NAGP KUBERNETES AND DEVOPS ASSIGNMENT DOCUMENTATION

This document provides a step-by-step guide to the implementation of the Kubernetes and DevOps assignment

Table of Contents

1. Application Details

1.1. Requirement Understanding

The goal is to design and deploy a containerized multi-tier application on Google Kubernetes Engine (GKE). It consists of:

- **Service API Tier:** A Java Spring Boot application exposing an HTTP API that fetches data from a PostgreSQL database.
- **Database Tier:** A PostgreSQL instance storing 5–10 records with persistent storage.

Kubernetes Requirements:

- 4 replicas for API, 1 pod for DB
- API exposed externally via Ingress, DB is internal-only
- Rolling updates for API, data persistence for DB
- DB config managed via ConfigMap, password via Secret
- Communication via service DNS, not Pod Ips

1.2. Assumptions

- Java Spring Boot with Maven used for the API service
- PostgreSQL database used, preloaded with test data
- GKE cluster with default StorageClass and Ingress controller (e.g., GCLB)
- Docker images hosted on Docker Hub
- No CI/CD pipeline is included, only manual deployment considered.

1.3. Solution Overview

The architecture includes two main components:

API Tier:

- A **Spring Boot** application containerized and deployed with 4 replicas using a Kubernetes **Deployment**.
- A **ClusterIP Service** exposes the API internally, while an **Ingress** exposes it externally.
- Configurations such as DB URL, host, and user are injected via **ConfigMap**, the DB password is provided via a **Secret**.
- Rolling updates are enabled by default, ensuring zero downtime during deployments.

Database Tier:

- A single **PostgreSQL** pod managed via Deployment or StatefulSet.
- Uses PersistentVolumeClaim **(PVC)** to store data on a Persistent Disk.
- A **ClusterIP Service** allows the API tier to communicate via DNS (postgres-svc).
- Environment variables like POSTGRES_DB, POSTGRES_USER, and POSTGRES_PASSWORD are set using ConfigMap and Secret.

Communication and Exposure:

- The Spring Boot app connects to the DB using environment variables injected from ConfigMap and Secret.
- Connection pooling ensures efficient DB access.
- The database is accessed using Kubernetes DNS (postgres-svc:5432), not Pod IPs.
- External HTTP traffic is routed through **Ingress**, which maps to the API service.

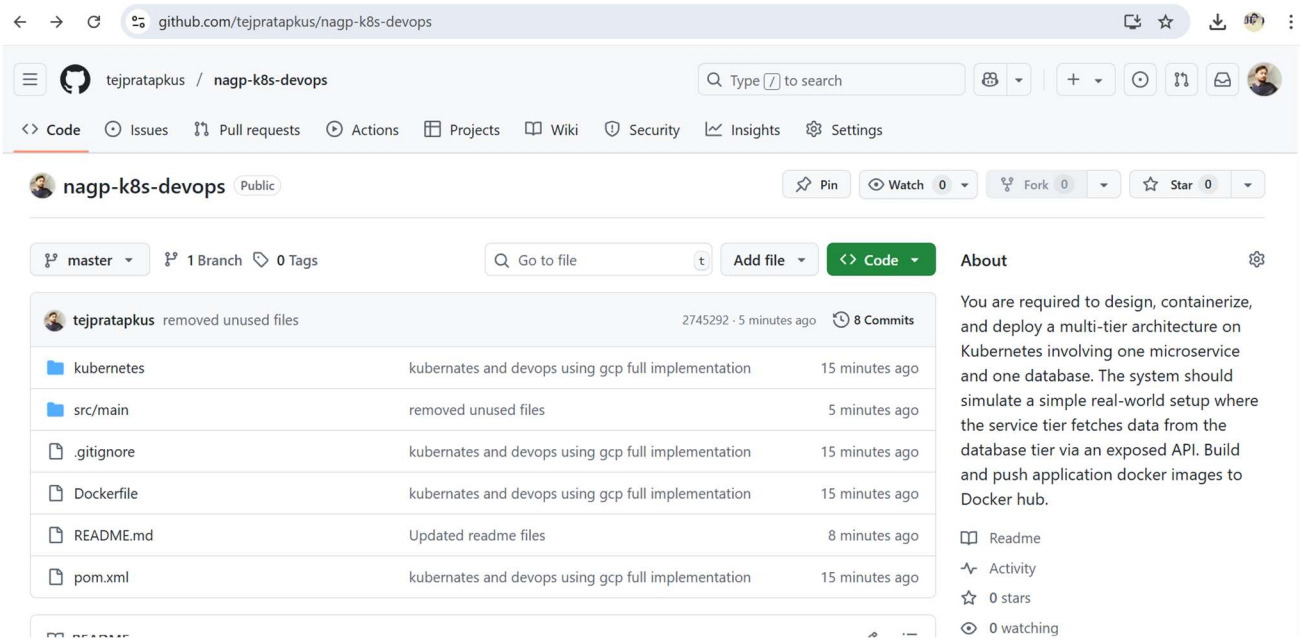
1.4. Justification for Resource Utilization

- **Spring Boot** is production-ready, integrates well with PostgreSQL, and supports external configuration.
- **PostgreSQL** provides reliable data persistence and is widely supported in Kubernetes.
- **Deployments** manage lifecycle, replicas, and rolling updates for the API, 4 replicas ensure scalability and high availability.
- **PersistentVolumeClaim** ensures database data is retained even when the pod restarts or redeploys.
- **ConfigMap** and **Secret** cleanly separate configuration and credentials from code, aligning with security best practices.
- **Ingress** enables controlled, path-based routing of external requests into the cluster and simplifies load balancing with one IP.

2. GitHub Link

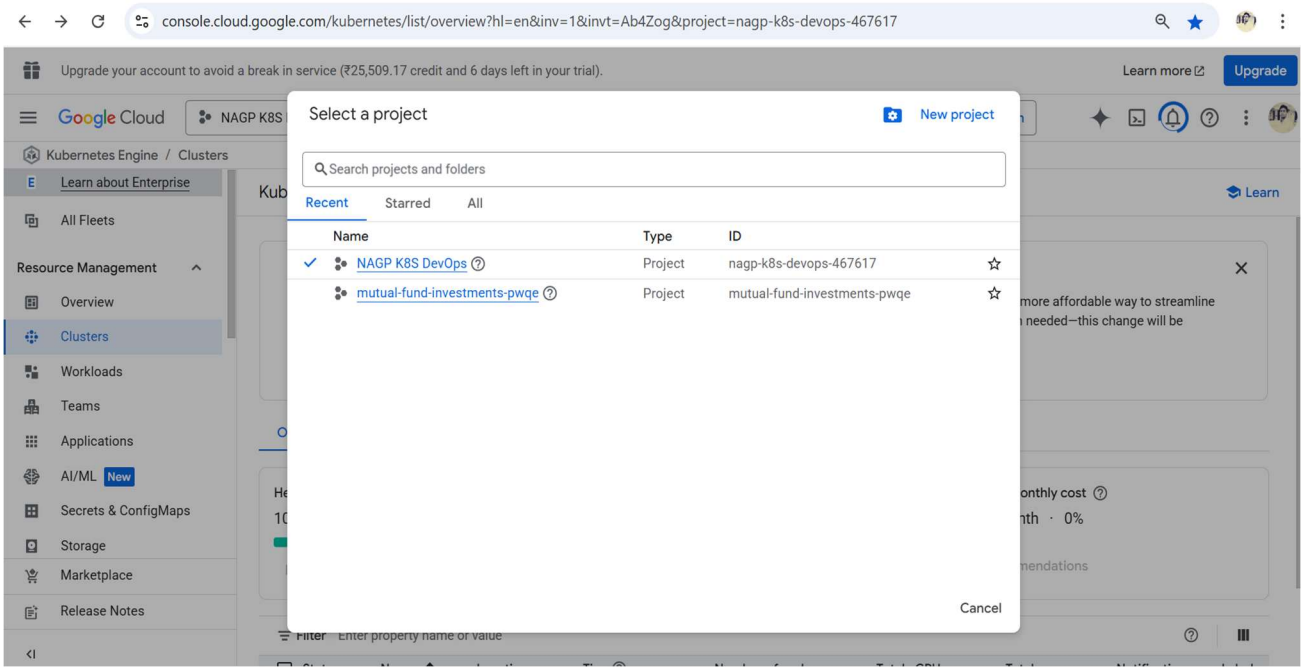
2.1. GitHub Link

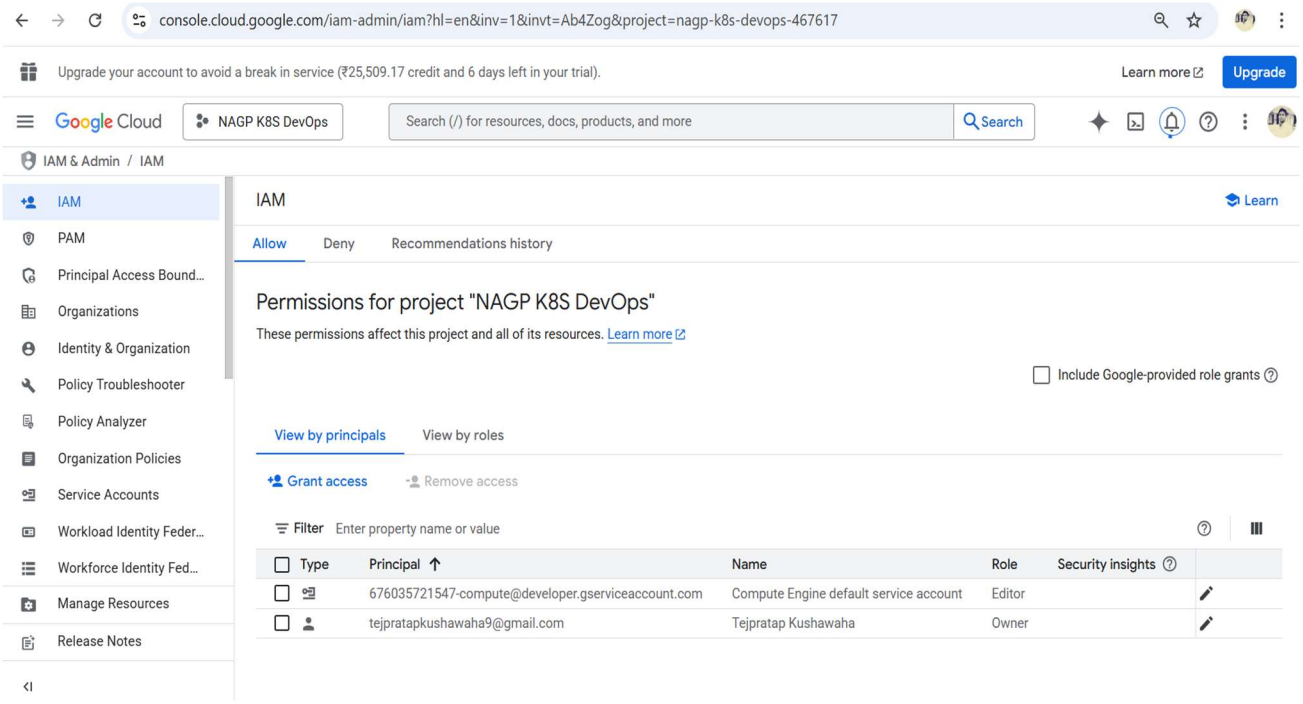
https://github.com/tejpratapkus/nagp-k8s-devops



3. GCP DevOps Setup

3.1. Project Creation

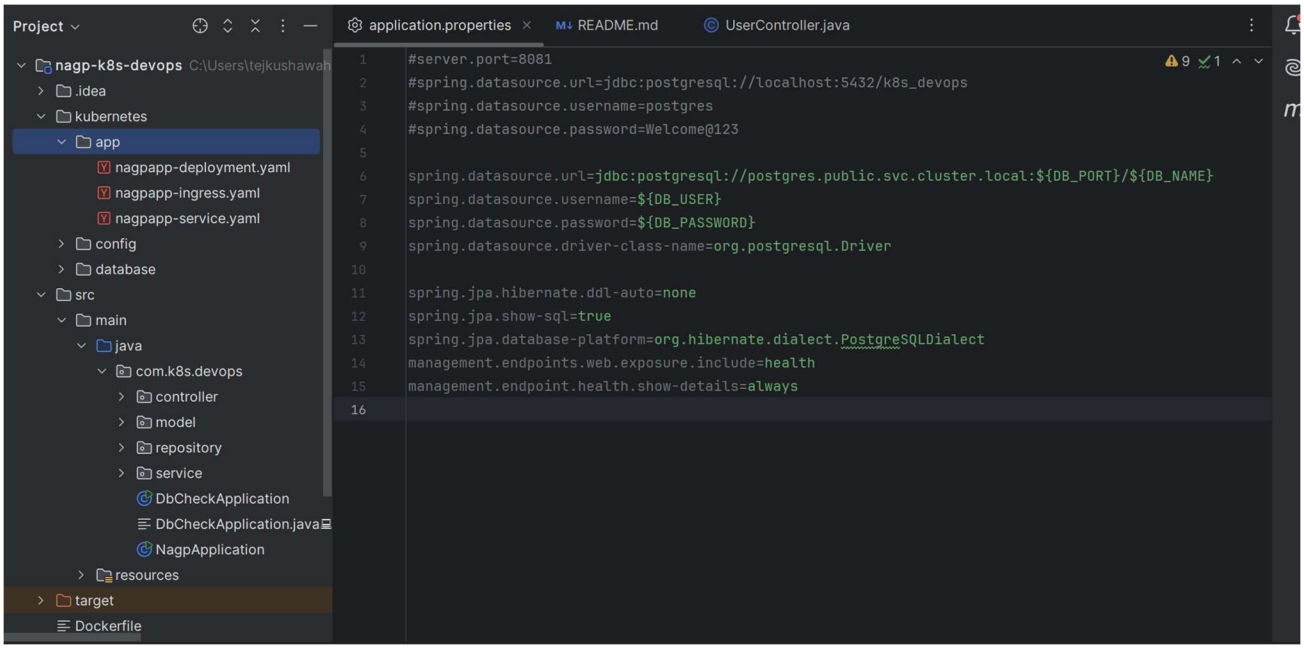




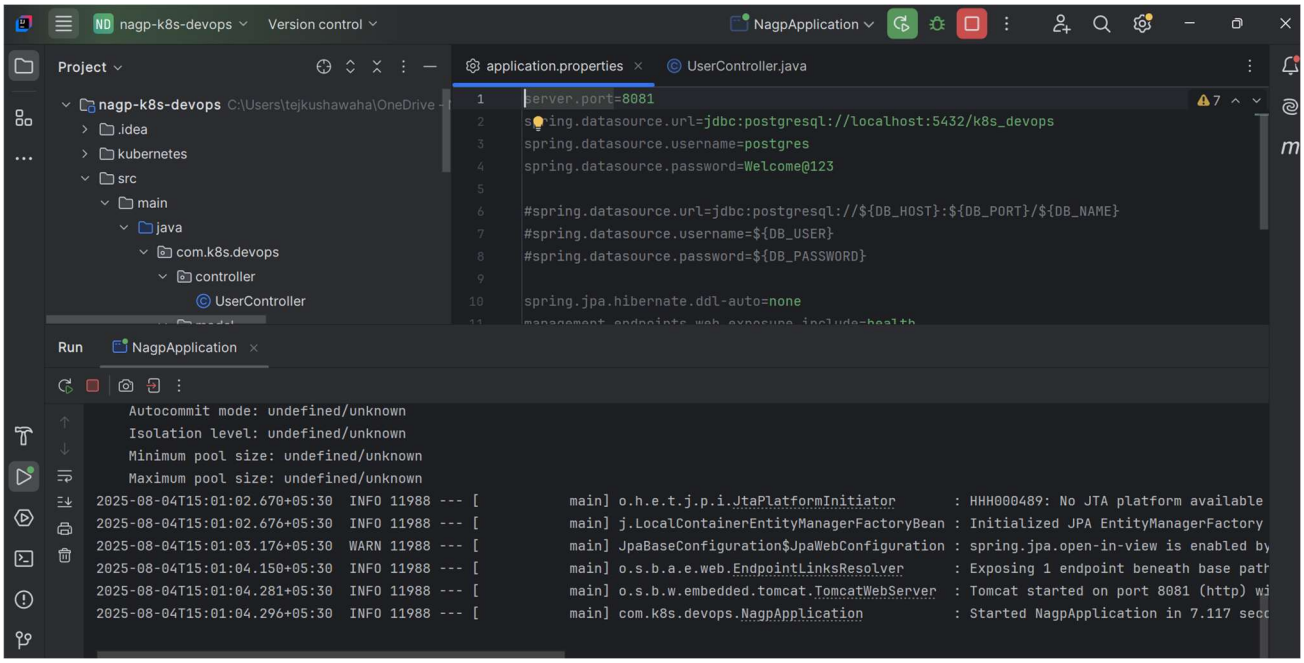
4. Application Code

4.1. Application Structure

The application is a simple Java, Spring Boot, and Maven application with Postgres database.



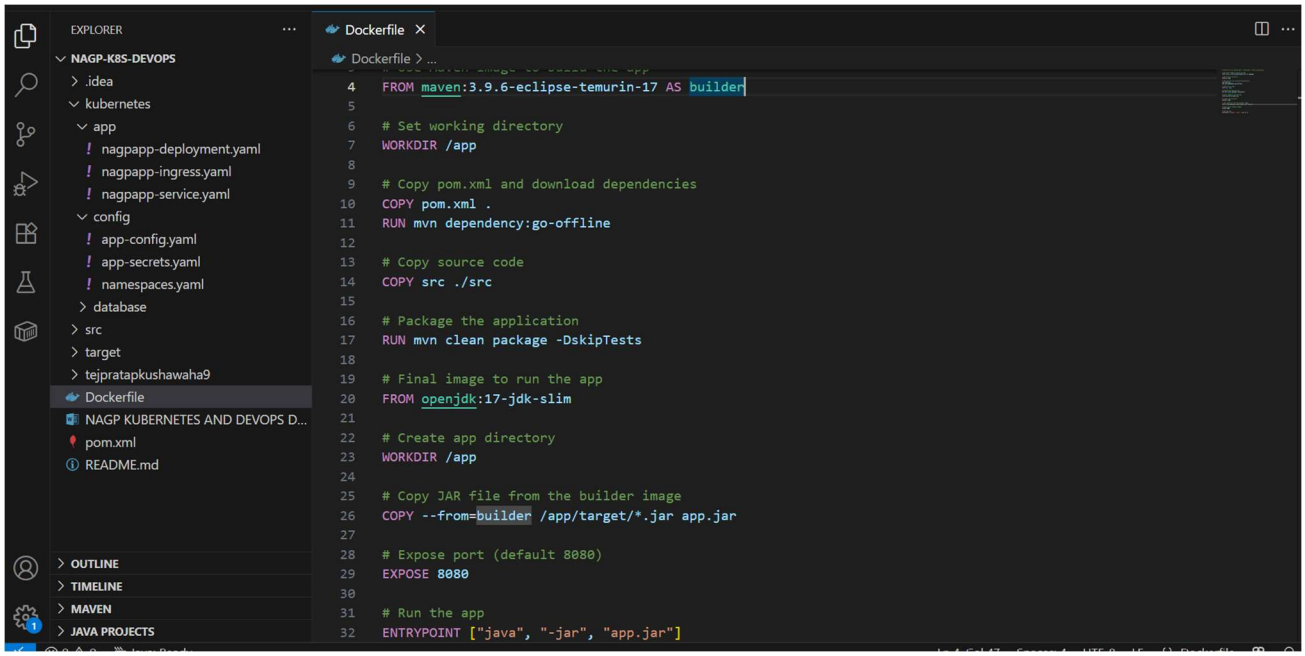
4.2. Running Application Locally



4.3. Running Application Locally

5. Docker Configuration

5.1. Dockerfile



5.2. Build Docker Image

```
sudo docker build -t nagp-app:latest .
```

```
tejpratap@IN-92R6C54:~/nagp-assignment$ sudo docker build -t nagp-app:latest .
[sudo] password for tejpratap:
[+] Building 11.1s (15/15) FINISHED                                                                                    docker:default
 => [internal] load build definition from Dockerfile                                                                              0.0s
 => => transferring dockerfile: 678B                                                                                              0.0s
 => [internal] load metadata for docker.io/library/openjdk:17-jdk-slim                                                            1.9s
 => [internal] load metadata for docker.io/library/maven:3.9.6-eclipse-temurin-17                                                 1.9s
 => [internal] load .dockerignore                                                                                                 0.0s
 => => transferring context: 2B                                                                                                   0.0s
 => [stage-1 1/3] FROM docker.io/library/openjdk:17-jdk-slim@sha256:aaa3b3cb27e3e520b8f116863d0580c438ed55ecfa0bc126b41f68c3f6    0.0s
 => [internal] load build context                                                                                                 0.0s
 => => transferring context: 2.13kB                                                                                               0.0s
 => [builder 1/6] FROM docker.io/library/maven:3.9.6-eclipse-temurin-17@sha256:29a1658b1f3078e07c2b17f7b519b45eb47f65d9628e887    0.0s
 => CACHED [builder 2/6] WORKDIR /app                                                                                             0.0s
 => CACHED [builder 3/6] COPY pom.xml .                                                                                           0.0s
 => CACHED [builder 4/6] RUN mvn dependency:go-offline                                                                            0.0s
 => [builder 5/6] COPY src ./src                                                                                                  0.1s
 => [builder 6/6] RUN mvn clean package -DskipTests                                                                               7.7s
 => CACHED [stage-1 2/3] WORKDIR /app                                                                                             0.0s
 => [stage-1 3/3] COPY --from=builder /app/target/*.jar app.jar                                                                   0.1s
 => exporting to image                                                                                                            0.1s
 => => exporting layers                                                                                                           0.1s
 => => writing image sha256:8ac14b8bce30488cf98ffb278106b4c9bb734ff5d5acb9d27cb4f98c89ef1852                                      0.0s
 => => naming to docker.io/library/nagp-app:latest                                                                                0.0s
tejpratap@IN-92R6C54:~/nagp-assignment$
```

5.3. Create Docker Tag

```
docker tag nagp-app tejpratap1/nagp-app:latest
```

```
tejpratap@IN-92R6C54:~/nagp-assignment$
tejpratap@IN-92R6C54:~/nagp-assignment$ docker tag nagp-app tejpratap1/nagp-app:latest
tejpratap@IN-92R6C54:~/nagp-assignment$
```

5.4. Check Docker Images

`docker images`

```
tejpratap@IN-92R6C54:~/nagp-assignment$ docker images
REPOSITORY                    TAG       IMAGE ID       CREATED          SIZE
tejpratap1/nagp-assignment    latest    8ac14b8bce30   4 minutes ago    462MB
nagp-app                      latest    8ac14b8bce30   4 minutes ago    462MB
tejpratap1/nagp-app           latest    8ac14b8bce30   4 minutes ago    462MB
nagp-assignment               latest    0e8192bf783b   23 minutes ago   462MB
tejpratap1/nagp-assignment    <none>    0e8192bf783b   23 minutes ago   462MB
tejpratap1/nagp-assignment    <none>    bf3921ba508b   37 minutes ago   462MB
tejpratap1/nagp-assignment    <none>    8b8bc06aa6cf   25 hours ago     462MB
postgres                      15        af3ea6376a7b   8 weeks ago      430MB
hello-world                   latest    74cc54e27dc4   6 months ago     10.1kB
tejpratap@IN-92R6C54:~/nagp-assignment$
```

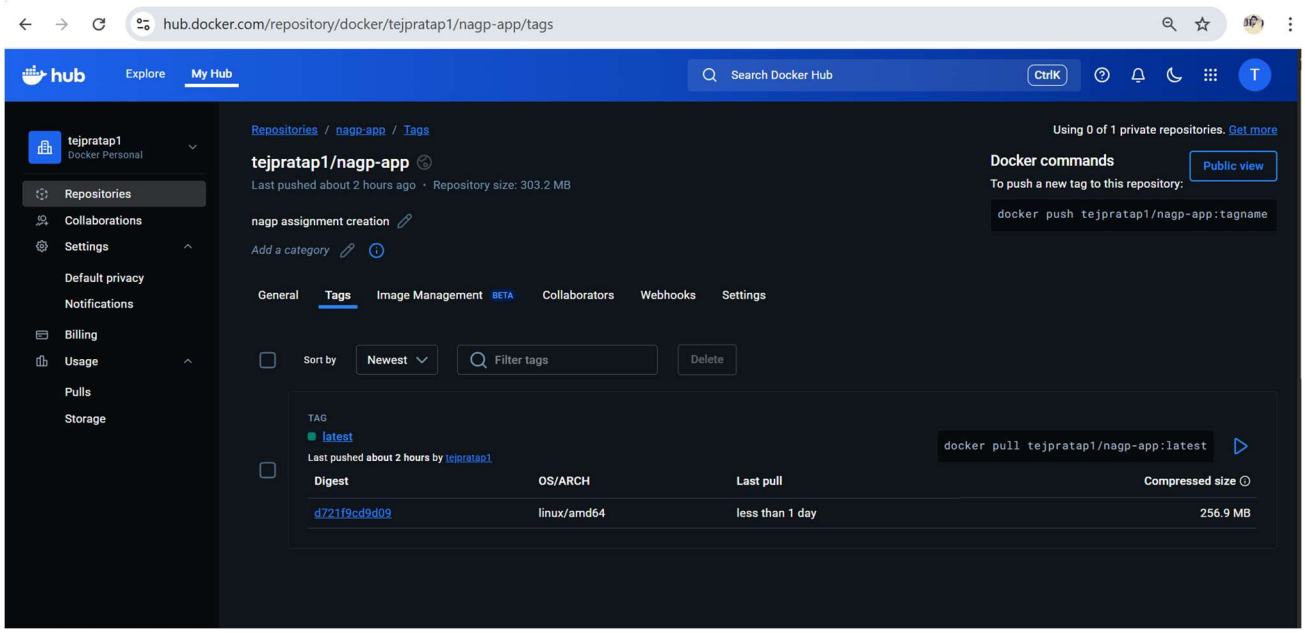
5.5. Push Docker Image

`docker push tejpratap1/nagp-app:latest`

```
tejpratap@IN-92R6C54:~/nagp-assignment$ docker push tejpratap1/nagp-app:latest
The push refers to repository [docker.io/tejpratap1/nagp-app]
c5481e38522a: Pushed
9f21964aa4a5: Mounted from tejpratap1/nagp-assignment
6be690267e47: Mounted from tejpratap1/nagp-assignment
13a34b6fff78: Mounted from tejpratap1/nagp-assignment
9c1b6dd6c1e6: Mounted from tejpratap1/nagp-assignment
latest: digest: sha256:4e56868b637ac906a3b0eb65985a28745710b3783e9127f6a85e3e39391f02b0 size: 1371
tejpratap@IN-92R6C54:~/nagp-assignment$
```

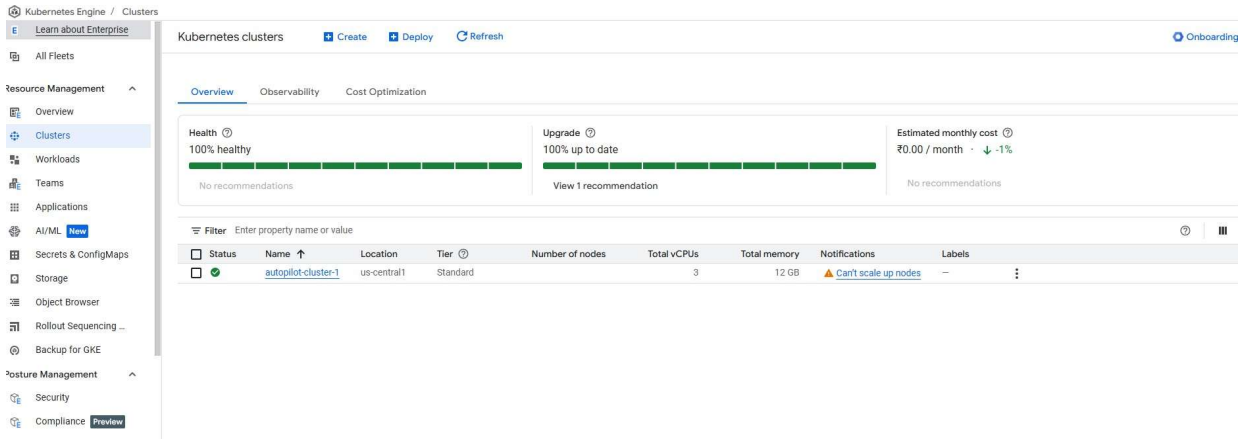
5.6. Docker Hub Repository

https://hub.docker.com/repositories/tejpratap1



6. Kubernetes Cluster Setup

6.1. Cluster Creation



6.2. Check Running Pod

kubectl get pods

```
tejpratapkushawaha9@cloudshell:~ (nagp-k8s-devops-467617)$ kubectl get pods
No resources found in default namespace.
tejpratapkushawaha9@cloudshell:~ (nagp-k8s-devops-467617)$
```

kubectl get pods -n public

```
tejpratapkushawaha9@cloudshell:~ (nagp-k8s-devops-467617)$ kubectl get pods -n public
No resources found in public namespace.
tejpratapkushawaha9@cloudshell:~ (nagp-k8s-devops-467617)$
```

6.3. Namespace Creation

cd kubernetes/config/
kubectl apply -f namespaces.yaml

```
tejpratapkushawaha9@cloudshell:~/kubernetes/config (nagp-k8s-devops-467617)$ kubectl apply -f namespaces.yaml
namespace/public created
tejpratapkushawaha9@cloudshell:~/kubernetes/config (nagp-k8s-devops-467617)$
```

6.4. ConfigMaps and Secrets

kubectl apply -f app-config.yaml

```
tejpratapkushawaha9@cloudshell:~/kubernetes/config (nagp-k8s-devops-467617)$ kubectl apply -f app-config.yaml
configmap/app-config created
tejpratapkushawaha9@cloudshell:~/kubernetes/config (nagp-k8s-devops-467617)$
```

kubectl apply -f app-secrets.yaml

```
tejpratapkushawaha9@cloudshell:~/kubernetes/config (nagp-k8s-devops-467617)$ kubectl apply -f app-secrets.yaml
secret/app-secrets created
tejpratapkushawaha9@cloudshell:~/kubernetes/config (nagp-k8s-devops-467617)$
```

7. PostgreSQL Deployment

7.1. Deploy PersistentVolumeClaim for PostgreSQL

```
cd ../database/
kubectl apply -f postgres-pvc.yaml
```

```
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$ kubectl apply -f postgres-pvc.yaml
persistentvolumeclaim/postgres-pvc created
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$
```

This will create a PersistentVolume automatically via default storage class.

7.2. Deploy PostgreSQL

```
kubectl apply -f postgres-deployment.yaml
```

```
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$ kubectl apply -f postgres-deployment.yaml
Warning: autopilot-default-resources-mutator:Autopilot updated Deployment public/postgres: defaulted unspecified 'cpu' resource for containers [postgres] (see http://g.co/gke/autopilot-defaults).
deployment.apps/postgres created
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$
```

```
kubectl apply -f postgres-service.yaml
```

```
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$ kubectl apply -f postgres-service.yaml
service/postgres created
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$
```

This creates the DB pod and service. Verify with:

```
kubectl get pods -n public
```

```
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$ kubectl get pods -n public
NAME                        READY   STATUS    RESTARTS   AGE
postgres-8549d8f6b9-hxqt2   1/1     Running   0          7m7s
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$
```

```
kubectl get svc postgres -n public
```

```
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$ kubectl get svc postgres -n public
NAME       TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)    AGE
postgres   ClusterIP   34.118.230.188   <none>        5432/TCP   3m41s
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$
```

```
kubectl get pvc -n public
```

```
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$ kubectl get pvc -n public
NAME           STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
postgres-pvc   Bound    pvc-104b59d0-7774-4f21-8936-44f3602a09a4   1Gi        RWO            standard       <unset>                 5m34s
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$
```

```
kubectl describe pod postgres-8549d8f6b9-62qpm -n public
```

```
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$ kubectl describe pod postgres-8549d8f6b9-hxqt2 -n public
Name:             postgres-8549d8f6b9-hxqt2
Namespace:        public
Priority:         0
Service Account:  default
Node:             gk3-nagp-assignment-demo-nap-kxe2abqm-2aff26da-9kvp/10.160.15.224
Start Time:       Tue, 05 Aug 2025 06:27:14 +0000
Labels:           app=postgres
                  pod-template-hash=8549d8f6b9
Annotations:      cloud.google.com/cluster_autoscaler_unhelpable_since: 2025-08-05T06:24:07+0000
                  cloud.google.com/cluster_autoscaler_unhelpable_until: Inf
Status:           Running
SeccompProfile:   RuntimeDefault
IP:               10.85.128.18
IPs:
  IP:           10.85.128.18
Controlled By:  ReplicaSet/postgres-8549d8f6b9
Containers:
  postgres:
    Container ID:   containerd://0ded077816f701342010a006e34efc3c599b731b0f798f8b437748d1ca81f26c
    Image:          postgres:14
    Image ID:       docker.io/library/postgres@sha256:563a4985838fcb5ac2e60fd58a1055ceafa791665e75e18d236221af0d478a33
    Port:           5432/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Tue, 05 Aug 2025 06:27:29 +0000
    Ready:          True
```

7.3. Run insert script

kubectl apply -f postgres-initdb-configmap.yaml

```
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$ kubectl apply -f postgres-initdb-configmap.yaml
configmap/postgres-initdb created
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$
```

8. Deploy Java Spring Boot service

8.1. Application deployment

cd ../app/
kubectl apply -f nagpapp-deployment.yaml

```
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$ kubectl apply -f nagpapp-deployment.yaml
Warning: autopilot-default-resources-mutator:Autopilot updated Deployment public/nagpapp-deployment: defaulted unspecified 'cpu' resource for containers [nagpapp-container] (se
e http://g.co/gke/autopilot-defaults).
deployment.apps/nagpapp-deployment created
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$
```

kubectl apply -f nagpapp-service.yaml

```
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$ kubectl apply -f nagpapp-service.yaml
service/nagpapp created
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$
```

Check replicas are running:
kubectl get pods -l app=nagpapp -n public

```
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$ kubectl get pods -l app=nagpapp -n public
NAME                                READY   STATUS    RESTARTS   AGE
nagpapp-deployment-69c4f64b8b-g5j6d   1/1     Running   0          66s
nagpapp-deployment-69c4f64b8b-k87vl   1/1     Running   0          66s
nagpapp-deployment-69c4f64b8b-kv8sk   1/1     Running   0          66s
nagpapp-deployment-69c4f64b8b-rv5sl   1/1     Running   0          66s
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$
```

Service logs:
kubectl logs <container_id> -n public

```
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$ kubectl logs nagpapp-deployment-69c4f64b8b-rv5sl -n public

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/

 :: Spring Boot ::                (v3.4.5)

2025-08-05T06:32:35.503Z  INFO 1 --- [           main] com.k8s.devops.DbCheckApplication        : Starting DbCheckApplication v1.0.0 using Java 17.0.2 with PID 1 (/app/app.jar
started by root in /app)
2025-08-05T06:32:35.507Z  INFO 1 --- [           main] com.k8s.devops.DbCheckApplication        : No active profile set, falling back to 1 default profile: "default"
2025-08-05T06:32:39.868Z  INFO 1 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2025-08-05T06:32:40.094Z  INFO 1 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 205 ms. Found 1 JPA repository int
erface.
2025-08-05T06:32:42.898Z  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 8080 (http)
2025-08-05T06:32:42.964Z  INFO 1 --- [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2025-08-05T06:32:42.965Z  INFO 1 --- [           main] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache Tomcat/10.1.40]
2025-08-05T06:32:43.264Z  INFO 1 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2025-08-05T06:32:43.266Z  INFO 1 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 7576 ms
2025-08-05T06:32:44.872Z  INFO 1 --- [           main] o.hibernate.jpa.internal.util.LogHelper  : HHH000204: Processing PersistenceUnitInfo [name: default]
2025-08-05T06:32:45.027Z  INFO 1 --- [           main] org.hibernate.Version                    : HHH000412: Hibernate ORM core version 6.6.13.Final
2025-08-05T06:32:45.099Z  INFO 1 --- [           main] o.h.c.internal.RegionFactoryInitiator    : HHH000026: Second-level cache disabled
2025-08-05T06:32:46.107Z  INFO 1 --- [           main] o.s.o.j.p.SpringPersistenceUnitInfo      : No LoadTimeWeaver setup: ignoring JPA class transformer
2025-08-05T06:32:46.255Z  INFO 1 --- [           main] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Starting...
2025-08-05T06:32:46.957Z  INFO 1 --- [           main] com.zaxxer.hikari.pool.HikariPool        : HikariPool-1 - Added connection org.postgresql.jdbc.PgConnection@3c8758d1
2025-08-05T06:32:46.960Z  INFO 1 --- [           main] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Start completed.
2025-08-05T06:32:47.076Z  WARN 1 --- [           main] org.hibernate.orm.deprecation            : HHH90000025: PostgreSQLDialect does not need to be specified explicitly using
```

8.2. Apply Ingress

kubectl apply -f nagpapp-ingress.yaml

```
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$ kubectl apply -f nagpapp-ingress.yaml
ingress.networking.k8s.io/nagpapp-ingress created
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$
```

kubectl get ingress -n public

```
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$ kubectl get ingress -n public
NAME              CLASS    HOSTS   ADDRESS        PORTS   AGE
nagpapp-ingress   <none>   *       34.8.79.79     80      3m16s
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$
```

kubectl describe ingress nagpapp-ingress -n public

```
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$ kubectl describe ingress nagpapp-ingress -n public
Name:             nagpapp-ingress
Labels:           <none>
Namespace:        public
Address:          34.8.79.79
Ingress Class:    <none>
Default backend:  <default>
Rules:
  Host        Path  Backends
  ----        ----  --------
  *
              /   nagpapp:80 (10.85.128.20:8080,10.85.128.134:8080,10.85.129.6:8080 + 1 more...)
Annotations:  ingress.kubernetes.io/backends:
                {"k8s1-46e3de69-kube-system-default-http-backend-80-e3dbae4a":"HEALTHY","k8s1-46e3de69-public-nagpapp-80-d3fe9bbc":"HEALTHY"}
              ingress.kubernetes.io/forwarding-rule: k8s2-fr-5fq0apjc-public-nagpapp-ingress-304smf14
              ingress.kubernetes.io/target-proxy: k8s2-tp-5fq0apjc-public-nagpapp-ingress-304smf14
              ingress.kubernetes.io/url-map: k8s2-um-5fq0apjc-public-nagpapp-ingress-304smf14
              nginx.ingress.kubernetes.io/rewrite-target: /
Events:
  Type    Reason     Age                From                     Message
  ----    ------     ----               ----                     -------
  Normal  Sync       79s                loadbalancer-controller  UrlMap "k8s2-um-5fq0apjc-public-nagpapp-ingress-304smf14" created
  Normal  Sync       76s                loadbalancer-controller  TargetProxy "k8s2-tp-5fq0apjc-public-nagpapp-ingress-304smf14" created
  Normal  Sync       56s                loadbalancer-controller  ForwardingRule "k8s2-fr-5fq0apjc-public-nagpapp-ingress-304smf14" created
  Normal  IPChanged  56s                loadbalancer-controller  IP is now 34.8.79.79
  Normal  Sync       47s (x4 over 4m4s) loadbalancer-controller  Scheduled for sync
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$
```

will show an external IP under the ADDRESS column. This is the Load Balancer for nagp-app.

8.3. Verify Service by External IP, CMD:

curl http://34.8.79.79/

```
tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$ curl http://34.8.79.79/
[{"id":1,"name":"Alice","email":"alice@gmail.com"},{"id":2,"name":"John","email":"jhon@gmail.com"},{"id":3,"name":"Bob","email":"bob@gmail.com"},{"id":4,"name":"Ram","email":"r
am@gmail.com"},{"id":5,"name":"Shyam","email":"shyam@gmail.com"}]tejpratapkushawaha9@cloudshell:~/kubernetes/database (nagp-k8s-devops-467617)$
```

8.4. Verify Service by External IP, Browser:


```
[
  {
    "id": 1,
    "name": "Alice",
    "email": "alice@gmail.com"
  },
  {
    "id": 2,
    "name": "John",
    "email": "jhon@gmail.com"
  },
  {
    "id": 3,
    "name": "Bob",
    "email": "bob@gmail.com"
  },
  {
    "id": 4,
    "name": "Ram",
    "email": "ram@gmail.com"
  },
  {
    "id": 5,
    "name": "Shyam",
    "email": "shyam@gmail.com"
  }
]
```

8.5. Restart Service Deployment:

kubectl rollout restart deployment nagpapp-deployment -n public

```
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$ kubectl rollout restart deployment nagpapp-deployment -n public
deployment.apps/nagpapp-deployment restarted
tejpratapkushawaha9@cloudshell:~/kubernetes/app (nagp-k8s-devops-467617)$
```

9. Deploy Application Using Docker

9.1. Create a Docker network

docker login

```
tejpratap@IN-92R6C54:~/nagp-assignment$ docker login
Authenticating with existing credentials... [Username: tejpratap1]

i Info → To login with a different account, run 'docker logout' followed by 'docker login'


Login Succeeded
tejpratap@IN-92R6C54:~/nagp-assignment$
```

9.2. Create a Docker network

docker network create nagp-app-network-demo

```
tejpratap@IN-92R6C54:~/nagp-assignment$ docker network create nagp-app-network-demo
39bb5a2ecc221a3637f0f28c6c8dc572acc53e86cef9d8992be9ec61581cd563
tejpratap@IN-92R6C54:~/nagp-assignment$
```

9.3. Start PostgreSQL container

docker run -d --name nagp-postgres-demo --network nagp-app-network-demo -e POSTGRES_DB=k8s_devops -e POSTGRES_USER=postgres -e POSTGRES_PASSWORD=Welcome@123 -p 5432:5432 postgres:15

```
tejpratap@IN-92R6C54:~/nagp-assignment$ docker run -d --name nagp-postgres-demo --network nagp-app-network-demo -e POSTGRES_DB=k8s_de
vops -e POSTGRES_USER=postgres -e POSTGRES_PASSWORD=Welcome@123 -p 5432:5432 postgres:15
28f2ddafd7c72804a1988c46fabed93bea12cd4921fee72e88649ec92499ed62
tejpratap@IN-92R6C54:~/nagp-assignment$
```

9.4. Run your Spring Boot app container

docker run -d --name nagp-app-demo --network nagp-app-network-demo -e SPRING_DATASOURCE_URL=jdbc:postgresql://nagp-postgres-demo:5432/k8s_devops -e SPRING_DATASOURCE_USERNAME=postgres -e SPRING_DATASOURCE_PASSWORD=Welcome@123 -p 8080:8080 tejpratap1/nagp-app:latest

```
tejpratap@IN-92R6C54:~/nagp-assignment$ docker run -d --name nagp-app-demo --network nagp-app-network-demo -e SPRING_DATASOURCE_URL=j
dbc:postgresql://nagp-postgres-demo:5432/k8s_devops -e SPRING_DATASOURCE_USERNAME=postgres -e SPRING_DATASOURCE_PASSWORD=Welcome@123
-p 8080:8080 tejpratap1/nagp-app:latest
f227b94a3b5845be5136b5c8197a54372e1fac88a95bcb64bd10236bbdc3cd53
tejpratap@IN-92R6C54:~/nagp-assignment$
```

9.5. Verify Docker Network

`docker network ls`

```
tejpratap@IN-92R6C54:~/nagp-assignment$ docker network ls
NETWORK ID     NAME                   DRIVER    SCOPE
32fe427a729f   bridge                 bridge    local
d220405fbe17   host                   host      local
c557d7e5f804   my-app-network         bridge    local
580a1fbcf38e   nagp-app-network       bridge    local
39bb5a2ecc22   nagp-app-network-demo   bridge    local
7e13a954be07   none                   null      local
tejpratap@IN-92R6C54:~/nagp-assignment$
```

9.6. Docker Running container

`docker ps -a`

```
tejpratap@IN-92R6C54:~/nagp-assignment$ docker ps -a
CONTAINER ID   IMAGE                          COMMAND                  CREATED         STATUS                     PORTS
                     NAMES
f227b94a3b58   tejpratap1/nagp-app:latest     "java -jar app.jar"      3 minutes ago   Up 3 minutes               0.0.0.0:8080->8080/t
cp, [::]:8080->8080/tcp   nagp-app-demo
28f2ddafd7c7   postgres:15                    "docker-entrypoint.s…"   4 minutes ago   Up 4 minutes               0.0.0.0:5432->5432/t
cp, [::]:5432->5432/tcp   nagp-postgres-demo
```

9.7. Run your Spring Boot app container

`Docker logs -f nagp-app-demo`

```
tejpratap@IN-92R6C54:~/nagp-assignment$ docker logs -f nagp-app-demo

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/

 :: Spring Boot ::                (v3.4.5)

2025-08-04T16:02:20.106Z  INFO 1 --- [           main] com.k8s.devops.DbCheckApplication        : Starting DbCheckApplication v1.0.0
using Java 17.0.2 with PID 1 (/app/app.jar started by root in /app)
2025-08-04T16:02:20.109Z  INFO 1 --- [           main] com.k8s.devops.DbCheckApplication        : No active profile set, falling back
 to 1 default profile: "default"
2025-08-04T16:02:21.518Z  INFO 1 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repos
itories in DEFAULT mode.
2025-08-04T16:02:21.619Z  INFO 1 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository sca
nning in 85 ms. Found 1 JPA repository interface.
2025-08-04T16:02:22.451Z  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 8080 (
http)
2025-08-04T16:02:22.471Z  INFO 1 --- [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2025-08-04T16:02:22.471Z  INFO 1 --- [           main] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache To
mcat/10.1.40]
2025-08-04T16:02:22.501Z  INFO 1 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApp
licationContext
2025-08-04T16:02:22.502Z  INFO 1 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initial
ization completed in 2317 ms
2025-08-04T16:02:22.990Z  INFO 1 --- [           main] o.hibernate.jpa.internal.util.LogHelper  : HHH000204: Processing PersistenceUn
itInfo [name: default]
2025-08-04T16:02:23.092Z  INFO 1 --- [           main] org.hibernate.Version                    : HHH000412: Hibernate ORM core versi
```

9.8. Stop all running containers

`docker stop f227b94a3b58`

```
                              sharp_boyd
tejpratap@IN-92R6C54:~/nagp-assignment$ docker stop f227b94a3b58
f227b94a3b58
tejpratap@IN-92R6C54:~/nagp-assignment$
```