1. Deploy a **kubernetes cluster** using KIND, Minikube, or Public Clouds

2. Deploy a **DB cluster** on K8s with **persist** data (MySQL or MariaDB).

3. Deploy a **Web Server** on K8s (Nginx, Apache, …) with the following conditions:
    a. use **multiple replicas** of the web-server pods
    b. The web-page should be accessible from the browser.
    c. Custom configuration of the webserver should be **mounted** and used in the pod.
    d. The web-page should:
        i. Show the Pod IP.
        ii. include a field called **"serving-host"**. This field should be modified in an **init container** to be "Host-{the last 5 character of the web-ceerver pod name}"

        For EX. web-server pod name is web-server-7f89cf47bf-25gxj the web-page should show: serving-host=Host-5gxj

4. Add a policy to only allows the web server pods to initiate connections to the database pods on the correct port (e.g., 3306 for MySQL). All other traffic to the database should be denied.
5. Add autoscaling HPA to webserver based on CPU/memory or custom metrics.
6. Demonstrate the HPA working. You can do this by using a simple load-testing tool.
7. Suggest a Disaster recovery solution for the DB.
8. Write a **Golang applications** for monitoring the status of the above pods. This application should print a log at any point in time if there is any of the following changes:
    a. A new pod has been created
    b. pod has been deleted
    c. pod has been updated

9. Use the Helm chart to deploy all above components

Deliverables:

1. Design of the internal and external connections.
2. Helm-charts
3. Source code Golang Applications
4. Dockerfiles
5. Access to the cluster preferably or a working demo