ICCD Automation in DevOps

## - Run Book Documentation

Enumeration of given task :

      Create a react.js app , build it as deployable container image, deploy it in cloud instances. Create a cicd flow any commit in github should trigger the build and deploy process.
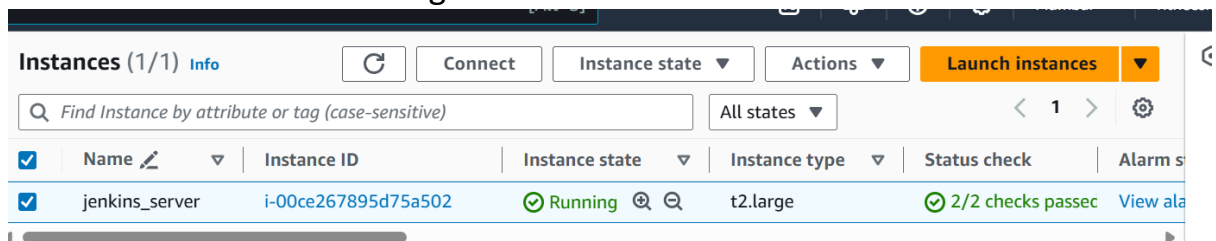
Solution:

1. Selecting of appropriate tool
   a. Github - repository
   b. Docker – containerisation tool
   c. Terraform – Infrasctructure As A Code tool
   d. AWS EKS – kubernetes cluster formation services from AWS
   e. Jenkins – Automation tool

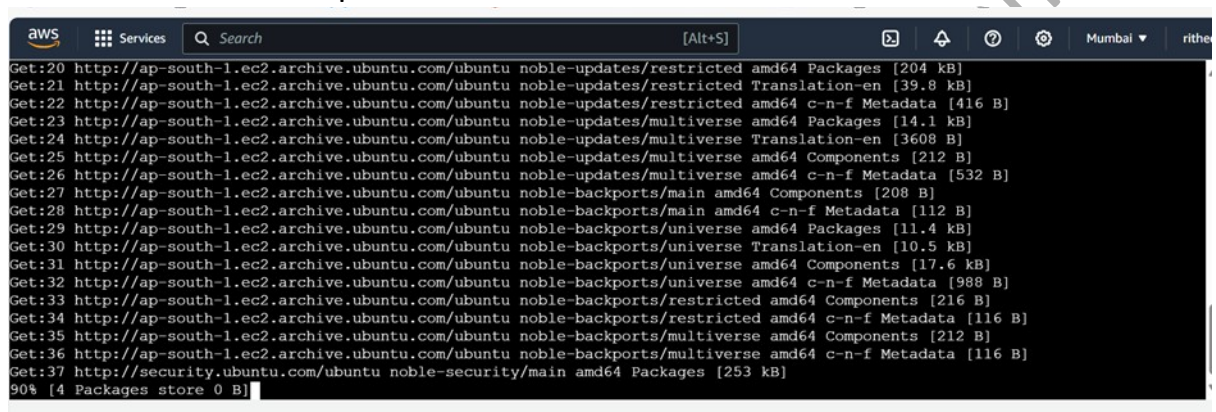2. Creating mind map for workflow

PROJECT BY
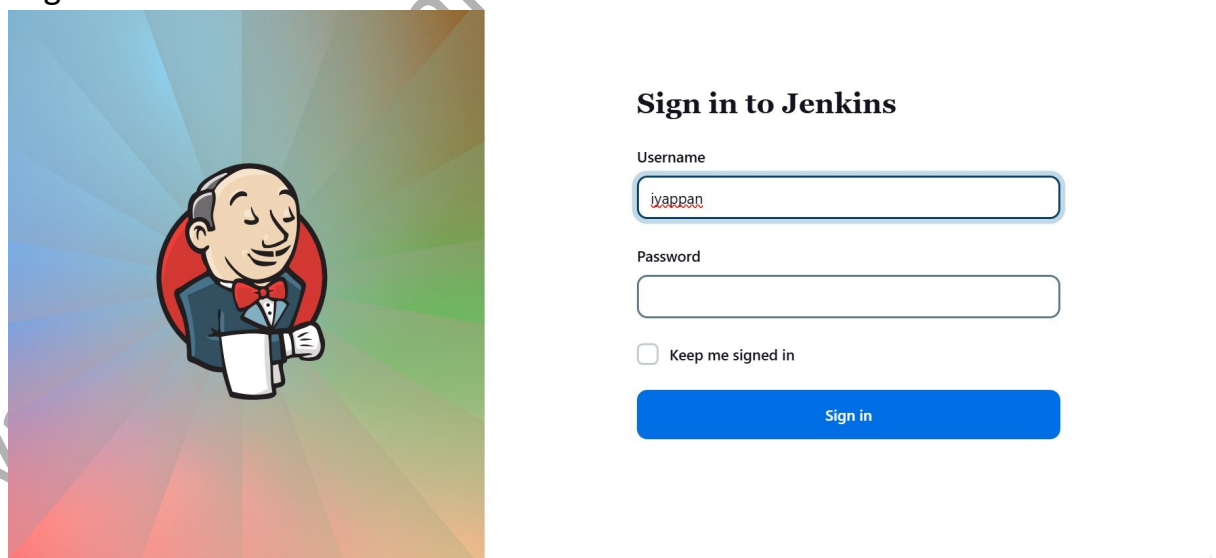IYAPPAN A
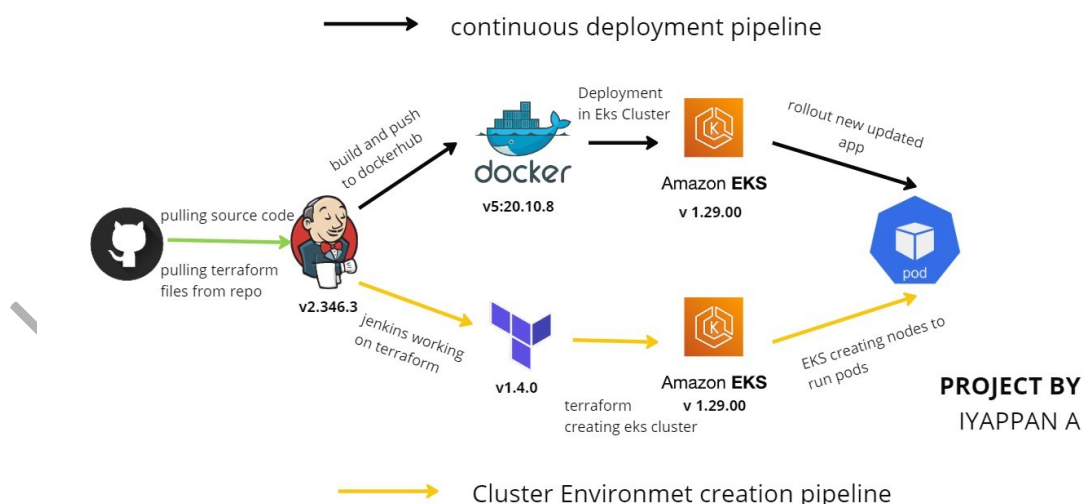
Run book:

1. Create a instance for running Jenkins

2. Install tools and its dependencies

3. Configure Jenkins to use those tools , give appropriate privileges to Jenkins.

4. Login into Jenkins console

   a. Configure Jenkins with credentials ,tools and plugins

5. Create a react js App using npm packages and upload it to Github

6. Create terraform file to establish an eks cluster and push it to Github
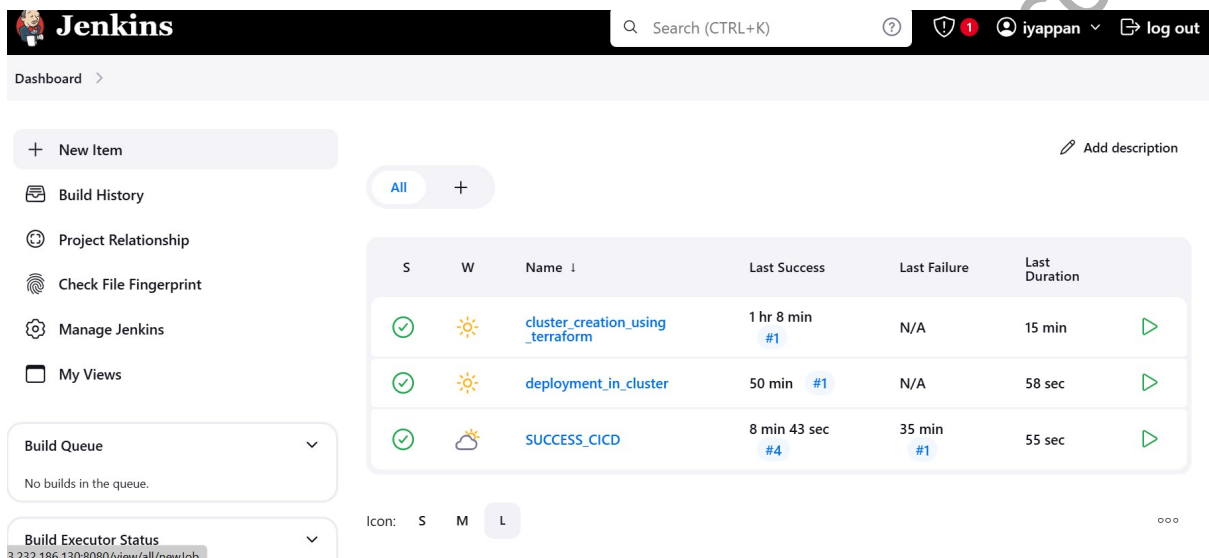
7. Create pipeline
    a. Pipeline – cluster_creation_using_terraform
    b. Pipeline – app deployment in cluster
       # first manual deployment is done to check whether the service is running properly or not
    c. Pipeline – SUCCESS_CICD pipeline is final automated pipeline
       # this pipeline is responsible for checking Github for any new commit and triggers build process automatically
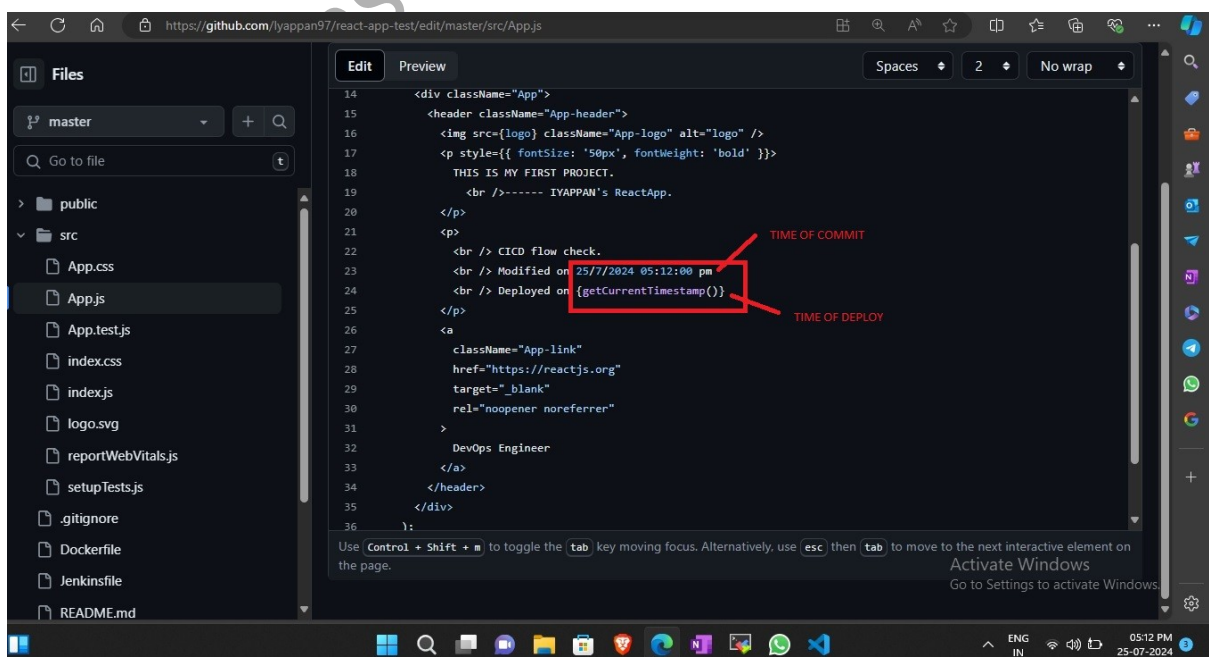

8. Make any new commit in Github repository
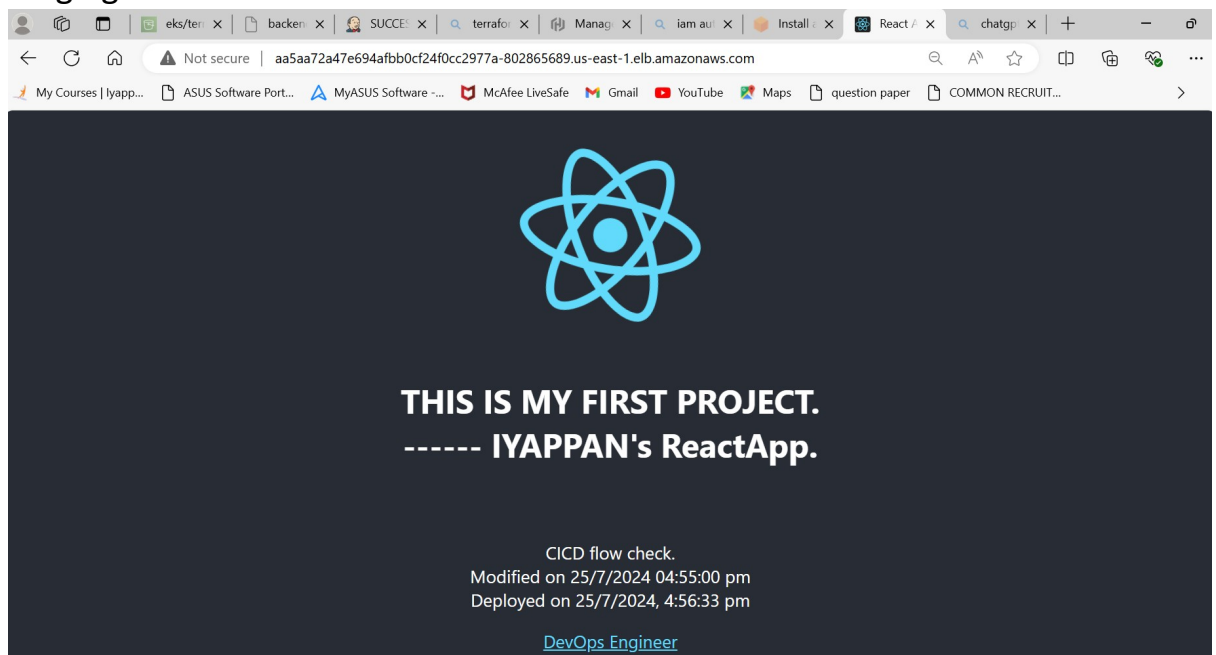   # While committing I add time manually of commit for reference

9. Testing
   a. This new commit will automatically trigger pipeline to build (SUCCESS_CICD)

```
[Pipeline] cleanws
[WS-CLEANUP] Deleting project workspace...
[WS-CLEANUP] Deferred wipeout is used...
[WS-CLEANUP] done
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

   b. Getting Loadbalancer IP from logs

```
[Pipeline] script
[Pipeline] {
[Pipeline] sh
+ kubectl rollout restart deployment/scalable-nginx-example
deployment.apps/scalable-nginx-example restarted
[Pipeline] sh
+ kubectl get svc
NAME              TYPE            CLUSTER-IP        EXTERNAL-IP
PORT(S)           AGE
kubernetes        ClusterIP       172.20.0.1        <none>
443/TCP           53m
nginx-example     LoadBalancer    172.20.229.215    aa5aa72a47e694afbb0cf24f0cc2977a-802865689.us-east-
1.elb.amazonaws.com   80:30991/TCP    41m
[Pipeline] }
[Pipeline] // script
[Pipeline] }
```

10. Pinging the url from browser

THIS IS MY FIRST PROJECT.
------ IYAPPAN's ReactApp.

CICD flow check.
Modified on 25/7/2024 04:55:00 pm
Deployed on 25/7/2024, 4:56:33 pm

DevOps Engineer

- Newly committed changes in the React app source code are visible in the browser.
- The provided timestamp shows the time of the commit.
- The deployment time is also displayed.
- The automated pipeline took only 1.5 minutes to build and deploy the app from the newly committed source code.
- Finally ,The CI/CD pipeline has been created from scratch and tested.