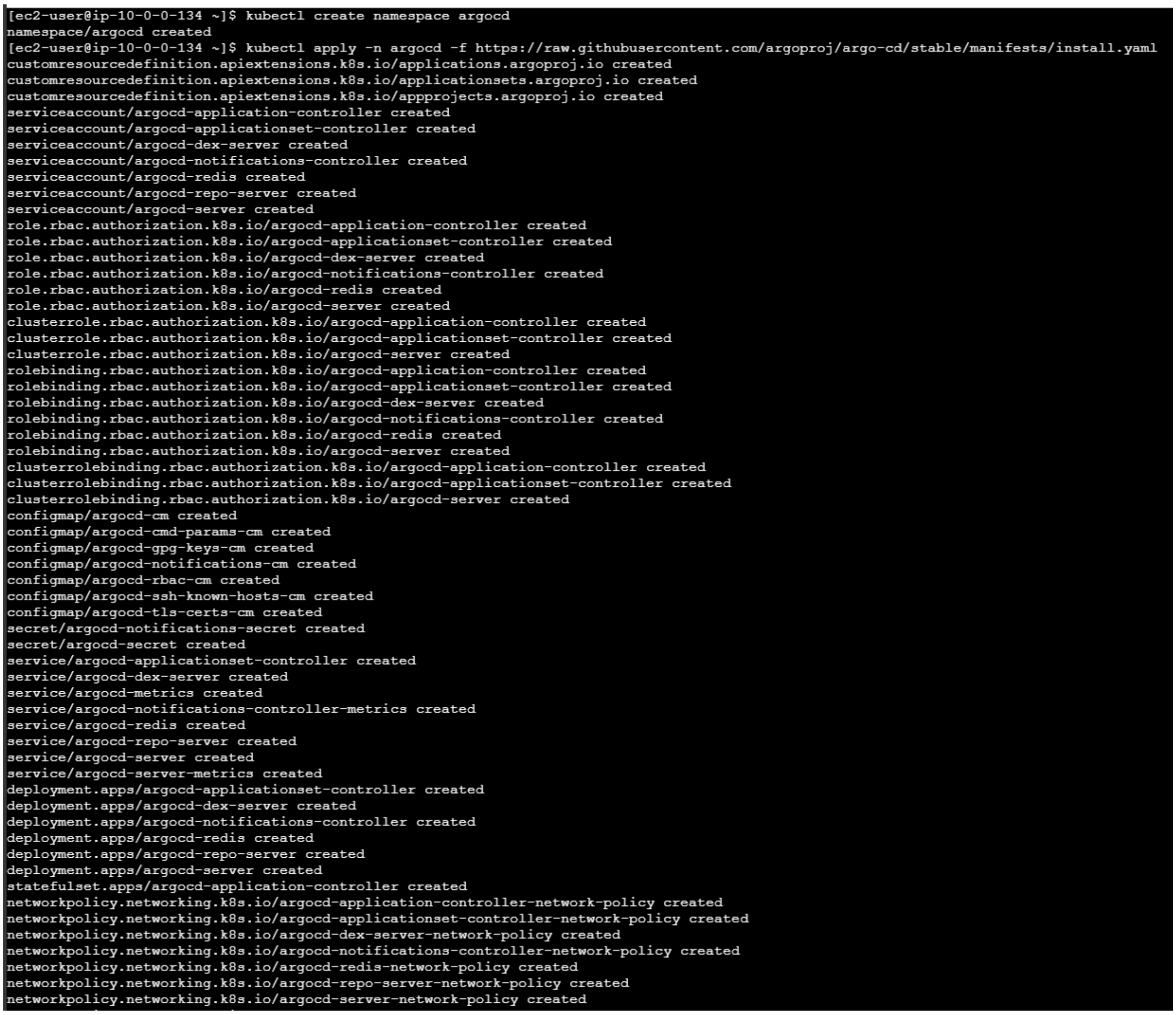
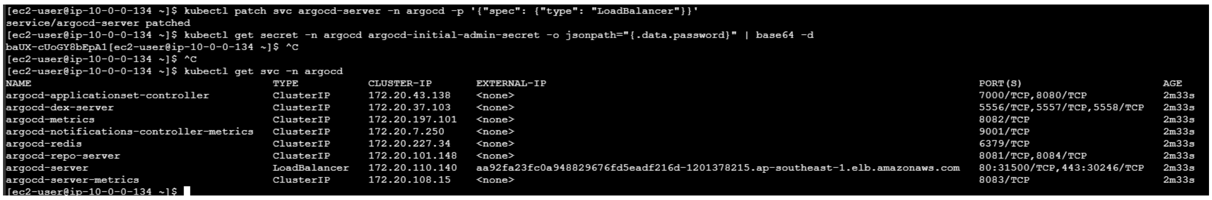
2. Reuse Jenkins CI and use GitOps for the CD pipeline

Install ArgoCD



```
[ec2-user@ip-10-0-0-134 ~]$ kubectl create namespace argocd
namespace/argocd created
[ec2-user@ip-10-0-0-134 ~]$ kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml
customresourcedefinition.apiextensions.k8s.io/applications.argoproj.io created
customresourcedefinition.apiextensions.k8s.io/applicationsets.argoproj.io created
customresourcedefinition.apiextensions.k8s.io/appprojects.argoproj.io created
serviceaccount/argocd-application-controller created
serviceaccount/argocd-applicationset-controller created
serviceaccount/argocd-dex-server created
serviceaccount/argocd-notifications-controller created
serviceaccount/argocd-redis created
serviceaccount/argocd-repo-server created
serviceaccount/argocd-server created
role.rbac.authorization.k8s.io/argocd-application-controller created
role.rbac.authorization.k8s.io/argocd-applicationset-controller created
role.rbac.authorization.k8s.io/argocd-dex-server created
role.rbac.authorization.k8s.io/argocd-notifications-controller created
role.rbac.authorization.k8s.io/argocd-redis created
role.rbac.authorization.k8s.io/argocd-server created
clusterrole.rbac.authorization.k8s.io/argocd-application-controller created
clusterrole.rbac.authorization.k8s.io/argocd-applicationset-controller created
clusterrole.rbac.authorization.k8s.io/argocd-server created
rolebinding.rbac.authorization.k8s.io/argocd-application-controller created
rolebinding.rbac.authorization.k8s.io/argocd-applicationset-controller created
rolebinding.rbac.authorization.k8s.io/argocd-dex-server created
rolebinding.rbac.authorization.k8s.io/argocd-notifications-controller created
rolebinding.rbac.authorization.k8s.io/argocd-redis created
rolebinding.rbac.authorization.k8s.io/argocd-server created
clusterrolebinding.rbac.authorization.k8s.io/argocd-application-controller created
clusterrolebinding.rbac.authorization.k8s.io/argocd-applicationset-controller created
clusterrolebinding.rbac.authorization.k8s.io/argocd-server created
configmap/argocd-cm created
configmap/argocd-cmd-params-cm created
configmap/argocd-gpg-keys-cm created
configmap/argocd-notifications-cm created
configmap/argocd-rbac-cm created
configmap/argocd-ssh-known-hosts-cm created
configmap/argocd-tls-certs-cm created
secret/argocd-notifications-secret created
secret/argocd-secret created
service/argocd-applicationset-controller created
service/argocd-dex-server created
service/argocd-metrics created
service/argocd-notifications-controller-metrics created
service/argocd-redis created
service/argocd-repo-server created
service/argocd-server created
service/argocd-server-metrics created
deployment.apps/argocd-applicationset-controller created
deployment.apps/argocd-dex-server created
deployment.apps/argocd-notifications-controller created
deployment.apps/argocd-redis created
deployment.apps/argocd-repo-server created
deployment.apps/argocd-server created
statefulset.apps/argocd-application-controller created
networkpolicy.networking.k8s.io/argocd-application-controller-network-policy created
networkpolicy.networking.k8s.io/argocd-applicationset-controller-network-policy created
networkpolicy.networking.k8s.io/argocd-dex-server-network-policy created
networkpolicy.networking.k8s.io/argocd-notifications-controller-network-policy created
networkpolicy.networking.k8s.io/argocd-redis-network-policy created
networkpolicy.networking.k8s.io/argocd-repo-server-network-policy created
networkpolicy.networking.k8s.io/argocd-server-network-policy created
```



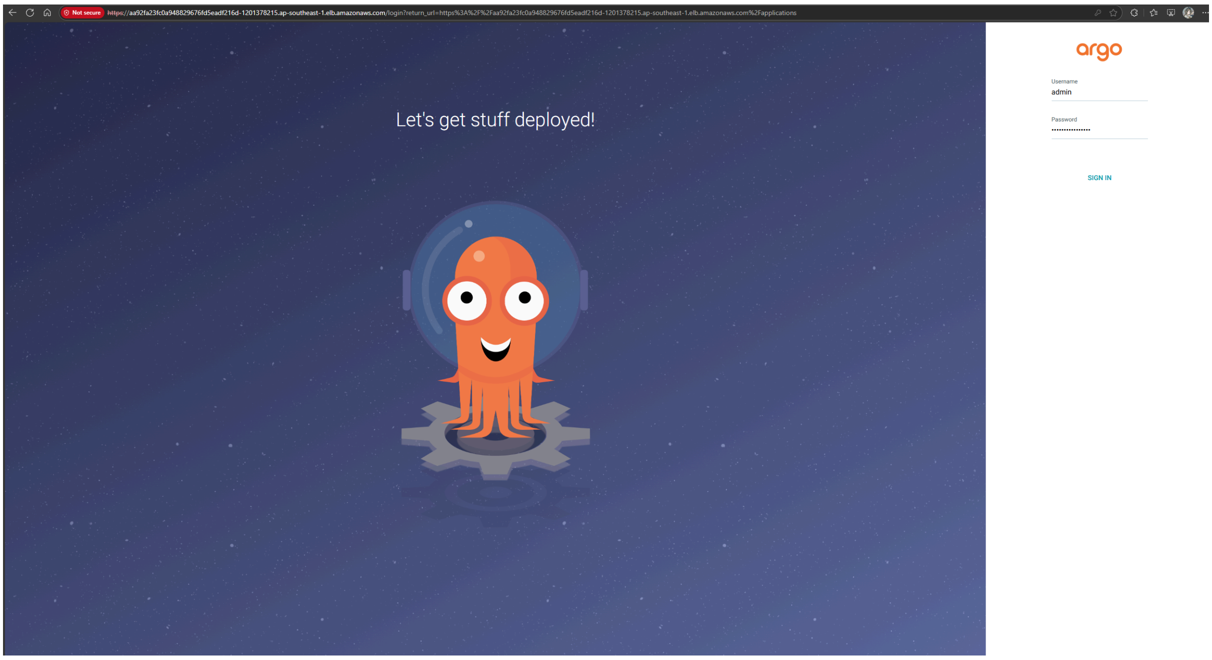
```
[ec2-user@ip-10-0-0-134 ~]$ kubectl patch svc argocd-server -n argocd -p '{"spec": {"type": "LoadBalancer"}}'
service/argocd-server patched
[ec2-user@ip-10-0-0-134 ~]$ kubectl get secret -n argocd argocd-initial-admin-secret -o jsonpath="{.data.password}" | base64 -d
baUX-cUoGY8bEpA1[ec2-user@ip-10-0-0-134 ~]$ ^C
[ec2-user@ip-10-0-0-134 ~]$ ^C
[ec2-user@ip-10-0-0-134 ~]$ kubectl get svc -n argocd
NAME                                      TYPE           CLUSTER-IP       EXTERNAL-IP                                                                  PORT(S)                      AGE
argocd-applicationset-controller          ClusterIP      172.20.43.138    <none>                                                                       7000/TCP,8080/TCP            2m33s
argocd-dex-server                         ClusterIP      172.20.37.103    <none>                                                                       5556/TCP,5557/TCP,5558/TCP   2m33s
argocd-metrics                            ClusterIP      172.20.197.101   <none>                                                                       8082/TCP                     2m33s
argocd-notifications-controller-metrics   ClusterIP      172.20.7.250     <none>                                                                       9001/TCP                     2m33s
argocd-redis                              ClusterIP      172.20.227.34    <none>                                                                       6379/TCP                     2m33s
argocd-repo-server                        ClusterIP      172.20.101.148   <none>                                                                       8081/TCP,8084/TCP            2m33s
argocd-server                             LoadBalancer   172.20.110.140   aa92fa23fc0a948829676fd5eadf216d-1201378215.ap-southeast-1.elb.amazonaws.com   80:31500/TCP,443:30246/TCP   2m33s
argocd-server-metrics                     ClusterIP      172.20.108.15    <none>                                                                       8083/TCP                     2m33s
[ec2-user@ip-10-0-0-134 ~]$ 
```

https://aa92fa23fc0a948829676fd5eadf216d-1201378215.ap-southeast-1.elb.amazonaws.com/

username: admin

password: baUX-cUoGY8bEpA1

(Get from `kubectl get secret -n argocd argocd-initial-admin-secret -o jsonpath="{.data.password}" | base64 -d`)

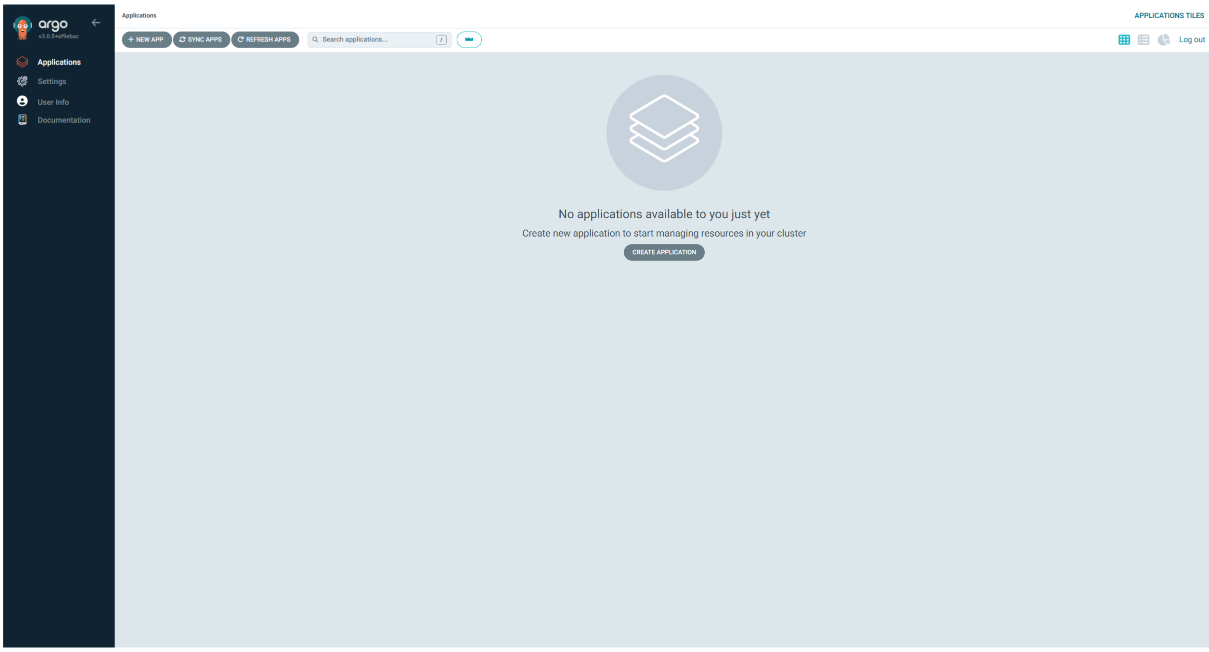


https://aa92fa23fc0a948829676fd5eadf216d-1201378215.ap-southeast-1.elb.amazonaws.com/login?return_url=https%3A%2F%2Faa92fa23fc0a948829676fd5eadf216d-1201378215.ap-southeast-1.elb.amazonaws.com%2Fapplications

Let's get stuff deployed!

Username: admin

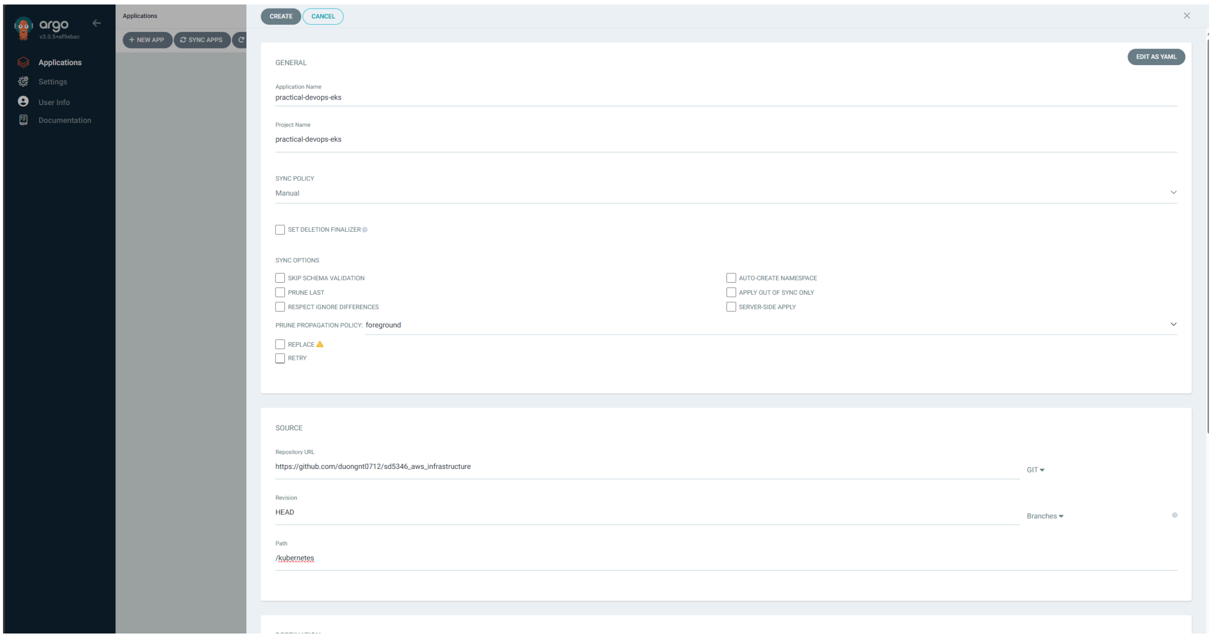
Password: ················



No applications available to you just yet

Create new application to start managing resources in your cluster

Create application



GENERAL

Application Name
practical-devops-eks

Project Name
practical-devops-eks

SYNC POLICY
Manual

□ SET DELETION FINALIZER

SYNC OPTIONS

□ SKIP SCHEMA VALIDATION
□ PRUNE LAST
□ RESPECT IGNORE DIFFERENCES
□ AUTO-CREATE NAMESPACE
□ APPLY OUT OF SYNC ONLY
□ SERVER-SIDE APPLY

PRUNE PROPAGATION POLICY: foreground

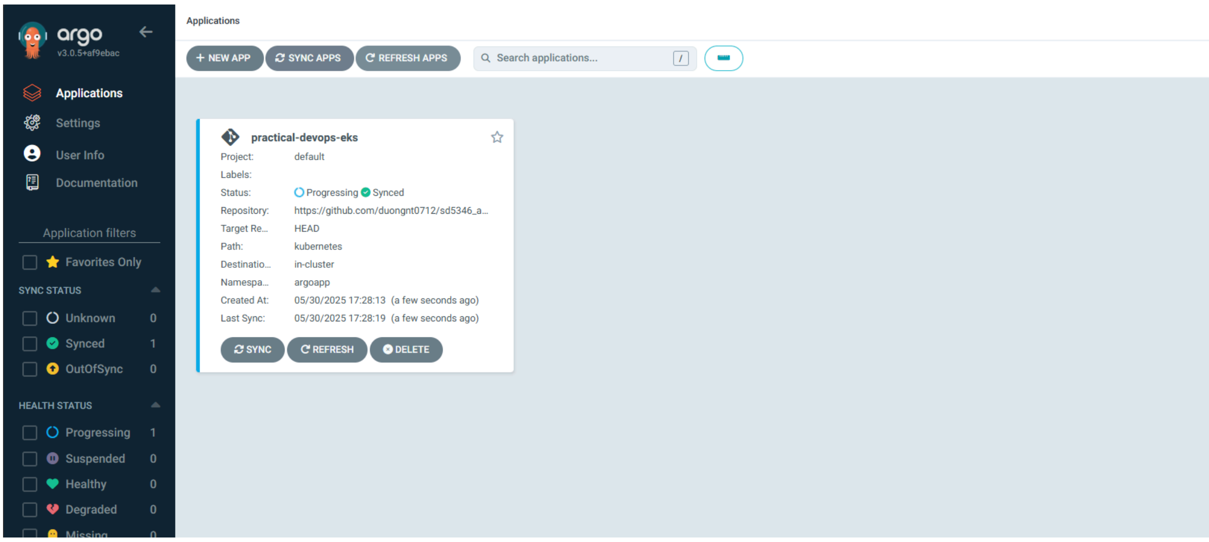
□ REPLACE ⚠
□ RETRY

SOURCE

Repository URL
https://github.com/duongnt0712/sd5346_aws_infrastructure
GIT ▾

Revision
HEAD
Branches ▾

Path
/kubernetes



□ ★ Favorites Only

SYNC STATUS

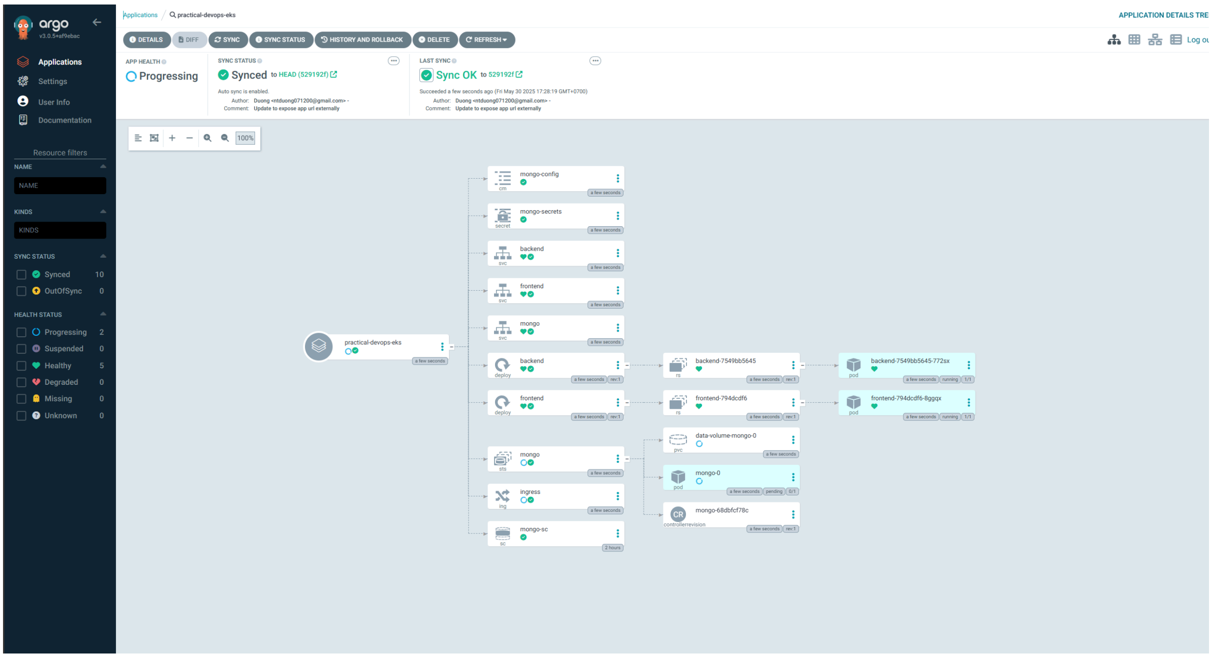
□	○ Unknown	0
□	✓ Synced	1
□	OutOfSync	0

HEALTH STATUS

□	○ Progressing	1
□	Suspended	0
□	♥ Healthy	0
□	Degraded	0
□	Missing	0

practical-devops-eks

Project:	default
Labels:	
Status:	○ Progressing ✓ Synced
Repository:	https://github.com/duongnt0712/sd5346_a...
Target Re...	HEAD
Path:	kubernetes
Destinatio...	in-cluster
Namespa...	argoapp
Created At:	05/30/2025 17:28:13 (a few seconds ago)
Last Sync:	05/30/2025 17:28:19 (a few seconds ago)



Applications / practical-devops-eks

APP HEALTH: Progressing

SYNC STATUS: Synced to HEAD (529192f)

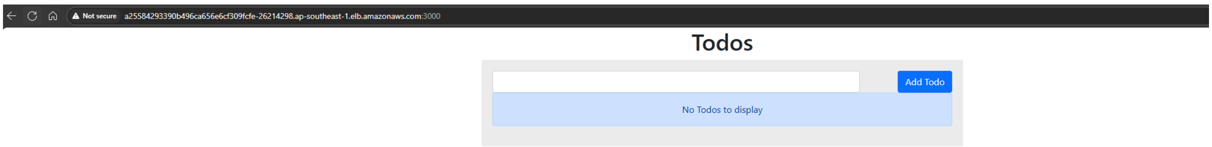
Auto sync is enabled.

Author: Duong <ntduong071200@gmail.com> -
Comment: Update to expose app url externally

LAST SYNC: Sync OK to 529192f

Succeeded a few seconds ago (Fri May 30 2025 17:28:19 GMT+0700)

Author: Duong <ntduong071200@gmail.com> -
Comment: Update to expose app url externally



Not secure a25584293390b496ca656e6cf309fcfe-26214298.ap-southeast-1.elb.amazonaws.com:3000

Todos

No Todos to display