settings to activate Windows.

localhost:8080/job/java-microservice-ci-cd/job/feature%2Flogin/5/pipeline-console