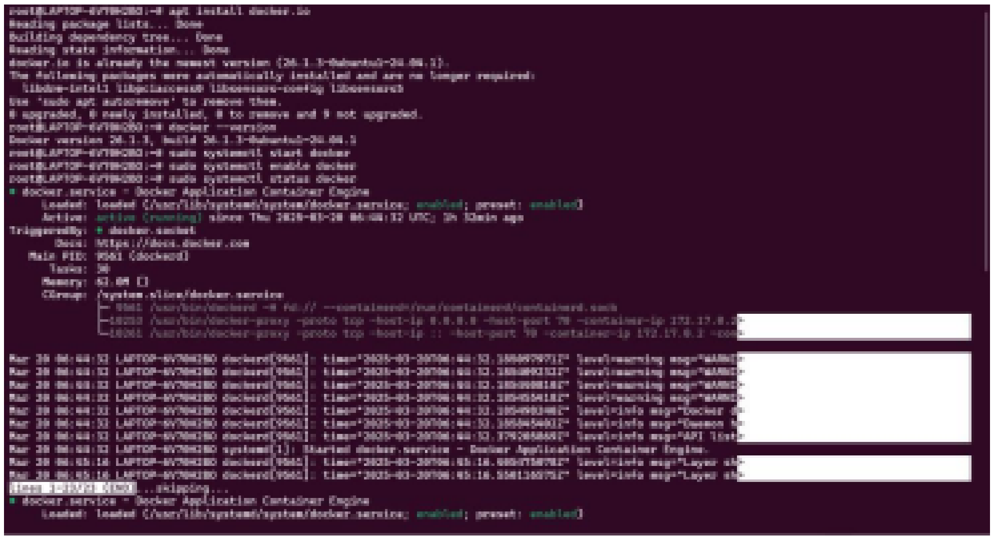
DEVOPS TASK- 2

1) Installation of Docker:

CODE :

sudo apt install docker.io
Docker –version
sudo systemctl start docker
sudo systemctl enable docker
sudo systemctl status docker

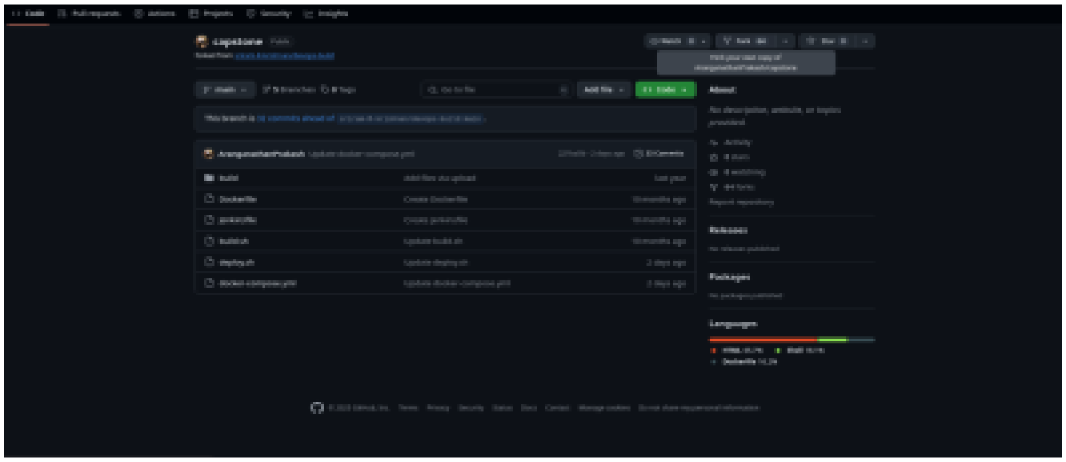
SCREENSHOT:

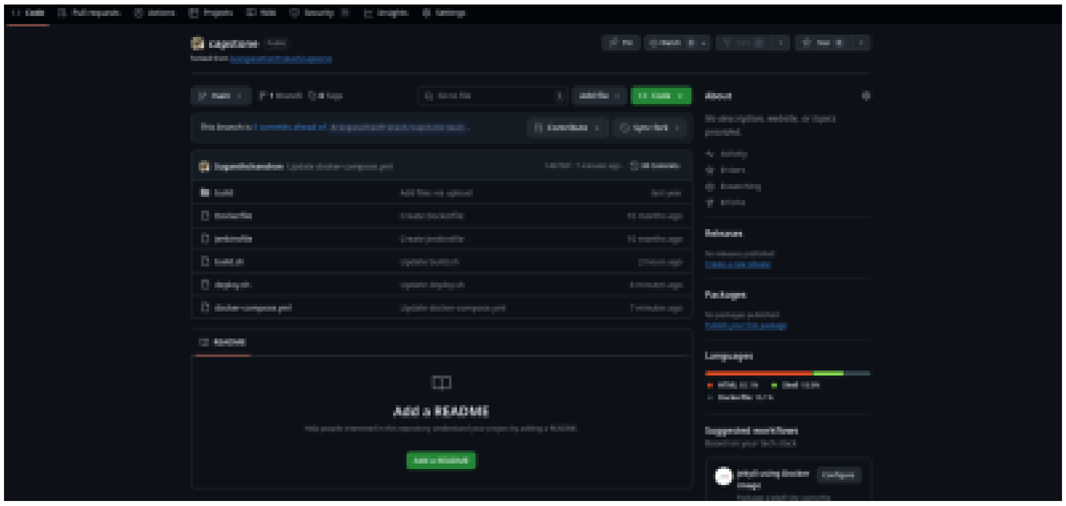


```
root@LAPTOP-[illegible]:~# apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
docker.io is already the newest version (26.1.3-0ubuntu1~24.04.1).
The following packages were automatically installed and are no longer required:
  libdrm-intel1 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-[illegible]:~# docker --version
Docker version 26.1.3, build 26.1.3-0ubuntu1~24.04.1
root@LAPTOP-[illegible]:~# sudo systemctl start docker
root@LAPTOP-[illegible]:~# sudo systemctl enable docker
root@LAPTOP-[illegible]:~# 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: 61.0M ()
     CGroup: /system.slice/docker.service
             ├─ 9561 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
             ├─18252 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0 -host-port 70 -container-ip 172.17.0.2>
             └─18261 /usr/bin/docker-proxy -proto tcp -host-ip :: -host-port 70 -container-ip 172.17.0.2 -con>

Mar 20 06:44:32 LAPTOP-[illegible] dockerd[9561]: time="2025-03-20T06:44:32.[illegible]" level=warning msg="WARNI>
Mar 20 06:44:32 LAPTOP-[illegible] dockerd[9561]: time="2025-03-20T06:44:32.[illegible]" level=warning msg="WARNI>
Mar 20 06:44:32 LAPTOP-[illegible] dockerd[9561]: time="2025-03-20T06:44:32.[illegible]" level=warning msg="WARNI>
Mar 20 06:44:32 LAPTOP-[illegible] dockerd[9561]: time="2025-03-20T06:44:32.[illegible]" level=warning msg="WARNI>
Mar 20 06:44:32 LAPTOP-[illegible] dockerd[9561]: time="2025-03-20T06:44:32.[illegible]" level=info msg="Docker d>
Mar 20 06:44:32 LAPTOP-[illegible] dockerd[9561]: time="2025-03-20T06:44:32.[illegible]" level=info msg="Daemon h>
Mar 20 06:44:32 LAPTOP-[illegible] dockerd[9561]: time="2025-03-20T06:44:32.[illegible]" level=info msg="API list>
Mar 20 06:44:32 LAPTOP-[illegible] systemd[1]: Started docker.service - Docker Application Container Engine.
Mar 20 06:45:16 LAPTOP-[illegible] dockerd[9561]: time="2025-03-20T06:45:16.[illegible]" level=info msg="Layer sh>
Mar 20 06:45:16 LAPTOP-[illegible] dockerd[9561]: time="2025-03-20T06:45:16.[illegible]" level=info msg="Layer sh>
lines 1-22/22 (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

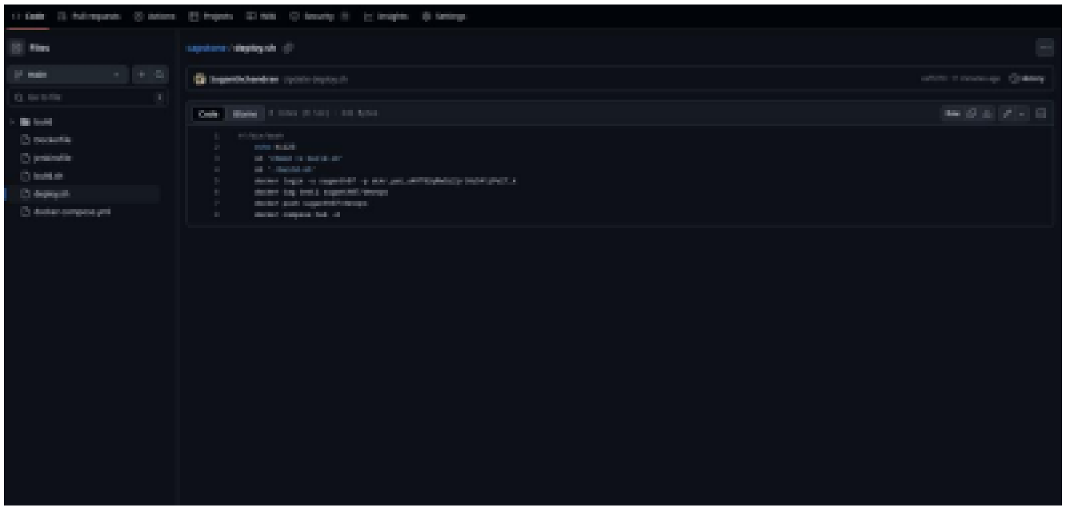
SCREENSHOT :





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

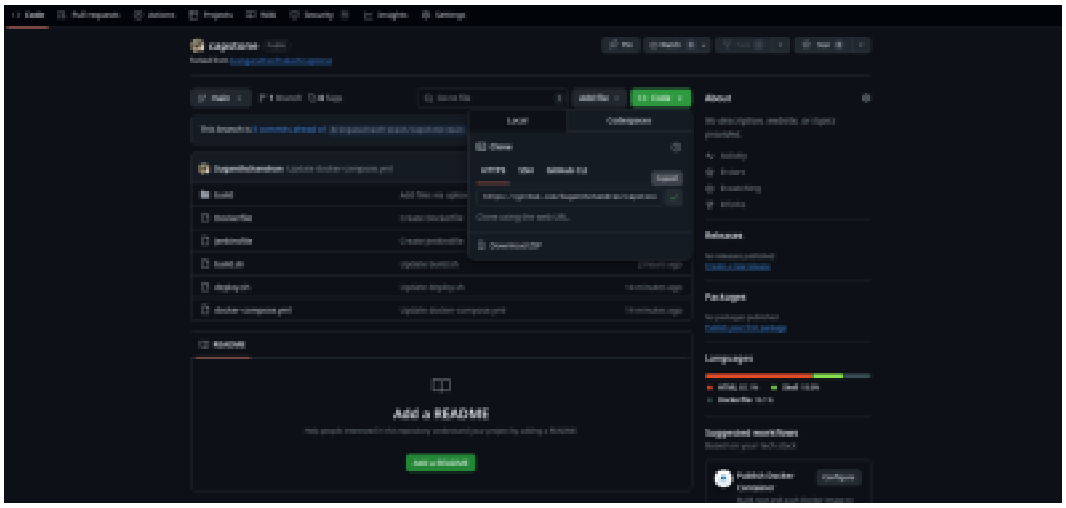
SCREENSHOT :



[illegible]

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

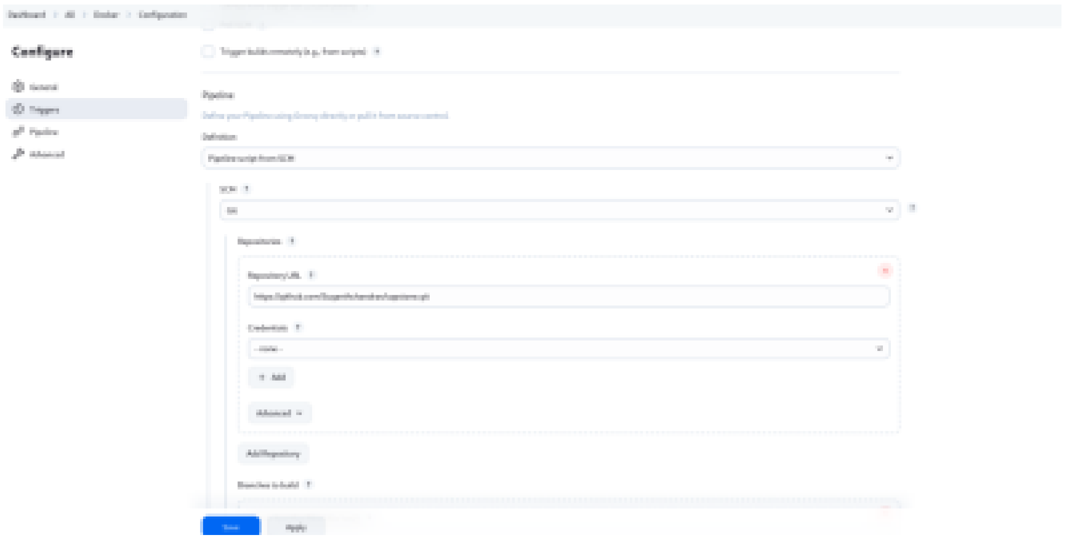
Jenkins. **SCREENSHOT:**



[illegible]

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:



Dashboard > All > Docker > Configuration

Configure

Trigger builds remotely (e.g., from scripts)

Pipeline

Definition: Pipeline script from SCM

SCM: Git

Repositories

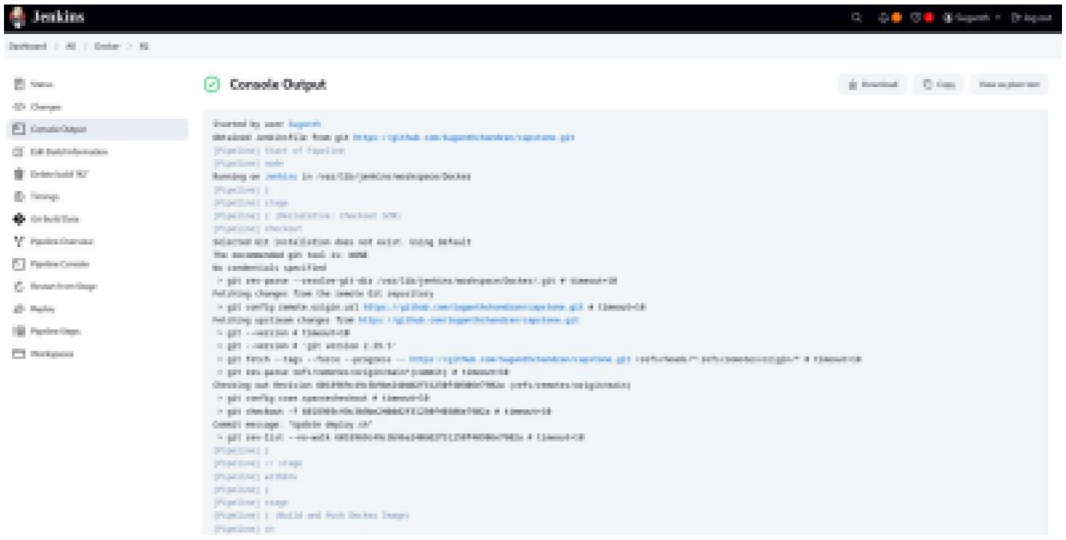
Repository URL: https://github.com/[illegible]/capstone.git

Credentials: - none -

Branches to build

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

SCREENSHOT:



Dashboard > All > Docker > #2

Console Output

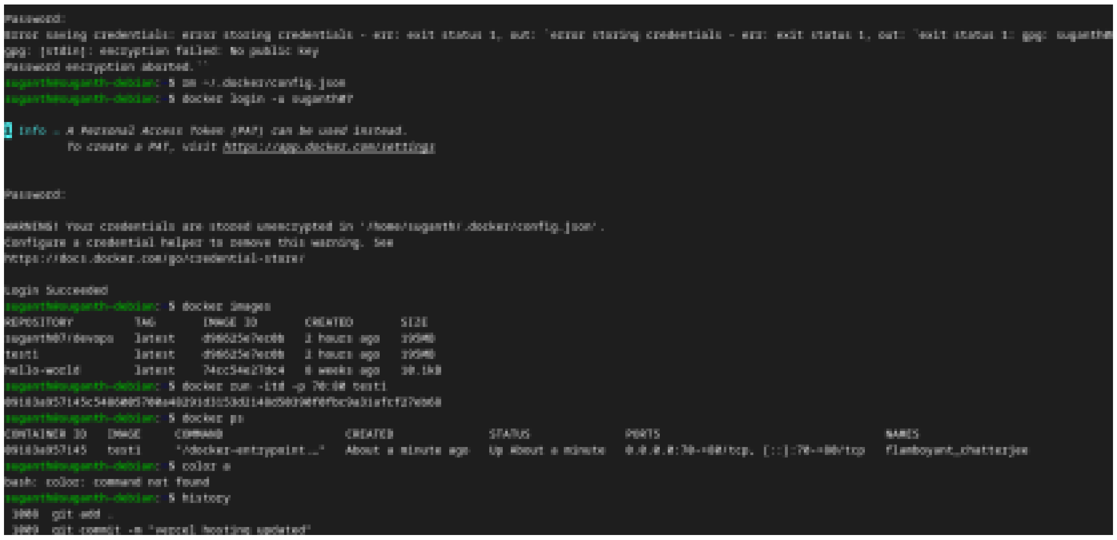
```
Started by user [illegible]
Obtained Jenkinsfile from git https://github.com/[illegible]/capstone.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Docker
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Docker/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/[illegible]/capstone.git # timeout=10
Fetching upstream changes from https://github.com/[illegible]/capstone.git
 > git --version # timeout=10
 > git --version # 'git version 2.[illegible]'
 > git fetch --tags --force --progress -- https://github.com/[illegible]/capstone.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision [illegible] (refs/remotes/origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f [illegible] # timeout=10
Commit message: "Update deploy.sh"
 > git rev-list --no-walk [illegible] # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Build and Push Docker Image)
[Pipeline] sh
```

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

CODE:

```
docker images

docker run –itd –p 70:80 test1
```

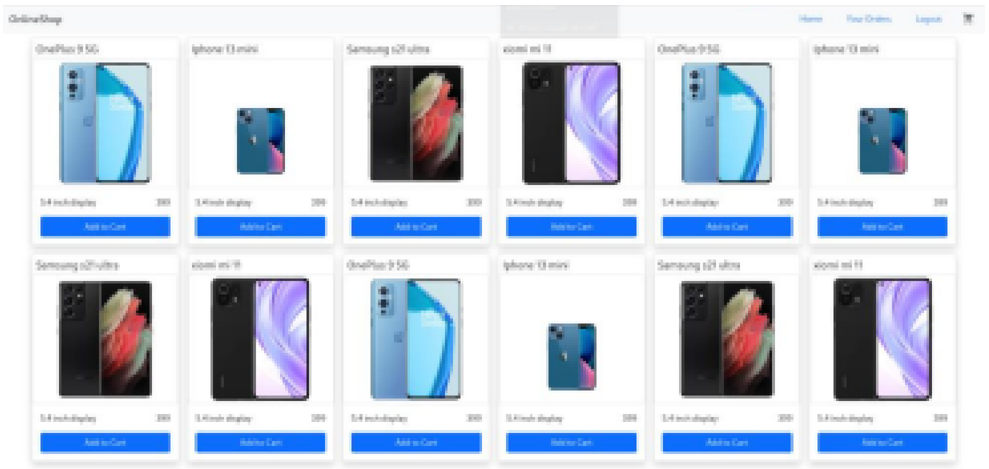
SCREENSHOT:



```
Password:
Error saving credentials: error storing credentials - err: exit status 1, out: `error storing credentials - err: exit status 1, out: `exit status 1: gpg: suganth
gpg: [stdin]: encryption failed: No public key
Password encryption aborted.``
suganth@suganth-debian:~$ rm ~/.docker/config.json
suganth@suganth-debian:~$ docker login -u suganth07

i Info → A Personal Access Token (PAT) can be used instead.
         To create a PAT, visit https://app.docker.com/settings


Password:

WARNING! Your credentials are stored unencrypted in '/home/suganth/.docker/config.json'.
Configure a credential helper to remove this warning. See
https://docs.docker.com/go/credential-store/

Login Succeeded
suganth@suganth-debian:~$ docker images
REPOSITORY         TAG       IMAGE ID       CREATED       SIZE
suganth07/devops   latest    d90625e7ec0b   2 hours ago   195MB
test1              latest    d90625e7ec0b   2 hours ago   195MB
hello-world        latest    74cc54e27dc4   6 weeks ago   10.1kB
suganth@suganth-debian:~$ docker run -itd -p 70:80 test1
09183a057145c5486005700a4029id3153d2148d503f0f0fbc9a31afcf27e068
suganth@suganth-debian:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED              STATUS              PORTS                                 NAMES
09183a057145   test1     "/docker-entrypoint.…"   About a minute ago   Up About a minute   0.0.0.0:70->80/tcp, [::]:70->80/tcp   flamboyant_chatterjee
suganth@suganth-debian:~$ color a
bash: color: command not found
suganth@suganth-debian:~$ history
 3008  git add .
 3009  git commit -m "vercel hosting updated"
```

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

SCREENSHOT:



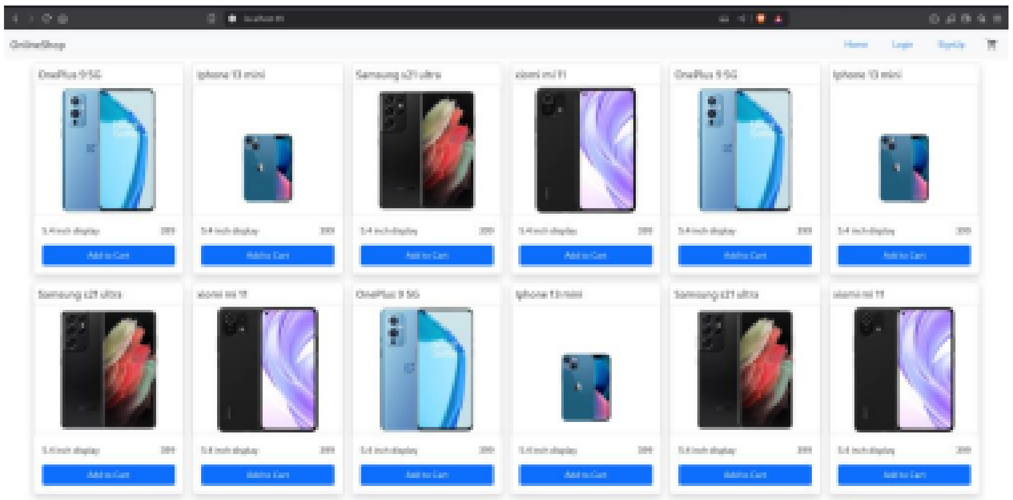
9) But, Instead of running the image by manually , we can also write the command for running in a file called docker-compose.yml

CODE:

```
version: '3'
services:
 react-capstone:
```

```
image: "test1"
ports:
-"85:80"
```

SCREENSHOT:



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