**SPRING MicroServices**

**Pre- requisites :**

**Participants should know following modules:-**

1. **Core Java**
2. **Spring Boot**
3. **Spring REST**

**DAY 1:**

**Spring Microservice Concepts**

* Monolithic Application vs Microservices
* Pros and Cons of Microservices
* Microservices Infrastructure – PAAS[ Platform as a Service]
* Microservices Challenges
* Spring Cloud umbrella
* Using Spring Boot to create a Microservice
* Create 2 microservices
* RestTemplate to invoke a service

**DAY 2:**

* Microservice return List
* Microservice return a wrapper around the list
* Understand the advantages of wrapper over List
* Service Discovery and Registration
  + Need of Eureka Server
  + Add Eureka dependency
  + Service Registration with Eureka
  + Discover Service
  + Dynamic Url
  + Rum multiple instances of service
  + Load Balancing
  + @EnableEurekaClient
  + @EnableEurekaServer

**DAY 3:**

* Microservice communicate with 3rd party API
* 4 services communicate with each other
* FeignClient
  + Declarative REST Client
  + Add Feign Dependency
  + @EnableFeignClients
  + @FeignClients

**DAY 4:**

* Circuit Breaker
  + Understand Circuit Breaker pattern
  + Fallback methods for delayed or no response
  + Add hystrix dependency
  + @EnableHystrix
  + @HystrixCommand
* Zuul Proxy Gateway
  + API Gateway
  + @EnableZuulProxy

**DAY 5:**

* Config Server and Client
  + Centralized application to manage all application related configuratios
  + Add dependency
  + Create a Config Server
  + @EnableConfigServer
  + Create a GIT repository
  + Create a Config Client
* Swagger UI
  + Tool for visualizing API’s
  + Describe your REST API
  + Add dependency
  + @EnableSwagger2
  + Swagger annotations
    - @ApiOperation
    - @ApiResponses
    - @ApiParam
    - @ApiModel
    - @ApiModelProperty