**MZUMBE UNIVERSITY**

**FACULTY OF SCIENCE AND TECHNOLOGY**

**COURSE NAME: DISTRIBUTIVE SYSTEMS**

**SUBJECT CODE: ICT 311**

**TASK: GROUP ASSIGNMENT**

|  |  |  |
| --- | --- | --- |
| NAME | course | REGISTRATION NUMBER |
| VERONICA. A. MDEPA | ITS | 13301030/T.19 |
| SIDDY IDDY SIDDY | ICTM | 14320011/T.20 |
| MTATIRO M. CHILARE | ITS | 13301037/T.19 |
| JENETA M. WILLIAM | ICTM | 13304199/T.18 |

Task:

DEVELOP MICROSERVICES PROJECT WITH JAVA

Contents

[ABSTRACT 3](#_Toc127041652)

[1. INTRODUCTION 4](#_Toc127041653)

[Key Components of a Microservices Architecture 4](#_Toc127041654)

[1.1 Purpose 5](#_Toc127041655)

[Advantages 5](#_Toc127041656)

[1.2 Scope 6](#_Toc127041657)

[1.2.1 Existing system with limitation 6](#_Toc127041658)

[1.2.2 Proposed system features: 6](#_Toc127041659)

[2. ANALYSIS 7](#_Toc127041660)

[2.1 Project software requirements specification 7](#_Toc127041661)

[2.2 Project hardware requirements specification 7](#_Toc127041662)

[3. Implementation 8](#_Toc127041663)

[1. Identifying the requirements 8](#_Toc127041664)

[2. Identifying the architecture 9](#_Toc127041665)

[3. building each Microservices and its modules 9](#_Toc127041666)

[4. Testing of each Microservices using postman 10](#_Toc127041667)

[5.to make underservices communication 11](#_Toc127041668)

[6. API GATEWAY 11](#_Toc127041669)

[7. IMPLEMENTING THE SECURITY USING KEYCLOAK 12](#_Toc127041670)

[8. MONITORING THE MICROSERVICES USING THE GRAFANA AND PROMETHEUS 13](#_Toc127041671)

[REFERENCES 14](#_Toc127041672)

# ABSTRACT

The project MZUMBE ONLINE SHOPPING MICROSERVICES WITH JAVA PROJECT software architectural style in which a large application is built as a collection of small, independent services that communicate with each other over a network.

This project shows how various services like order service, notification service, inventory service, and product service can communicate to make a system of shopping.

Early it was difficult to integrate the various services to work as a single system.

This project helps to show how various services work in a distributed manner with necessary characteristics of a distributed system like security, heterogeneity an others

# INTRODUCTION

Mzumbe online shopping Microservices with java Microservices are a software architectural style in which a large application is built as a collection of small, independent services that communicate with each other over a network.

It contains services that are product services, order service, inventory service, notification service,

API gateway and the discovery server.

Each service is a self-contained unit of functionality that can be developed, tested, and deployed independently of the other services. This allows for more flexibility and scalability than a monolithic architecture, where all the functionality is contained in a single, large codebase.

Microservices can be written in different programming languages and use different technologies, as long as they can communicate with each other through a common API.

They are designed to be loosely coupled, meaning that changes to one service should not affect the other services. This makes it easier to update, maintain, and scale the application. Microservices arch must handle a high volume of traffics and action that need to handle a high volume of traffic and need to be scaled horizontally.

# Key Components of a Microservices Architecture

Key components of a Microservices architecture include:

1. **Core Services**: Each service is a self-contained unit of functionality that can be developed, tested, and deployed independently of the other services.
2. **Service registry**: A service registry is a database of all the system’s services, locations, and capabilities. It allows services to discover and communicate with each other.
3. **API Gateway:** An API gateway is a single entry point for all incoming requests to the Microservices. It acts as a reverse proxy, routing requests to the appropriate service and handling tasks such as authentication and rate limiting.
4. **Message bus:** A message bus is a messaging system that allows services to communicate asynchronously with each other. This can be done through protocols like HTTP, RabbitMQ, or Kafka.
5. **Monitoring and logging:** Monitoring and logging are necessary to track the health of the services and troubleshoot problems.
6. **Service discovery and load balancing:** This component is responsible for discovering service instances and directing traffic to the appropriate service instances based on load and availability.
7. **Continuous integration and continuous deployment (CI/CD):**To make the development and deployment process of Microservices as smooth as possible, it is recommended to use a tool such as Jenkins, TravisCI, or CircleCI to automate the process of building, testing, and deploying Microservices.

In Mzumbe online shopping project customers may place orders for products and the system will look for the availability of a particular product in the store thus providing feedback or notification.

## Purpose

The purpose of Mzumbe online shopping Microservices project with java is to:

* to provide a fast and reliable platform for online shopping
* to offer the seamless user experience, from browsing products to making payments
* to improve application security by implementing the security measure at the Microservices level
* to enable effective resource utilization by allowing each Microservices to be deployed and scaled independently

### Advantages

* Productive and focused teams
* Keeping tabs on security
* Quick deployment
* Isolation
* Flexibility
* Improvement in quantity
* Scalability
* Continuous delivery

## Scope

### Existing system with limitation

The existed systems online shopping system are not capable to integrate and work as a single system to save time and effective utilization of resources.

### Proposed system features:

The Microservices architecture is a system of data exchanges designed to form a complete application. Services and components are working in tandem to provide the web of services and apps necessary to customize each entity's data architecture.

Additionally, it provides the best fault tolerance to prevent failures resulting from a single service malfunction or error. This system also uses API gateways to achieve communication between services and offers data separation for Independent data storage.

It also provides a decentralized web of services with few dependencies. With these unique features, it's easy to design and deploy specific service functions to help solve issues and address certain issues.

In some instances, the Microservices architecture uses what's referred to as libraries but is defined by the way each service links yet operates independently. These linked services use a remote procedure recall or web service requests to communicate with one another.

A benefit of using services is that it allows applications to move more freely from one another without complex encapsulation.

Another characteristic of this architecture is that there may be instances of more than one process being deployed simultaneously to fit specific needs.

# ANALYSIS

System analysis for Mzumbe online shopping Microservices with java project involve following steps:

1. Identify the requirements for online shopping Microservices
2. Determine the system architecture
3. Analyze performance of each Microservices
4. Evaluate security
5. Determine scalability
6. Identify integration requirements
7. Assess the maintenance requirements

## Project software requirements specification

## A set of programs associated with the operation of a computer is called software. Software is the part of the computer system, which enables the user to interact with several physical hardware devices. The minimum software requirement specifications for developing this project are as follows:

Operating system: Windows XP/Vista/2000, Linux, mac.

Presentation layer: spring boot with java

Database: spring boot with java.

## Project hardware requirements specification

## The collection of internal electronic circuits and external physical devices used in building a computer is called the Hardware. The minimum hardware requirement specifications for developing this project are as follows:

Processor: Standard processor with a speed of 1.6 GHz RAM: 26 MB RAM or more

Hard Dis : 20 GB or more

Monitor : Standard color monitor

# Implementation

The following are the stages for implementation for the online shopping Microservices project with java.

## 1. Identifying the requirements

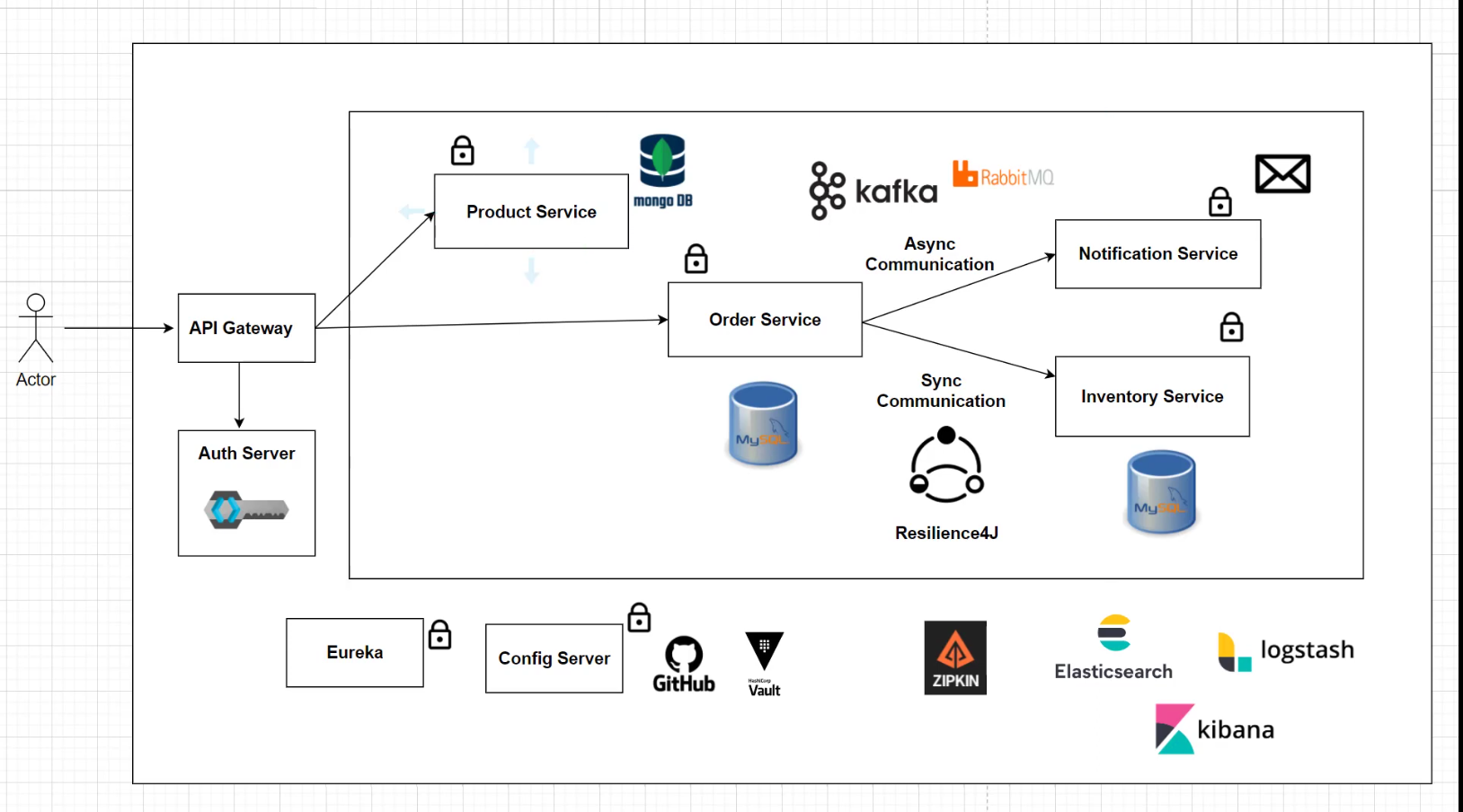
The online shopping application involve has the following Microservices.

* Order service; which is able for managing the process of creating, storing, and processing the customer order.
* Products services: for managing product catalog, including the creation, storage and retrieval of product information.
* Inventory services; for tracking the availability of product and ensuring there is enough stock to fulfill customer orders.
* Notification service; for managing the process of sending notifications to customers
* API gateway service; act as single entry for all incoming requests, providing a centralized location for routing, security and monitoring of the APIs in the platform

It needs following platform for implementation;

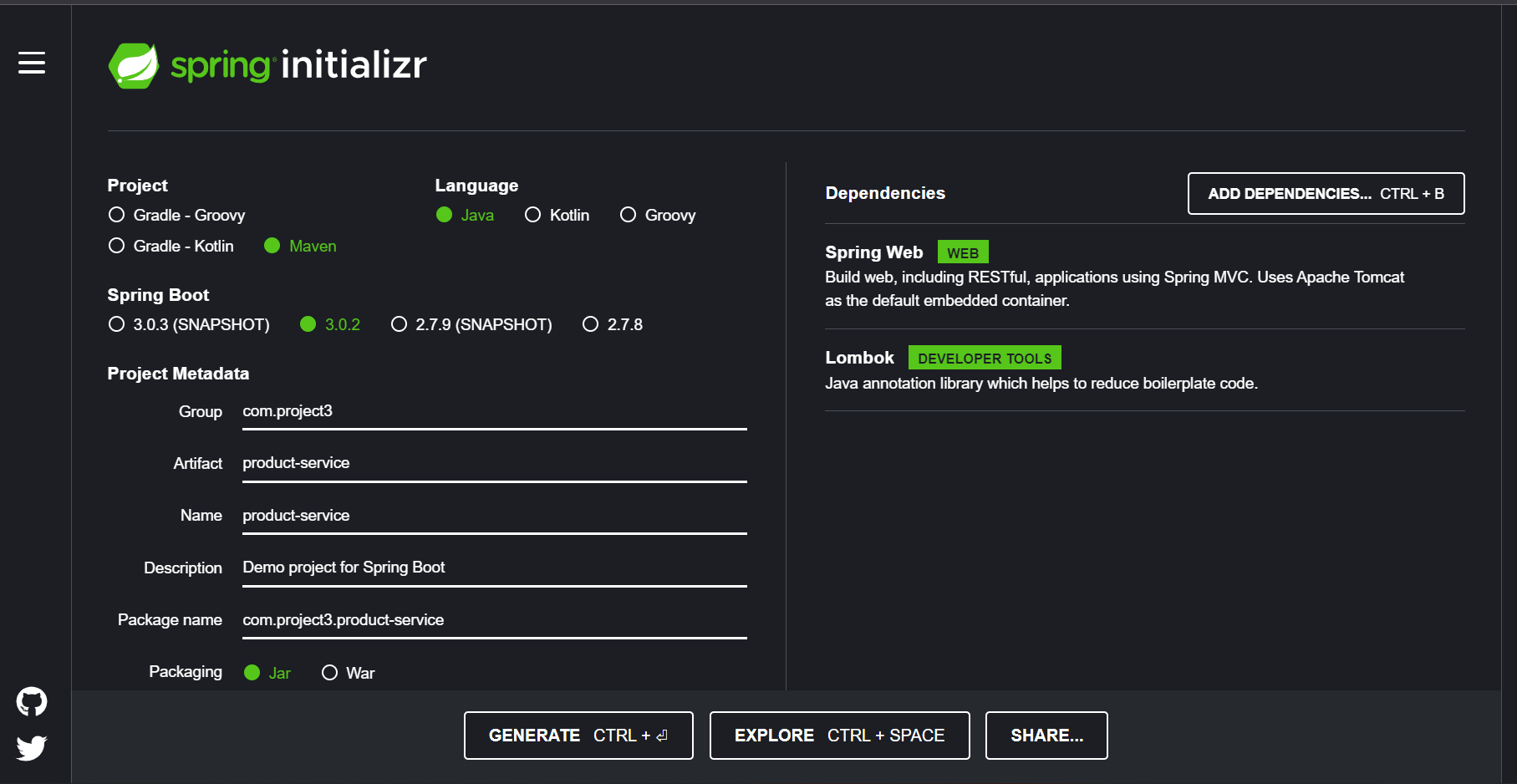
* Eureka
* Github
* Zipkin
* Kafka
* Resilience 4j
* Mongo database

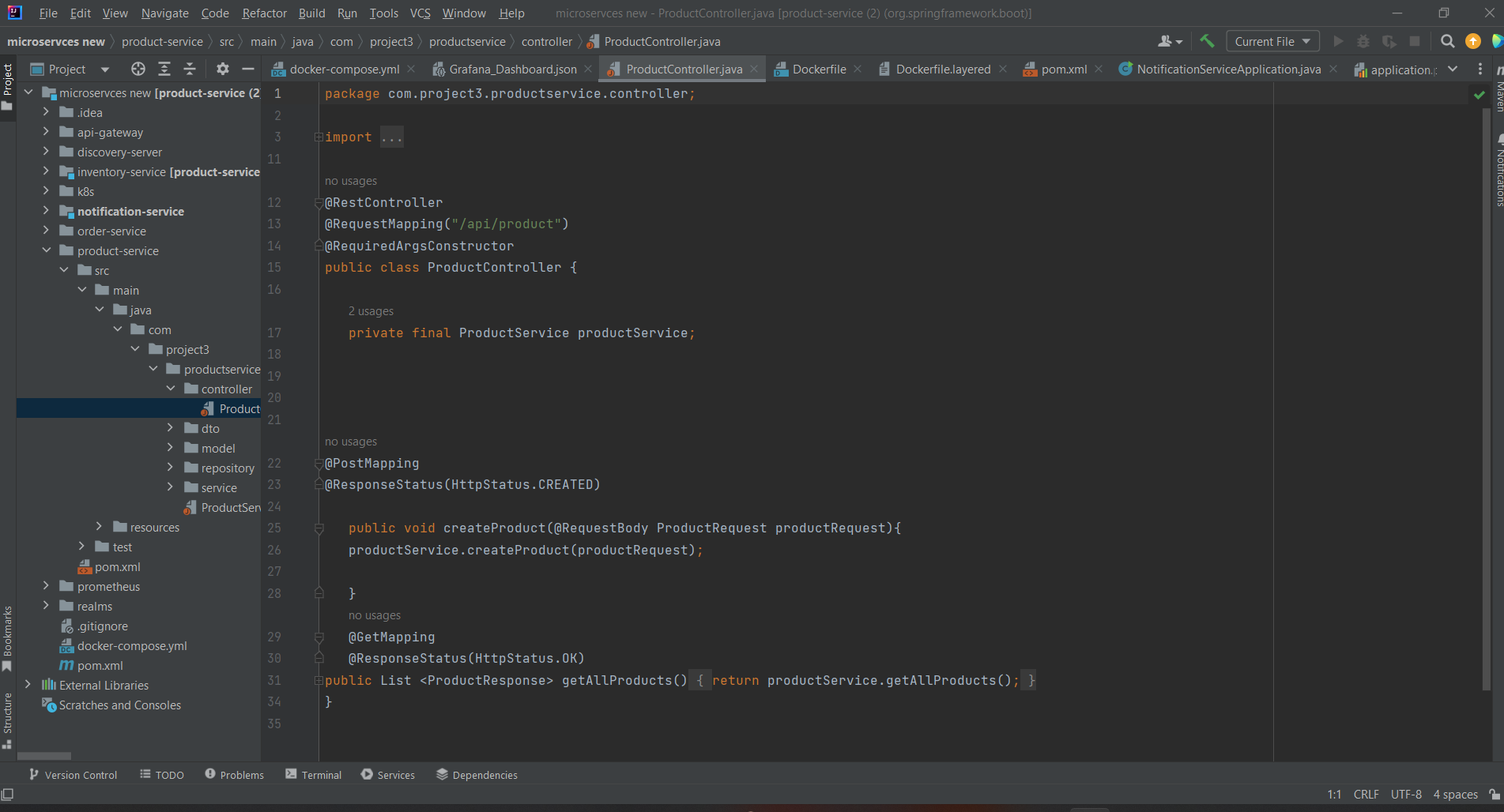
## 2. Identifying the architecture



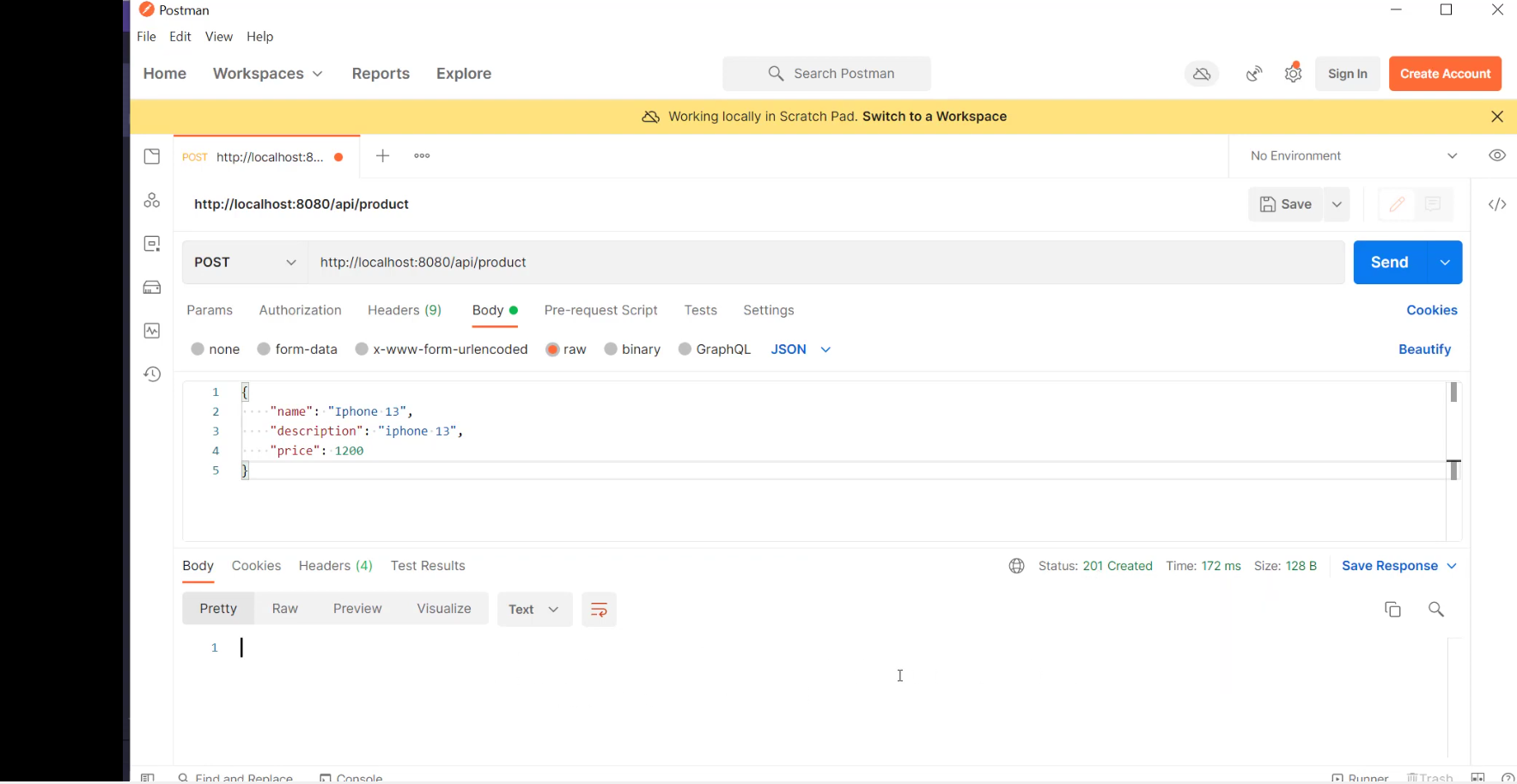
## 3. building each Microservices and its modules

This was done using the SPRINGBOOT WITH JAVA and INTELLIJ IDEA

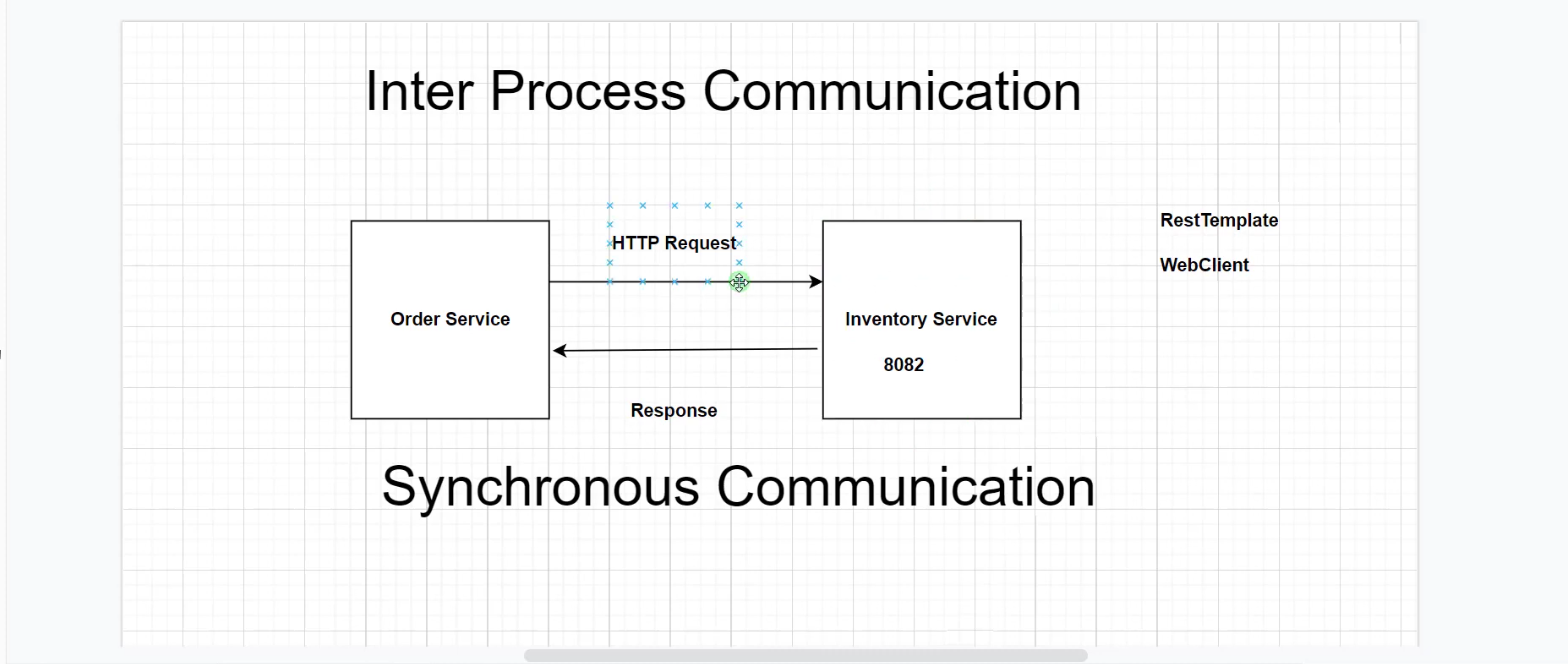




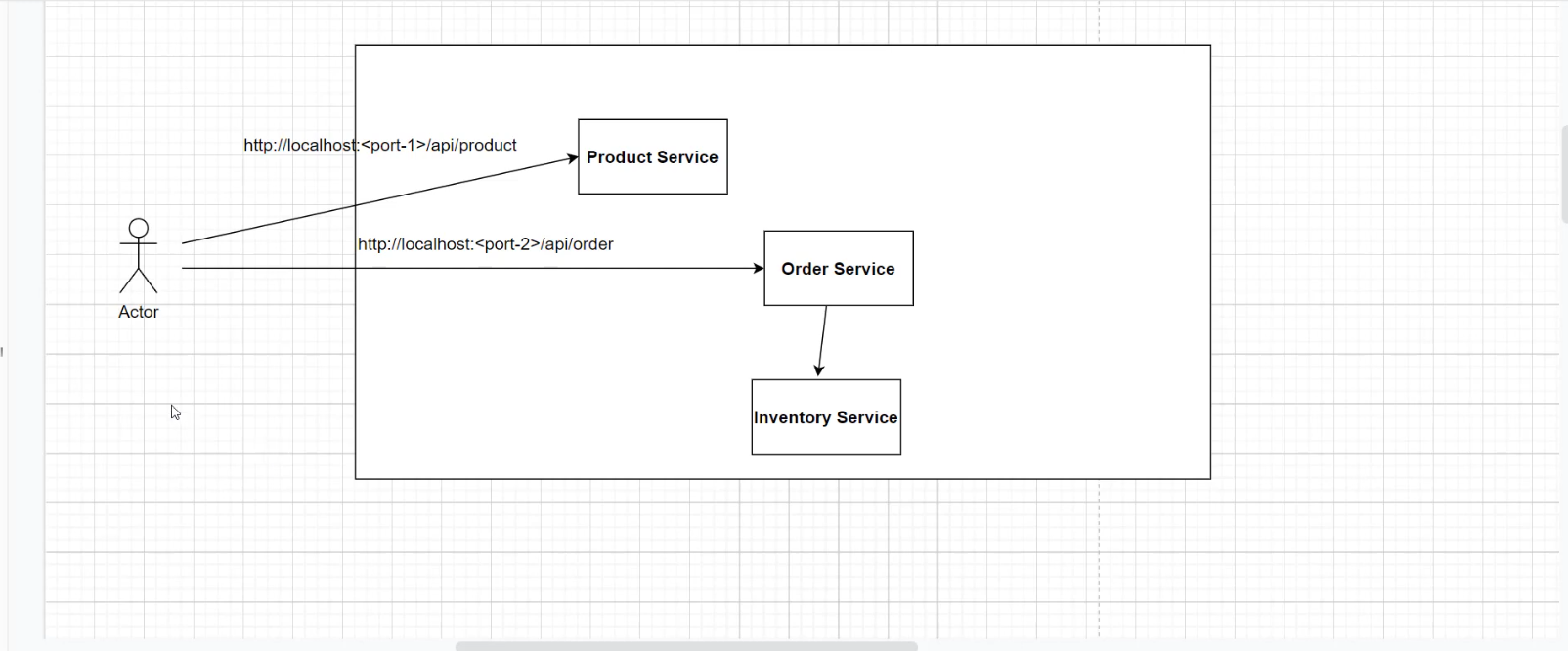
## Testing of each Microservices using postman

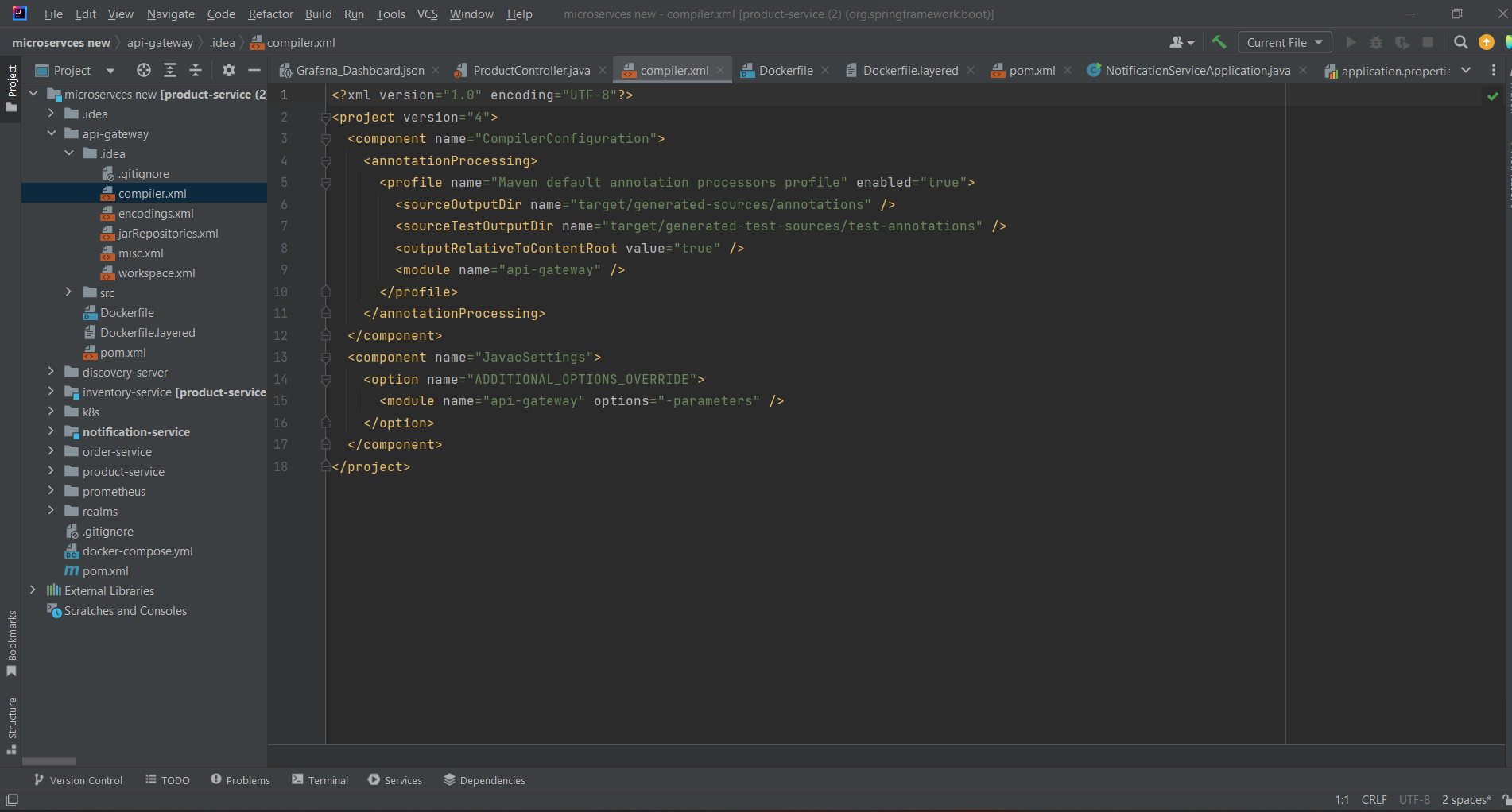


## 5.to make underservices communication

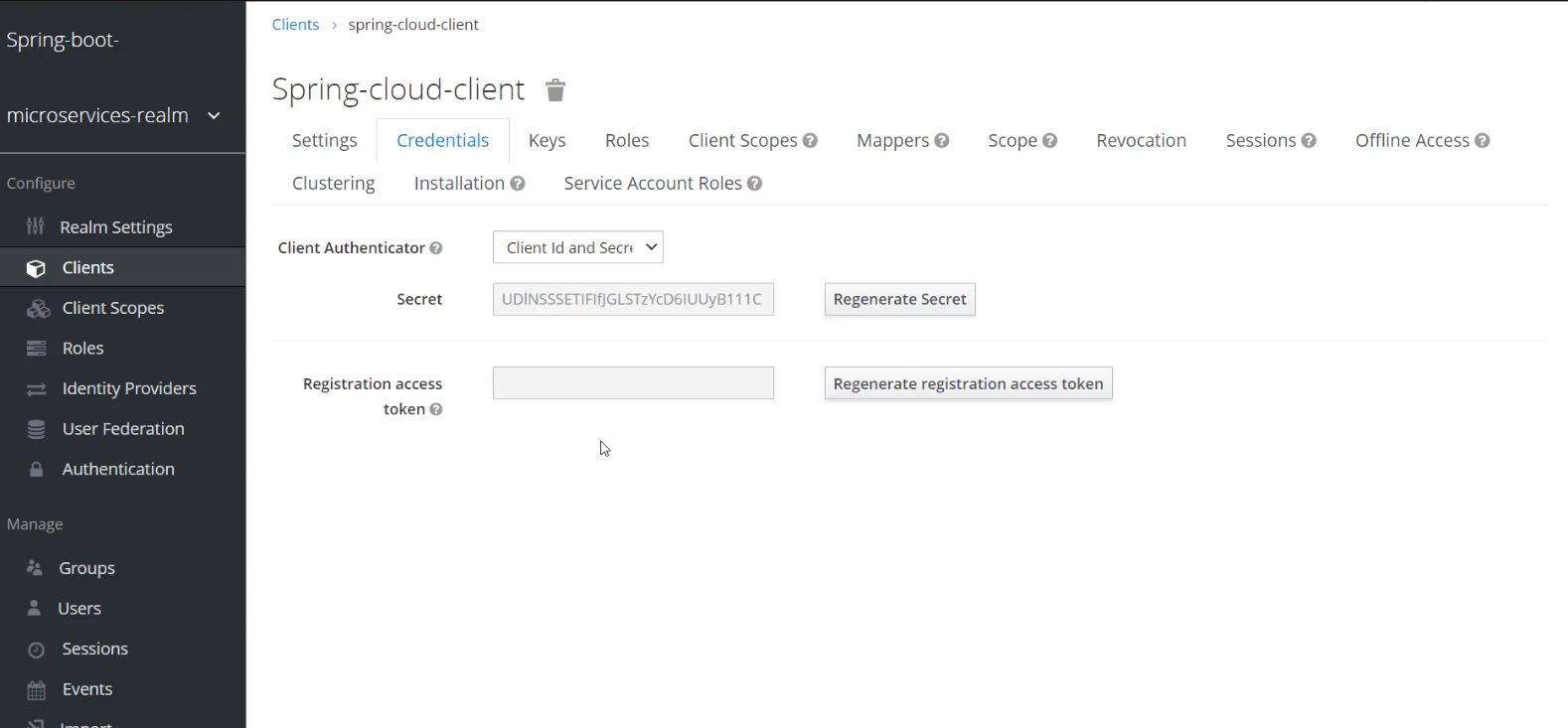


## 6. API GATEWAY

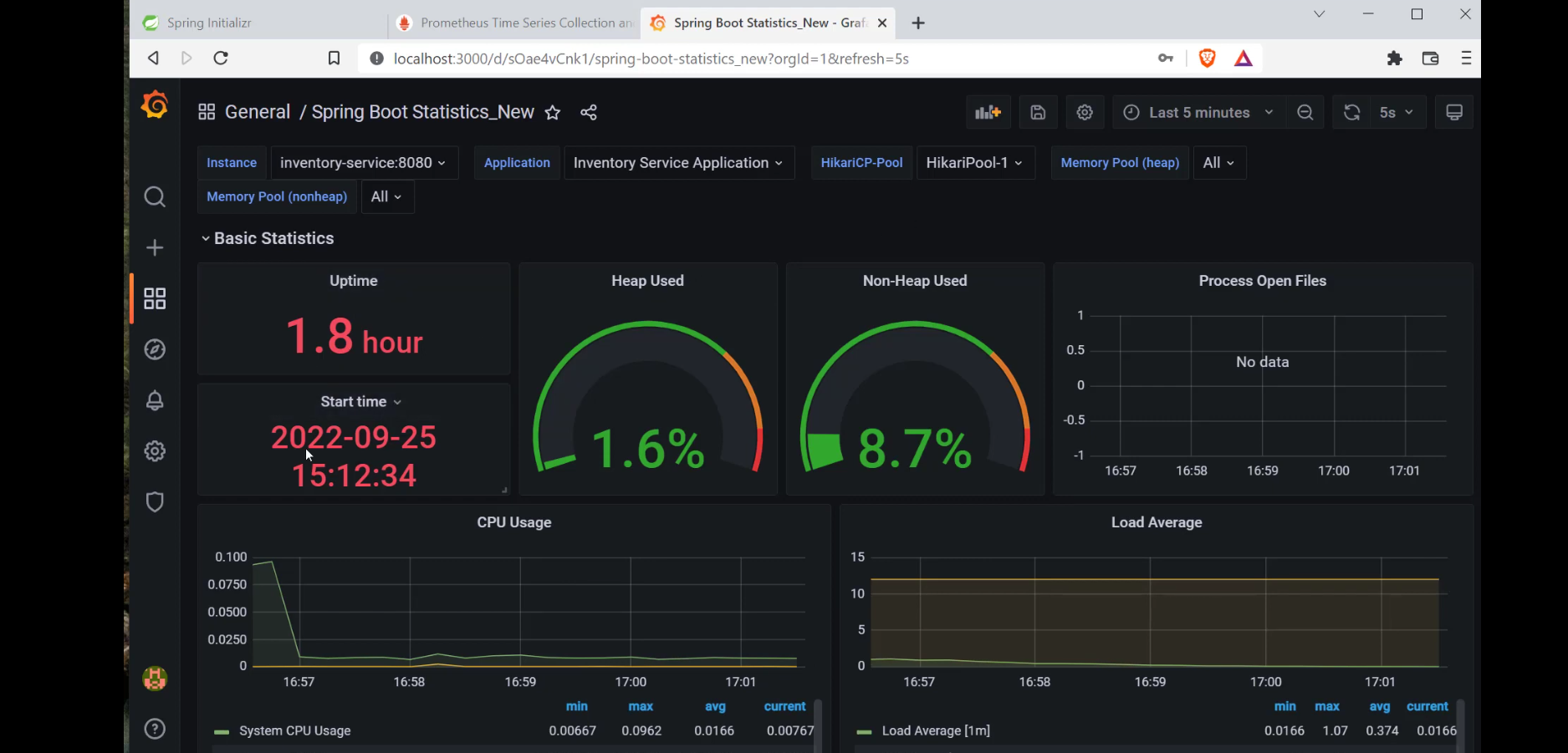




## 7. IMPLEMENTING THE SECURITY USING KEYCLOAK



## MONITORING THE MICROSERVICES USING THE GRAFANA AND PROMETHEUS



# REFERENCES

<https://start.spring.io/>

<https://vfunction.com/blog/what-is-the-use-of-microservices-in-java/>

<https://www.adservio.fr/post/what-are-microservice-architecture-features-and-components>

<https://www.keycloak.org/>