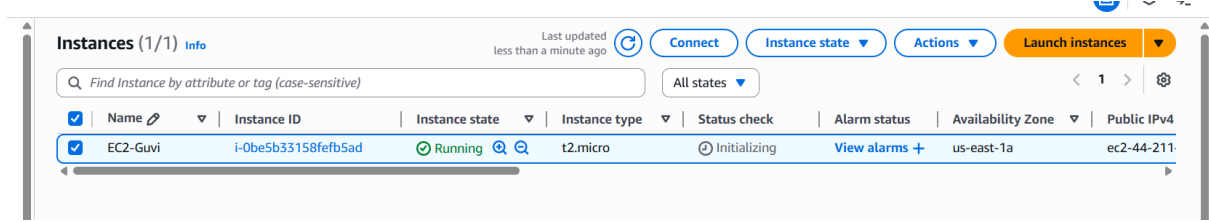


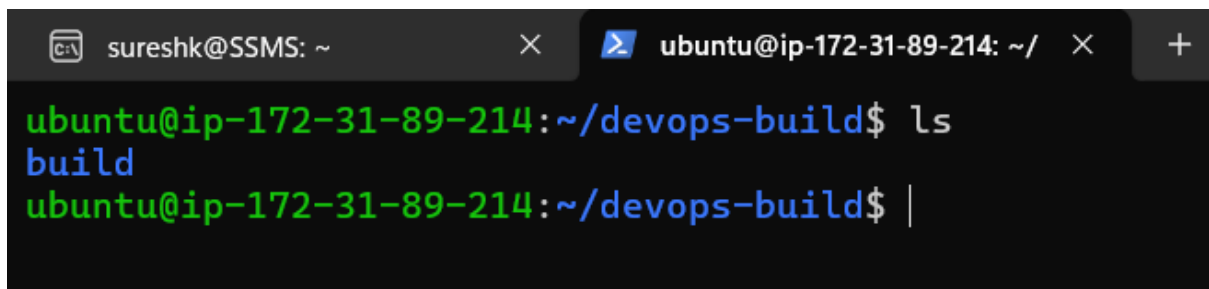
## Application Deployment

(Deploy the given react application to a production ready state)

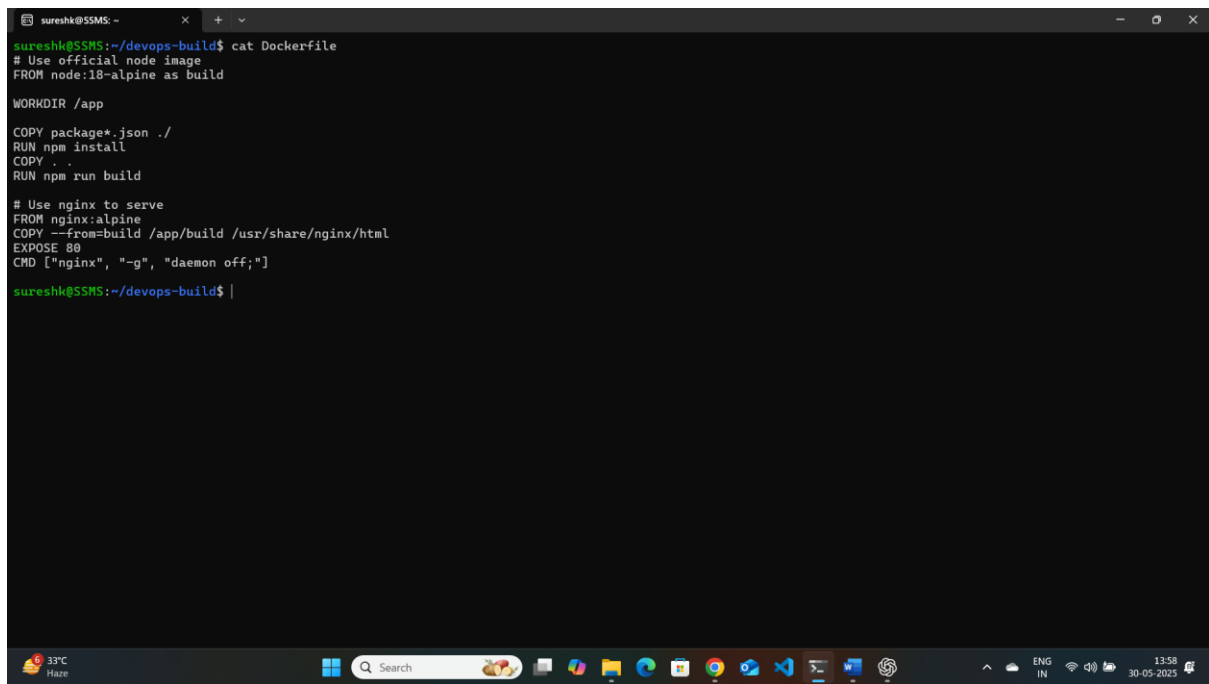
### Creating EC2 Instance:



### Cloning the Repo:



### Dockerization:



```
sureshk@SSMS: ~  
sureshk@SSMS:~/devops-build$ cat Dockerfile  
# Use official node image  
FROM node:18-alpine as build  
  
WORKDIR /app  
  
COPY package*.json ./  
RUN npm install  
COPY . .  
RUN npm run build  
  
# Use nginx to serve  
FROM nginx:alpine  
COPY --from=build /app/build /usr/share/nginx/html  
EXPOSE 80  
CMD ["nginx", "-g", "daemon off;"]  
  
sureshk@SSMS:~/devops-build$ vi .dockerignore  
sureshk@SSMS:~/devops-build$ cat .dockerignore  
node_modules  
build  
.git  
Dockerfile  
docker-compose.yml  
  
sureshk@SSMS:~/devops-build$ |
```

## Creating Docker Compose:

```
sureshk@SSMS: ~  
sureshk@SSMS:~/devops-build$ vi docker-compose.yml  
sureshk@SSMS:~/devops-build$ cat docker-compose.yml  
version: '3.8'  
  
services:  
  react-app:  
    build: .  
    ports:  
      - "80:80"  
  
sureshk@SSMS:~/devops-build$ |
```

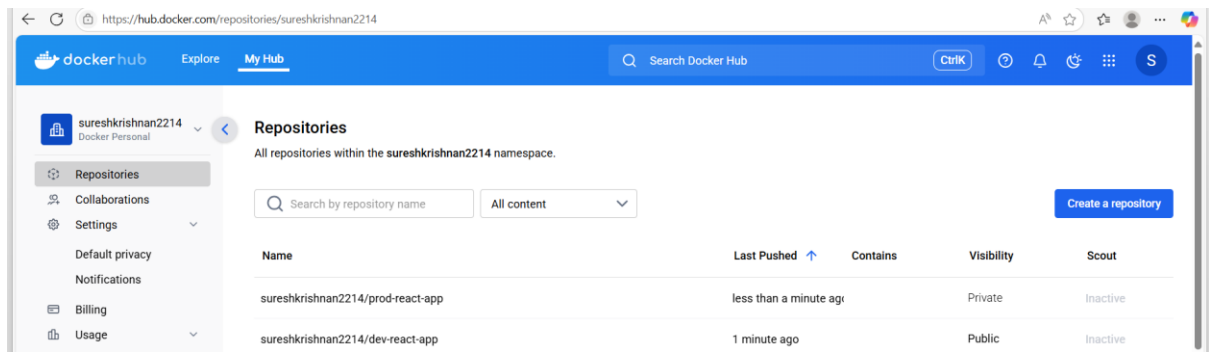
## Creating Bash Scripts:

```
sureshk@SSMS: ~  
sureshk@SSMS:~/devops-build$ vi build.sh  
sureshk@SSMS:~/devops-build$ vi deploy.sh  
sureshk@SSMS:~/devops-build$ cat build.sh deploy.sh  
#!/bin/bash  
echo "Building Docker image..."  
docker build -t react-app:latest .  
  
#!/bin/bash  
echo "Deploying Docker container..."  
docker-compose down  
docker-compose up -d --build  
  
sureshk@SSMS:~/devops-build$ |
```

## GitHub Version Control:

```
sureshk@SSMS: ~  
sureshk@SSMS:~/devops-build$ git init  
Reinitialized existing Git repository in /home/sureshk/devops-build/.git/  
sureshk@SSMS:~/devops-build$ git remote add origin https://github.com/Sureshkrishnan2214/Guvi-Finalproject.git  
sureshk@SSMS:~/devops-build$ git checkout -b dev  
fatal: a branch named 'dev' already exists  
sureshk@SSMS:~/devops-build$ echo "node_modules/" > .gitignore  
echo ".env" >> .gitignore  
sureshk@SSMS:~/devops-build$ git add .  
sureshk@SSMS:~/devops-build$ git commit -m "Initial commit"  
[detached HEAD e6885a8] Initial commit  
10 files changed, 19669 insertions(+), 4 deletions(-)  
create mode 100644 devops-build@1.0.0  
create mode 100644 package-lock.json  
create mode 100644 package.json  
create mode 100644 react-scripts  
sureshk@SSMS:~/devops-build$ git push origin dev  
Username for 'https://github.com': sureshkrishnan2214  
Password for 'https://sureshkrishnan2214@github.com':  
Enumerating objects: 46, done.  
Counting objects: 100% (46/46), done.  
Delta compression using up to 18 threads  
Compressing objects: 100% (41/41), done.  
Writing objects: 100% (46/46), 980.28 KiB | 24.33 MiB/s, done.  
Total 46 (delta 2), reused 21 (delta 0), pack-reused 0  
remote: Resolving deltas: 100% (2/2), done.  
To https://github.com/Sureshkrishnan2214/Guvi-Finalproject.git  
* [new branch] dev -> dev  
sureshk@SSMS:~/devops-build$ |
```

## Creating Docker Hub:



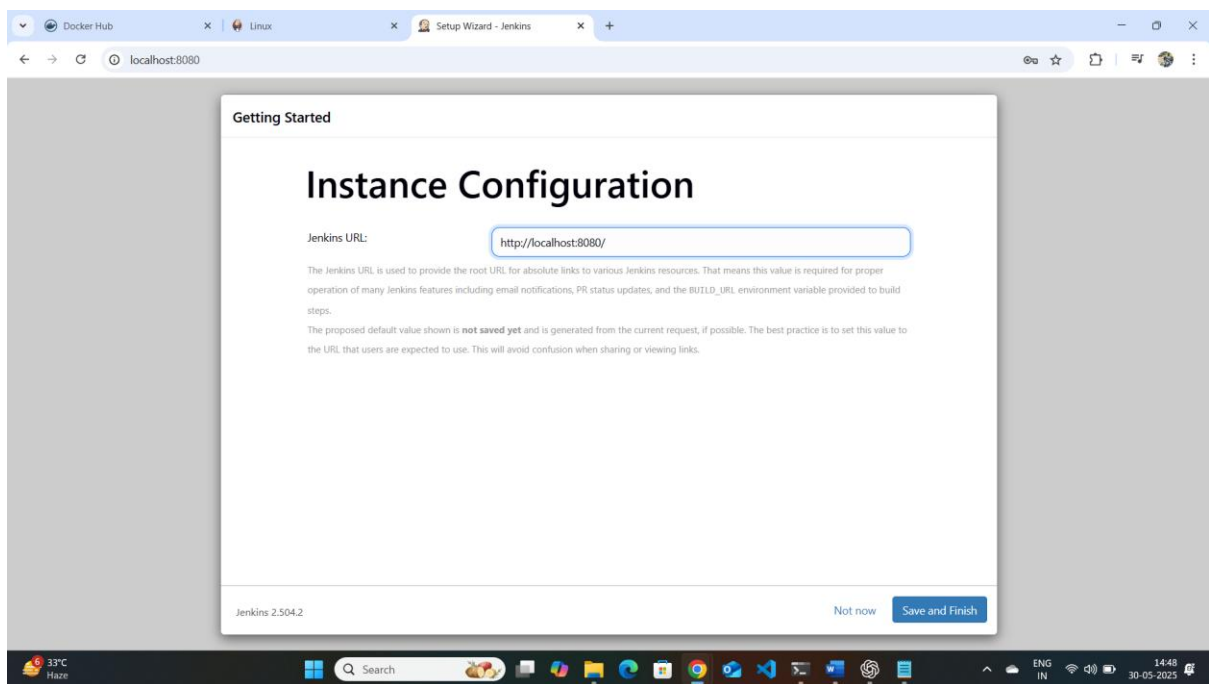
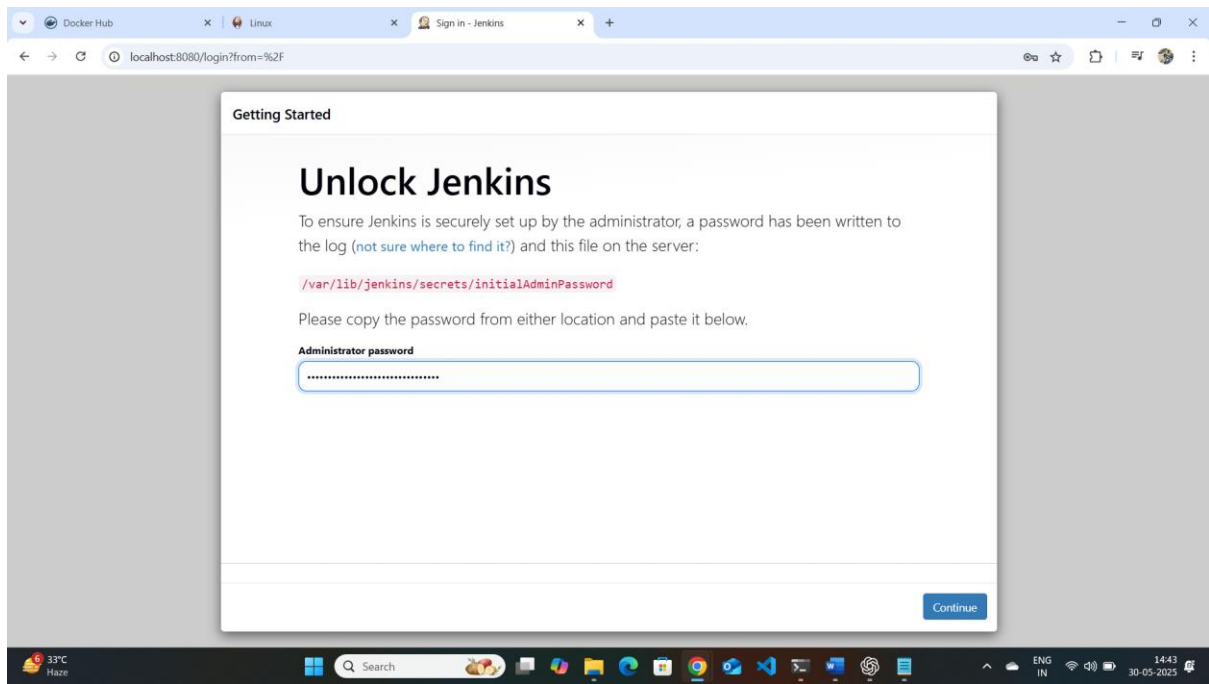
## Installing Jenkins:

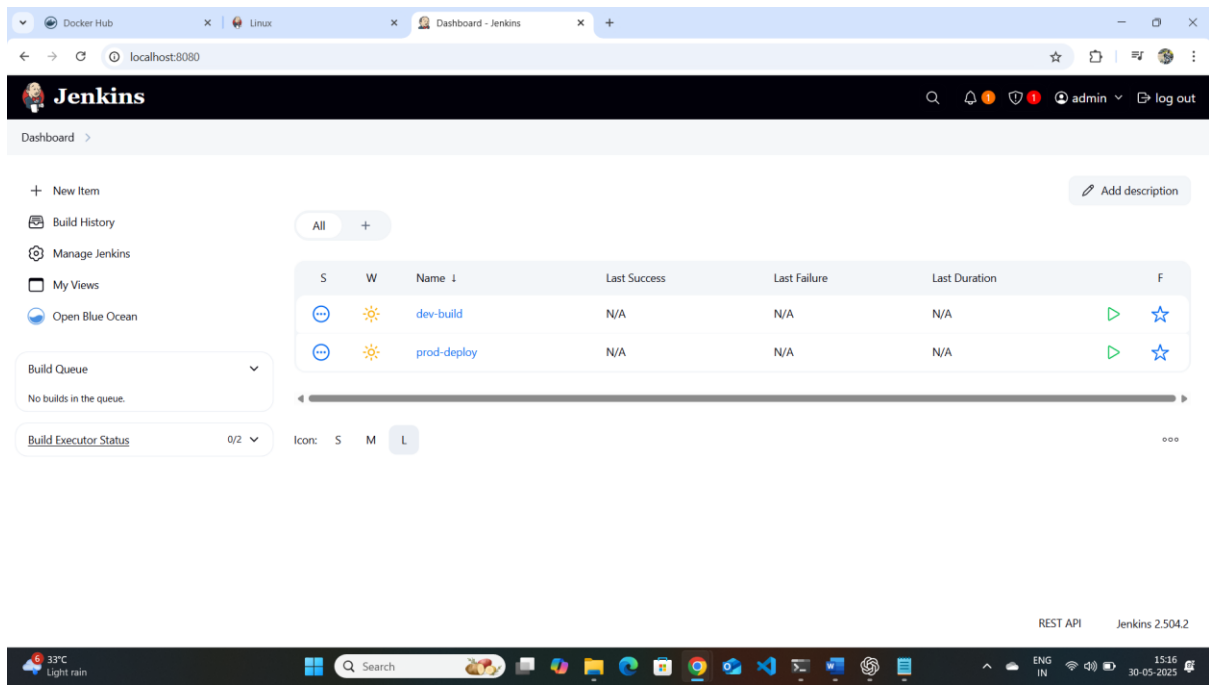
```
sureshk@SSMS: ~  
sureshk@SSMS:~/devops-build$ java -version  
openjdk version "21.0.7" 2025-04-15  
OpenJDK Runtime Environment (build 21.0.7+6-Ubuntu-0ubuntu124.04)  
OpenJDK 64-Bit Server VM (build 21.0.7+6-Ubuntu-0ubuntu124.04, mixed mode, sharing)  
sureshk@SSMS:~/devops-build$
```

```
sureshk@SSMS:~/devops-build$ sudo wget -O /etc/apt/keyrings/jenkins-keyring.asc \  
https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key  
echo "deb [signed-by=/etc/apt/keyrings/jenkins-keyring.asc] \  
https://pkg.jenkins.io/debian-stable binary/" | sudo tee \  
/etc/apt/sources.list.d/jenkins.list > /dev/null  
--2025-05-30 14:33:40-- https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key  
Resolving pkg.jenkins.io (pkg.jenkins.io)... 151.101.158.133, 2a04:4e42:25::645  
Connecting to pkg.jenkins.io (pkg.jenkins.io)[151.101.158.133]:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 3175 (3.1K) [application/pgp-keys]  
Saving to: '/etc/apt/keyrings/jenkins-keyring.asc'  
  
/etc/apt/keyrings/jenkins-keyring.asc 100%[=====] 3.10K --.-KB/s in 0s  
2025-05-30 14:33:40 (20.5 MB/s) - '/etc/apt/keyrings/jenkins-keyring.asc' saved [3175/3175]  
sureshk@SSMS:~/devops-build$
```

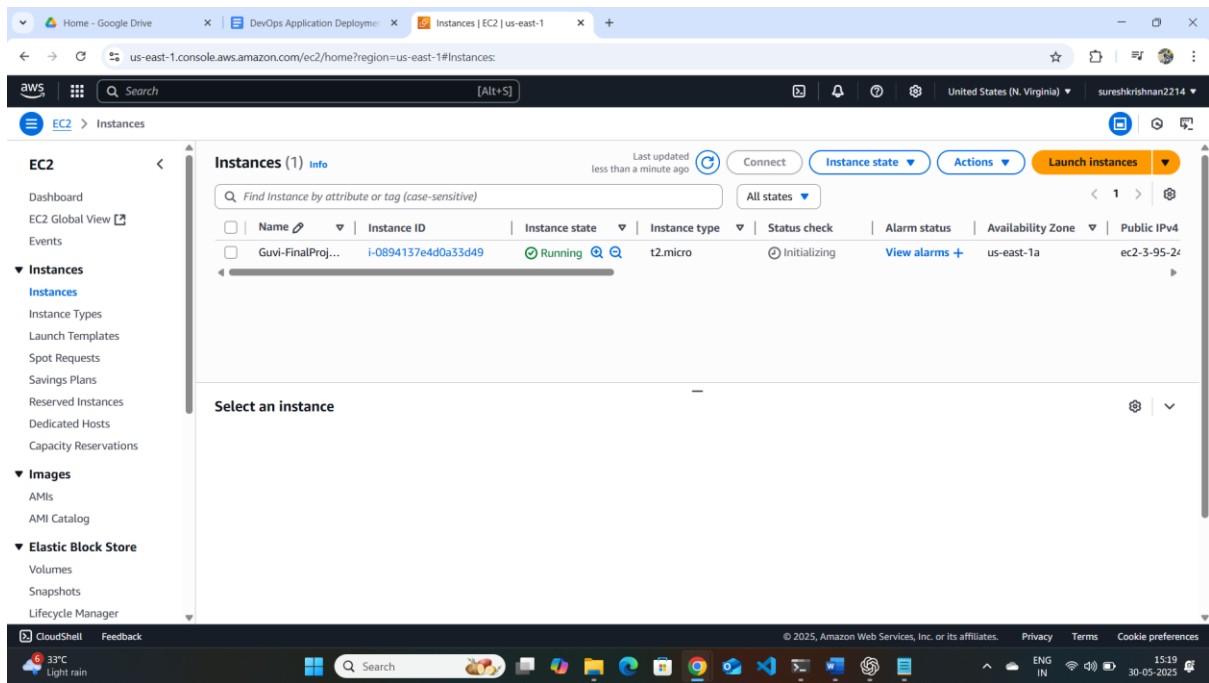
```
sureshk@SSMS:~/devops-build$ sudo apt-get install jenkins  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following packages were automatically installed and are no longer required:  
  bridge-utils dns-root-data dnsmasq-base libllvm17t64 ubuntu-fan  
Use 'sudo apt autoremove' to remove them.  
The following NEW packages will be installed:  
  jenkins  
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.  
Need to get 92.2 MB of archives.  
After this operation, 94.4 MB of additional disk space will be used.  
Get:1 https://pkg.jenkins.io/debian-stable binary/ jenkins 2.504.2 [92.2 MB]  
Fetched 92.2 MB in 38s (2398 kB/s)  
Selecting previously unselected package jenkins.  
(Reading database ... 108969 files and directories currently installed.)  
Preparing to unpack .../jenkins_2.504.2_all.deb ...  
Unpacking jenkins (2.504.2) ...  
Setting up jenkins (2.504.2) ...  
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /usr/lib/systemd/system/jenkins.service.  
sureshk@SSMS:~/devops-build$
```

```
sureshk@SSMS: ~  
sureshk@SSMS:~/devops-build$ sudo systemctl enable jenkins  
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.  
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins  
sureshk@SSMS:~/devops-build$ sudo systemctl start jenkins  
sureshk@SSMS:~/devops-build$
```





## Creating EC2 Instance:



## Installing Docker(Inside the EC2):

```
sureshk@SSMS: ~
x ubuntu@ip-172-31-91-152: ~
+ v
Unpacking docker.io (27.5.1-0ubuntu3-24.04.1) ...
Selecting previously unselected package ubuntu-fan.
Preparing to unpack .../15-ubuntu-fan-0.12.16_all.deb ...
Unpacking ubuntu-fan (0.12.16) ...
Setting up python3-dotenv (1.0.1-1) ...
Setting up python3-texttable (1.6.7-1) ...
Setting up python3-docopt (0.6.2-6) ...
Setting up dnsmasq-base (2.90-2build2) ...
Setting up runc (1.2.5-0ubuntu1-24.04.1) ...
Setting up dns-root-data (2024071801-ubuntu0.24.04.1) ...
Setting up bridge-utils (1.7.1-1ubuntu2) ...
Setting up pigz (2.8-1) ...
Setting up containerd (1.7.27-0ubuntu1-24.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /usr/lib/systemd/system/containerd.service.
Setting up python3-websocket (1.7.0-1) ...
Setting up python3-dockerpty (0.4.1-5) ...
Setting up ubuntu-fan (0.12.16) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /usr/lib/systemd/system/ubuntu-fan.service.
Setting up python3-docker (5.0.3-1ubuntu1.1) ...
Setting up docker.io (27.5.1-0ubuntu3-24.04.1) ...
info: Selecting GID from range 100 to 999 ...
info: Adding group 'docker' (GID 113) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.
Setting up python3-compose (1.29.2-6ubuntu1) ...
Setting up docker-compose (1.29.2-6ubuntu1) ...
Processing triggers for dbus (1.14.10-4ubuntu4.1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-91-152:~$
```

```
sureshk@SSMS: ~
x ubuntu@ip-172-31-91-152: ~
+ v
Preparing to unpack .../15-ubuntu-fan-0.12.16_all.deb ...
Unpacking ubuntu-fan (0.12.16) ...
Setting up python3-dotenv (1.0.1-1) ...
Setting up python3-texttable (1.6.7-1) ...
Setting up python3-docopt (0.6.2-6) ...
Setting up dnsmasq-base (2.90-2build2) ...
Setting up runc (1.2.5-0ubuntu1-24.04.1) ...
Setting up dns-root-data (2024071801-ubuntu0.24.04.1) ...
Setting up bridge-utils (1.7.1-1ubuntu2) ...
Setting up pigz (2.8-1) ...
Setting up containerd (1.7.27-0ubuntu1-24.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /usr/lib/systemd/system/containerd.service.
Setting up python3-websocket (1.7.0-1) ...
Setting up python3-dockerpty (0.4.1-5) ...
Setting up ubuntu-fan (0.12.16) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /usr/lib/systemd/system/ubuntu-fan.service.
Setting up python3-docker (5.0.3-1ubuntu1.1) ...
Setting up docker.io (27.5.1-0ubuntu3-24.04.1) ...
info: Selecting GID from range 100 to 999 ...
info: Adding group 'docker' (GID 113) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.
Setting up python3-compose (1.29.2-6ubuntu1) ...
Setting up docker-compose (1.29.2-6ubuntu1) ...
Processing triggers for dbus (1.14.10-4ubuntu4.1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-91-152:~$ sudo systemctl start docker
ubuntu@ip-172-31-91-152:~$ sudo usermod -aG docker $USER
ubuntu@ip-172-31-91-152:~$
```

## Cloning the Repo inside the EC2:

```
sureshk@SSMS: ~
ubuntu@ip-172-31-91-152: ~$ cd ~
ubuntu@ip-172-31-91-152:~$ git clone https://github.com/Sureshkrishnan2214/devops-build.git
Cloning into 'devops-build'...
remote: Enumerating objects: 29, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 29 (delta 0), reused 5 (delta 0), pack-reused 24 (from 3)
Receiving objects: 100% (29/29), 721.01 KiB | 18.49 MiB/s, done.
ubuntu@ip-172-31-91-152:~$
```

33°C Light rain 15:24 30-05-2025

```
sureshk@SSMS: ~
ubuntu@ip-172-31-91-152: ~$ git clone https://github.com/Sureshkrishnan2214/Guvi-Finalproject.git
Cloning into 'Guvi-Finalproject'...
remote: Enumerating objects: 46, done.
remote: Counting objects: 100% (46/46), done.
remote: Compressing objects: 100% (39/39), done.
remote: Total 46 (delta 2), reused 46 (delta 2), pack-reused 0 (from 0)
Receiving objects: 100% (46/46), 900.28 KiB | 20.01 MiB/s, done.
Resolving deltas: 100% (2/2), done.
ubuntu@ip-172-31-91-152:~$
```

33°C Light rain 15:26 30-05-2025



```
sureshk@SSMS: ~
ubuntu@ip-172-31-91-152: ~/Guvi-Finalproject$ ./deploy.sh
Deploying Docker container...
Removing network guvi-finalproject_default
WARNING: Network guvi-finalproject_default not found,
Creating network "guvi-finalproject_default" with the default driver
Building react-app
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/

Sending build context to Docker daemon 761.9kB
Step 1/10 : FROM node:18-alpine as build
18-alpine: Pulling from library/node
f18232174bc9: Pull complete
dd71dde834b5: Pull complete
1e5a4c89cee5: Pull complete
25ff2da83641: Pull complete
Digest: sha256:8d6421d663b4c28fd3ebc498332f249011d118945588d0a35cb9bc4b8ca09d9e
Status: Downloaded newer image for node:18-alpine
--> ee77c6cd7c18
Step 2/10 : WORKDIR /app
--> Running in 6c79a16cdf8
--> Removed intermediate container 6c79a16cdf8
--> 5683b2041be
Step 3/10 : COPY package*.json ./
--> 470542beed3d
Step 4/10 : RUN npm install
--> Running in bb8ccd3a7962
```

```
sureshk@SSMS: ~
ubuntu@ip-172-31-91-152: ~/Guvi-Finalproject$
The project was built assuming it is hosted at /.
You can control this with the homepage field in your package.json.

The build folder is ready to be deployed.
You may serve it with a static server:

  npm install -g serve
  serve -s build

Find out more about deployment here:

  bit.ly/CRA-deploy

--> Removed intermediate container 27b52ac35845
--> 105004c2d734
Step 8/11 : FROM nginx:alpine
alpine: Pulling from library/nginx
f18232174bc9: Already exists
61ca4f73c80: Pull complete
b464cfdf2a63: Pull complete
d7e507024086: Pull complete
81bd8ed7ec67: Pull complete
197eb75867ef: Pull complete
34a64644b756: Pull complete
39c2ddfd6019: Pull complete
Digest: sha256:65645c7bb6a0661892a8b03b89d0743208a18dd2f3f17a54ef4b76fb8e2f2a10
Status: Downloaded newer image for nginx:alpine
--> 6769dc3a703c
Step 9/11 : COPY --from=build /app/build /usr/share/nginx/html
--> 6c6a3fbedc1c
Step 10/11 : EXPOSE 80
--> Running in 19f813394b3d
--> Removed intermediate container 19f813394b3d
--> 6eafdad98cb0
Step 11/11 : CMD ["nginx", "-g", "daemon off;"]
--> Running in 86a05e825194
--> Removed intermediate container 86a05e825194
--> 6633dd38a2e9
Successfully built 6633dd38a2e9
Successfully tagged guvi-finalproject_react-app:latest
ubuntu@ip-172-31-91-152: ~/Guvi-Finalproject$
```

```

ubuntu@ip-172-31-91-152:~/Guvi-Finalproject$ docker volume create uptime-kuma
uptime-kuma
ubuntu@ip-172-31-91-152:~/Guvi-Finalproject$ docker run -d \
  --restart=always \
  -p 3001:3001 \
  -v uptime-kuma:/app/data \
  --name uptime-kuma \
  louislam/uptime-kuma
Unable to find image 'louislam/uptime-kuma:latest' locally
latest: Pulling from louislam/uptime-kuma
b338562f40a7: Pull complete
874bf4d93720: Pull complete
b16337721583: Pull complete
7d955db85b85: Pull complete
2c706596bd17: Pull complete
88a5c59ed14f: Pull complete
5a1d0a896c33: Pull complete
e68c2f25b946: Pull complete
2e6c90f010d6: Pull complete
ff15b10fabb8: Pull complete
4f4fb700ef54: Pull complete
d2a400cc8adb: Pull complete
Digest: sha256:431fee3be822b04861cf0e35daf4beef6b7cb37391c5f26c3ad6e12ce280fe18
Status: Downloaded newer image for louislam/uptime-kuma:latest
326ef775dcb594367dfaa57766a20f6d90b2e1c9a97ca97db945c7d1ee807fc7
ubuntu@ip-172-31-91-152:~/Guvi-Finalproject$ |

```

The screenshot shows the Uptime Kuma web interface in a browser. The address bar shows the URL `3.95.244.138:3001/add`. The page title is "Add New Monitor". On the left, there is a sidebar with a search bar and filters for Status, Active, and Tags. The main content area is divided into three sections: General, Notifications, and HTTP Options.

**General**

- Monitor Type: HTTP(s)
- Friendly Name: My React App
- URL: `http://3.95.244.138:80`
- Heartbeat Interval (Check every 60 seconds): 60
- Retries: 0
- Maximum retries before the service is marked as down and a notification is sent
- Heartbeat Retry Interval (Retry every 60 seconds): 60
- Save button (Timeout after 48 seconds)

**Notifications**

- Not available, please setup.
- Setup Notification button

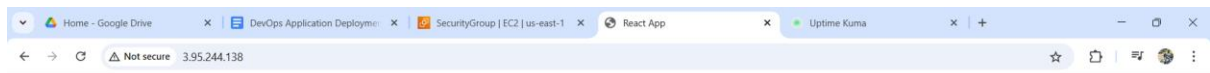
**Proxy**

- Not available, please setup.
- Setup Proxy button

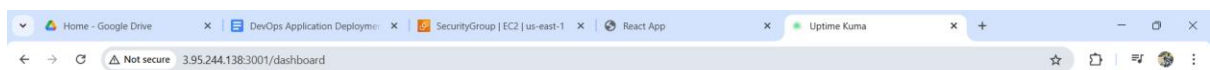
**HTTP Options**

- Method: GET
- Body Encoding: JSON
- Body: Example: `{ "key": "value" }`

The bottom of the image shows a Windows taskbar with the date and time set to 15:48 on 30-05-2023.



Hello from React!



Uptime Kuma

Status Pages Dashboard

+ Add New Monitor

Select Search...

Status Active Tags

3.29% My React App

### Quick Stats

Up	Down	Maintenance	Unknown	Pause
1	0	0	0	0

Name	Status	DateTime	Message
My React App	Up	2025-05-30 16:04:35	200 - OK
My React App	Down	2025-05-30 15:48:50	connect ECONNREFUSED 3.95.244.138:80



