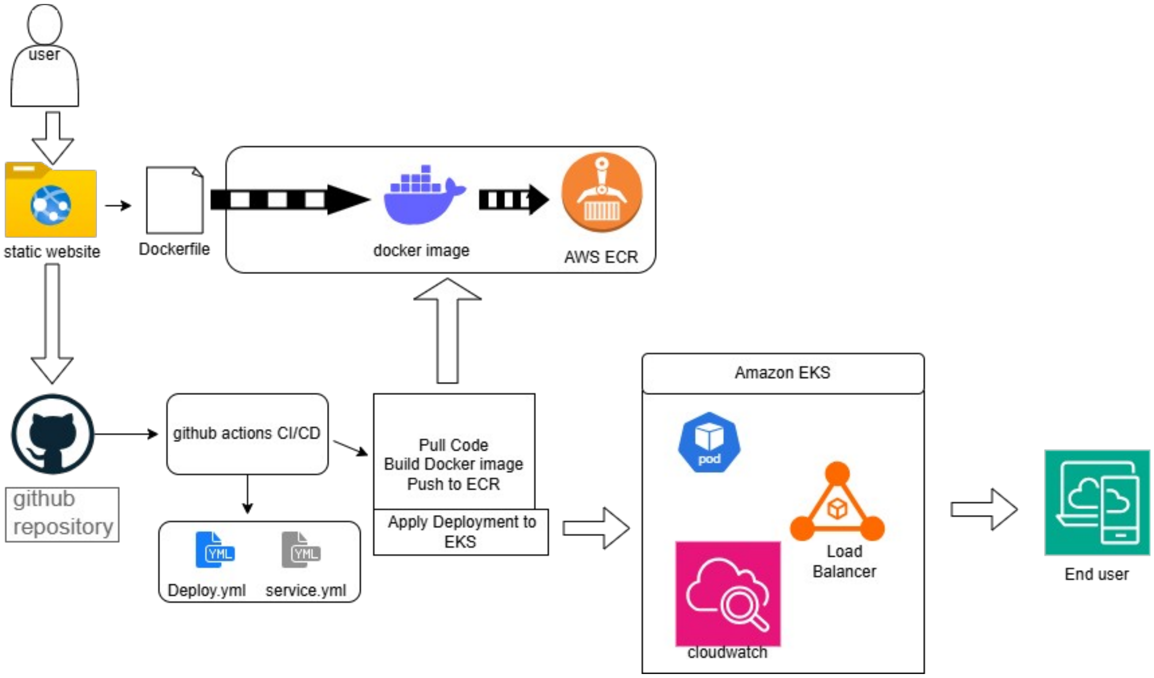
Brain Tasks – DevOps Project Documentation

1. Overview

This project demonstrates a complete DevOps workflow using Docker, Amazon ECR, Amazon EKS, GitHub Actions, and CloudWatch Observability.

2. Architecture Diagram



3. Worked Environment:

1. Linuxmint 22.2 xfce installed in virtualbox
2. Aws Freetire account
3. Browser

4. Tools Used

1. Docker
2.AWSCLI
3. Create Amazon ECR
4. Amazon EKS
5. GitHub Actions
6. CloudWatch
7. Kubernetes

5. Steps Performed

1. Created static website.

2. Dockerized the application.

3. Pushed image to Amazon ECR.

4. Created EKS cluster using eksctl.

5. Deployed application using Deployment & Service YAML.

6. Exposed LoadBalancer to access application.

7. Added CloudWatch Observability Add-on.

8. Optional Auto-deployment using GitHub Actions.

6. Kubernetes Deployment YAML

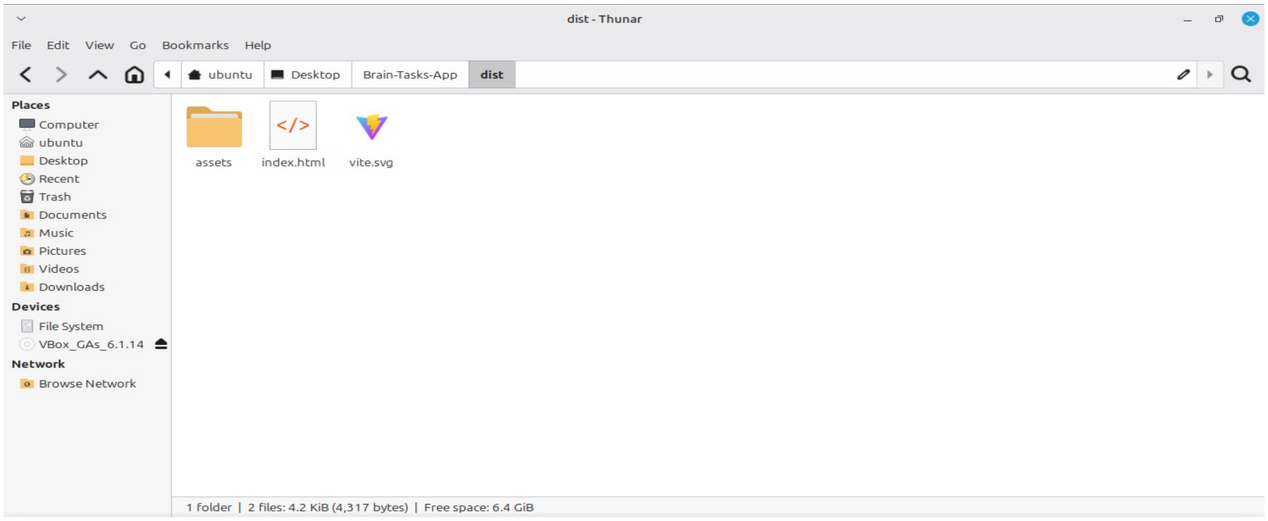
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: brain-static-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: brain-static
  template:
    metadata:
      labels:
        app: brain-static
    spec:
      containers:
      - name: brain-static
        image: <ECR_URL>:latest
        imagePullPolicy: Always
        ports:
        - containerPort: 80
```

7. Service YAML

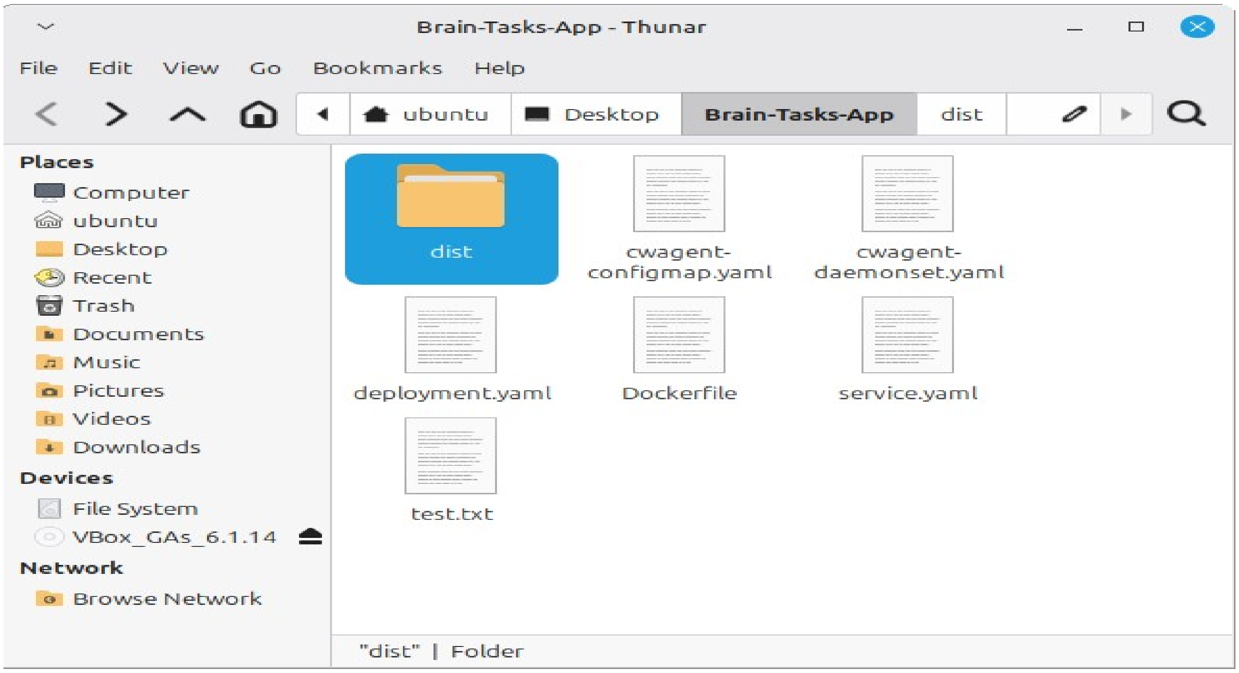
```
apiVersion: v1
kind: Service
metadata:
  name: brain-static-service
spec:
```

```
type: LoadBalancer
selector:
  app: brain-static
ports:
- port: 80
  targetPort: 80
```

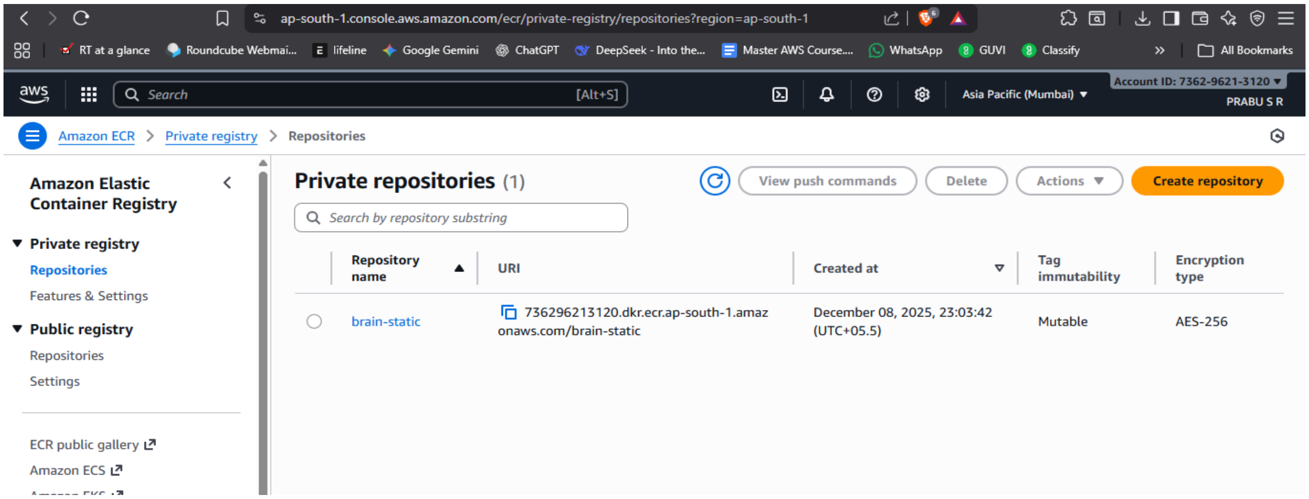
8. Screenshots



1 folder | 2 files: 4.2 KiB (4,317 bytes) | Free space: 6.4 GiB

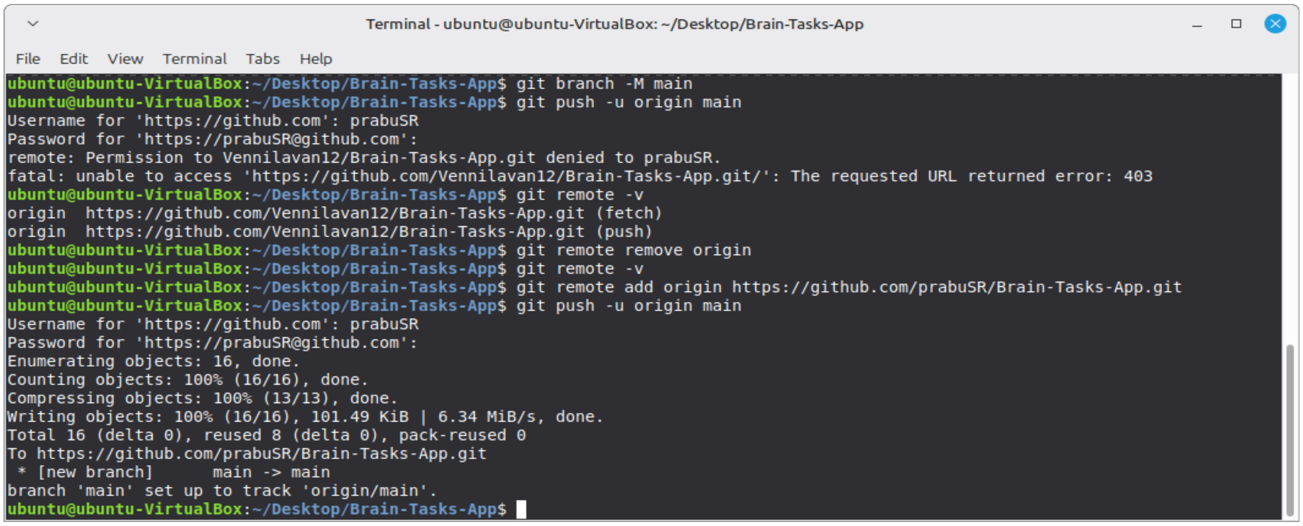


"dist" | Folder

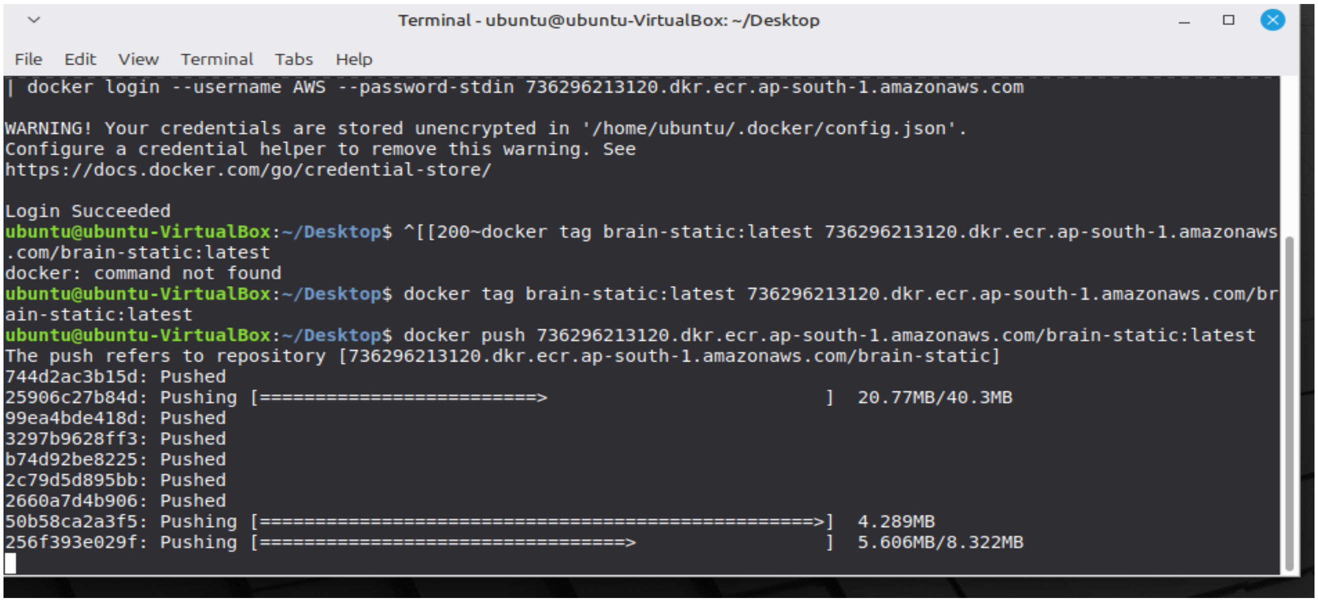


Private repositories (1)

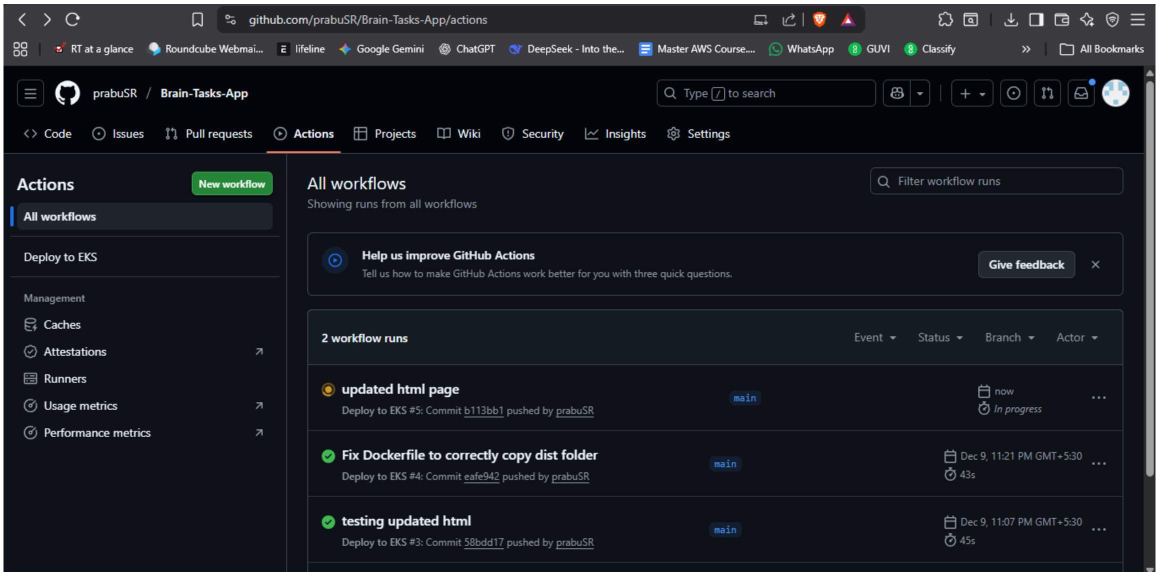
	Repository name	URI	Created at	Tag immutability	Encryption type
○	brain-static	736296213120.dkr.ecr.ap-south-1.amazonaws.com/brain-static	December 08, 2025, 23:03:42 (UTC+05.5)	Mutable	AES-256


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop/Brain-Tasks-App
File Edit View Terminal Tabs Help

```
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ git branch -M main
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ git push -u origin main
Username for 'https://github.com': prabuSR
Password for 'https://prabuSR@github.com':
remote: Permission to Vennilavan12/Brain-Tasks-App.git denied to prabuSR.
fatal: unable to access 'https://github.com/Vennilavan12/Brain-Tasks-App.git/': The requested URL returned error: 403
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ git remote -v
origin  https://github.com/Vennilavan12/Brain-Tasks-App.git (fetch)
origin  https://github.com/Vennilavan12/Brain-Tasks-App.git (push)
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ git remote remove origin
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ git remote -v
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ git remote add origin https://github.com/prabuSR/Brain-Tasks-App.git
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ git push -u origin main
Username for 'https://github.com': prabuSR
Password for 'https://prabuSR@github.com':
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Compressing objects: 100% (13/13), done.
Writing objects: 100% (16/16), 101.49 KiB | 6.34 MiB/s, done.
Total 16 (delta 0), reused 8 (delta 0), pack-reused 0
To https://github.com/prabuSR/Brain-Tasks-App.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ 
```


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop
File Edit View Terminal Tabs Help

```
| docker login --username AWS --password-stdin 736296213120.dkr.ecr.ap-south-1.amazonaws.com

WARNING! Your credentials are stored unencrypted in '/home/ubuntu/.docker/config.json'.
Configure a credential helper to remove this warning. See
https://docs.docker.com/go/credential-store/

Login Succeeded
ubuntu@ubuntu-VirtualBox:~/Desktop$ ^[[200~docker tag brain-static:latest 736296213120.dkr.ecr.ap-south-1.amazonaws
.com/brain-static:latest
docker: command not found
ubuntu@ubuntu-VirtualBox:~/Desktop$ docker tag brain-static:latest 736296213120.dkr.ecr.ap-south-1.amazonaws.com/br
ain-static:latest
ubuntu@ubuntu-VirtualBox:~/Desktop$ docker push 736296213120.dkr.ecr.ap-south-1.amazonaws.com/brain-static:latest
The push refers to repository [736296213120.dkr.ecr.ap-south-1.amazonaws.com/brain-static]
744d2ac3b15d: Pushed
25906c27b84d: Pushing [=========================>                        ]  20.77MB/40.3MB
99ea4bde418d: Pushed
3297b9628ff3: Pushed
b74d92be8225: Pushed
2c79d5d895bb: Pushed
2660a7d4b906: Pushed
50b58ca2a3f5: Pushing [==================================================>]  4.289MB
256f393e029f: Pushing [=================================>                ]  5.606MB/8.322MB
```



All workflows

Showing runs from all workflows

Help us improve GitHub Actions

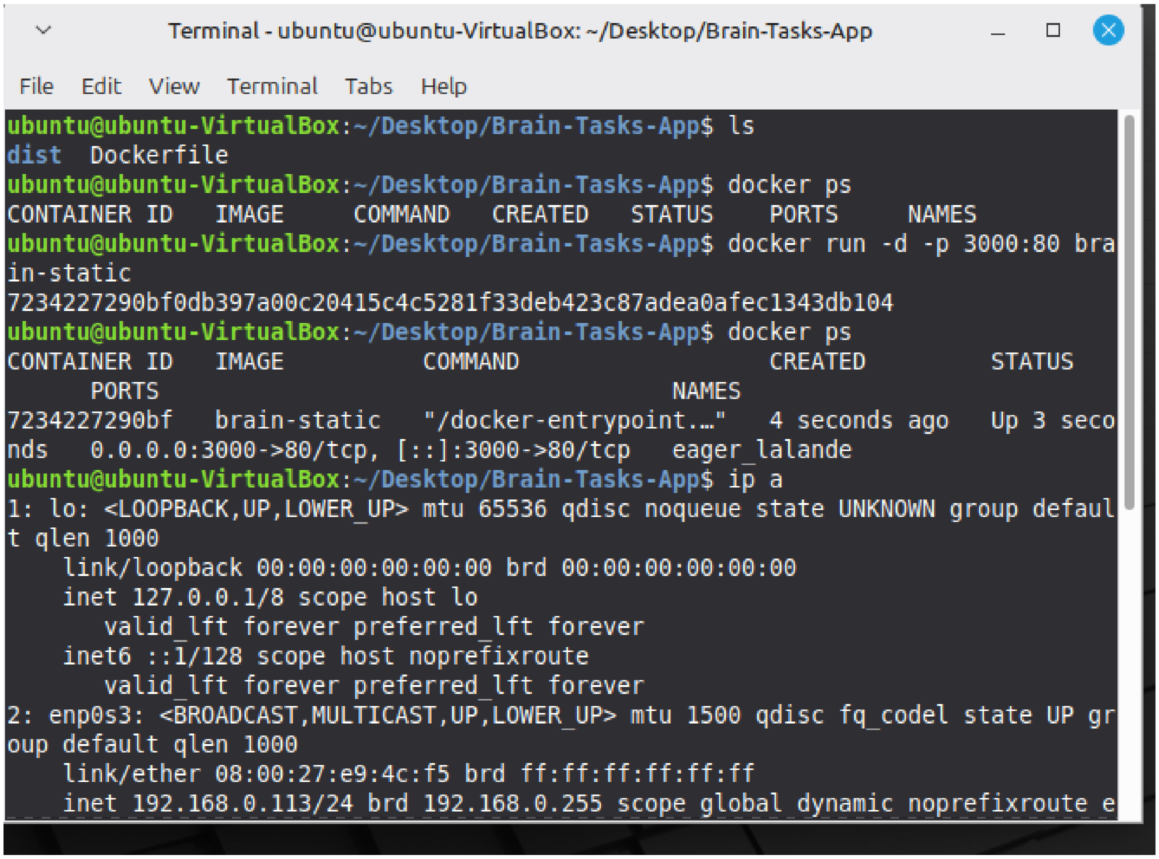
Tell us how to make GitHub Actions work better for you with three quick questions.

2 workflow runs

updated html page `main`
Deploy to EKS #5: Commit b113bb1 pushed by prabuSR
now
In progress

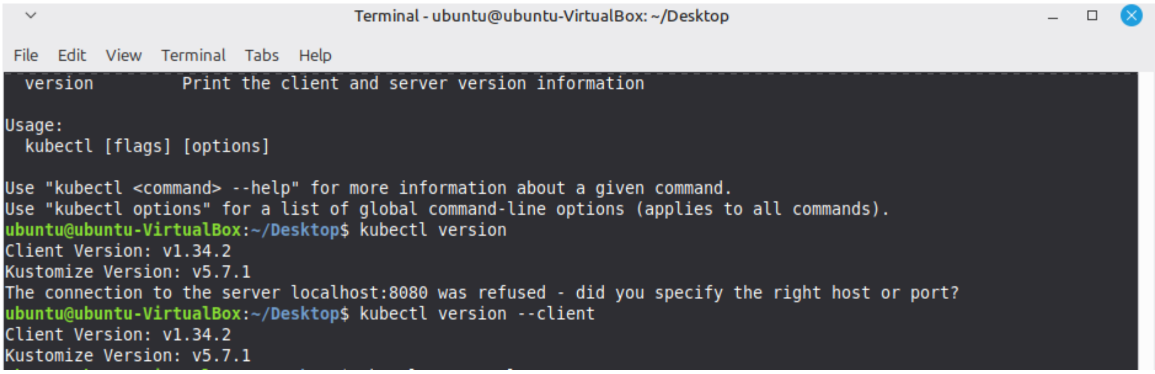
Fix Dockerfile to correctly copy dist folder `main`
Deploy to EKS #4: Commit eafe942 pushed by prabuSR
Dec 9, 11:21 PM GMT+5:30
43s

testing updated html `main`
Deploy to EKS #3: Commit 58bdd17 pushed by prabuSR
Dec 9, 11:07 PM GMT+5:30
45s

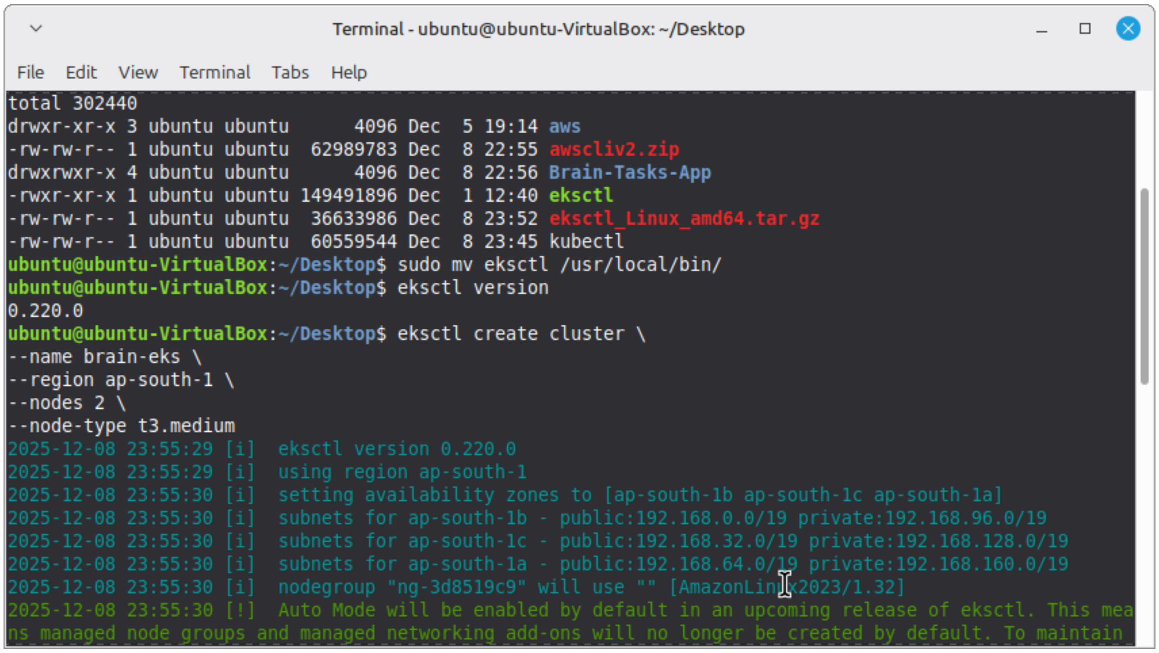


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop/Brain-Tasks-App
File
Edit
View
Terminal
Tabs
Help

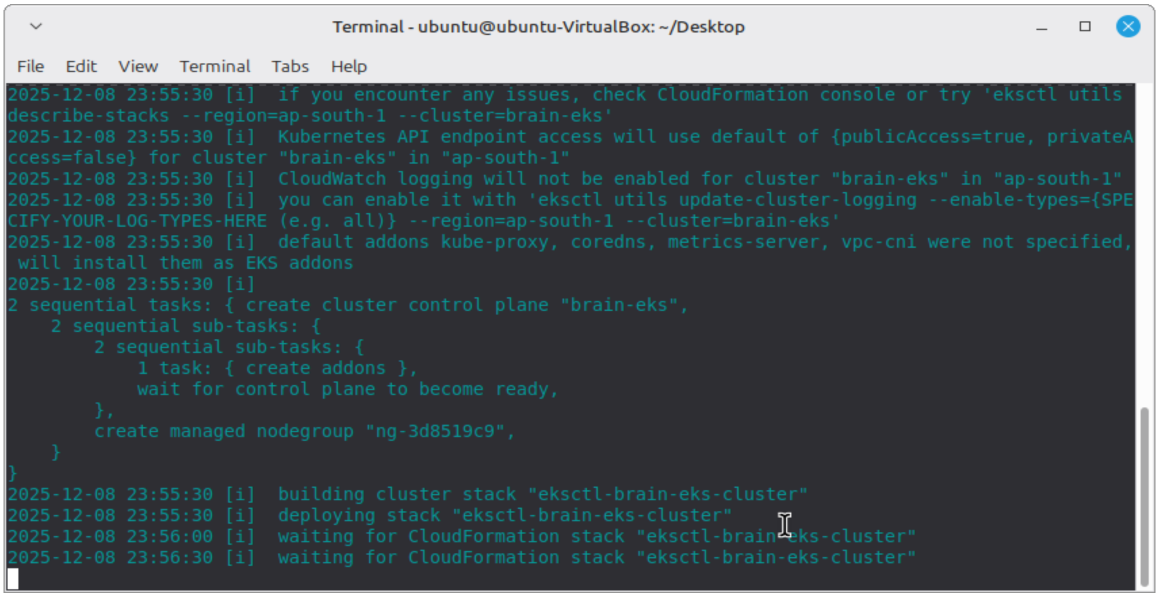
```
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ ls
dist  Dockerfile
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ docker run -d -p 3000:80 bra
in-static
7234227290bf0db397a00c20415c4c5281f33deb423c87adea0afec1343db104
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED         STATUS
      PORTS                                   NAMES
7234227290bf   brain-static   "/docker-entrypoint.…"   4 seconds ago   Up 3 seco
nds   0.0.0.0:3000->80/tcp, [::]:3000->80/tcp   eager_lalande
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:e9:4c:f5 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.113/24 brd 192.168.0.255 scope global dynamic noprefixroute e
```


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop
File Edit View Terminal Tabs Help

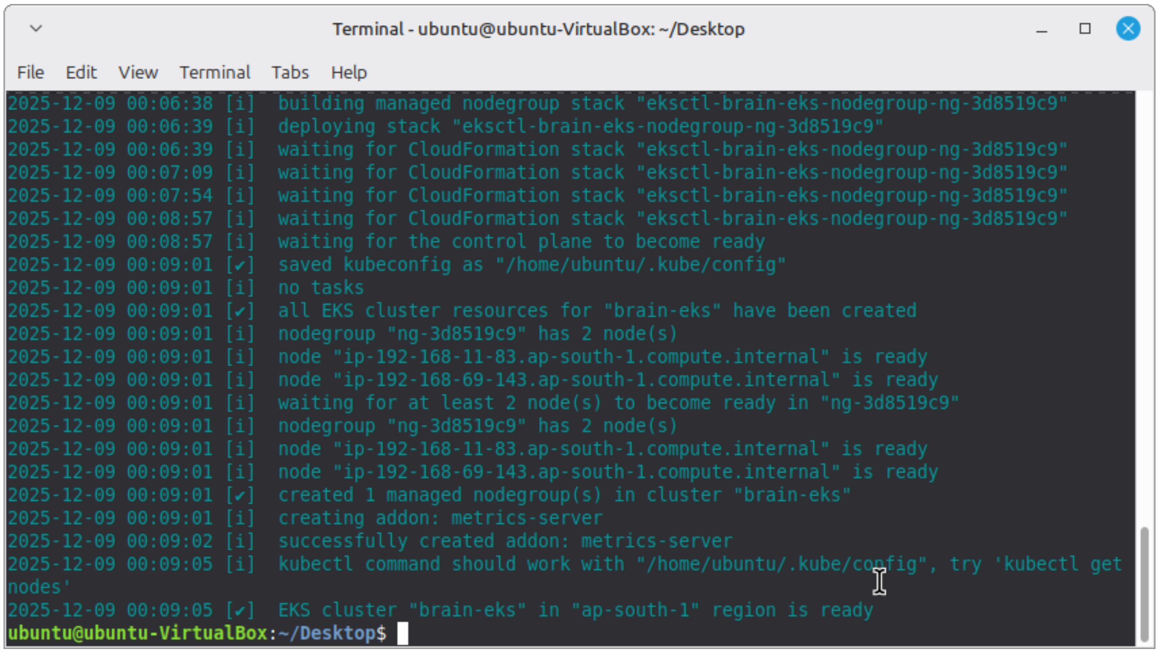
```
  version         Print the client and server version information

Usage:
  kubectl [flags] [options]

Use "kubectl <command> --help" for more information about a given command.
Use "kubectl options" for a list of global command-line options (applies to all commands).
ubuntu@ubuntu-VirtualBox:~/Desktop$ kubectl version
Client Version: v1.34.2
Kustomize Version: v5.7.1
The connection to the server localhost:8080 was refused - did you specify the right host or port?
ubuntu@ubuntu-VirtualBox:~/Desktop$ kubectl version --client
Client Version: v1.34.2
Kustomize Version: v5.7.1
```


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop
File Edit View Terminal Tabs Help

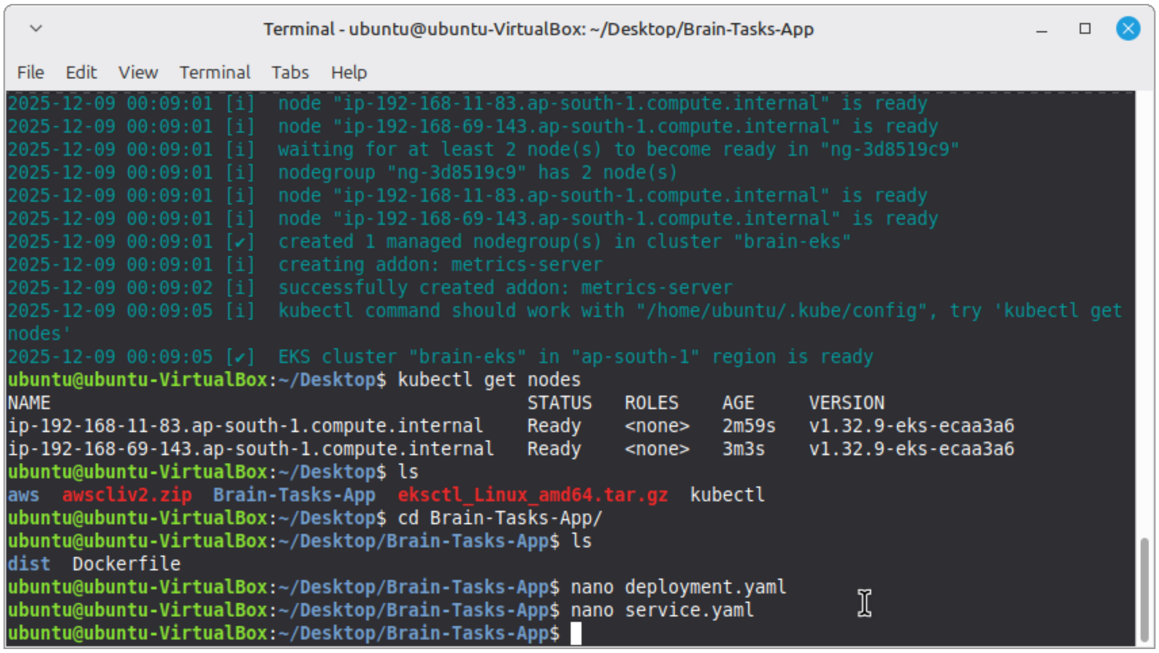
```
total 302440
drwxr-xr-x 3 ubuntu ubuntu      4096 Dec  5 19:14 aws
-rw-rw-r-- 1 ubuntu ubuntu  62989783 Dec  8 22:55 awscliv2.zip
drwxrwxr-x 4 ubuntu ubuntu      4096 Dec  8 22:56 Brain-Tasks-App
-rwxr-xr-x 1 ubuntu ubuntu 149491896 Dec  1 12:40 eksctl
-rw-rw-r-- 1 ubuntu ubuntu  36633986 Dec  8 23:52 eksctl_Linux_amd64.tar.gz
-rw-rw-r-- 1 ubuntu ubuntu  60559544 Dec  8 23:45 kubectl
ubuntu@ubuntu-VirtualBox:~/Desktop$ sudo mv eksctl /usr/local/bin/
ubuntu@ubuntu-VirtualBox:~/Desktop$ eksctl version
0.220.0
ubuntu@ubuntu-VirtualBox:~/Desktop$ eksctl create cluster \
--name brain-eks \
--region ap-south-1 \
--nodes 2 \
--node-type t3.medium
2025-12-08 23:55:29 [i]  eksctl version 0.220.0
2025-12-08 23:55:29 [i]  using region ap-south-1
2025-12-08 23:55:30 [i]  setting availability zones to [ap-south-1b ap-south-1c ap-south-1a]
2025-12-08 23:55:30 [i]  subnets for ap-south-1b - public:192.168.0.0/19 private:192.168.96.0/19
2025-12-08 23:55:30 [i]  subnets for ap-south-1c - public:192.168.32.0/19 private:192.168.128.0/19
2025-12-08 23:55:30 [i]  subnets for ap-south-1a - public:192.168.64.0/19 private:192.168.160.0/19
2025-12-08 23:55:30 [i]  nodegroup "ng-3d8519c9" will use "" [AmazonLinux2023/1.32]
2025-12-08 23:55:30 [!]  Auto Mode will be enabled by default in an upcoming release of eksctl. This mea
ns managed node groups and managed networking add-ons will no longer be created by default. To maintain
```


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop
File Edit View Terminal Tabs Help

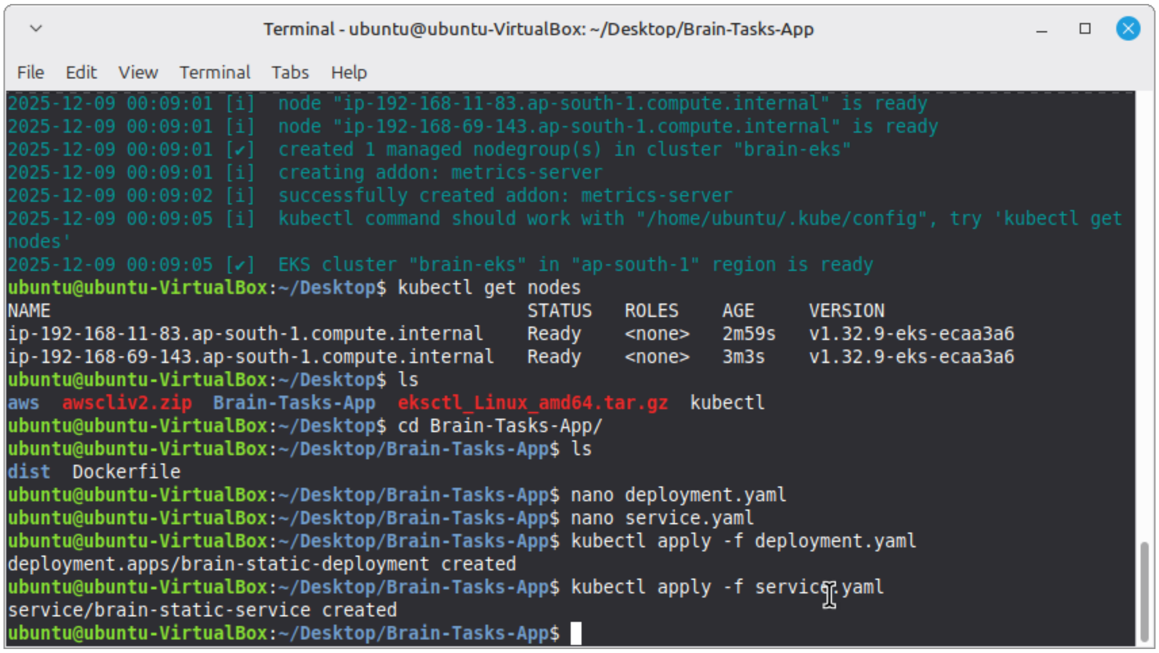
```
2025-12-08 23:55:30 [i]  if you encounter any issues, check CloudFormation console or try 'eksctl utils
describe-stacks --region=ap-south-1 --cluster=brain-eks'
2025-12-08 23:55:30 [i]  Kubernetes API endpoint access will use default of {publicAccess=true, privateA
ccess=false} for cluster "brain-eks" in "ap-south-1"
2025-12-08 23:55:30 [i]  CloudWatch logging will not be enabled for cluster "brain-eks" in "ap-south-1"
2025-12-08 23:55:30 [i]  you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPE
CIFY-YOUR-LOG-TYPES-HERE (e.g. all)} --region=ap-south-1 --cluster=brain-eks'
2025-12-08 23:55:30 [i]  default addons kube-proxy, coredns, metrics-server, vpc-cni were not specified,
 will install them as EKS addons
2025-12-08 23:55:30 [i]
2 sequential tasks: { create cluster control plane "brain-eks",
    2 sequential sub-tasks: {
        2 sequential sub-tasks: {
            1 task: { create addons },
            wait for control plane to become ready,
        },
        create managed nodegroup "ng-3d8519c9",
    }
}
2025-12-08 23:55:30 [i]  building cluster stack "eksctl-brain-eks-cluster"
2025-12-08 23:55:30 [i]  deploying stack "eksctl-brain-eks-cluster"
2025-12-08 23:56:00 [i]  waiting for CloudFormation stack "eksctl-brain-eks-cluster"
2025-12-08 23:56:30 [i]  waiting for CloudFormation stack "eksctl-brain-eks-cluster"
```


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop
File Edit View Terminal Tabs Help

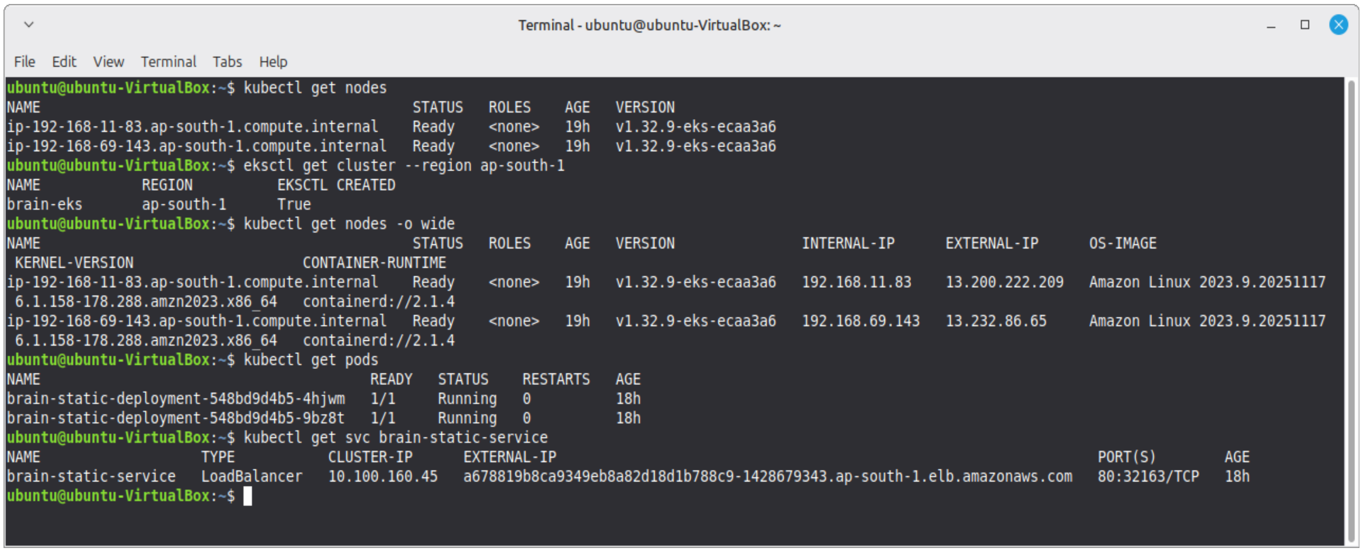
```
2025-12-09 00:06:38 [i]  building managed nodegroup stack "eksctl-brain-eks-nodegroup-ng-3d8519c9"
2025-12-09 00:06:39 [i]  deploying stack "eksctl-brain-eks-nodegroup-ng-3d8519c9"
2025-12-09 00:06:39 [i]  waiting for CloudFormation stack "eksctl-brain-eks-nodegroup-ng-3d8519c9"
2025-12-09 00:07:09 [i]  waiting for CloudFormation stack "eksctl-brain-eks-nodegroup-ng-3d8519c9"
2025-12-09 00:07:54 [i]  waiting for CloudFormation stack "eksctl-brain-eks-nodegroup-ng-3d8519c9"
2025-12-09 00:08:57 [i]  waiting for CloudFormation stack "eksctl-brain-eks-nodegroup-ng-3d8519c9"
2025-12-09 00:08:57 [i]  waiting for the control plane to become ready
2025-12-09 00:09:01 [✔]  saved kubeconfig as "/home/ubuntu/.kube/config"
2025-12-09 00:09:01 [i]  no tasks
2025-12-09 00:09:01 [✔]  all EKS cluster resources for "brain-eks" have been created
2025-12-09 00:09:01 [i]  nodegroup "ng-3d8519c9" has 2 node(s)
2025-12-09 00:09:01 [i]  node "ip-192-168-11-83.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [i]  node "ip-192-168-69-143.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [i]  waiting for at least 2 node(s) to become ready in "ng-3d8519c9"
2025-12-09 00:09:01 [i]  nodegroup "ng-3d8519c9" has 2 node(s)
2025-12-09 00:09:01 [i]  node "ip-192-168-11-83.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [i]  node "ip-192-168-69-143.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [✔]  created 1 managed nodegroup(s) in cluster "brain-eks"
2025-12-09 00:09:01 [i]  creating addon: metrics-server
2025-12-09 00:09:02 [i]  successfully created addon: metrics-server
2025-12-09 00:09:05 [i]  kubectl command should work with "/home/ubuntu/.kube/config", try 'kubectl get
nodes'
2025-12-09 00:09:05 [✔]  EKS cluster "brain-eks" in "ap-south-1" region is ready
ubuntu@ubuntu-VirtualBox:~/Desktop$
```


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop/Brain-Tasks-App
File Edit View Terminal Tabs Help

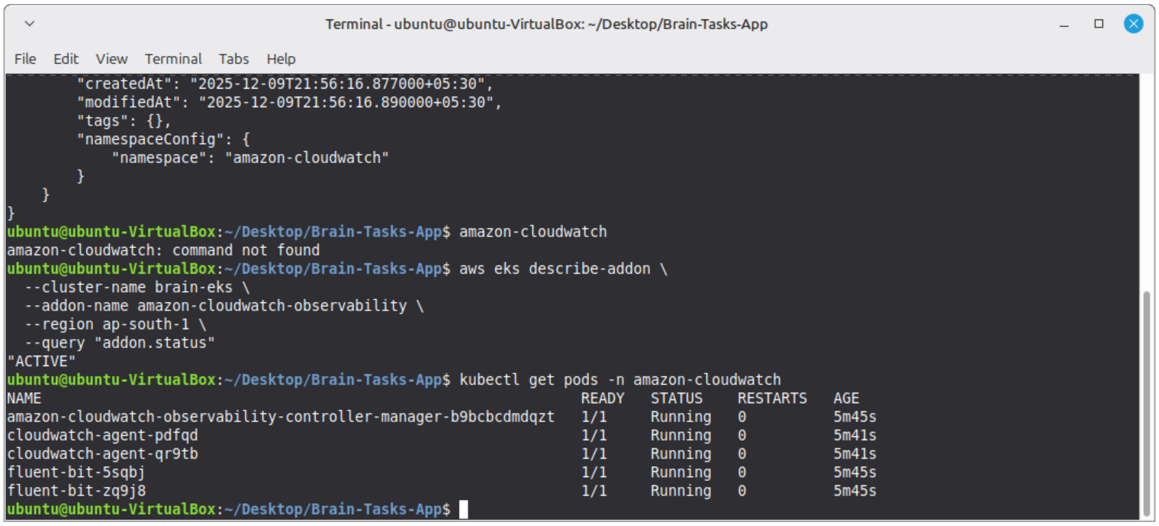
```
2025-12-09 00:09:01 [i]  node "ip-192-168-11-83.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [i]  node "ip-192-168-69-143.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [i]  waiting for at least 2 node(s) to become ready in "ng-3d8519c9"
2025-12-09 00:09:01 [i]  nodegroup "ng-3d8519c9" has 2 node(s)
2025-12-09 00:09:01 [i]  node "ip-192-168-11-83.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [i]  node "ip-192-168-69-143.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [✔]  created 1 managed nodegroup(s) in cluster "brain-eks"
2025-12-09 00:09:01 [i]  creating addon: metrics-server
2025-12-09 00:09:02 [i]  successfully created addon: metrics-server
2025-12-09 00:09:05 [i]  kubectl command should work with "/home/ubuntu/.kube/config", try 'kubectl get
nodes'
2025-12-09 00:09:05 [✔]  EKS cluster "brain-eks" in "ap-south-1" region is ready
ubuntu@ubuntu-VirtualBox:~/Desktop$ kubectl get nodes
NAME                                            STATUS   ROLES    AGE     VERSION
ip-192-168-11-83.ap-south-1.compute.internal    Ready    <none>   2m59s   v1.32.9-eks-ecaa3a6
ip-192-168-69-143.ap-south-1.compute.internal   Ready    <none>   3m3s    v1.32.9-eks-ecaa3a6
ubuntu@ubuntu-VirtualBox:~/Desktop$ ls
aws  awscliv2.zip  Brain-Tasks-App  eksctl_Linux_amd64.tar.gz  kubectl
ubuntu@ubuntu-VirtualBox:~/Desktop$ cd Brain-Tasks-App/
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ ls
dist  Dockerfile
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ nano deployment.yaml
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ nano service.yaml
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$
```


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop/Brain-Tasks-App
File Edit View Terminal Tabs Help

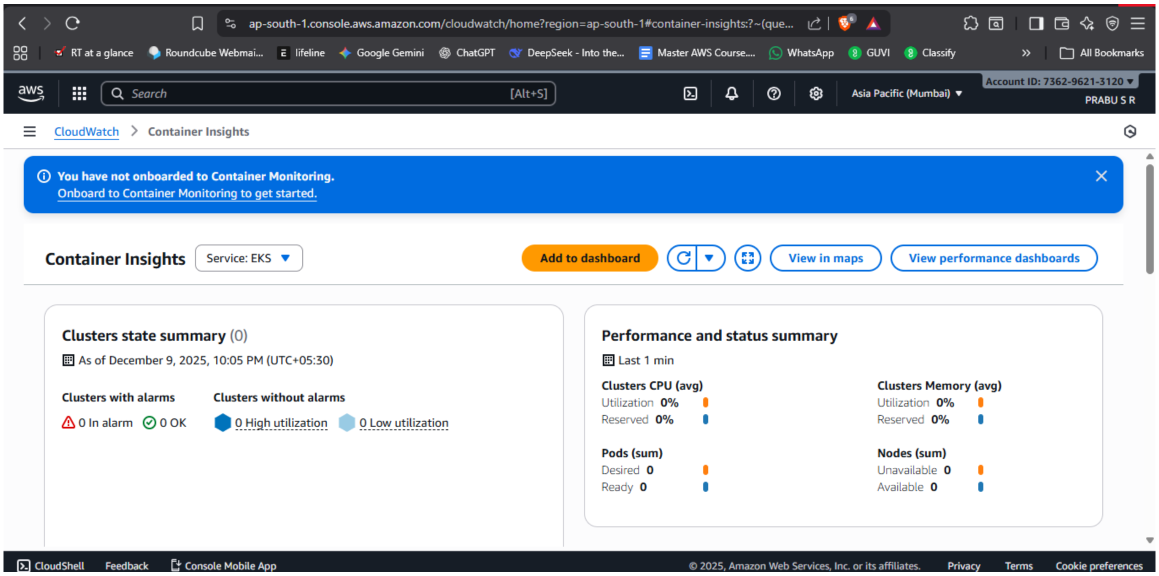
```
2025-12-09 00:09:01 [i]  node "ip-192-168-11-83.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [i]  node "ip-192-168-69-143.ap-south-1.compute.internal" is ready
2025-12-09 00:09:01 [✔]  created 1 managed nodegroup(s) in cluster "brain-eks"
2025-12-09 00:09:01 [i]  creating addon: metrics-server
2025-12-09 00:09:02 [i]  successfully created addon: metrics-server
2025-12-09 00:09:05 [i]  kubectl command should work with "/home/ubuntu/.kube/config", try 'kubectl get
nodes'
2025-12-09 00:09:05 [✔]  EKS cluster "brain-eks" in "ap-south-1" region is ready
ubuntu@ubuntu-VirtualBox:~/Desktop$ kubectl get nodes
NAME                                            STATUS   ROLES    AGE     VERSION
ip-192-168-11-83.ap-south-1.compute.internal    Ready    <none>   2m59s   v1.32.9-eks-ecaa3a6
ip-192-168-69-143.ap-south-1.compute.internal   Ready    <none>   3m3s    v1.32.9-eks-ecaa3a6
ubuntu@ubuntu-VirtualBox:~/Desktop$ ls
aws  awscliv2.zip  Brain-Tasks-App  eksctl_Linux_amd64.tar.gz  kubectl
ubuntu@ubuntu-VirtualBox:~/Desktop$ cd Brain-Tasks-App/
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ ls
dist  Dockerfile
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ nano deployment.yaml
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ nano service.yaml
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ kubectl apply -f deployment.yaml
deployment.apps/brain-static-deployment created
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ kubectl apply -f service.yaml
service/brain-static-service created
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$
```


Terminal - ubuntu@ubuntu-VirtualBox: ~
File Edit View Terminal Tabs Help

```
ubuntu@ubuntu-VirtualBox:~$ kubectl get nodes
NAME                                            STATUS   ROLES    AGE   VERSION
ip-192-168-11-83.ap-south-1.compute.internal    Ready    <none>   19h   v1.32.9-eks-ecaa3a6
ip-192-168-69-143.ap-south-1.compute.internal   Ready    <none>   19h   v1.32.9-eks-ecaa3a6
ubuntu@ubuntu-VirtualBox:~$ eksctl get cluster --region ap-south-1
NAME            REGION          EKSCTL CREATED
brain-eks       ap-south-1      True
ubuntu@ubuntu-VirtualBox:~$ kubectl get nodes -o wide
NAME                                            STATUS   ROLES    AGE   VERSION               INTERNAL-IP      EXTERNAL-IP     OS-IMAGE
 KERNEL-VERSION                    CONTAINER-RUNTIME
ip-192-168-11-83.ap-south-1.compute.internal    Ready    <none>   19h   v1.32.9-eks-ecaa3a6   192.168.11.83    13.200.222.209   Amazon Linux 2023.9.20251117
 6.1.158-178.288.amzn2023.x86_64   containerd://2.1.4
ip-192-168-69-143.ap-south-1.compute.internal   Ready    <none>   19h   v1.32.9-eks-ecaa3a6   192.168.69.143   13.232.86.65     Amazon Linux 2023.9.20251117
 6.1.158-178.288.amzn2023.x86_64   containerd://2.1.4
ubuntu@ubuntu-VirtualBox:~$ kubectl get pods
NAME                                       READY   STATUS    RESTARTS   AGE
brain-static-deployment-548bd9d4b5-4hjwm   1/1     Running   0          18h
brain-static-deployment-548bd9d4b5-9bz8t   1/1     Running   0          18h
ubuntu@ubuntu-VirtualBox:~$ kubectl get svc brain-static-service
NAME                   TYPE           CLUSTER-IP     EXTERNAL-IP                                                                PORT(S)        AGE
brain-static-service   LoadBalancer   10.100.160.45   a678819b8ca9349eb8a82d18d1b788c9-1428679343.ap-south-1.elb.amazonaws.com   80:32163/TCP   18h
ubuntu@ubuntu-VirtualBox:~$
```


Terminal - ubuntu@ubuntu-VirtualBox: ~/Desktop/Brain-Tasks-App
File Edit View Terminal Tabs Help

```
        "createdAt": "2025-12-09T21:56:16.877000+05:30",
        "modifiedAt": "2025-12-09T21:56:16.890000+05:30",
        "tags": {},
        "namespaceConfig": {
            "namespace": "amazon-cloudwatch"
        }
    }
}
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ amazon-cloudwatch
amazon-cloudwatch: command not found
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ aws eks describe-addon \
  --cluster-name brain-eks \
  --addon-name amazon-cloudwatch-observability \
  --region ap-south-1 \
  --query "addon.status"
"ACTIVE"
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$ kubectl get pods -n amazon-cloudwatch
NAME                                                              READY   STATUS    RESTARTS   AGE
amazon-cloudwatch-observability-controller-manager-b9bcbcdmdqzt   1/1     Running   0          5m45s
cloudwatch-agent-pdfqd                                            1/1     Running   0          5m41s
cloudwatch-agent-qr9tb                                            1/1     Running   0          5m41s
fluent-bit-5sqbj                                                  1/1     Running   0          5m45s
fluent-bit-zq9j8                                                  1/1     Running   0          5m45s
ubuntu@ubuntu-VirtualBox:~/Desktop/Brain-Tasks-App$
```



You have not onboarded to Container Monitoring.
Onboard to Container Monitoring to get started.

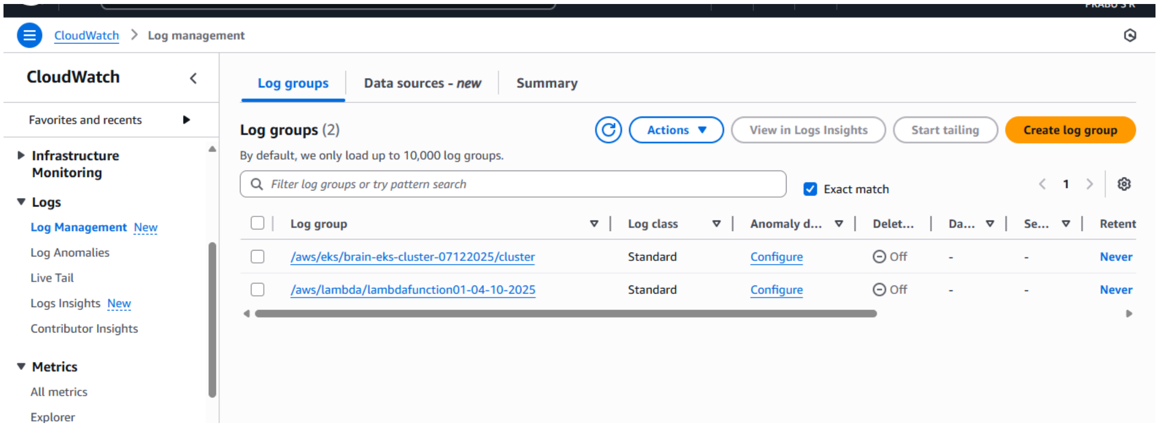
Container Insights Service: EKS

Clusters state summary (0)
As of December 9, 2025, 10:05 PM (UTC+05:30)

Clusters with alarms: 0 In alarm, 0 OK
Clusters without alarms: 0 High utilization, 0 Low utilization

Performance and status summary
Last 1 min

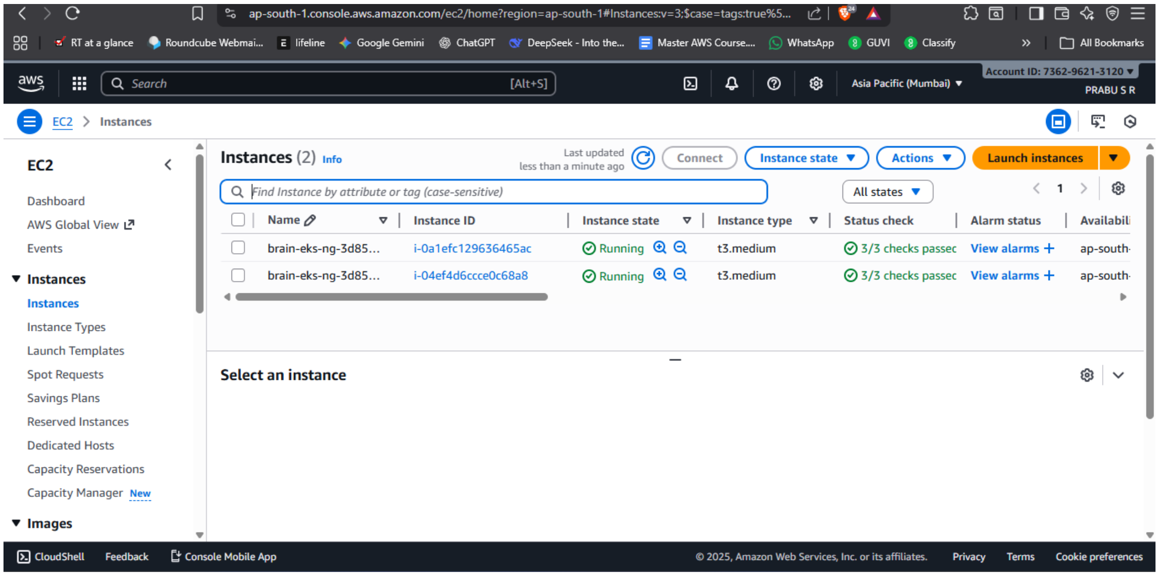
Clusters CPU (avg) Utilization 0% Reserved 0%	Clusters Memory (avg) Utilization 0% Reserved 0%
Pods (sum) Desired 0 Ready 0	Nodes (sum) Unavailable 0 Available 0



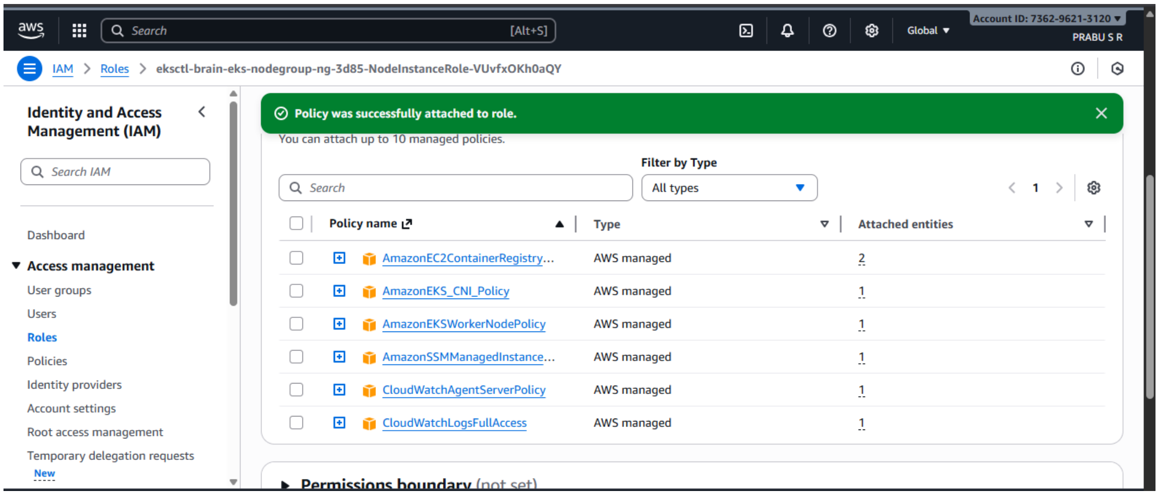
Log groups (2)
By default, we only load up to 10,000 log groups.

Filter log groups or try pattern search ☑ Exact match

□	Log group	Log class	Anomaly d...	Delet...	Da...	Se...	Retent
□	/aws/eks/brain-eks-cluster-07122025/cluster	Standard	Configure	Off	-	-	Never
□	/aws/lambda/lambdafunction01-04-10-2025	Standard	Configure	Off	-	-	Never



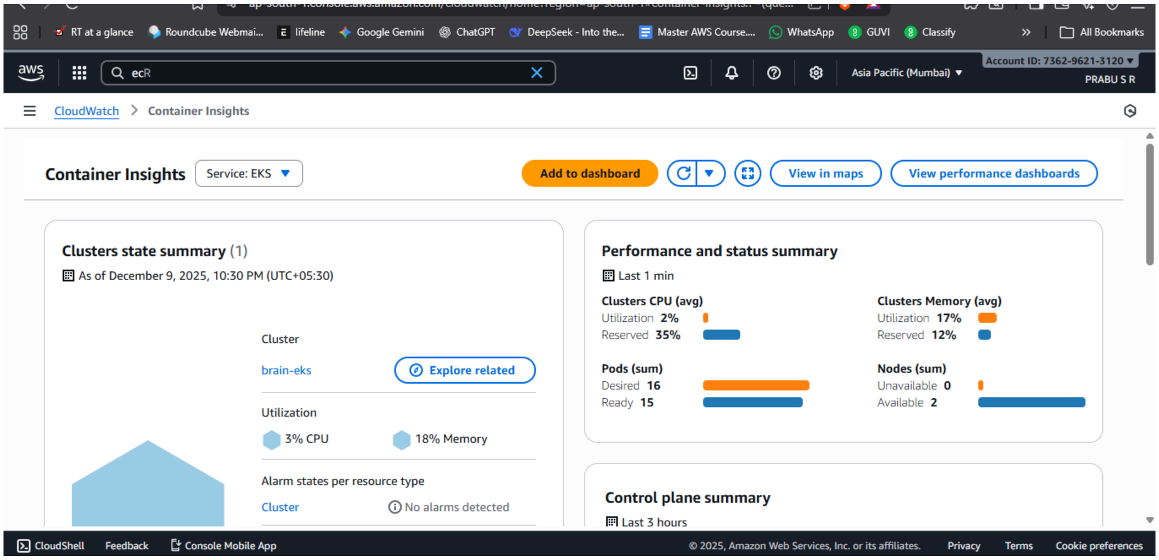
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availabil
brain-eks-ng-3d85...	i-0a1efc129636465ac	Running	t3.medium	3/3 checks passec	View alarms +	ap-south-
brain-eks-ng-3d85...	i-04ef4d6ccce0c68a8	Running	t3.medium	3/3 checks passec	View alarms +	ap-south-


Search [Alt+S]
Global
Account ID: 7362-9621-3120
PRABU S R
IAM > Roles > eksctl-brain-eks-nodegroup-ng-3d85-NodeInstanceRole-VUvfxOKh0aQY
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Root access management
Temporary delegation requests New
Filter by Type
Search
All types
1
Permissions boundary (not set)

Policy was successfully attached to role.

You can attach up to 10 managed policies.

Policy name	Type	Attached entities
AmazonEC2ContainerRegistry...	AWS managed	2
AmazonEKS_CNI_Policy	AWS managed	1
AmazonEKSWorkerNodePolicy	AWS managed	1
AmazonSSMManagedInstance...	AWS managed	1
CloudWatchAgentServerPolicy	AWS managed	1
CloudWatchLogsFullAccess	AWS managed	1



Clusters state summary (1)

As of December 9, 2025, 10:30 PM (UTC+05:30)

Cluster: brain-eks

Utilization: 3% CPU, 18% Memory

Alarm states per resource type: Cluster — No alarms detected

Performance and status summary

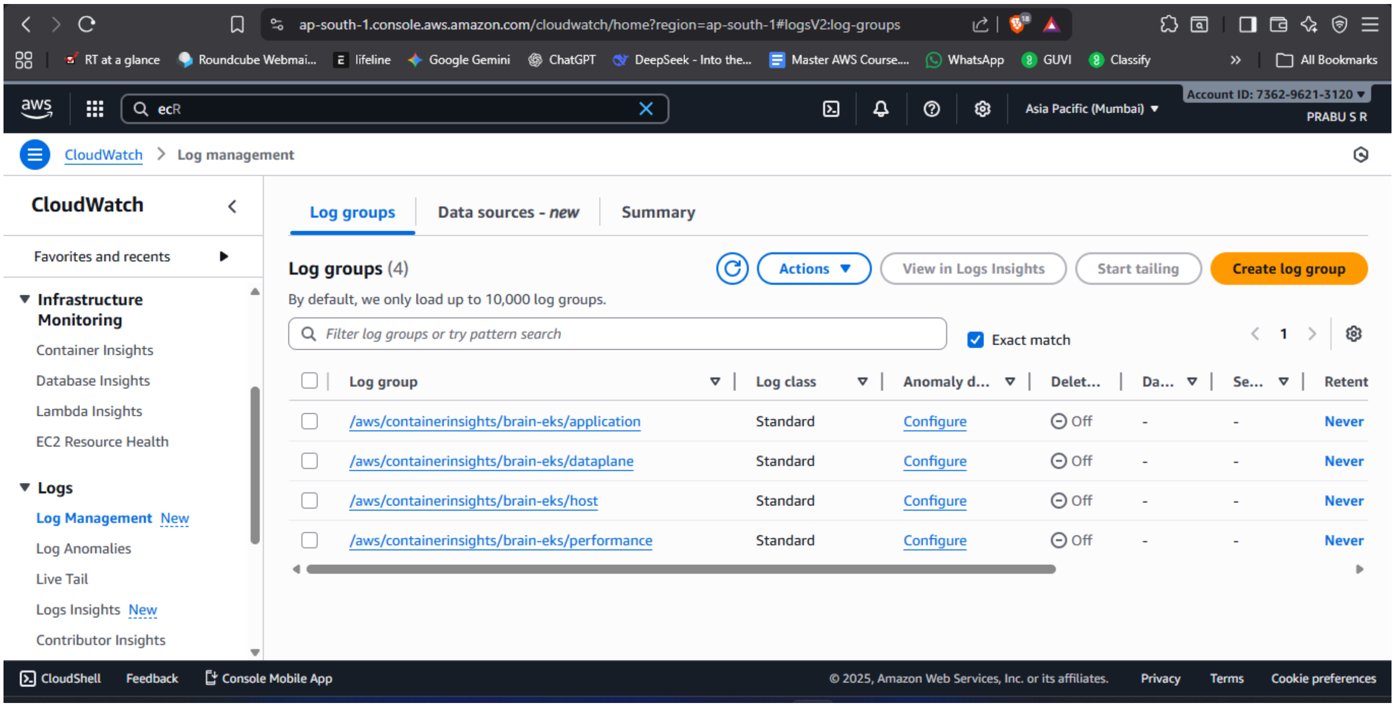
Last 1 min

Clusters CPU (avg): Utilization 2%, Reserved 35%

Clusters Memory (avg): Utilization 17%, Reserved 12%

Pods (sum): Desired 16, Ready 15

Nodes (sum): Unavailable 0, Available 2



ap-south-1.console.aws.amazon.com/cloudwatch/home?region=ap-south-1#logsV2:log-groups

RT at a glance | Roundcube Webmai... | lifeline | Google Gemini | ChatGPT | DeepSeek - Into the... | Master AWS Course... | WhatsApp | GUVI | Classify | All Bookmarks

ecR

Asia Pacific (Mumbai) ▼ | Account ID: 7362-9621-3120 ▼ | PRABU S R

CloudWatch > Log management

CloudWatch

Favorites and recents ▶

▼ Infrastructure Monitoring
- Container Insights
- Database Insights
- Lambda Insights
- EC2 Resource Health

▼ Logs
- Log Management New
- Log Anomalies
- Live Tail
- Logs Insights New
- Contributor Insights

Log groups | Data sources - *new* | Summary

Log groups (4)

Actions ▼ | View in Logs Insights | Start tailing | Create log group

By default, we only load up to 10,000 log groups.

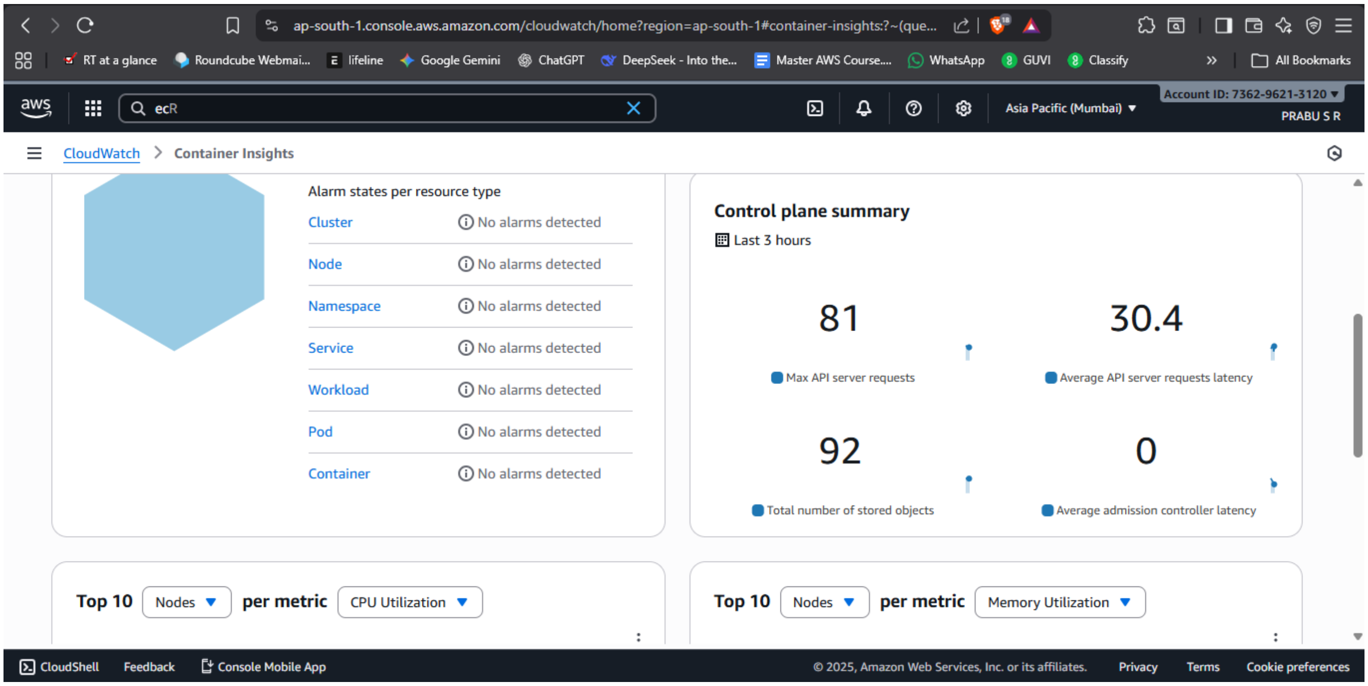
Filter log groups or try pattern search

☑ Exact match

< 1 >

□	Log group	Log class	Anomaly d...	Delet...	Da...	Se...	Retent
□	/aws/containerinsights/brain-eks/application	Standard	Configure	⊖ Off	-	-	Never
□	/aws/containerinsights/brain-eks/dataplane	Standard	Configure	⊖ Off	-	-	Never
□	/aws/containerinsights/brain-eks/host	Standard	Configure	⊖ Off	-	-	Never
□	/aws/containerinsights/brain-eks/performance	Standard	Configure	⊖ Off	-	-	Never

CloudShell | Feedback | Console Mobile App | © 2025, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences



ap-south-1.console.aws.amazon.com/cloudwatch/home?region=ap-south-1#container-insights:?~(que...

RT at a glance | Roundcube Webmai... | lifeline | Google Gemini | ChatGPT | DeepSeek - Into the... | Master AWS Course... | WhatsApp | GUVI | Classify | All Bookmarks

ecR

Asia Pacific (Mumbai) ▼ | Account ID: 7362-9621-3120 ▼ | PRABU S R

CloudWatch > Container Insights

Alarm states per resource type

Alarm states per resource type	
Cluster	ⓘ No alarms detected
Node	ⓘ No alarms detected
Namespace	ⓘ No alarms detected
Service	ⓘ No alarms detected
Workload	ⓘ No alarms detected
Pod	ⓘ No alarms detected
Container	ⓘ No alarms detected

Control plane summary

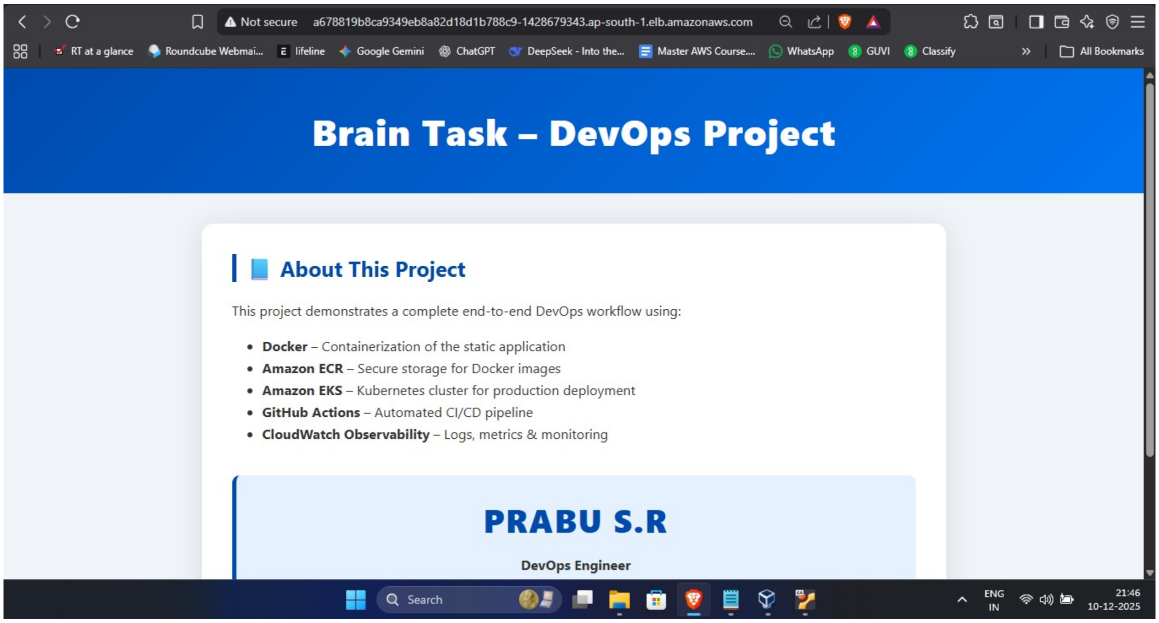
Last 3 hours

Value	Metric
81	Max API server requests
30.4	Average API server requests latency
92	Total number of stored objects
0	Average admission controller latency

Top 10 Nodes ▼ **per metric** CPU Utilization ▼

Top 10 Nodes ▼ **per metric** Memory Utilization ▼

CloudShell | Feedback | Console Mobile App | © 2025, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences



Brain Task – DevOps Project

📘 About This Project

This project demonstrates a complete end-to-end DevOps workflow using:

- **Docker** – Containerization of the static application
- **Amazon ECR** – Secure storage for Docker images
- **Amazon EKS** – Kubernetes cluster for production deployment
- **GitHub Actions** – Automated CI/CD pipeline
- **CloudWatch Observability** – Logs, metrics & monitoring

PRABU S.R

DevOps Engineer

Project Conclusion:

This project successfully demonstrates a complete end-to-end DevOps pipeline for deploying a containerized static web application on Amazon EKS using a fully automated CI/CD workflow. Starting from local development, the application was Dockerized and integrated with GitHub Actions, enabling seamless and repeatable builds. Upon each code change, the pipeline automatically builds a new Docker image, pushes it to Amazon ECR, and updates the deployment running in the EKS cluster.

The system implements core DevOps principles, including automation, scalability, observability, and container orchestration. Kubernetes ensures reliable workload distribution through replica sets and provides high availability using the LoadBalancer service. With Amazon CloudWatch integrated, logs and performance metrics from the cluster are continuously monitored, offering insights into application health and cluster performance.

By completing this project, the full lifecycle of modern cloud-native application deployment is demonstrated—from coding, containerization, CI/CD automation, and image versioning, to Kubernetes orchestration and monitoring. The final setup ensures that any new code pushed to GitHub results in an automated, zero-downtime deployment, allowing end users to always access the latest version of the application.

Overall, the project achieves a production-ready deployment workflow and reflects real-world DevOps practices widely used in the industry today.