***Kavitha\_Capstone Project***

**Step 1:** Installation of required tools for the project.

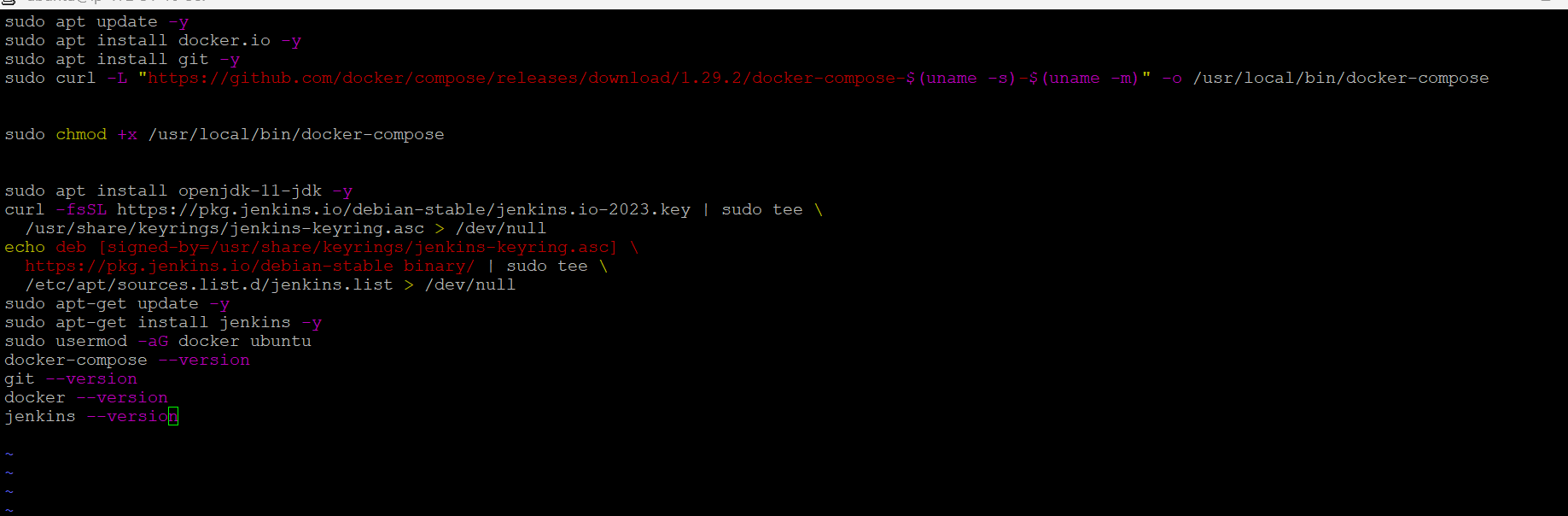
**Install.sh** - Created a script file to install the below tools

1. Git
2. Docker
3. Docker-Compose
4. Jenkins
5. Cloudwatch
6. Dockerfile
7. Docker-compose file
8. Git Hub - URL
9. Docker Hub - URL

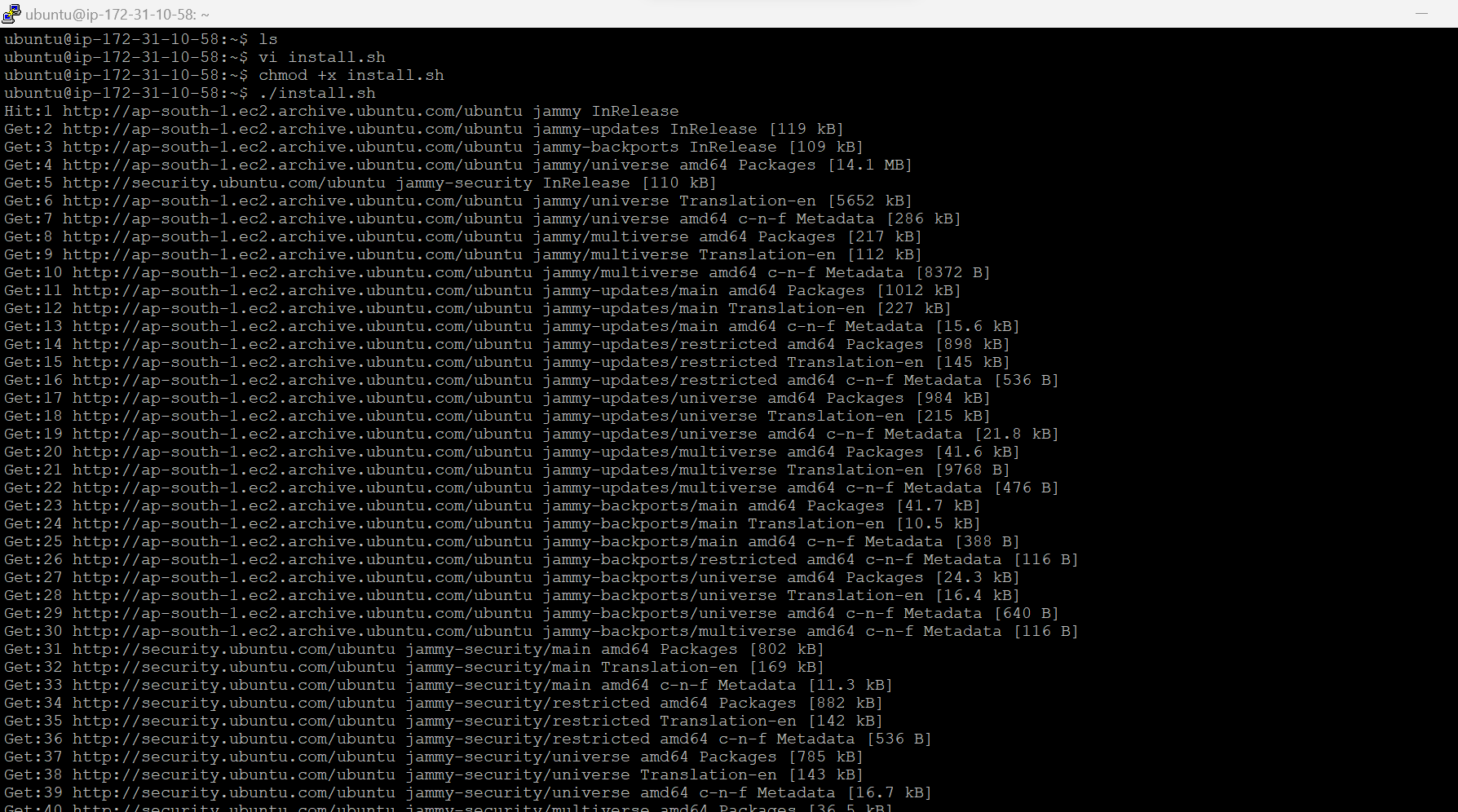
Install.sh

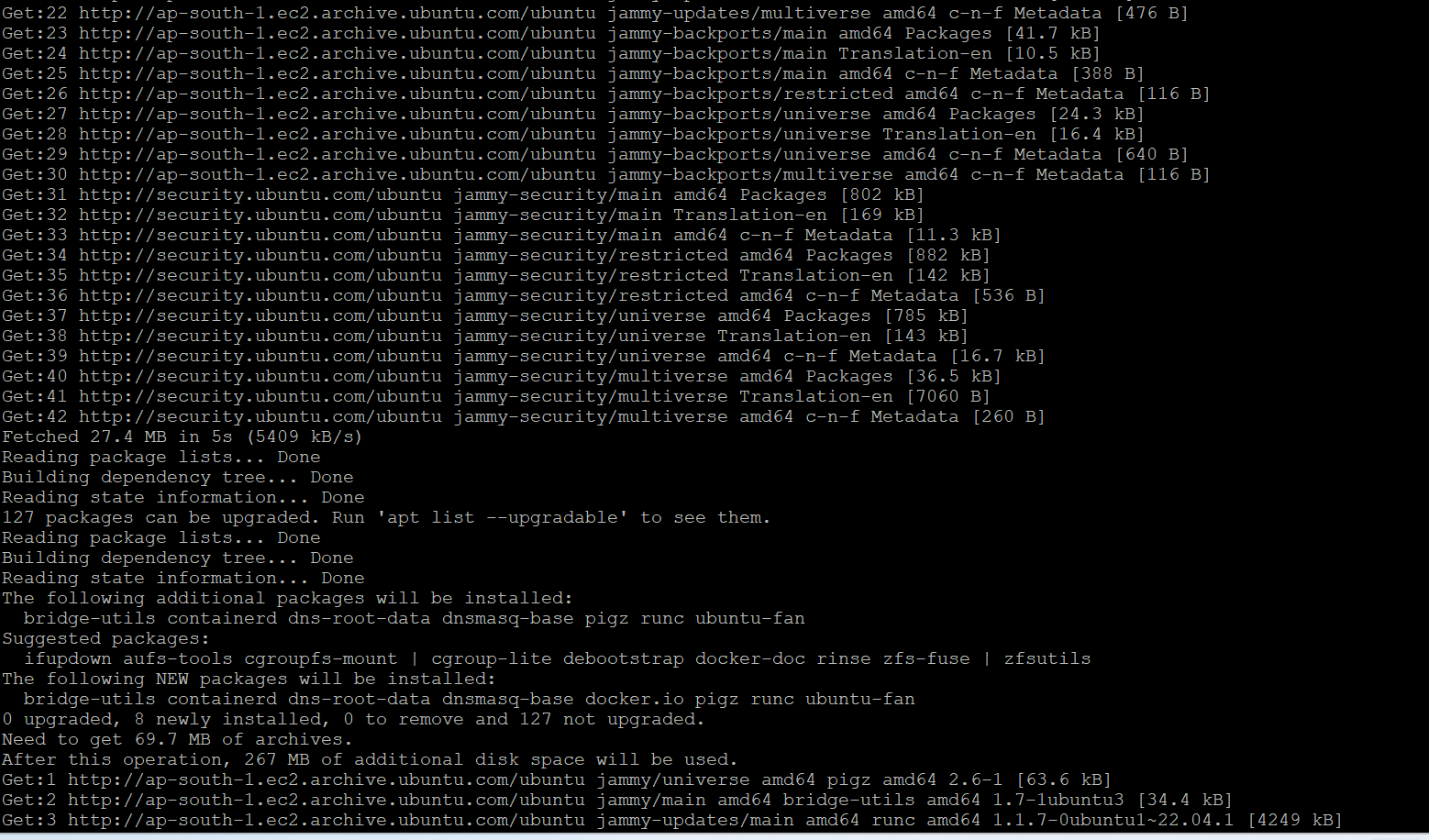


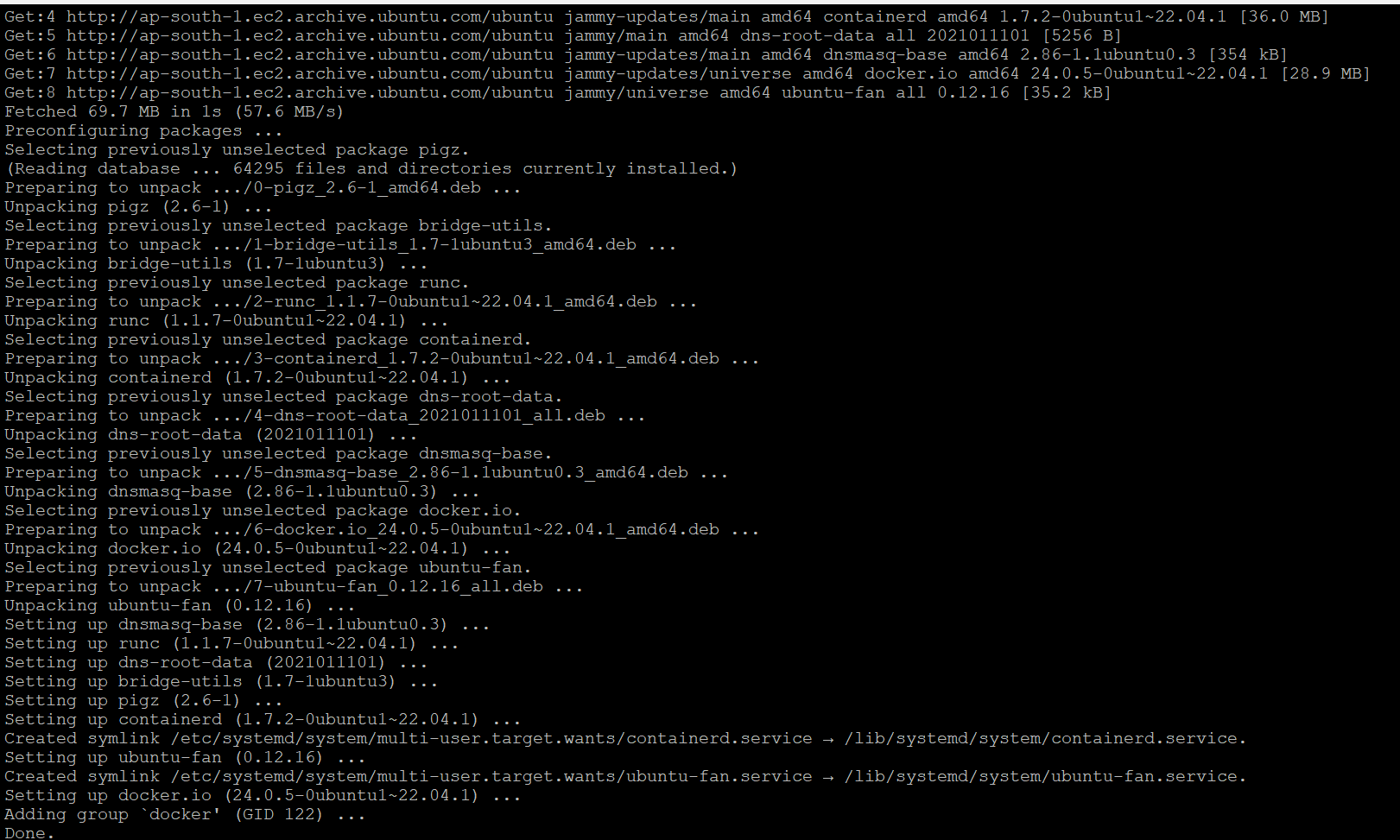
Pic.1

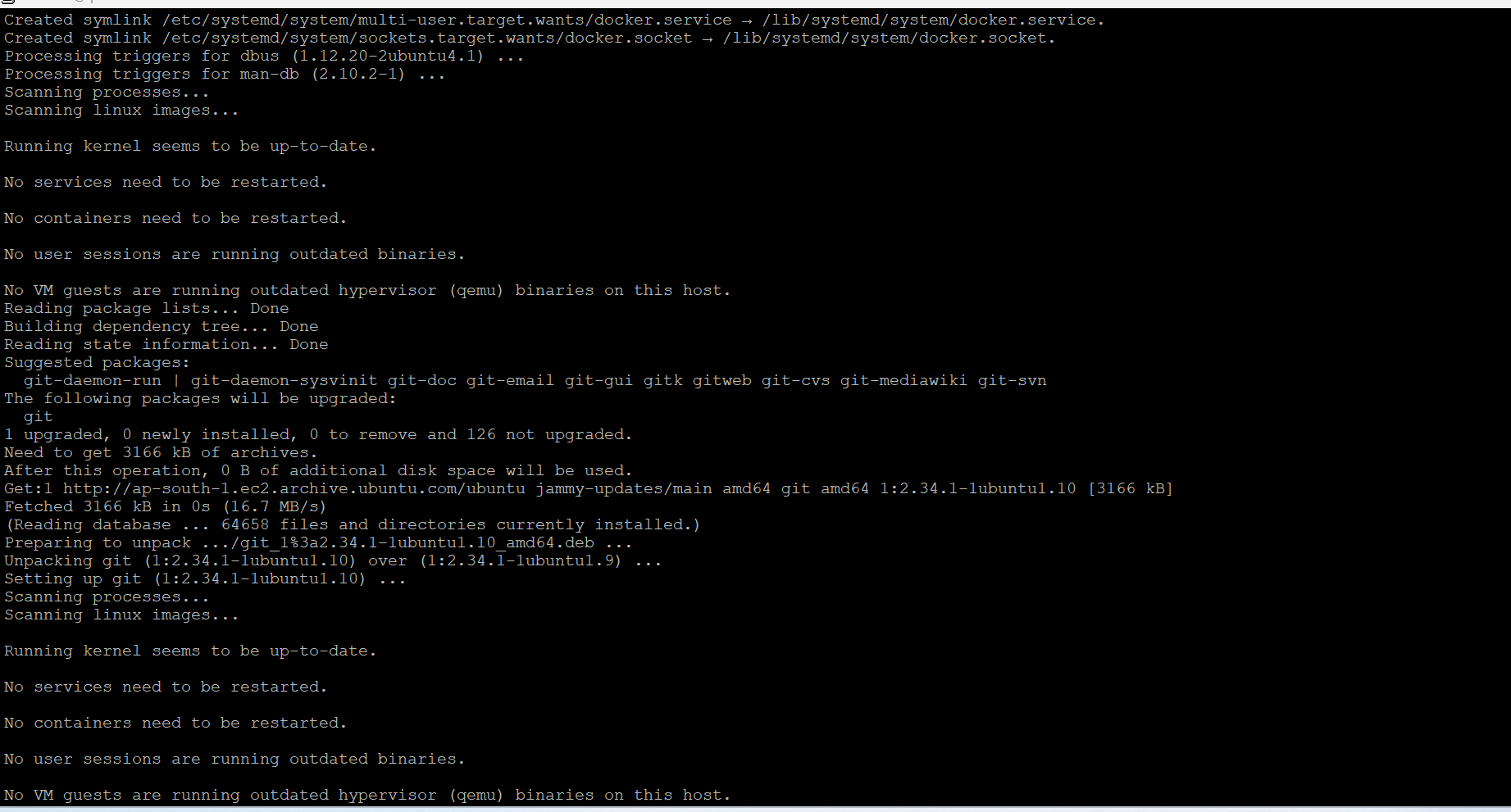


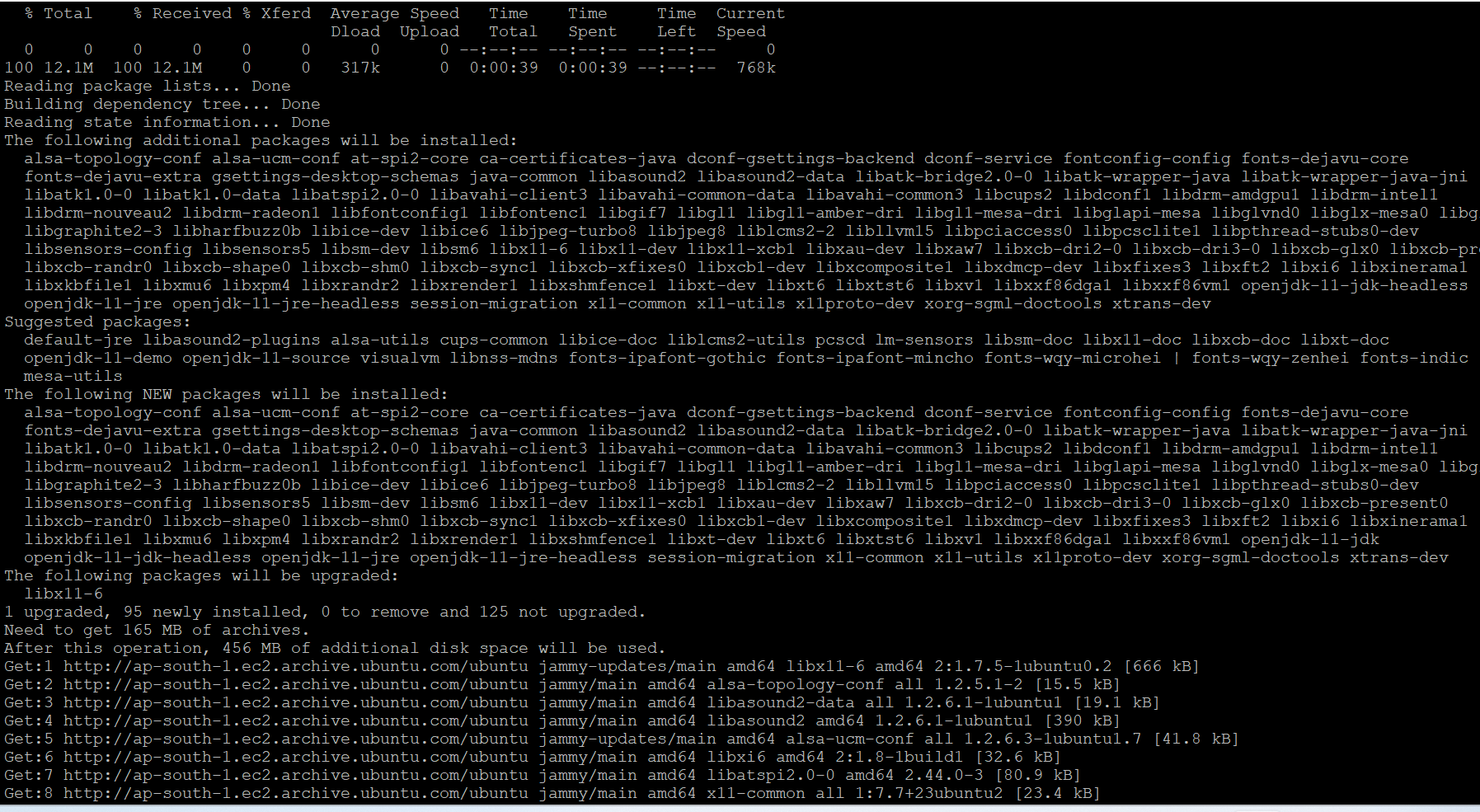
Pic.2

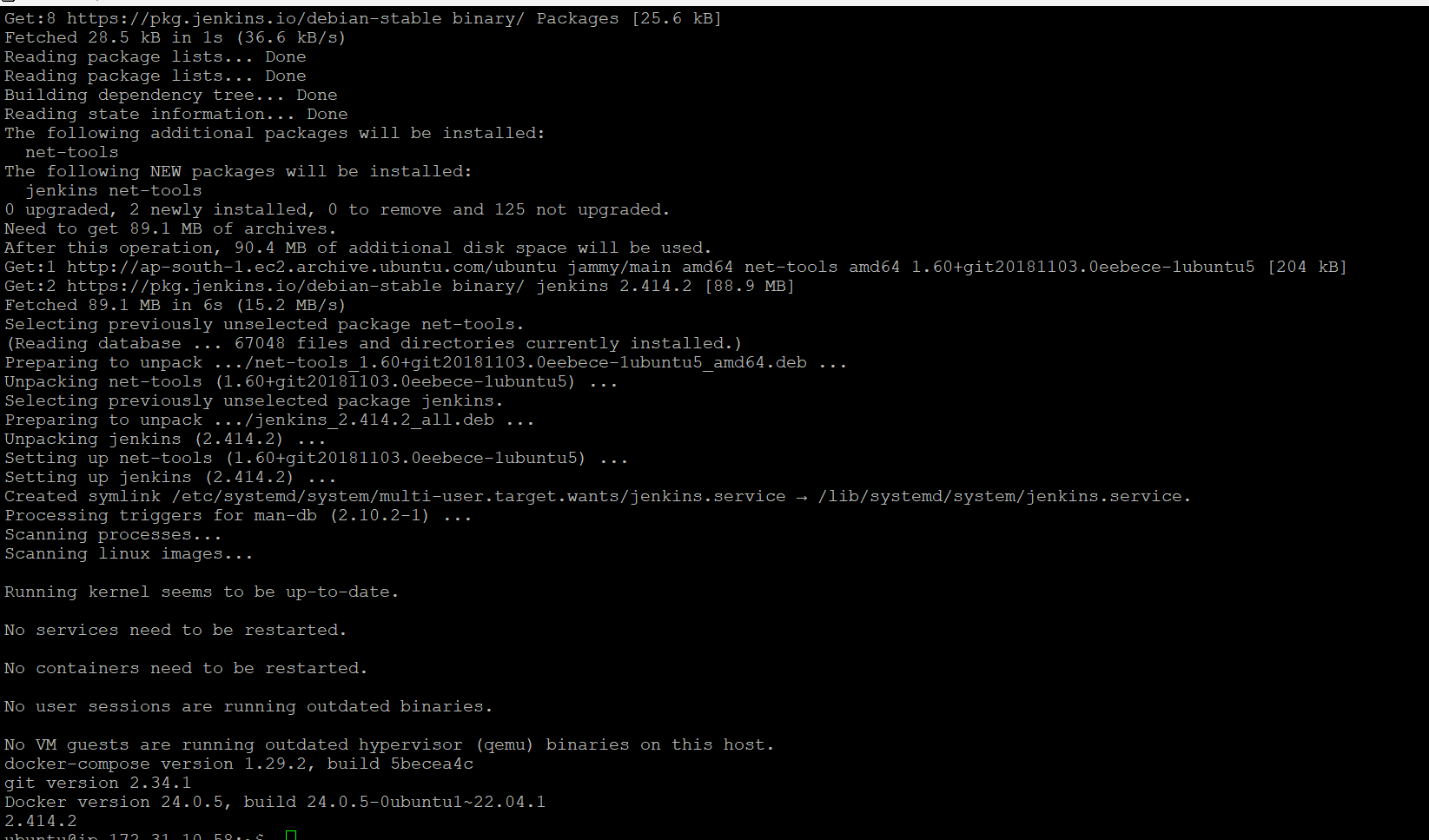












**Granted permission to Docker and Jenkins**

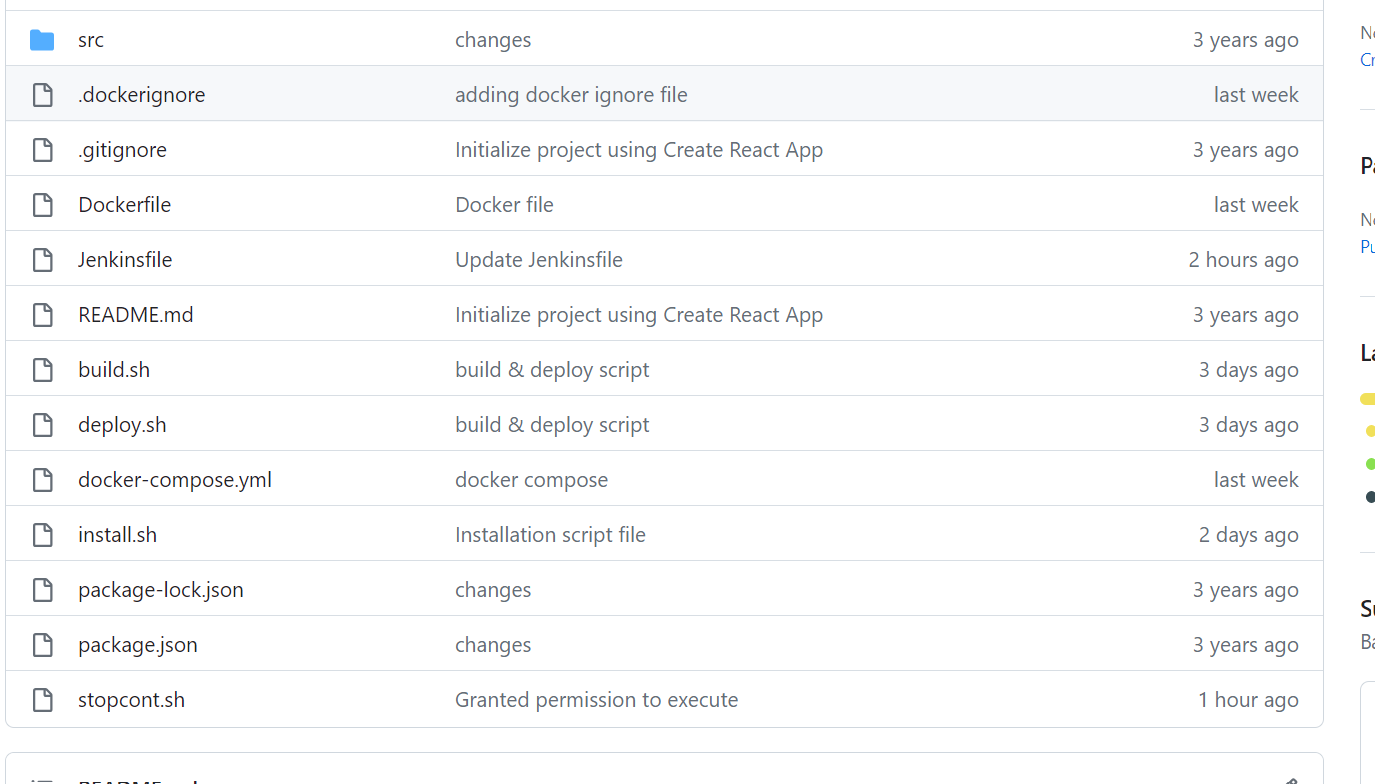
1. Sudo usermod -aG docker ubuntu
2. Sudo usermod -aG docker jenkins

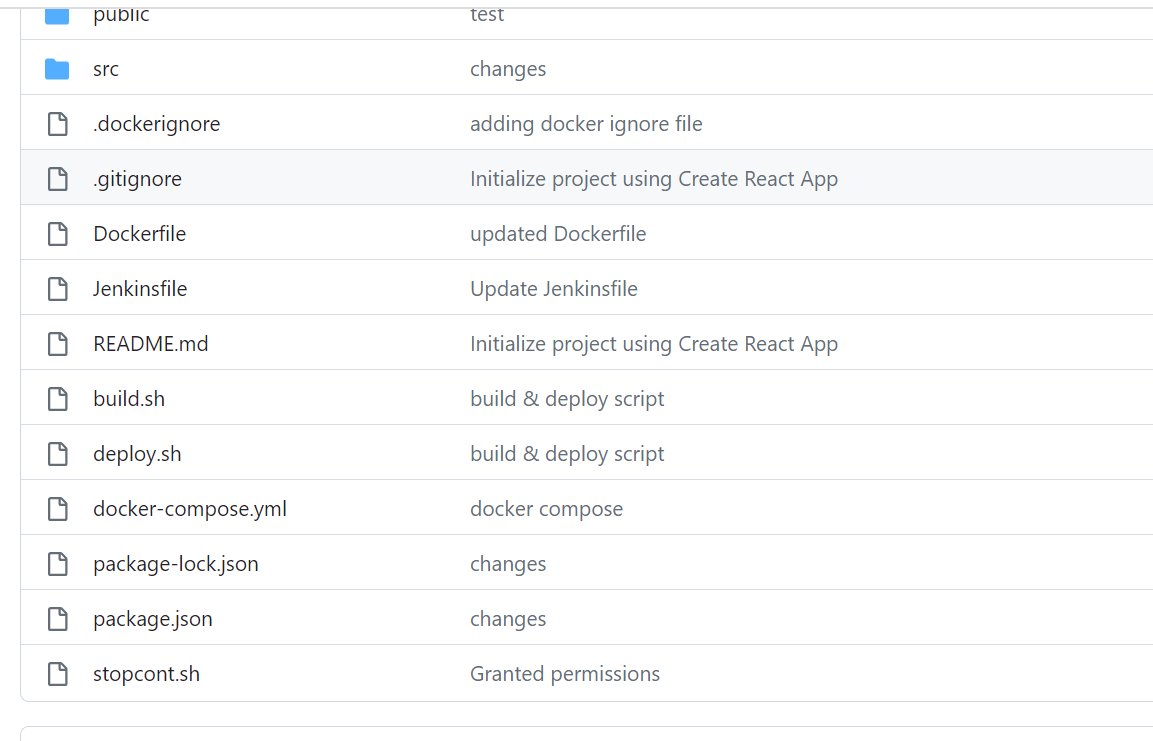
***Step 1:*** Created 2 branches in Git - > master & dev

***Step 2:*** Clone the capstone into my github repository

**Github url:** <https://github.com/KavithaVenugopal/reactjs-demo.git>

*\*/master branch*



*\*/dev branch*

***Docker file***

# Stage 1: Build the Node.js application

FROM node:14-alpine as build

WORKDIR /app

COPY package.json package-lock.json\* ./

RUN npm install

COPY . /app

RUN npm run build

# Stage 2: Serve the application using Nginx

FROM nginx:latest

# Copy the built files from the previous stage

COPY --from=build /app/build /usr/share/nginx/html

# Expose port 80 (the default HTTP port)

EXPOSE 80

# Start Nginx and keep it running in the foreground

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

***Build.sh***

docker system prune -f

docker build -t kavitha001/react .

***Deploy.sh***

docker-compose up -d

***Stopcont.sh***

#!/bin/bash

# Define the port you need for your application

PORT=80

# List all running containers

CONTAINERS=$(docker ps -q)

for CONTAINER in $CONTAINERS; do

# Check if the container is using the specified port

if docker port $CONTAINER | grep -qE "[0-9]+/tcp.\*:$PORT"; then

# Stop the container

echo "Stopping container $CONTAINER to free up port $PORT"

docker stop $CONTAINER

fi

done

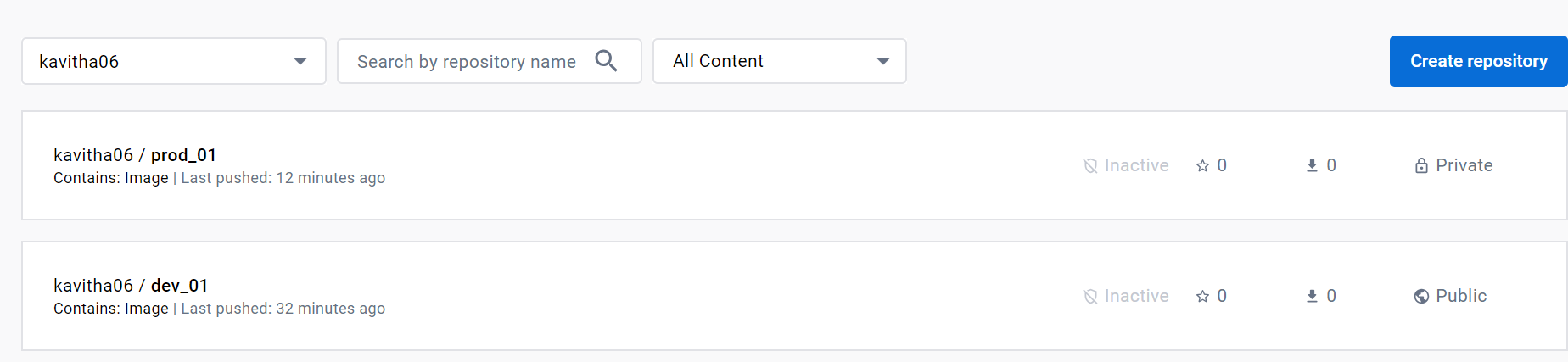
# Run your deployment command here (e.g., docker-compose up)

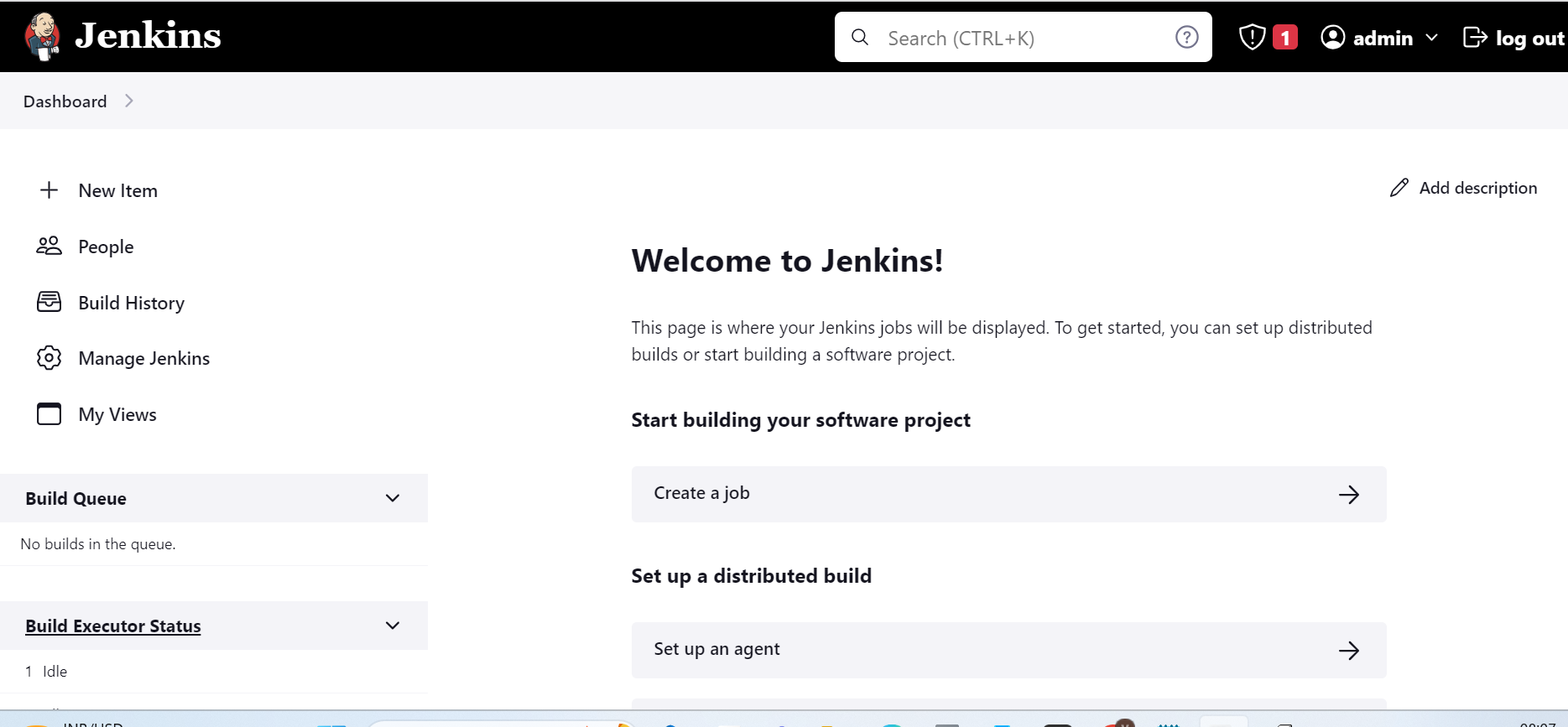
./deploy.sh

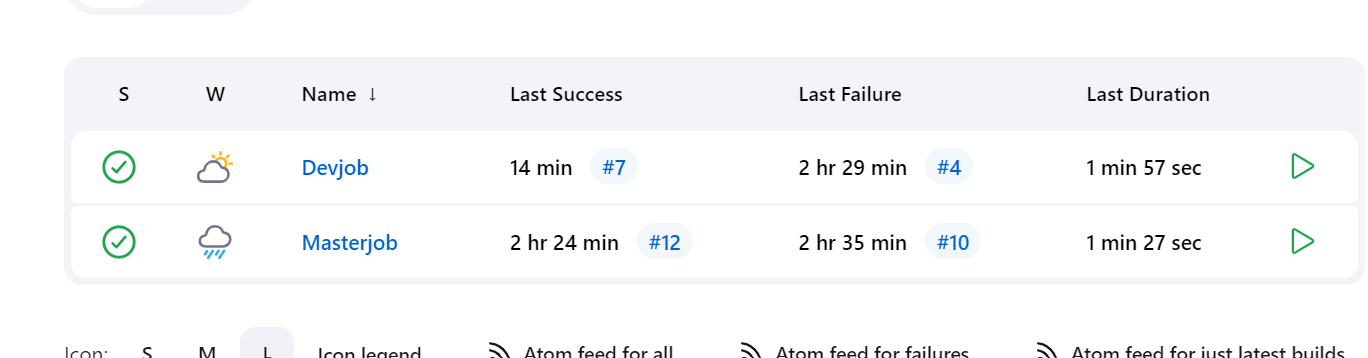
**Docker Hub URL**

<https://hub.docker.com/u/kavitha06>

***2 repo’s***

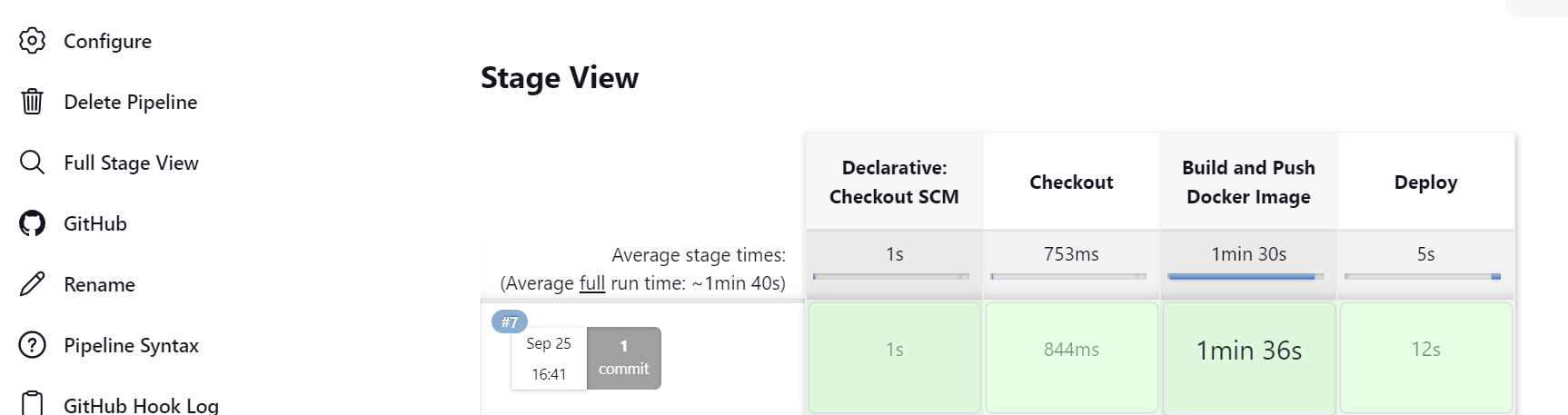


**Jenkins**

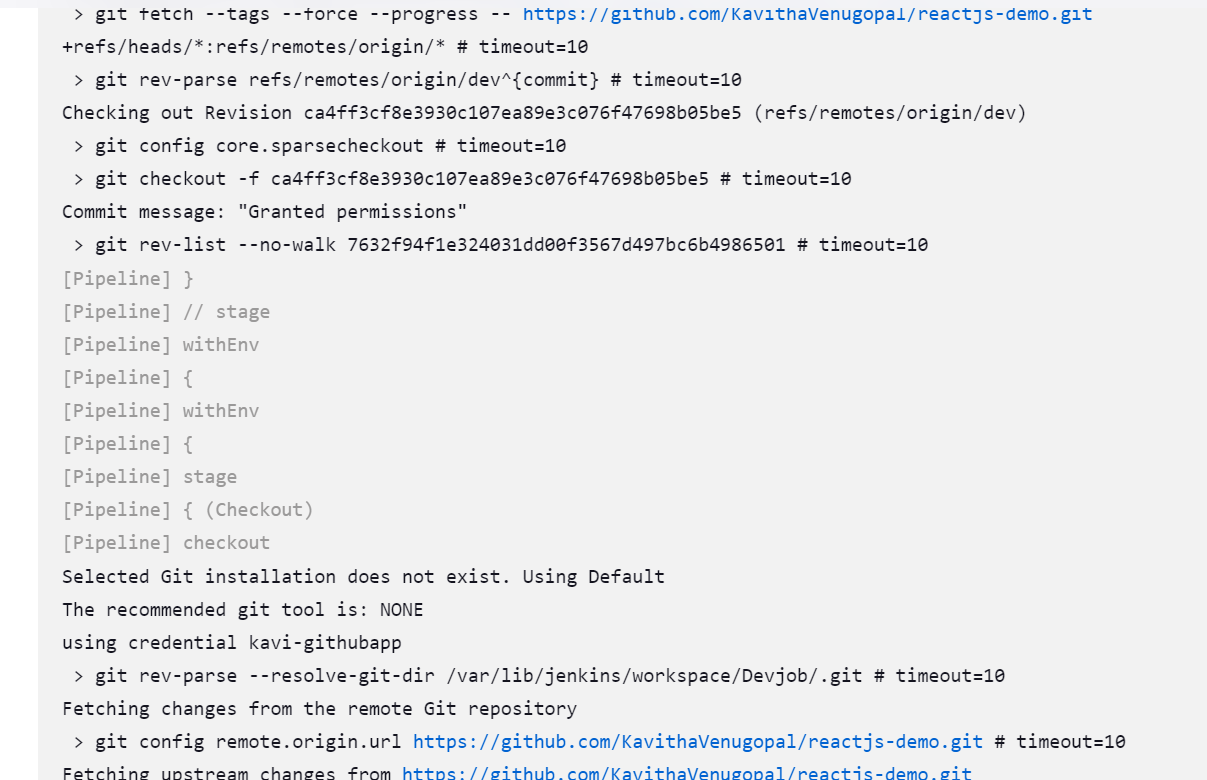


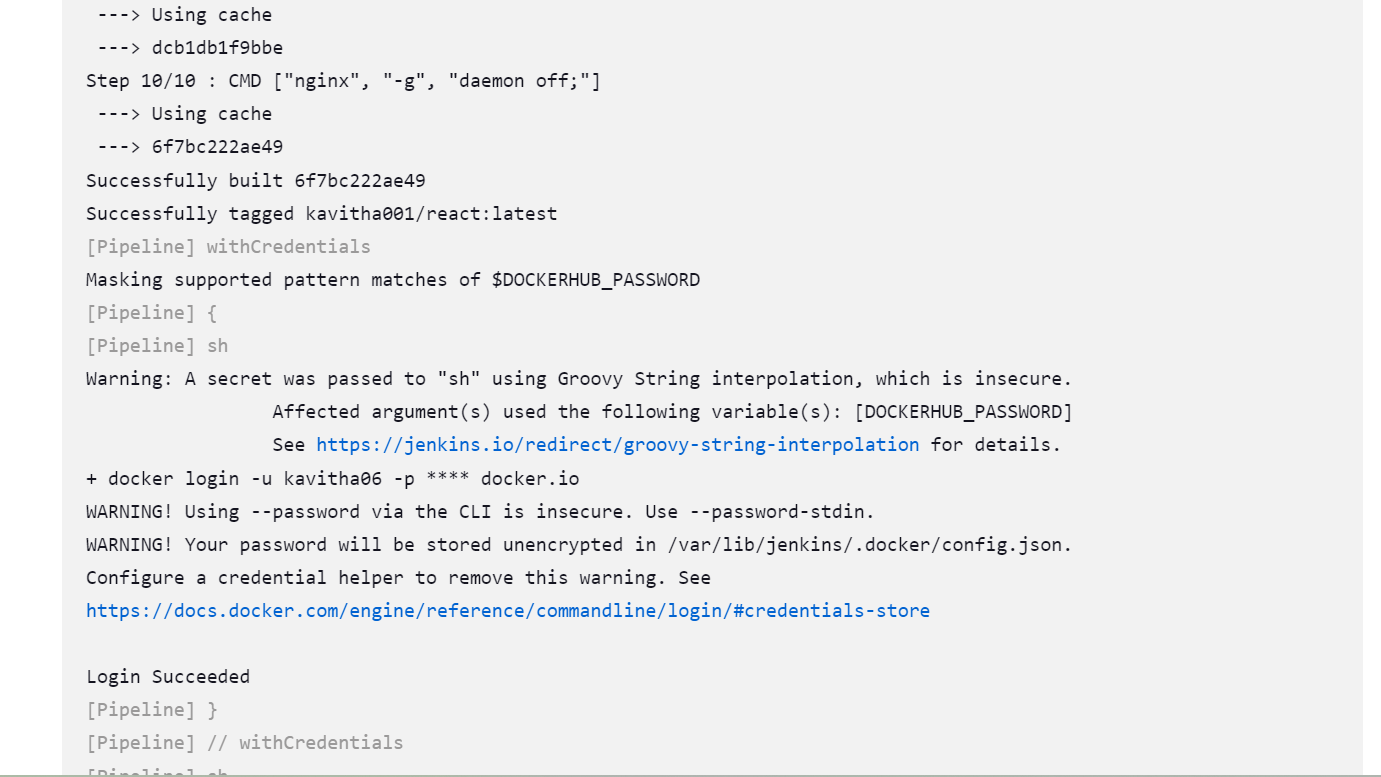
1. *If a code is pushed to the Dev branch, an image should be built and pushed to the docker hub public repository*

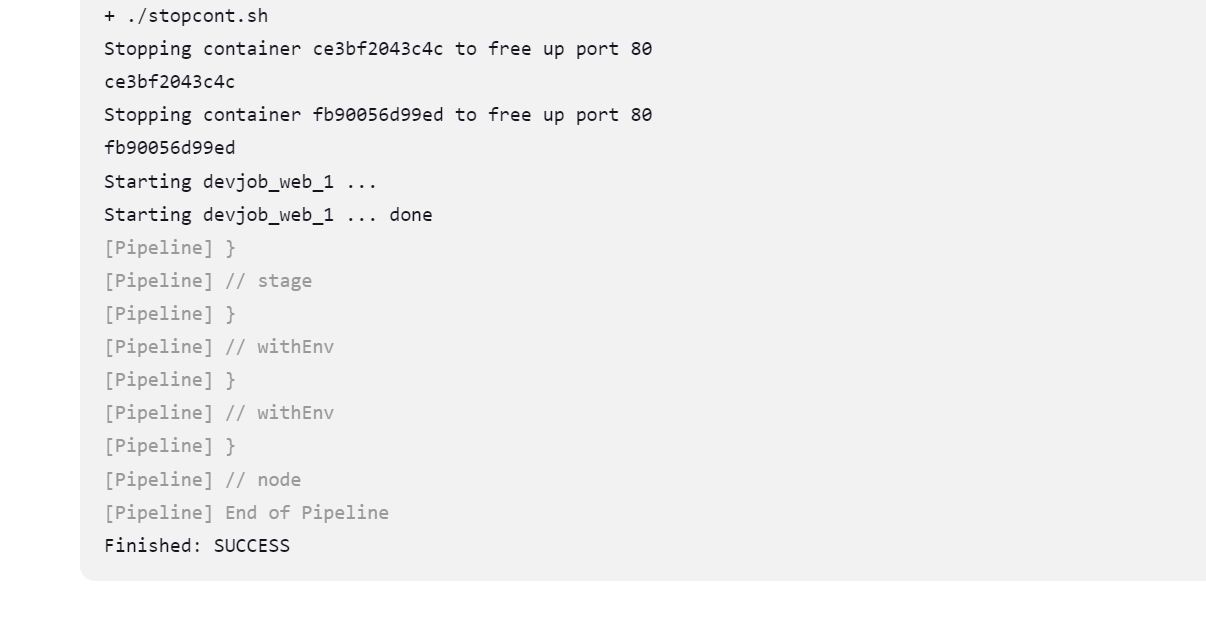
**Pipeline successful**

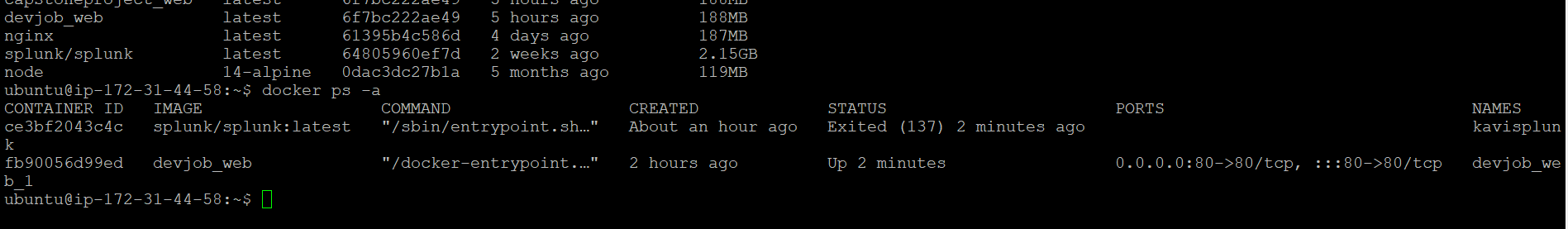




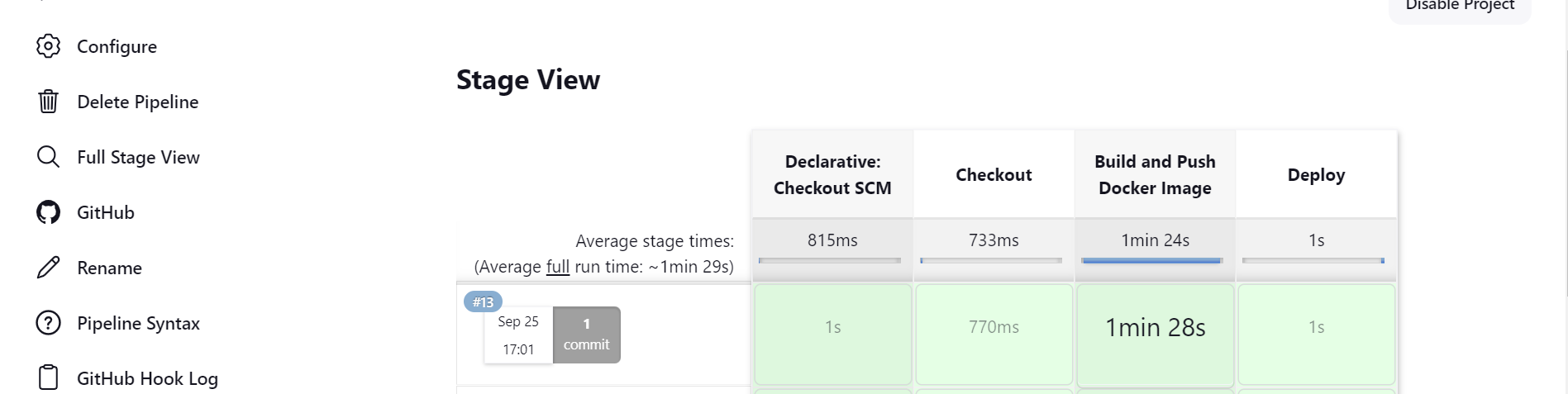


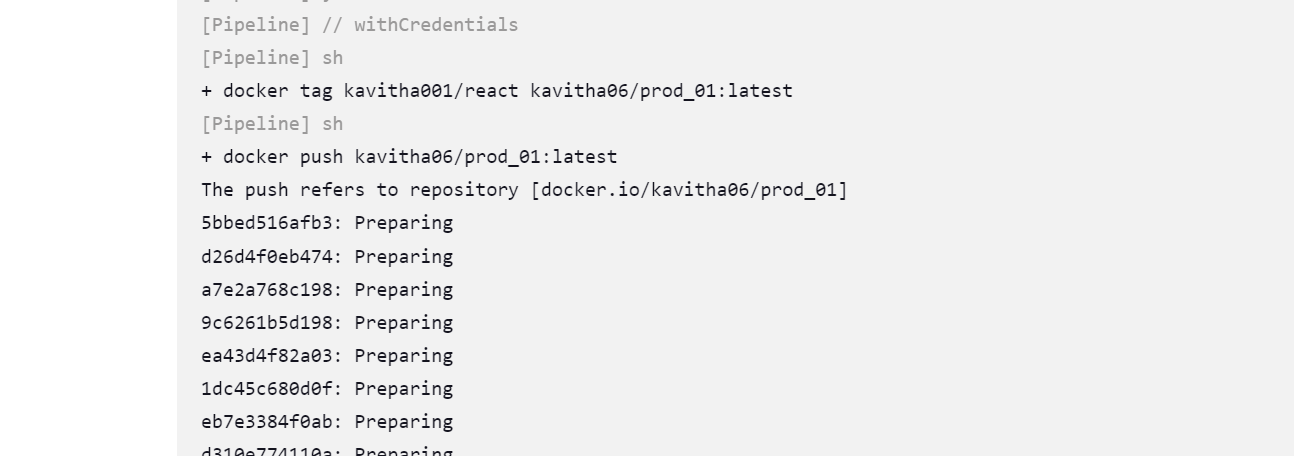


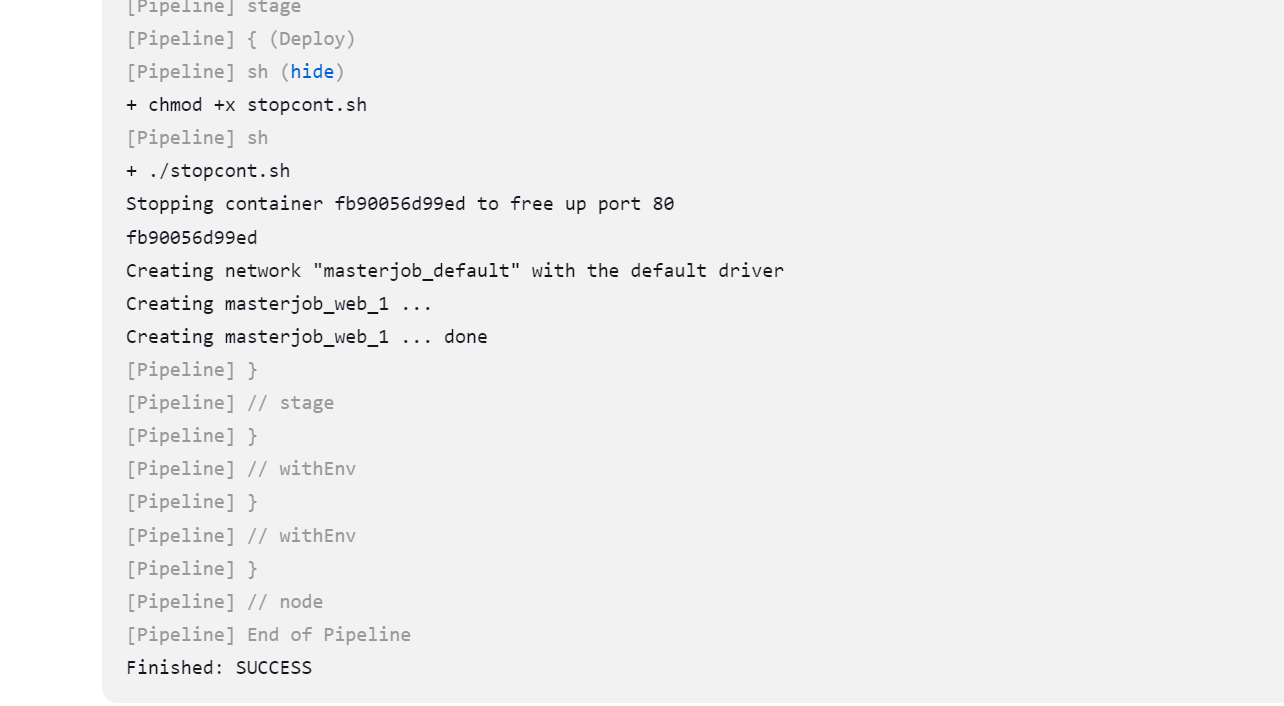


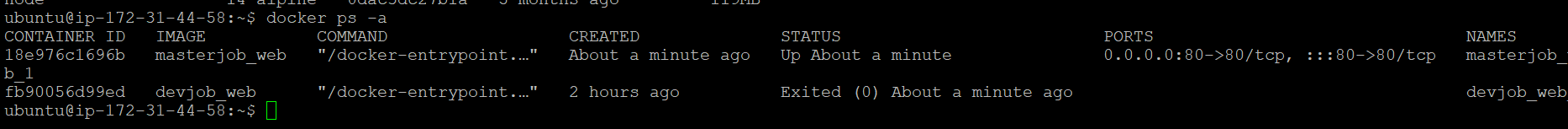


1. *If a code is merged with master branch then the docker image should be built and pushed to the docker hub private repository*

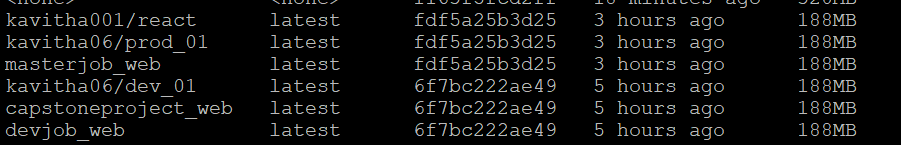




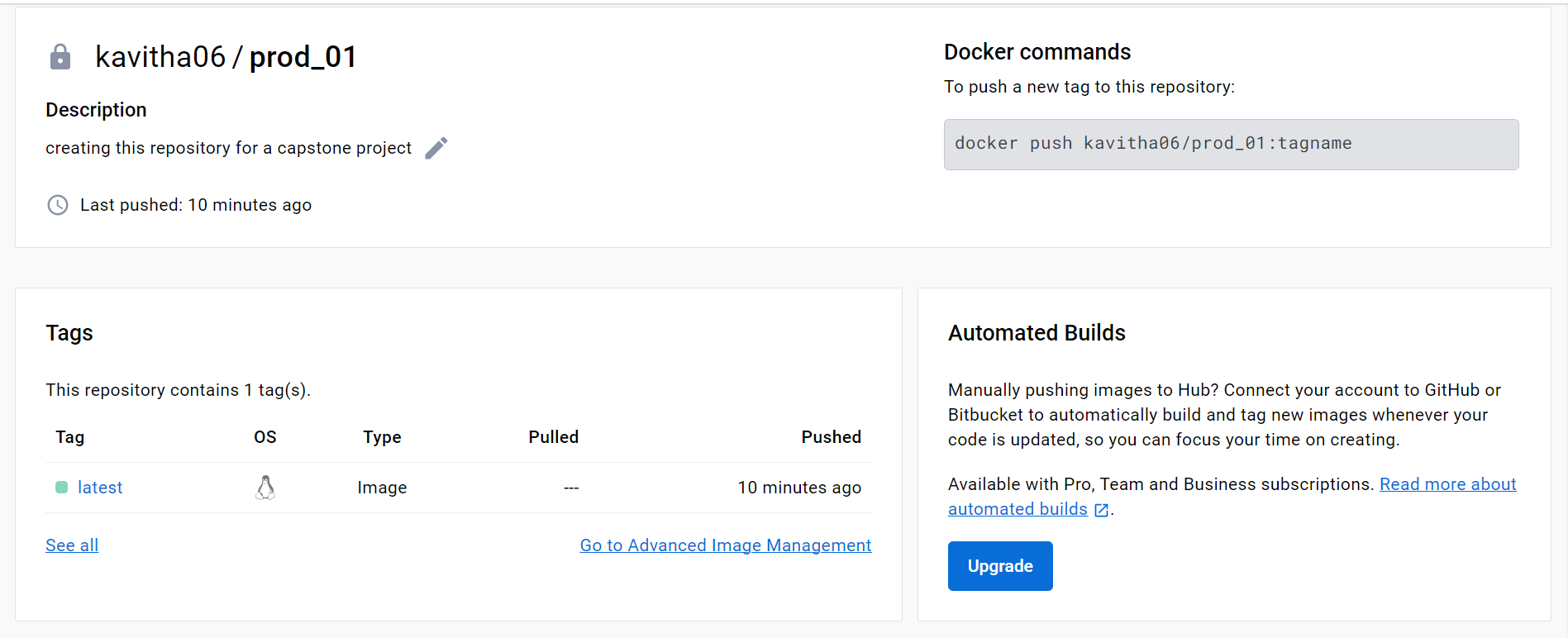




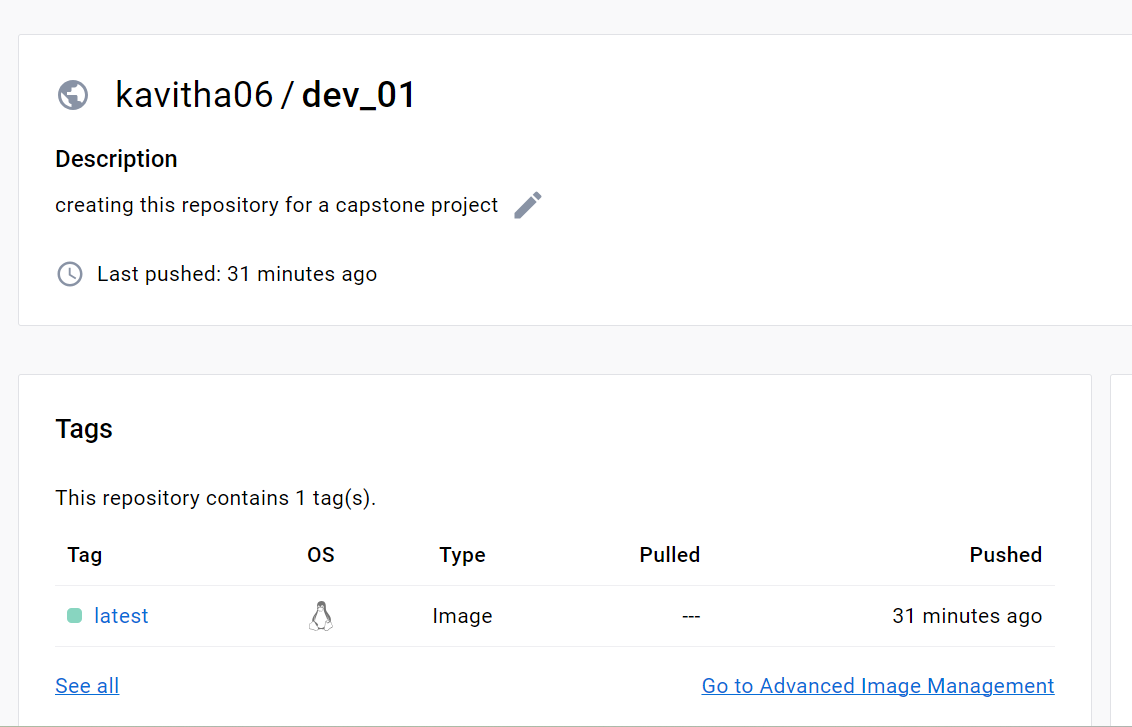
***Docker Images***

******

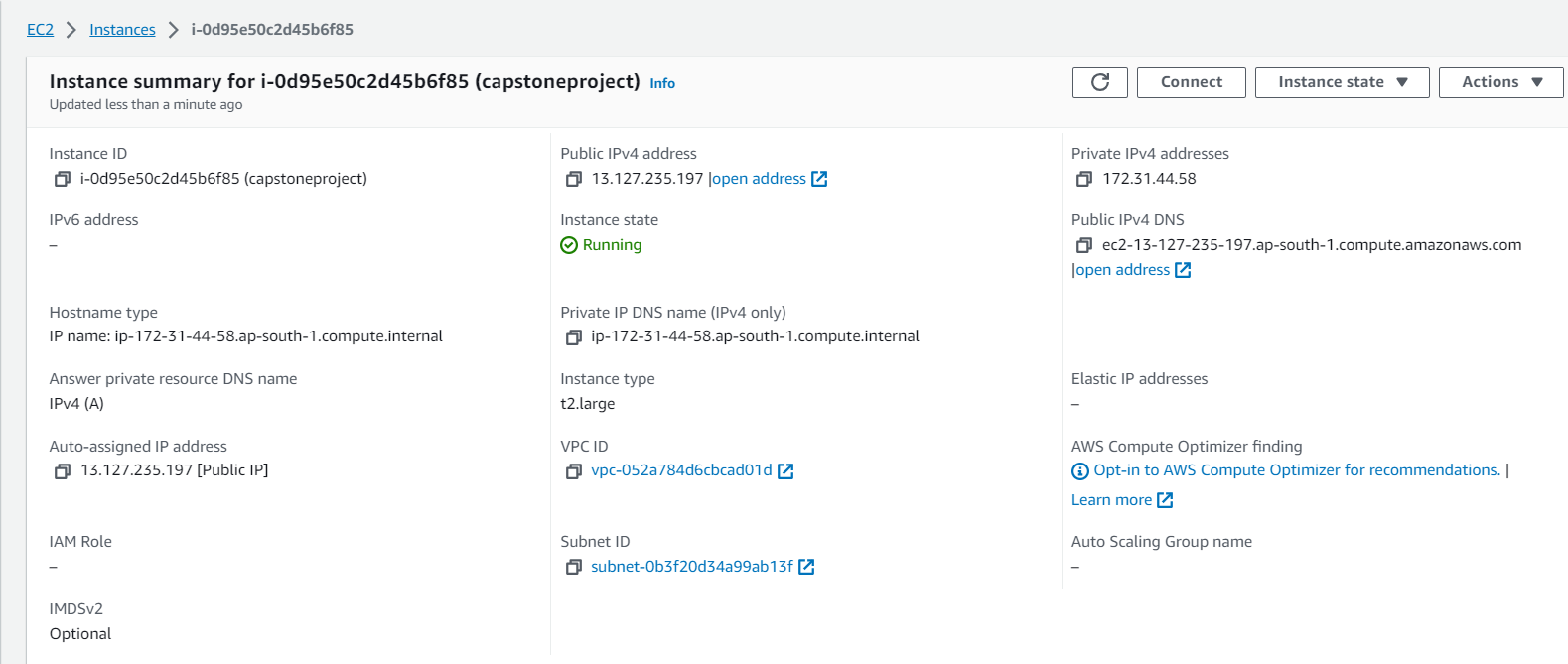
***Docker Hub repository - private***

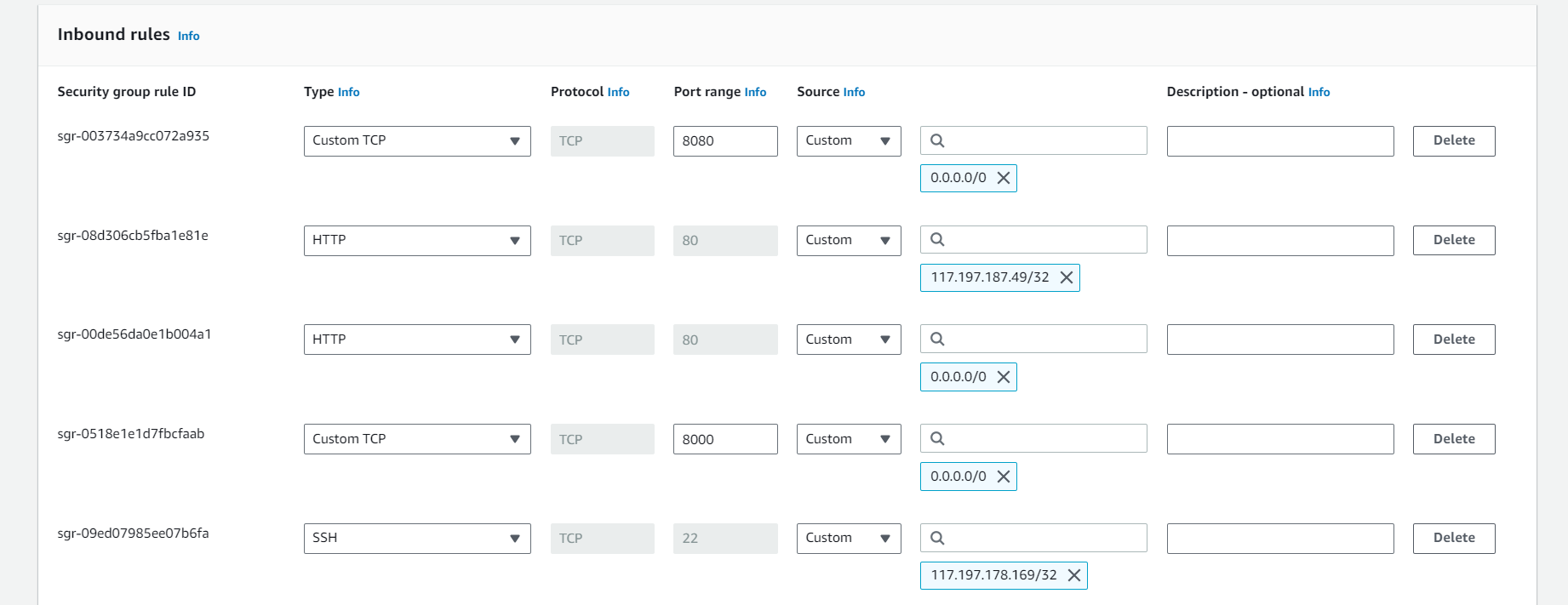


**Docker Hub - Public Repository**

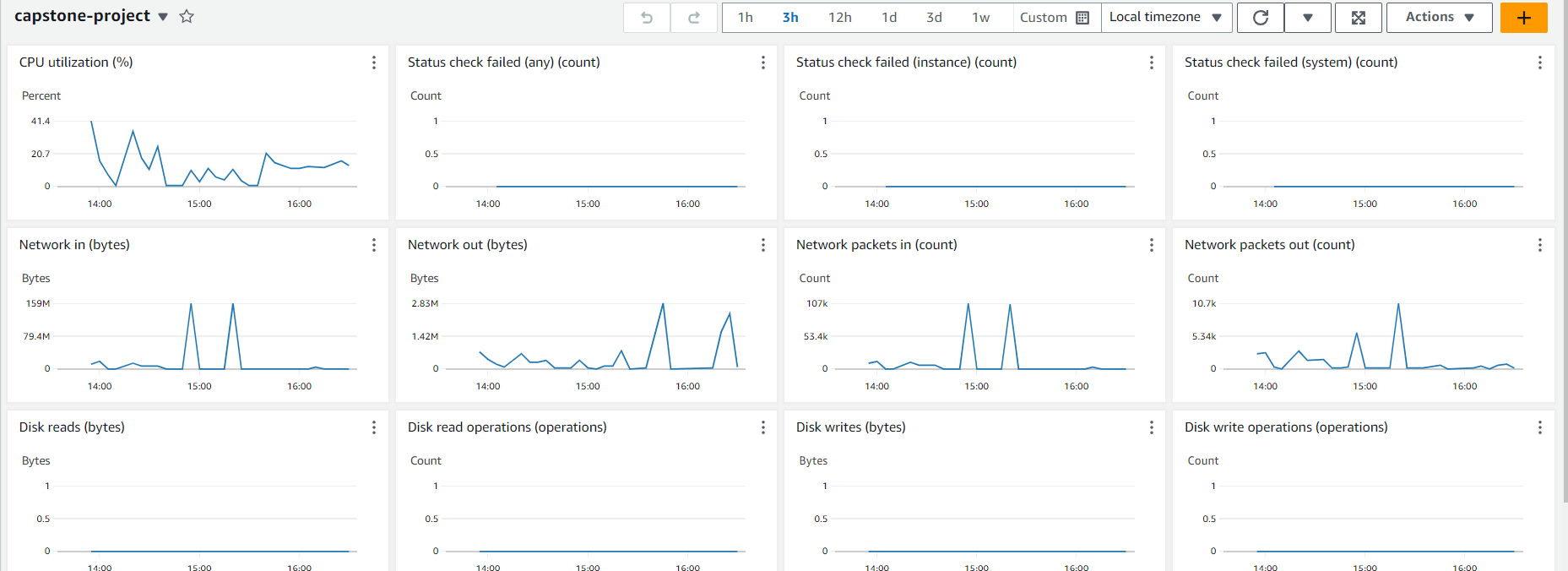
****

**AWS**

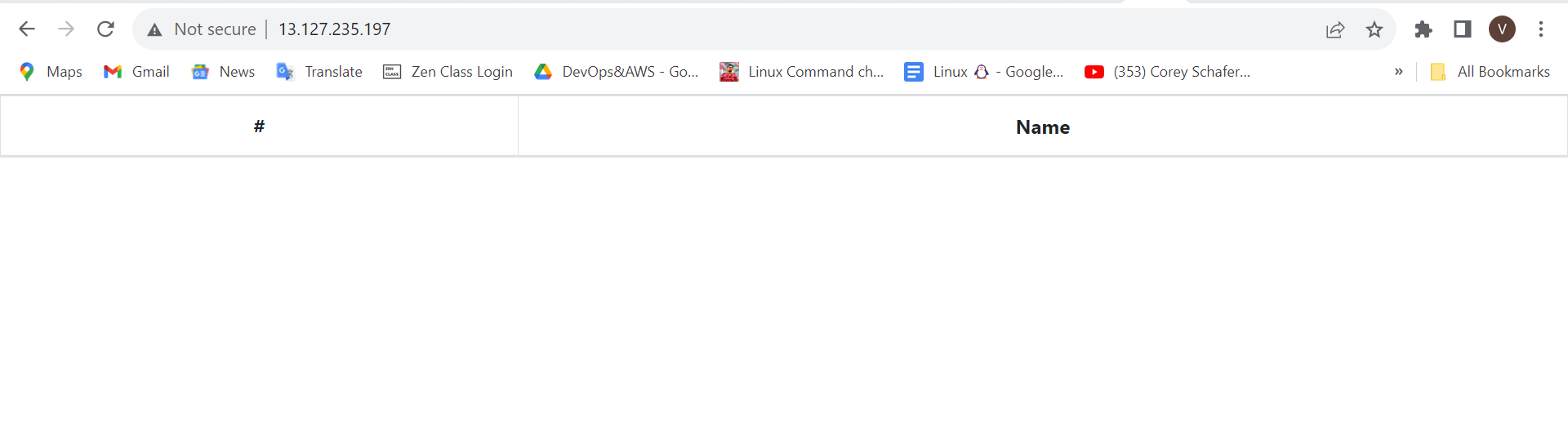
****

****

Monitoring using Cloudwatch



***Deployed application***



*Deployed application site :* [*http://13.127.235.197/*](http://13.127.235.197/)

*Github url:* [*https://github.com/KavithaVenugopal/reactjs-demo.git*](https://github.com/KavithaVenugopal/reactjs-demo.git)

*Dockerhub url:* [*https://hub.docker.com/u/kavitha06*](https://hub.docker.com/u/kavitha06)