DevOps Certification Training

Certification Project – Insure Me

Insurance Domain

Insure Me is a Global leading Insurance provider based out of USA. The company offers products and services like Home Insurance, Health Insurance, Car Insurance and Life Insurances. Initially the company was using a Monolithic application architecture, As the company grown, It started facing difficulties in managing the application infrastructure and application deployments.

Insure-Me has decided to transform its monolithic application architecture to microservice application architecture and opted to go DevOps by implementing CICD pipeline and necessary automations. Insure me has decided to use AWS as primary cloud services provider to create servers, databases, and application deployments.

The company's goal is to deliver the product updates frequently to production with High quality & Reliability. They also want to accelerate software delivery speed, quality and reduce feedback time between developers and testers.

Following are the problems the company is facing at the moment

- ✓ Building Complex builds is difficult
- ✓ Manual efforts to test various components/modules of the project
- ✓ Incremental builds are difficult to manage, test and deploy
- ✓ Creation of infrastructure and configure it manually is very time consuming
- ✓ Continuous manual monitoring the application is quite challenging.

In order to implement a POC, you are requested to develop a mavenized microservice using spring boot and in memory h2 database.

1. a microservice which exposes below mentioned endpoints as APIs and uses in memory h2 database to store the data.

   a. /createPolicy (HTTP Method : POST) (Request Body : JSON)
   b. /updatePolicy/{policy id} (HTTP Method : PUT) (Request Body : JSON)
   c. /viewPolicy/{policy id} (HTTP Method : GET) (No Request Body)
   d. /deletePolicy/{policy id} (HTTP Method : DELETE) (No Request Body)

2. Write necessary Junit testcase.
3. Generate HTML report using TestNG.
4. Push your code into your GitHub Repository.

Note : Preload some data into the database.

Later, you need to implement Continuous Integration & Continuous Deployment using following tools:

- ✓ Git - For version control for tracking changes in the code files
- ✓ Jenkins - For continuous integration and continuous deployment
- ✓ Docker - For deploying containerized applications
- ✓ Ansible - Configuration management tools
- ✓ Selenium - For automating tests on the deployed web application
- ✓ AWS : For creating ec2 machines as servers and deploy the web application.

This project will be about how to test the services and deploy code to dev/stage/prod etc, just on a click of button.

Business challenge/requirement

As soon as the developer pushes the updated code on the GIT master branch, the Jenkins job should be triggered using a GitHub Webhook and Jenkins job should be triggered, The code should be checked out, compiled, tested, packaged and containerized and deployed to the preconfigured test-server automatically.

The deployment should then be tested using a test automation tool (Selenium), and if the build is successful, it should be deployed to the prod server. All this should happen automatically and should be triggered from a push to the GitHub master branch.

The solution is available at below repository, use it to validate the solution with the solution available at - **https://github.com/StarAgileDevOpsTraining/star-agile-insurance-project.git**

Note : To have a detailed information about running the application and exposed APIs, Input/Output format, Refer to the README.md in the GitHub repository.