**Table of Contents :   Microservices – L2**

**Pre-Requisites (if any)** : Prior knowledge and understanding of following

1. Microservices – L1

**Microservices – L2**

|  |  |  |  |
| --- | --- | --- | --- |
| Sl No | Topic | Sub-Topic | Approx. Duration (Hrs.) |
| 1 | Introduction to important Patterns in Microservices | Decomposition Patterns  -Decompose by Business Capability  -Decompose by Subdomain  -Strangler Pattern  Integration Patterns  -API Gateway Pattern  -Aggregator Pattern  -Client-Side UI Composition Pattern  Database Patterns  -Database per Service  -Shared Database  -Command Query Responsibility Segregation (CQRS)  -Saga Pattern  Observability Patterns   * Log Aggregation * Performance Metrics * Distributed Tracing * Health Check   Cross-Cutting Concern Patterns   * External Configuration * Service Discovery Pattern * Circuit Breaker Pattern * Blue-Green Deployment Pattern | 2 Hrs |
| 1 | Centralized Configuration using Spring Cloud Config Server and Client | Need for having configuration in a centralized place  Configuring Spring Boot Application to act as Spring Cloud Config Server  Configuring Spring Boot Application to acts as Spring Cloud Config Client  Git repo for maintaining configuration  Client application getting config information from Git repo, through the Config server  If configuration changes in Git repo, how client obtains the changed information | 2 Hrs. |
| 2 | Routing and Filtering using Netflix Zuul API Gateway | Discuss the use of a Zuul proxy in microservices development to create more fluid, robust design that is less prone to errors  Netflix Zuul as an API gateway or Edge service  Configuring MS as API gateway (Edge Service) to put common aspects like CORS, authentication, and security  Implementing routing rules and filtering | 2 Hrs. |
| 3 | Spring Cloud Gateway | Spring Cloud Gateway Features  Comparing Spring Cloud Gateway with Zuul API Gateway  Demo of building gateway using Spring Cloud Gateway. Configuring routes to process requests to downstream services | 2 Hrs. |
| 4 | Distributed Log Tracing in Microservices using Spring Cloud Sleuth | Introducing distributed Log Tracing  What is Spring Cloud Sleuth?  Implementing Spring Cloud Sleuth based application  Deploying the App on Pivotal Cloud Foundry(PaaS) and view logs  Understanding TraceID and SpanID | 3 Hrs. |
| 5 | Feign REST Client | Introduction to Feign REST Client (- REST Consumer)  How is it used?  Example of Microservices Application with Feign Client | 3 Hrs. |
| 6 | Microservices security using Spring Security (OAuth2) / JSON Web Token (JWT) | Spring Security in a Spring Boot Application  What is OAuth2?  How does OAuth2 work?  Spring Security using OAuth2 in Spring Boot  What is JWT?  JWT v/s OAuth  Spring Security with JWT in Spring Boot App | 4 Hrs. |
| 7 | Store and Load Microservices secrets using Hashicorp Vault | Managing secrets using Hashicorp vault  Installing Hashicorp vault  Starting up Hashicorp vault server in development Mode  Creating and storing secrets in Hashicorp vault  Securing Spring Applications with Hashicorp vault | 2 Hrs. |
| 8 | Transactions across Microservices | Introduction  Avoiding Transactions across microservices  Alternative approach without transactions  Two-Phase Commit protocol (2PC)  Architecture of 2PC  XA Standard  REST-AT Standard draft  Eventual consistency and Compensation | 2 Hrs. |