CI/CD Pipeline Project

# Steps:

1. Create an AWS EC2 instance (ubuntu)
2. Install Jenkins on AWS
   1. *Install Java*
3. sudo apt update
4. sudo apt install openjdk-11-jre
5. java --version
   1. *Install Jenkins*
6. curl -fsSL https://pkg.jenkins.io/debian/jenkins.io-2023.key | sudo tee \

/usr/share/keyrings/jenkins-keyring.asc > /dev/null

1. echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \

https://pkg.jenkins.io/debian binary/ | sudo tee \

/etc/apt/sources.list.d/jenkins.list > /dev/null

1. sudo apt-get update
2. sudo apt-get install fontconfig openjdk-11-jre
3. sudo apt-get install jenkins
   1. *Start Jenkins*
4. sudo systemctl enable jenkins
5. sudo systemctl start jenkins
6. sudo systemctl status jenkins
7. Open port 8080 from AWS Console
8. Unlock Jenkins
   1. sudo cat /var/lib/jenkins/secrets/initialAdminPassword
9. Customize Jenkins
   1. Install suggested plugins
   2. Create First Admin User
   3. Save and Finish
10. Jenkins
    1. New item
    2. Freestyle project
    3. Select Github project checkbox
    4. Enter the github project repository url
    5. Source Code Management -> Select Git
    6. Enter same url
    7. Goto aws EC2 terminal
    8. ssh-keygen
    9. cd .ssh
    10. ls
    11. cat id\_rsa -> This gives Primary key
    12. cat id\_rsa.pub -> This gives Public key
    13. Goto Github -> settings -> SSH and GPG keys
    14. New SSH key
    15. Enter the public key in the ‘Key’ section
    16. Goto Jenkins -> Add Credentials
    17. In Kind section select : SSH Username with private key
    18. ID & Description – any
    19. Username – ubuntu
    20. Private key -> Enter directly -> Add -> Enter the private key
    21. Select the created credential in dropdown
    22. Save
    23. Then click on Build now in left pane
    24. Click on Console output
    25. Building in workspace “/var/lib/jenkins/workspace/todo-app”
11. Goto Instances -> Select the instance -> Security -> Security group link -> Edit inbound rules
12. Add rule: Custom TCP -> 8000 -> Anywhere – Ipv4
13. Save rule
14. Aws EC2 terminal:
    1. cd -> to come out of .ssh folder
    2. cd /var/lib/jenkins/workspace/todo-app
    3. ls
    4. sudo rm Dockerfile
    5. sudo apt install docker.io
    6. sudo vim Dockerfile
    7. Press “i”
    8. FROM node:12.2.0-alpine
    9. WORKDIR app
    10. COPY . .
    11. RUN npm install
    12. EXPOSE 8000
    13. CMD [“node”,”app.js”]
    14. Then press “Esc” -> :wq -> Press “Enter”
15. sudo usermod -a -G docker $USER
16. sudo reboot
17. Goto to Instance and Connect again
18. cd /var/lib/jenkins/workspace/todo-app
19. docker build . -t todo-app
20. docker run -d --name container-app -p 8000:8000 todo-app
21. docker ps
22. docker kill “Container ID”
23. Goto Jenkins -> Configure -> Build steps -> Execute shell
24. Save
25. sudo chmod 777 /var/lib/jenkins/workspace/todo-app
26. sudo usermod -a -G docker jenkins
27. sudo systemctl restart jenkins
28. Goto Jenkins -> Build now
29. If(Console output : FAILURE) -> Change “Container name”
30. Else Console output : SUCCESS
31. Goto Jenkins -> Dashboard -> Manage Jenkins -> Plugins -> Available Plugins -> “github integration” -> Install without restart
32. Add rule: Custom TCP -> 8080 -> Anywhere – Ipv4
33. Goto Github repository settings -> Webhooks
34. Enter the Jenkins page url till 8080
    1. Eg: <http://3.94.206.105:8080/github-webhook/>
    2. Content type : application/json
    3. Save
35. Refresh Github page -> you will see tick mark
36. Goto Jenkins -> Configure -> Build Triggers -> Select “GitHub hook trigger for GITScm polling”
37. Save