

KUBERNETES PROJECT – 3

DEPLOY A MULTI-TIER WEB APPLICATION ON KUBERNETES

❖ Prerequisites

- Kubernetes Cluster (minikube/kubeadm)
- Docker
- kubectl CLI

Folder Structure & File Usages

k8s-project/

```
|—— mysql/          # MySQL Database Configuration
|   |—— mysql-pv.yaml    # Persistent Volume for MySQL Data Storage
|   |—— mysql-secret.yaml  # Stores MySQL Root Password Securely
|   |—— mysql-deployment.yaml # Deploys MySQL Database as a StatefulSet
|—— flask/           # Flask Backend Configuration
|   |—— app.py        # Flask API Code to Handle Requests
|   |—— Dockerfile     # Flask App Containerization Instructions
|   |—— requirements.txt # Dependencies for Flask
|   |—— flask-deployment.yaml # Deploys Flask Application
|   |—— flask-service.yaml # Exposes Flask App as a Cluster Service
|—— nginx/           # Nginx Configuration
|   |—— nginx-configmap.yaml # Reverse Proxy Configuration for Flask
|   |—— nginx-deployment.yaml # Deploys Nginx
|   |—— nginx-service.yaml # Exposes Nginx via NodePort
```

📄 Explanation of Each File

Persistent Volume (`mysql-pv.yaml`)

- Kubernetes Persistent Volume (PV) is used to store MySQL data permanently.
- Ensures MySQL data is not lost even if the pod restarts.
- The PersistentVolumeClaim (PVC) allows the pod to request storage dynamically.

StatefulSet (`mysql-deployment.yaml`)

- Used instead of Deployment because MySQL requires stable network identity and persistent storage.
- Ensures MySQL pods maintain the same hostname across restarts.

Secret (`mysql-secret.yaml`)

- Stores sensitive credentials securely (e.g., MySQL password).
- Used in MySQL deployment as an environment variable to avoid exposing secrets in YAML.

ConfigMap (`nginx-configmap.yaml`)

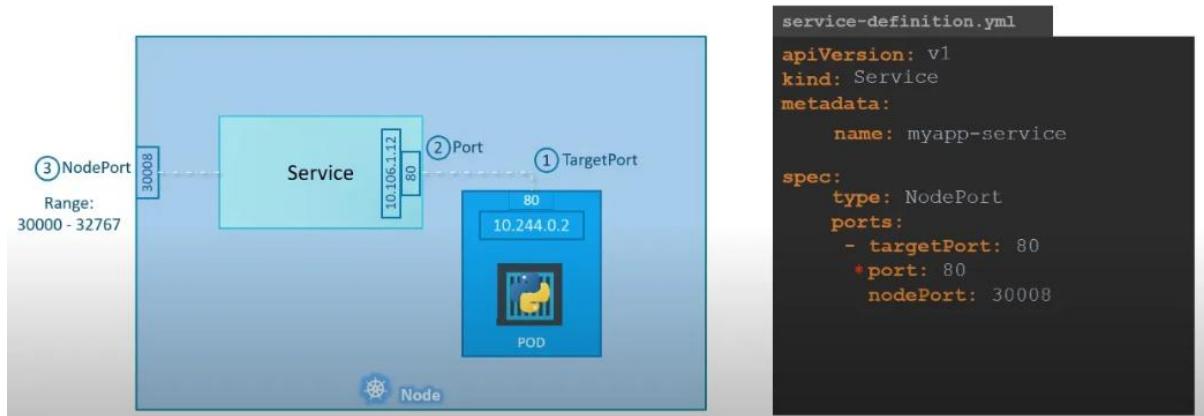
- Stores non-sensitive configuration data.
- Used to configure Nginx as a reverse proxy, mapping requests to Flask.

Deployment (`flask-deployment.yaml` & `nginx-deployment.yaml`)

- Flask Deployment: Runs the Flask application.
- Nginx Deployment: Runs Nginx as a proxy for Flask.

Service (`flask-service.yaml` & `nginx-service.yaml`)

- **Flask Service:** Exposes Flask inside the cluster using `ClusterIP`.
- **Nginx Service:** Exposes Nginx externally using `NodePort`, allowing access from a browser.



Code link : [GitHub – pratheek08/multi-tier-application](https://github.com/pratheek08/multi-tier-application)

Commands:

After code setup enter these commands:

- 1) Build and push the docker image:
 - a. Navigate to dockerfile folder
 - b. docker build -t dockerhub_username/flaskapp .
 - c. docker push dockerhub_username/flaskapp

- 1) cd flask
- 2) kubectl apply -f flask-deployment.yaml
- 3) kubectl apply -f flask-service.yaml
- 4) cd ..; cd mysql
- 5) kubectl apply -f mysql-deployment.yaml
- 6) kubectl apply -f mysql-pv.yaml
- 7) kubectl apply -f mysql-secret.yaml
- 8) cd ..; cd nginx
- 9) kubectl apply -f nginx-configmap.yaml
- 10) kubectl apply -f nginx-deployment.yaml
- 11) kubectl apply -f nginx-service.yaml
- 12) Check Kubernetes pods, deployments, services running status

kubectl get all -o wide

- 13) To access the application:

Kubectl get nodes

Copy worker 1 IP or worker 2 IP and in browser enter <http://<IP>:30007/>

To access users <http://<IP>:30007/users>

```
Activities Workstation || | Home Master Worker1 Wo... master@master-vm: ~/Desktop/project/multi-tier-application/nginx
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 19 (delta 1), reused 16 (delta 1), pack-reused 0 (from 0)
Unpacking objects: 100% (19/19), 4.07 KiB | 520.00 KiB/s, done.
master@master-vm:~/Desktop/project$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
igris08/flask-kube latest 482858c4b5b7 3 days ago 1.03GB
igris08/flask-tube latest 3d3b964541fe 3 days ago 1.03GB
alpine latest aded1e1a5b37 4 weeks ago 7.83MB
nginx latest b52e0b094bc0 5 weeks ago 192MB
gcr.io/k8s-minikube/kicbase v0.0.46 e72c4cbe9b29 2 months ago 1.31GB
python 3.11 78553a4d82cb 3 months ago 1.01GB
master@master-vm:~/Desktop/project$ docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
igris08/multitier latest 61ecec2c07ca 12 seconds ago 1.07GB
igris08/flask-kube latest 482858c4b5b7 3 days ago 1.03GB
igris08/flask-tube latest 3d3b964541fe 3 days ago 1.03GB
alpine latest aded1e1a5b37 4 weeks ago 7.83MB
nginx latest b52e0b094bc0 5 weeks ago 192MB
gcr.io/k8s-minikube/kicbase v0.0.46 e72c4cbe9b29 2 months ago 1.31GB
python 3.11 78553a4d82cb 3 months ago 1.01GB
python 3.8 3ea6eaad4f17 6 months ago 995MB
master@master-vm:~/Desktop/project$
master@master-vm:~/Desktop/project$ cd multi-tier-application/
master@master-vm:~/Desktop/project/multi-tier-application$ cd flask/
Activate Windows
Go to Settings to activate Windows.
```

```
Activities Workstation || | Home Master Worker1 Wo... master@master-vm: ~/Desktop/project/multi-tier-application/nginx
deployment.apps/flask-app created
master@master-vm:~/Desktop/project/multi-tier-application/flask$ kubectl apply -f flask-service.yaml
service/flask-service created
master@master-vm:~/Desktop/project/multi-tier-application/flask$ cd ..
master@master-vm:~/Desktop/project/multi-tier-application$ ls
flask mysql nginx README.md
master@master-vm:~/Desktop/project/multi-tier-application$ cd mysql/
master@master-vm:~/Desktop/project/multi-tier-application/mysql$ ls
mysql-deployment.yaml mysql-pv.yaml mysql-secret.yaml
master@master-vm:~/Desktop/project/multi-tier-application/mysql$ kubectl get svc
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
flask-service ClusterIP 10.97.152.95 <none> 5000/TCP 12m
kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 33m
master@master-vm:~/Desktop/project/multi-tier-application/mysql$ ls
mysql-deployment.yaml mysql-pv.yaml mysql-secret.yaml
master@master-vm:~/Desktop/project/multi-tier-application/mysql$ kubectl apply -f mysql-deployment.yaml
deployment.apps/mysql created
service/mysql created
master@master-vm:~/Desktop/project/multi-tier-application/mysql$ kubectl apply -f mysql-pv.yaml
persistentvolume/mysql-pv created
persistentvolumeclaim/mysql-pvc created
[[master@master-vm:~/Desktop/project/multi-tier-application/mysql$ kubectl apply -f mysql-secret.yaml
secret/mysql-secret created
master@master-vm:~/Desktop/project/multi-tier-application/mysql$ cd .
master@master-vm:~/Desktop/project/multi-tier-application/mysql$ cd ..
master@master-vm:~/Desktop/project/multi-tier-application$ cd nginx/
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ ld
Activate Windows
Go to Settings to activate Windows.
```

Command 'ld' not found, but can be installed with:

The screenshot shows a terminal window titled "master@master-vm: ~/Desktop/project/multi-tier-application/nginx". The terminal displays the following command sequence:

```
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ kubectl get pods -o wide
NAME          READY   STATUS    RESTARTS   AGE     IP           NODE    NOMINATED NODE   READINESS
SS GATES
flask-app-557854b94f-5s7sj  1/1    Running   0          108m   192.168.147.81  worker2-vm  <none>      <none>
flask-app-557854b94f-fhsdz  1/1    Running   0          108m   192.168.147.44  worker1-vm  <none>      <none>
flask-app-557854b94f-tllt4  1/1    Running   0          108m   192.168.147.43  worker1-vm  <none>      <none>
flask-app-557854b94f-z79s9  1/1    Running   0          108m   192.168.147.80  worker2-vm  <none>      <none>
mysql-66d468f74c-zcbzk    1/1    Running   0          16m    192.168.147.82  worker2-vm  <none>      <none>
nginx-78b5b78b99-mmm4h    1/1    Running   0          15m    192.168.147.83  worker2-vm  <none>      <none>
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ kubectl delete deployment ^C
Error from server (NotFound): deployments.apps "nginx-78b5b78b99-mmm4h" not found
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ kubectl delete deployment nginx
deployment.apps "nginx" deleted
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ ls
nginx-configmap.yaml  nginx-deployment.yaml  nginx-service.yaml
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ kubectl apply -f nginx-configmap.yaml
configmap/nginx-config unchanged
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx created
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ kubectl apply -f nginx-service.yaml
service/nginx-service unchanged
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ nano nginx-configmap.yaml
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ kubectl delete deployment nginx
deployment.apps "nginx" deleted
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ kubectl apply -f nginx-configmap.yaml
configmap/nginx-config configured
master@master-vm:~/Desktop/project/multi-tier-application/nginx$ kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx created
```

A tooltip "Activate Windows" with "to activate Windows." is visible near the bottom right of the terminal window.

Access MySQL Inside the Pod and add the data

```
kubectl exec -it mysql-0 -- mysql -u root -p

CREATE DATABASE mydb;

USE mydb;

SHOW TABLES;

CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    email VARCHAR(100)
);

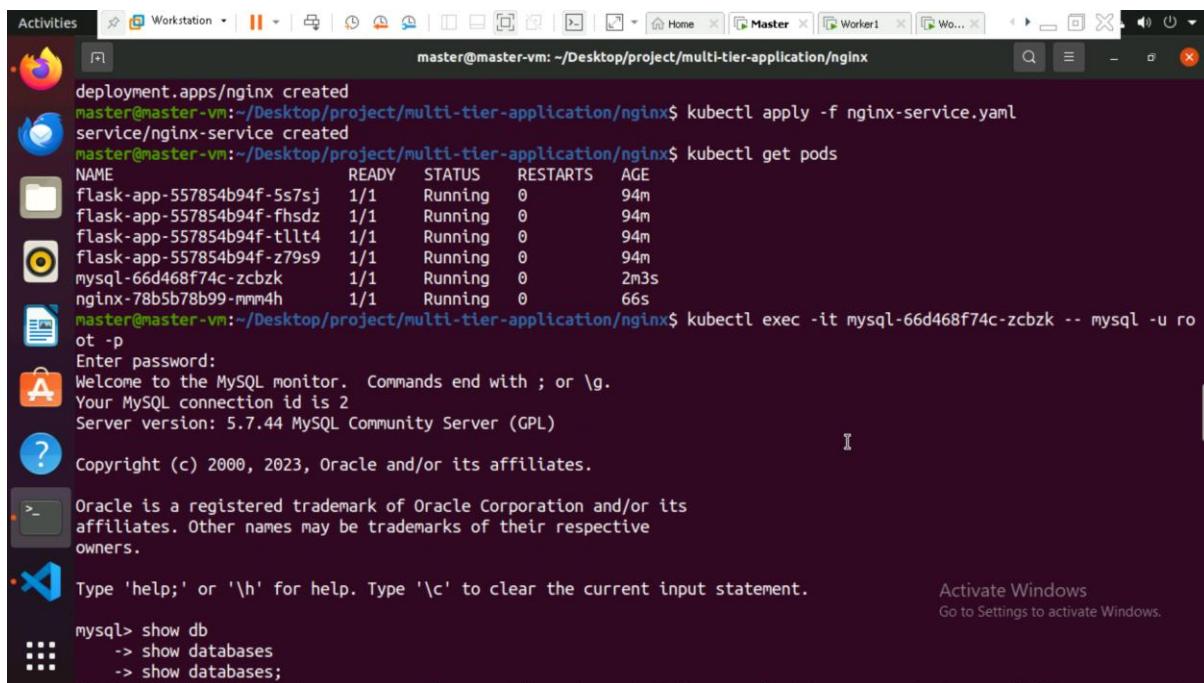
INSERT INTO users (name, email) VALUES ('Alice', 'alice@example.com');

INSERT INTO users (name, email) VALUES ('Bob', 'bob@example.com');

SELECT * FROM users;

GRANT ALL PRIVILEGES ON mydb.* TO 'user'@'%';

FLUSH PRIVILEGES;
```



The screenshot shows a terminal window titled "master@master-vm: ~/Desktop/project/multi-tier-application/nginx\$". The terminal displays the following sequence of commands and outputs:

- Deployment of nginx: deployment.apps/nginx created
- Service creation: service/nginx-service created
- Pod listing: kubectl get pods

NAME	READY	STATUS	RESTARTS	AGE
flask-app-557854b94f-5s7sj	1/1	Running	0	94m
flask-app-557854b94f-fhsdz	1/1	Running	0	94m
flask-app-557854b94f-tllt4	1/1	Running	0	94m
flask-app-557854b94f-z79s9	1/1	Running	0	94m
mysql-66d468f74c-zcbzk	1/1	Running	0	2m3s
nginx-78b5b78b99-mmw4h	1/1	Running	0	66s

- MySQL connection: kubectl exec -it mysql-66d468f74c-zcbzk -- mysql -u root -p
- Welcome message from MySQL monitor
- MySQL version: Server version: 5.7.44 MySQL Community Server (GPL)
- Copyright notice: Copyright (c) 2000, 2023, Oracle and/or its affiliates.
- Trademark notice: Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.
- Help message: Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
- Database listing: mysql> show db
-> show databases
-> show databases;

```
n for the right syntax to use near 'db
show databases
show databases' at line 1
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mydb |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

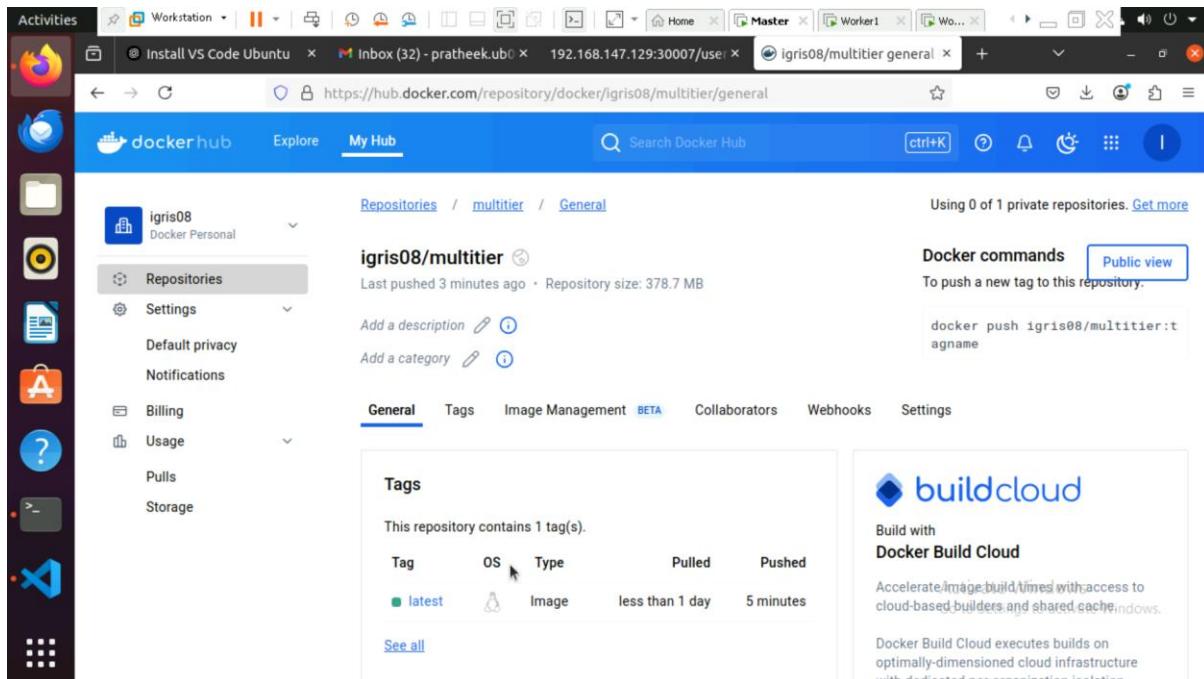
mysql> use mydb;
Database changed
mysql> show tables;
Empty set (0.00 sec)

mysql> CREATE TABLE users (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     name VARCHAR(100),
    ->     email VARCHAR(100)
    -> );
Query OK, 0 rows affected (0.04 sec)

mysql> insert into users(name, email) values ('Pratheeek', 'pratheeek.ub008@gmail.com');
Query OK, 1 row affected (0.01 sec)
```

Activate Windows
Go to Settings to activate Windows.

OUTPUT:



The screenshot shows a Linux desktop environment with a terminal window and a browser window.

Terminal Window:

```
192.168.147.129:30007/users
```

Browser Window:

192.168.147.129:30007/users

JSON Raw Data Headers

Save Copy Collapse All Expand All Filter JSON

```
[{"id": 1, "name": "Pratheek", "email": "pratheek.ub008@gmail.com"}]
```

Activate Windows
Go to Settings to activate Windows.

The screenshot shows a Linux desktop environment with a terminal window and a browser window.

Terminal Window:

```
192.168.147.129:30007
```

Browser Window:

Not Secure 192.168.147.129:30007

Hello from Flask with MySQL!

Activate Windows
Go to Settings to activate Windows.