

Introduction

You are an Openshift or Kubernetes engineer at ProjectPhoenix Innovations, a leading software development firm specializing in cloud-native solutions. Your task is to demonstrate your expertise in Kubernetes and OpenShift by deploying a containerized application for the company's new project, "Project Phoenix." This application will be critical in showcasing the company's capabilities in delivering resilient, scalable, and efficient cloud-native applications.

Assignment Details

1. Deploy an OpenShift Cluster

- Objective: Deploy an OpenShift cluster. You may choose a managed OpenShift, local/MicroShift, or a standard Kubernetes cluster.
- Deliverables:
 - A concise description or presentation (200 words max) of the setup process, including your choice's rationale.
- Time Estimation: 0.5-2 hours, depending on your choice.

2. Application Deployment and Management

- Objective: Deploy and manage the containerized application provided in the following repository: <https://github.com/nordcloud/openshift-assignment/ohb>.
- Requirements:
 - Set up a CI/CD or GitOps pipeline for the application deployment.
 - Ensure the application is resilient to failures.
 - Create two separate deployments: one for testing and another for production.
- Deliverables:
 - Hyperlink to the running application.
 - Detailed commands and actions used for the deployment and exposure of the application, including any troubleshooting steps.
 - Time Estimation: 5-15 minutes.

3. Custom Container Deployment

- Objective: Run the container `docker.io/httpd` and modify its output web page to display your name.
- Deliverables:
 - Hyperlink to the modified application.
 - Exact commands and actions for creating and exposing the application, along with troubleshooting steps and best practices for running this container on OpenShift/Kubernetes.

- Time Estimation: 10-15 minutes.

4. Showcase Your Expertise

Here are some example assignments that you can choose from, pick one or two and showcase to us how it can help or improve the Project Phoenix .

Advanced Deployment Strategies:

Assignment: Implement a blue-green deployment strategy for an application. Document the steps and rationale behind your strategy, including how you would manage traffic and rollback if necessary.

Monitoring and Logging:

Assignment: Set up a monitoring and logging solution using Prometheus and Grafana (or similar tools). Create a dashboard to monitor key metrics of the Kubernetes cluster and the deployed application.

Security Practices:

Assignment: Harden a Kubernetes/OpenShift cluster. Implement RBAC, configure network policies, and securely manage secrets. Provide a report detailing each security measure implemented.

Resource Optimization:

Assignment: Optimize resource allocation for a set of running applications. Use resource limits, requests, and Horizontal Pod Autoscalers. Demonstrate the effectiveness of your optimization through metrics.

Stateful Applications Management:

Assignment: Deploy a stateful application using StatefulSets. Include persistent storage and demonstrate how to handle data persistence and volume management.

Disaster Recovery and High Availability:

Assignment: Design and implement a disaster recovery plan for an application. This should include strategies for data backup, high availability setup, and a failover mechanism.

Networking Concepts:

Assignment: Configure and demonstrate a Kubernetes networking solution. Include service discovery, Ingress/Egress controls, and demonstrate how network policies are applied.

Custom Resource Definitions (CRDs) and Operators:

Assignment: Create a Custom Resource Definition (CRD) and a simple Operator to manage a custom resource lifecycle in a Kubernetes cluster.

Serverless Deployments:

Assignment: Deploy a serverless function using Knative or OpenShift Serverless, and demonstrate scaling based on demand.

Service Mesh Implementation:

Assignment: Implement a basic service mesh using Istio or Linkerd. Showcase traffic management, secure service-to-service communication, and observability features.

Tips

- Do not overthink; pick what you are familiar with
- If you encounter issues and it takes more time than intended, provide a brief description of your troubleshooting steps and how it could be solved.

Evaluation Criteria

- Knowledge shown on Kubernetes and OpenShift
- Effectiveness in deploying and managing the application.
- Efficiency and innovation in the CI/CD or GitOps setup.
- Resilience and scalability of the application in both testing and production environments.
- Clarity and comprehensiveness of documentation and presentation

Submission and questions

To submit your work create a private Github repository and add the Github users that are provided to you by the recruiter. If there are any questions that arise from this assignment please contact the recruiter who will help with answering those questions.