TASK 2

Step 1: Open Jenkins and click **"New Item."**

Step 2: Enter a project name, select **"Pipeline,"** then click **OK.**

Step 3: In the **General** tab, scroll down to the **"Pipeline"** section.

Step 4: Under **"Definition,"** select **"Pipeline script from SCM."**

Step 5: In the **SCM** field, choose **"Git,"** then enter your GitHub repository URL.

Step 6: Ensure your GitHub repository contains the following files:

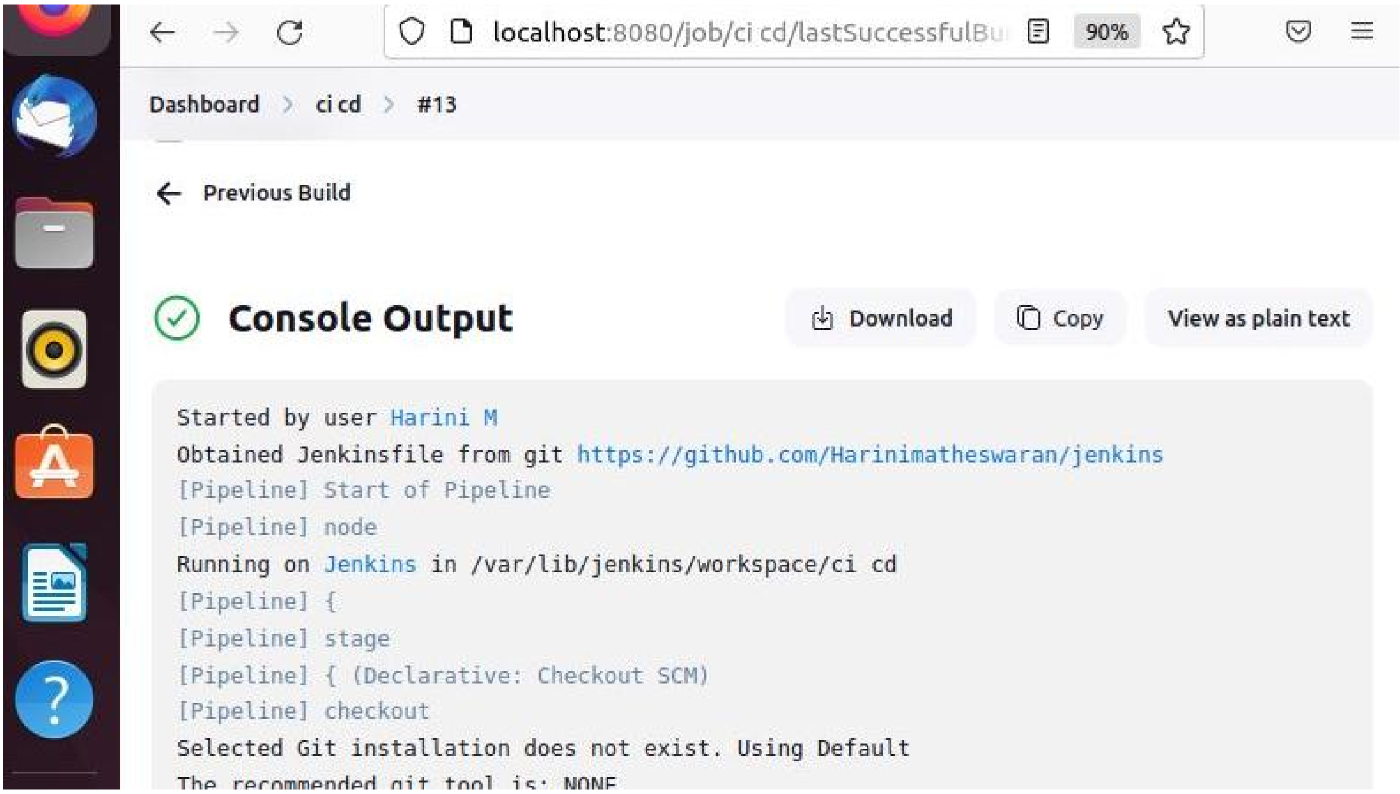
- app.py (Application code)
- Dockerfile (Instructions to build the Docker image)
- docker-compose.yml (For container orchestration)
- Jenkinsfile (Defines the pipeline stages)
- requirements.txt (Dependencies for the application)

Step 7: Click **"Save"** to apply the configuration.

Step 8: On the project page, click **"Build Now"** to start the job.

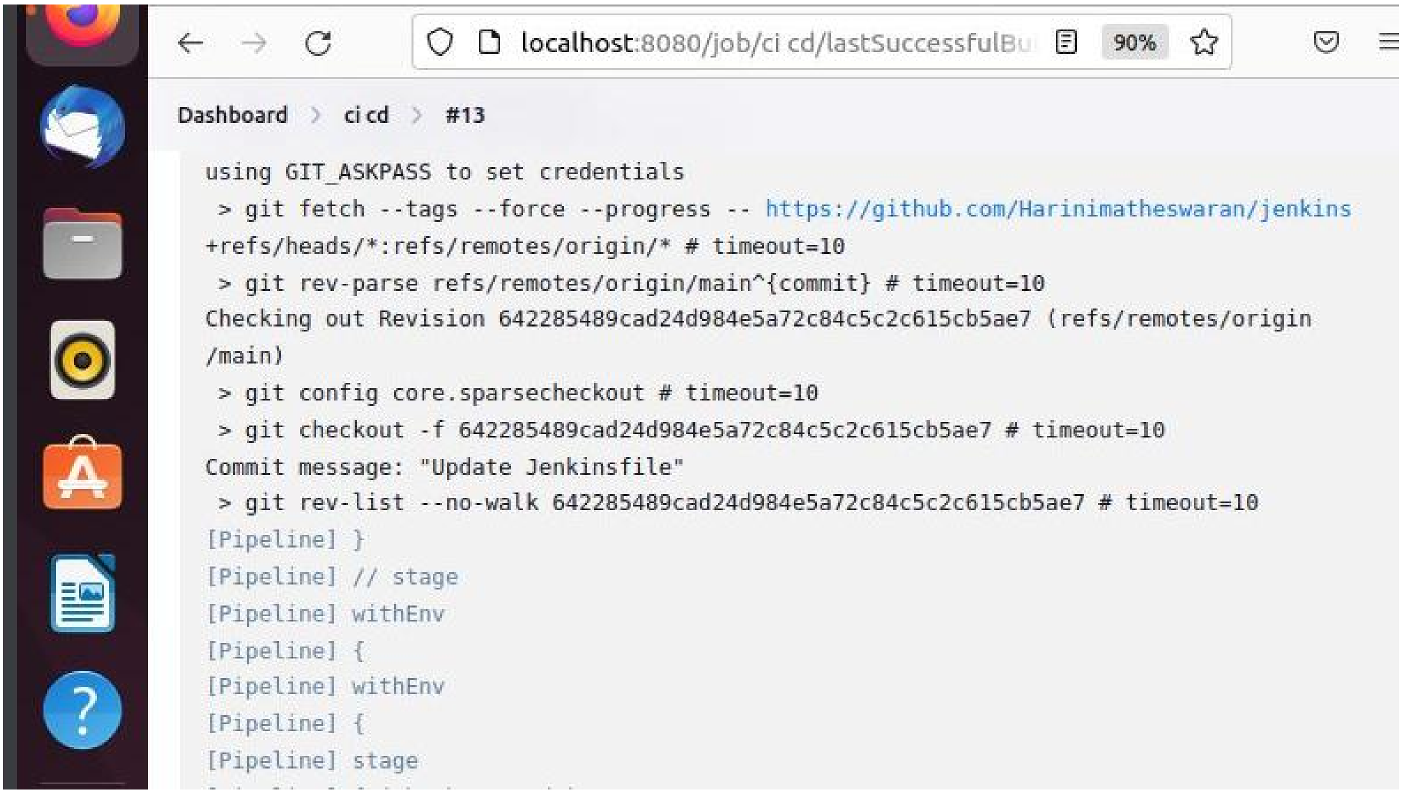
Step 9: The build status will appear in the **Build History** panel (left-side panel).

Step 10: Click on the latest build (#1, #2, etc.) and select **"Console Output"** to view logs and results.

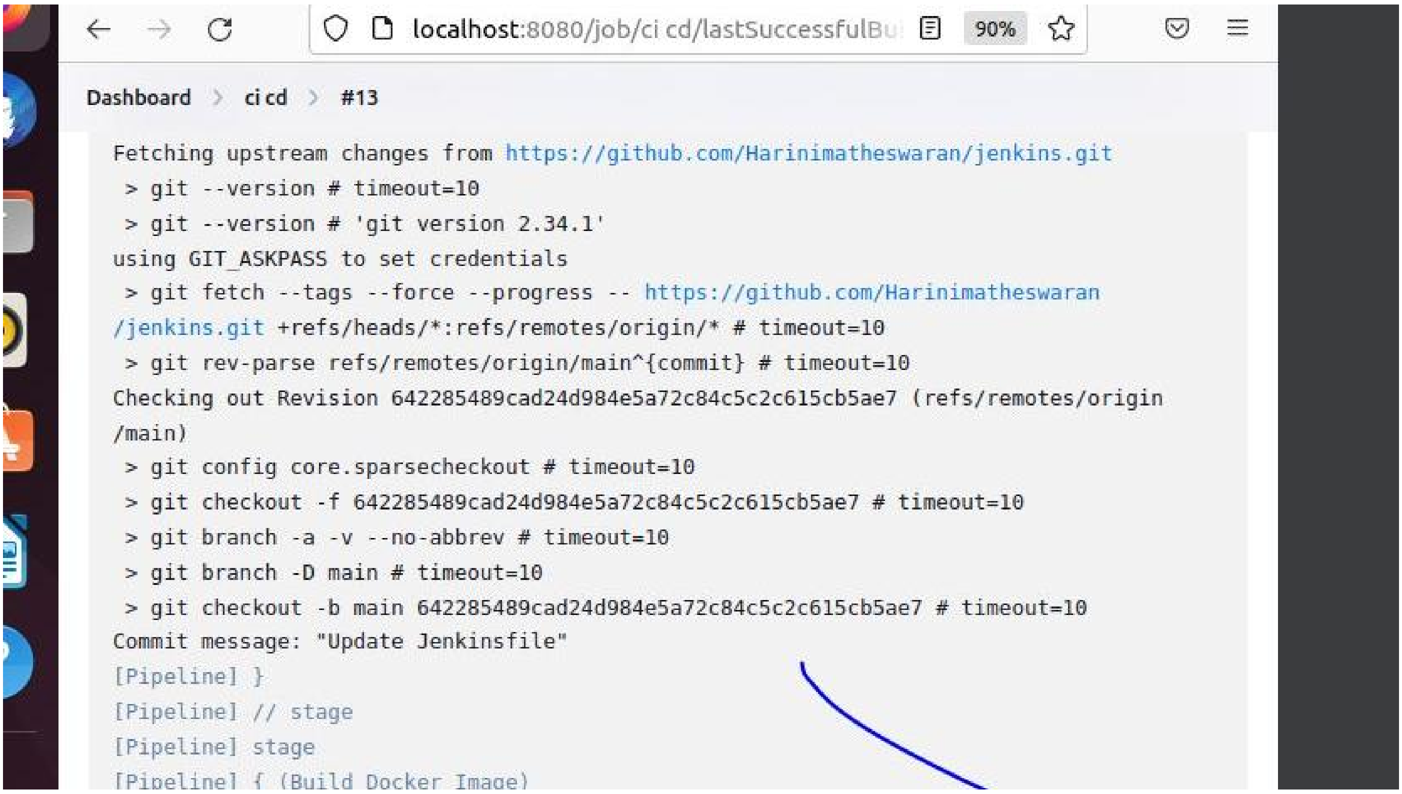


Console Output

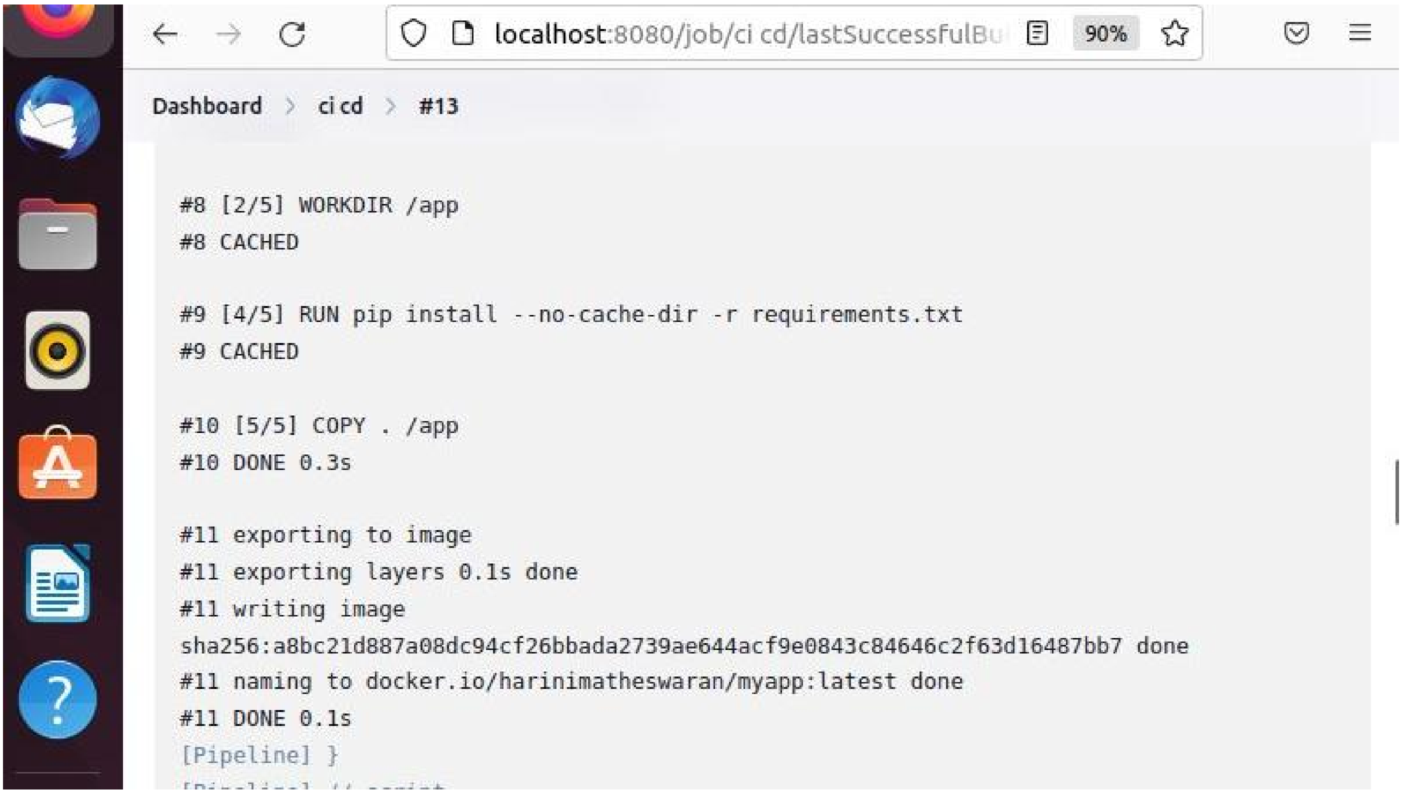
```
Started by user Harini M
Obtained Jenkinsfile from git https://github.com/Harinimatheswaran/jenkins
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/ci cd
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
```



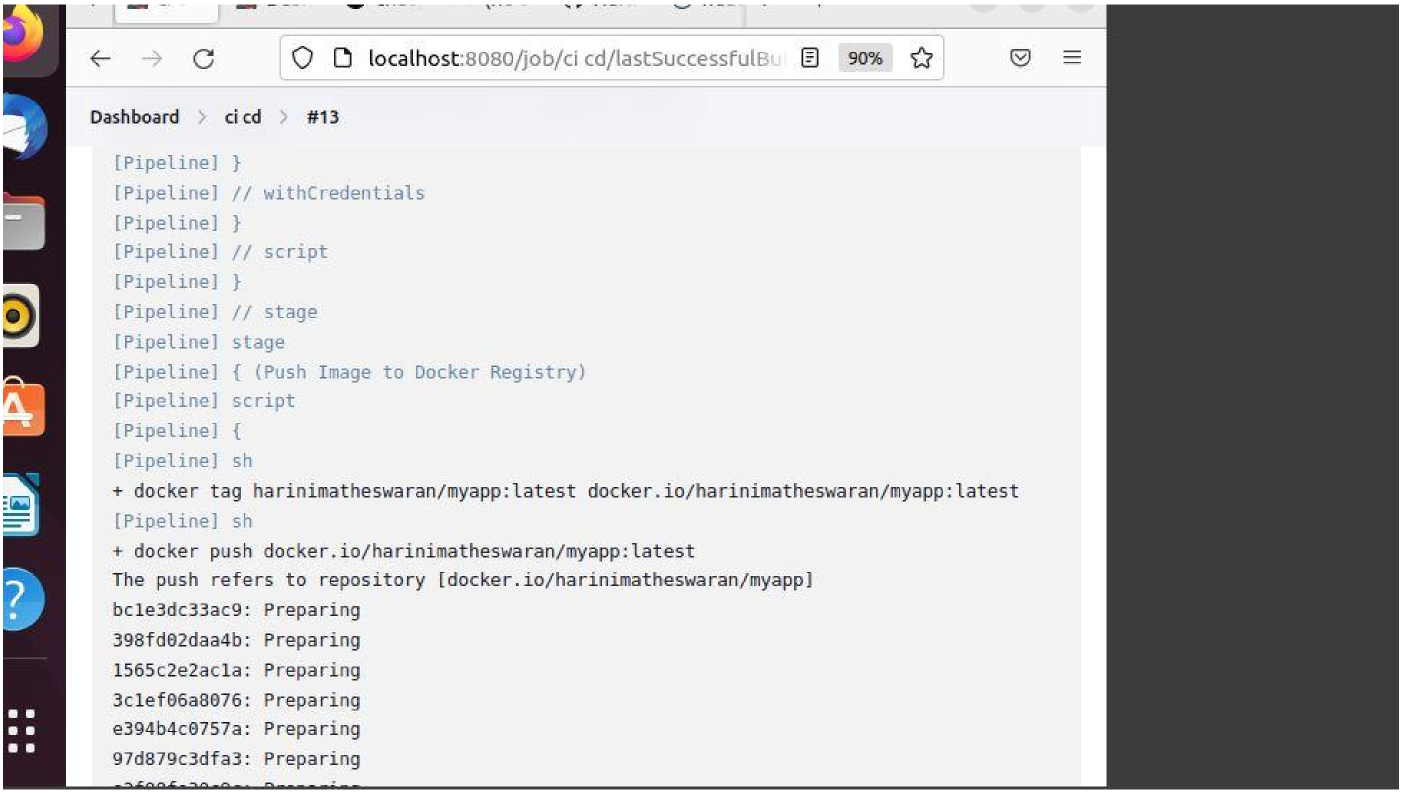
```
using GIT_ASKPASS to set credentials
 > git fetch --tags --force --progress -- https://github.com/Harinimatheswaran/jenkins
+refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 642285489cad24d984e5a72c84c5c2c615cb5ae7 (refs/remotes/origin
/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 642285489cad24d984e5a72c84c5c2c615cb5ae7 # timeout=10
Commit message: "Update Jenkinsfile"
 > git rev-list --no-walk 642285489cad24d984e5a72c84c5c2c615cb5ae7 # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
```



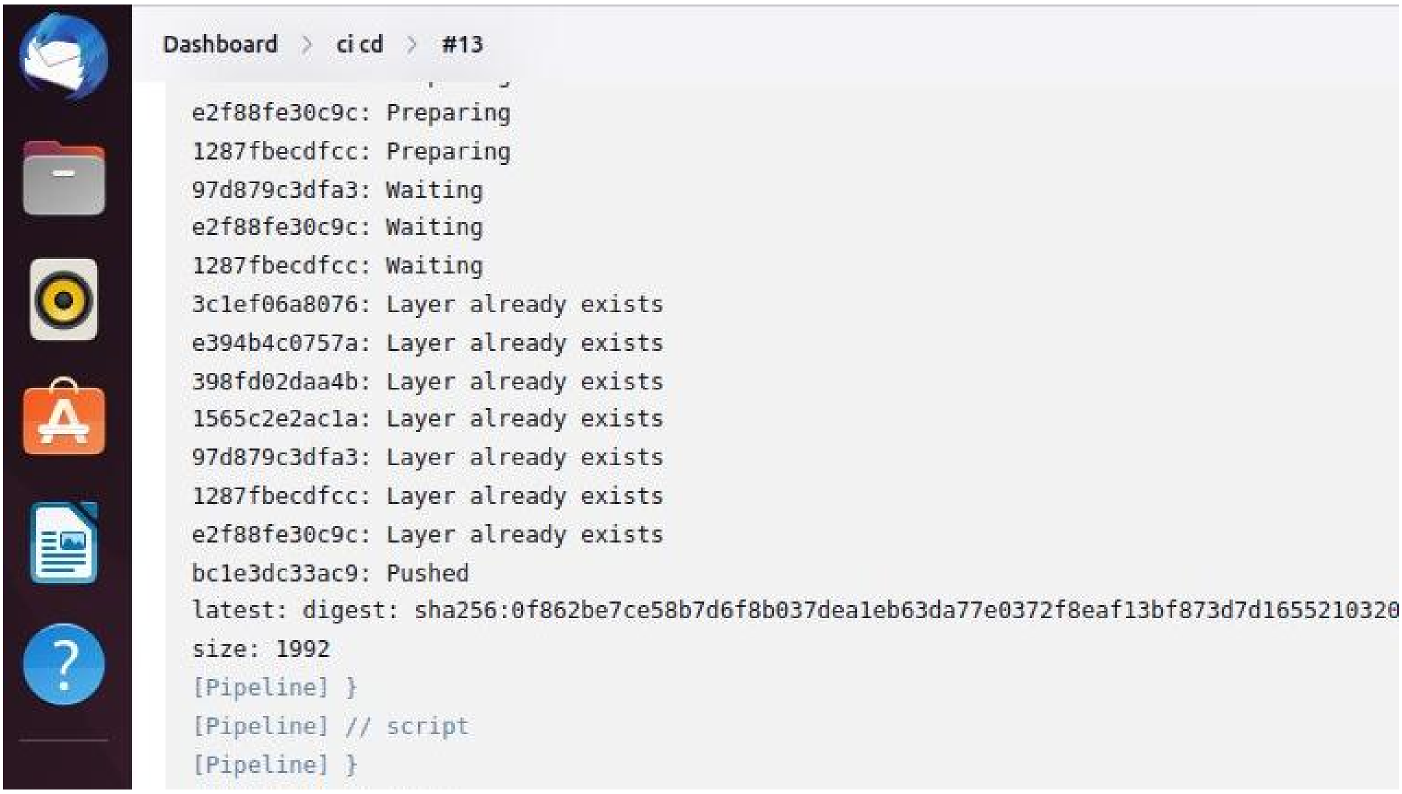
```
Fetching upstream changes from https://github.com/Harinimatheswaran/jenkins.git
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_ASKPASS to set credentials
 > git fetch --tags --force --progress -- https://github.com/Harinimatheswaran
/jenkins.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 642285489cad24d984e5a72c84c5c2c615cb5ae7 (refs/remotes/origin
/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 642285489cad24d984e5a72c84c5c2c615cb5ae7 # timeout=10
 > git branch -a -v --no-abbrev # timeout=10
 > git branch -D main # timeout=10
 > git checkout -b main 642285489cad24d984e5a72c84c5c2c615cb5ae7 # timeout=10
Commit message: "Update Jenkinsfile"
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build Docker Image)
```



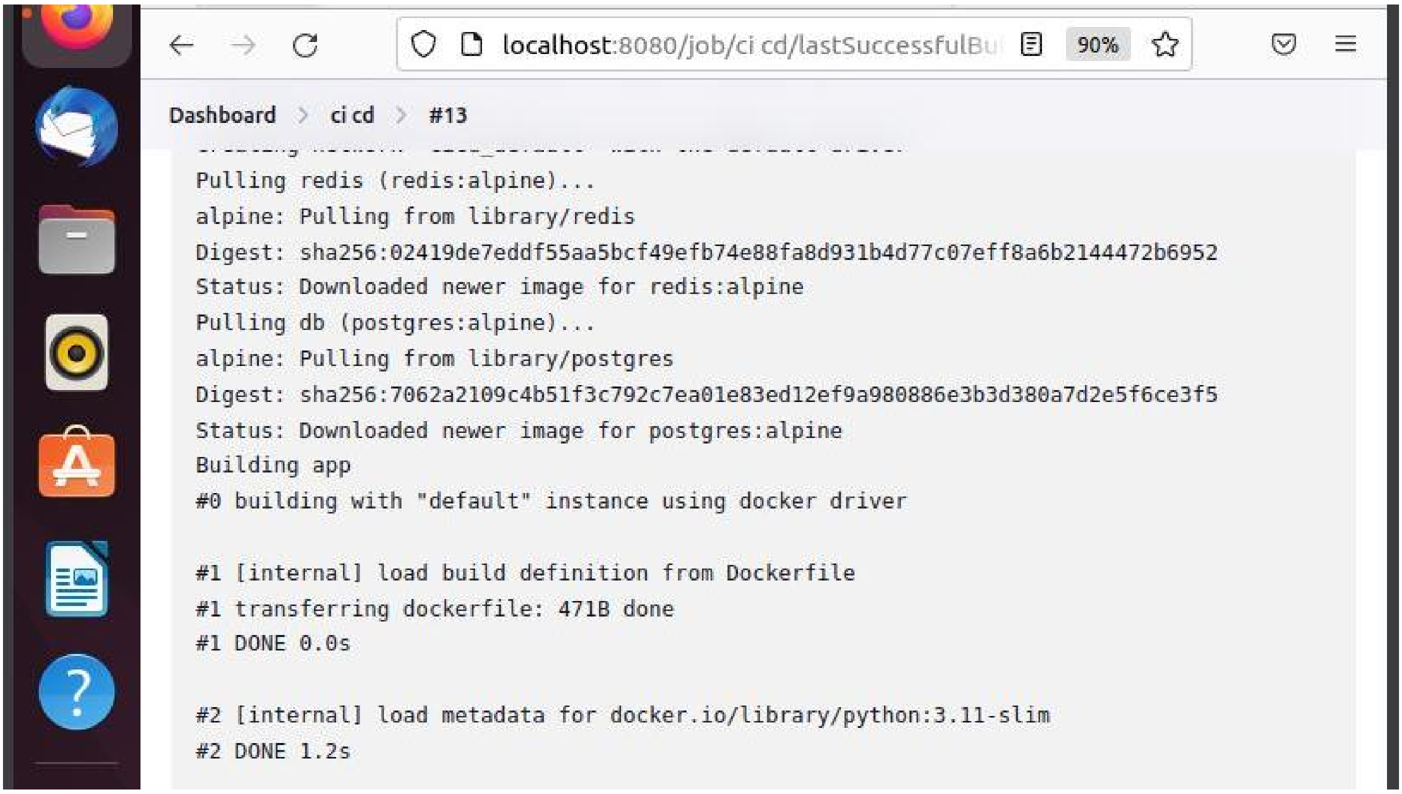
```
#8 [2/5] WORKDIR /app
#8 CACHED

#9 [4/5] RUN pip install --no-cache-dir -r requirements.txt
#9 CACHED

#10 [5/5] COPY . /app
#10 DONE 0.3s

#11 exporting to image
#11 exporting layers 0.1s done
#11 writing image
sha256:a8bc21d887a08dc94cf26bbada2739ae644acf9e0843c84646c2f63d16487bb7 done
#11 naming to docker.io/harinimatheswaran/myapp:latest done
#11 DONE 0.1s
[Pipeline] }
[Pipeline] // script
```



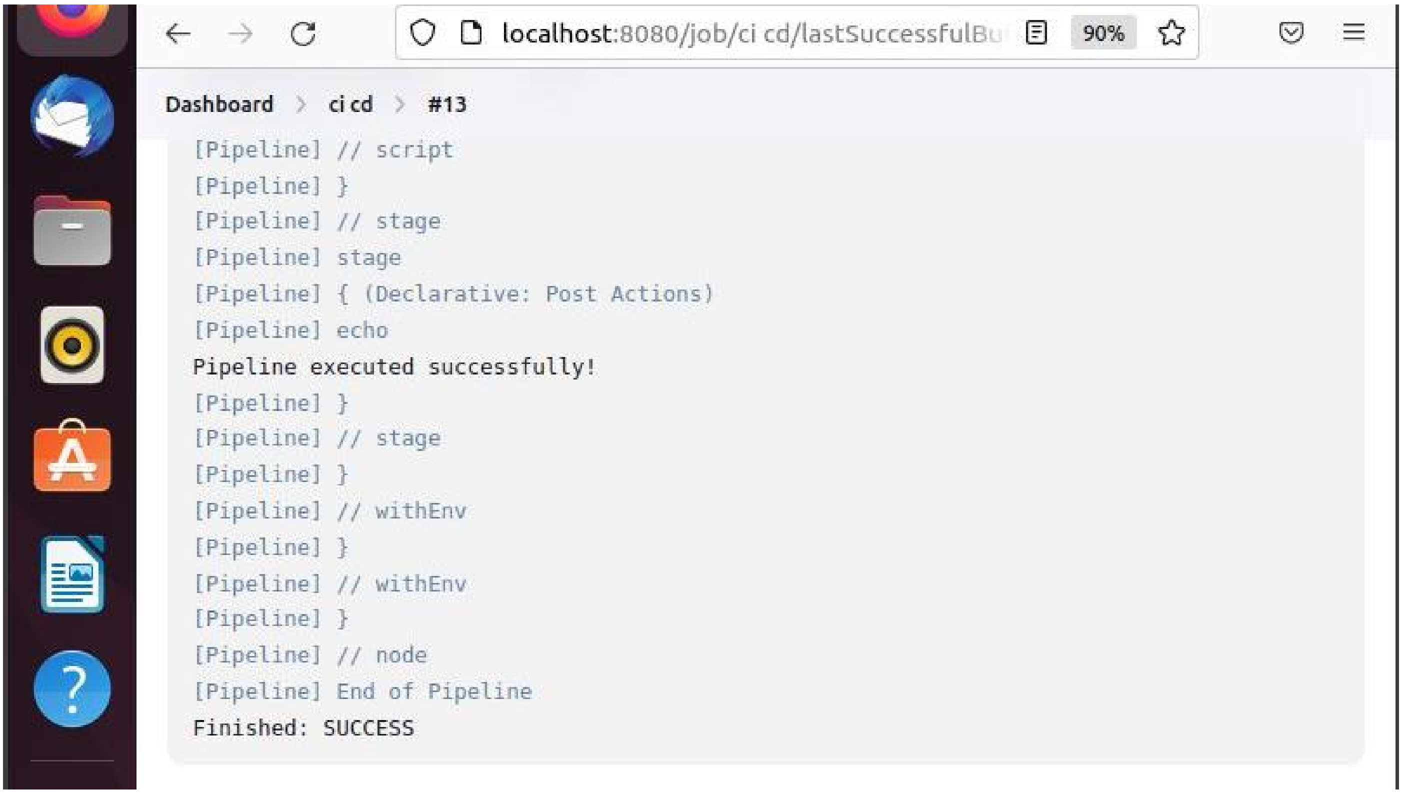
```
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Push Image to Docker Registry)
[Pipeline] script
[Pipeline] {
[Pipeline] sh
+ docker tag harinimatheswaran/myapp:latest docker.io/harinimatheswaran/myapp:latest
[Pipeline] sh
+ docker push docker.io/harinimatheswaran/myapp:latest
The push refers to repository [docker.io/harinimatheswaran/myapp]
bc1e3dc33ac9: Preparing
398fd02daa4b: Preparing
1565c2e2ac1a: Preparing
3c1ef06a8076: Preparing
e394b4c0757a: Preparing
97d879c3dfa3: Preparing
```



```
e2f88fe30c9c: Preparing
1287fbecdfcc: Preparing
97d879c3dfa3: Waiting
e2f88fe30c9c: Waiting
1287fbecdfcc: Waiting
3c1ef06a8076: Layer already exists
e394b4c0757a: Layer already exists
398fd02daa4b: Layer already exists
1565c2e2ac1a: Layer already exists
97d879c3dfa3: Layer already exists
1287fbecdfcc: Layer already exists
e2f88fe30c9c: Layer already exists
bc1e3dc33ac9: Pushed
latest: digest: sha256:0f862be7ce58b7d6f8b037dea1eb63da77e0372f8eaf13bf873d7d1655210320
size: 1992
[Pipeline] }
[Pipeline] // script
[Pipeline] }
```



```
Pulling redis (redis:alpine)...
alpine: Pulling from library/redis
Digest: sha256:02419de7eddf55aa5bcf49efb74e88fa8d931b4d77c07eff8a6b2144472b6952
Status: Downloaded newer image for redis:alpine
Pulling db (postgres:alpine)...
alpine: Pulling from library/postgres
Digest: sha256:7062a2109c4b51f3c792c7ea01e83ed12ef9a980886e3b3d380a7d2e5f6ce3f5
Status: Downloaded newer image for postgres:alpine
Building app
#0 building with "default" instance using docker driver

#1 [internal] load build definition from Dockerfile
#1 transferring dockerfile: 471B done
#1 DONE 0.0s

#2 [internal] load metadata for docker.io/library/python:3.11-slim
#2 DONE 1.2s
```



```
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] echo
Pipeline executed successfully!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```