

## GitLab-CI/CD

### 1. Implement a Sample GitLab Pipeline with Artifacts

#### YAML script

stages:

- build
- test
- deploy

# build stage

build-job:

stage: build

script:

- echo "Building application"
- mkdir build
- echo "This is a build artifact file" > build/test.txt
- echo "Build completed"

artifacts:

paths:

- build/

# test stage

test-job:

stage: test

script:

- echo "Running test stage"
- cat build/test.txt || echo "artifact not found!!!"
- echo "Test passed successfully"

# deploy stage

deploy-job:

stage: deploy

script:

- echo "Deploying application"
- echo "Content:"
- cat build/test.txt
- echo "Deploy completed"

## Output

### GitLab build pipeline

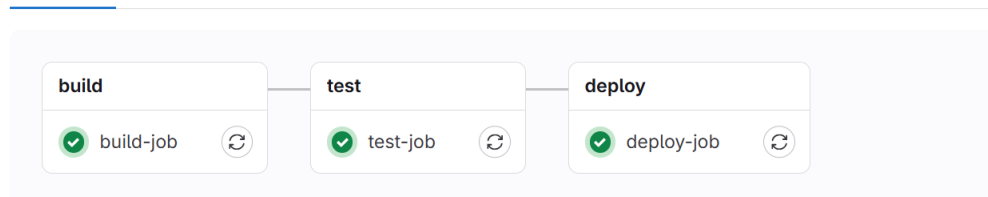
## create a artifact application

✓ Passed Vishnu Raj created pipeline for commit 81bc7b0d 3 minutes ago, finished 1 minute ago

For main

latest branch 3 jobs 1.71 1 minute 42 seconds, queued for 3 seconds

Pipeline Jobs 3 Tests 0



### build stage

Search visible log output

```
18 Created fresh repository.
19 Checking out 26881141 as detached HEAD (ref is main)...
20 Skipping Git submodules setup
21 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
22 Executing "step_script" stage of the job script
23 Using default image
24 Using effective pull policy of [always] for container ruby:3.1
25 Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 for ruby:3.1 with digest sha256:91627f55e896900a6b67d15c92fb930500ff73948803da1330b8a853f
  ecebb5 ...
26 $ echo "Building application"
27 Building application
28 $ mkdir build
29 $ echo "This is a build artifact file" > build/test.txt
30 $ echo "Build completed"
31 Build completed
32 Uploading artifacts for successful job
33 Uploading artifacts...
34 build/: found 2 matching artifact files and directories
35 Uploading artifacts as "archive" to coordinator... 201 Created correlation_id=5c7061f2c80956
  d2b818de4cc0eb1084 id=11052793466 responseStatus=201 Created token=eyJraWQiO
36 Cleaning up project directory and file based variables
37 Job succeeded
```

Duration: 33 seconds  
Finished: 14 hours ago  
Queued: 0 seconds  
Timeout: 1h (from project)  
Runner: #12270852 (Jhc\_Jxvh) 3-green.saas-linux-small-amd64.runners-manager.gitlab.com/default  
Source: Push

Job artifacts

These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.

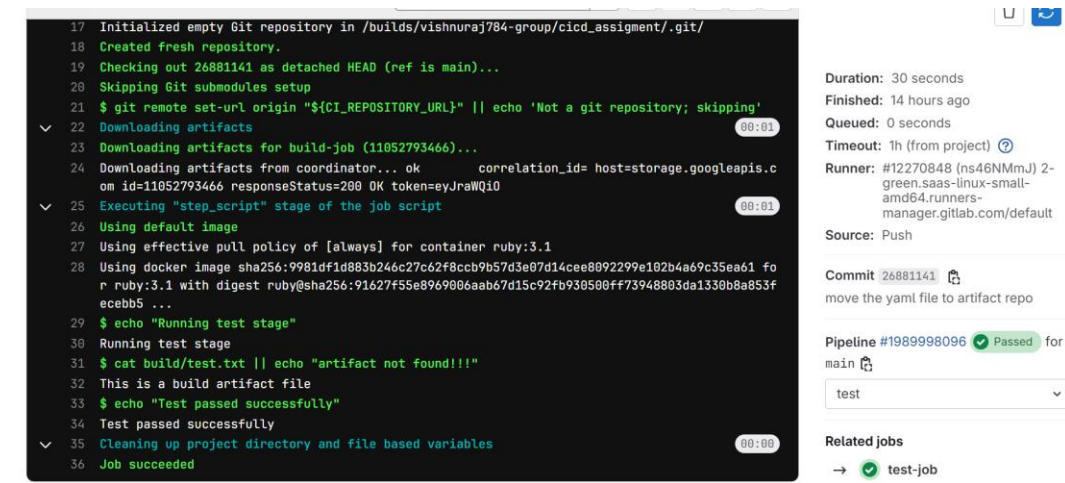
Keep Download Browse

Commit 26881141





move the yaml file to artifact repo

Pipeline #1989998096 ✓ Passed for

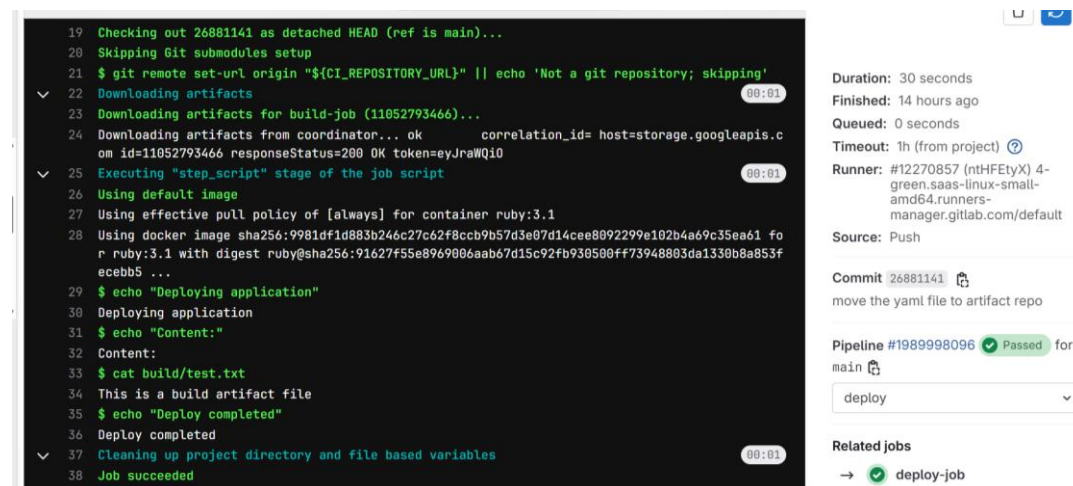
## test stage







```
17 Initialized empty Git repository in /builds/vishnuraj784-group/cicd_assignment/.git/
18 Created fresh repository.
19 Checking out 26881141 as detached HEAD (ref is main)...
20 Skipping Git submodules setup
21 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
22 Downloading artifacts
23 Downloading artifacts for build-job (11052793466)...
24 Downloading artifacts from coordinator... ok correlation_id= host=storage.googleapis.c
om id=11052793466 responseStatus=200 OK token=eyJraWQiO
25 Executing "step_script" stage of the job script
26 Using default image
27 Using effective pull policy of [always] for container ruby:3.1
28 Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 fo
r ruby:3.1 with digest ruby@sha256:91627f55e8969086aab67d15c92fb930500ff73948803da1330b8a853f
eceb5 ...
29 $ echo "Running test stage"
30 Running test stage
31 $ cat build/test.txt || echo "artifact not found!!!"
32 This is a build artifact file
33 $ echo "Test passed successfully"
34 Test passed successfully
35 Cleaning up project directory and file based variables
36 Job succeeded
```

Duration: 30 seconds  
Finished: 14 hours ago  
Queued: 0 seconds  
Timeout: 1h (from project) ⓘ  
Runner: #12270848 (ns46NMmJ) 2-green.saas-linux-small-amd64.runners-manager.gitlab.com/default  
Source: Push  
Commit 26881141  move the yaml file to artifact repo  
Pipeline #1989998096  Passed for main   
test  
Related jobs  
→  test-job

## deploy stage



```
19 Checking out 26881141 as detached HEAD (ref is main)...
20 Skipping Git submodules setup
21 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
22 Downloading artifacts
23 Downloading artifacts for build-job (11052793466)...
24 Downloading artifacts from coordinator... ok correlation_id= host=storage.googleapis.c
om id=11052793466 responseStatus=200 OK token=eyJraWQiO
25 Executing "step_script" stage of the job script
26 Using default image
27 Using effective pull policy of [always] for container ruby:3.1
28 Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 fo
r ruby:3.1 with digest ruby@sha256:91627f55e8969086aab67d15c92fb930500ff73948803da1330b8a853f
eceb5 ...
29 $ echo "Deploying application"
30 Deploying application
31 $ echo "Content:"
32 Content:
33 $ cat build/test.txt
34 This is a build artifact file
35 $ echo "Deploy completed"
36 Deploy completed
37 Cleaning up project directory and file based variables
38 Job succeeded
```

Duration: 30 seconds  
Finished: 14 hours ago  
Queued: 0 seconds  
Timeout: 1h (from project) ⓘ  
Runner: #12270857 (nHFtYX) 4-green.saas-linux-small-amd64.runners-manager.gitlab.com/default  
Source: Push  
Commit 26881141  move the yaml file to artifact repo  
Pipeline #1989998096  Passed for main   
deploy  
Related jobs  
→  deploy-job

## 2. Implement Maven Build Job in GitLab CI/CD

### YAML script

#yaml file for java application

stages:

- build
- deploy

#docker image

image: maven:latest

#build job stage

build-job:

stage: build

script:

- echo "Building the application"
- mvn clean package
- echo "Build stage completed"

artifacts:

paths:

- target/

#deploy job stage

deploy-job:

stage: deploy

script:

- echo "Deploy stage in-progress"
- ls target
- java -cp target/sum-app-1.0-SNAPSHOT.jar com.example.App 5 7
- echo "Completed the application"

## Output

### GitLab build pipeline

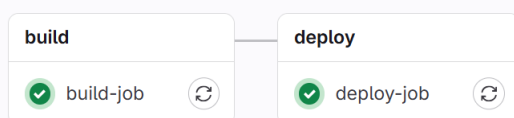
#### change the java application

✓ Passed Vishnu Raj created pipeline for commit c144389e 9 minutes ago, finished 8 minutes ago

For main

latest branch 2 jobs 0.95 56 seconds, queued for 2 seconds

Pipeline Jobs 2 Tests 0



## build stage

```
3/3.14.0/commons-lang3-3.14.0.jar (658 kB at 18 MB/s)
483 Downloaded from central: https://repo.maven.apache.org/maven2/org/tukaani/xz/1.9/xz-1.9.jar
(116 kB at 1.6 MB/s)
484 Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-compress/1.26.1/commons-compress-1.26.1.jar (1.1 MB at 13 MB/s)
485 Downloaded from central: https://repo.maven.apache.org/maven2/com/github/luben/zstd-jni/1.5.5-11/zstd-jni-1.5.5-11.jar (6.8 MB at 38 MB/s)
486 [INFO] Building jar: /builds/vishnuraj784-group/cicdmaven/target/sum-app-1.0-SNAPSHOT.jar
487 [INFO] -----
488 [INFO] BUILD SUCCESS
489 [INFO] -----
490 [INFO] Total time: 6.915 s
491 [INFO] Finished at: 2025-08-19T09:06:01Z
492 [INFO] -----
493 $ echo "Build stage completed"
494 Build stage completed
495 Uploading artifacts for successful job
496 Uploading artifacts...
497 target/: found 29 matching artifact files and directories
498 Uploading artifacts as "archive" to coordinator... 201 Created correlation_id=ee4a295424e7bd
1b9c320db03a55b11b id=11059632240 responseStatus=201 Created token=eyJnaWQ1O
499 Cleaning up project directory and file based variables
500 Job succeeded
```

Duration: 32 seconds  
Finished: 9 minutes ago  
Queued: 0 seconds  
Timeout: 1h (from project) [?](#)  
Runner: #12270845 (JLgUopmM) 1-green.saas-linux-small-amd64.runners-manager.gitlab.com/default  
Source: Push

**Job artifacts** [?](#)  
These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.

[Keep](#) [Download](#) [Browse](#)

Commit [c144389e](#) [?](#)  
change the java application

Pipeline #1991081314 [?](#) Passed for m

## deploy stage

```
400 id=11059632240 responseStatus=200 OK token=eyJnaWQ1O
23 Executing "step_script" stage of the job script
24 Using effective pull policy of [always] for container maven:latest
25 Using docker image sha256:9cf0d19b18091e0daa06f91ef426d44cba5586c75d48f97d882054e416a0839 fo
r maven:latest with digest maven:sha256:fe1de86d91ca06519b58fc85ff737e159c18c0d0a8c7f4c66601b
74dba136c5c ...
26 $ echo "Deploy stage in-progress"
27 Deploy stage in-progress
28 $ ls target
29 classes
30 generated-sources
31 generated-test-sources
32 maven-archiver
33 maven-status
34 sum-app-1.0-SNAPSHOT.jar
35 surefire-reports
36 test-classes
37 $ java -cp target/sum-app-1.0-SNAPSHOT.jar com.example.App 5 7
38 Sum: 12
39 $ echo "Completed the application"
40 Completed the application
41 Cleaning up project directory and file based variables
42 Job succeeded
```

Duration: 23 seconds  
Finished: 9 minutes ago  
Queued: 0 seconds  
Timeout: 1h (from project) [?](#)  
Runner: #32976645 (YKxHNyex) 6-green.saas-linux-small-amd64.runners-manager.gitlab.com/default  
Source: Push

Commit [c144389e](#) [?](#)  
change the java application

Pipeline #1991081314 [?](#) Passed for m  
[ain](#) [?](#)

deploy

**Related jobs**  
→ [deploy-job](#)

## 3. Implement a Sample GitLab Pipeline with environment

### YAML script

stages:

- build
- test
- deploy

#build job

build-job:

stage: build

script:

- echo "Compiling code stage"
- echo "Build stage successfully"

#test job

test-job:

stage: test

script:

- echo "Running tests stage"
- echo "Test stage successfully"

# deploy stage for version/1.1.0

deploy-job:

stage: deploy

script:

- echo "Deploying application stage"
- echo "application deployed by \$first\_name"
- echo "Deploy successfully"

environment:

name: staging

only:

- version/1.1.0

# deploy stage for production

deploy-prod:

stage: deploy

script:

- echo "Deploying application stage"
- echo "application deployed by \$second\_name"
- echo "Deploy successfully"

environment:

name: production

only:

- main

## Output

### GitLab pipeline

This pipeline is run on version/1.1.0

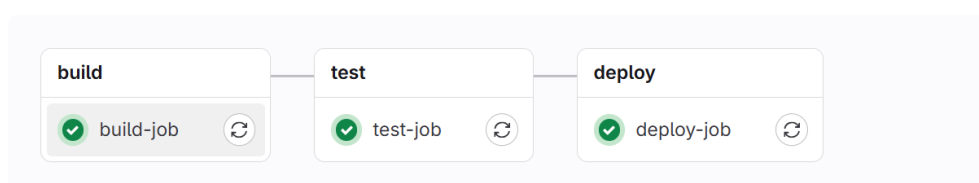
## deployment on version/1.1.0

✓ Passed Vishnu Raj created pipeline for commit 96411878 6 minutes ago, finished 4 minutes ago

For [version/1.1.0](#)

latest branch 3 jobs 1.66 1 minute 39 seconds, queued for 2 seconds

Pipeline Jobs 3 Tests 0



### Build stage

Search visible log output

```
08751a44ec3e56/cfcb6078c2d283c5b
13 Running on runner-ykxhnyexq-project-73186302-concurrent-0 via runner-ykxhnyexq-s-l-s-amd64-17
55686526-d3fd0c2...
14 Getting source from Git repository 00:01
15 Gitlaly correlation ID: 616a4486a345540746c70b0c9d6a1b6e
16 Fetching changes with git depth set to 20...
17 Initialized empty Git repository in /builds/vishnuraj784-group/maven_sample/.git/
18 Created fresh repository.
19 Checking out 96411878 as detached HEAD (ref is version/1.1.0)...
20 Skipping Git submodules setup
21 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
22 Executing "step_script" stage of the job script 00:01
23 Using default image
24 Using effective pull policy of [always] for container ruby:3.1
25 Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35aa61 fo
r ruby:3.1 with digest ruby@sha256:91627f55e8969086aab67d15c92fb930500ff73948803da1330b8a853f
e4ebb5 ...
26 $ echo "Compiling code stage"
27 Compiling code stage
28 $ echo "Build stage successfully"
29 Build stage successfully
30 Cleaning up project directory and file based variables 00:00
31 Job succeeded
```

Duration: 37 seconds

Finished: 6 minutes ago

Queued: 0 seconds

Timeout: 1h (from project)

Runner: #32976645 (YKxHnyex) 6-green.saas-linux-small-amd64.runners-manager.gitlab.com/default

Source: Push

Commit 96411878 deployment on version/1.1.0

Pipeline #1993508212 ✓ Passed for version/1.1.0

build

Related jobs

→ build-job

## Test stage

This screenshot shows a GitHub Actions workflow in the 'Test stage'. The log output on the left details the steps: getting source from Git repository, fetching changes, initializing a fresh repository, checking out the code, and running tests. The right sidebar provides metadata: Duration (31 seconds), Finished (6 minutes ago), and a 'Passed' status for the pipeline.

```
13 Running on runner-xs6vzpvq-project-73186382-concurrent-8 via runner-xs6vzpvq-s-l-s-amd64-17
14 Getting source from Git repository
15 Gitly correlation ID: 8aefff4f1bfd9a2b22f28e63736360a3
16 Fetching changes with git depth set to 20...
17 Initialized empty Git repository in /builds/vishnuraj784-group/maven_sample/.git/
18 Created fresh repository.
19 Checking out 96411878 as detached HEAD (ref is version/1.1.0)...
20 Skipping Git submodules setup
21 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
22 Executing "step_script" stage of the job script
23 Using default image
24 Using effective pull policy of [always] for container ruby:3.1
25 Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 for
  ruby:3.1 with digest ruby@sha256:91627f55e8969086aab67d15c92fb930508ff73948803da1330b8a853f
  ecebb5 ...
26 $ echo "Running tests stage"
27 Running tests stage
28 $ echo "Test stage successfully"
29 Test stage successfully
30 Cleaning up project directory and file based variables
31 Job succeeded
```

Duration: 31 seconds  
Finished: 6 minutes ago  
Queued: 0 seconds  
Timeout: 1h (from project)  
Runner: #12270859 (xs6Vzpvq) 5-green.saas-linux-small-amd64.runners-manager.gitlab.com/default  
Source: Push  
Commit 96411878 deployment on version/1.1.0  
Pipeline #1993508212 **Passed** for version/1.1.0  
test  
Related jobs  
→ test-job

## Deploy stage on staging environment

This screenshot shows a GitHub Actions workflow in the 'Deploy stage on staging environment'. The log output details the steps: getting source from Git repository, fetching changes, initializing a fresh repository, checking out the code, and deploying the application to the staging environment. The right sidebar shows the pipeline status as 'Passed'.

```
14 Getting source from Git repository
15 Gitly correlation ID: 6160588a17beb242c295e6f5035c9ab7
16 Fetching changes with git depth set to 20...
17 Initialized empty Git repository in /builds/vishnuraj784-group/maven_sample/.git/
18 Created fresh repository.
19 Checking out 96411878 as detached HEAD (ref is version/1.1.0)...
20 Skipping Git submodules setup
21 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
22 Executing "step_script" stage of the job script
23 Using default image
24 Using effective pull policy of [always] for container ruby:3.1
25 Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 for
  ruby:3.1 with digest ruby@sha256:91627f55e8969086aab67d15c92fb930508ff73948803da1330b8a853f
  ecebb5 ...
26 $ echo "Deploying application stage"
27 Deploying application stage
28 $ echo "application deployed by $first_name"
29 application deployed by Bobby bob
30 $ echo "Deploy successfully"
31 Deploy successfully
32 Cleaning up project directory and file based variables
33 Job succeeded
```

Duration: 30 seconds  
Finished: 7 minutes ago  
Queued: 0 seconds  
Timeout: 1h (from project)  
Runner: #32976645 (YKxHMyex) 6-green.saas-linux-small-amd64.runners-manager.gitlab.com/default  
Source: Push  
Commit 96411878 deployment on version/1.1.0  
Pipeline #1993508212 **Passed** for version/1.1.0  
deploy  
Related jobs  
→ deploy-job

## Deploy stage on production environment

This screenshot shows a GitHub Actions workflow in the 'Deploy stage on production environment'. The log output details the steps: getting source from Git repository, fetching changes, initializing a fresh repository, checking out the code, and deploying the application to the production environment. The right sidebar shows the pipeline status as 'Passed'.

```
14 Getting source from Git repository
15 Gitly correlation ID: 72024b41abc32611601ba5b553bd0fd4
16 Fetching changes with git depth set to 20...
17 Initialized empty Git repository in /builds/vishnuraj784-group/maven_sample/.git/
18 Created fresh repository.
19 Checking out 18b27efc as detached HEAD (ref is main)...
20 Skipping Git submodules setup
21 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
22 Executing "step_script" stage of the job script
23 Using default image
24 Using effective pull policy of [always] for container ruby:3.1
25 Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 for
  ruby:3.1 with digest ruby@sha256:91627f55e8969086aab67d15c92fb930508ff73948803da1330b8a853f
  ecebb5 ...
26 $ echo "Deploying application stage"
27 Deploying application stage
28 $ echo "application deployed by $second_name"
29 application deployed by aleen aleen
30 $ echo "Deploy successfully"
31 Deploy successfully
32 Cleaning up project directory and file based variables
33 Job succeeded
```

Duration: 30 seconds  
Finished: 16 minutes ago  
Queued: 0 seconds  
Timeout: 1h (from project)  
Runner: #12270848 (ns46NMmJ) 2-green.saas-linux-small-amd64.runners-manager.gitlab.com/default  
Source: Web  
Commit 18b27efc application on main branch  
Pipeline #1993489663 **Passed** for main  
deploy  
Related jobs  
→ deploy-prod



#### **4. Implement a Sample GitLab Pipeline with extends**

##### YAML script

#yaml file for extends

stages:

- build
- test
- deploy

#template for all job

.job-template:

image: maven:latest

before\_script:

- echo "Preparing environment..."

#build job

build-job:

stage: build

extends: .job-template

script:

- echo "Building stage"
- mkdir build
- echo "This application is build for extends file" > build/app.txt

artifacts:

paths:

- build/

#test job

test-job:

stage: test

extends: .job-template

script:

- echo "Testing stage"
- echo "Running test"
- cat build/app.txt
- echo "All test passed!"

#deploy job

deploy-job:

stage: deploy

extends: .job-template

script:

- echo "Deploying stage"
- echo "Deployment completed"

## Output

### GitLab main branch

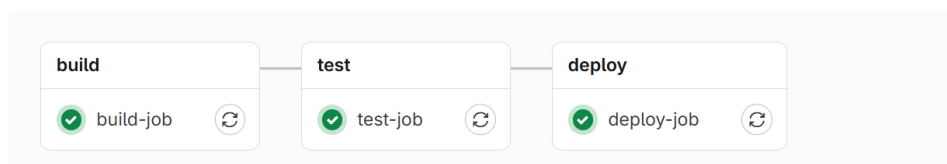
## add deploy stage

✓ Passed Vishnu Raj created pipeline for commit 0df410a7 5 minutes ago, finished 3 minutes ago

For main

latest branch 3 jobs 1.24 1 minute 14 seconds, queued for 9 seconds

Pipeline Jobs 3 Tests 0



## build stage

Search visible log output

```
15 Initialized empty Git repository in /builds/vishnuraj784-group/extends/.git/
16 Created fresh repository.
17 Checking out 0df410a7 as detached HEAD (ref is main)...
18 Skipping Git submodules setup
19 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
20 Executing "step_script" stage of the job script
21 Using effective pull policy of [always] for container maven:latest
22 Using docker image sha256:9cf0d19b18091e0daa06f91ef426d44cbaf5586c75d48f97d882054e416a0839 for maven:latest with digest maven@sha256:fe1de86d91ca06519b58fc85ff737e159c18c0d0a8c7f4c66601b74dba136c5c ...
23 $ echo "Preparing environment..."
24 Preparing environment...
25 $ echo "Building stage"
26 Building stage
27 $ mkdir build
28 $ echo "This application is build for extends file" > build/app.txt
29 Uploading artifacts for successful job
30 Uploading artifacts...
31 build/: found 2 matching artifact files and directories
32 Uploading artifacts as "archive" to coordinator... 201 Created correlation_id=db54ae30f08af05da0ccc86410f13c29 id=11141911669 responseStatus=201 Created token=eyJraWQiO
33 Cleaning up project directory and file based variables
34 Job succeeded
```

Duration: 24 seconds

Finished: 6 minutes ago

Queued: 0 seconds

Timeout: 1h (from project)

Runner: #12270852 (Jhc\_Jxvh) 3-green.saas-linux-small-amd64.runners-manager.gitlab.com/default

Source: Push

Job artifacts

These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.

Keep Download Browse

Commit 0df410a7

add deploy stage

Pipeline #2004445457 Passed for

## test stage

Search visible log output

```
18 Skipping Git submodules setup
19 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
20 Downloading artifacts
21 Downloading artifacts for build-job (11141911669)...
22 Downloading artifacts from coordinator... ok correlation_id= host=storage.googleapis.com id=11141911669 responseStatus=200 OK token=eyJraWQiO
23 Executing "step_script" stage of the job script
24 Using effective pull policy of [always] for container maven:latest
25 Using docker image sha256:9cf0d19b18091e0daa06f91ef426d44cbaf5586c75d48f97d882054e416a0839 for maven:latest with digest maven@sha256:fe1de86d91ca06519b58fc85ff737e159c18c0d0a8c7f4c66601b74dba136c5c ...
26 $ echo "Preparing environment..."
27 Preparing environment...
28 $ echo "Testing stage"
29 Testing stage
30 $ echo "Running test"
31 Running test
32 $ cat build/app.txt
33 This application is build for extends file
34 $ echo "All test passed!"
35 All test passed!
36 Cleaning up project directory and file based variables
37 Job succeeded
```

Duration: 23 seconds

Finished: 6 minutes ago

Queued: 0 seconds

Timeout: 1h (from project)

Runner: #12270845 (JLgUopmM) 1-green.saas-linux-small-amd64.runners-manager.gitlab.com/default

Source: Push

Commit 0df410a7

add deploy stage

Pipeline #2004445457 Passed for main

test

Related jobs

→ test-job

## deploy stage

Search visible log output

```
14 Fetching changes with git depth set to 20...
15 Initialized empty Git repository in /builds/vishnuraj784-group/extends/.git/
16 Created fresh repository.
17 Checking out 0df410a7 as detached HEAD (ref is main)...
18 Skipping Git submodules setup
19 $ git remote set-url origin "${CI_REPOSITORY_URL}" || echo 'Not a git repository; skipping'
20 Downloading artifacts
21 Downloading artifacts for build-job (11141911669)...
22 Downloading artifacts from coordinator... ok correlation_id= host=storage.googleapis.com id=11141911669 responseStatus=200 OK token=eyJraWQiO
23 Executing "step_script" stage of the job script
24 Using effective pull policy of [always] for container maven:latest
25 Using docker image sha256:9cf0d19b18091e0daa06f91ef426d44cbaf5586c75d48f97d882054e416a0839 for maven:latest with digest maven@sha256:fe1de86d91ca06519b58fc85ff737e159c18c0d0a8c7f4c66601b74dba136c5c ...
26 $ echo "Preparing environment..."
27 Preparing environment...
28 $ echo "Deploying stage"
29 Deploying stage
30 $ echo "Deployment completed"
31 Deployment completed
32 Cleaning up project directory and file based variables
33 Job succeeded
```

Duration: 27 seconds

Finished: 6 minutes ago

Queued: 0 seconds

Timeout: 1h (from project)

Runner: #12270852 (Jhc\_Jxvh) 3-green.saas-linux-small-amd64.runners-manager.gitlab.com/default

Source: Push

Commit 0df410a7

add deploy stage

Pipeline #2004445457 Passed for main

deploy

Related jobs

→ deploy-job

## 5. Maven Build Lifecycle Assignment: Validate, Test, Install, Publish, and Deploy

### Set up a Local Maven Build

- Clone or create a simple Maven-based Java project (pom.xml must be present).
  - To create a new project:

```
mvn archetype:generate -DgroupId=com.example \
    -DartifactId=myapp \
    -DarchetypeArtifactId=maven-archetype-quickstart \
    -DinteractiveMode=false
```

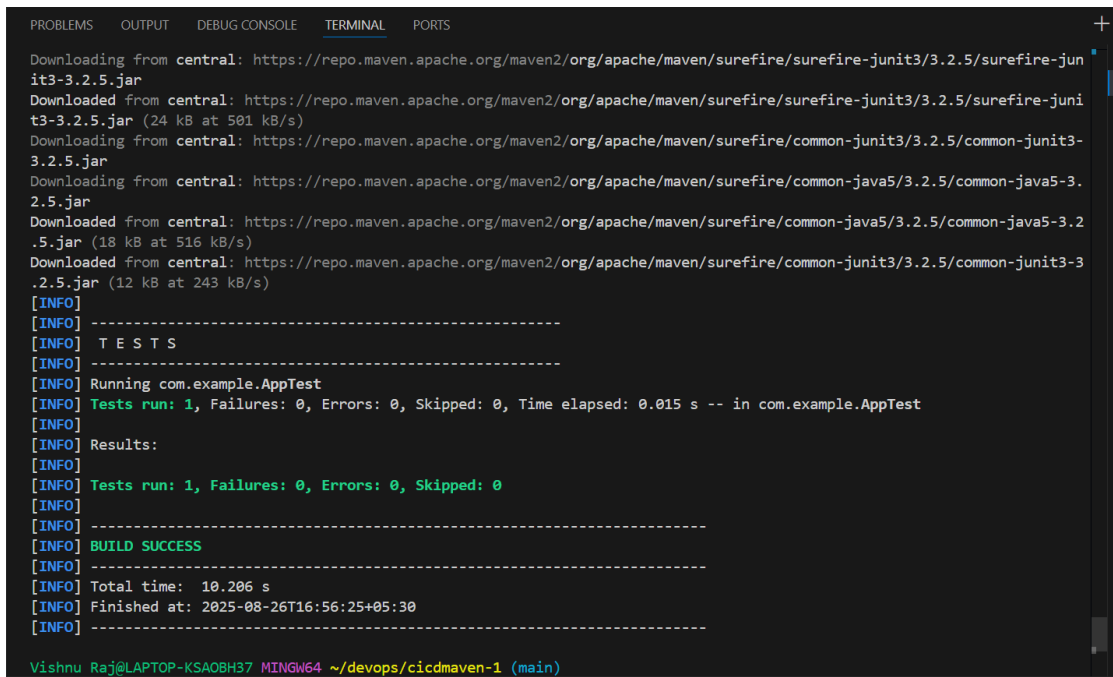
- This will generate a project with a pom.xml.
- Run the **validate** phase to ensure the project is correctly configured.

```
mvn validate
```

### Run Tests

- Execute unit tests using the mvn test command.

```
mvn test
```
- Ensure the tests pass successfully



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire-junit3/3.2.5/surefire-junit3-3.2.5.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire-junit3/3.2.5/surefire-junit3-3.2.5.jar (24 kB at 501 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/common-junit3/3.2.5/common-junit3-3.2.5.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/common-junit3/3.2.5/common-junit3-3.2.5.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/common-java5/3.2.5/common-java5-3.2.5.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/common-java5/3.2.5/common-java5-3.2.5.jar (18 kB at 516 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/common-junit3/3.2.5/common-junit3-3.2.5.jar (12 kB at 243 kB/s)
[INFO]
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running com.example.AppTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.015 s -- in com.example.AppTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 10.206 s
[INFO] Finished at: 2025-08-26T16:56:25+05:30
[INFO]
Vishnu Raj@LAPTOP-KSAOBH37 MINGW64 ~/devops/cicdmaven-1 (main)
```

## Install the Build Locally

- Use mvn install to install the built package into the local Maven repository

mvn install

```
solver-api-1.9.18.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/resolver/maven-resolver-api/1.9.18/maven-resolver-api-1.9.18.jar (157 kB at 3.9 MB/s)
[INFO] Installing C:\Users\Vishnu Raj\devops\cicdmaven-1\pom.xml to C:\Users\Vishnu Raj\.m2\repository\com\example\sum-app\1.0-SNAPSHOT\sum-app-1.0-SNAPSHOT.pom
[INFO] Installing C:\Users\Vishnu Raj\devops\cicdmaven-1\target\sum-app-1.0-SNAPSHOT.jar to C:\Users\Vishnu Raj\.m2\repository\com\example\sum-app\1.0-SNAPSHOT\sum-app-1.0-SNAPSHOT.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 4.452 s
[INFO] Finished at: 2025-08-26T17:16:37+05:30
[INFO] -----
Vishnu Raj@LAPTOP-KSAOBH37 MINGW64 ~/devops/cicdmaven-1 (main)
$
```

## Publish the Build to an Artifact Repository (Optional)

- Configure **GitLab Package Registry** or **Apache Nexus** or **JFrog Artifactory** as a Maven repository.

In your pom.xml, add a <distributionManagement> section:

```
<distributionManagement>

  <repository>

    <id>internal-releases</id>

    <url>https://your-repo-url/repository/maven-releases/</url>

  </repository>

  <snapshotRepository>

    <id>internal-snapshots</id>

    <url>https://your-repo-url/repository/maven-snapshots/</url>

  </snapshotRepository>

</distributionManagement>
```

In ~/.m2/settings.xml, configure credentials:

```
<servers>

  <server>
```

```

<id>internal-releases</id>

<username>your-username</username>

<password>your-password</password>

</server>

<server>

  <id>internal-snapshots</id>

  <username>your-username</username>

  <password>your-password</password>

</server>

</servers>

```

- Use mvn deploy to publish the artifact to the repository.

mvn deploy

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

[INFO] --- compiler:3.13.0:testCompile (default-testCompile) @ sum-app ---
[INFO] Nothing to compile - all classes are up to date.
[INFO] --- surefire:3.2.5:test (default-test) @ sum-app ---
[INFO] Using auto detected provider org.apache.maven.surefire.junit.JUnit3Provider
[INFO]
[INFO] -----
[INFO]  T E S T S
[INFO] -----
[INFO] Running com.example.AppTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.018 s -- in com.example.AppTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] --- jar:3.4.1:jar (default-jar) @ sum-app ---

```

## Deploy the Application

- Package the application (mvn package).

mvn package

```
[INFO] --- resources:3.3.1:testResources (default-testResources) @ sum-app ---
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory C:\Users\Vishnu Raj\devops\cicdmaven-1\src\test\resources
[INFO] --- compiler:3.13.0:testCompile (default-testCompile) @ sum-app ---
[INFO] Nothing to compile - all classes are up to date.
[INFO] --- surefire:3.2.5:test (default-test) @ sum-app ---
[INFO] Using auto detected provider org.apache.maven.surefire.junit.JUnit3Provider
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running com.example.AppTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.016 s -- in com.example.AppTest
[INFO] Results:
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO] --- jar:3.4.1:jar (default-jar) @ sum-app ---
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.578 s
[INFO] Finished at: 2025-08-26T18:11:19+05:30
[INFO] -----
```

- Deploy the generated JAR/WAR file to a **local or remote application server** (Tomcat, WildFly, etc.).

mvn deploy

## 6. Implement Protected Branches in GitLab

Create a new branch (e.g., main or release or master) that you want to protect.

- Go to **Code** → **Branches**.
- Click **New branch**.
- Enter a branch name (e.g., main, master, or release).
- Select the source branch (commonly `develop` or an existing stable branch).
- Click **Create branch**.

devops / DemoCICD / Branches / New branch

### New branch

Branch name

Create from

Existing branch name, tag, or commit SHA

**Navigate to Repository Settings** in GitLab and configure the selected branch as a protected branch.

- Navigate to:
  - **Project → Settings → Repository.**
  - Expand the **Protected branches** section.
- In the dropdown, select **Protected branches** and click on **Add protected branch** (main, master, or release).
- Select your new branch (main, master, or release).
- Click **Protect**.

**Set restrictions** on who can push, merge, and force-push to the protected branch.

- Select **Protected branches** and click on **Add protected branch** (main, master, or release).
- **Allowed to merge** → Choose roles (e.g., *Maintainers only*).
- **Allowed to push and merge** → Restrict to *No one* or *Maintainers only* (to enforce merge requests).
- **Allowed to force push** → Leave disabled (to prevent history rewriting).
- **Require approval from code owners** → Enable if CODEOWNERS file is used.
- Click **Protect**.

**Test the protection rules** by attempting actions with different user roles and permissions.

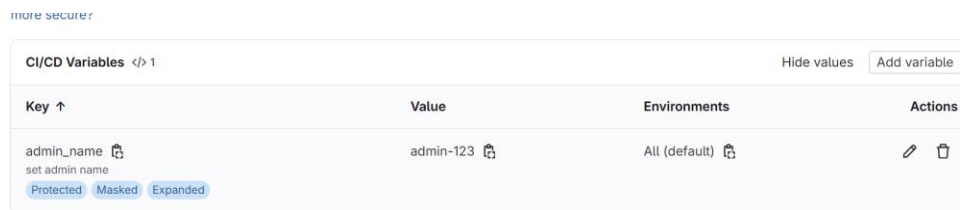
- **As a Developer:**
  - Try pushing directly → should be **blocked**.
  - Try merging an MR → may be blocked depending on settings.
- **As a Maintainer:**
  - Try merging an MR → should be **allowed**.
  - Try force-pushing → should be **blocked** unless explicitly allowed.
- **As a Guest/Reporter:**
  - Should not be able to push/merge at all.



## 7. Implement Protected Variables in GitLab

### Navigate to CI/CD Settings

1. Open your GitLab project.
2. Go to **Settings** → **CI/CD** → **Variables**.
3. Click **Expand** next to Variables.
4. Click **Add Variable**.
5. Check **Protected** and **Masked**
6. Enter **Key** and **Value**
7. Click **Save Variable**.



### YAML script

stages:

- test

- deploy

test\_protected:

stage: test

script:

- echo "Running tests..."

- echo "admin\_name is not accessible here for unprotected branches."

deploy:

stage: deploy

script:

- echo "Deploying with secret token:\$admin\_name"

# Output

## GitLab build pipeline

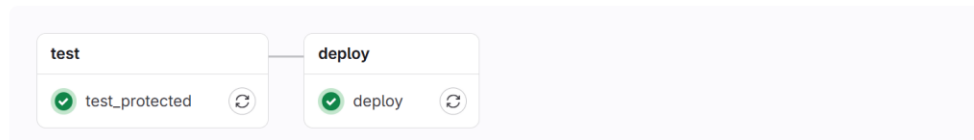
### to check main is protect for variable

✓ Passed Vishnu Raj created pipeline for commit 6825fa84 3 minutes ago, finished 2 minutes ago

For main

latest branch 2 jobs 0.97 58 seconds, queued for 0 seconds

Pipeline Jobs 2 Tests 0



## test stage

devops / maven\_sample / Jobs / #11072562155

Search visible log output

14 Getting source from Git repository

15 Gitly correlation ID: a50788bbb82e643f47cd70d732c1e04a

16 Fetching changes with git depth set to 20...

17 Initialized empty Git repository in /builds/vishnuraj784-group/maven\_sample/.git/

18 Created fresh repository.

19 Checking out 6825fa84 as detached HEAD (ref is main)...

20 Skipping Git submodules setup

21 \$ git remote set-url origin "\${CI\_REPOSITORY\_URL}" || echo 'Not a git repository; skipping'

22 Executing "step\_script" stage of the job script

23 Using default image

24 Using effective pull policy of [always] for container ruby:3.1

25 Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 for ruby:3.1 with digest ruby@sha256:91627f55e8969086aab67d15c92fb930500ff73948803da1330b8a853f

26 \$ echo "Running tests..."

27 Running tests...

28 \$ echo "admin\_name is not accessible here for unprotected branches."

29 admin\_name is not accessible here for unprotected branches.

30 Cleaning up project directory and file based variables

31 Job succeeded

Duration: 29 seconds

Finished: 7 minutes ago

Queued: 0 seconds

Timeout: 1h (from project)

Runner: #12270852 (Jhc\_xjvh) 3-green.saas-linux-small-amd64.runners-manager.gitlab.com/default

Source: Push

Commit 6825fa84

to check main is protect for variable

Pipeline #1992945314 ✓ Passed for main

test

Related jobs

→ test\_protected

## deploy stage

devops / maven\_sample / Jobs / #11072562156

Search visible log output

11 Preparing environment

12 Using effective pull policy of [always] for container sha256:b3f7103f2e63be34f06e08ac5d3f4307ce1b053694fba67eb5a35a98305c2b57

13 Running on runner-jlguopmmv-project-73186302-concurrent-0 via runner-jlguopmmv-s-l-s-amd64-1755670818-fdd8d7e4...

14 Getting source from Git repository

15 Gitly correlation ID: 6134ea7216fa249e60301ea620d8488e

16 Fetching changes with git depth set to 20...

17 Initialized empty Git repository in /builds/vishnuraj784-group/maven\_sample/.git/

18 Created fresh repository.

19 Checking out 6825fa84 as detached HEAD (ref is main)...

20 Skipping Git submodules setup

21 \$ git remote set-url origin "\${CI\_REPOSITORY\_URL}" || echo 'Not a git repository; skipping'

22 Executing "step\_script" stage of the job script

23 Using default image

24 Using effective pull policy of [always] for container ruby:3.1

25 Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 for ruby:3.1 with digest ruby@sha256:91627f55e8969086aab67d15c92fb930500ff73948803da1330b8a853f

26 \$ echo "Deploying with secret token:\$admin\_name"

27 Deploying with secret token:admin-123

28 Cleaning up project directory and file based variables

29 Job succeeded

Duration: 28 seconds

Finished: 7 minutes ago

Queued: 0 seconds

Timeout: 1h (from project)

Runner: #12270845 (JLgUopmM) 1-green.saas-linux-small-amd64.runners-manager.gitlab.com/default

Source: Push

Commit 6825fa84

to check main is protect for variable

Pipeline #1992945314 ✓ Passed for main

deploy

Related jobs

→ deploy

## 8. Implement a Sample GitLab Pipeline with rules

### YAML script

stages:

- build
- test

build-job:

stage: build

script:

- echo "Building the project..."

rules:

- if: '\$CI\_COMMIT\_BRANCH == "main"'

test-job:

stage: test

script:

- echo "Running tests..."

rules:

- if: '\$CI\_COMMIT\_BRANCH == "version/1.1.0"'

### Output

#### GitLab main branch

### change on main branch

✓ Passed Vishnu Raj created pipeline for commit `c602d82b` 16 minutes ago, finished 15 minutes ago

For `main`

latest branch 1 job 0.5 30 seconds, queued for 0 seconds

Pipeline Jobs 1 Tests 0

build



build-job



Search visible log output

11

Preparing environment

00:04

12

Using effective pull policy of [always] for container sha256:27e2e61dd9bc2d3cf62bdeb24b759c7512e27f6c36a708de856614c6ea1b311a

13

Running on runner-nthfetyxq-project-73186302-concurrent-0 via runner-nthfetyxq-s-l-s-amd64-1755696503-614939b5...

14

Getting source from Git repository

00:01

15

Gitaly correlation ID: 8c48fd9f0a48e5edbb74b65003d71bf6

16

Fetching changes with git depth set to 20...

17

Initialized empty Git repository in /builds/vishnuraj784-group/maven\_sample/.git/

18

Created fresh repository.

19

Checking out c602d82b as detached HEAD (ref is main)...

20

Skipping Git submodules setup

21

\$ git remote set-url origin "\${CI\_REPOSITORY\_URL}" || echo 'Not a git repository; skipping'

22

Executing "step\_script" stage of the job script

00:01

23

Using default image

24

Using effective pull policy of [always] for container ruby:3.1

25

Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 for ruby:3.1 with digest ruby@sha256:91627f55e8969006aab67d15c92fb930500ff73948803da1330b8a853fecebb5...

26

\$ echo "Building the project..."

27

Building the project...

28

Cleaning up project directory and file based variables

00:00

29

Job succeeded

Duration: 30 seconds

Finished: 22 minutes ago

Queued: 0 seconds

Timeout: 1h (from project)

Runner: #12270857 (nthFETyX) 4-green.saas-linux-small-amd64.runners-manager.gitlab.com/default

Source: Push

Commit c602d82b  
change on main branch

Pipeline #1993953069 Passed for main  
build

Related jobs  
→ build-job

## GitLab sub branch

# feature version added

✓ Passed Vishnu Raj created pipeline for commit `bbba83fa` 13 minutes ago, finished 13 minutes ago

For `version/1.1.0`

latest branch 1 job 0.49 29 seconds, queued for 1 seconds

Pipeline Jobs 1 Tests 0

test

✓ test-job

Search visible log output

11

Preparing environment

00:05

12

Using effective pull policy of [always] for container sha256:849cf5a8c1941a7233eacae902398d9d807710975924ae6e394adebb94fc2733

13

Running on runner-xs6vzpqo-project-73186302-concurrent-0 via runner-xs6vzpqo-s-l-s-amd64-1755696815-3a502149...

14

Getting source from Git repository

00:01

15

Gitaly correlation ID: 86eb9f4bf2f7f517bee01d908b725bba

16

Fetching changes with git depth set to 20...

17

Initialized empty Git repository in /builds/vishnuraj784-group/maven\_sample/.git/

18

Created fresh repository.

19

Checking out bbba83fa as detached HEAD (ref is version/1.1.0)...

20

Skipping Git submodules setup

21

\$ git remote set-url origin "\${CI\_REPOSITORY\_URL}" || echo 'Not a git repository; skipping'

22

Executing "step\_script" stage of the job script

00:01

23

Using default image

24

Using effective pull policy of [always] for container ruby:3.1

25

Using docker image sha256:9981df1d883b246c27c62f8ccb9b57d3e07d14cee8092299e102b4a69c35ea61 for ruby:3.1 with digest ruby@sha256:91627f55e8969006aab67d15c92fb930500ff73948803da1330b8a853fecebb5...

26

\$ echo "Running tests..."

27

Running tests...

28

Cleaning up project directory and file based variables

00:00

29

Job succeeded

Duration: 29 seconds

Finished: 16 minutes ago

Queued: 0 seconds

Timeout: 1h (from project)

Runner: #12270859 (xs6Vzpqo) 5-green.saas-linux-small-amd64.runners-manager.gitlab.com/default

Source: Push

Commit bbba83fa  
feature version added

Pipeline #1993966075 Passed for version/1.1.0  
test

Related jobs  
→ test-job

## 9. Setting Up Shell and Docker Executors

### Shell Executor Runner

#### Step 1: Configure a **Shell Executor Runner**

- Register Runner:

`gitlab-runner register`

- Your project/group's **Settings > CI/CD > Runners > Create project runner**.
- Fill the **Tags** and **Runner description > Create runner**

Copy and paste the command into your command line to register the runner

- To check Runner is now active:

`gitlab-runner list`

#### Step 2: create a `.gitlab-ci.yml`

stages:

- test

test-job:

stage:

tags:

- runner-shell

script:

- echo "Running on shell executor"

- whoami

- pwd

### Step 3: Output of the simple shell script job

#### GitLab main branch

## add new runner for the shell

✓ Passed Vishnu Raj created pipeline for commit `b3326346` 20 minutes ago, finished 19 minutes ago

For `main`

latest branch 1 job 15 seconds, queued for 52 seconds

Pipeline Jobs 1 Tests 0

test

✓ test-job

#### test stage

Search visible log output

```
5 Using shell (pwsh) executor...
6 Preparing environment
7 Running on LAPTOP-KSA0BH37...
8 Getting source from Git repository
9 Gitaly correlation ID: 96488a4265e54d8aa4531f019ecdbbe0
10 Fetching changes with git depth set to 20...
11 Initialized empty Git repository in C:/Users/Vishnu Raj/builds/HXGqIus-I/0/vishnuraj784-group/runner/.git/
12 Created fresh repository.
13 Checking out b3326346 as detached HEAD (ref is main)...
14 git-lfs/3.6.1 (GitHub; windows amd64; go 1.23.3; git ea47a34b)
15 Skipping Git submodules setup
16 Executing "step_script" stage of the job script
17 $ echo "Running on shell executor"
18 Running on shell executor
19 $ whoami
20 nt authority\system
21 $ pwd
22 Path
23 ----
24 C:\Users\Vishnu Raj\builds\HXGqIus-I\0\vishnuraj784-group\runner
25 Cleaning up project directory and file based variables
26 Job succeeded
```

Duration: 15 seconds  
Finished: 20 minutes ago  
Queued: 51 seconds  
Timeout: 1h (from project)  
Runner: #49613550 (HXGqlus-) runner-shell  
Source: Push  
Tags: runner-shell  
Commit `b3326346`  
add new runner for the shell  
Pipeline #2004928229 ✓ Passed for main  
test  
Related jobs  
→ ✓ test-job

### Docker Executor Runner

#### Step 1: Configure a Shell Executor Runner

- Register Runner:  
gitlab-runner register
- Your project/group's **Settings > CI/CD > Runners > Create project runner**.
- Fill the **Tags** and **Runner description > Create runner**

Copy and paste the command into your command line to register the runner

- To check Runner is now active:

`gitlab-runner list`

Step 2: Ensure the runner is registered with a specific Docker image

create a .gitlab-ci.yml

stages:

- test

docker-job:

stage: test

tags:

- sample\_docker

image: docker:latest

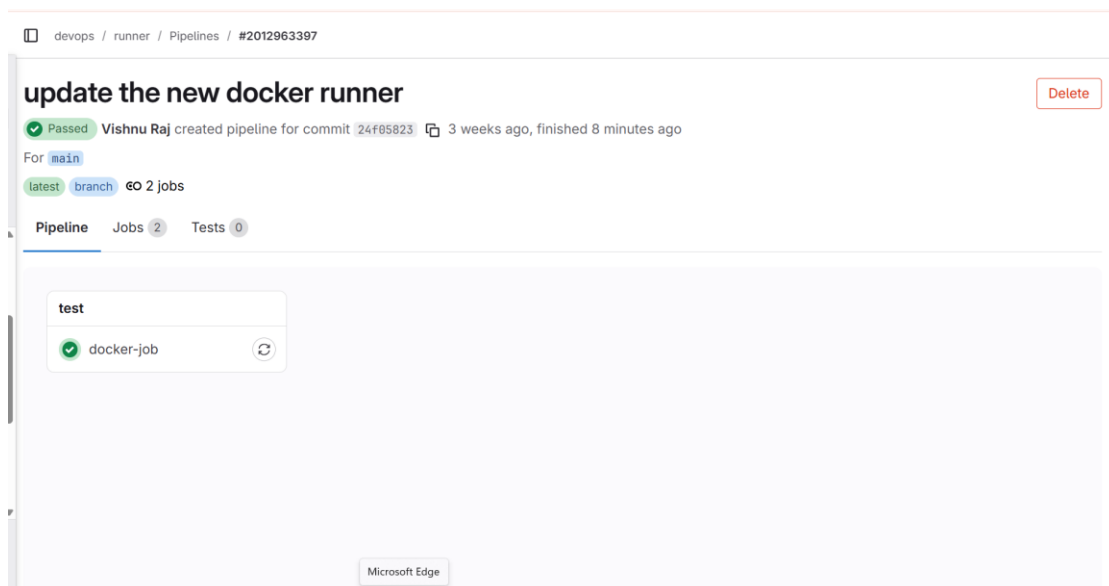
script:

- echo "Running inside Docker!"

- cat /etc/os-release

Step 3: Output of the simple shell script job

GitLab main branch



## test stage

devops / runner / Jobs / #11428823687

Search visible log output

12 Getting source from Git repository 00:03  
13 Gitly correlation ID: b1c0cfc68da4933a0ff91f771a4df5c  
14 Fetching changes with git depth set to 20...  
15 Initialized empty Git repository in /builds/vishnuraj784-group/runner/.git/  
16 Created fresh repository.  
17 Checking out 24f05823 as detached HEAD (ref is main)...  
18 Skipping Git submodules setup  
19 Executing "step\_script" stage of the job script 00:00  
20 Using effective pull policy of [always] for container docker:latest  
21 Using docker image sha256:831644212c5bdd0b3362b5855c87b980ea39a83c9e9adcef2ce03eced99a737a for docker:latest with digest docker@sha256:831644212c5bdd0b3362b5855c87b980ea39a83c9e9adcef2ce03eced99a737a ...  
22 \$ echo "Running inside Docker!"  
23 Running inside Docker!  
24 \$ cat /etc/os-release  
25 NAME="Alpine Linux"  
26 ID=alpine  
27 VERSION\_ID=3.22.1  
28 PRETTY\_NAME="Alpine Linux v3.22"  
29 HOME\_URL="https://alpinelinux.org/"  
30 BUG\_REPORT\_URL="https://gitlab.alpinelinux.org/alpine/aports/-/issues"  
31 Cleaning up project directory and file based variables 00:01  
32 Job succeeded

Duration: 29 seconds  
Finished: 9 minutes ago  
Queued: 32 seconds  
Timeout: 1h (from project) ?  
Runner: #49656741 (R-EQZzH5)  
sample docker runner for application  
Source: Push  
Tags: sample\_docker  
Commit 24f05823 update the new docker runner  
Pipeline #2012963397 Passed for main  
test  
Related jobs  
→ docker-job

## 10.Implement a Sample GitLab Pipeline

### YAML script

#### #stages of the pipeline

stages:

- build
- test
- deploy

#### #global variables

variables:

APP\_NAME: "MySampleApp"

DEV\_NAME: "Demo Name"

#### #build job stage

build-job:

stage: build

tags:



- simple-runner

script:

- echo "Starting build stage for \$APP\_NAME"
- echo "Compiling source code..."
- echo "Build stage completed..."

#test job stage

test-job:

stage: test

tags:

- simple-runner

script:

- echo "Starting test stage for \$APP\_NAME"
- echo "Executing unit test"
- echo "All test successfully completed"

#deploy job stage

deploy-job:

stage: deploy

tags:

- simple-runner

script:

- echo "Start deploying application \$APP\_NAME"
- echo "#APP\_NAME is deployed by \$DEV\_NAME"
- echo "Deploy stage completed"

## Output

### GitLab main branch

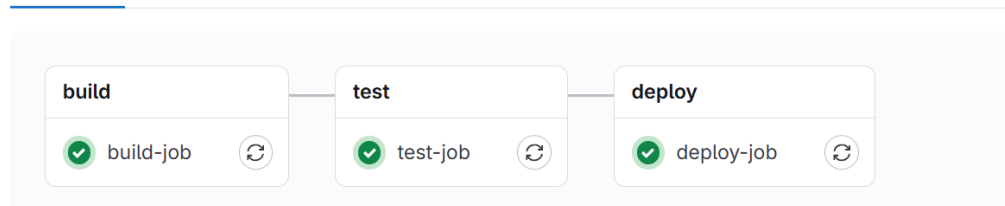
## completed the application

✓ Passed Vishnu Raj created pipeline for commit 3887f831 7 minutes ago, finished 3 minutes ago

For **main**

latest branch 3 jobs 37 seconds, queued for 8 seconds

Pipeline Jobs 3 Tests 0



### build stage

Search visible log output

```
2 on simple-runner esUxQULPW, system ID: s_c3e30dc0b14f
3 Resolving secrets
4 Preparing the "shell" executor 00:00
5 Using Shell (pwsh) executor...
6 Preparing environment 00:02
7 Running on LAPTOP-KSA0BH37...
8 Getting source from Git repository 00:05
9 Gitaly correlation ID: 7bdcbb5984e047c0aa169a2903ee02d5
10 Fetching changes with git depth set to 20...
11 Reinitialized existing Git repository in C:/Users/Vishnu Raj/builds/esUxQULPW/0/vishnuraj784-group/simple_pipeline/.git/
12 Checking out 3887f831 as detached HEAD (ref is main)...
13 git-lfs/3.6.1 (GitHub; windows amd64; go 1.23.3; git ea47a34b)
14 Skipping Git submodules setup
15 Executing "step_script" stage of the job script 00:02
16 $ echo "Starting build stage for $APP_NAME"
17 Starting build stage for MySampleApp
18 $ echo "Compiling source code..."
19 Compiling source code...
20 $ echo "Build stage completed..."
21 Build stage completed...
22 Cleaning up project directory and file based variables 00:03
23 Job succeeded
```

Duration: 13 seconds  
Finished: 8 minutes ago  
Queued: 7 seconds  
Timeout: 10m (from runner) ⓘ  
Runner: #49620673 (esUxQUIP) this runner is used only on this project  
Source: Push  
Tags: simple-runner  
Commit 3887f831 completed the application  
Pipeline #2006101854 ✓ Passed for main  
build

Related jobs  
→ build-job

### test stage

Search visible log output

```
2 on simple-runner esUxQULPW, system ID: s_c3e30dc0b14f
3 Resolving secrets
4 Preparing the "shell" executor 00:00
5 Using Shell (pwsh) executor...
6 Preparing environment 00:03
7 Running on LAPTOP-KSA0BH37...
8 Getting source from Git repository 00:05
9 Gitaly correlation ID: 481d9d3c95d641ff8703f4e3d5fb10b8
10 Fetching changes with git depth set to 20...
11 Reinitialized existing Git repository in C:/Users/Vishnu Raj/builds/esUxQULPW/0/vishnuraj784-group/simple_pipeline/.git/
12 Checking out 3887f831 as detached HEAD (ref is main)...
13 git-lfs/3.6.1 (GitHub; windows amd64; go 1.23.3; git ea47a34b)
14 Skipping Git submodules setup
15 Executing "step_script" stage of the job script 00:02
16 $ echo "Starting test stage for $APP_NAME"
17 Starting test stage for MySampleApp
18 $ echo "Executing unit test"
19 Executing unit test
20 $ echo "All test successfully completed"
21 All test successfully completed
22 Cleaning up project directory and file based variables 00:03
23 Job succeeded
```

Duration: 12 seconds  
Finished: 6 minutes ago  
Queued: 1 minute 41 seconds  
Timeout: 10m (from runner) ⓘ  
Runner: #49620673 (esUxQUIP) this runner is used only on this project  
Source: Push  
Tags: simple-runner  
Commit 3887f831 completed the application  
Pipeline #2006101854 ✓ Passed for main  
test

Related jobs  
→ test-job

## deploy stage

Search visible log output

2 on simple-runner esUxQULPW, system ID: s\_c3e39dc0b14f

3 Resolving secrets

4 Preparing the "shell" executor 00:00

5 Using Shell (pwsh) executor...

6 Preparing environment 00:02

7 Running on LAPTOP-KSA0BH37...

8 Getting source from Git repository 00:04

9 Gitaly correlation ID: bd69399fb24b4b6a8d01b0db00ba1dae

10 Fetching changes with git depth set to 20...

11 Reinitialized existing Git repository in C:/Users/Vishnu Raj/builds/esUxQULPW/0/vishnuraj784-group/simple\_pipeline/.git/

12 Checking out 3887f831 as detached HEAD (ref is main)...

13 git-lfs/3.6.1 (GitHub; windows amd64; go 1.23.3; git ea47a34b)

14 Skipping Git submodules setup

15 Executing "step\_script" stage of the job script 00:01

16 \$ echo "Start deploying application \$APP\_NAME"

17 Start deploying application MySampleApp

18 \$ echo "#APP\_NAME is deployed by \$DEV\_NAME"

19 #APP\_NAME is deployed by Demo Name

20 \$ echo "Deploy stage completed"

21 Deploy stage completed

22 Cleaning up project directory and file based variables 00:02

23 Job succeeded

Duration: 11 seconds

Finished: 5 minutes ago

Queued: 1 minute 40 seconds

Timeout: 10m (from runner) ⓘ

Runner: #49620673 (esUxQUIP) this runner is used only on this project

Source: Push

Tags: simple-runner

Commit 3887f831 ⓘ  
completed the application

Pipeline #2006101854 Passed for main ⓘ  
deploy

Related jobs  
→ deploy-job