Capstone Project

DevOps Engineer Masters Program

TABLE OF CONTENTS

BACKGROUND

Supertech is an upcoming enterprise contact management portal that is trying to make its mark in the already crowded market. To make themselves relevant, they need to be up to mark with all the desired features with minimal time to market. To achieve the same, it is decided that the portal will be developed using agile with Devops. This should help the team achieve the shortest time to market and incorporate the prioritized features in the product.

BUSINESS CONTEXT

The Objective is to develop a Continuous deployment pipeline to deploy the code changes as they are committed to codebase. The codebase has to be tested thoroughly for any regression/missing features before deployment.

The pipeline should be built in a way that ensures lower maintenance efforts and maximum re-usability.

USE CASES

The pipeline architecture should support the following usage scenarios:

On-Premise:

a. *Deploy Infrastructure:* Deploy the Underlying services (Jenkins, git, Docker daemon etc.) using Puppet/Ansible

b. *Continuous delivery*: any code commit should be subjected to all the standard phases of CI and CD.

c. *Continuous monitoring*: Implement monitoring of the entire infrastructure of your DevOps implementation to make sure that the systems are running properly, and all the system data is preserved for future.

d. *Manage On-Premise infrastructure*: Infrastructure should be configured as code using IaC principles. The IaC code sould be version controlled and any changes shall automatically be deployed to the infra (ref Step a).

On-Cloud:

- *Deploy DevOps services on cloud (AWS):* Deploy your entire pipeline on AWS using Native AWS services.
- *Continuous Delivery*: Using AWS CodeCommit, CodeBuild and CodeDeploy, Target Elastic Beanstalk
- *Continuous Monitoring*: Using ClowfWatch (Alarms, Agent etc.), Implement monitoring of the entire infrastructure of your devops implementation to make sure that the systems are running properly and all the system data is preserved for future.
- *OnDemand Deployment*: Implement one-click deployment using CloudFormation
- *Manage-Costs*: Define a strategy to minimize cost with high availability of the Application

PROCESS FLOW

Suggested Approach

The following concepts need to be applied in the project:

- Use Address book web application for the implementation of the whole DevOps pipeline:

  Github repository path for the project: https://github.com/edureka-git/DevOpsClassCodes.git
- Continuous Delivery: any code commit shall be subjected to:
  - Static code analysis (any preferred tool can be used)
  - Testing: Unit test
    - Should be performed on a temporary Docker container
    - Use Maven targets
  - Compilation: check for build breaks
    - Using Maven targets
  - Deploy to test
    - Deploy using a docker container

        - Test the deployment
    - System UAT test
        - Using selenium WD on the compiled and deployed target
    - If UAT is passed:
        - Tag the codebase with a release label
    - Implement 1 click deployment of any release label
        - Implement using Puppet
        - Use manifests, reuse classes and modules
- Continuous Monitoring:
    - Use Nagios
    - Add the below hosts for monitoring:
        - Jenkins master
        - Build slave
        - Docker host
    - Enable monitoring of below resources on all hosts
        - CPU
        - HDD
        - RAM
    - Enable monitoring of Host specific monitors
        - Jenkins master: URL, Jenkins service status
        - Build slave: Slave connection status
        - Docker host: Docker daemon, running containers
- For Cloud, setup AWS native services as below and deploy the pipeline:
    - CloudFormation for 1 click deploy of infra
        - Code Commit
        - Code Build/Pipeline/Deploy target to Beanstalk
        - ECS cluster
    - Pipeline
        - Create entire CI/CD pipeline (like on Premise) on AWS

- Setup cloudwatch to send an alert whenever any EC2 instance uses more than 30% CPU
- SNS for email notifications

TARGET ENVIRONMENT

The installation of the tools required to work on the capstone project is already covered in the courses that are part of this masters program. Install these tools and solve the project problem. Submit your answer to Edureka Support team for review.

On-Premise: Edureka Provided VMs

On-Cloud: AWS (Setup your Own Lab)

TASKS

Following are the tasks, which need to be developed while executing the project:

1. Complete Jenkins installation folder (using thin Backup)
2. Puppet created Manifests
3. Ansible Playbooks
4. CodeCommit repository and CodeBuild pipelines
5. Cloudformation deployment file in JSON format
6. Screenshots of deployment