24/07/24

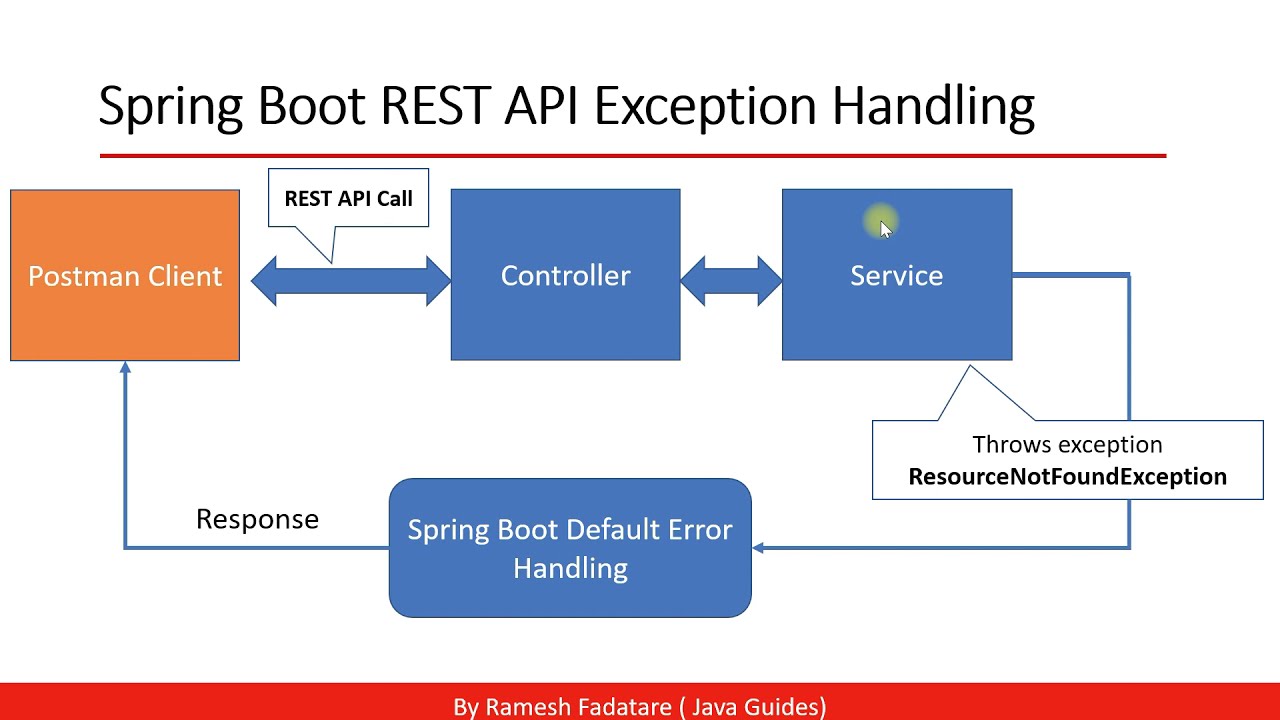
* @GeneratedValue

Cross cutting concerns in microservices: Cross-cutting concerns in microservices refer to aspects of a software system that affect multiple components or services across the architecture. These concerns often span different layers or modules of an application and are orthogonal to the core functionality of individual services. Cross-cutting concerns typically include areas such as logging, security, monitoring, authentication, authorization, and error handling.

[ControllerAdvice (Spring Framework 6.1.11 API)](https://docs.spring.io/spring-framework/docs/current/javadoc-api/org/springframework/web/bind/annotation/ControllerAdvice.html)

[Spring - RestTemplate - GeeksforGeeks](https://www.geeksforgeeks.org/spring-resttemplate/)

Webrequest will handle the output send to the client.



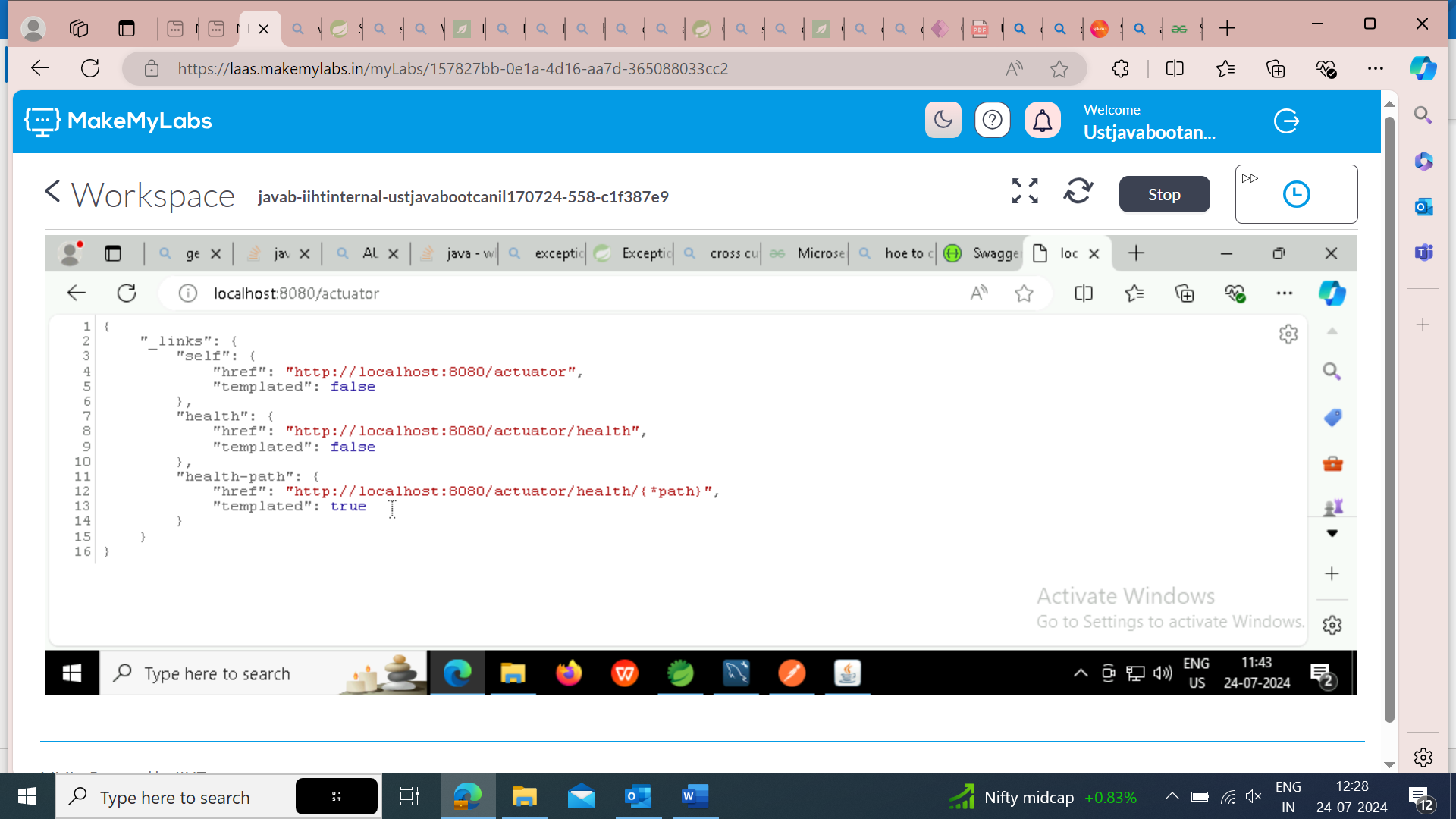
handleExceptionInternal-like responseentity for exception handling

* checked exceptions should be notified in the code(throws should be given).

SWAGGER:

http:/localhost:8080/swagger-ui/index.html

c:/programjavajava versionjconsole/local connection🡪we will get all the beans



[Endpoints :: Spring Boot](https://docs.spring.io/spring-boot/reference/actuator/endpoints.html#actuator.endpoints.exposing)

management.endpoints.web.exposure.include=\* 🡪(this is to be added n application properties)

A screenshot of a computer

Description automatically generated

-Mappings(onetomany,manytoone)- we have to learn.

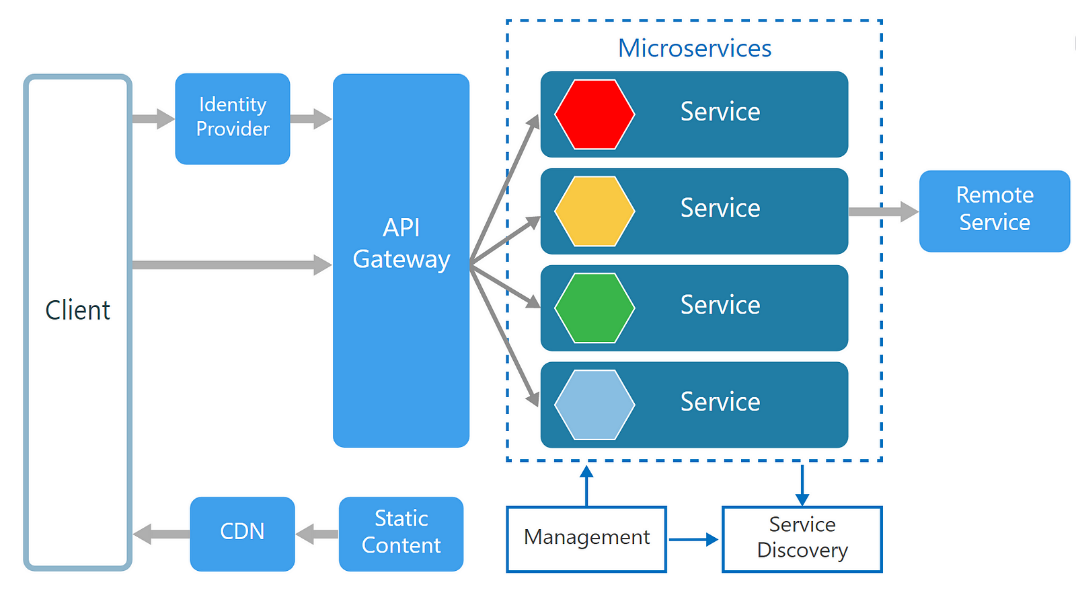
**Profiles**: Dev, Test, and Production. 🡪application-properties.profile=dev

* application-dev.properties
* application-test.properties
* application-prod.properties

Difference between landline and mobile phone:

* wired connection
* display
* not secure
* few number of landphones
* not handy

Microservices:

* Api gateway is also known as edge service.
* 
* The api gateway controls the access of various microservices by the clients.
* Service discover(in SOA)
* All the software available in CDN

[Spring Cloud](https://spring.io/projects/spring-cloud)

25/07/2024

[Spring Cloud Gateway](https://spring.io/projects/spring-cloud-gateway)

[Getting Started | Building a Gateway (spring.io)](https://spring.io/guides/gs/gateway)

* Cloud gateway:
  + Ports🡪registry:8761, Helloservice:8061, helloclient:8072, gateway:8070

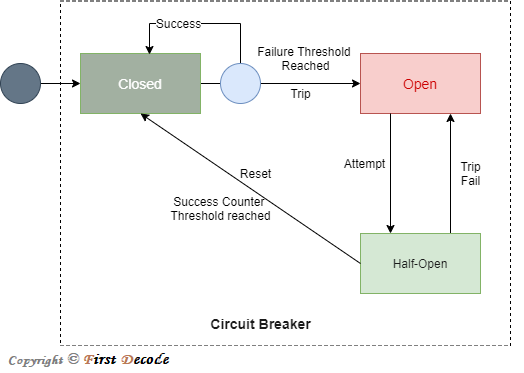
[The Twelve-Factor App (12factor.net)](https://12factor.net/)

Yaml lint-> for yaml files.

[Spring Cloud OpenFeign](https://spring.io/projects/spring-cloud-openfeign)

26/07/24

CircuitBreakers🡪implemented in microservices,on top of methods and ensures that the corresponding method returns a value properly. To avoid services from blocking another, this will let you know whether its responding.



LoadBAALANCER:

Gateway load balancing: 3 services

* Lbeureka
* Lbservice
* Lbgateway

Environment environment;(try)

[application name, traceId,spandId,export]

Application name – This is the name we set in the properties file, and can be used to aggregate logs from multiple instances of the same application.

TraceId – This is an id that’s assigned to a single request, job, or action. Something like each unique user initiated web request will have its own traceId.

SpanId – Tracks a unit of work. Think of a request that consists of multiple steps. Each step could have its own spanId and be tracked individually. By default, any application flow will start with the same TraceId and SpanId.

Export – This property is a boolean that indicates whether or not this log was exported to an aggregator like Zipkin. Zipkin is beyond the scope of this article, but plays an important role in analyzing logs created by Sleuth.

SECURITY:

[Architecture :: Spring Security](https://docs.spring.io/spring-security/reference/servlet/architecture.html)

[Lightweight Directory Access Protocol (LDAP) - GeeksforGeeks](https://www.geeksforgeeks.org/lightweight-directory-access-protocol-ldap/)

Authenticated and authorized