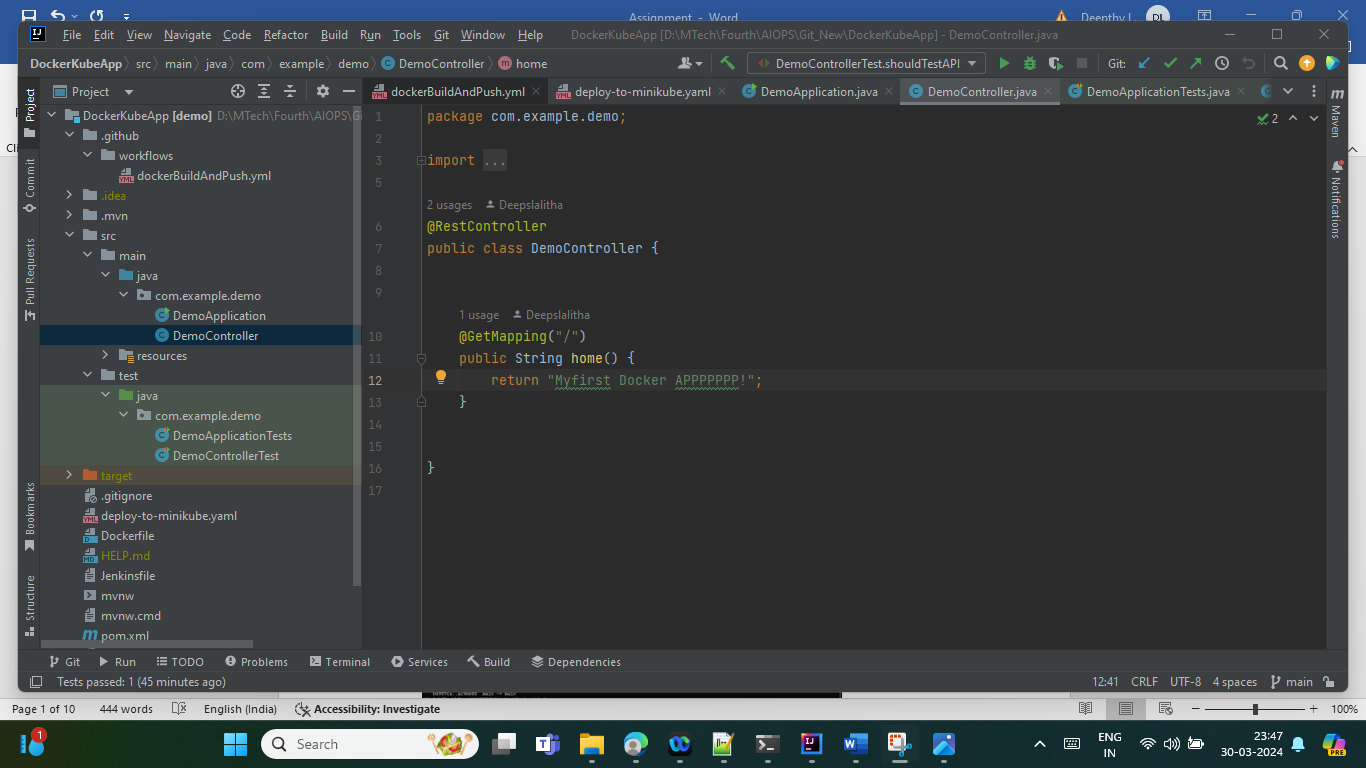
Assignment:

As part of Implementing CI/CD pipeline , the below steps were carried out .

1. Build a Spring boot application (version 2.7.18) that points to an API on load of application which displays a static text (JDK-11)

**Tools Used- Intellij**

Rest API:

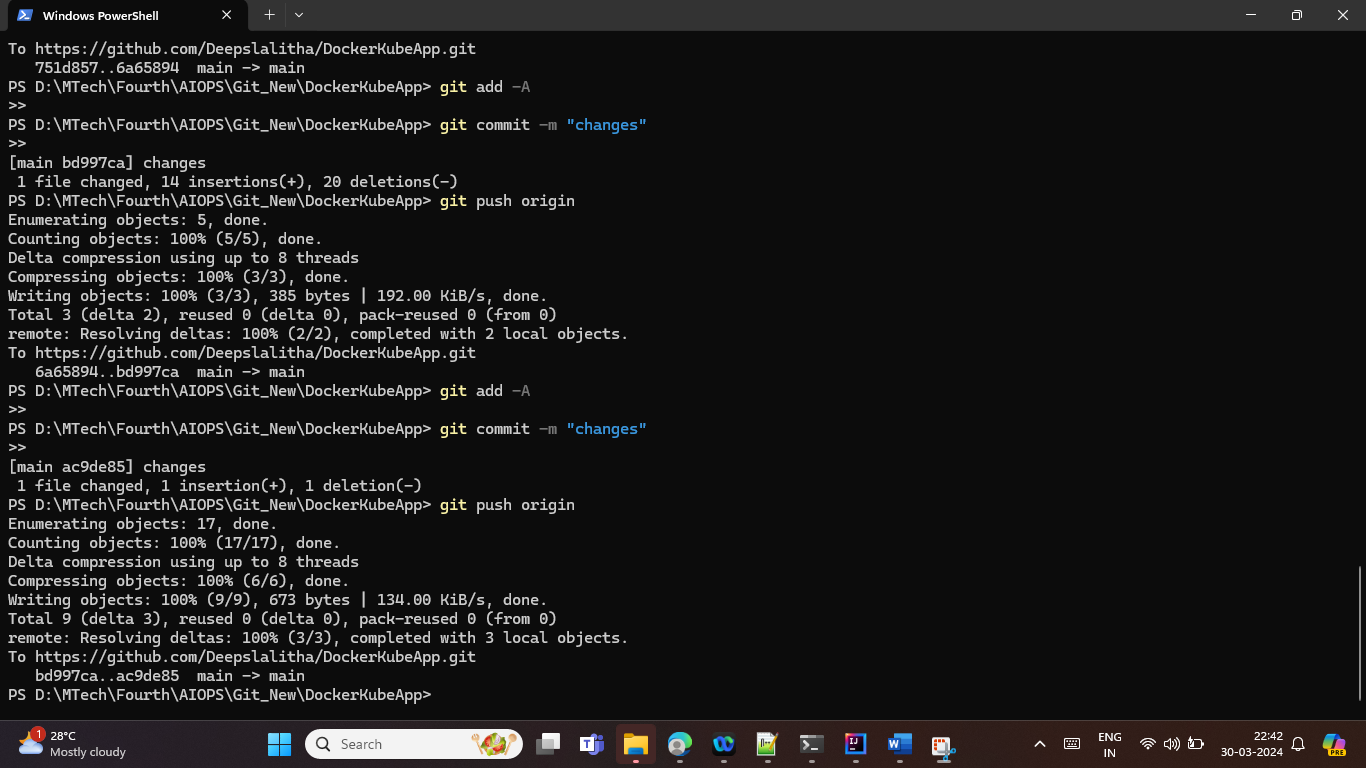


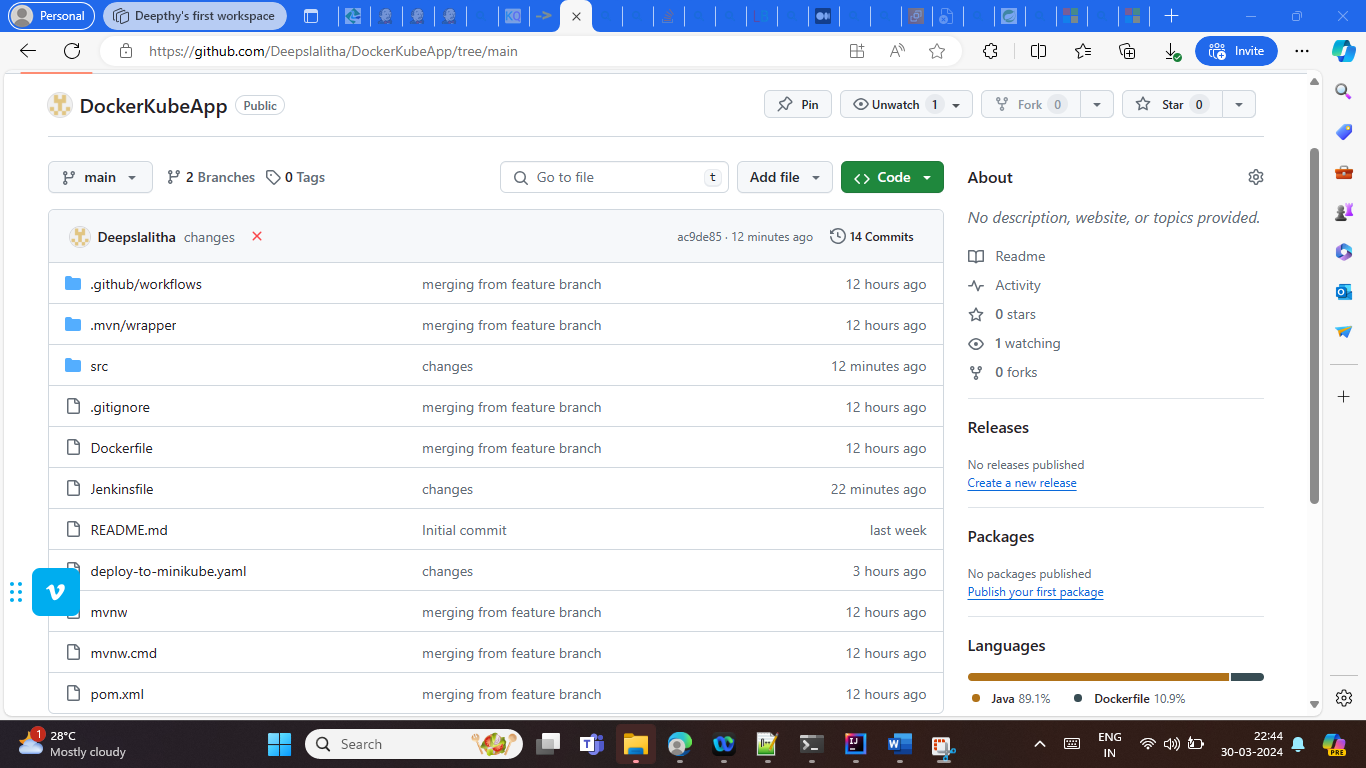
1. Version Control

**Tools Used – GIT**

Git was installed on local system. Repo created for the assignment is Deepslalitha/DockerKubeApp

Feature branch was created to try out the changes . Main is the branch that is the stable version of code.





1. CI/CD Pipeline:

CI/CD pipeline was designed that included the below stages

1. Build

Maven is used to build the application . This stage will build the code

1. Testing

This stage includes running the Junit and the Spring Boot Integration tests

Mvn test is used .

1. Checkout the Repo and create a Docker Image
2. Push the Docker Image to Docker Hub (deepthylalithatech/mydemoapp:latest)
3. Deploy to Mini Kube Cluster
4. Test the API

1. Tools Used:

**Jenkins** was used to set up the pipeline

Initially I tried using git-hub workflows . There ate git hub actions available to set up a mini kube cluster. But it was seen that even if the deployment was successful to cluster , there was a delay in connecting to the service url .

Jenkins image was deployed to local and mini kube was set up. But somehow system was having performance issues and became very slow. So I have used AWS ec2 cluster as the Jenkins server.

1. Installed Jenkins , docker and mini kube in ec2 cluster
2. Gave necessary permissions for Jenkins to access docker and minikube.

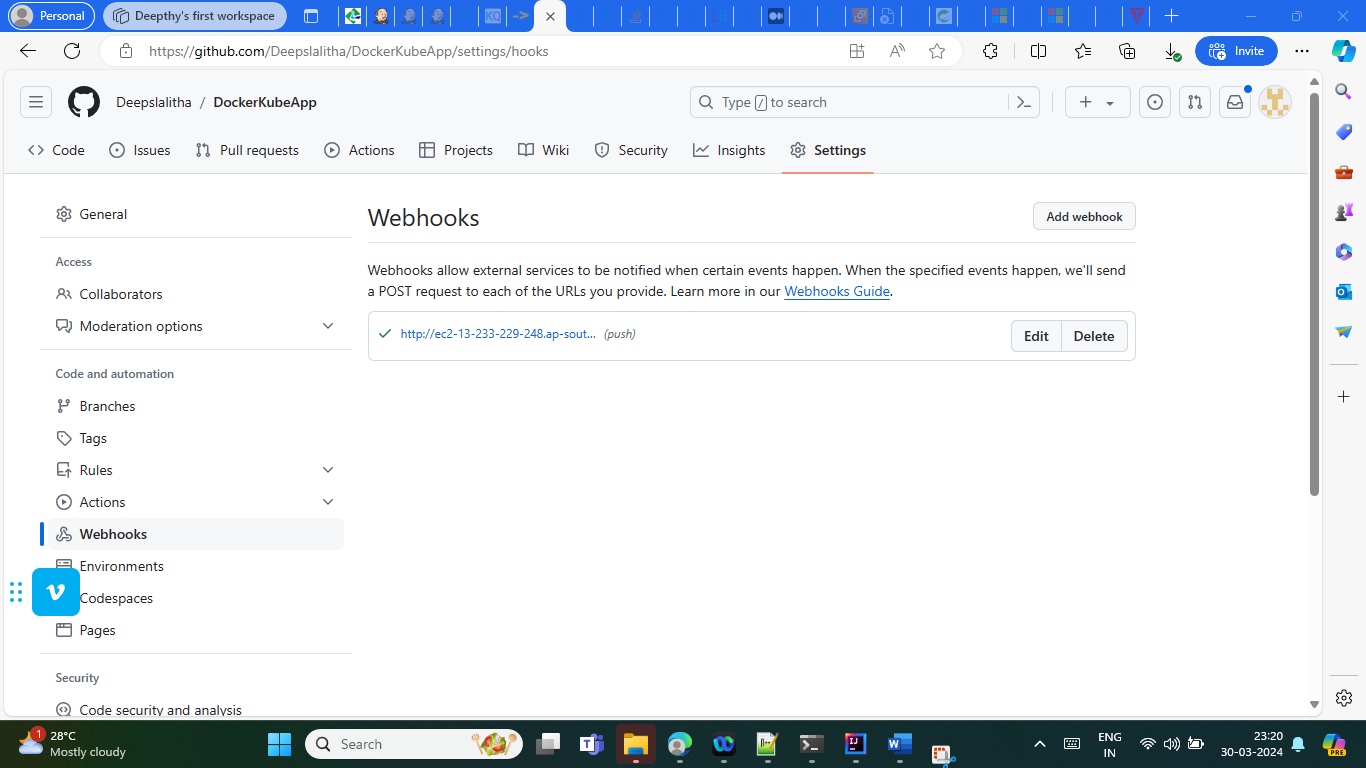
<http://ec2-13-233-229-248.ap-south-1.compute.amazonaws.com:8080/> - Jenkins URL

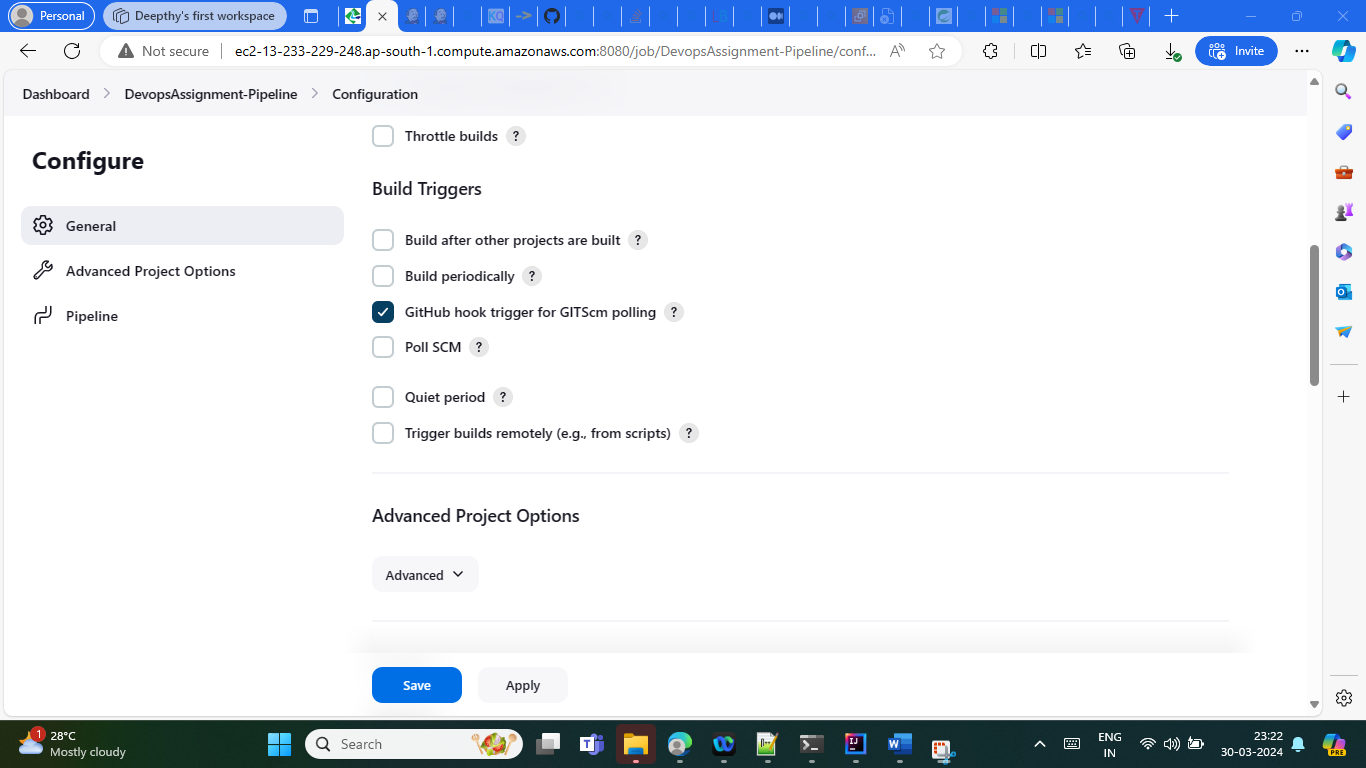
Git hub hook was configured so that any commits to the DockerkubeApp repo triggered a pipeline . (Config in GitHub & Jenkins)

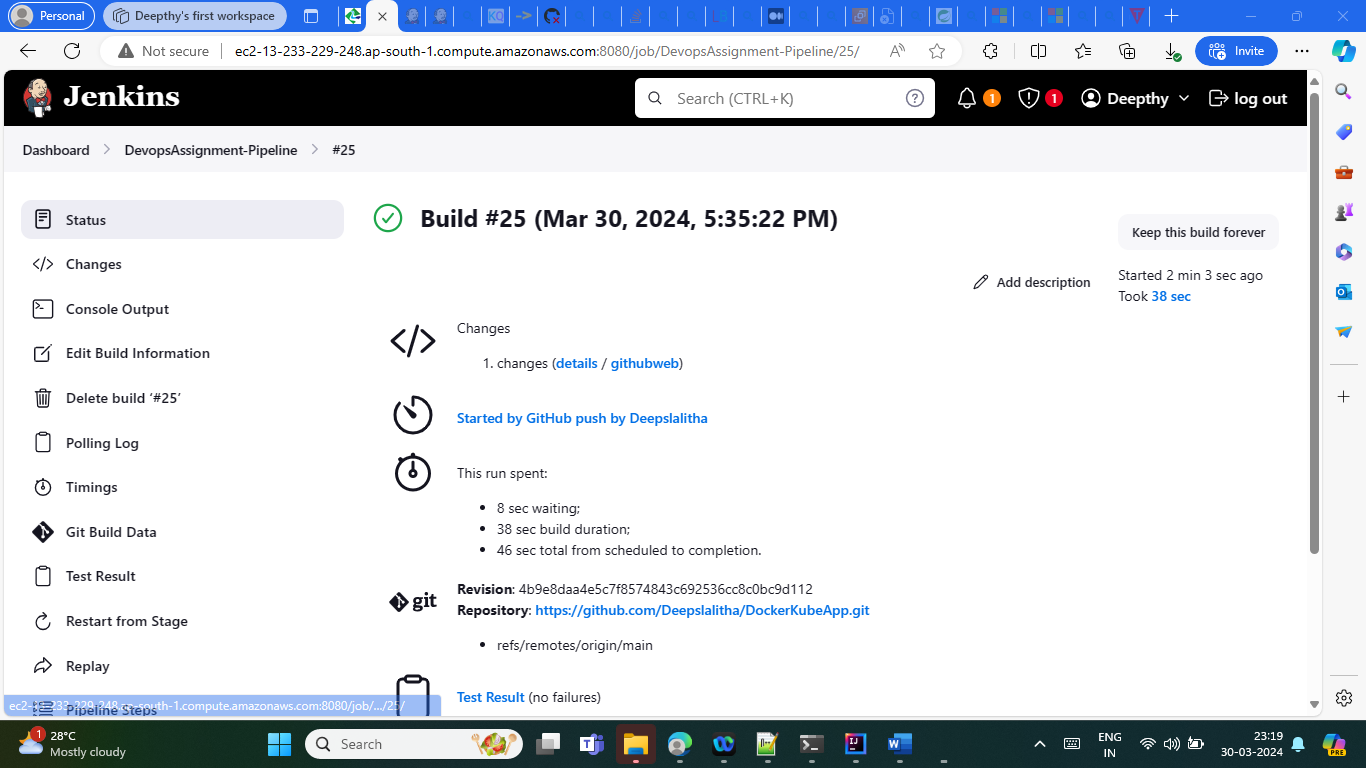
Jenkins file path : [DockerKubeApp/Jenkinsfile at main · Deepslalitha/DockerKubeApp (github.com)](https://github.com/Deepslalitha/DockerKubeApp/blob/main/Jenkinsfile)

Screenshots:

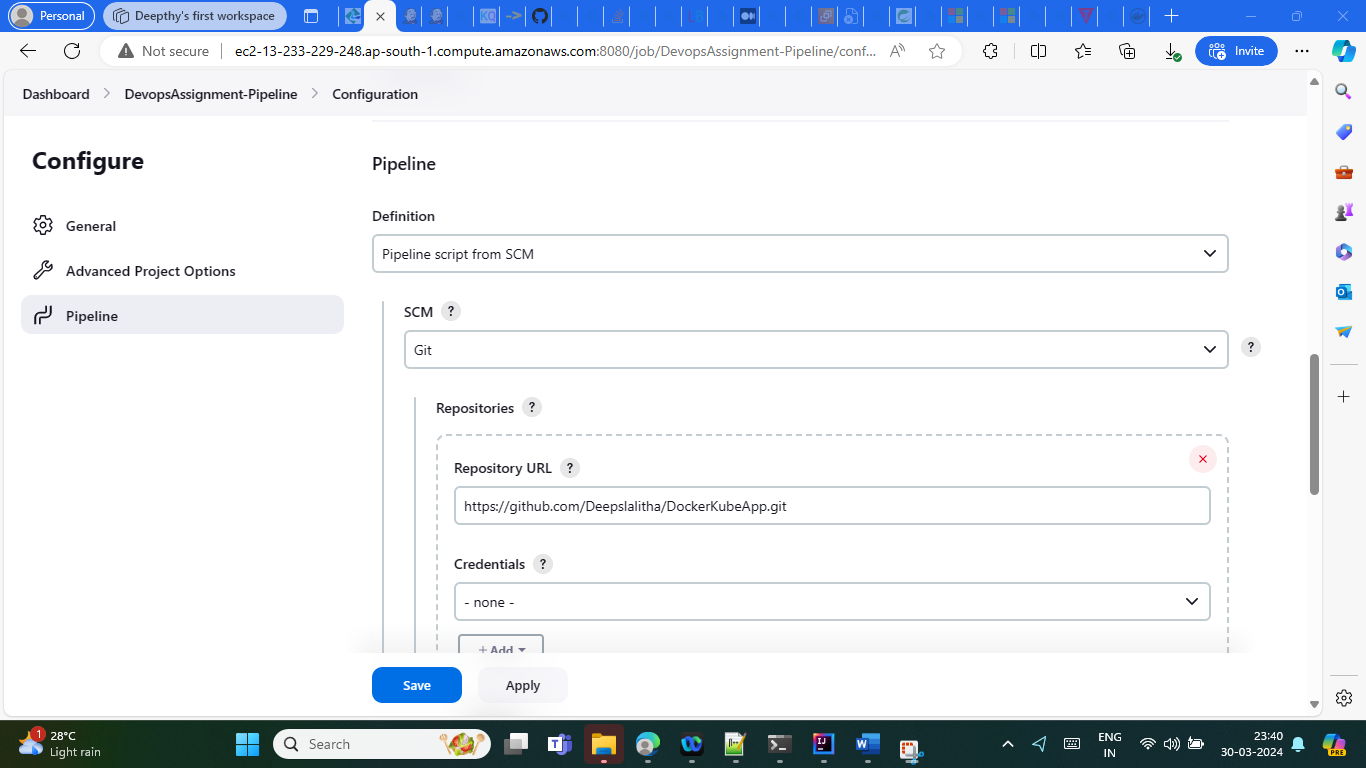
1. Git hub hook config

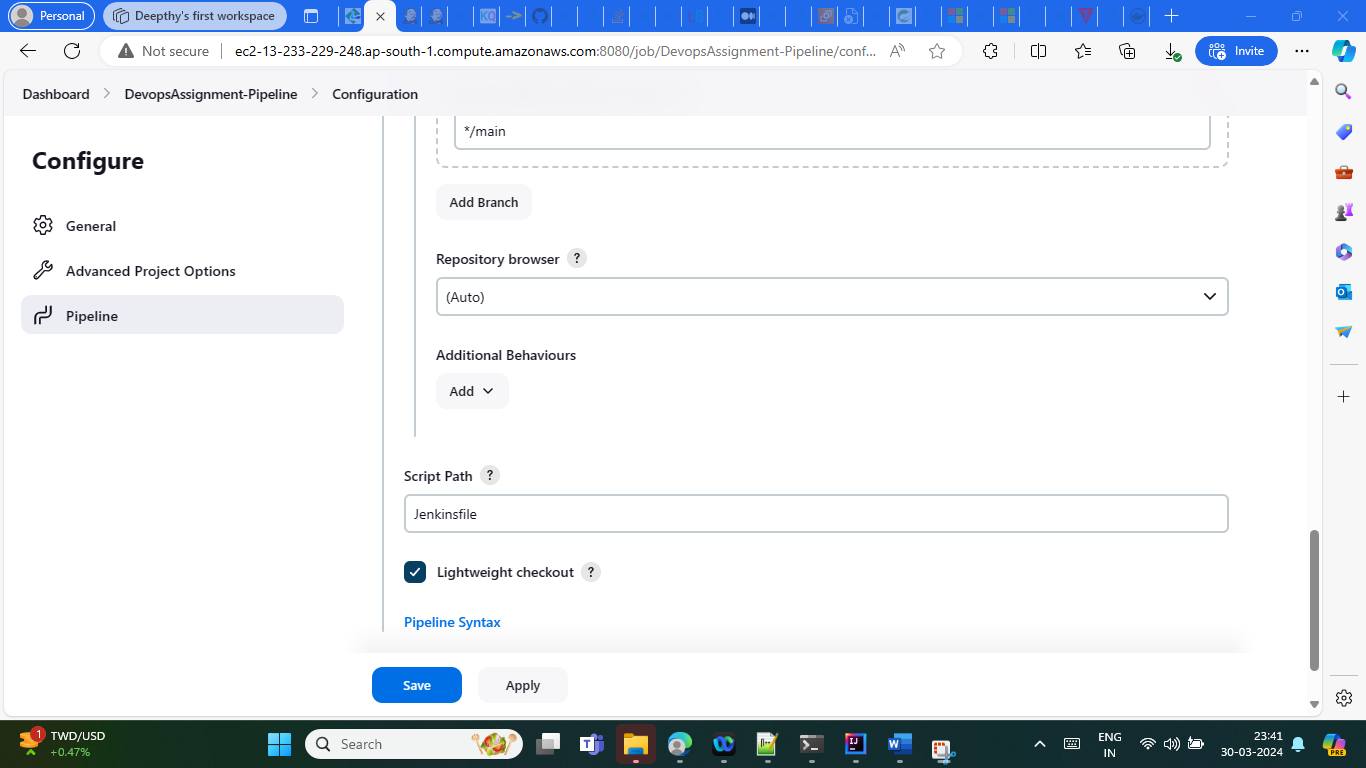




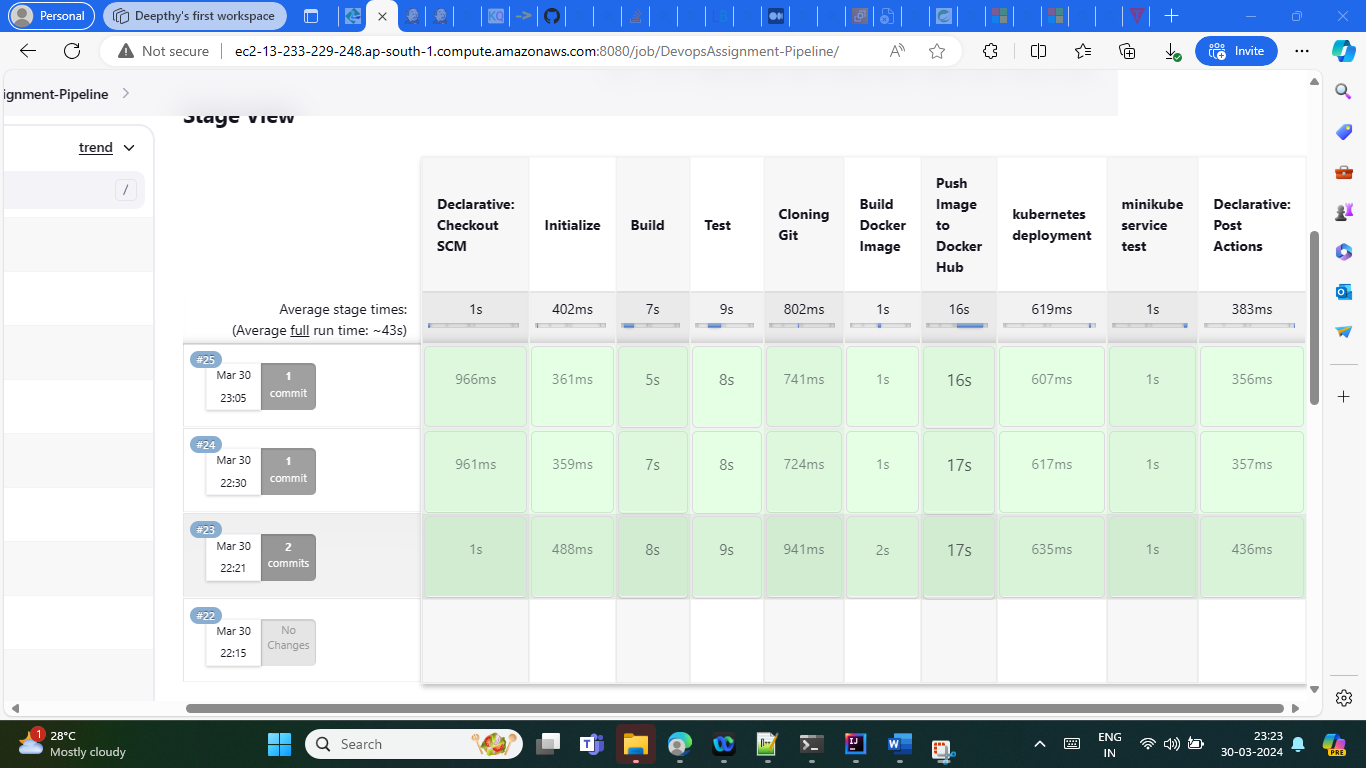


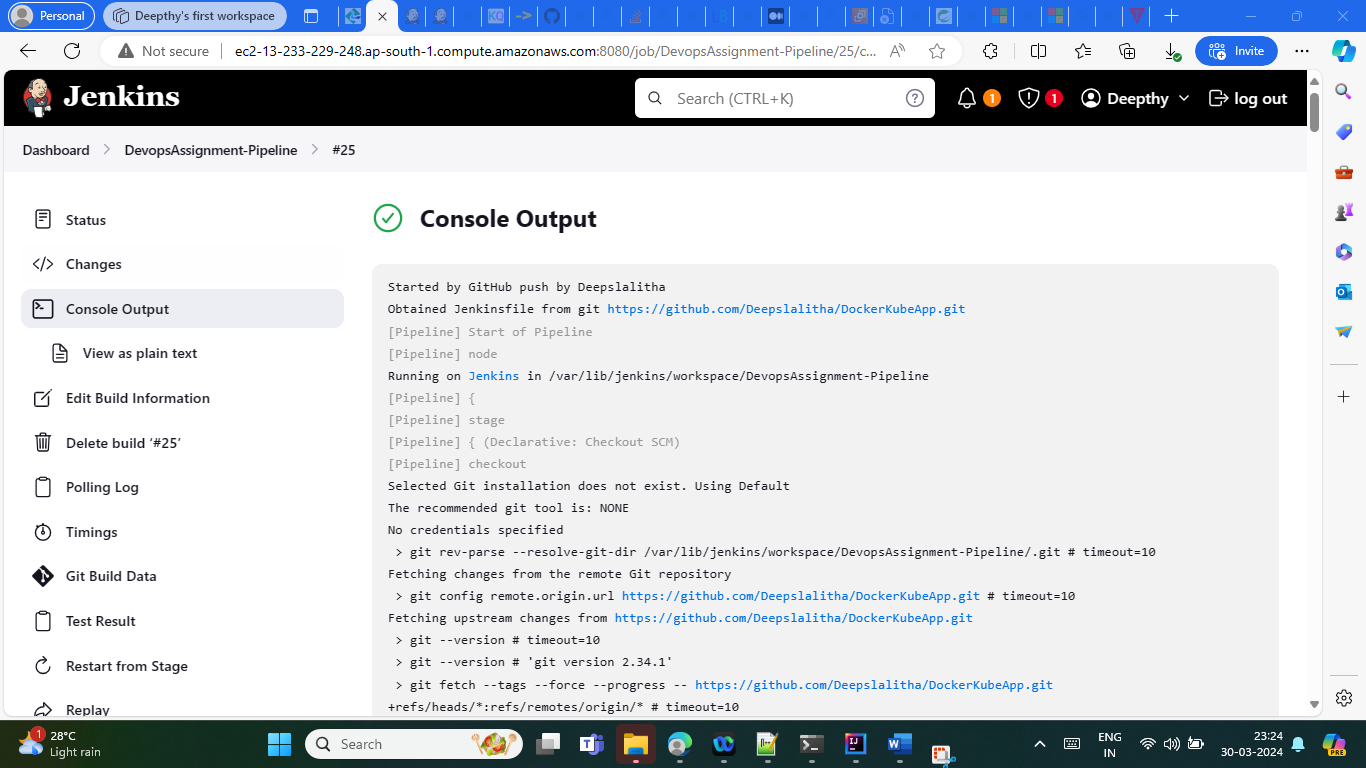
**Pipeline set UP** – Used Jenkinsfile from gIt



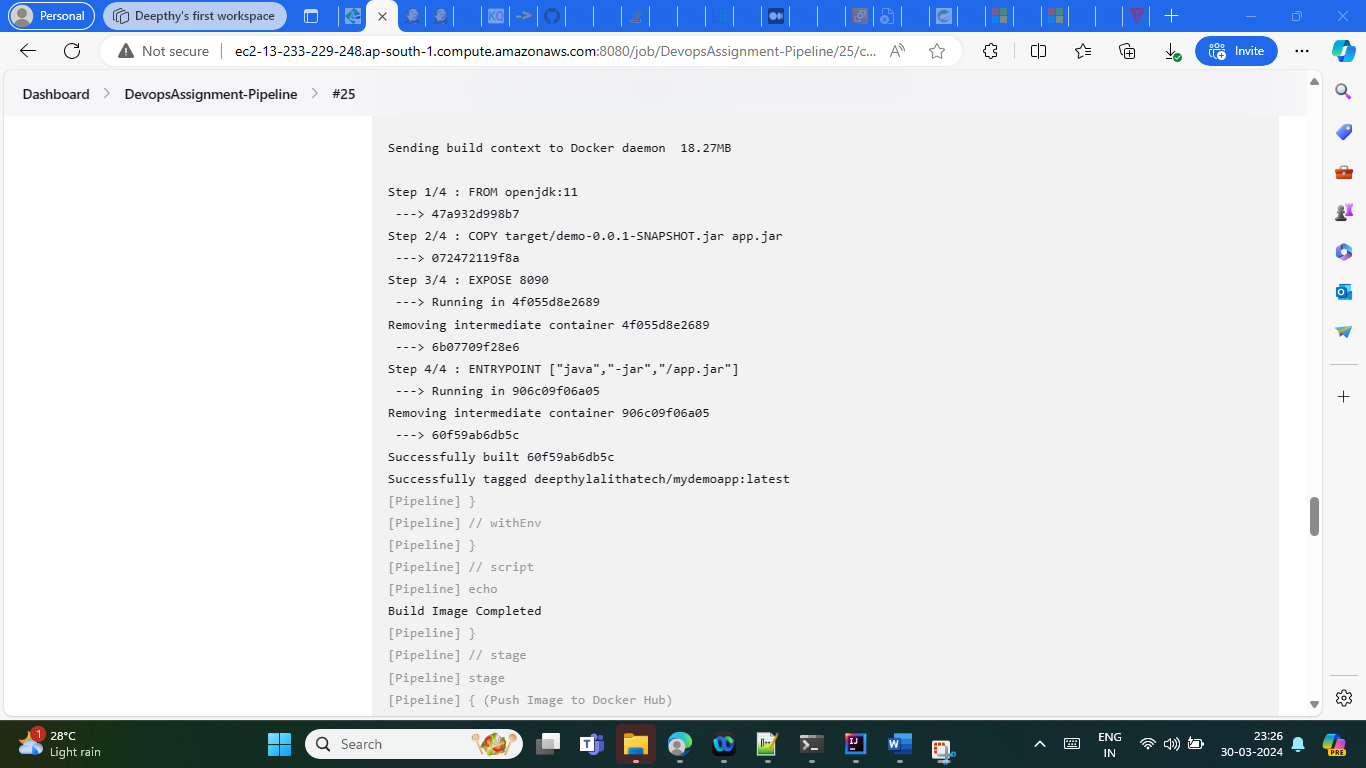


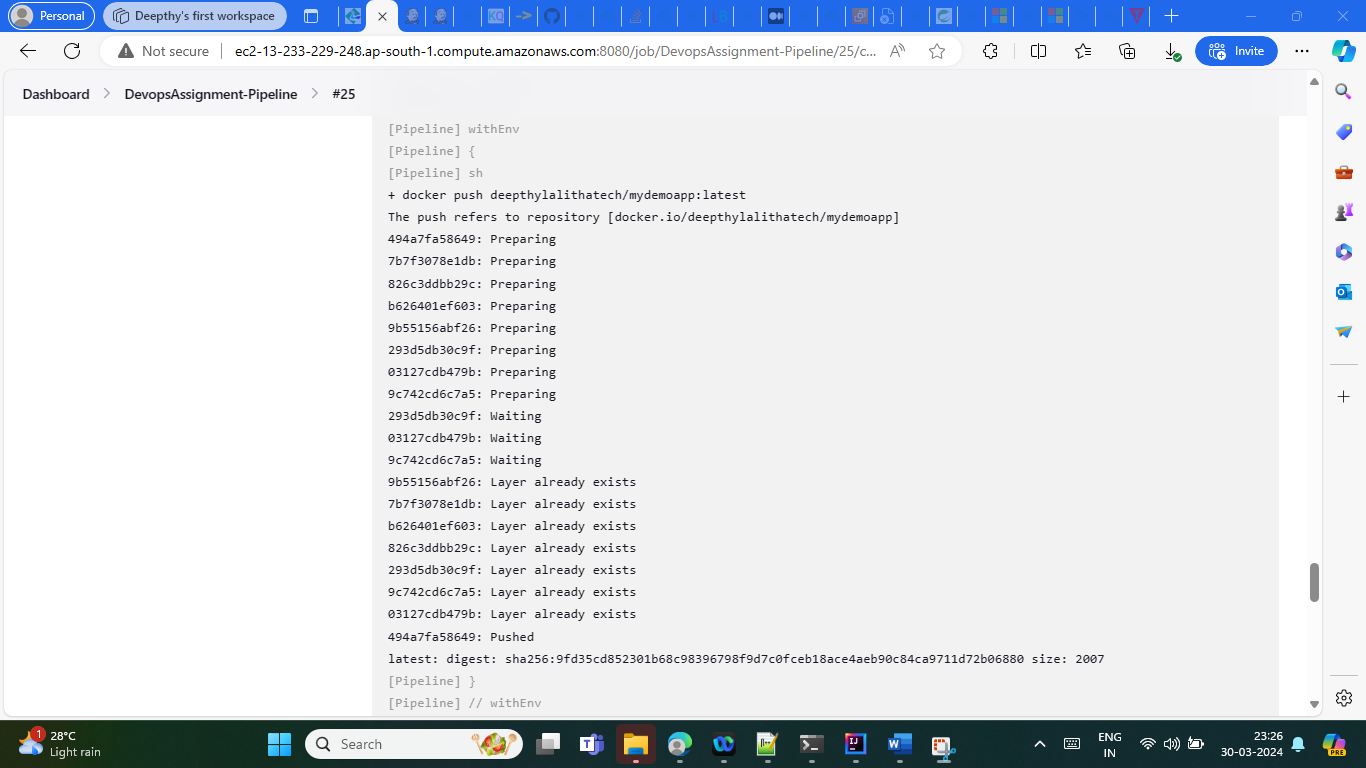
**Jenkins Pipeline execution Images**



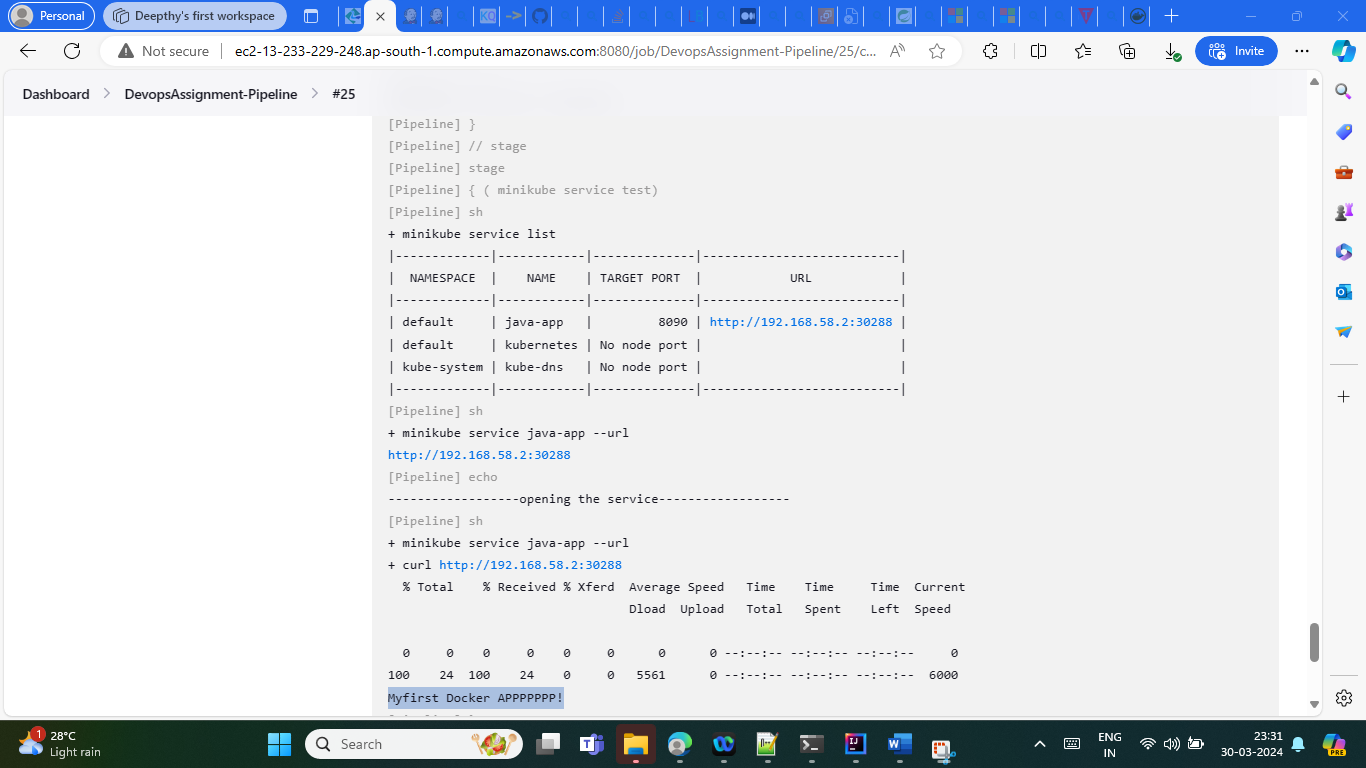


**Docker Build and Push:**

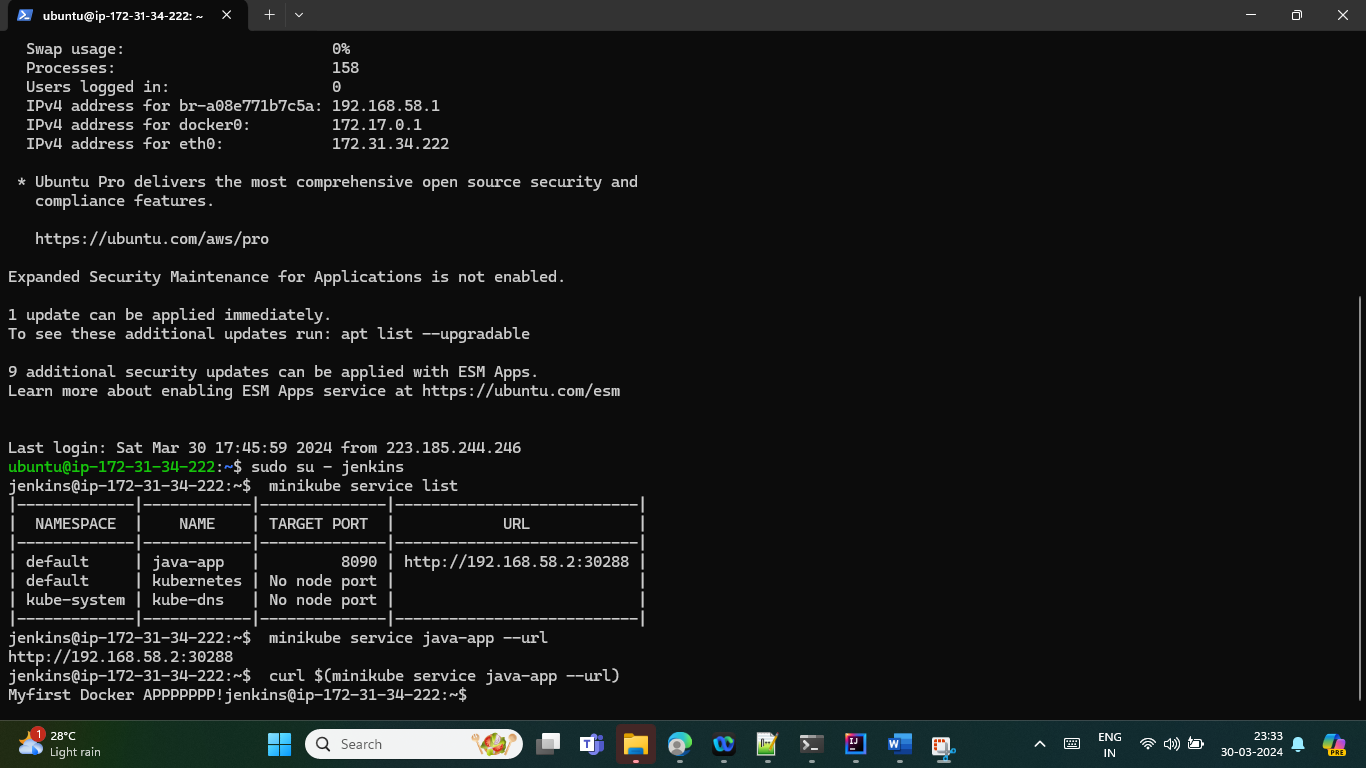




**Minikube deployment**



Mini kube deployment status in ec2 Instane (where kube is set up)



1. IAC:

I tried setting up a Azure Cosmos DB through terraform so that I could show some messages from DB . How ever there were some failures in the terraform Apply step though plan was successful

1. Containerization – Docker was used.

Docker file path -[DockerKubeApp/Dockerfile at main · Deepslalitha/DockerKubeApp (github.com)](https://github.com/Deepslalitha/DockerKubeApp/blob/main/Dockerfile)

1. Orchestration - Minikube

Kube deployment files:

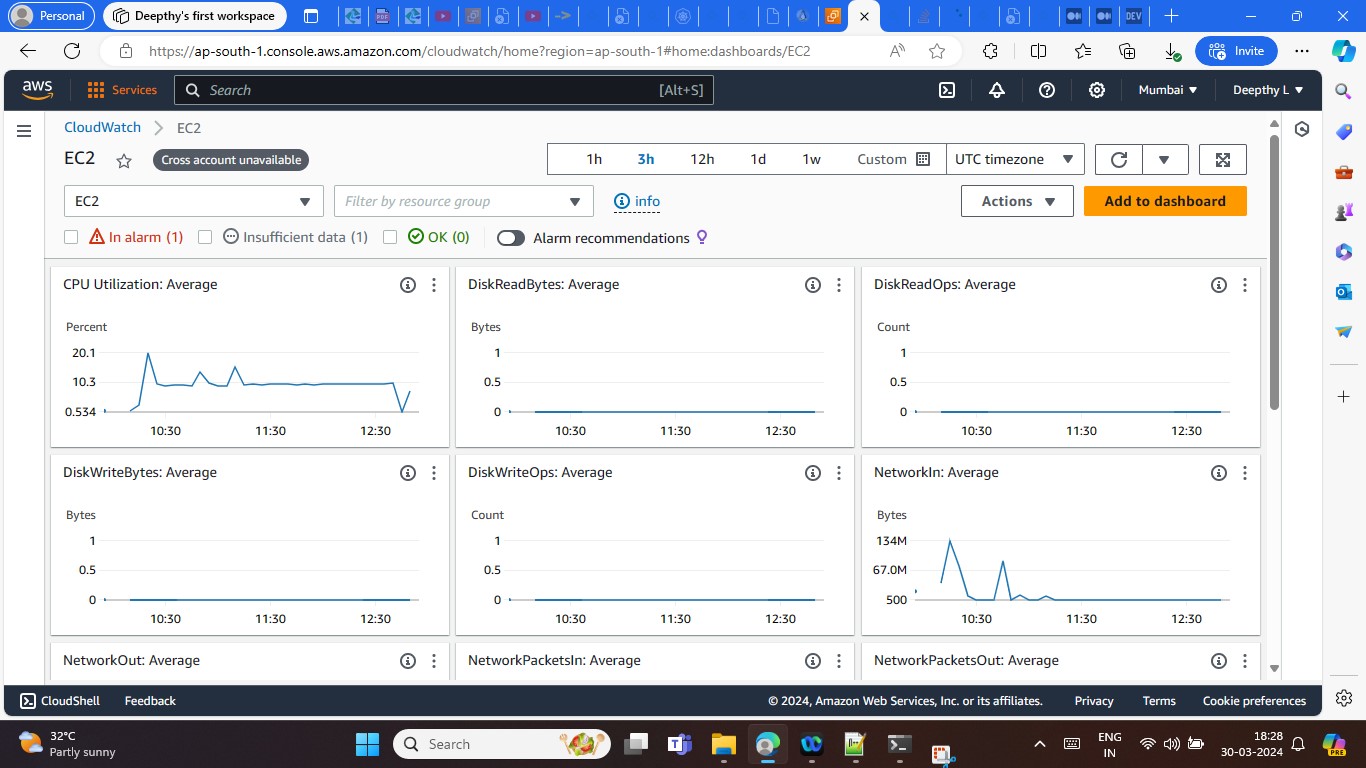
[DockerKubeApp/deploy-to-minikube.yaml at main · Deepslalitha/DockerKubeApp (github.com)](https://github.com/Deepslalitha/DockerKubeApp/blob/main/deploy-to-minikube.yaml)

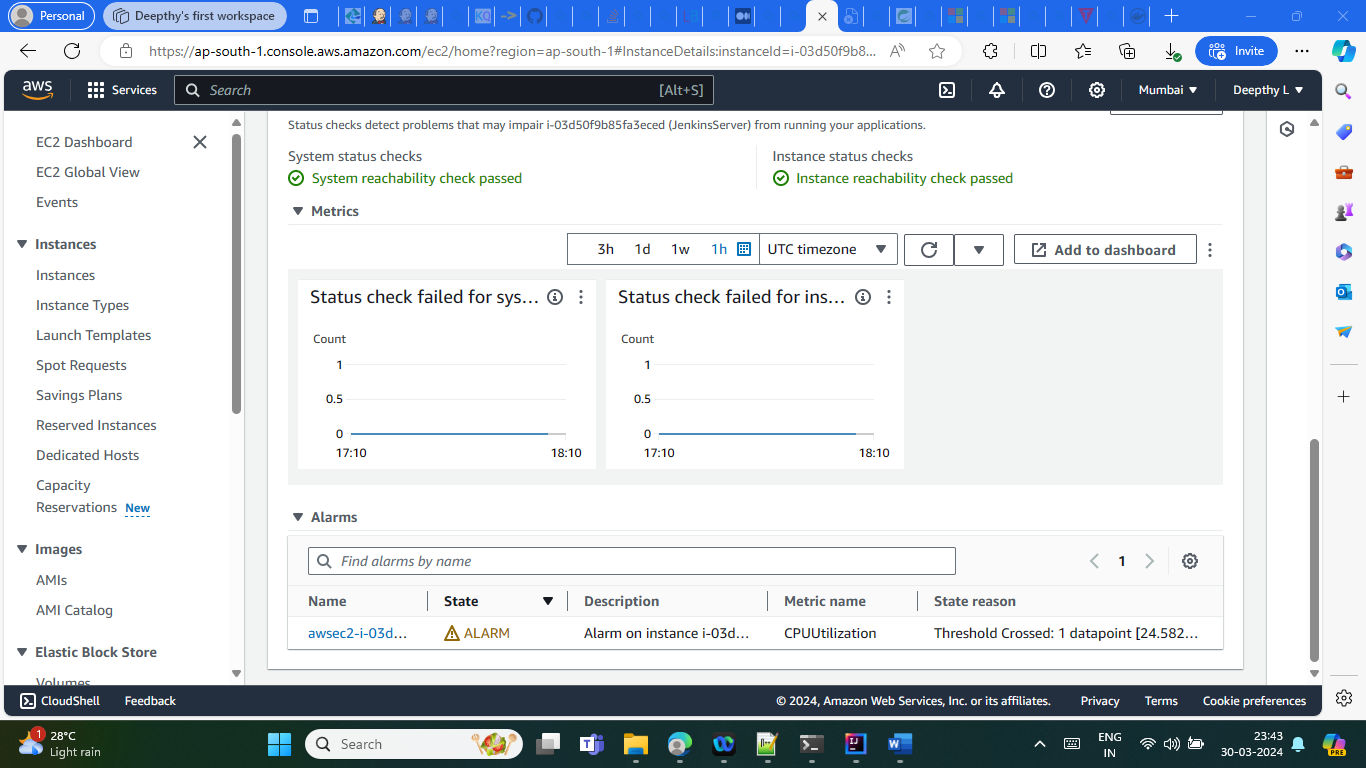
1. Monitoring & Logging

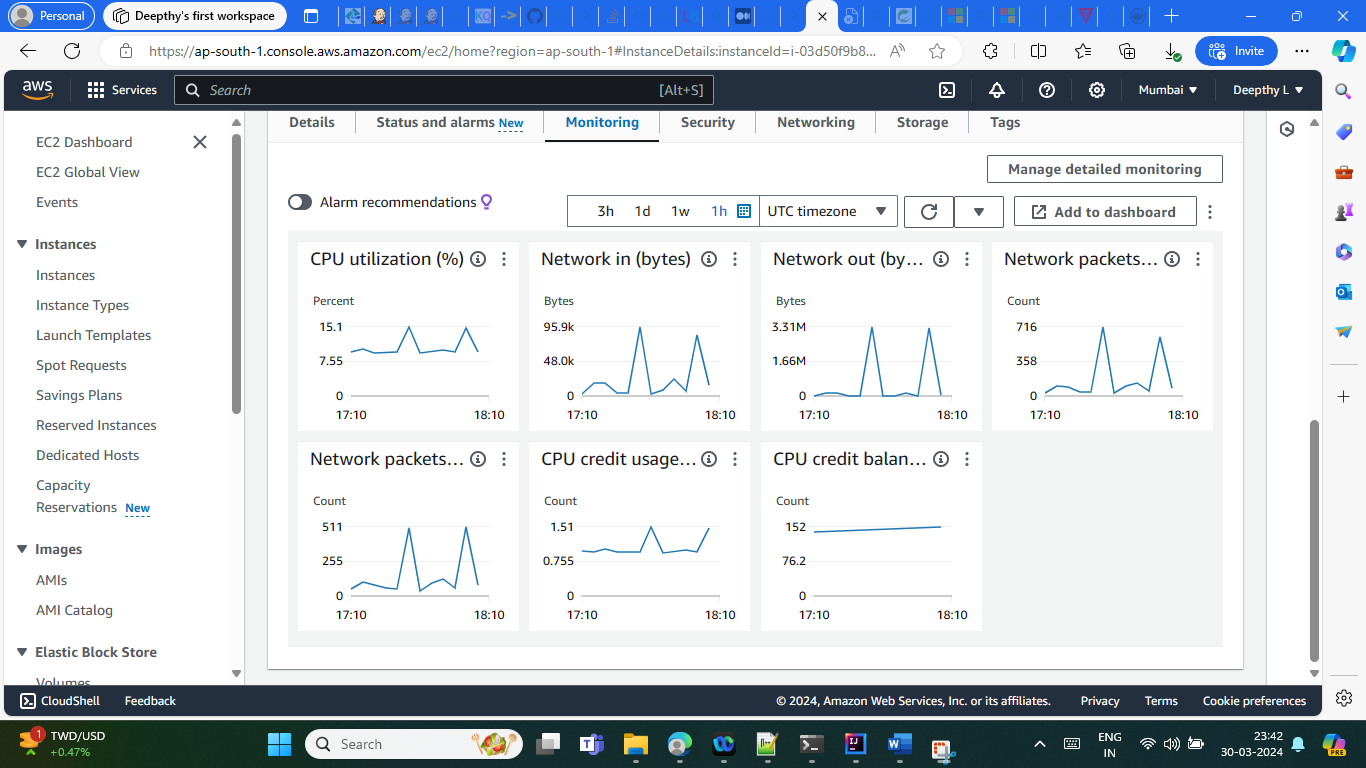
Since AWS cluster was set up - Used Cloud watch.

Alerts were set up for CPU utilization . There were alerts as I was using a t2.medium instance and at times , CPU utilization went high

Screenshots







Git Repo:

<https://github.com/Deepslalitha/DockerKubeApp/tree/main>