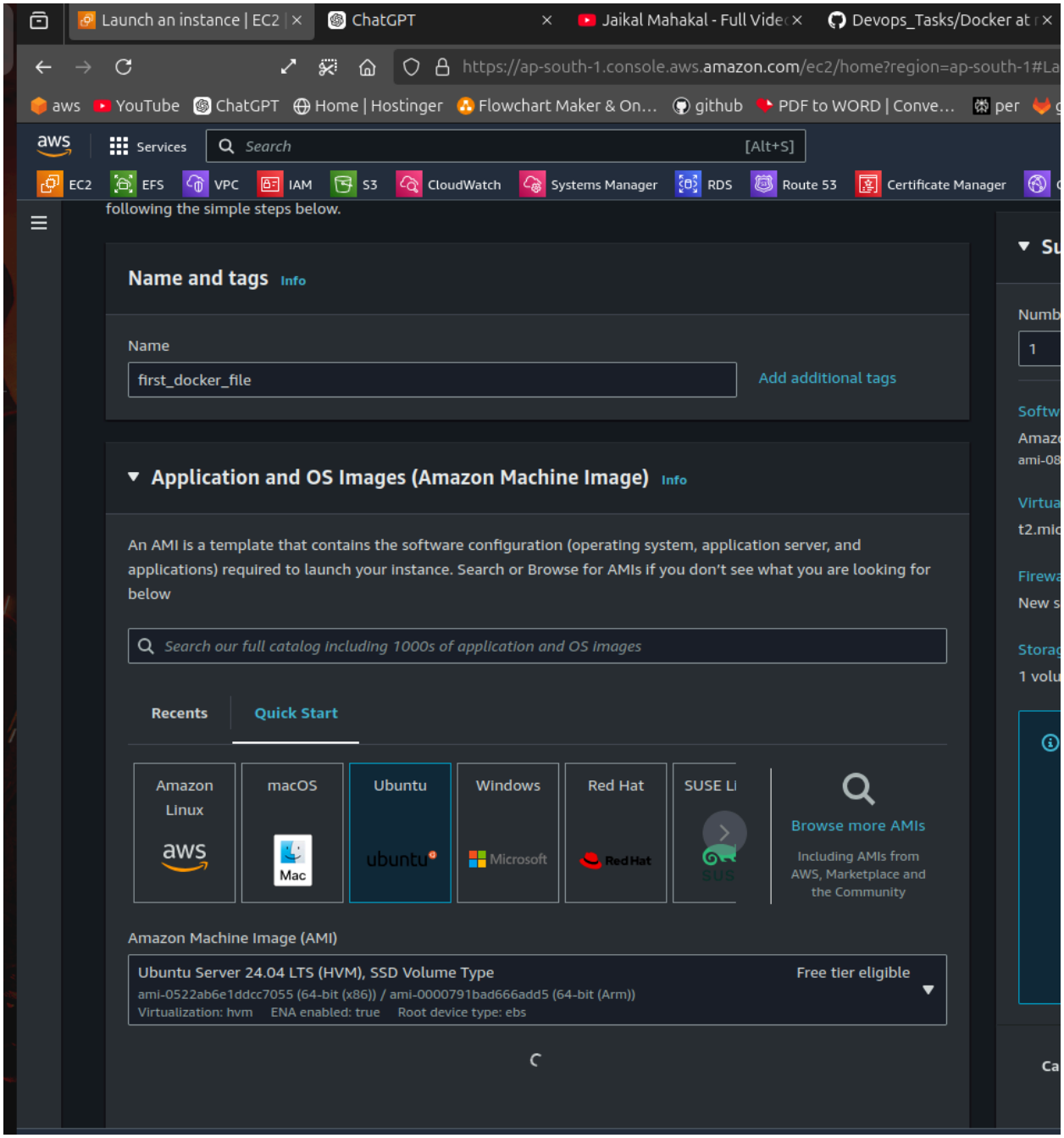
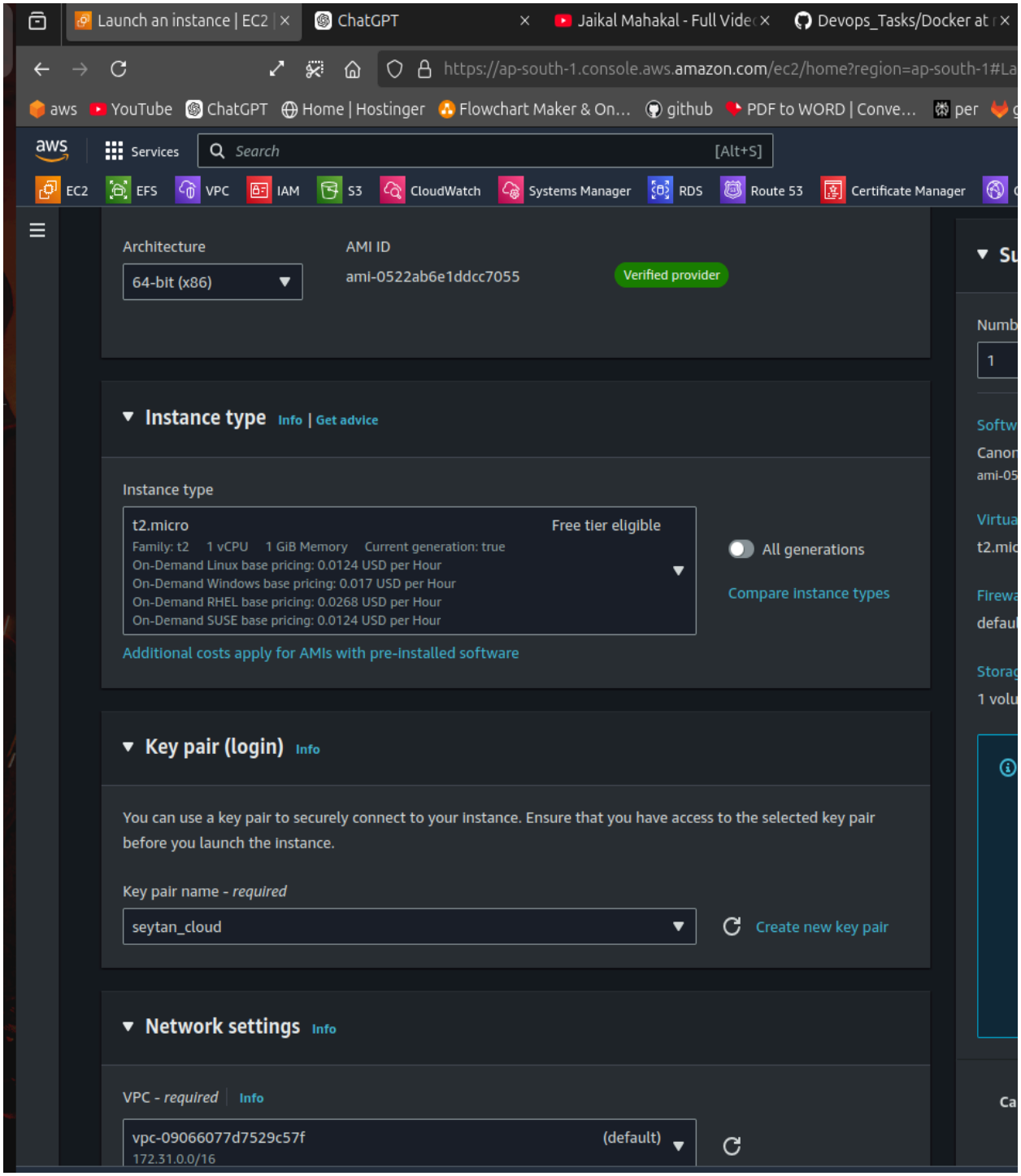
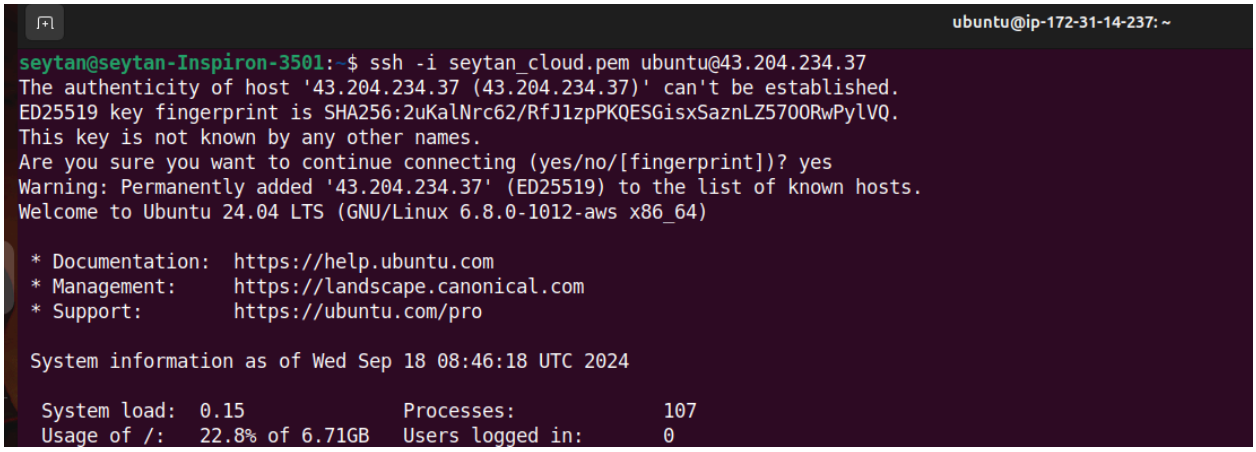
Task : Creating_First_Dockerfile

Step 1: Create and ec2 instance




Launch an instance | EC2
ChatGPT
Jaikal Mahakal - Full Vide
Devops_Tasks/Docker at
https://ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#La
aws
YouTube
ChatGPT
Home | Hostinger
Flowchart Maker & On...
github
PDF to WORD | Conve...
per
Services
Search
[Alt+S]
EC2
EFS
VPC
IAM
S3
CloudWatch
Systems Manager
RDS
Route 53
Certificate Manager
Architecture
64-bit (x86)
AMI ID
ami-0522ab6e1ddcc7055
Verified provider
Instance type
Info | Get advice
Instance type
t2.micro
Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0268 USD per Hour
On-Demand SUSE base pricing: 0.0124 USD per Hour
All generations
Compare instance types
Additional costs apply for AMIs with pre-installed software
Key pair (login)
Info
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.
Key pair name - required
seytan_cloud
Create new key pair
Network settings
Info
VPC - required
Info
vpc-09066077d7529c57f
(default)
172.31.0.0/16

Step 2: Take ssh of that instance .



Step 3: Create a Dockerfile and then build the docker image.
Command : docker build -t first_docker .

What is docker file?

→A Dockerfile is a text file with instructions to build a Docker image, defining the base image, commands to run, and files to include. It ensures consistent environments and simplifies deployment of containerized applications.

```
ubuntu@ip-172-31-14-237:~$ nano Dockerfile
ubuntu@ip-172-31-14-237:~$ sudo docker build -t first_docker .
[+] Building 3.6s (9/9) FINISHED                                                                                          docker:default
 => [internal] load build definition from Dockerfile                                                                                0.0s
 => => transferring dockerfile: 453B                                                                                                0.0s
 => [internal] load metadata for docker.io/library/nginx:latest                                                                     0.6s
 => [internal] load .dockerignore                                                                                                   0.0s
 => => transferring context: 2B                                                                                                     0.0s
 => [1/5] FROM docker.io/library/nginx:latest@sha256:04ba374043ccd2fc5c593885c0eacddebabd5ca375f9323666f28dfd5a9710e3               0.0s
 => CACHED [2/5] RUN apt-get update &&     apt-get install -y unzip wget                                                            0.0s
 => [3/5] RUN wget -O /usr/share/nginx/html/restoran.zip https://www.free-css.com/assets/files/free-css-templates/download/page290/restoran.zip     && unzip /usr/share/nginx/ht  1.9s
 => [4/5] WORKDIR /usr/share/nginx/html/                                                                                            0.1s
 => [5/5] RUN mv bootstrap-restaurant-template/* ./                                                                                 0.6s
 => exporting to image                                                                                                              0.3s
 => => exporting layers                                                                                                             0.2s
 => => writing image sha256:eeca8874ed364f89f7c36b07e4a199d5b15afa4964826fcf2555393a85e99776                                        0.0s
 => => naming to docker.io/library/first_docker                                                                                     0.0s
ubuntu@ip-172-31-14-237:~$ _
```

```
  GNU nano 7.2                                          Dockerfile
FROM nginx:latest


RUN apt-get update && \
    apt-get install -y unzip wget

RUN wget -O /usr/share/nginx/html/restoran.zip https://www.free-css.com/assets/files/free-css-templates/download/page290/restoran.zip \
    && unzip /usr/share/nginx/html/restoran.zip -d /usr/share/nginx/html/

WORKDIR /usr/share/nginx/html/
RUN mv bootstrap-restaurant-template/* ./



EXPOSE 80

CMD ["nginx", "-g", "daemon off;"]
```

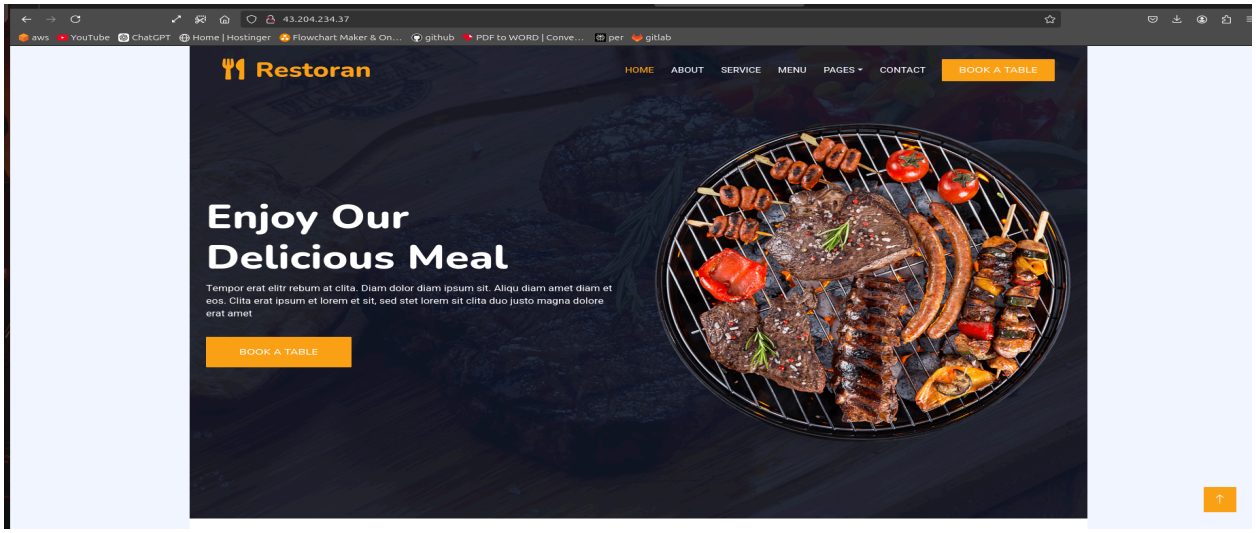
Step 4: Now verify the build image and run the container using following command:

docker run -d -p 80:80 image_id

Once done check container status using →docker ps

```
ubuntu@ip-172-31-14-237:~$ sudo docker images
REPOSITORY     TAG       IMAGE ID       CREATED          SIZE
first_docker   latest    eeca8874ed36   34 seconds ago   216MB
ubuntu@ip-172-31-14-237:~$ sudo docker run -d -p 80:80 eeca
00669db3d6945e501afb4bb32d184631c4b4eb813bf4c2330a9c6c01ee14f5d1
ubuntu@ip-172-31-14-237:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                                 NAMES
00669db3d694   eeca      "/docker-entrypoint.…"   5 seconds ago   Up 4 seconds   0.0.0.0:80->80/tcp, :::80->80/tcp   admiring_ride
ubuntu@ip-172-31-14-237:~$ _
```

Output: Now go and paste public ip in web browser and see the output.



Conclusion: By following these steps, you have created and run a Docker container for a simple application. The Dockerfile you created is a text document that contains all the commands to assemble the Docker image.