**Practical DevOps– Assignment on AWS**

**Prerequisites**

1. Need to have AWS with the necessary permissions and access credentials.
2. GitHub account.

**Assignment submitting.**

We will provide you with a folder named practical\_pi\_sharp. You need to:  
- Create a sub folder inside it with your staff code's name + your name. (ex: sd0660-lecao)

* Inside your staff code folder, put a text file with the following information:
  + Your full name: A Nguyen Van
  + Your Staff Code: SD0660
  + Link of your GitHub repositories

Notes: Inside the application repository, you need to have a folder named 'Document.' In the 'Document' folder, include a Word file or video recording that captures your performance.

**Deadline**

For those who passed Practical DevOps for Devs: 09/11/2023

For others: 15-11-2023

1. **Overall structure**

You will be provided with a CI/CD diagram outlining the pipeline's high-level structure. Your task is to follow the diagram and set up the CI/CD pipeline based on the tools and configurations specified.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Assignment Tasks**

1. **Manage K8s Application**

DevOps for Devs:  
You would need to have a repository for the MSA application.   
You can clone the source code from <https://github.com/nashtech-garage/kubernetes/tree/master/src>

App repository pattern: [https://github.com/<your-github-name>/<staffcode\_msa](https://github.com/%3cyour-github-name%3e/%3cstaffcode_msa)>

Example: https://github.com/hoanglecao/sd0660\_msa  
You also need another repository for infrastructure. Provision codes and manifest files will be stored in this repository.  
App repository pattern: [https://github.com/<your-github-name>/<staffcode\_aws\_infrastructure](https://github.com/%3cyour-github-name%3e/%3cstaffcode_aws_infrastructure)>

Example: <https://github.com/hoanglecao/sd0660_aws_infrastructure>

DevOps for Devs:

Create K8s manifest.

Pi-sharp:  
Use helm-charts to manage K8s manifest.

1. **Provision AWS resources**

Use <https://github.com/nashtech-garage/terraform-demo> as a reference.Use Terraform to provision VPC, EC2, ECR and EKS on AWS  
DevOps for Devs:

* EKS can created in one zone and
* Can use default VPC for worker nodes

Pi-sharp:

* EKS must be created in High Availability (Use Multi-AZs)
* Create your VPC and use it for worker nodes

1. **Installation**

You need to have Docker and Jenkins servers installed on EC2.

1. **Setup Jenkins pipeline for CI**

Use <https://github.com/nashtech-garage/devops-ci-cd>as a reference.

DevOps for Devs:

* You can use normal Jenkins pipeline or create Jenkins pipeline using Groovy.

Pi-sharp:

* Use Trivy in Jenkins CI pipeline.

1. **Monitoring**

DevOps for Devs:Set upPrometheus and Grafana to monitor EKS resources and default EKS resource metric.

Pi-sharp:

Use Istio to provide observability metrics, which can be visualized and collected using Grafana and Prometheus.

1. **Use GitOps for the CD pipeline.**DevOps for Devs:

Install Argo CD on your EKS Cluster

Create an application in Argo CD (you can use GUI or CLI)

Pi-sharp:

* Install Argo CD Image Update (https://argocd-image-updater.readthedocs.io/en/stable/install/installation/). This will help you listen to changes the image on ECR and update the image tag in Ops repo.
* Use Blue/Green strategy for deployment.