Project Documentation

Description:

Create WordPress Site with A MySQL Database Then Using Kubernetes To Deploy On AWS. This project focuses on building a dynamic, containerized WordPress website connected to a MySQL database, and deploying the entire stack using Kubernetes on Amazon Web Services (AWS). The aim is to leverage containerization and orchestration to achieve scalability, high availability, and efficient management of web application resources in the cloud. The project demonstrates the use of Docker, Kubernetes, and AWS services (such as EKS, S3, IAM, and Elastic Load Balancing) to manage the lifecycle, networking, and persistence of the application components.

Project Title:

Using Kubernetes to Deploy online-shop website using WordPress website with MySQL Database on AWS.

Group Members:

1. Abeer Mahrous.

2. Nayra Ahmed.

3. Mina Issac.

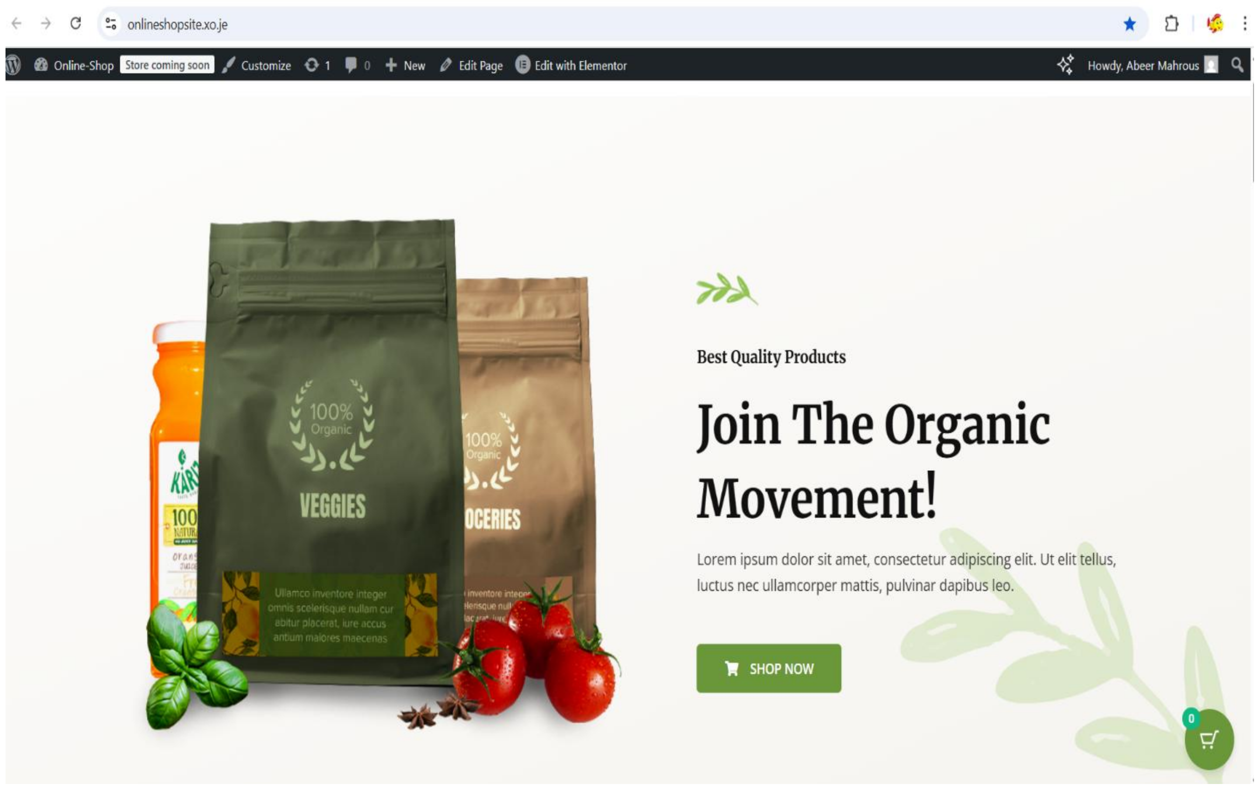
4. Omar Ashraf.

5. Mahmoud Ashraf.

6. Mohamed Mohamed.

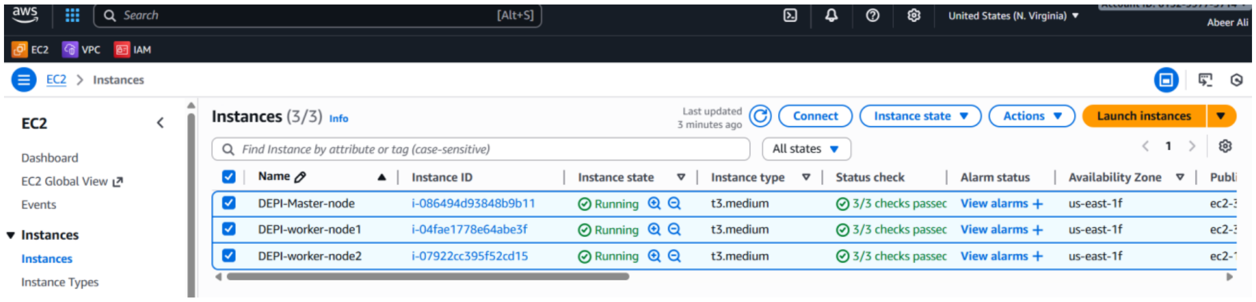
Create Online shop using WordPress:

Link : https://onlineshopsite.xo.je/



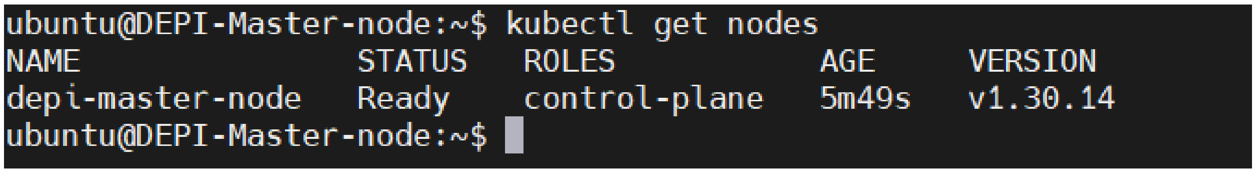
Create Kubeadm cluster using 3 EC2 instance :

1. Master EC2 .(Private IP : 172.31.72.83)
2. Node1 EC2 (Private IP :172.31.79.216)
3. Node2 EC2 . (Private IP :172.31.66.122)

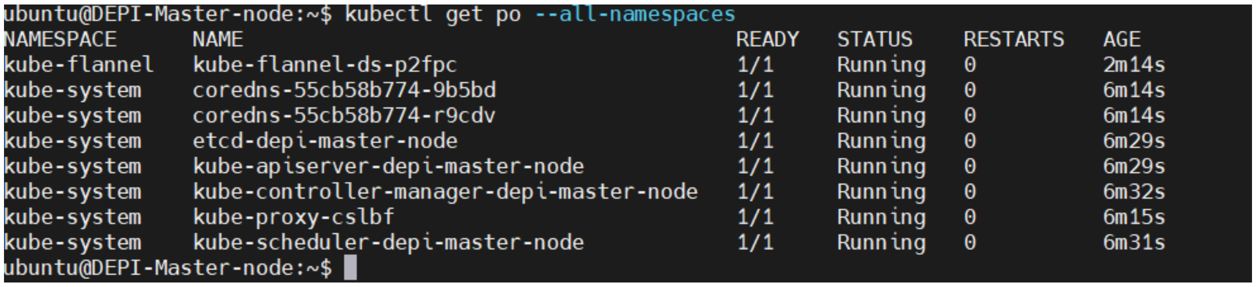


Instances (3/3) Info

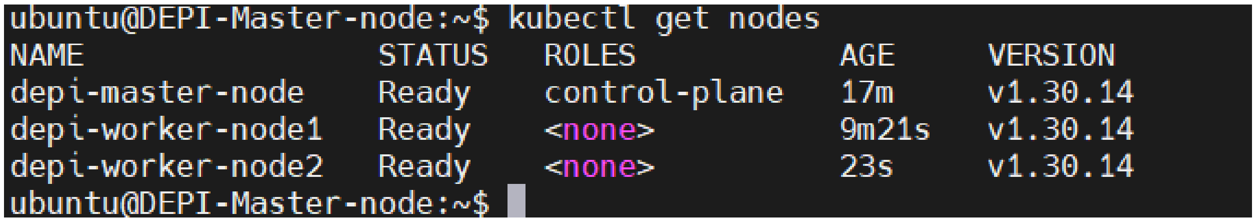
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Publi
DEPI-Master-node	i-086494d93848b9b11	Running	t3.medium	3/3 checks passec	View alarms +	us-east-1f	ec2-3
DEPI-worker-node1	i-04fae1778e64abe3f	Running	t3.medium	3/3 checks passec	View alarms +	us-east-1f	ec2-3
DEPI-worker-node2	i-07922cc395f52cd15	Running	t3.medium	3/3 checks passec	View alarms +	us-east-1f	ec2-1



```
ubuntu@DEPI-Master-node:~$ kubectl get nodes
NAME               STATUS   ROLES           AGE     VERSION
depi-master-node   Ready    control-plane   5m49s   v1.30.14
ubuntu@DEPI-Master-node:~$
```

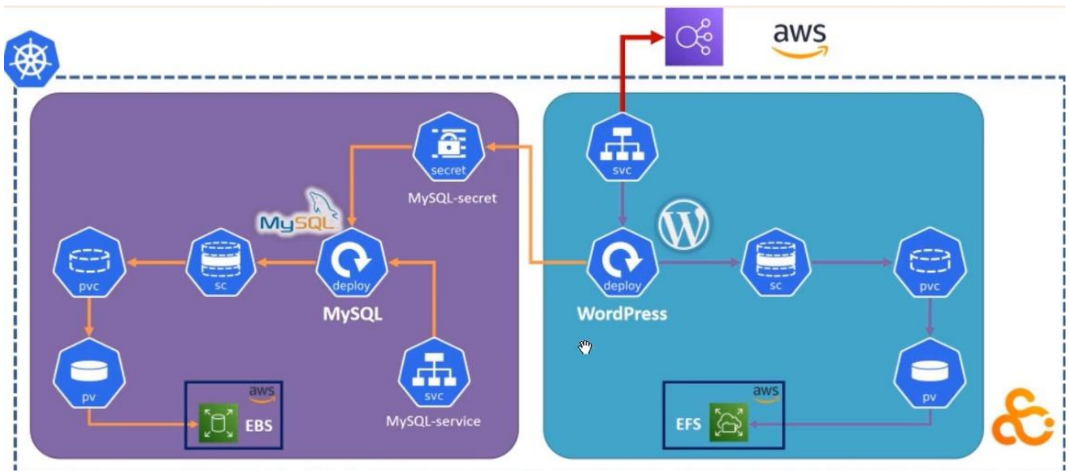


```
ubuntu@DEPI-Master-node:~$ kubectl get po --all-namespaces
NAMESPACE      NAME                                       READY   STATUS    RESTARTS   AGE
kube-flannel   kube-flannel-ds-p2fpc                      1/1     Running   0          2m14s
kube-system    coredns-55cb58b774-9b5bd                   1/1     Running   0          6m14s
kube-system    coredns-55cb58b774-r9cdv                   1/1     Running   0          6m14s
kube-system    etcd-depi-master-node                      1/1     Running   0          6m29s
kube-system    kube-apiserver-depi-master-node            1/1     Running   0          6m29s
kube-system    kube-controller-manager-depi-master-node   1/1     Running   0          6m32s
kube-system    kube-proxy-cslbf                           1/1     Running   0          6m15s
kube-system    kube-scheduler-depi-master-node            1/1     Running   0          6m31s
ubuntu@DEPI-Master-node:~$
```



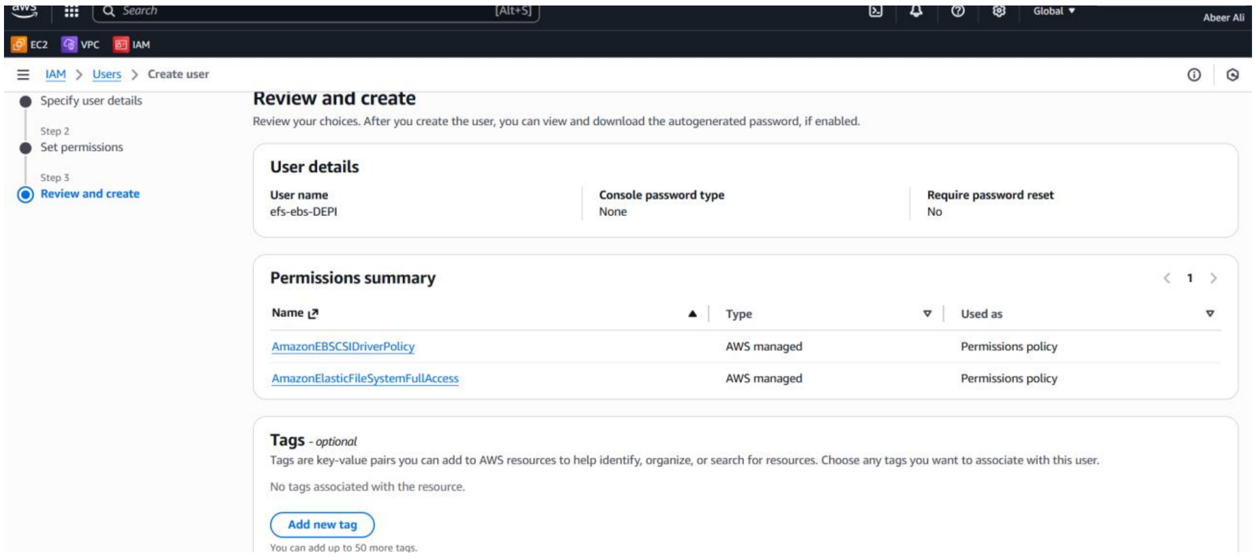
```
ubuntu@DEPI-Master-node:~$ kubectl get nodes
NAME                STATUS   ROLES           AGE     VERSION
depi-master-node    Ready    control-plane   17m     v1.30.14
depi-worker-node1   Ready    <none>          9m21s   v1.30.14
depi-worker-node2   Ready    <none>          23s     v1.30.14
ubuntu@DEPI-Master-node:~$
```

Project Design:



Create IAM User :

- Create User with permissions (EBS, EFS):
- create user with permissions ("AmazonEBSCSIDriverPolicy", "AmazonElasticFileSystemFullAccess")
- create access key for user:
- create role and Add permissions (EBS, EFS):
- Add Role in All   Nodes .
-



Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name	Console password type	Require password reset
efs-ebs-DEPI	None	No

Permissions summary

Name	Type	Used as
AmazonEBSCSIDriverPolicy	AWS managed	Permissions policy
AmazonElasticFileSystemFullAccess	AWS managed	Permissions policy

Tags - *optional*

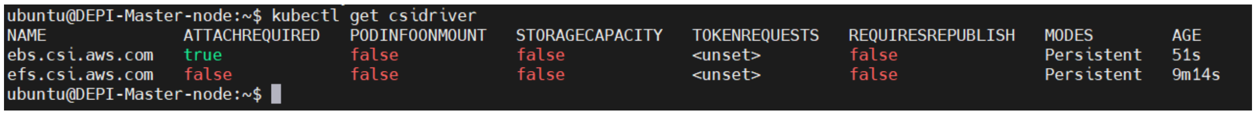
Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

Add new tag

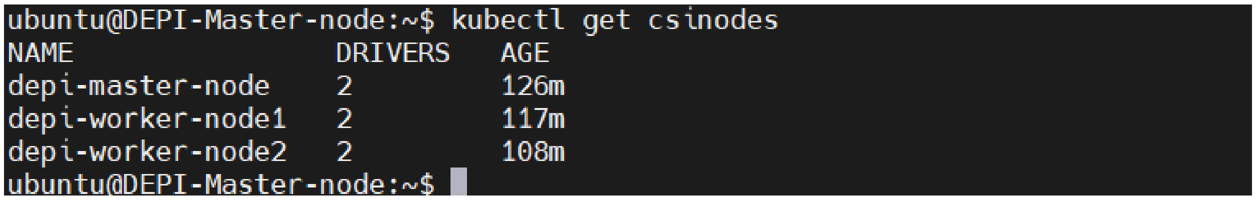
You can add up to 50 more tags.

List CSI drivers:



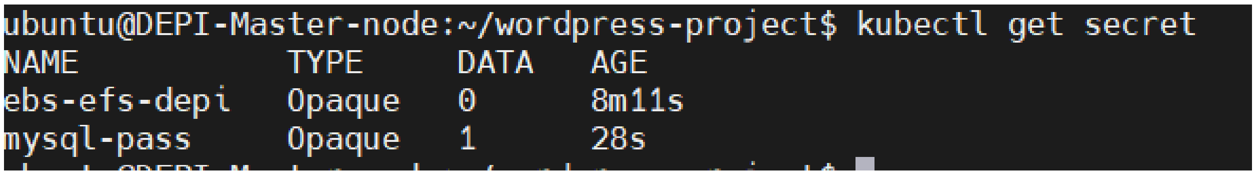
```
ubuntu@DEPI-Master-node:~$ kubectl get csidriver
NAME              ATTACHREQUIRED   PODINFOONMOUNT   STORAGECAPACITY   TOKENREQUESTS   REQUIRESREPUBLISH   MODES        AGE
ebs.csi.aws.com   true             false            false             <unset>         false               Persistent   51s
efs.csi.aws.com   false            false            false             <unset>         false               Persistent   9m14s
ubuntu@DEPI-Master-node:~$
```

Get all CSI Driver on all EC2 instance :



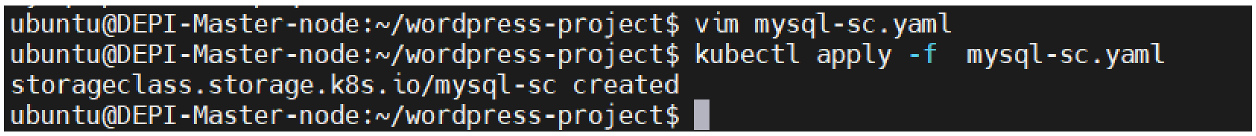
```
ubuntu@DEPI-Master-node:~$ kubectl get csinodes
NAME                 DRIVERS   AGE
depi-master-node     2         126m
depi-worker-node1    2         117m
depi-worker-node2    2         108m
ubuntu@DEPI-Master-node:~$
```

Create Secrets



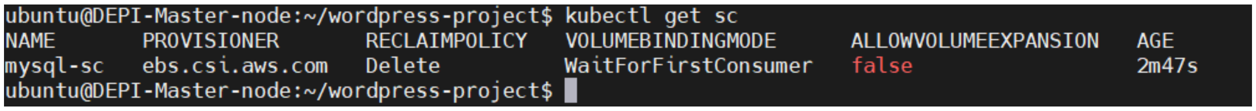
```
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl get secret
NAME           TYPE     DATA   AGE
ebs-efs-depi   Opaque   0      8m11s
mysql-pass     Opaque   1      28s
```

Create storage Class



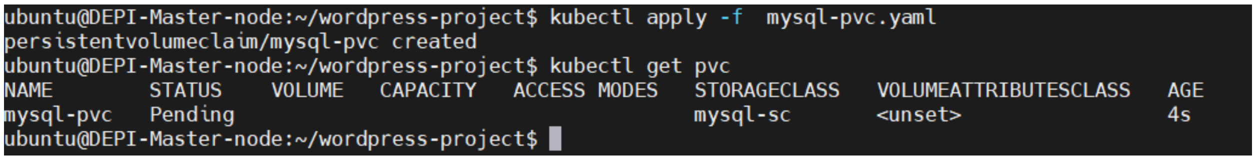
```
ubuntu@DEPI-Master-node:~/wordpress-project$ vim mysql-sc.yaml
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl apply -f  mysql-sc.yaml
storageclass.storage.k8s.io/mysql-sc created
ubuntu@DEPI-Master-node:~/wordpress-project$
```

List Storage class :



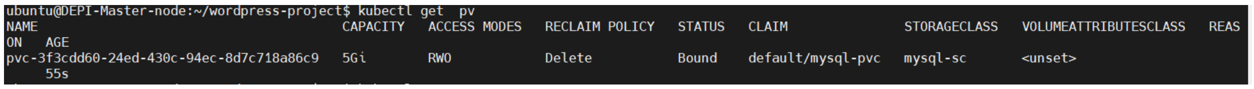
```
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl get sc
NAME       PROVISIONER       RECLAIMPOLICY   VOLUMEBINDINGMODE      ALLOWVOLUMEEXPANSION   AGE
mysql-sc   ebs.csi.aws.com   Delete          WaitForFirstConsumer   false                  2m47s
ubuntu@DEPI-Master-node:~/wordpress-project$
```

Create PVC :



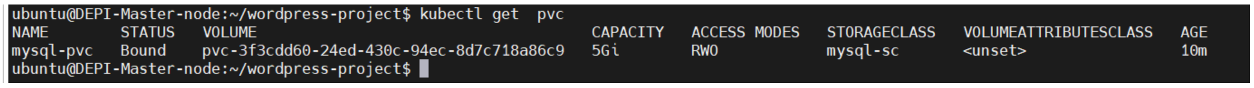
```
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl apply -f  mysql-pvc.yaml
persistentvolumeclaim/mysql-pvc created
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl get pvc
NAME        STATUS    VOLUME   CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
mysql-pvc   Pending                                      mysql-sc       <unset>                 4s
ubuntu@DEPI-Master-node:~/wordpress-project$
```

List PV



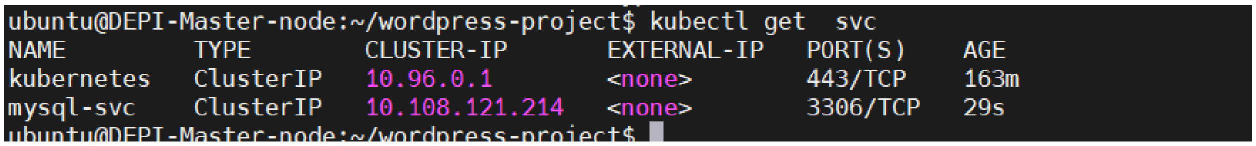
```
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl get  pv
NAME                                       CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                STORAGECLASS   VOLUMEATTRIBUTESCLASS   REAS
ON   AGE
pvc-3f3cdd60-24ed-430c-94ec-8d7c718a86c9   5Gi        RWO            Delete           Bound    default/mysql-pvc    mysql-sc       <unset>
     55s
```

List PVC:



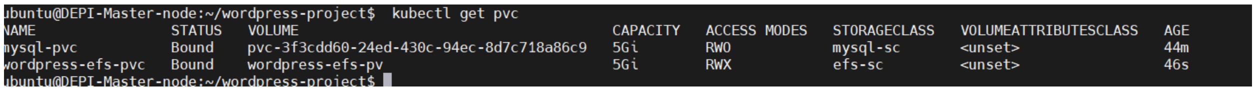
```
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl get  pvc
NAME        STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
mysql-pvc   Bound    pvc-3f3cdd60-24ed-430c-94ec-8d7c718a86c9   5Gi        RWO            mysql-sc       <unset>                 10m
ubuntu@DEPI-Master-node:~/wordpress-project$
```

List service:



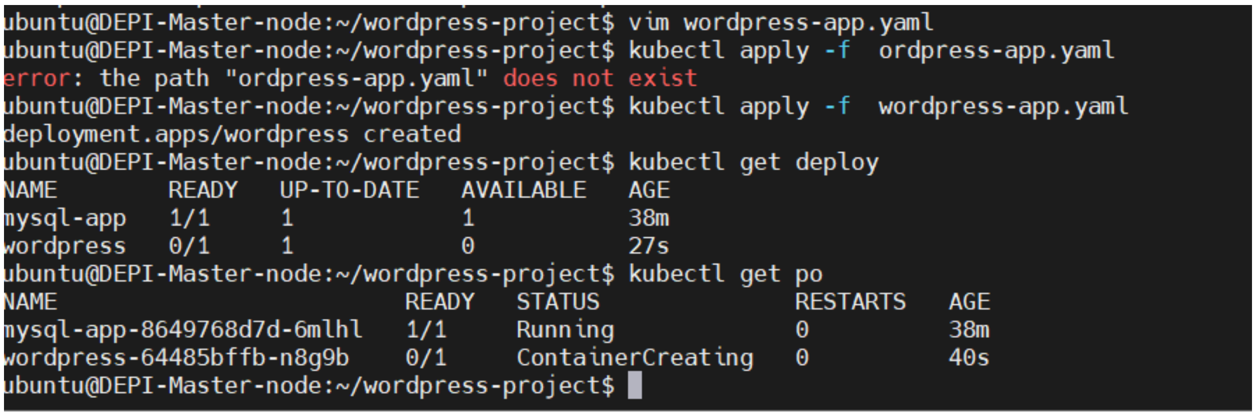
```
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl get  svc
NAME         TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)    AGE
kubernetes   ClusterIP   10.96.0.1        <none>        443/TCP    163m
mysql-svc    ClusterIP   10.108.121.214   <none>        3306/TCP   29s
ubuntu@DEPI-Master-node:~/wordpress-project$
```

List PVC



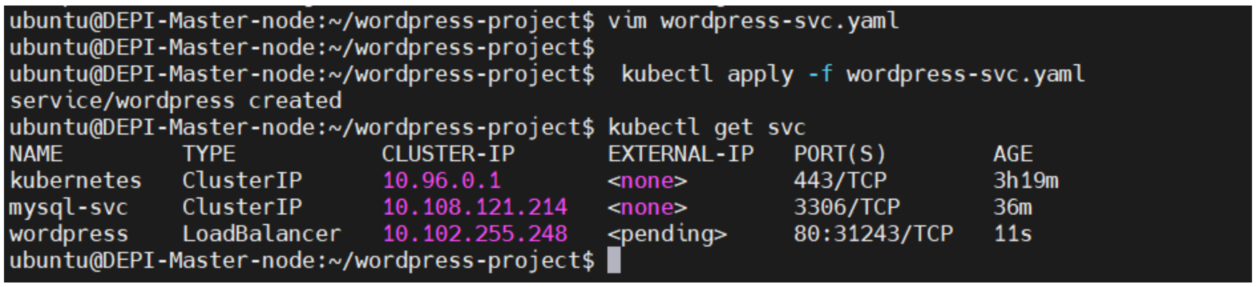
```
ubuntu@DEPI-Master-node:~/wordpress-project$  kubectl get pvc
NAME                 STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
mysql-pvc            Bound    pvc-3f3cdd60-24ed-430c-94ec-8d7c718a86c9   5Gi        RWO            mysql-sc       <unset>                 44m
wordpress-efs-pvc    Bound    wordpress-efs-pv                           5Gi        RWX            efs-sc         <unset>                 46s
ubuntu@DEPI-Master-node:~/wordpress-project$
```

Create deployment



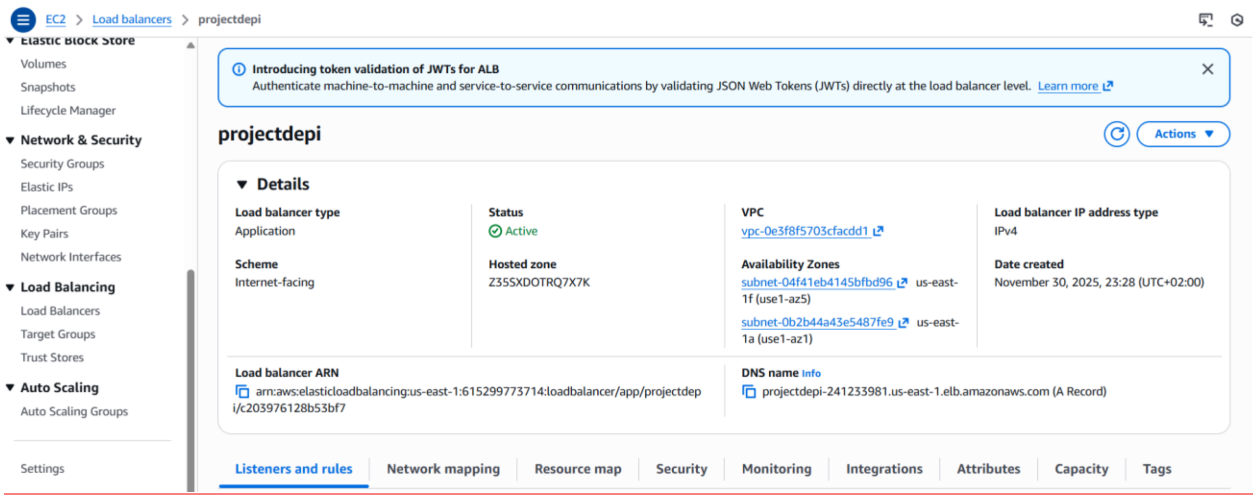
```
ubuntu@DEPI-Master-node:~/wordpress-project$ vim wordpress-app.yaml
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl apply -f  ordpress-app.yaml
error: the path "ordpress-app.yaml" does not exist
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl apply -f  wordpress-app.yaml
deployment.apps/wordpress created
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl get deploy
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
mysql-app   1/1     1            1           38m
wordpress   0/1     1            0           27s
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl get po
NAME                          READY   STATUS              RESTARTS   AGE
mysql-app-8649768d7d-6mlhl    1/1     Running             0          38m
wordpress-64485bffb-n8g9b     0/1     ContainerCreating   0          40s
ubuntu@DEPI-Master-node:~/wordpress-project$
```

Create WordPress service :



```
ubuntu@DEPI-Master-node:~/wordpress-project$ vim wordpress-svc.yaml
ubuntu@DEPI-Master-node:~/wordpress-project$
ubuntu@DEPI-Master-node:~/wordpress-project$  kubectl apply -f wordpress-svc.yaml
service/wordpress created
ubuntu@DEPI-Master-node:~/wordpress-project$ kubectl get svc
NAME         TYPE           CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
kubernetes   ClusterIP      10.96.0.1        <none>        443/TCP        3h19m
mysql-svc    ClusterIP      10.108.121.214   <none>        3306/TCP       36m
wordpress    LoadBalancer   10.102.255.248   <pending>     80:31243/TCP   11s
ubuntu@DEPI-Master-node:~/wordpress-project$
```

Access website through DNS Doman :



Introducing token validation of JWTs for ALB
Authenticate machine-to-machine and service-to-service communications by validating JSON Web Tokens (JWTs) directly at the load balancer level. Learn more

projectdepi

Details

Load balancer type
Application

Status
Active

VPC
vpc-0e3f8f5703cfacdd1

Load balancer IP address type
IPv4

Scheme
Internet-facing

Hosted zone
Z35SXDOTRQ7X7K

Availability Zones
subnet-04f41eb4145bfbd96 us-east-1f (use1-az5)
subnet-0b2b44a43e5487fe9 us-east-1a (use1-az1)

Date created
November 30, 2025, 23:28 (UTC+02:00)

Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:615299773714:loadbalancer/app/projectdepi/c203976128b53bf7

DNS name Info
projectdepi-241233981.us-east-1.elb.amazonaws.com (A Record)