**MICROSERVICES ARCHITECTURE WITH REST TEMPLATE**

This POC is mainly focus on communicating one service to another service through the Gateway.

**Used Technology:**

* Java 1.8
* Spring Boot for API
* Spring Boot JPA.
* Spring Cloud.
* Eureka Server.
* Swagger UI.
* MySQL for Database

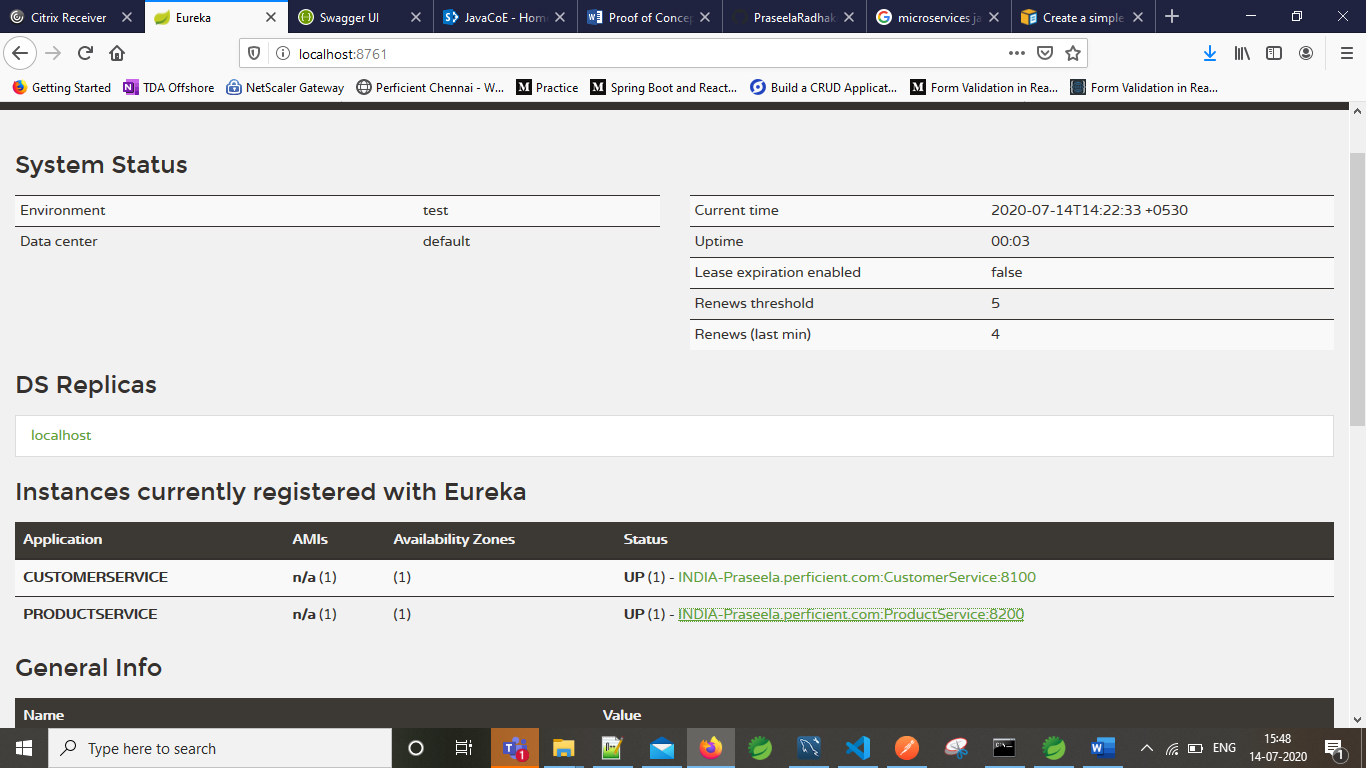
**Microservice:**

**Microservices** - also known as the **microservice** architecture - is an architectural style that structures an **application** as a collection of services that are. Highly maintainable and testable. Loosely coupled. Independently deployable. Organized around business capabilities.

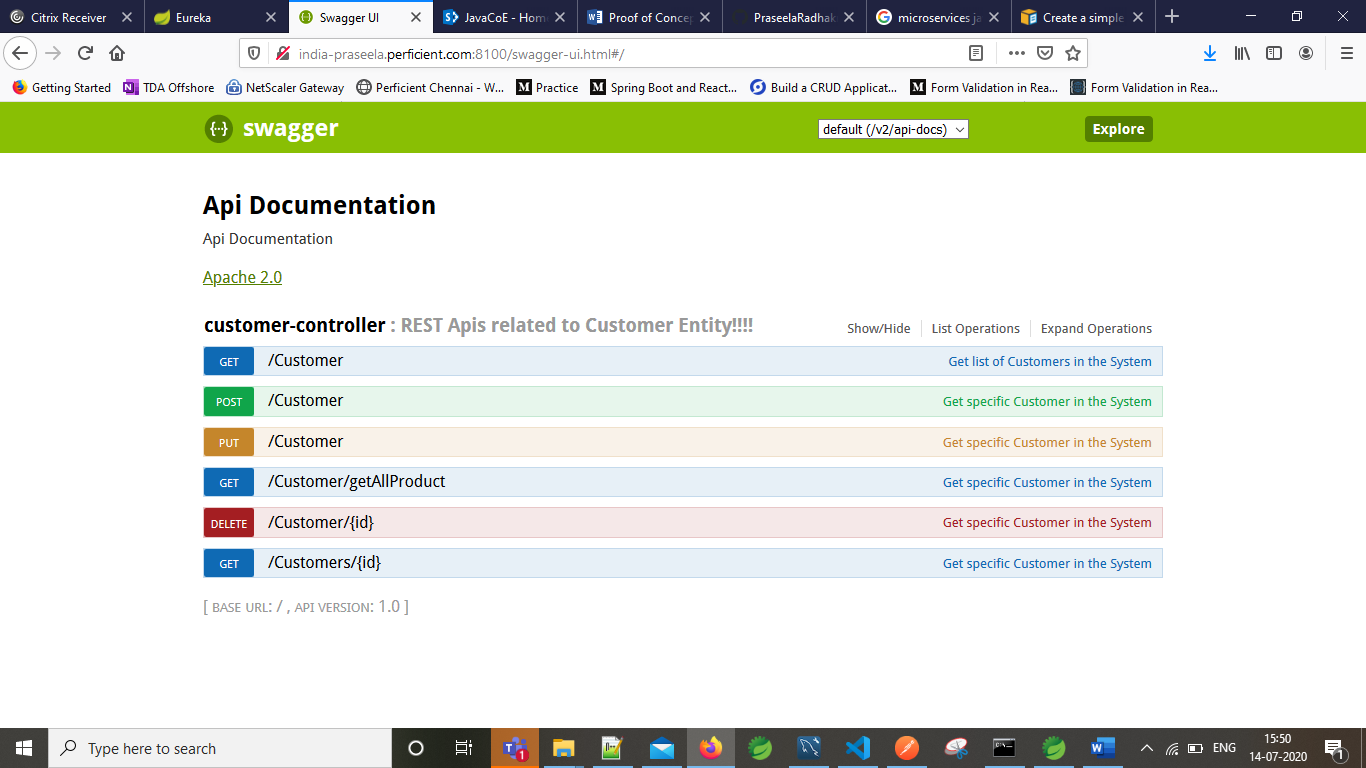
**Application Description:**

In this POC, I have created one Gateway and two services. These two services are communicating through the Gateway. The EcartGateway contain Product and Customer Service. Each service contains ( GET, POST, PUT, DELETE ) endpoints to trigger that function and stored and retrieved it from database. The Customer service will get the list of Product from the Product service with the help of gateway while its running stage.

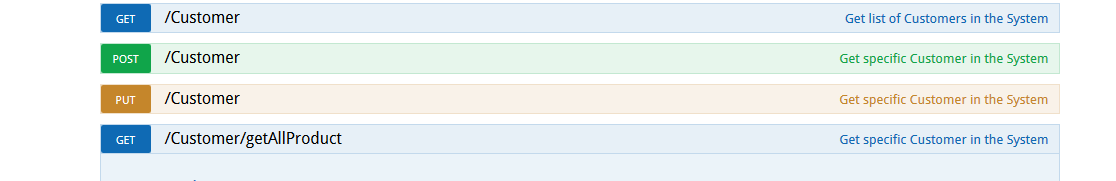
**Eureka Server:**



**Customer Service:**



**Customer Service get product from Product Service:**



**Product Service:**

