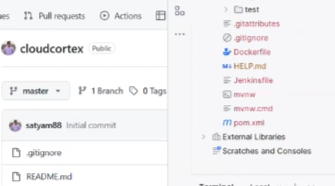
Continued from 13 July - work on Jenkins host

>> Goto Github >> Create repository >> name=”cloudcortex” >> public domain >> gitignore java >> Create >> Copy repo URL

Goto IntelliJ >> Goto VCS >> update repository to GIT >> update GIT and update manage repository >> new github URL

* Git pull origin master ……Pulls gitignore and README from repo to intelliJ



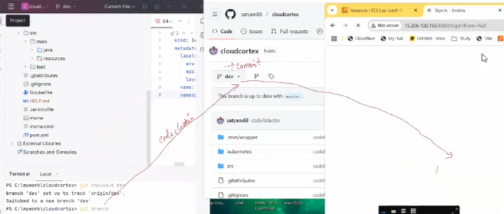
* Git add –all
* Git commit -a “Code Refactor”
* Git push origin master -f

>> Goto Github when the push from intelliJ completed successfully.

>> Create 3 more branches dev, preprod, prod 

* Git fetch origin dev
* Git branch
* Git checkout dev

Summary of expected code movement >> update IP of jenkins



>> Goto Jenkins >> New >> Multibranch pipeline >> cloudcortex >> Display name=cloudcortex >> Set GIT and update GIT URL >> Save >> Goto Settings >> Create webhook

 …..Saving starts branch scan

>> Verify dev branch if log looks fine for Maven\_3.9.9 and update Jenkinsfile accordingly

>> Update intelliJ documents environment-wise for preprod and prod

* Git add –all
* Git commit -m “mvn change”
* Git push origin dev >> Scan multibranch pipeline and see status in Jenkins



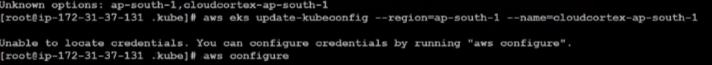
>> Merge PR from dev >> preprod >> prod >> Build manually each time ……Jenkins runs stages as per environment and Jenkinsfile match



Resolve all above errors



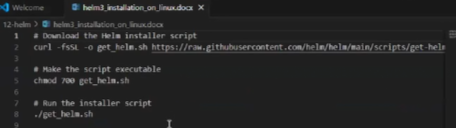
Create cluster and worker node when required → Login to jumphost

Take kube-config file from :

Verify and see if kuebctl is returning cluster information:

* Kubectl get node
* Kubectl get ns
* Kubectl create ns dev; kubectl create ns preprod; kubectl create ns prod

Install Helm



| Also try connecting to cluster from Jump host:    And install helm here as well |
| --- |

When we are comfortable to see docker images uploaded to ECR, we will need an ingress controller before jenkins deployment happens in pp & prod. ……v.imp step

* Kubectl create ns ingress-nginx ……on jump-host.

>> Anything to install on kubernetes, HELM is mandatory

>> If I install NGINX ingress controller k8s with helm what all resource will it create in CHATGPT, shall we go with nlb or alb service in nginx in CHATGPT ……critical nginxserviceAWS compatibility tests

>> NGINX vs TOMCAT in CHATGPT

>> kubectl get all -n ingress-nginx; kubectl delete ns ingress-nginx ……1st command gives everything install with that ns, 2nd command delete everything under that ns

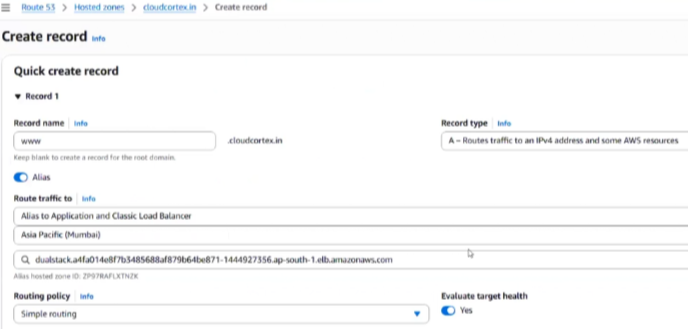
* Helm repo add ingress-nginx <https://kubernetes.github.io/ingress-nginx>
* Helm repo update



* Git add –all
* Git commit “fix”
* Git push origin dev ……Triggers pipeline

>> Fix errors >> check if jenkins has login credentials of binbash in /etc/passwd and should not be /bin/false

>> Goto Route53 >> Create record [www.cloudcortex.in](http://www.cloudcortex.in) >> update the ingress load balancer to www domain in Route53



>> When anyone hits [www.cloudcortex.in](http://www.cloudcortex.in), it will ping rules - host: [dev.cloudcortx.in](http://dev.cloudcortx.in) in 00-ingress.yaml; and when anyone hits [www.cloudcortex.in/preprod](http://www.cloudcortex.in/preprod), it will ping preprod directory

>> When code splits at “Deploy app to dev env” stage , clear the errors and Build pipeline with updated jenkins user

>> Deployment finished in dev, verify the deployment in jump host:

* Kubectl get all -n dev
* Kubectl get pod -n dev …count matches with deployment.yaml



* Kubectl get deploy -n dev …name matches deployment name



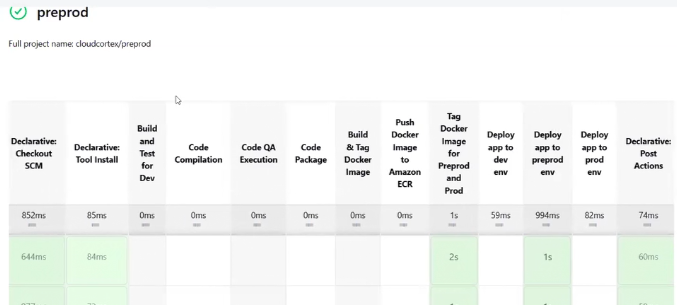
* Kubectl get ing -n dev

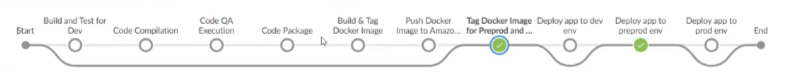


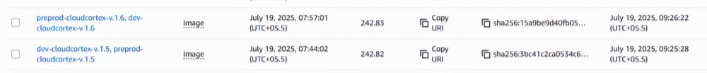
* Kubectl get svc -n dev –show-labels
* Kubectl get deploy -n dev –show-labels
* Kubectl get po -n dev –show-labels …all labels in 3 commands match



>> Goto Github and merge PR to preprod from dev





... ECR images

>> Applies the same for PROD, update your jenkinsfile to use 1. Sonarqube integration, 2. Upload jar war to nexus, 3. Docker image scan