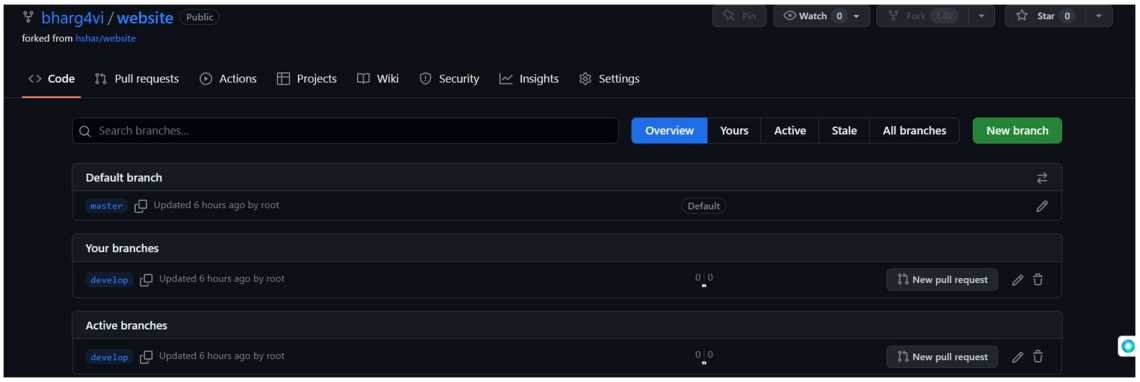
Name: Bhargavi Kamble

Student ID: 202051048

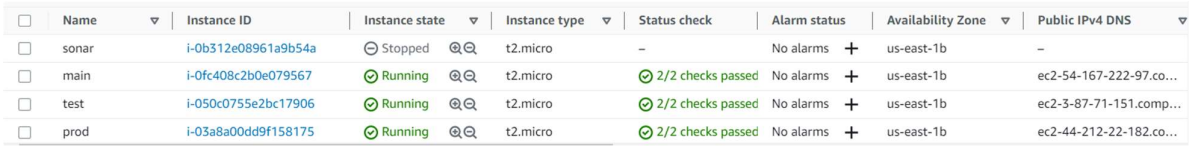
Capstone Project

1. Git Workflow has to be implemented:

To implement the gitflow I've created two branches master and develop. All the development changes will be committed to develop branch. Once the development is completed the changes made will be pushed to master branch.

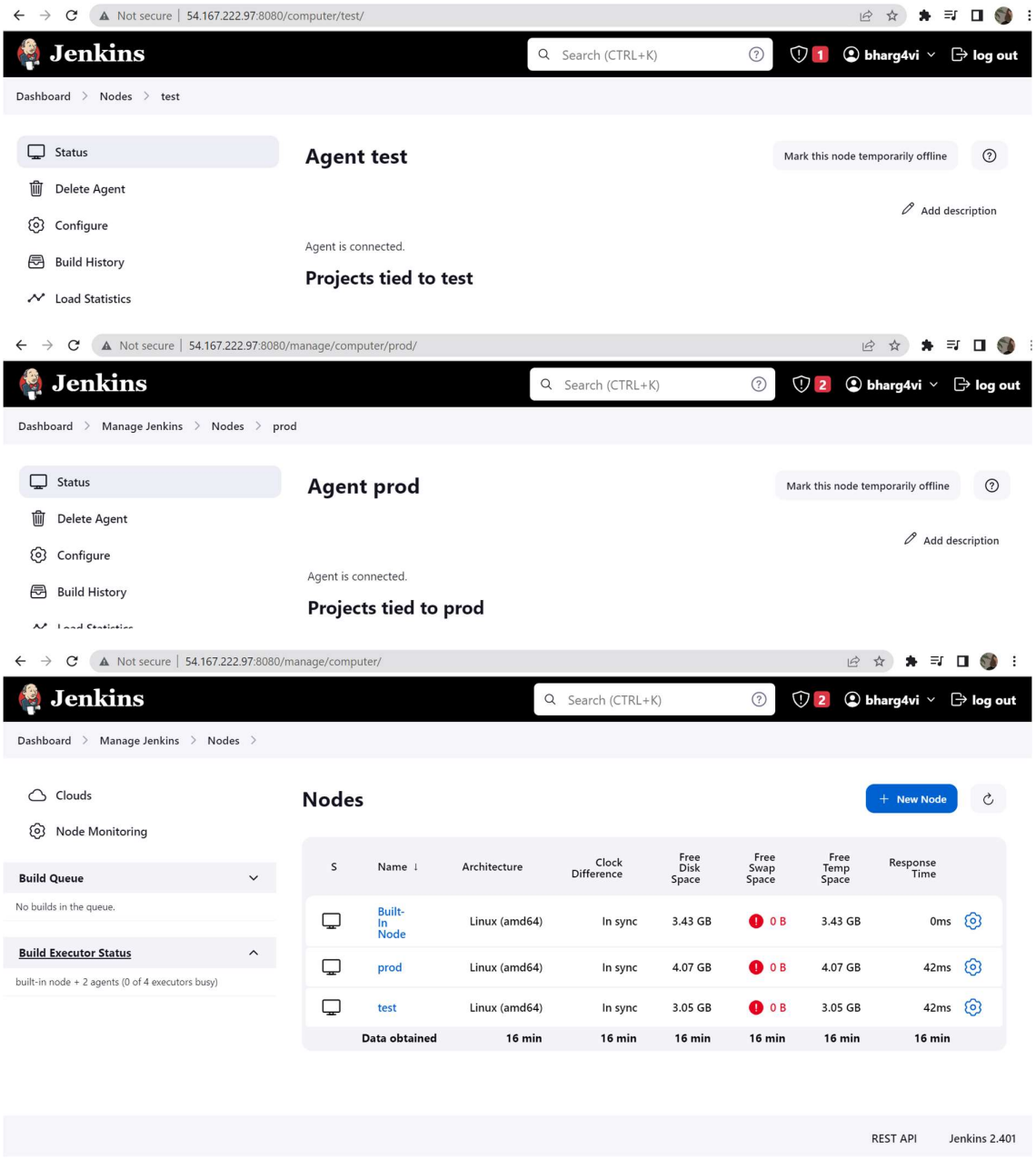


Create three instances called main, prod and test, install jenkins and configure inbound rules in both of them:

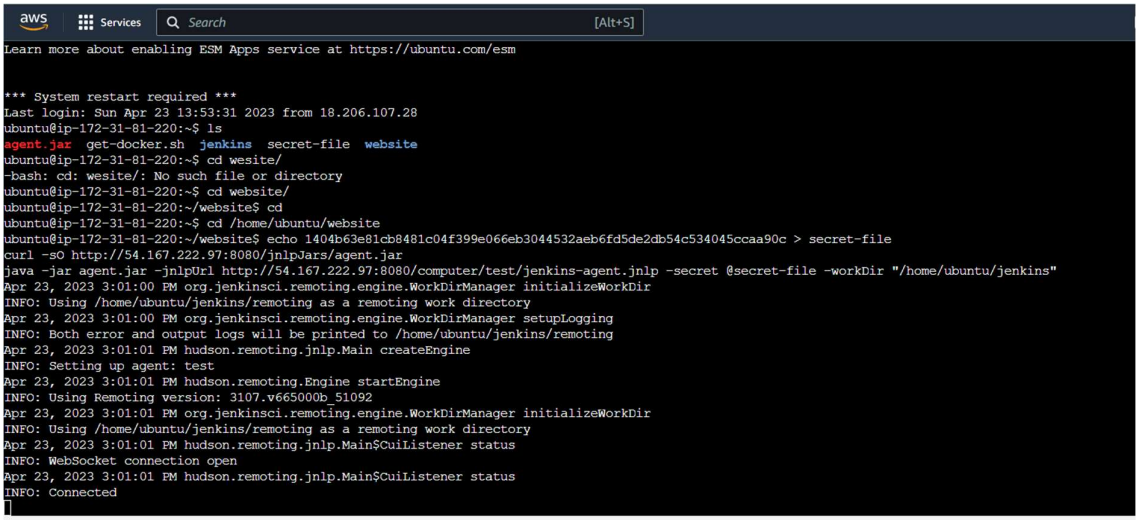


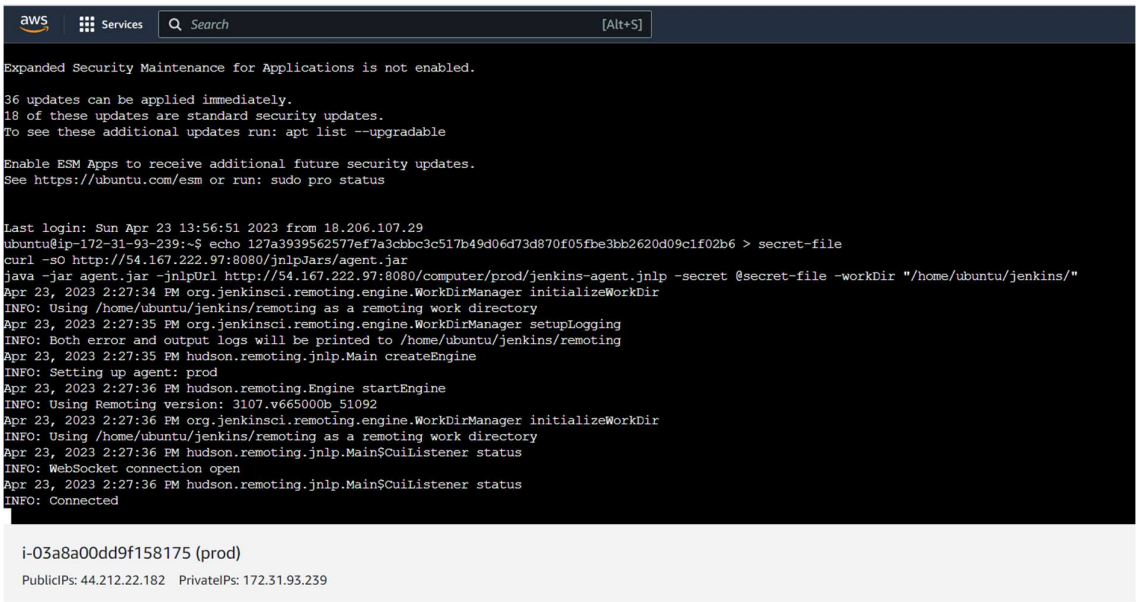
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
sonar	i-0b312e08961a9b54a	Stopped	t2.micro	–	No alarms	us-east-1b	–
main	i-0fc408c2b0e079567	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b	ec2-54-167-222-97.co...
test	i-050c0755e2bc17906	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b	ec2-3-87-71-151.comp...
prod	i-03a8a00dd9f158175	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b	ec2-44-212-22-182.co...

Create two agents as test and prod and connect it to test and prod instance respectively instance:



2. Code Build should automatically be triggered once commit is made to master branch or develop branch.
If commit is made to master branch, test and push to prod
If commit is made to develop branch, just test the product, do not push to prod:

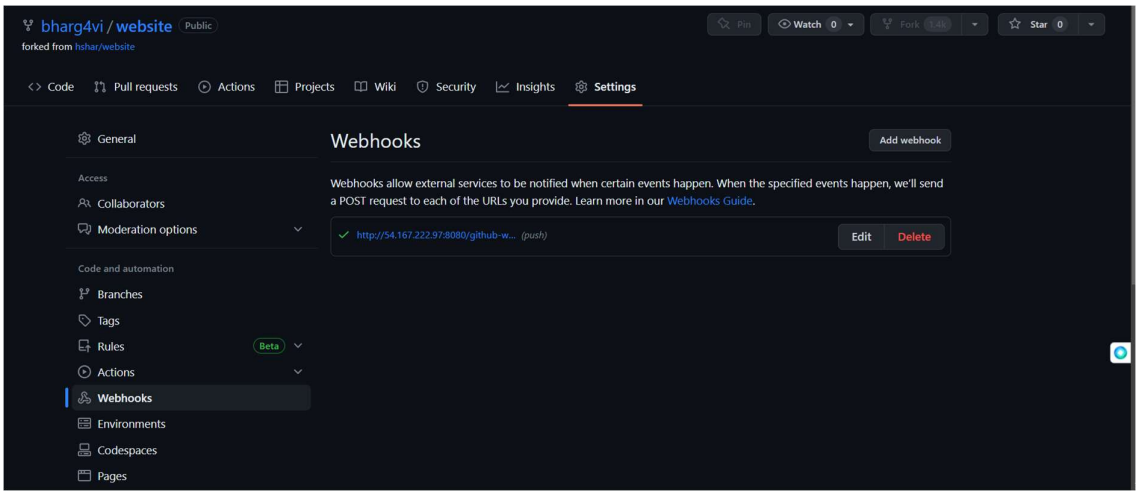




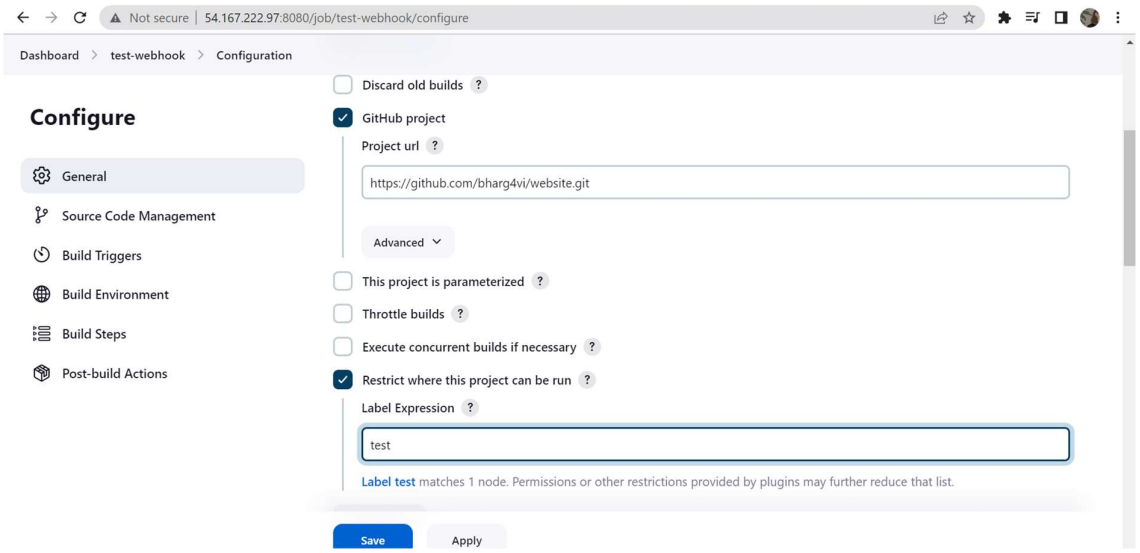
Clone given repo in the test instance:

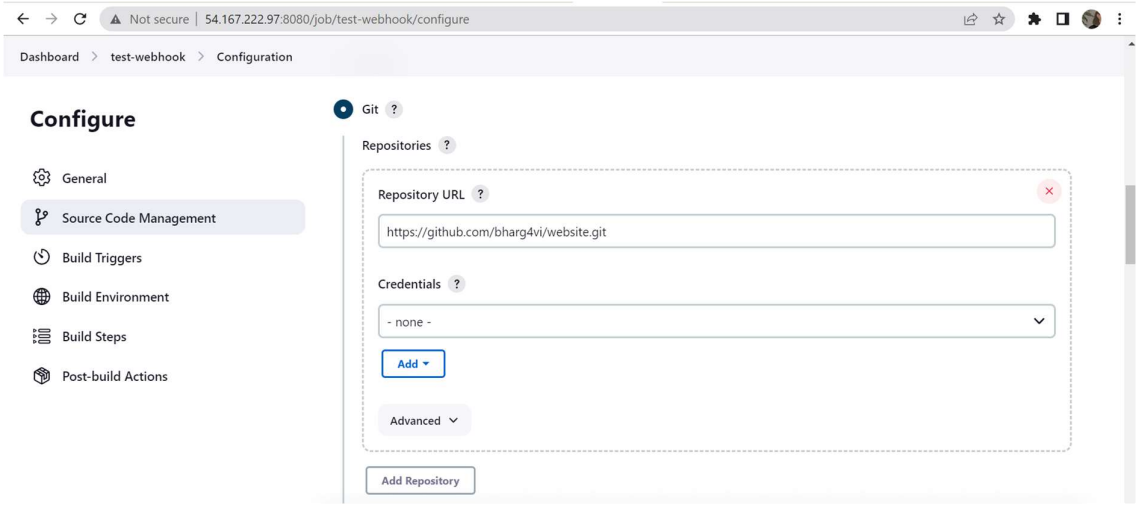
```
Last login: Sat Apr 22 12:20:43 2023 from 18.206.107.27
ubuntu@ip-172-31-81-220:~$ sudo git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ubuntu/.git/
ubuntu@ip-172-31-81-220:~$ sudo git clone https://github.com/bharg4vi/website.git
Cloning into 'website'...
remote: Enumerating objects: 8, done.
remote: Total 8 (delta 0), reused 0 (delta 0), pack-reused 8
Receiving objects: 100% (8/8), 82.69 KiB | 11.81 MiB/s, done.
Resolving deltas: 100% (1/1), done.
ubuntu@ip-172-31-81-220:~$ ls
agent.jar  jenkins  secret-file  website
ubuntu@ip-172-31-81-220:~$
```

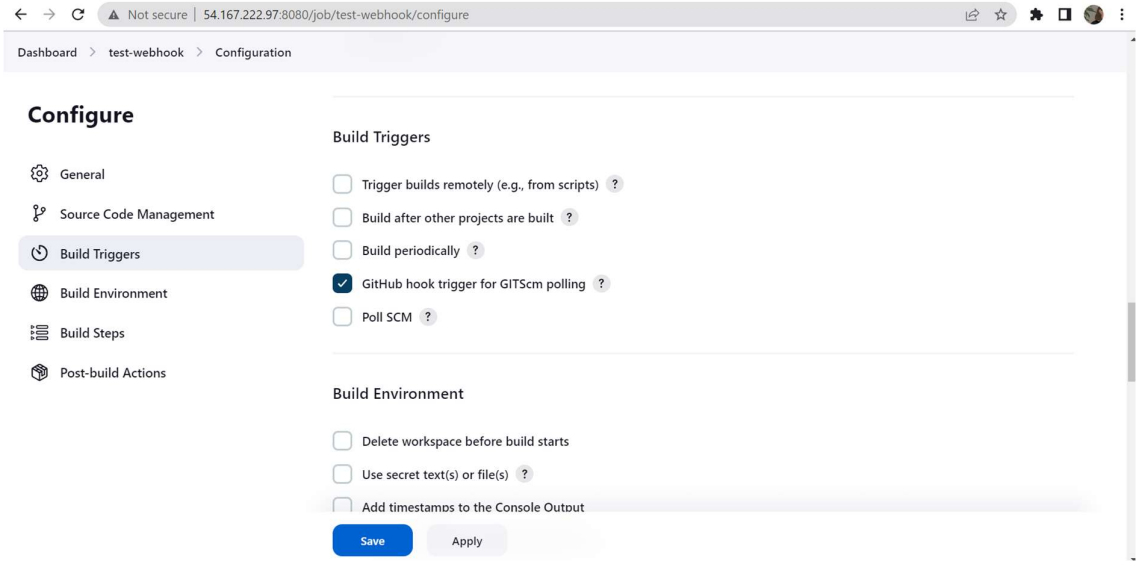
Add webhook to your project on github:

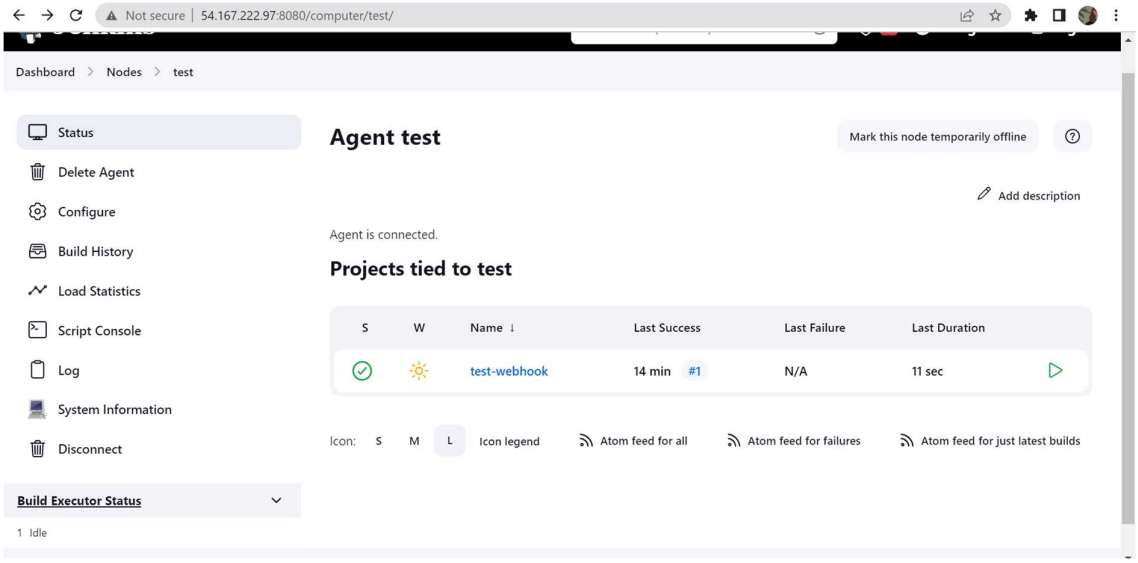


Create first job to check the web-hook:


Not secure | 54.167.222.97:8080/job/test-webhook/configure
Dashboard > test-webhook > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Discard old builds
GitHub project
Project url
https://github.com/bharg4vi/website.git
Advanced
This project is parameterized
Throttle builds
Execute concurrent builds if necessary
Restrict where this project can be run
Label Expression
test
Label test matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.
Save
Apply

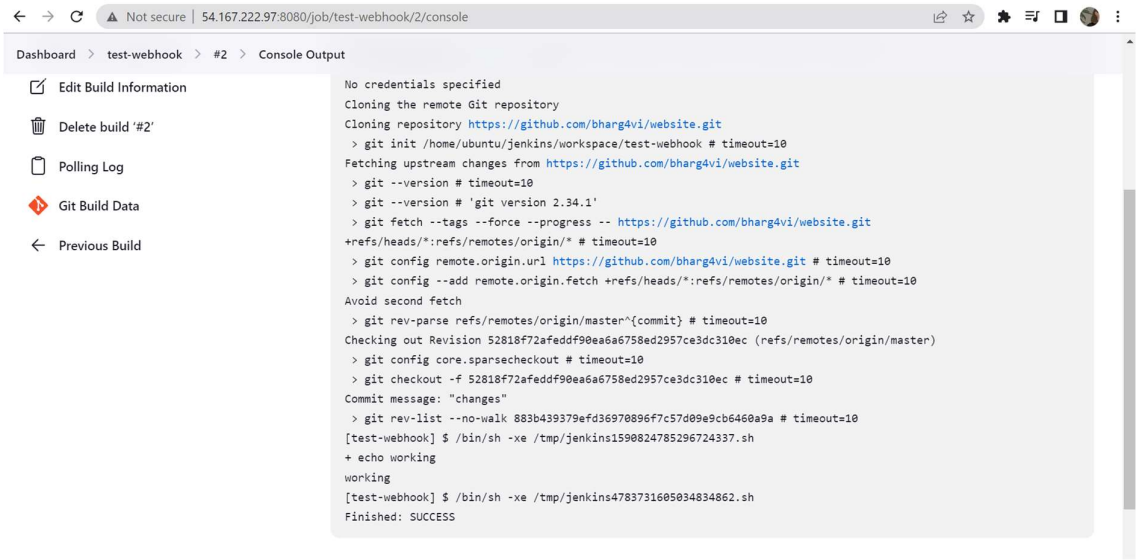

Not secure | 54.167.222.97:8080/job/test-webhook/configure
Dashboard > test-webhook > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Git
Repositories
Repository URL
https://github.com/bharg4vi/website.git
Credentials
- none -
Add
Advanced
Add Repository


Not secure | 54.167.222.97:8080/job/test-webhook/configure
Dashboard > test-webhook > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Build Triggers
Trigger builds remotely (e.g., from scripts)
Build after other projects are built
Build periodically
GitHub hook trigger for GITScm polling
Poll SCM
Build Environment
Delete workspace before build starts
Use secret text(s) or file(s)
Add timestamps to the Console Output
Save
Apply



Webhook is tested successfully.

We now try changing content in the repository and start the build:



The build was successful.

3. The Code should be containerized with the help of a Dockerfile. The Dockerfile should be built every time there is a push to Git-Hub. Use the following pre-built container for your application:
hshar/webapp
The code should reside in '/var/www/html':

We will now create a branch develop and make changes to the same:

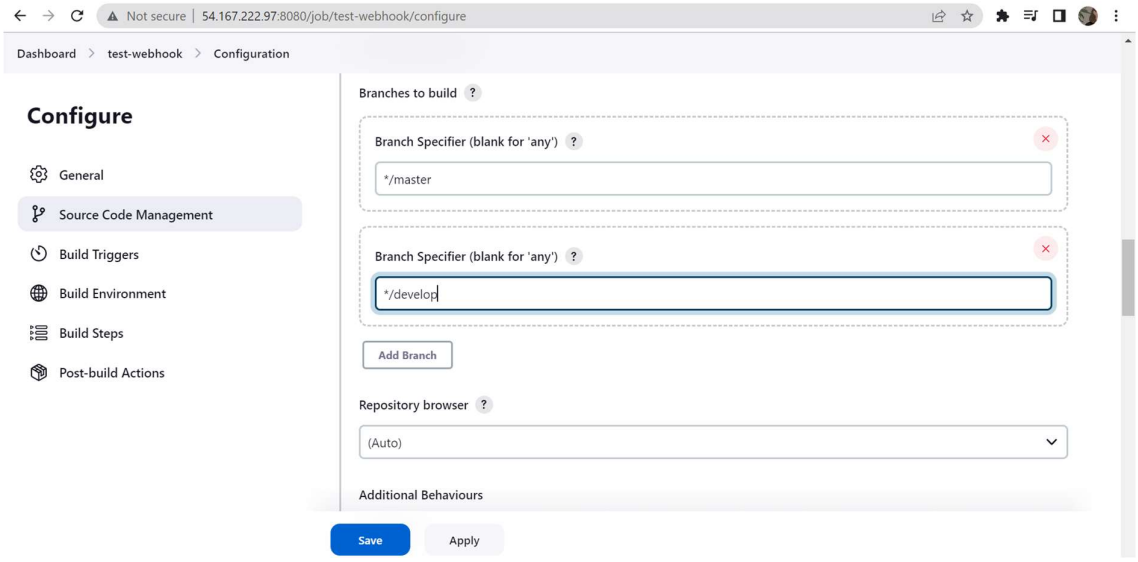
Create a docker file and commit it to git reposit:

```
ubuntu@ip-172-31-81-220:~/website$ sudo git branch
* master
ubuntu@ip-172-31-81-220:~/website$ sudo git branch develop
ubuntu@ip-172-31-81-220:~/website$ sudo git branch
  develop
* master
ubuntu@ip-172-31-81-220:~/website$ sudo git checkout develop
M       index.html
Switched to branch 'develop'
ubuntu@ip-172-31-81-220:~/website$ sudo nano dockerfile
```

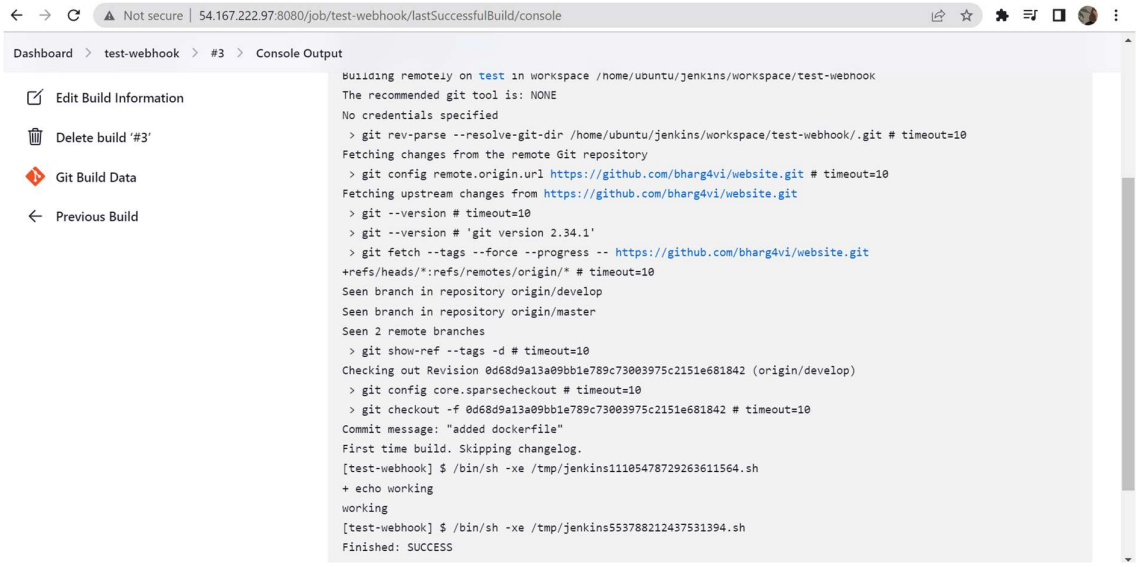
```
  GNU nano 6.2
FROM hshar/webapp
ADD ./var/www/html
```

```
ubuntu@ip-172-31-81-220:~/website$ sudo git add .
ubuntu@ip-172-31-81-220:~/website$ sudo commit -m "added dockerfile"
sudo: commit: command not found
ubuntu@ip-172-31-81-220:~/website$ sudo git commit -m "added dockerfile"
[develop 0d68d9a] added dockerfile
 Committer: root <root@ip-172-31-81-220.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 2 files changed, 3 insertions(+), 1 deletion(-)
 create mode 100644 dockerfile
```

Edit the build inserting develop branch to build:

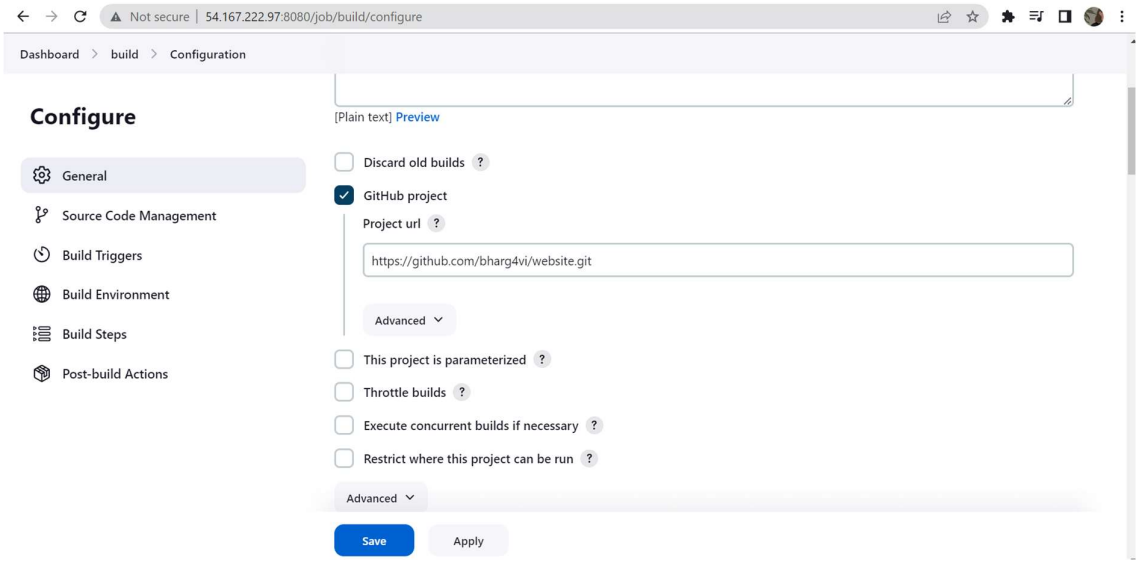


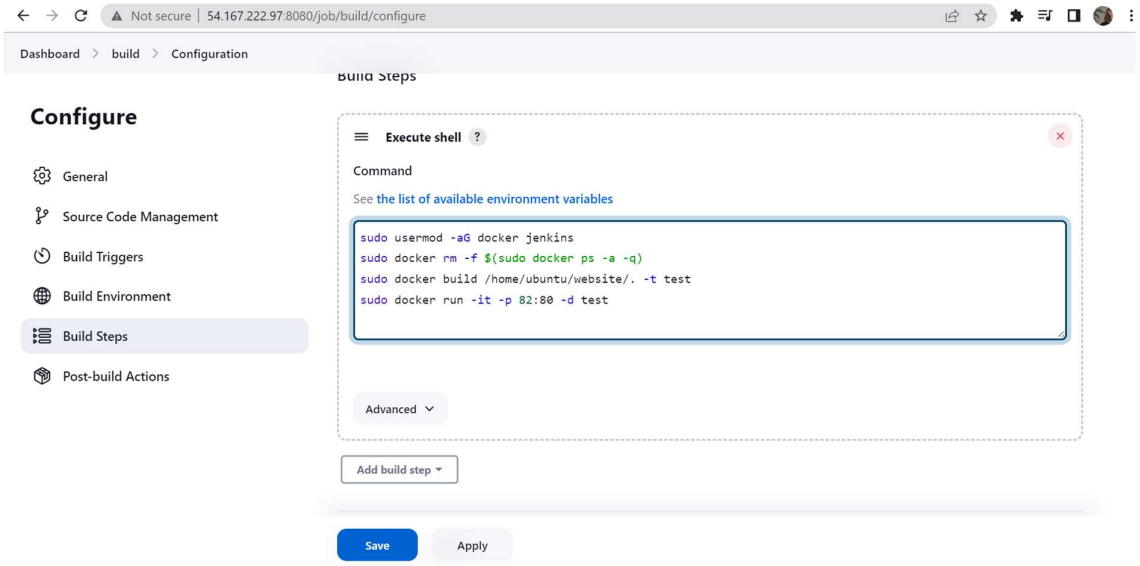
Push the file and start the build:

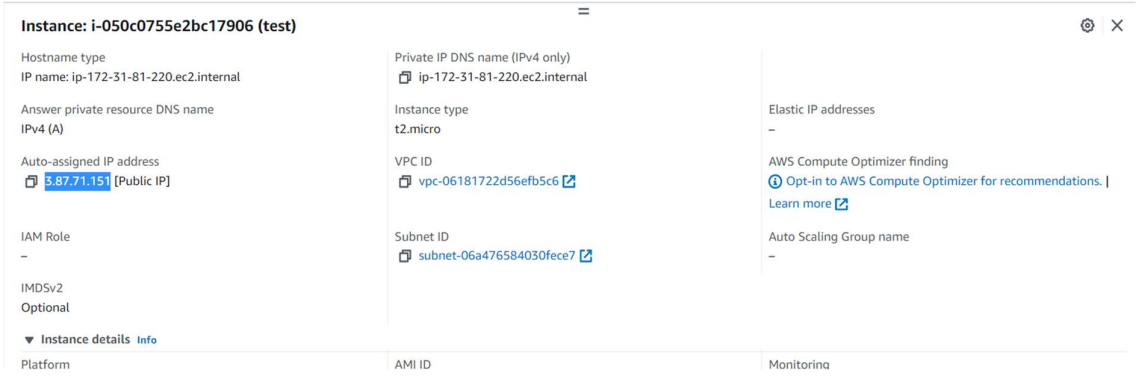


It was successfully built.

Create a new job to build website:



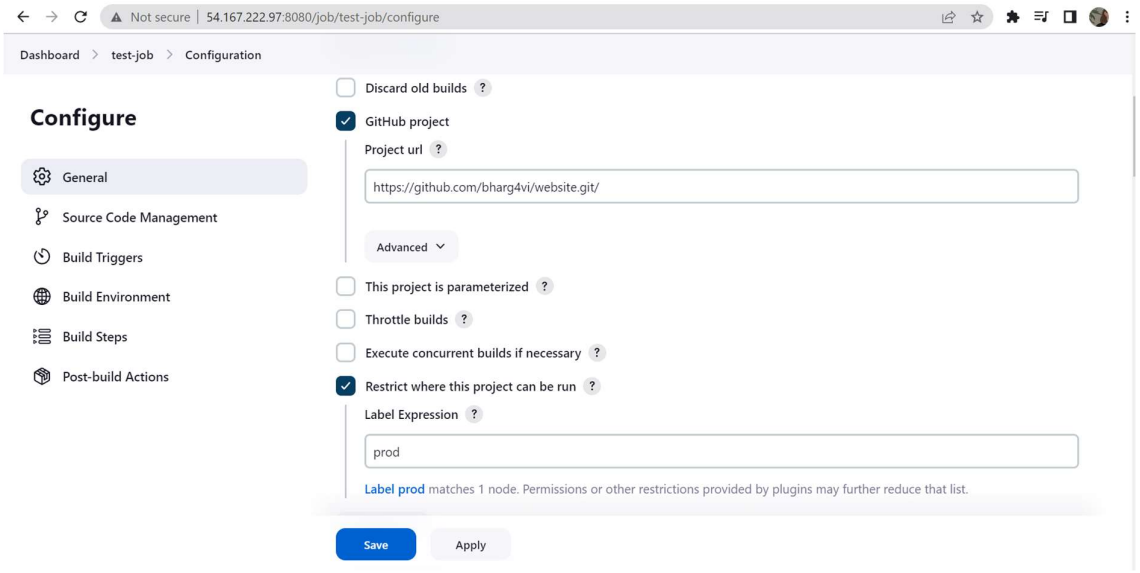

Not secure | 54.167.222.97:8080/job/build/configure
Dashboard > build > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Execute shell
Command
See the list of available environment variables
sudo usermod -aG docker jenkins
sudo docker rm -f $(sudo docker ps -a -q)
sudo docker build /home/ubuntu/website/. -t test
sudo docker run -it -p 82:80 -d test
Advanced
Add build step
Save
Apply

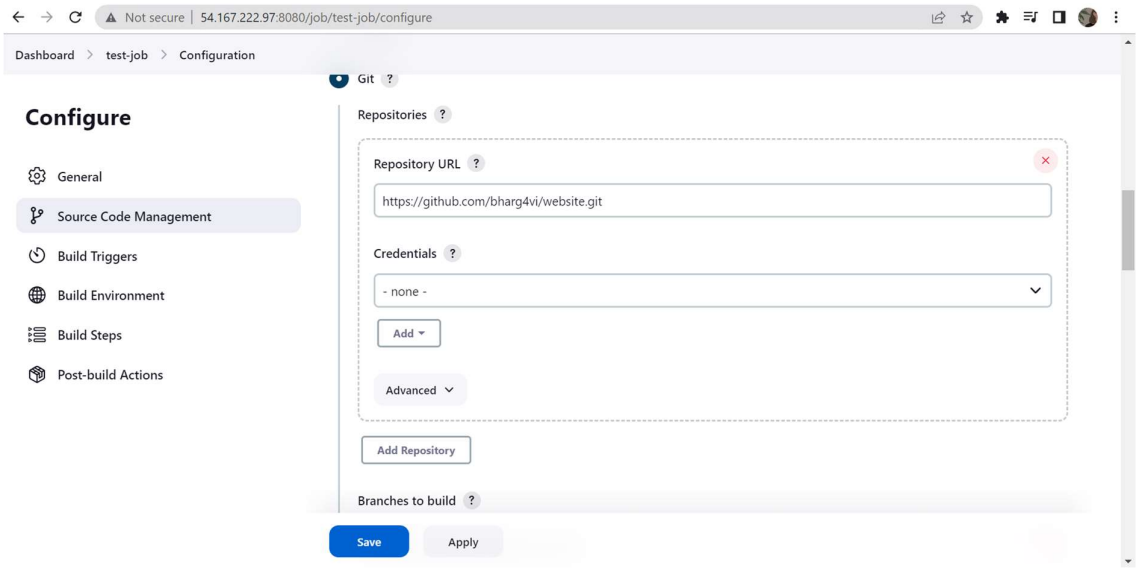

Instance: i-050c0755e2bc17906 (test)
Hostname type
IP name: ip-172-31-81-220.ec2.internal
Private IP DNS name (IPv4 only)
ip-172-31-81-220.ec2.internal
Answer private resource DNS name
IPv4 (A)
Instance type
t2.micro
Elastic IP addresses
–
Auto-assigned IP address
3.87.71.151 [Public IP]
VPC ID
vpc-06181722d56efb5c6
AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations.
Learn more
IAM Role
–
Subnet ID
subnet-06a476584030fece7
Auto Scaling Group name
–
IMDSv2
Optional
Instance details Info
Platform
AMI ID
Monitoring

Not secure | 3.87.71.151:82
Hi world!
GitHub

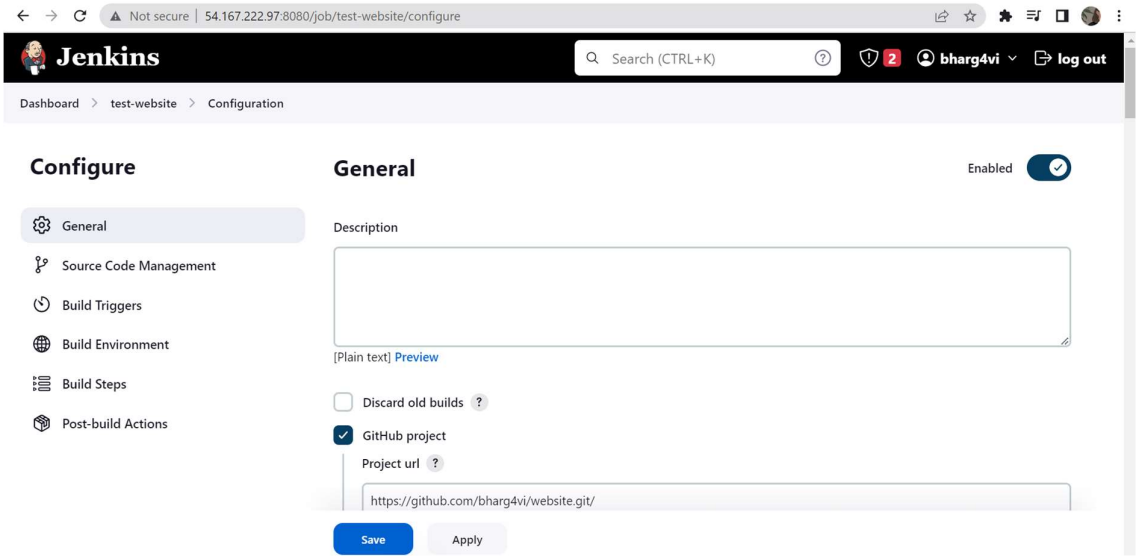
It was built successfully.

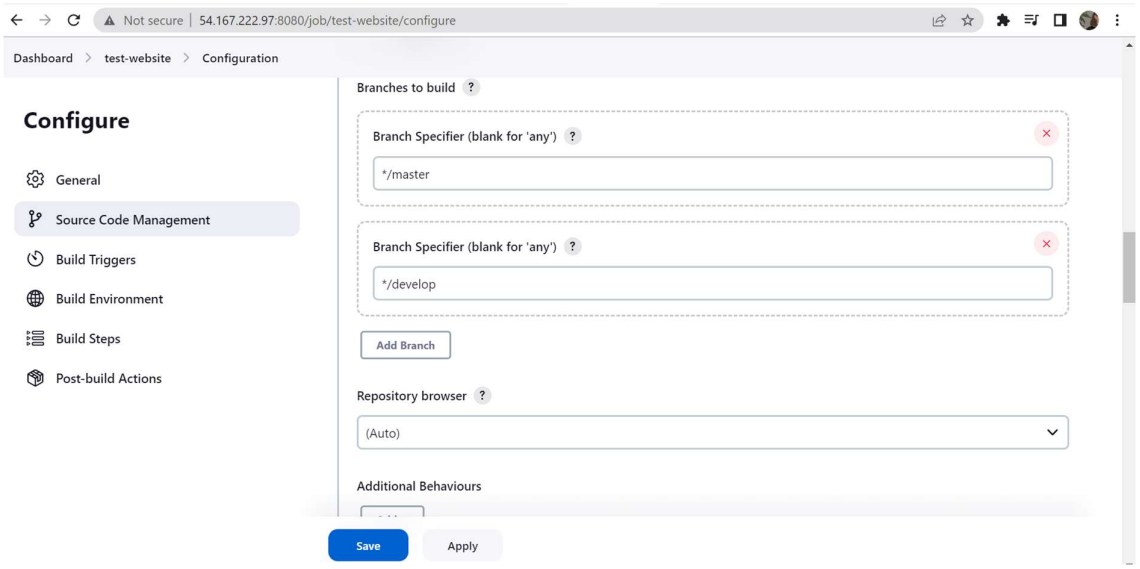
Create a job to test-job:

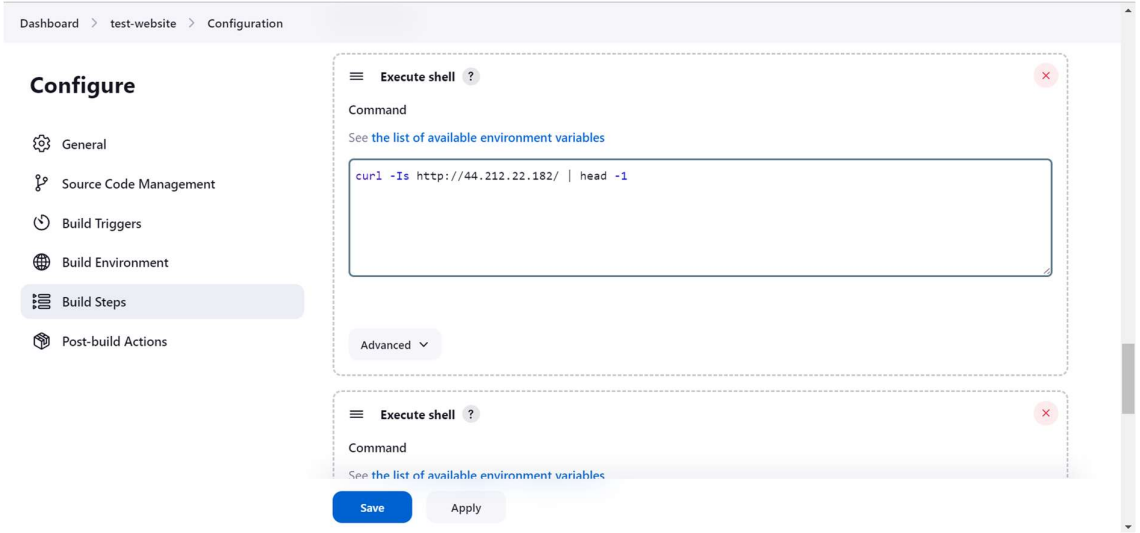




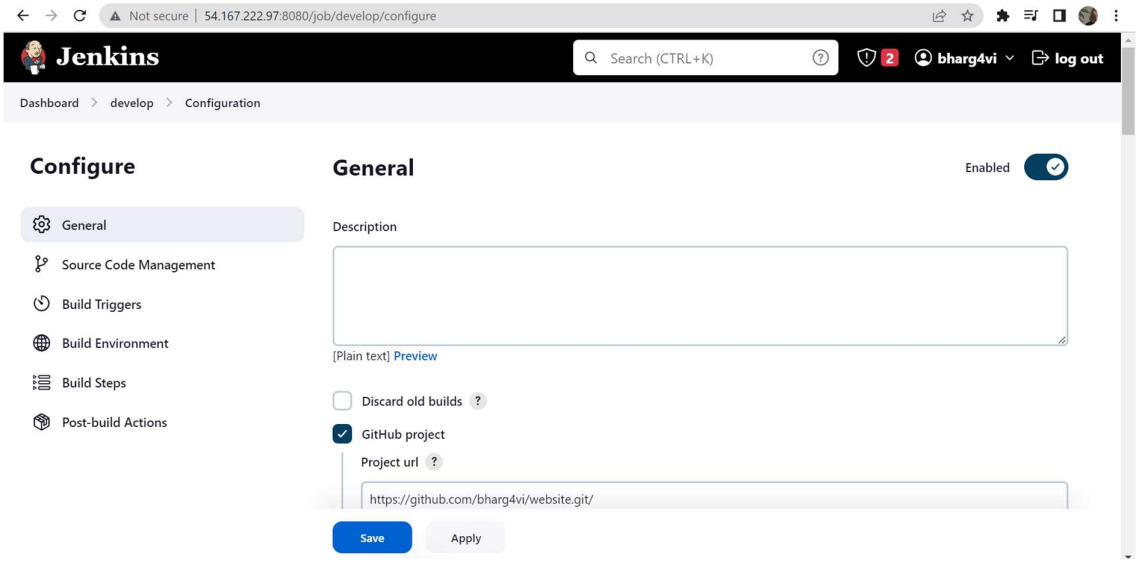
Create a job to test the website:

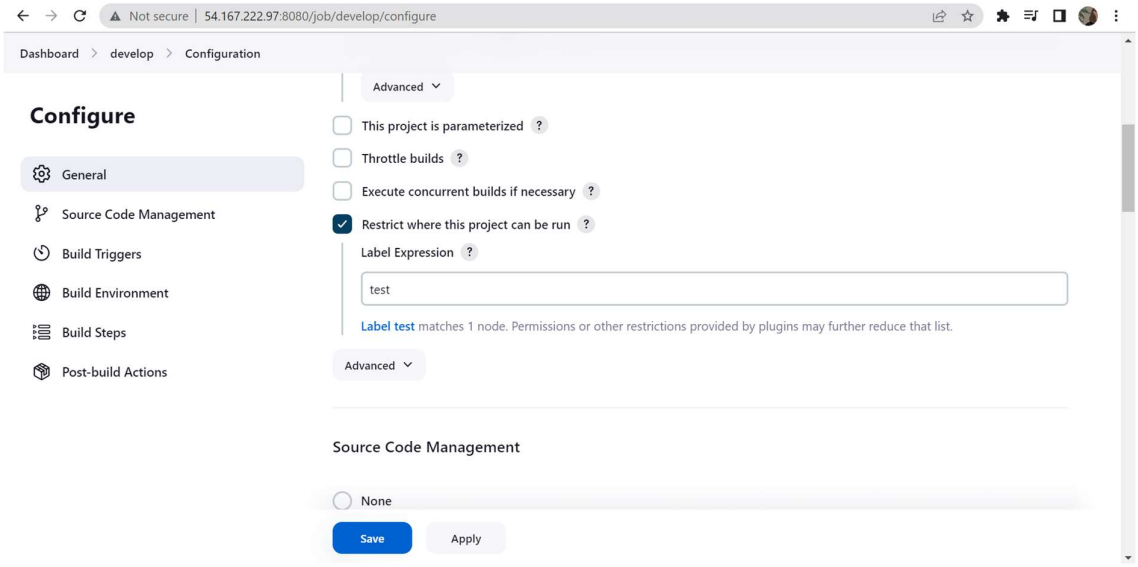

Not secure | 54.167.222.97:8080/job/test-website/configure
Jenkins
Search (CTRL+K)
bharg4vi
log out
Dashboard > test-website > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
General
Enabled
Description
[Plain text] Preview
Discard old builds
GitHub project
Project url
https://github.com/bharg4vi/website.git/
Save
Apply

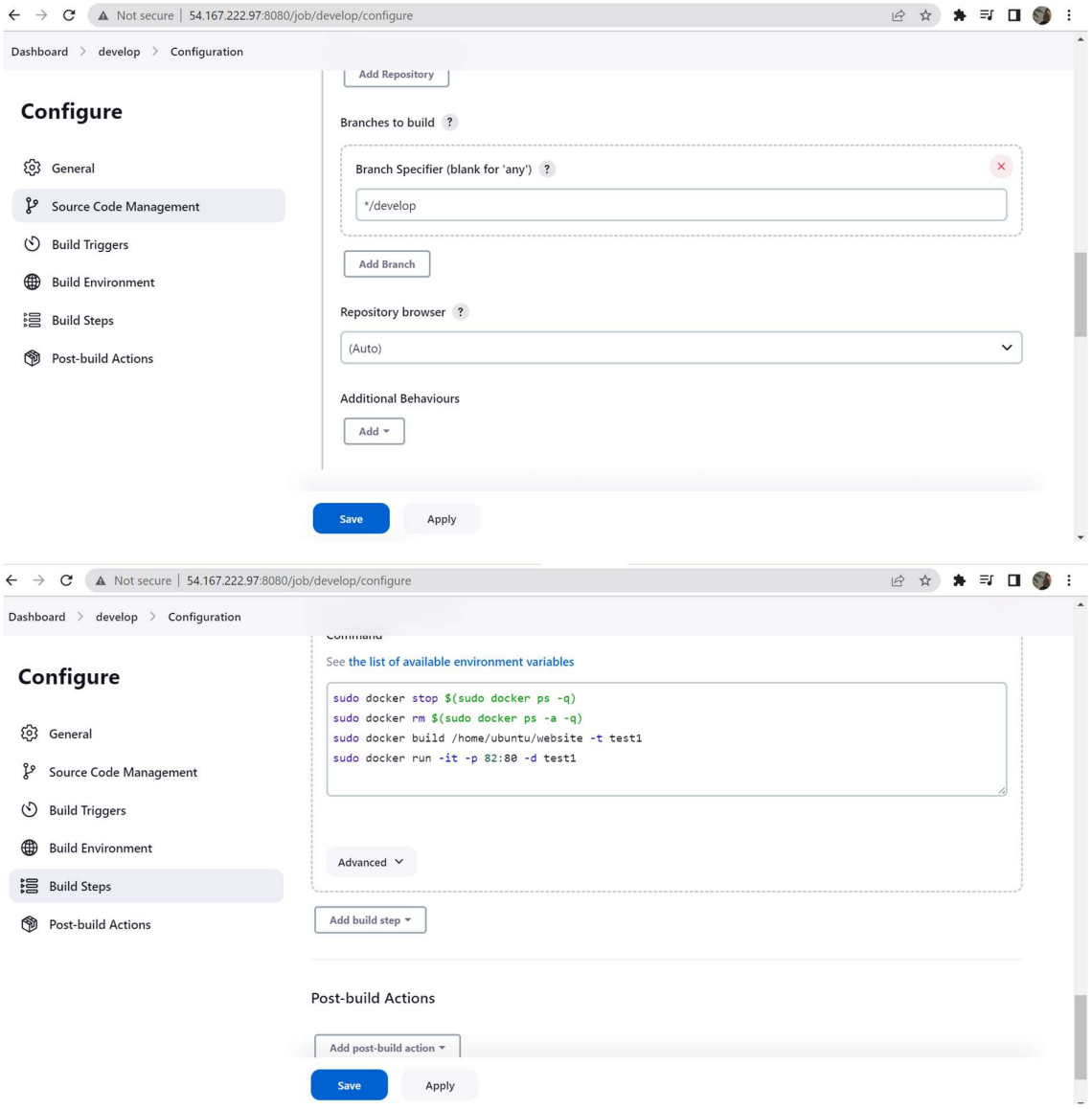

Not secure | 54.167.222.97:8080/job/test-website/configure
Dashboard > test-website > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Branches to build
Branch Specifier (blank for 'any')
*/master
Branch Specifier (blank for 'any')
*/develop
Add Branch
Repository browser
(Auto)
Additional Behaviours
Save
Apply


Dashboard > test-website > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Execute shell
Command
See the list of available environment variables
curl -Is http://44.212.22.182/ | head -1
Advanced
Execute shell
Command
Save
Apply

Create a new job for develop branch:







4. The above tasks should be defined in a Jenkins Pipeline, with the following Jobs Job 1 - Building Website = **build website**
Job 2 - Testing Website Job = **test-job, test website**

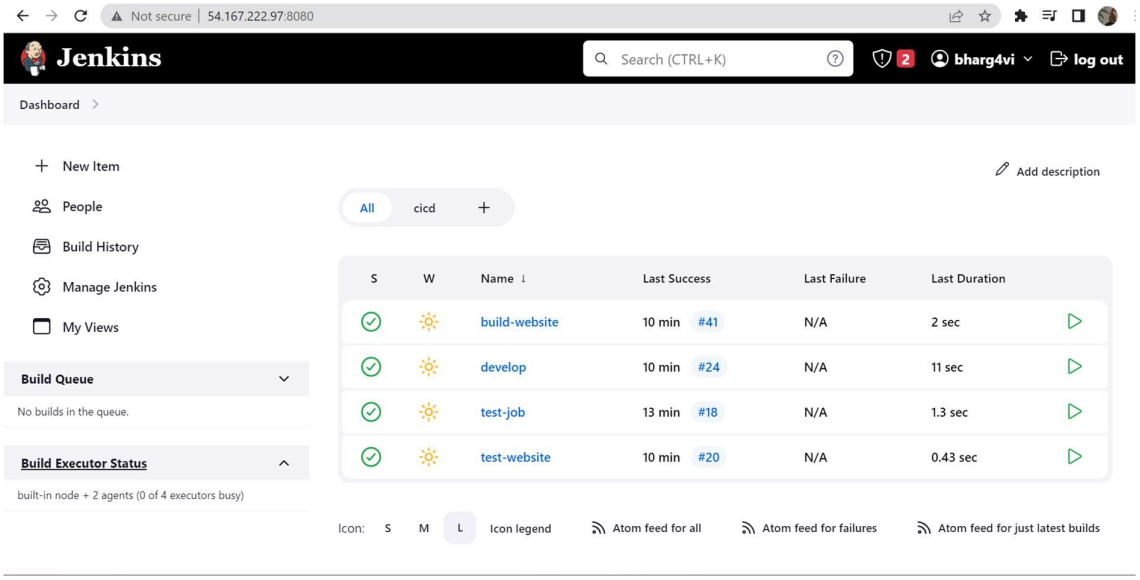
Job 3 - Push to Production = **develop**

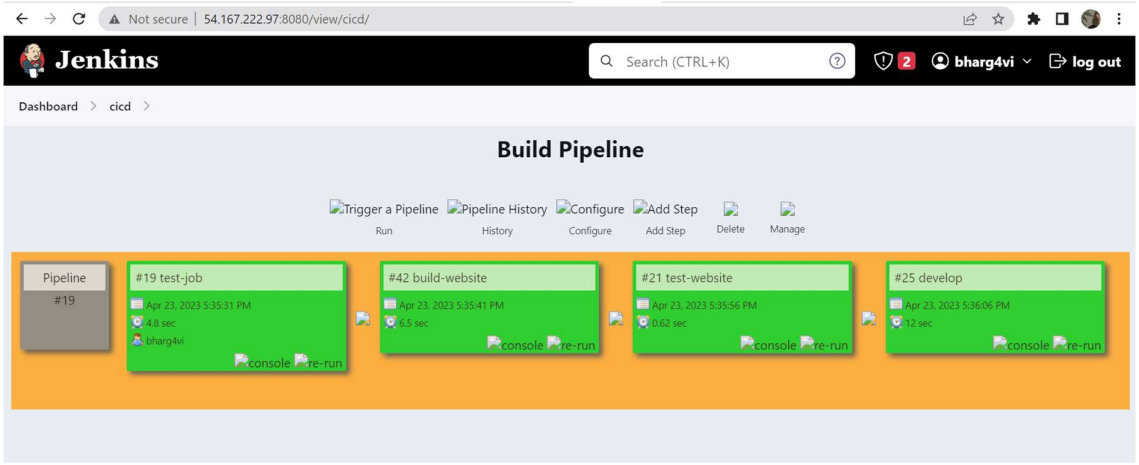
Queue projects one after other in the order:

test-job → build website → test website → develop

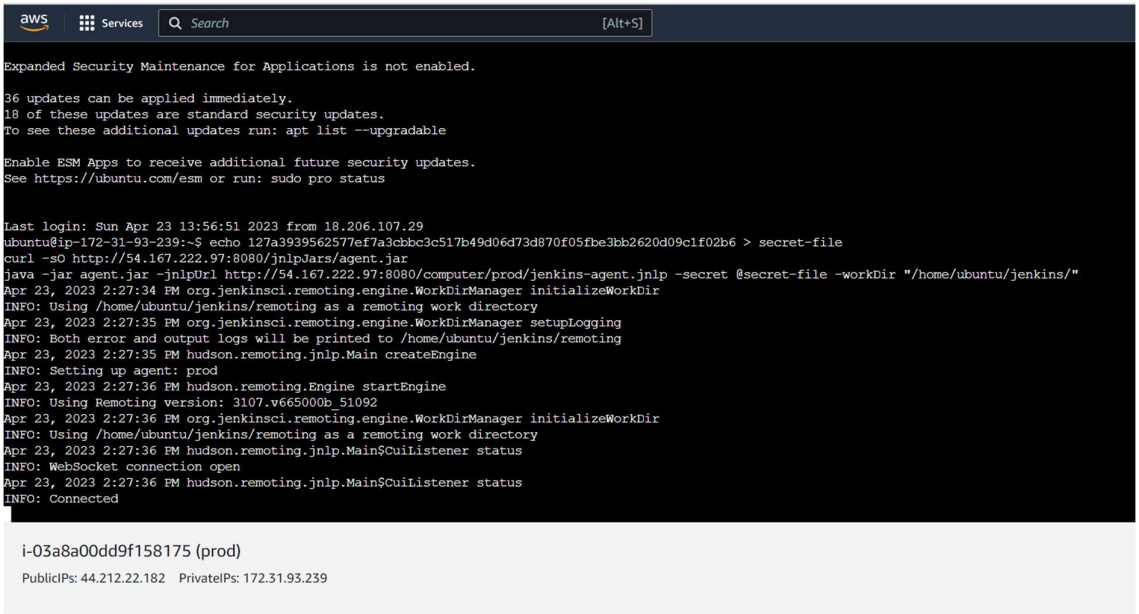
Similarly other projects can be configured the same way.

Build CI/CD pipeline as mentioned below:

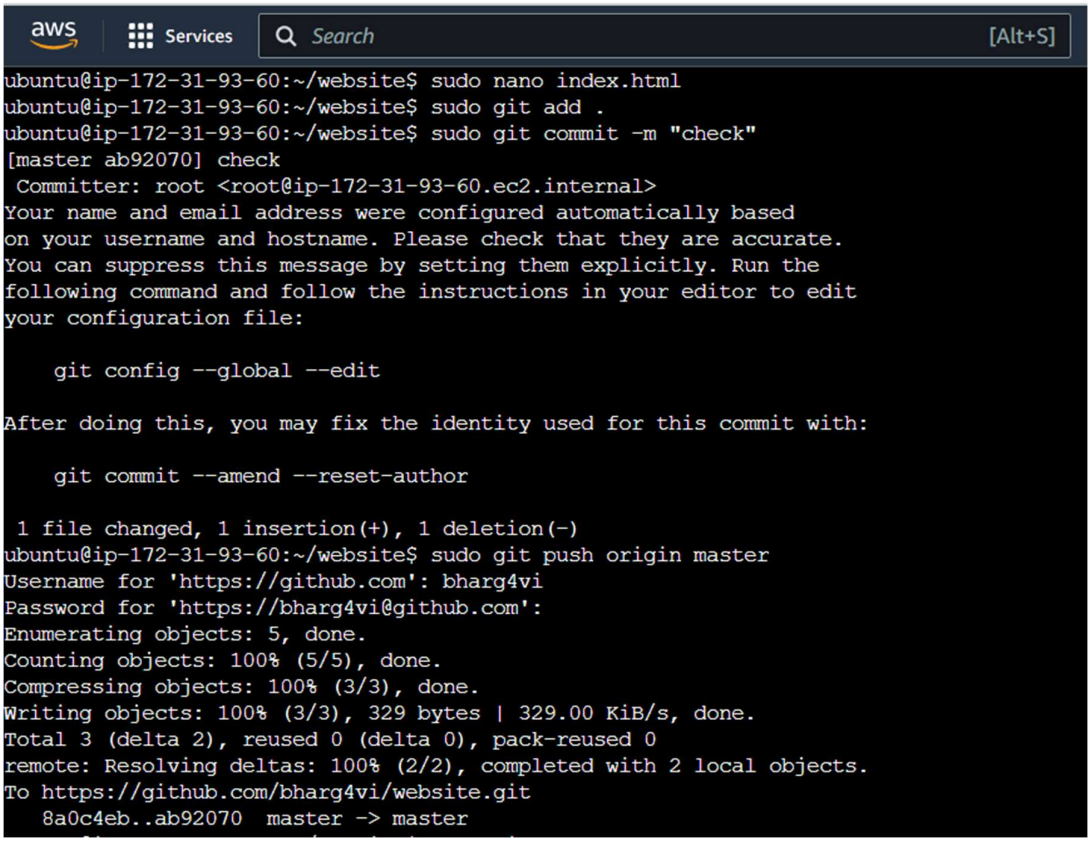




This is our prod:



We will make changes in master branch to test our project:


aws
Services
Search
[Alt+S]
ubuntu@ip-172-31-93-60:~/website$ sudo nano index.html
ubuntu@ip-172-31-93-60:~/website$ sudo git add .
ubuntu@ip-172-31-93-60:~/website$ sudo git commit -m "check"
[master ab92070] check
Committer: root <root@ip-172-31-93-60.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:
git config --global --edit
After doing this, you may fix the identity used for this commit with:
git commit --amend --reset-author
1 file changed, 1 insertion(+), 1 deletion(-)
ubuntu@ip-172-31-93-60:~/website$ sudo git push origin master
Username for 'https://github.com': bharg4vi
Password for 'https://bharg4vi@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 329 bytes | 329.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/bharg4vi/website.git
8a0c4eb..ab92070 master -> master

GNU nano 6.2
index
<html>
<head>
<title> Intellipaat </title>
</head>
<body style = "background-image:url('images/github3.jpg'); background-size: 100%">
<h2 ALIGN=CENTER>Namaste !!! 202051048 Bhargavi Kamble </h2>
</body>
</html>

Namaste !!! 202051048 Bhargavi Kamble

The changes can be seen.

```
ubuntu@ip-172-31-93-60:~/website$ sudo nano index.html
ubuntu@ip-172-31-93-60:~/website$ sudo git add .
ubuntu@ip-172-31-93-60:~/website$ sudo git commit -m "check"
[develop a2f600f] check
 Committer: root <root@ip-172-31-93-60.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 1 file changed, 1 insertion(+), 1 deletion(-)
ubuntu@ip-172-31-93-60:~/website$ sudo git push origin develop
Username for 'https://github.com': bharg4vi
Password for 'https://bharg4vi@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 311 bytes | 311.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/bharg4vi/website.git
   9db6cf0..a2f600f  develop -> develop
```

```
 GNU nano 6.2
<html>
<head>
<title> Intellipaat Website </title>
</head>
<body style = "background-image:url('images/github3.jpg'); background-size: 100%">
<h2 ALIGN=CENTER>Hello! This is bhargavi 202051048 </h2>
</body>
</html>
```

Namaste !!! 202051048 Bhargavi Kamble

GitHub

Whereas for the develop branch it remained the same.

5. Since you are setting up the server for the first time, ensure the following file exists on both Test and Prod server in /home/ubuntu/config-management/status.txt. This file will be used by a third-party tool. This should basically have the info whether apache is installed on the system or not
The content of this file, should be based on whether git is installed or not.
If apache is installed => Apache is Installed on this System"
If apache is not installed => "Apache is not installed on this System"

On prod server:

```
ubuntu@ip-172-31-93-239:~$ mkdir /home/ubuntu/config-management
ubuntu@ip-172-31-93-239:~$ nano /home/ubuntu/config-management/status.txt
```

```
  GNU nano 6.2                                                    /home/ubuntu/config-mar
if command -v apache2 &> /dev/null
then
    echo "Apache is Installed on this System" > /home/ubuntu/config-management/status.txt
else
    echo "Apache is not installed on this System" > /home/ubuntu/config-management/status.txt
fi
```

On test server:

```
ubuntu@ip-172-31-81-220:~$ mkdir /home/ubuntu/config-management
ubuntu@ip-172-31-81-220:~$ nano /home/ubuntu/config-management/status.txt
```

```
  GNU nano 6.2                                                    /home/ubuntu/config-managemer
if command -v apache2 &> /dev/null
then
    echo "Apache is Installed on this System" > /home/ubuntu/config-management/status.txt
else
    echo "Apache is not installed on this System" > /home/ubuntu/config-management/status.txt
fi
```

Git link: https://github.com/bharg4vi/website.git