

DEVOPS TASK 2

1) Installation of Docker:

Code:

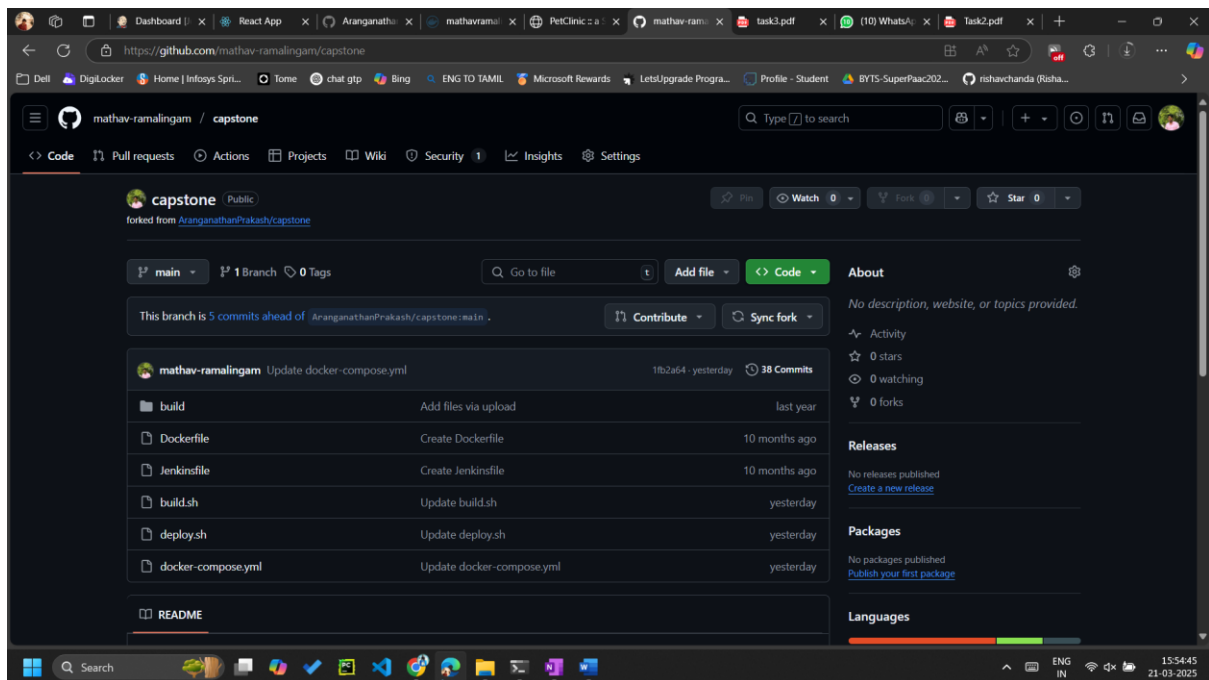
```
sudo apt install docker.io
docker --version
sudo systemctl start docker
sudo systemctl enable docker
sudo systemctl status docker
```

Screenshot:

```
root@LAPTOP-6V70H2B0:~# apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
docker.io is already the newest version (26.1.3-0ubuntu1~24.04.1).
The following packages were automatically installed and are no longer required:
  libdrm-intel1 libpciaccess0 libsensors-config libsensors5
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 9 not upgraded.
root@LAPTOP-6V70H2B0:~# docker --version
Docker version 26.1.3, build 26.1.3-0ubuntu1~24.04.1
root@LAPTOP-6V70H2B0:~# sudo systemctl start docker
root@LAPTOP-6V70H2B0:~# sudo systemctl enable docker
root@LAPTOP-6V70H2B0:~# sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-03-20 06:44:32 UTC; 1h 32min ago
   TriggeredBy: ● docker.socket
     Docs: https://docs.docker.com
    Main PID: 9561 (dockerd)
      Tasks: 30
     Memory: 62.0M (-)
    CGroup: /system.slice/docker.service
            └─ 9561 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
            └─10253 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-port 70 -container-ip 172.17.0.2 -com
            └─10261 /usr/bin/docker-proxy -proto tcp -host-ip :: -host-port 70 -container-ip 172.17.0.2 -com
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185097971Z" level=warning msg="WARNIN
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185409232Z" level=warning msg="WARNIN
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185440810Z" level=warning msg="WARNIN
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185455410Z" level=warning msg="WARNIN
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185498240Z" level=info msg="Docker c
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.185845402Z" level=info msg="Daemon h
Mar 20 06:44:32 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:44:32.379205869Z" level=info msg="API list
Mar 20 06:44:32 LAPTOP-6V70H2B0 systemd[1]: Started docker.service - Docker Application Container Engine.
Mar 20 06:45:16 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:45:16.405475070Z" level=info msg="Layer sh
Mar 20 06:45:16 LAPTOP-6V70H2B0 dockerd[9561]: time="2025-03-20T06:45:16.550116575Z" level=info msg="Layer sh
lines 1-23/23 (END) ... skipping ...
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
```

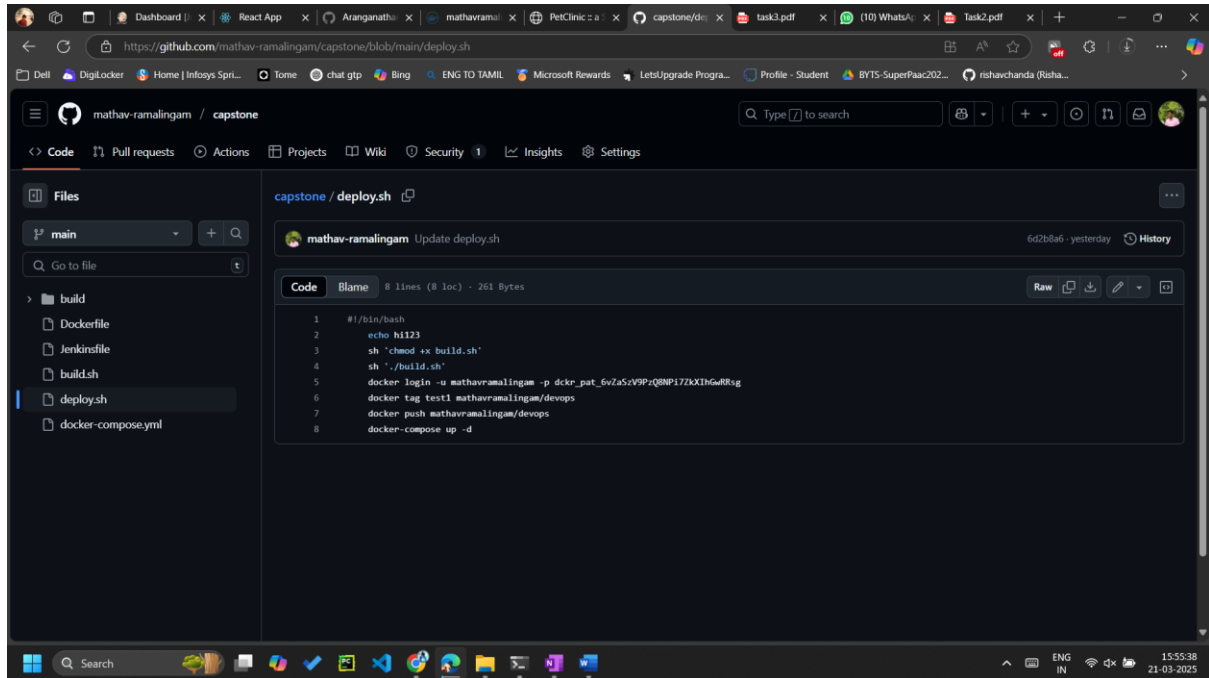
2) Fork a copy of a GitHub repo which contains the necessary files which will result in the clone of that repo in our own repository

Screenshot:



3) Then change the token and repo name of the docker Hub in the deploy.sh file which is in our repository.

Screenshot

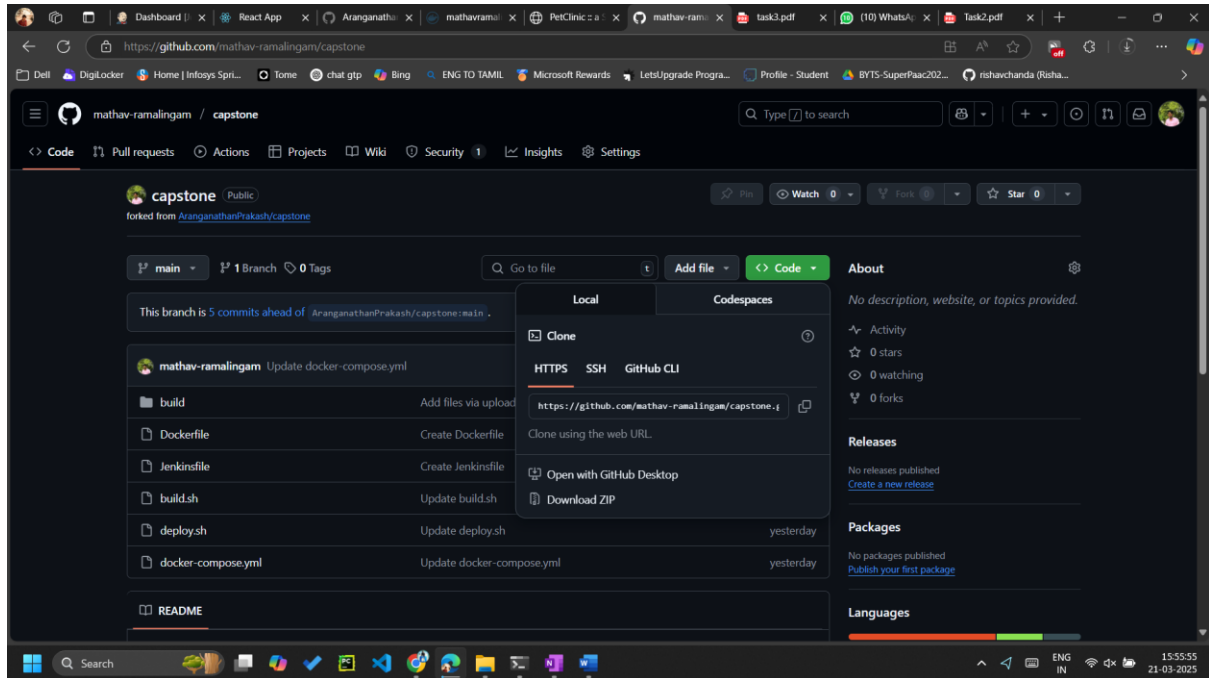


The screenshot shows a web browser displaying the GitHub repository page for 'mathav-ramalingam / capstone'. The file 'deploy.sh' is selected in the left sidebar under the 'build' directory. The main content area shows the code for 'deploy.sh', which is an 8-line shell script. The script sets the shell to /bin/bash, echoes 'hi123', makes 'build.sh' executable, runs 'build.sh', logs into Docker Hub with a specific token, tags the image 'test1 mathavramalingam/devops', pushes it to 'mathavramalingam/devops', and finally runs 'docker-compose up -d'.

```
1 #!/bin/bash
2 echo hi123
3 sh "chmod +x build.sh"
4 sh "./build.sh"
5 docker login -u mathavramalingam -p dckr_pat_6vzaS2VSPzQ8NP17ZkX1hGuRRsg
6 docker tag test1 mathavramalingam/devops
7 docker push mathavramalingam/devops
8 docker-compose up -d
```

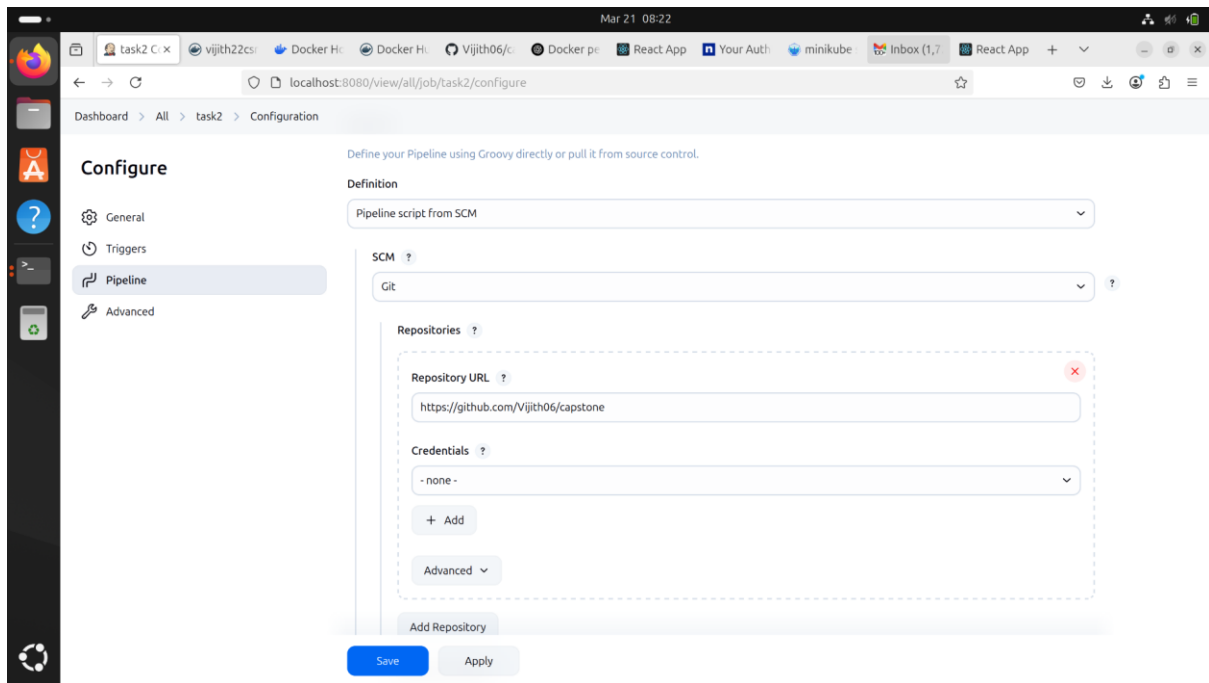
4) Then copy the GitHub link of the repository and go to Jenkins.

Screenshot:



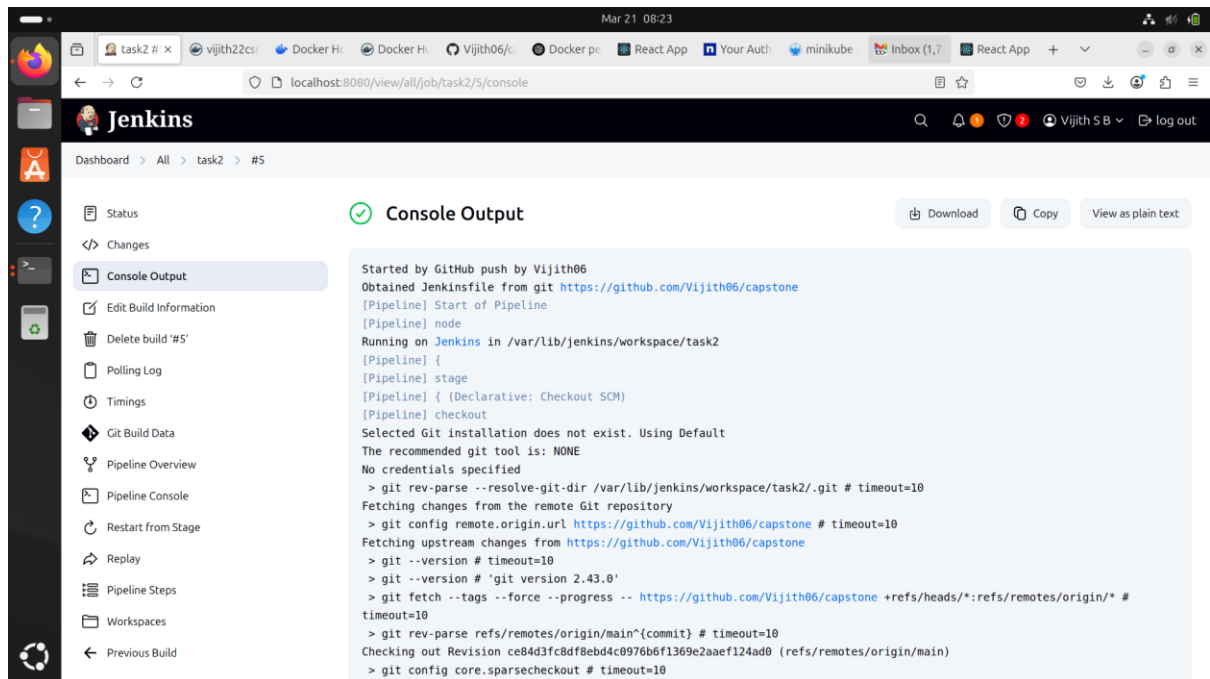
5) In Jenkins, create a new item (Job) with a type pipeline and add the copied GitHub url to it with the correct branch and Jenkinsfile.

ScreenShot:



6) After Creating the job, build it and it will give the console output and the docker image will be created.

Screenshot:



7) Now Build this docker image in the terminal with desired port number to it.

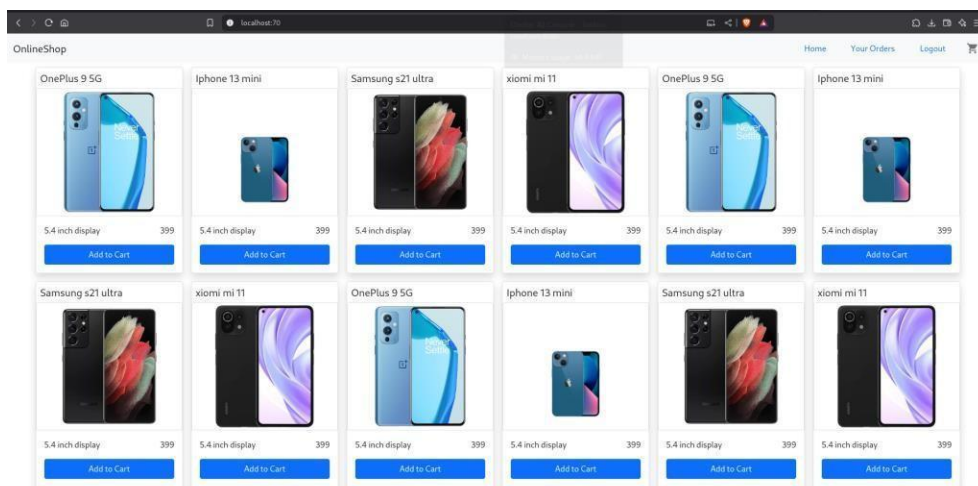
Code:

```
docker images
```

```
docker run -itd -p 70:80 test1
```

8) Go to the Browser and search for localhost:<PORT_NUMBER> and the respective application will be hosted.

Screenshot:

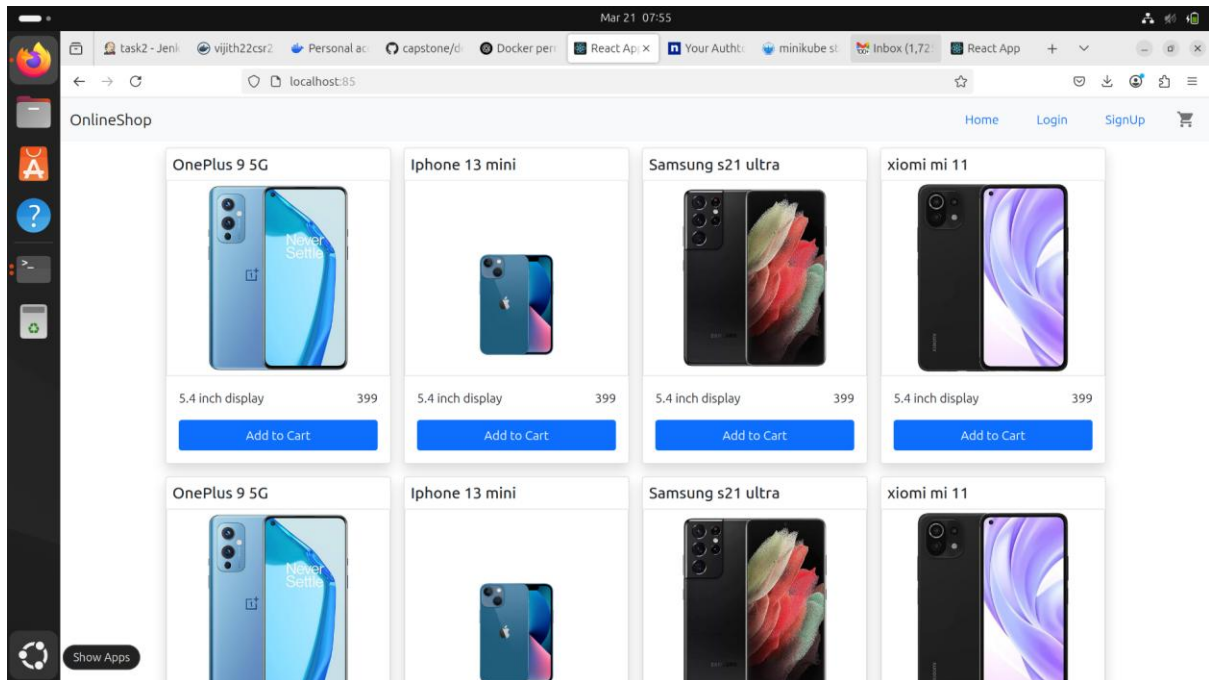


9) But, Instead of running the image by manually, we can also write the command for running in a file called docker-compose.yml

Code:

```
version: '3'
services:
  react-
  capstoneimage
  : "test1"
  ports:
    - "85:80"
```

Screenshot:



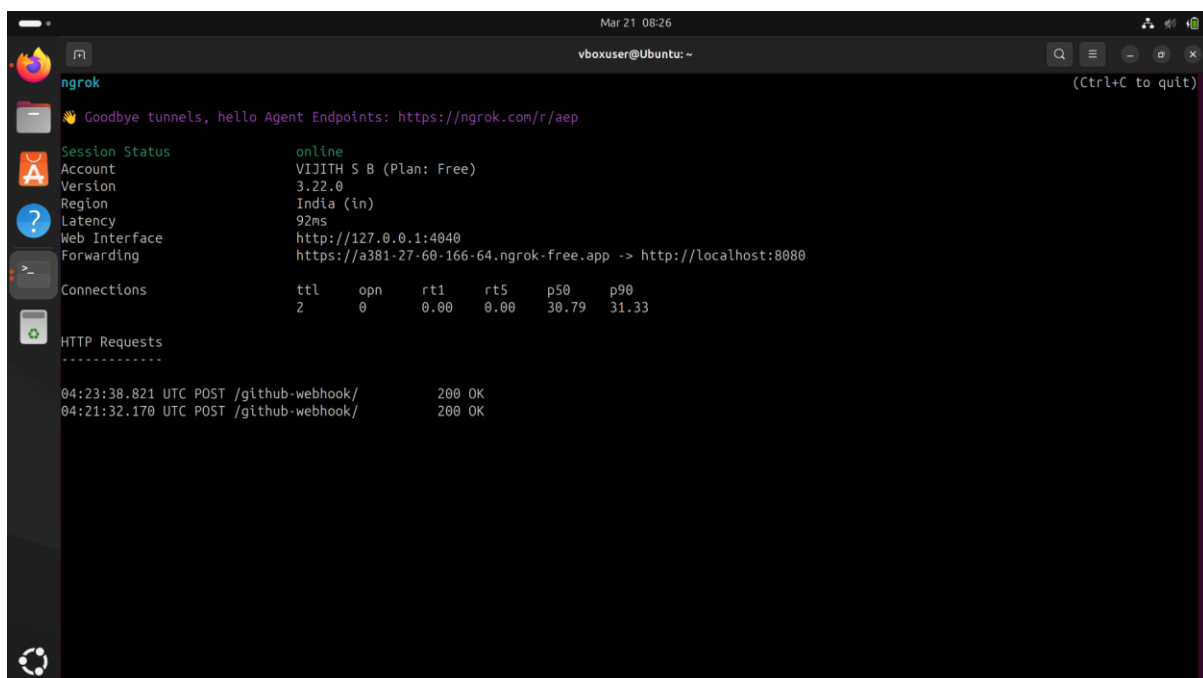
By Creating this, we no need to run the image by manually. (It will automatically run)

10) Adding Webhook to it which is available in GitHub for automatic build of the project.

Installing ngrok and with these command to get the Webhook Link.

Screenshot:

```
nganth@suganth-dellian:~$ sudo snap install ngrok
[sudo] password for suganth:
snap "ngrok" is already installed, see 'snap help refresh'
nganth@suganth-dellian:~$ ngrok config add-authtoken 2ua80vQJhkuuFC2Lz5UH28DyS4_SqRkxaw7boXsGwcd0wt
Authtoken saved to configuration file: /home/suganth/snap/ngrok/255/.config/ngrok/ngrok.yml
nganth@suganth-dellian:~$ ngrok http 8080
nganth@suganth-dellian:~$
```



github.com/Vijith06/capstone/settings/hooks

SWAYAMLeetCodehackerearthHackerRankChatGPTGDB onlineGitHubMeritcurveCodeChef6TH SEM - Google...Projects - GitLab

Vijith06 / capstone

Type to search

CodePull requestsActionsProjectsWikiSecurity1InsightsSettings

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Environments

Codespaces

Pages

Security

Code security

Webhooks

Add webhook

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](#).

✓ https://a381-27-60-166-64.ngrok-fr... (push)

EditDelete

Last delivery was successful.

11) Tick the checkbox of GitHub hook trigger for GITScm polling in Jenkins.

Screenshot:

