Devops Intern Hiring Task

Submission Date - **11:59 pm IST - Friday 6th December, 2024**

Time to Learn and Solve - **4-6 hours**

Problem Statement

You are required to (learn and) build Kubernetes and Helm resources in this task. The goal of this assignment is to assess your **fast learning skills** and **ability to debug and grasp concepts faster**, with your understanding of Kubernetes concepts and your ability to use Helm for managing Kubernetes applications. By the end of this assignment, you will have deployed a complete application stack using Kubernetes and Helm.

Part 1: Learn Kubernetes Basics

1. Learn the basics of Kubernetes, including:
    - **Pods**: Understand how they represent the smallest deployable units in Kubernetes.
    - **Deployments**: Learn how to manage scalable and self-healing applications.
    - **Services**:
        - **ClusterIP**: Internal communication within the cluster.
        - **NodePort**: Expose services externally.
    - **ConfigMaps**: Use them to attach environment variables for application code.
    - **Helm Charts:** Learn about the basics of Helm Charts
    - ***Do not spend time in learning anything more than what is needed.***

Part 2: Make a Helm Chart

1. Your goal is to create a Helm Chart (name: **cosmocloud-deploy**) to deploy all our applications together - 1 Backend, 1 Frontend, 1 Redis DB.
    a. Each of the services should be deployed as a **Deployment** with 1 scale replica.
    b. Backend image - **shreybatra/sample-backend**
        i. Pass the env variable `REDIS_URI` as redis://redis-svc:6379
    c. Frontend image - **shreybatra/sample-frontend**
        i. Pass the env variable `BACKEND_URL` as http://backend-svc:8000
    d. Redis image - **redis**
2. Expose the applications using services
    a. Backend service
        i. Type - ClusterIP
        ii. Name - **backend-svc**
        iii. Port - 8000
    b. Frontend service
        i. Type - NodePort
        ii. Name - **frontend-svc**

        iii. Port - 5173
        iv. ***NodePort - 31000***
    c. Redis service
        i. Type - ClusterIP
        ii. Name - **redis-svc**
        iii. Port - 6379
3. Namespace - **default**

We will be automatically pulling your github repo and automatically judging your submission. Make sure you have a top level folder (Helm Chart folder) in your Github repo named `***cosmocloud-deploy***`

We will be using this command to deploy and test your application -

helm install testapp cosmocloud-deploy –atomic –timeout 30s

Judging Criteria

You will be judged **(by automatic scripts)** on the correctness of your solution, by testing your solution in a Live Running environment. We will check -

- Correct deployment - 1 instance of backend, 1 of frontend, 1 of redis.
- Correct service types deployed
- Correct environment variables sent in each of the Deployment.
- Quality of submission – Helm files, formatting, explanation in Readme file, etc.

The next interview will be based on your submission – you should know exactly how things are running and why.

How to submit

Once done, make sure your Helm Charts are uploaded on your github repo. Make sure the github repo is public or access is given to username: **shreybatra**

You can then go ahead and submit your application via this form, **on or before 11:59 PM IST on 6th December, 2024**.