**What is microservice:-**

Microservice is a Architecture principal

Monolithic architecture contains entire as one WAR file

**Adv:-**

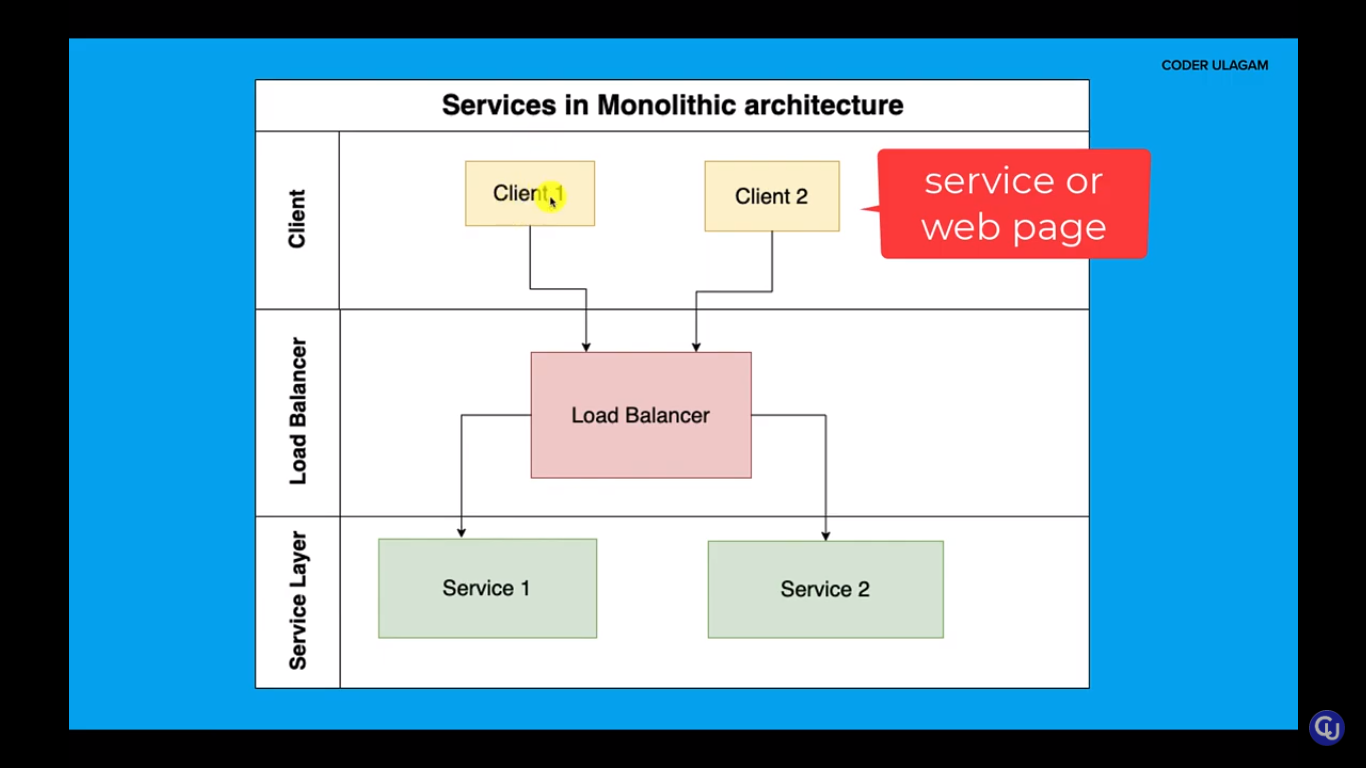
Scalable

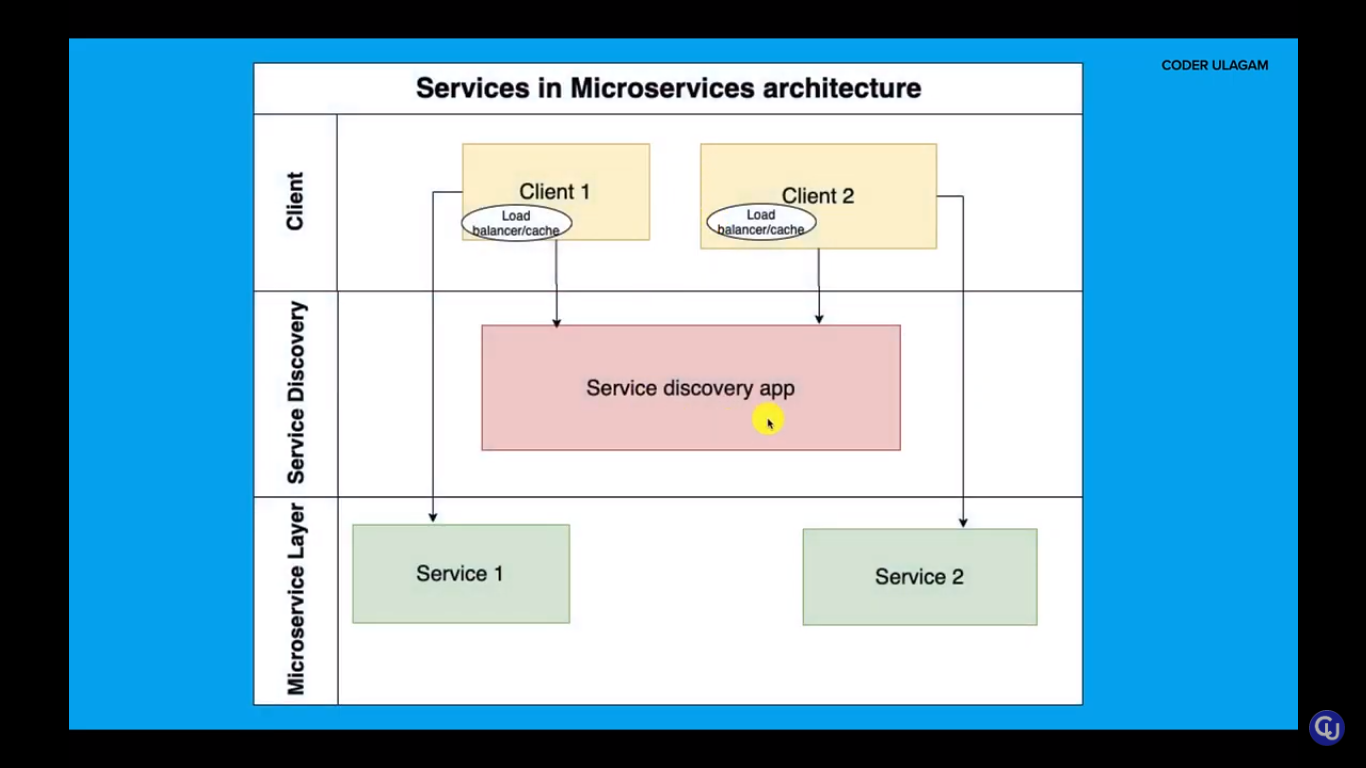
No single point of failure

**Dis adv:-**

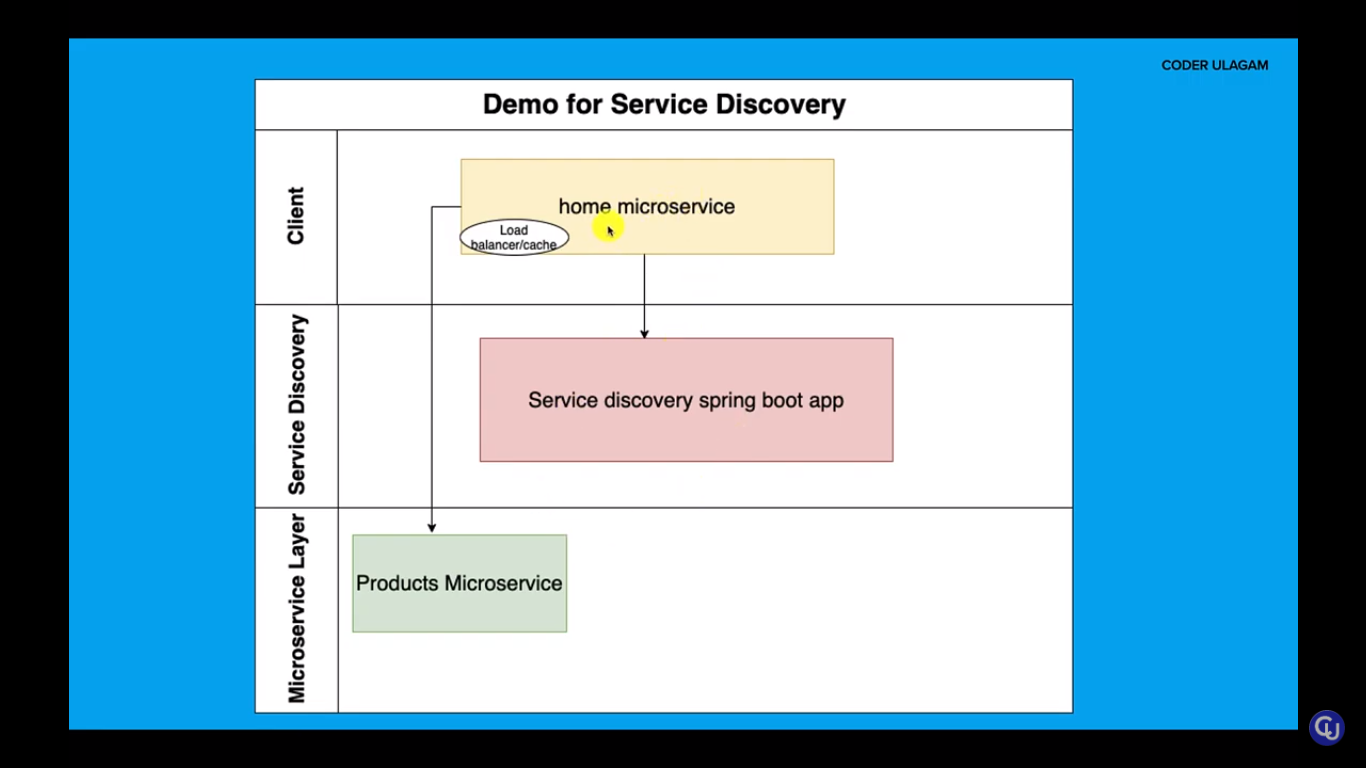
Network problem due to network call

**Service Discovery:-**





When we want to decouple client and service



Calling one microservice using another:-

Using REST template

Add eureka server

@EnableEurekaServer

To mention this service is EurekaServer, all other url will send it to discovery service

Eureka.client.register-with-eureka=false TO mention this is a server not a client with default port 8761

Eureka.client.fetch-registry=false

spring-boot:run in runas command

**To mention the service (Service discovery)is Server add this dependency in pom.xml:-**

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-netflix-eureka-server</artifactId>

</dependency>

**To mention the service is client add this dependency in pom.xml**

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>

</dependency>

**To change the name of application in service discovery use this statement:-**

spring.application.name=Home-Microservice

In RestTemplate:-

Add these annotations

@Bean

@LoadBalanced

**Config Server**

**Refresh using actuator**

**Spring config server**

1. Add config-server dependency in pom.xml
2. Add this annotation in main class @EnableConfigServer