DevOps Project

Name :- Prashant Balu Mohite

Mobile no :- 9960507964

Email id :- mohitep2704@gmail.com

linkedin :- Prashant Mohite

Project Introduction

Here I implement CICD for apache2 deployment

I have image.html , index.html on local server , If I push these 2 file on GitHub develop branch then deployment testing happened on test server

In that basically apache2 is deployed and default apache2 is replaced by custom image which I mentioned above.

If I push these 2 file on GitHub master branch then testing on test server and and application moved to production

Tools used for project

Ansible : -

Ansible used to do installation of Jenkins, for that I created playbook in which

Called master.sh for Jenkins installation on master server and slave.sh for slaves

GitHub :-

Here is created one public repo and on that repo I am pushing code develop and master branch

GitHub webhook are created for triggering Jenkins project

Jenkins

In Jenkins I created 3 freestyle project

Job1 :-

When code is pushed to GitHub develop branch then job1 is triggered and start testing on test server

Job2 :-

When code is pushed to GitHub master branch then start testing on test server

Job 3 :-

When job2 successfully completed then job3 is triggered then deployment happens on Prod server

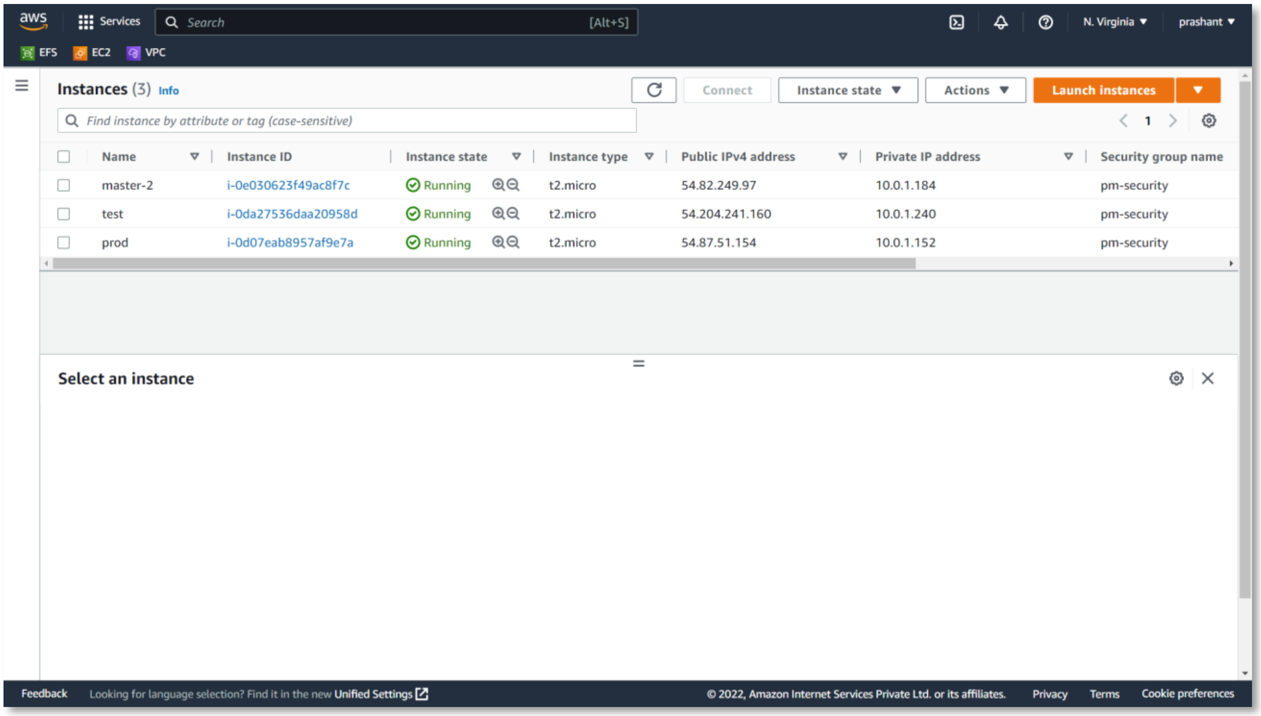
Docker :

Docker is used to containerize the GitHub code

Dockerfile should bebuilt every time there is a push to Git-Hub.

I created A Docker-file for containerization , in Jenkins project there is execute shell section in that I wrote a docker command for building docker image and to run the container so that our apache2 is deployed.

Created three ec2 instance on aws , 1 master and 2 slaves.


aws
Services
Search
[Alt+S]
N. Virginia ▼
prashant ▼
EFS
EC2
VPC
Connect
Instance state ▼
Actions ▼
Launch instances
Find instance by attribute or tag (case-sensitive)
1
Select an instance
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences

Instances (3) Info

□	Name	Instance ID	Instance state	Instance type	Public IPv4 address	Private IP address	Security group name
□	master-2	i-0e030623f49ac8f7c	✓ Running	t2.micro	54.82.249.97	10.0.1.184	pm-security
□	test	i-0da27536daa20958d	✓ Running	t2.micro	54.204.241.160	10.0.1.240	pm-security
□	prod	i-0d07eab8957af9e7a	✓ Running	t2.micro	54.87.51.154	10.0.1.152	pm-security

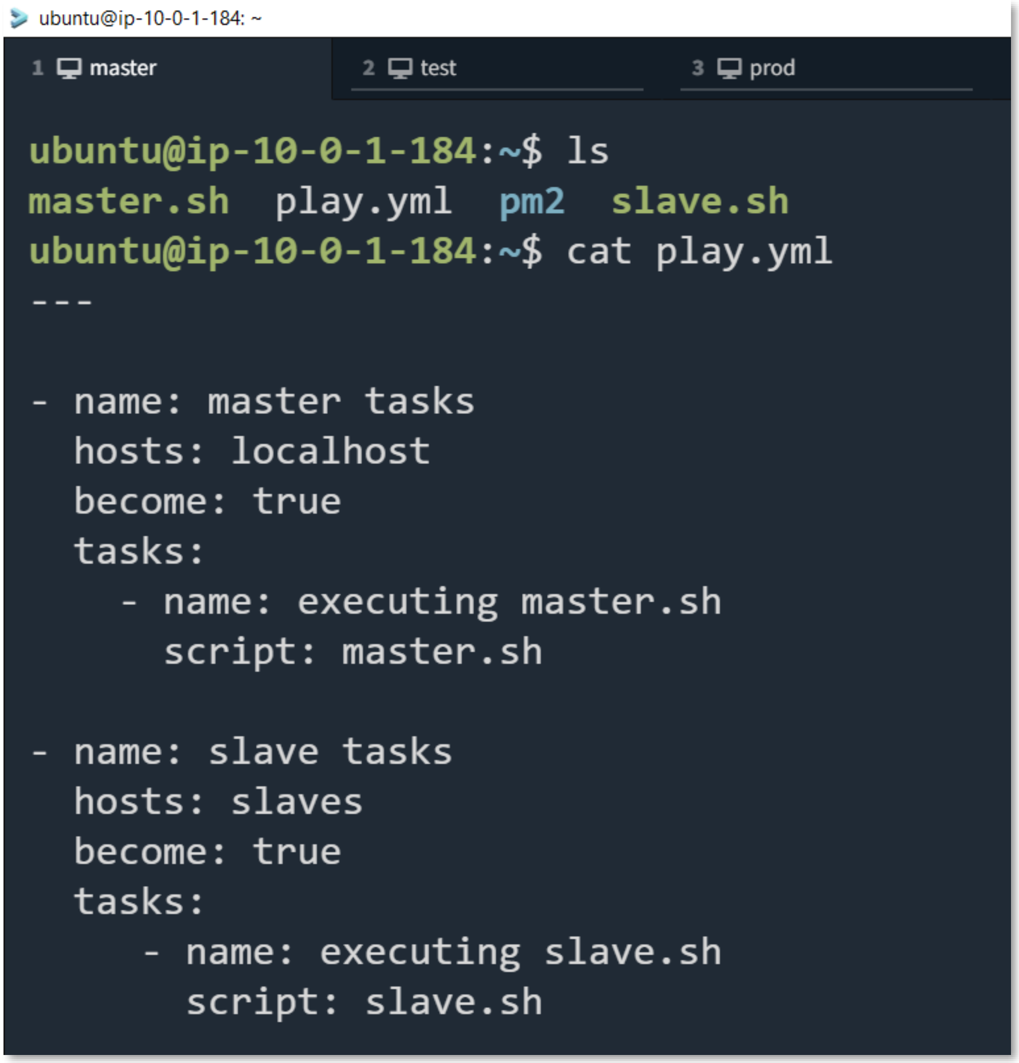
On master – installed ansible and write play.yml file run playbook

in playbook some task are there

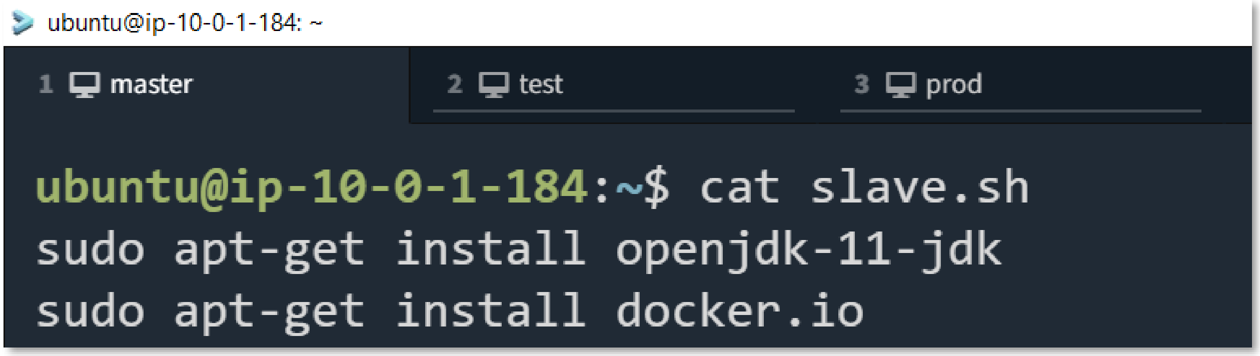
task1 for installing Jenkins on master by using master.sh scrip

in task2 for installing java and docker on test and prod using slave.sh script

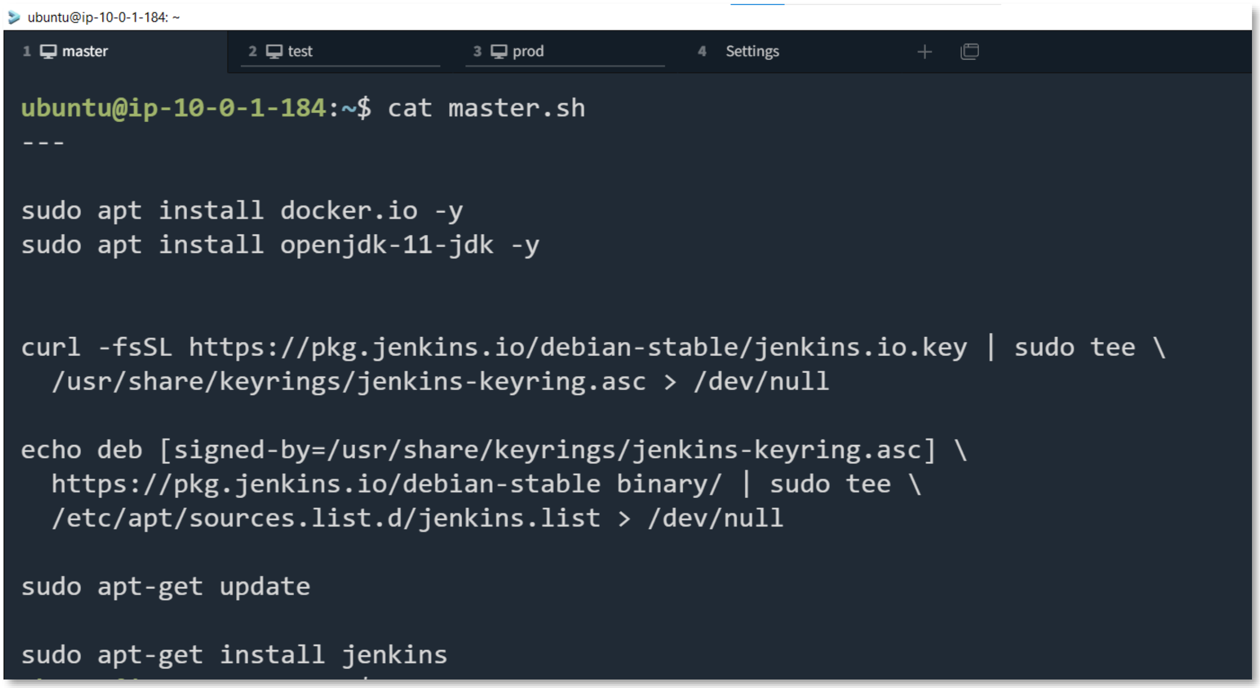
Playbooks and script written for installation



```
ubuntu@ip-10-0-1-184:~$ ls
master.sh  play.yml  pm2  slave.sh
ubuntu@ip-10-0-1-184:~$ cat play.yml
---

- name: master tasks
  hosts: localhost
  become: true
  tasks:
    - name: executing master.sh
      script: master.sh

- name: slave tasks
  hosts: slaves
  become: true
  tasks:
     - name: executing slave.sh
       script: slave.sh
```

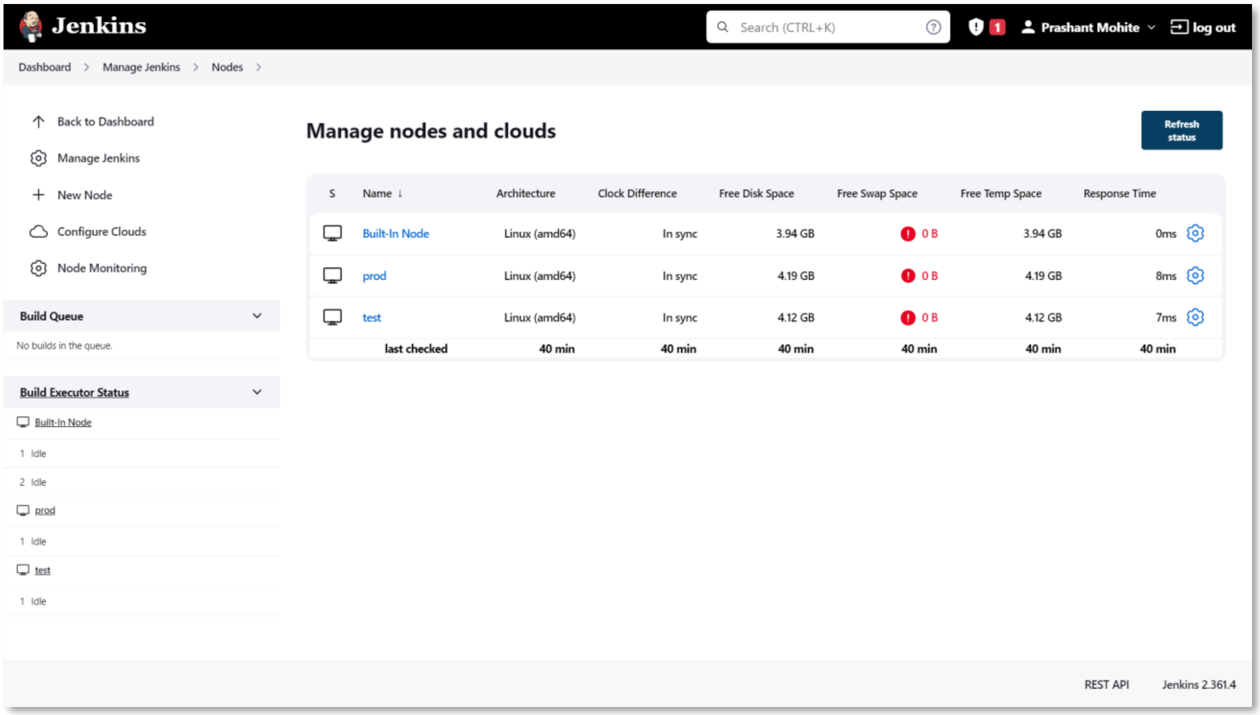


```
ubuntu@ip-10-0-1-184:~$ cat slave.sh
sudo apt-get install openjdk-11-jdk
sudo apt-get install docker.io
```



```
ubuntu@ip-10-0-1-184:~$ cat master.sh
---

sudo apt install docker.io -y
sudo apt install openjdk-11-jdk -y


curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo tee \
  /usr/share/keyrings/jenkins-keyring.asc > /dev/null

echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null

sudo apt-get update

sudo apt-get install jenkins
```

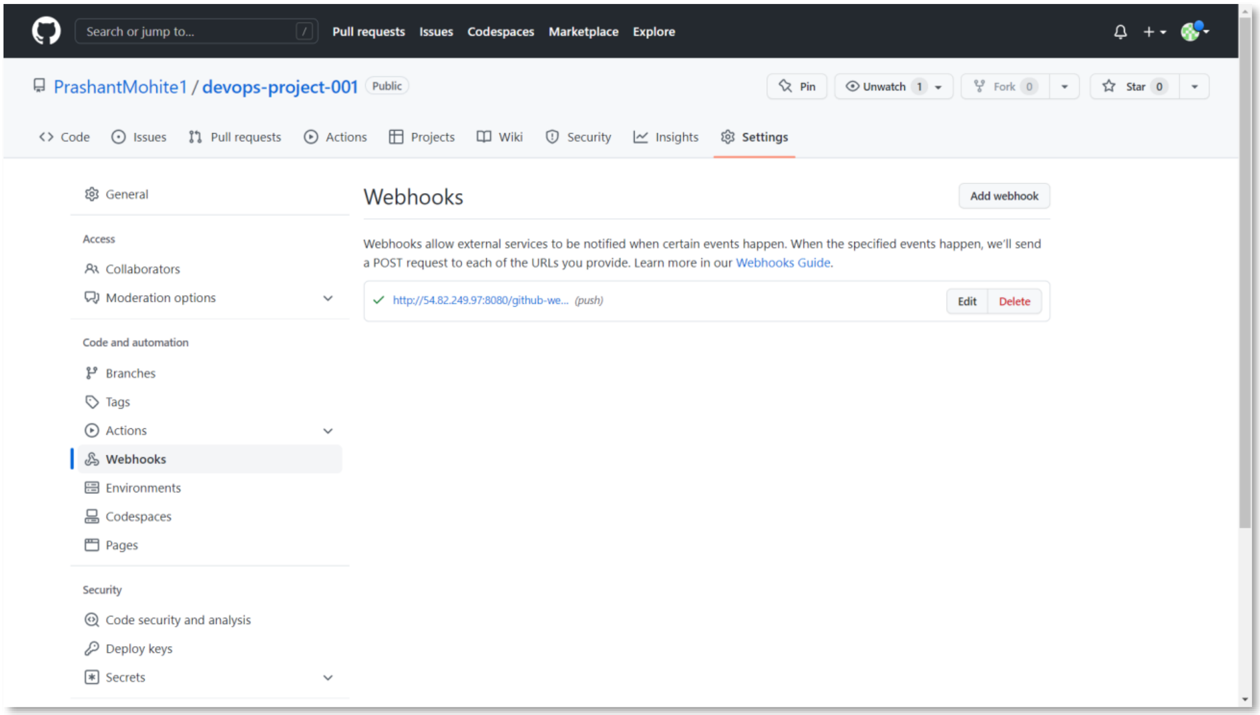
Added two nodes test and prod in jenkins



Manage nodes and clouds

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Linux (amd64)	In sync	3.94 GB	! 0 B	3.94 GB	0ms
	prod	Linux (amd64)	In sync	4.19 GB	! 0 B	4.19 GB	8ms
	test	Linux (amd64)	In sync	4.12 GB	! 0 B	4.12 GB	7ms
	last checked	40 min	40 min	40 min	40 min	40 min	40 min

Created webhooks in github for triggering Jenkins

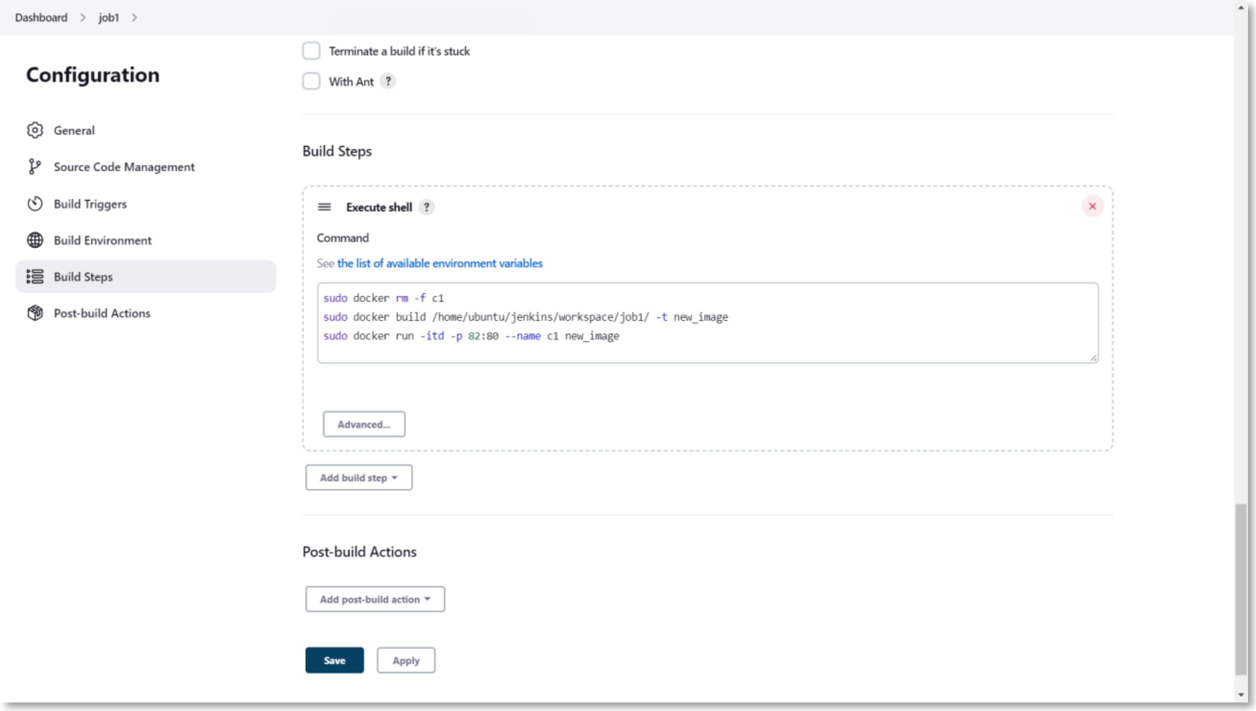


Webhooks

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our Webhooks Guide.

✓ http://54.82.249.97:8080/github-we... (push)

Created job 1 and configured it



☐ Terminate a build if it's stuck
☐ With Ant ?

Build Steps

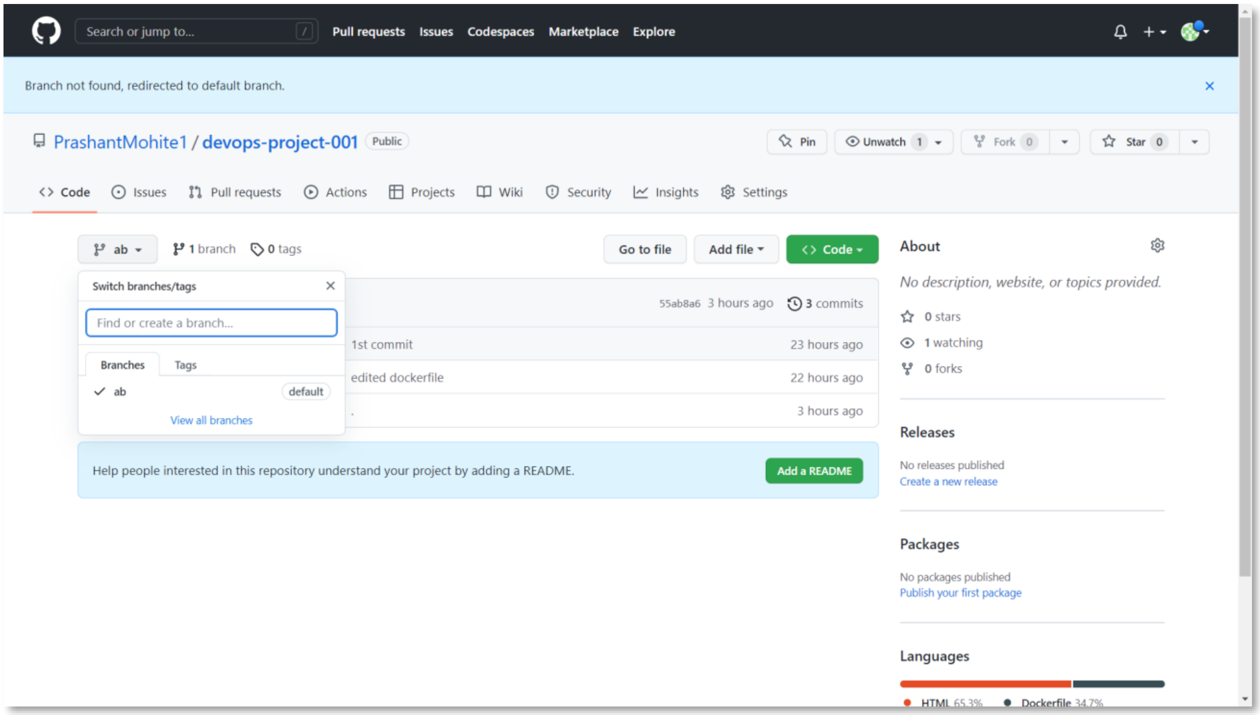
Execute shell ?

Command

See the list of available environment variables

```
sudo docker rm -f c1
sudo docker build /home/ubuntu/jenkins/workspace/job1/ -t new_image
sudo docker run -itd -p 82:80 --name c1 new_image
```

Before pushing code to github see below picture there is no develop branch



Branch not found, redirected to default branch.

55ab8a6 3 hours ago 3 commits

1st commit 23 hours ago
edited dockerfile 22 hours ago
. 3 hours ago

Help people interested in this repository understand your project by adding a README.

About

No description, website, or topics provided.

0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

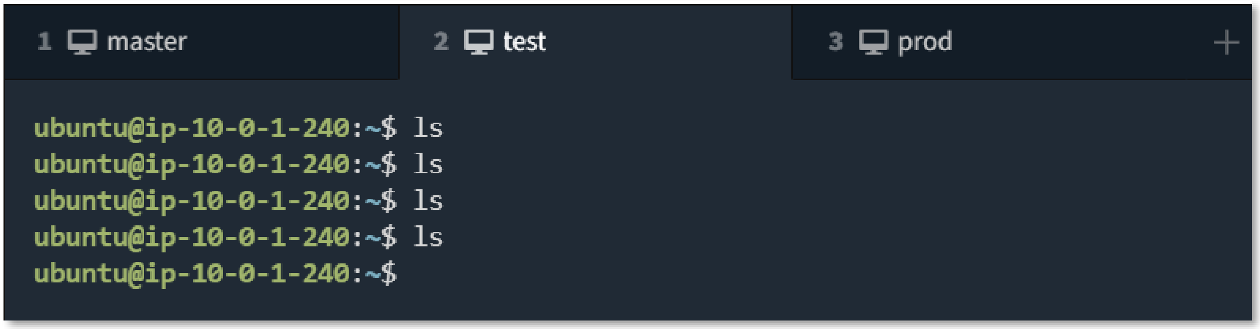
Packages

No packages published
Publish your first package

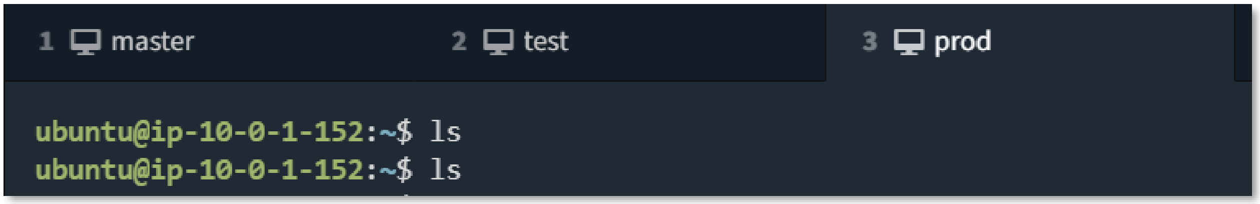
Languages

● HTML 65.3% ● Dockerfile 34.7%

Before push

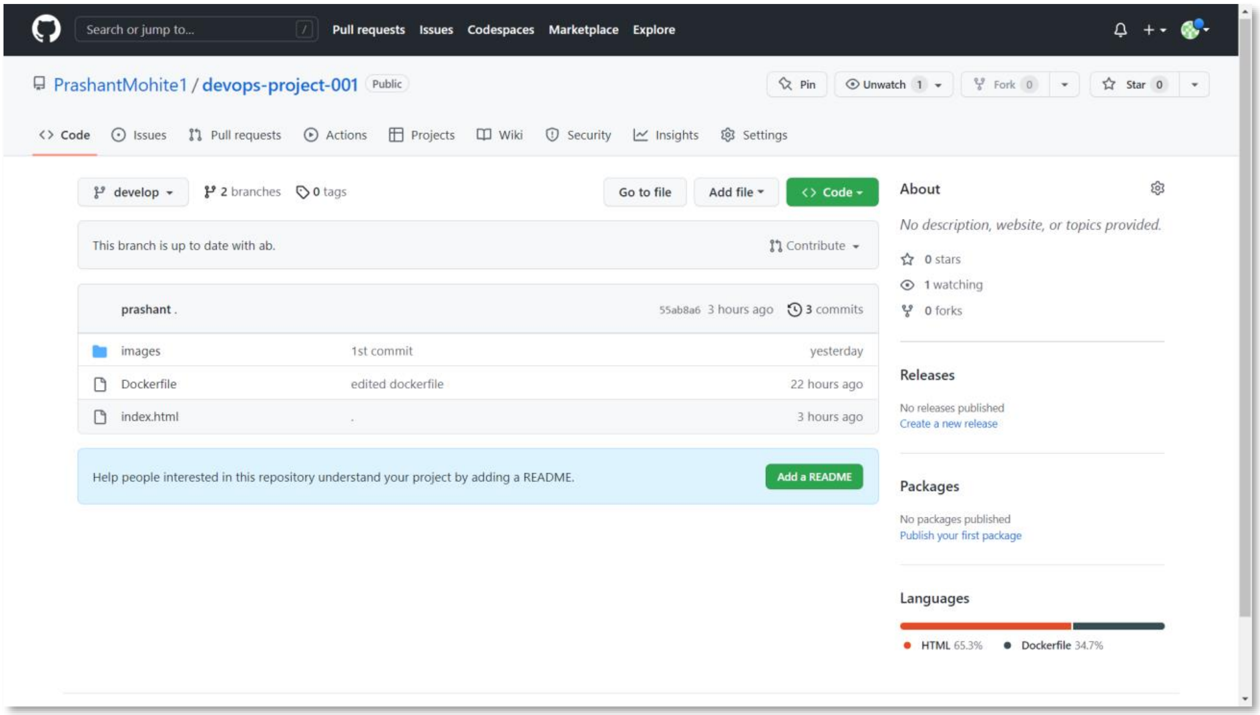


```
ubuntu@ip-10-0-1-240:~$ ls
ubuntu@ip-10-0-1-240:~$ ls
ubuntu@ip-10-0-1-240:~$ ls
ubuntu@ip-10-0-1-240:~$ ls
ubuntu@ip-10-0-1-240:~$
```



```
ubuntu@ip-10-0-1-152:~$ ls
ubuntu@ip-10-0-1-152:~$ ls
```

After code pushed to develop branch



PrashantMohite1 / devops-project-001 Public

2 branches 0 tags

This branch is up to date with ab.

prashant . 55ab8a6 3 hours ago 3 commits

images 1st commit yesterday
Dockerfile edited dockerfile 22 hours ago
index.html . 3 hours ago

Help people interested in this repository understand your project by adding a README.

About

No description, website, or topics provided.

0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

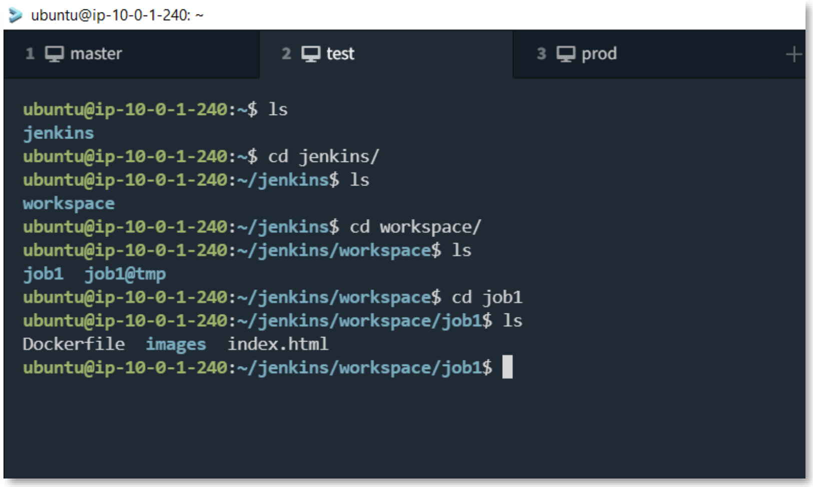
Packages

No packages published
Publish your first package

Languages

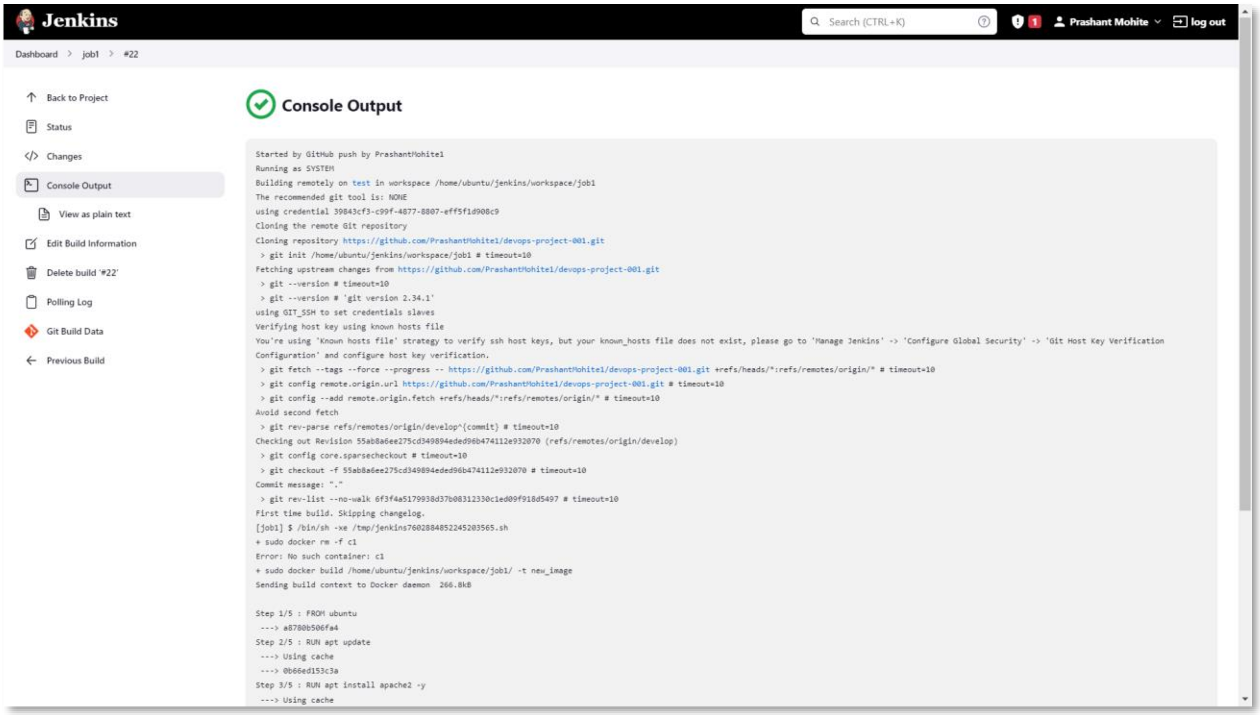
● HTML 65.3% ● Dockerfile 34.7%

New folder which specified as /home/ubuntu/jenkins while creating test node



```
ubuntu@ip-10-0-1-240:~$ ls
jenkins
ubuntu@ip-10-0-1-240:~$ cd jenkins/
ubuntu@ip-10-0-1-240:~/jenkins$ ls
workspace
ubuntu@ip-10-0-1-240:~/jenkins$ cd workspace/
ubuntu@ip-10-0-1-240:~/jenkins/workspace$ ls
job1  job1@tmp
ubuntu@ip-10-0-1-240:~/jenkins/workspace$ cd job1
ubuntu@ip-10-0-1-240:~/jenkins/workspace/job1$ ls
Dockerfile  images  index.html
ubuntu@ip-10-0-1-240:~/jenkins/workspace/job1$ 
```

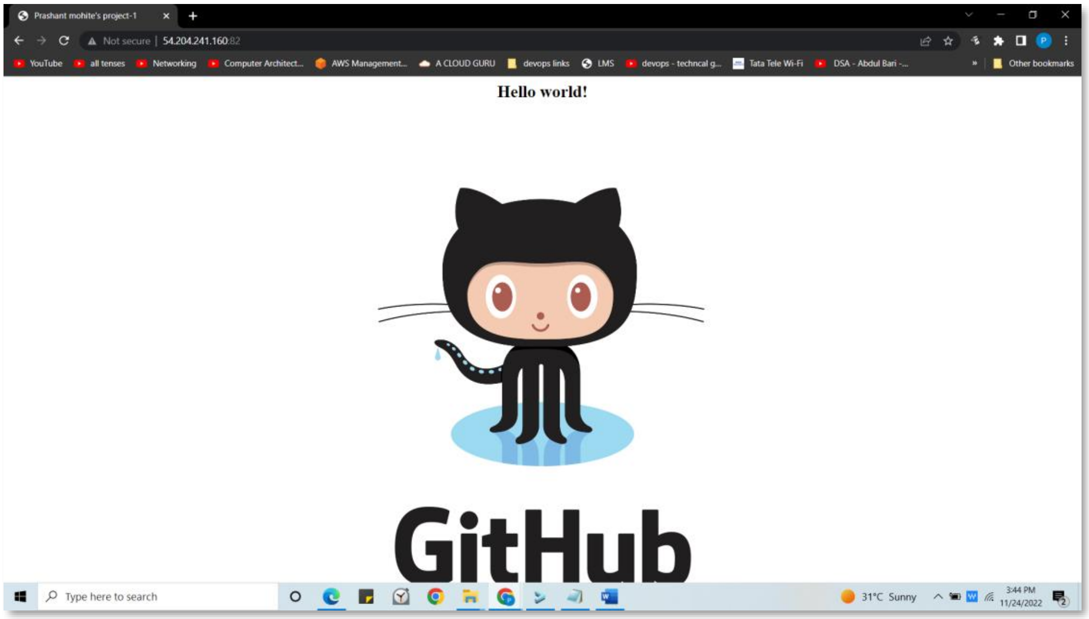
After pushing product (image , index.html , Docker file) to GitHub the job1 is triggered



Console Output

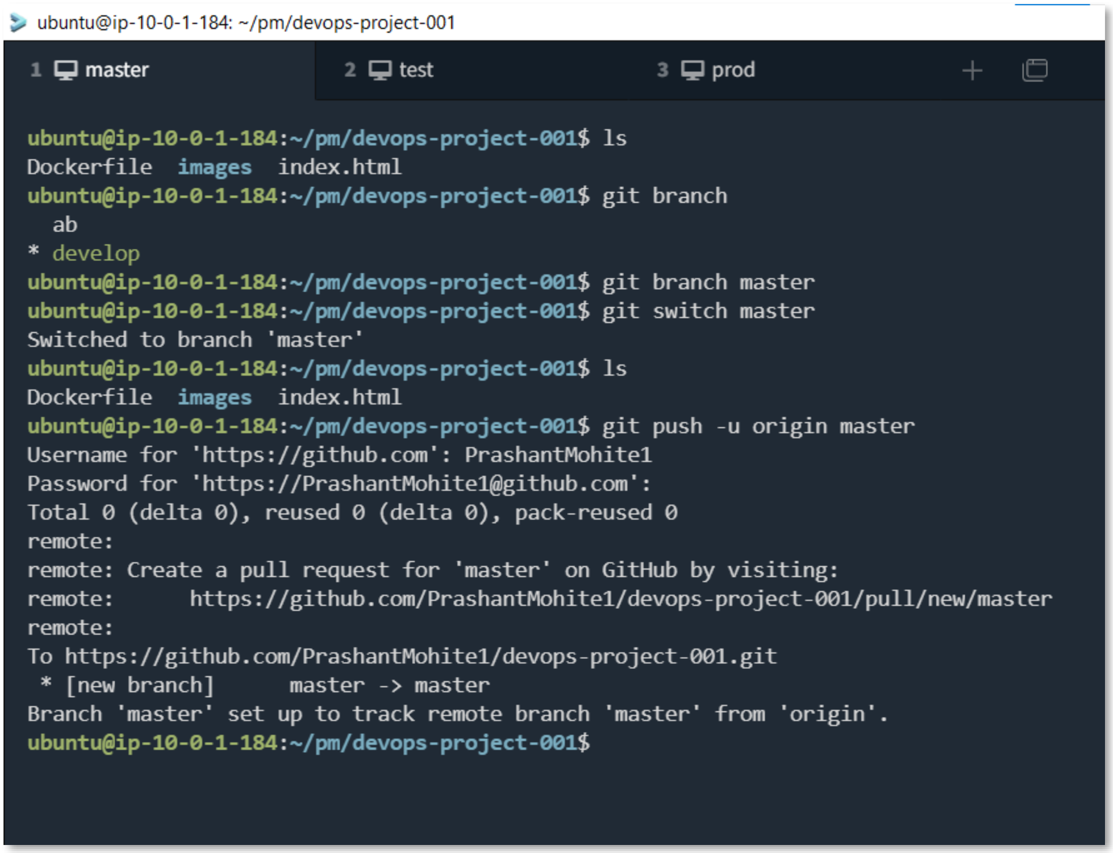
```
Started by GitHub push by PrashantMohite1
Running as SYSTEM
Building remotely on test in workspace /home/ubuntu/jenkins/workspace/job1
The recommended git tool is: NONE
using credential 39843cf3-c99f-4877-8807-eff5f1d908c9
Cloning the remote Git repository
Cloning repository https://github.com/PrashantMohite1/devops-project-001.git
 > git init /home/ubuntu/jenkins/workspace/job1 # timeout=10
Fetching upstream changes from https://github.com/PrashantMohite1/devops-project-001.git
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_SSH to set credentials slaves
Verifying host key using known hosts file
You're using 'Known hosts file' strategy to verify ssh host keys, but your known_hosts file does not exist, please go to 'Manage Jenkins' -> 'Configure Global Security' -> 'Git Host Key Verification
Configuration' and configure host key verification.
 > git fetch --tags --force --progress -- https://github.com/PrashantMohite1/devops-project-001.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git config remote.origin.url https://github.com/PrashantMohite1/devops-project-001.git # timeout=10
 > git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
 > git rev-parse refs/remotes/origin/develop^{commit} # timeout=10
Checking out Revision 55ab8a6ee275cd349894eded96b474112e932070 (refs/remotes/origin/develop)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 55ab8a6ee275cd349894eded96b474112e932070 # timeout=10
Commit message: "."
 > git rev-list --no-walk 6f3f4a5179938d37b08312330c1ed09f918d5497 # timeout=10
First time build. Skipping changelog.
[job1] $ /bin/sh -xe /tmp/jenkins7602884852245203565.sh
+ sudo docker rm -f c1
Error: No such container: c1
+ sudo docker build /home/ubuntu/jenkins/workspace/job1/ -t new_image
Sending build context to Docker daemon  266.8kB

Step 1/5 : FROM ubuntu
 ---> a8780b506fa4
Step 2/5 : RUN apt update
 ---> Using cache
 ---> 0b66ed153c3a
Step 3/5 : RUN apt install apache2 -y
 ---> Using cache
```

After job1 succesfully build the job1 then apache2 is installed and export at port 82 and also doned apache2 configuration which replace the actual index.html with our custom image and index.html

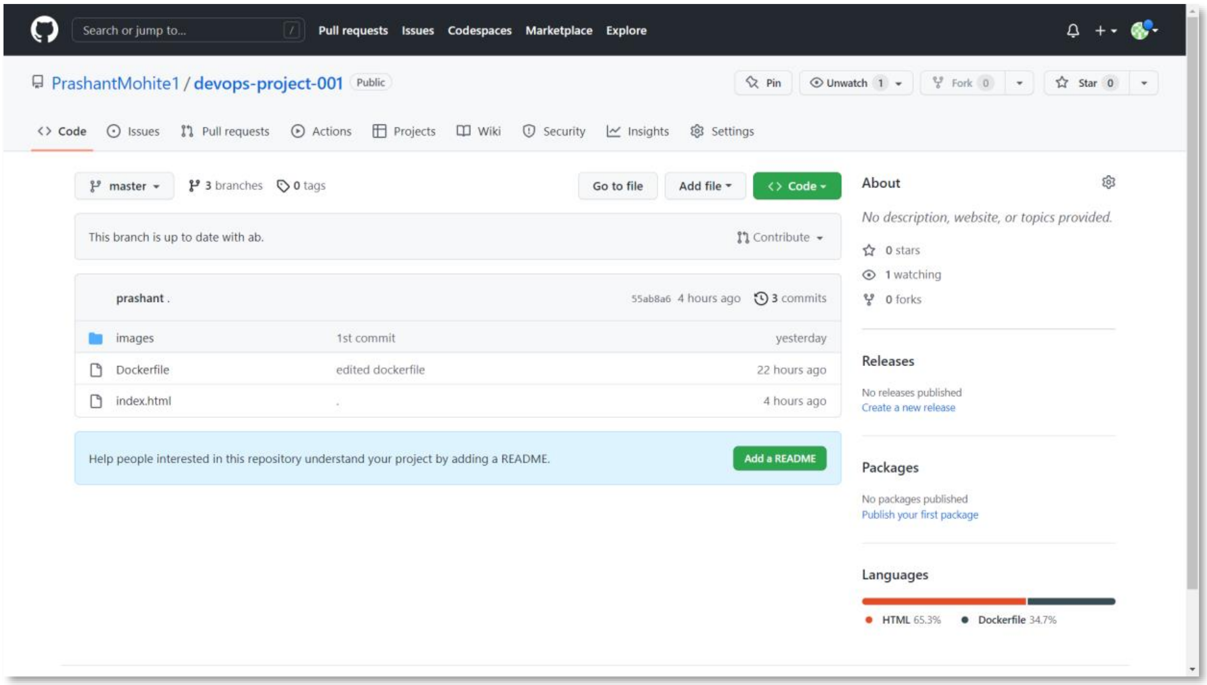


Hello world!

Code pushed to master branch



```
ubuntu@ip-10-0-1-184:~/pm/devops-project-001$ ls
Dockerfile  images  index.html
ubuntu@ip-10-0-1-184:~/pm/devops-project-001$ git branch
  ab
* develop
ubuntu@ip-10-0-1-184:~/pm/devops-project-001$ git branch master
ubuntu@ip-10-0-1-184:~/pm/devops-project-001$ git switch master
Switched to branch 'master'
ubuntu@ip-10-0-1-184:~/pm/devops-project-001$ ls
Dockerfile  images  index.html
ubuntu@ip-10-0-1-184:~/pm/devops-project-001$ git push -u origin master
Username for 'https://github.com': PrashantMohite1
Password for 'https://PrashantMohite1@github.com':
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:      https://github.com/PrashantMohite1/devops-project-001/pull/new/master
remote:
To https://github.com/PrashantMohite1/devops-project-001.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
ubuntu@ip-10-0-1-184:~/pm/devops-project-001$
```



PrashantMohite1 / devops-project-001 Public

3 branches 0 tags

This branch is up to date with ab.

prashant . 55ab8a6 4 hours ago 3 commits

images	1st commit	yesterday
Dockerfile	edited dockerfile	22 hours ago
index.html	.	4 hours ago

Help people interested in this repository understand your project by adding a README.

About

No description, website, or topics provided.

0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

● HTML 65.3% ● Dockerfile 34.7%

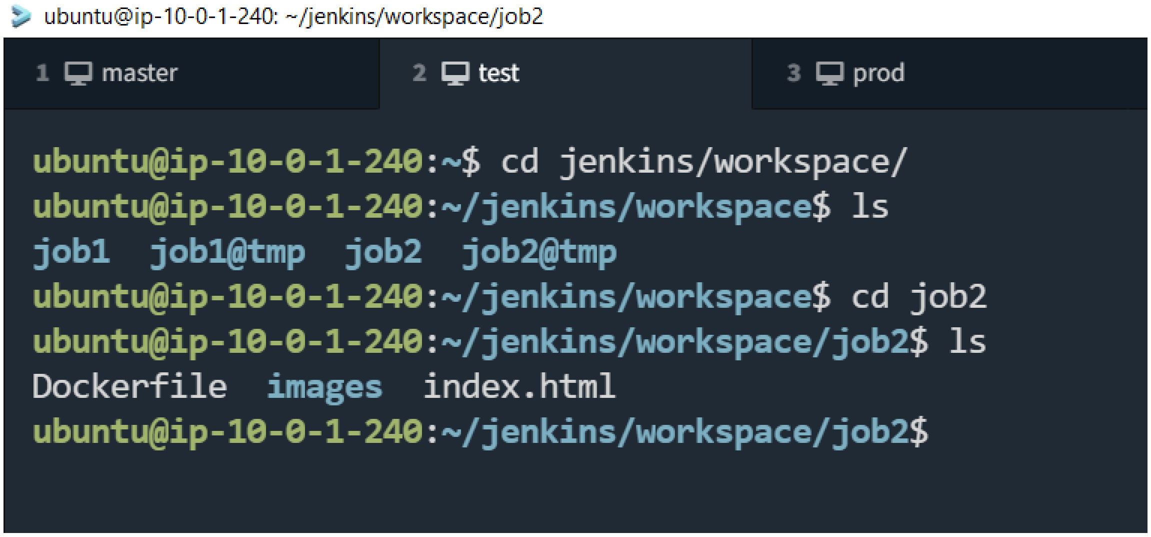
Job 2

job2 is triggered and code pushed to master branch , and after successfully building job2 then job3 is triggered

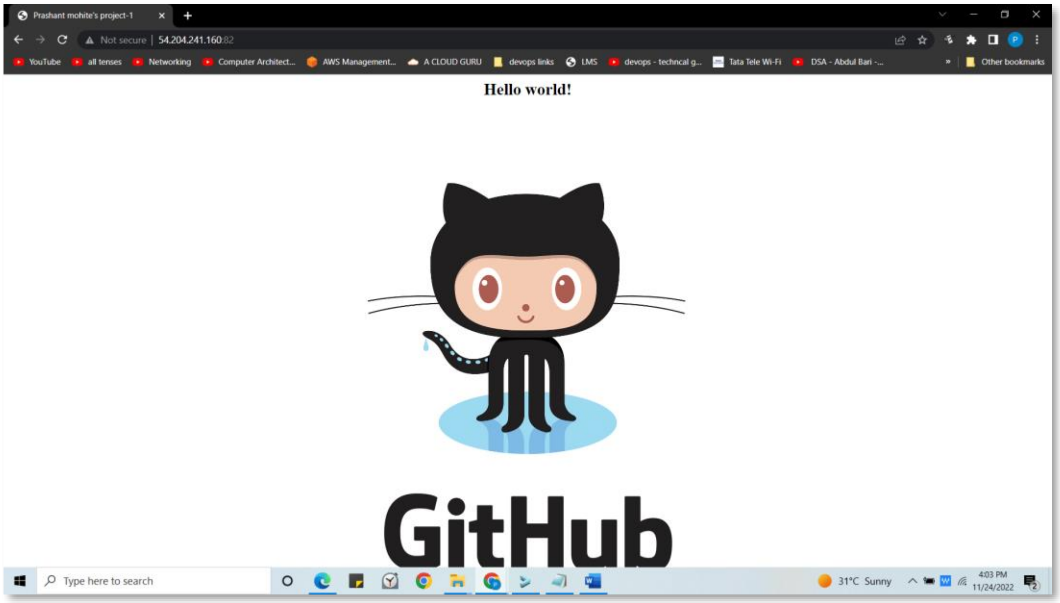
After job2 build

In directory home/ubuntu/Jenkins/workspace we get our product and from these the container image is created and finally runned a container

Like here in job2 we again do testing before move to production

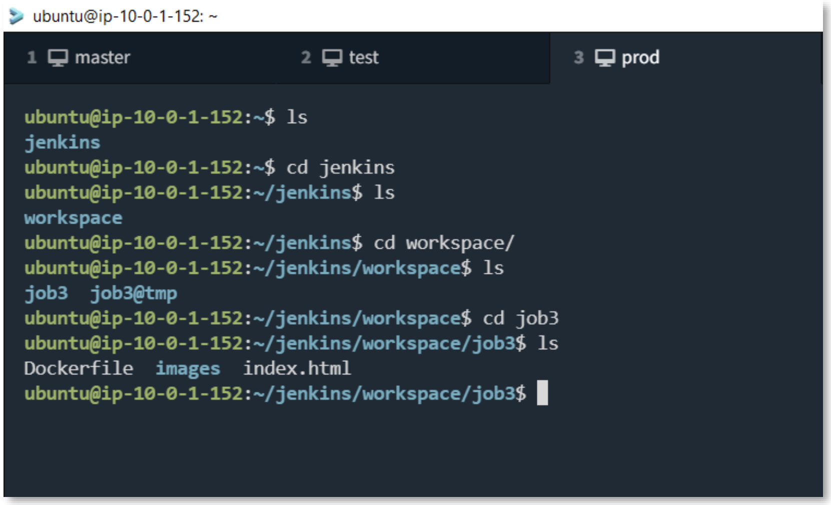


```
ubuntu@ip-10-0-1-240:~$ cd jenkins/workspace/
ubuntu@ip-10-0-1-240:~/jenkins/workspace$ ls
job1  job1@tmp  job2  job2@tmp
ubuntu@ip-10-0-1-240:~/jenkins/workspace$ cd job2
ubuntu@ip-10-0-1-240:~/jenkins/workspace/job2$ ls
Dockerfile  images  index.html
ubuntu@ip-10-0-1-240:~/jenkins/workspace/job2$
```

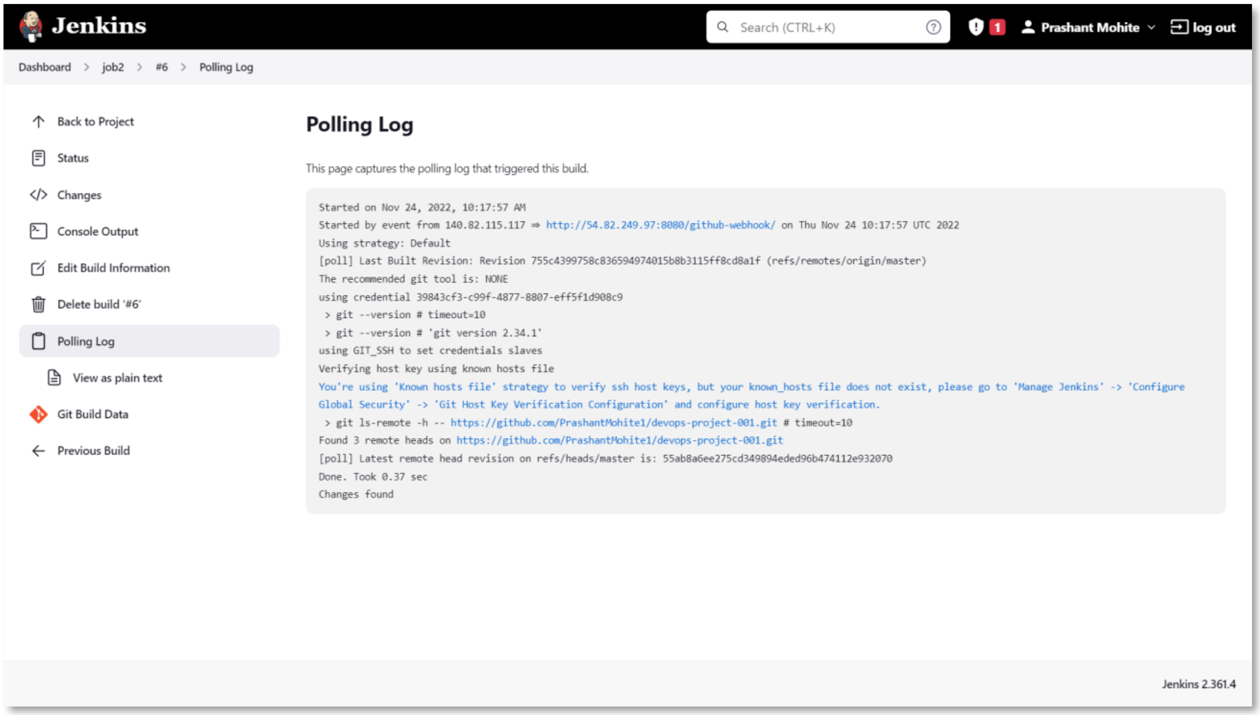


Hello world!

Job 3



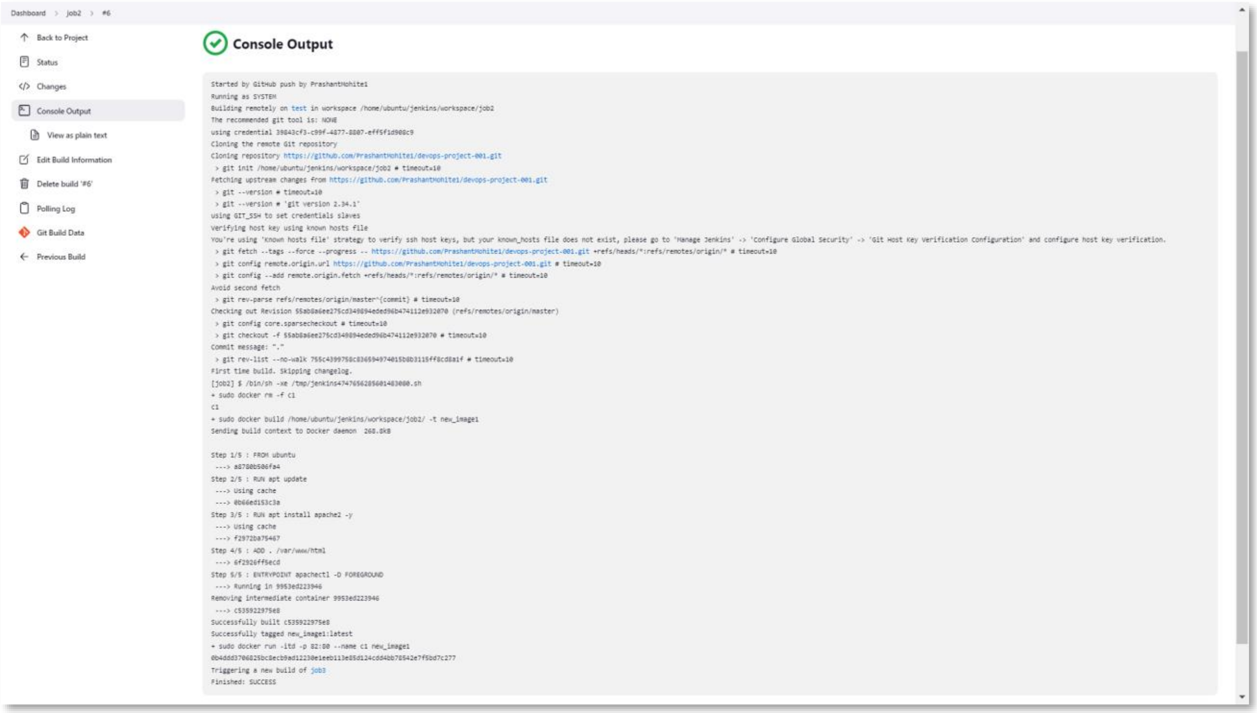
```
ubuntu@ip-10-0-1-152:~$ ls
jenkins
ubuntu@ip-10-0-1-152:~$ cd jenkins
ubuntu@ip-10-0-1-152:~/jenkins$ ls
workspace
ubuntu@ip-10-0-1-152:~/jenkins$ cd workspace/
ubuntu@ip-10-0-1-152:~/jenkins/workspace$ ls
job3  job3@tmp
ubuntu@ip-10-0-1-152:~/jenkins/workspace$ cd job3
ubuntu@ip-10-0-1-152:~/jenkins/workspace/job3$ ls
Dockerfile  images  index.html
ubuntu@ip-10-0-1-152:~/jenkins/workspace/job3$
```



Polling Log

This page captures the polling log that triggered this build.

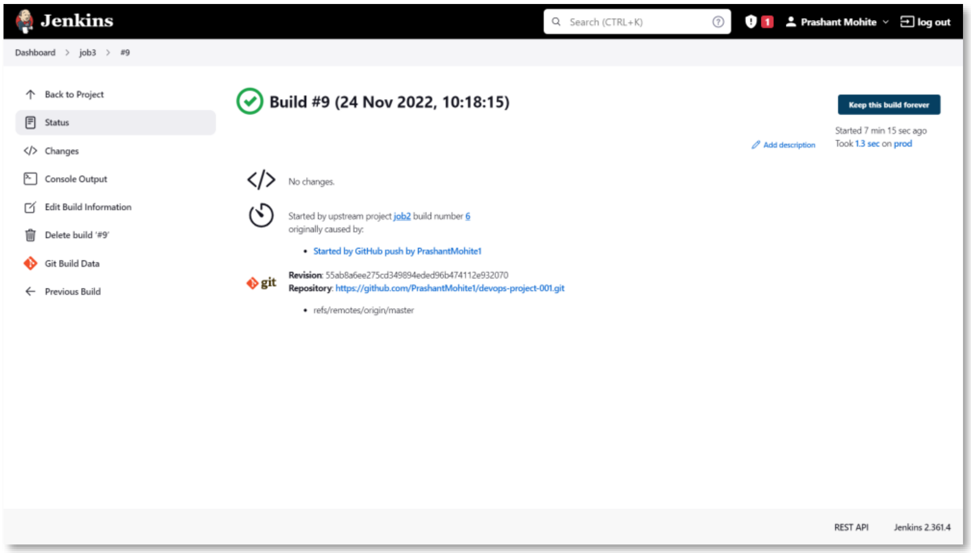
```
Started on Nov 24, 2022, 10:17:57 AM
Started by event from 140.82.115.117 ⇒ http://54.82.249.97:8080/github-webhook/ on Thu Nov 24 10:17:57 UTC 2022
Using strategy: Default
[poll] Last Built Revision: Revision 755c4399758c836594974015b8b3115ff8cd8a1f (refs/remotes/origin/master)
The recommended git tool is: NONE
using credential 39843cf3-c99f-4877-8807-eff5f1d908c9
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_SSH to set credentials slaves
Verifying host key using known hosts file
You're using 'Known hosts file' strategy to verify ssh host keys, but your known_hosts file does not exist, please go to 'Manage Jenkins' -> 'Configure
Global Security' -> 'Git Host Key Verification Configuration' and configure host key verification.
 > git ls-remote -h -- https://github.com/PrashantMohite1/devops-project-001.git # timeout=10
Found 3 remote heads on https://github.com/PrashantMohite1/devops-project-001.git
[poll] Latest remote head revision on refs/heads/master is: 55ab8a6ee275cd349894eded96b474112e932070
Done. Took 0.37 sec
Changes found
```



Console Output

```
Started by GitHub push by PrashantMohite1
Running as SYSTEM
Building remotely on test in workspace /home/ubuntu/jenkins/workspace/job2
The recommended git tool is: NONE
using credential 39843cf3-c99f-4877-8807-eff5f1d908c9
Cloning the remote Git repository
Cloning repository https://github.com/PrashantMohite1/devops-project-001.git
 > git init /home/ubuntu/jenkins/workspace/job2 # timeout=10
Fetching upstream changes from https://github.com/PrashantMohite1/devops-project-001.git
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_SSH to set credentials slaves
Verifying host key using known hosts file
You're using 'Known hosts file' strategy to verify ssh host keys, but your known_hosts file does not exist, please go to 'Manage Jenkins' -> 'Configure Global Security' -> 'Git Host Key Verification Configuration' and configure host key verification.
 > git fetch --tags --force --progress -- https://github.com/PrashantMohite1/devops-project-001.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git config remote.origin.url https://github.com/PrashantMohite1/devops-project-001.git # timeout=10
 > git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 55ab8a6ee275cd349894eded96b474112e932070 (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 55ab8a6ee275cd349894eded96b474112e932070 # timeout=10
Commit message: "."
 > git rev-list --no-walk 755c4399758c836594974015b8b3115ff8cd8a1f # timeout=10
First time build. Skipping changelog.
[job2] $ /bin/sh -xe /tmp/jenkins4747656285601483080.sh
+ sudo docker rm -f c1
c1
+ sudo docker build /home/ubuntu/jenkins/workspace/job2/ -t new_image1
Sending build context to Docker daemon  268.8kB

Step 1/5 : FROM ubuntu
 ---> a8780b506fa4
Step 2/5 : RUN apt update
 ---> Using cache
 ---> 0b66ed153c3a
Step 3/5 : RUN apt install apache2 -y
 ---> Using cache
 ---> f2972ba75467
Step 4/5 : ADD . /var/www/html
 ---> 6f2926ff5ecd
Step 5/5 : ENTRYPOINT apachectl -D FOREGROUND
 ---> Running in 9953ed223946
Removing intermediate container 9953ed223946
 ---> c535922975e8
Successfully built c535922975e8
Successfully tagged new_image1:latest
+ sudo docker run -itd -p 82:80 --name c1 new_image1
0b4ddd3706825bc8ecb9ad12230e1eeb113e85d124cdd4bb78542e7f5bd7c277
Triggering a new build of job3
Finished: SUCCESS
```

After completing job2 then triggered job 3



Build #9 (24 Nov 2022, 10:18:15)

Started 7 min 15 sec ago
Took 1.3 sec on prod

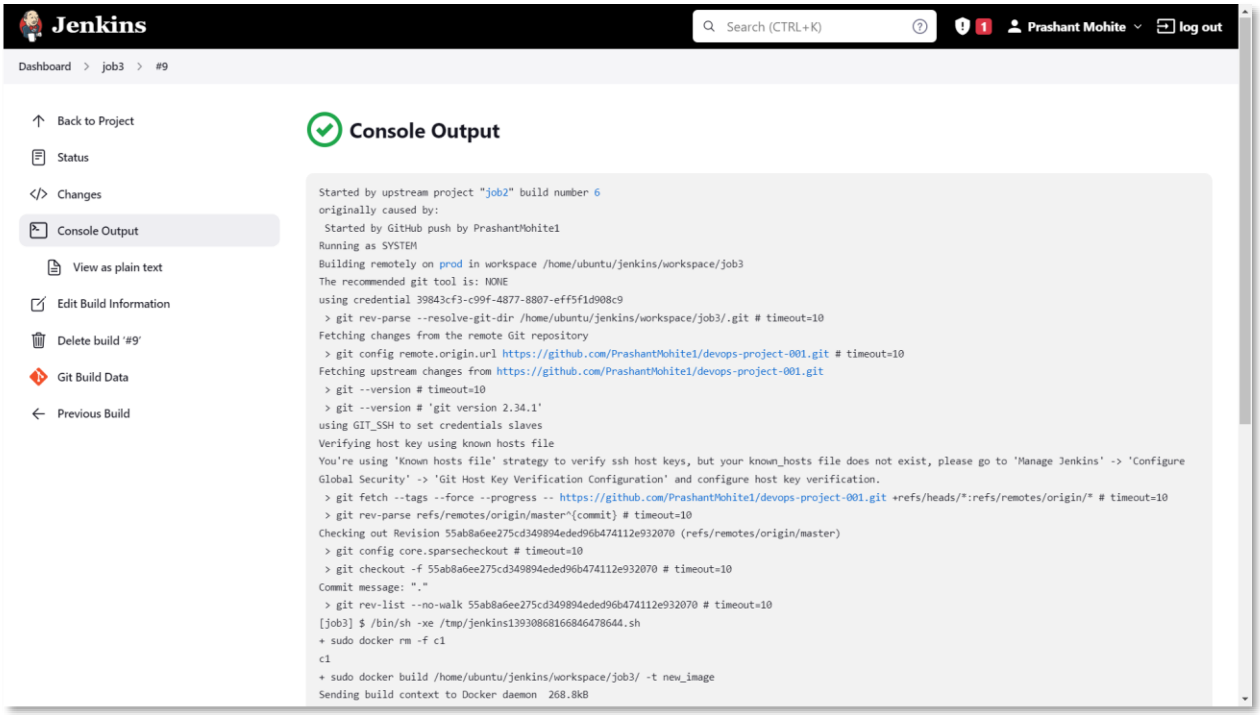
No changes.

Started by upstream project job2 build number 6
originally caused by:

- Started by GitHub push by PrashantMohite1

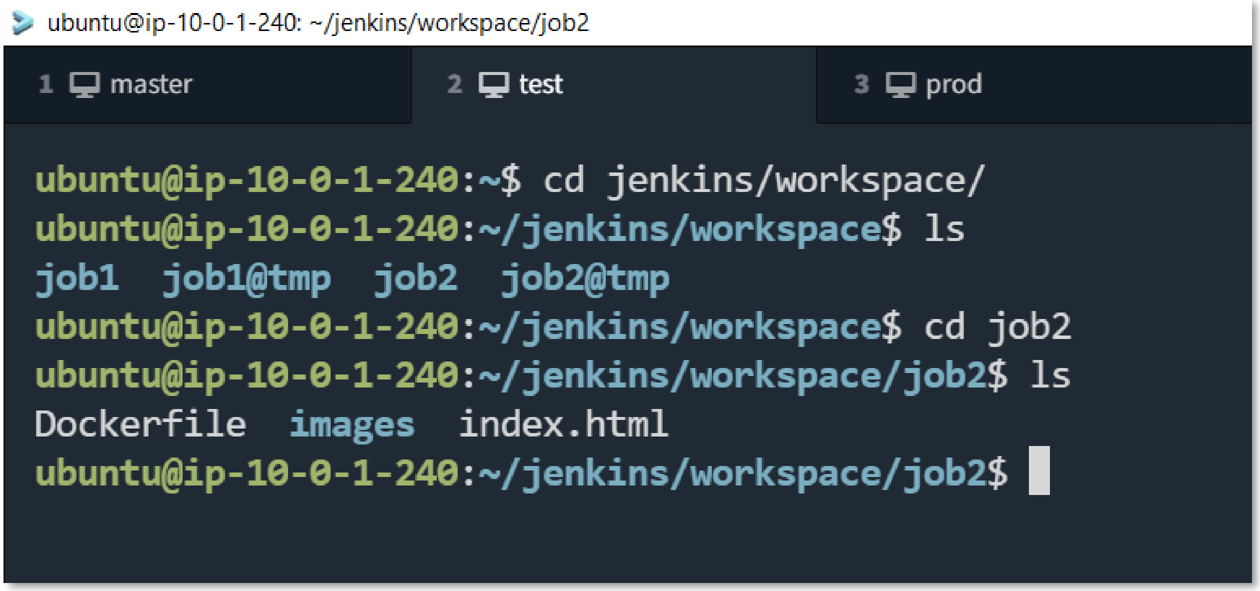
Revision: 55ab8a6ee275cd349894eded96b474112e932070
Repository: https://github.com/PrashantMohite1/devops-project-001.git

- refs/remotes/origin/master

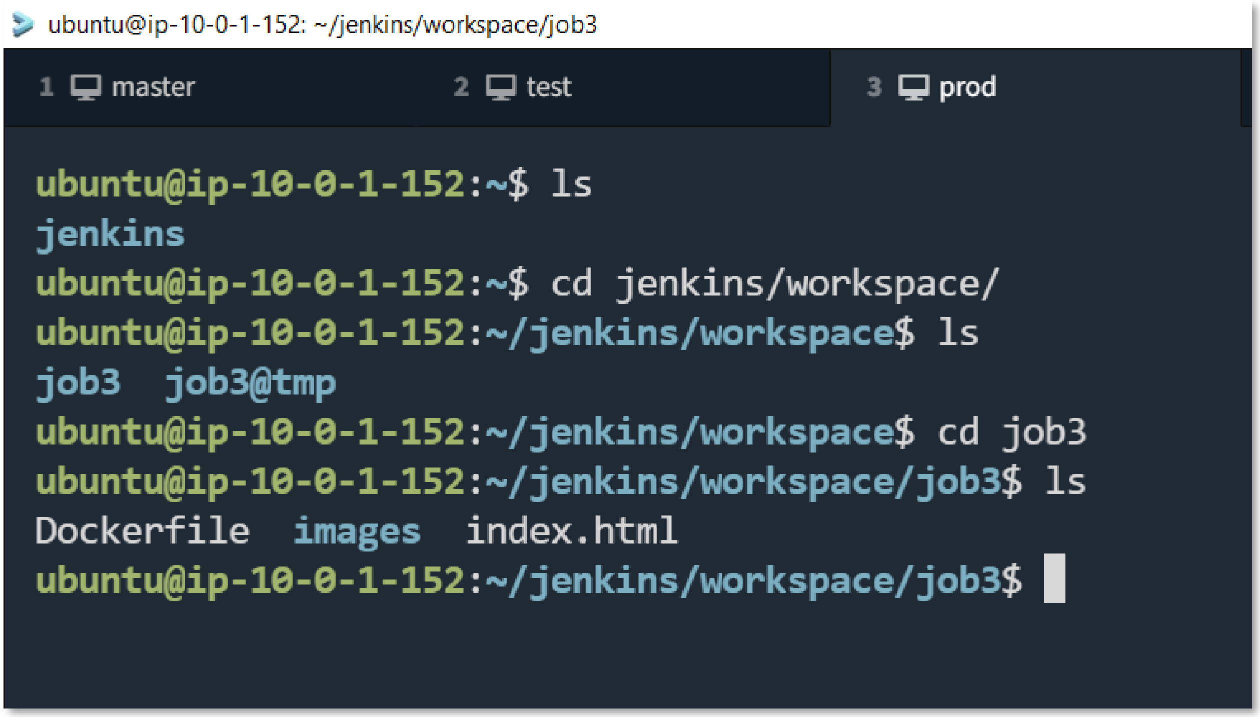


Console Output

```
Started by upstream project "job2" build number 6
originally caused by:
 Started by GitHub push by PrashantMohite1
Running as SYSTEM
Building remotely on prod in workspace /home/ubuntu/jenkins/workspace/job3
The recommended git tool is: NONE
using credential 39843cf3-c99f-4877-8807-eff5f1d908c9
 > git rev-parse --resolve-git-dir /home/ubuntu/jenkins/workspace/job3/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/PrashantMohite1/devops-project-001.git # timeout=10
Fetching upstream changes from https://github.com/PrashantMohite1/devops-project-001.git
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_SSH to set credentials slaves
Verifying host key using known hosts file
You're using 'Known hosts file' strategy to verify ssh host keys, but your known_hosts file does not exist, please go to 'Manage Jenkins' -> 'Configure
Global Security' -> 'Git Host Key Verification Configuration' and configure host key verification.
 > git fetch --tags --force --progress -- https://github.com/PrashantMohite1/devops-project-001.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 55ab8a6ee275cd349894eded96b474112e932070 (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 55ab8a6ee275cd349894eded96b474112e932070 # timeout=10
Commit message: "."
 > git rev-list --no-walk 55ab8a6ee275cd349894eded96b474112e932070 # timeout=10
[job3] $ /bin/sh -xe /tmp/jenkins13930868166846478644.sh
+ sudo docker rm -f c1
c1
+ sudo docker build /home/ubuntu/jenkins/workspace/job3/ -t new_image
Sending build context to Docker daemon  268.8kB
```

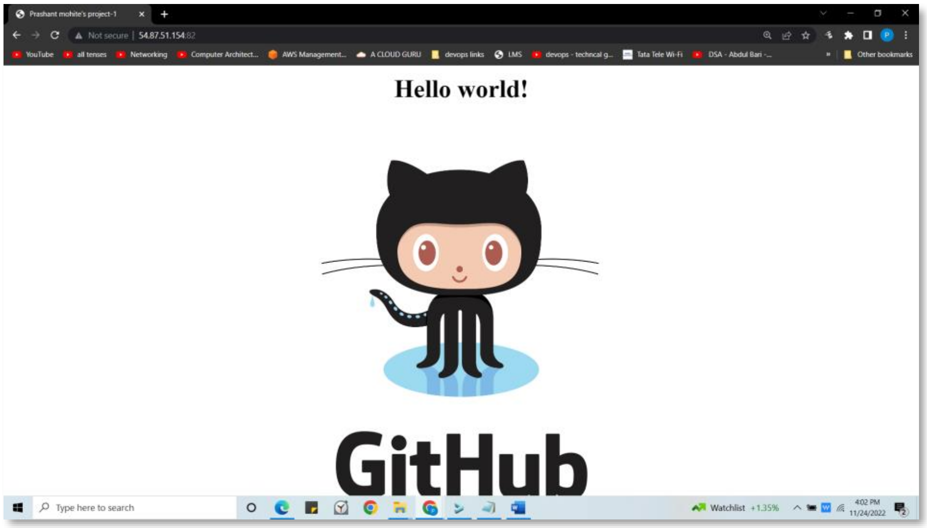


```
ubuntu@ip-10-0-1-240:~$ cd jenkins/workspace/
ubuntu@ip-10-0-1-240:~/jenkins/workspace$ ls
job1  job1@tmp  job2  job2@tmp
ubuntu@ip-10-0-1-240:~/jenkins/workspace$ cd job2
ubuntu@ip-10-0-1-240:~/jenkins/workspace/job2$ ls
Dockerfile  images  index.html
ubuntu@ip-10-0-1-240:~/jenkins/workspace/job2$
```



```
ubuntu@ip-10-0-1-152:~$ ls
jenkins
ubuntu@ip-10-0-1-152:~$ cd jenkins/workspace/
ubuntu@ip-10-0-1-152:~/jenkins/workspace$ ls
job3  job3@tmp
ubuntu@ip-10-0-1-152:~/jenkins/workspace$ cd job3
ubuntu@ip-10-0-1-152:~/jenkins/workspace/job3$ ls
Dockerfile  images  index.html
ubuntu@ip-10-0-1-152:~/jenkins/workspace/job3$ 
```

Job 3 result



Hello world!