Date: 26/04/2021
Spring Boot 9AM
Mr. RAGHU

Microservices Design Overview

Microservices:
=> An Independent deployable component.
=> A Decoupled Architecture
=> Module as a project / Service as a Application
=> Independent services makes resource utilization
and communication using webservices.

==========

Monolithic App : All Modules together as a single project
Microservices : One Module as one porject and interact
using webservices.

*) Advantages:
=> Effective Resource Utilization
=> Parallel implementation
=> CI/CD is easy.
=> Less Cohesion. So, one module problems may not effect
other modules. No need of re-build and deploy all
project even on one service change.

=> Adding new concept/module/service needs another instace
and integration also easy(webservices)
=>

*) Limitations in Microservices(MS#)
=> Maintainance (Multiple server/ instance/ log files/deployments)
=> Debug process (Find Bugs and execution flow)
=> Testing Process.

==========

Spring Cloud [Netflix]- Components

1. Register and Discovery Server --- Eureka Server
2. Intra Communication Clients --- DiscoveryClient
LoadBalancerClient**
FeignClient**
3. Faulttolerance/Circuit Breaker -- Hystrix with Dashboard
4. Log Trace and Visualize -- ELK (Elastic Search, Logstash, Kibana)
Zipkin and Sleuth
5. Message Queues -- ActiveMQ, Apache kafka
6. Production Support Endpoints -- Actuator + Admin UI
7. Config Server(common properties) -- Git Accounts(GitHub/Gitlab)
8. API Gateway -- Zuul
9. Security -- JWT, OAuth(FB,GOOGLE,..)

==========

SOA : Service Oriented Architecture

This design is used to link two applications which are running
in different places/servers.

Components:-

a. Service Provider : Provider application contains logic
to give actual service. Ex:Payment Service, Email Service,
Message Service...etc

b. Service Consumer : It contains logic to make HTTP call
to service provider that gets services executed and output.

c. Service Register and Discovery
It contains service provider information
Where it is exist(IP,PORT)
Load Details
Logical information(serviceName, instanceId, common PATH)

it will hold service information and provides on request
to consumer.

Operations:-

1. PUBLISH : Service Provider code is called as Skeleton
this code is converted into Document.
Placed inside Registery server.
2. FIND : Consumer able to detect the service.
3. BIND : Consumer code(Stubs) makes request and gets reponse

---