DevOpsSchool

Lets Learn, Share & Practice DevOps

MASTER IN MICROSERVICES

Level - Intermediate

FEBRUARY 20, 2021

DEVOPSSCHOO.COM

Bangalore, India

COTOCUS

TRAINING DATES	TBD
TIME	TBD
TRAINER NAME	RAJESH KUMAR
TRAINER CV	HTTP://RAJESHKUMAR.XYZ/CV

COURSE INTRODCUTUION

Introduction of Microservices

The microservice architecture enables the rapid, frequent and reliable delivery of large, complex applications. It also enables an organization to evolve its technology stack.

Microservices also known as the microservice architecture - is an architectural style that structures an application as a collection of services that are

- Highly maintainable and testable

- Loosely coupled

- Independently deployable

- Organized around business capabilities

- Owned by a small team

PREREQUISITE OF THIS COURSE

- Knowledge of SDLC
- AWS Fundamental
- Linux Fundamental
- Fundamental concept on any one programming language

COURSE OVERVIEW AND OBJECTIVES

This course has been divided into 5 days course. Object of the course is provide experience for software development in Microservices software development architecture. You can find the detail of outcome as below;

Planning Phase

Discussion about automating students records challenges of DevOpsSchool.com which leads to writing a software for it. As part of Software development, discussion about various Software architecture, Software development model, software running and hosting platforms etc. discussion on DevOps, DevSecOps, SRE, Agile, microservice, TDD concepts, principal, practices and tool sets.

Small Project Requirement which includes 3 micro service, Login, Registration & Students records CRUD operations for devopsschool.com students.

Tools Used - Jira, Github, DevOpsSchool Tools

Coding & Testing Phase

Option 1: JAVA - Write a code for "devopsschool-student-login", "devopsschool-student-registration" and"devopsschool-student-records" Project including Method --> Classes -> Interface using Core Java. Complete Demo Project using HTMl - CSS - JS - MySql. Convert all code into into SpringBoot project.

Option 2: Python - Write a code for "devopsschool-student-login", "devopsschool-student-registration" and"devopsschool-student-records" Project using Python. Complete Demo Project using HTMl - CSS - JS - MySql. Convert all projects into Flask or django framework.

Write some unit test using junit/unittest and UI test using selenium. Build tools should be gradle or Maven. Packaging into war file and archive the package into Artifactory and setup a CI for Build and Testing using Jenkins including deployment and config management using Ansible.

Tools Used - Git, Github, Visual Studio Code, Java, SpringBoot, Python, Flask, django, Junit, Gradle, Maven, Selenium, Jenkins, Ansible, SonarQube

Container Migration Phase

Migrate existing projects into containers. Write a dockerfile and build docker images and test the entire components of for "devopsschool-student-login", "devopsschool-student-registration" and"devopsschool-student-records". Validate entire application stack using docker-compose setup.

Tools Used - Docker and Docker Compose

Production & Ops Phase

Write a helm package of for "devopsschool-student-login", "devopsschool-student-registration" and "devopsschool-student-records" and deploy into kubernetes cluster. Setup kubernetes services and ingress with external load balancers. Configure to DNS services. Setup service proxy (Envoy), service networking (Istio) and discover services (Consul).

Tools Used - Kubernetes, Helm, Istio, Envoy, Consul

Setting up full stack Observability

Setup complete Observability using Datadog (Infra), NewRelic (APM) and Splunk (LOG).

Tools Used - Datadog, NewRelic, Splunk, Kibana

AGENDA

DAY 1 AGENDA

- Evolution of Software Development, Model and Practices
- Introduction of Microservices
- Evolution of Microservices
- Microservices Architecture
- Microservices Design Principles
- Microservices Design Approach
- Microservices Messaging & Communication
- Microservices Deployment patterns
- Microservices Service discovery
- Microservices Security
- Microservices Performance
- Microservices Reliability
- Microservices Observability
- Technology for Microservices
- Microservices Hosting Platforms
- Microservices Automation Tooling

- Migration & Transformation to Microservices
- Software Developement Tool Sets
- Overview of Git
- Overview of Jira, Github
- Overview of Core Java
- Overview of SpringBoot
- Overview of Python
- Overview of Flask and django
- Overview of mySql
- Overview of Junit and Selenium
- Overview of HTMl, CSS and Js.
- Overview of Maven, Gradle, Apache, Tomcat
- Overview of Ansible
- Overview of Jenkins
- Overview of Docker
- Overview of Kubernetes
- Overview of Datadog
- Overview of NewRelic
- Overview of Splunk
- Overview of envoy
- Overview of Istio
- Overview of Consul

Project - Demo - Lab: Discussion about automating sudents records challeneges of DevOpsSchool.com which leads to writing a software for it. As part of Software development, discussion about various Software architecture, Software development model, software running and hosting platforms etc. discussion on DevOps, DevSecOps, SRE, Agile,microservice, TDD concepts, principal, practices and tool sets. Small Project Requirement which includes 3 microservice, Login, Registration & Students records CRUD operations for devopsschool.com students.

DAY 2 AGENDA

Write a Code for one of the following option…

Option 1: JAVA - Write a code for "devopsschool-student-login", "devopsschool-student-registration" and"devopsschool-student-records" Project including Method --> Classes -> Interface using Core Java. Complete Demo Project using HTMl - CSS - JS - MySql. Convert all code into into SpringBoot project.

Option 2: Python - Write a code for "devopsschool-student-login", "devopsschool-student-registration" and"devopsschool-student-records" Project using Python. Complete Demo Project using HTMl - CSS - JS - MySql. Convert all projects into Flask or django framework.

Write some unit test using junit/unittest and UI test using selenium. Build tools should be gradle or Maven. Packaging into war file and archive the package into Artifactory and setup a CI for Build and Testing using Jenkins including deployment and config management using Ansible.

- Demo and Lab - Planning using Jira and Github
- Demo and Lab - Coding project using SpringBoot or Python
- Demo and Lab - Code Versioning and Sharing using Git using Github
- Demo and Lab - Code Analysys using SonarQube
- Demo and Lab - Code Build using Maven and Gradle
- Demo and Lab - Code Testing using Junit and Selenium
- Demo and Lab - Code Deployment using Ansible
- Demo and Lab - Setting up CI using Jenkins

DAY 3 AGENDA

- Overview of Docker
- Docker Architecture
- Docker Installing and Configuration
- Working with Containers
- Working with Dockerfile and Docker images
- Working with Docker Registry
- Working with Docker Compose

Project - Demo - Lab:

Migrate existing projects into containers. Write a dockerfile and build docker images and test the entire components of for "devopsschool-student-login", "devopsschool-student-registration" and"devopsschool-student-records". Validate entire application stack using docker-compose setup.

DAY 4 AGENDA

- Overview of Kubernetes
- Kubernetes Architecture
- Kubernetes Installing and Configuration
- Working with Pods
- Working with Deployment
- Working with Services
- Working with Ingress
- Working with Volume
- Working with Helm

Project - Demo - Lab: Write a helm package of for "devopsschool-student-login", "devopsschool-student-registration" and "devopsschool-student-records" and deploy into kubernetes clustor. Setup kubernetes services and ingress with exernal loadbalancers.

DAY 5 AGENDA

- Overview Envoy
- Setup and Configure Envoy
- Overview Istio
- Setup and Configure Istio
- Overview Consul
- Setup and Configure Consul
- Overview Datadog
- Setup and Configure Datadog
- Monitoring and Alerting Infra with Datadog

- Overview NewRelic
- Setup and Configure NewRelic
- Monitoring and Alerting APM with NewRelic
- Overview NewRelic
- Setup and Configure Splunk
- Monitoring and Alerting Log with Splunk

Project - Demo - Lab: Setting up full stack Observability and service discovery and networking. Configure to DNS services. Setup service proxy (Envoy), service networking(Istio) and discover services(Consul). Setup complete Observability using Datadog (Infra), NewRelic (APM) and Splunk (LOG).