

Configuration Manual

MSc Research Project
Cloud Computing

Ritika
Student ID: 22208691

School of Computing
National College of Ireland

Supervisor: Diego Lugones

National College of Ireland
MSc Project Submission Sheet
School of Computing



Student Name: ...Ritika.....
Student ID:22208691.....
Programme: ...Cloud Computing **Year:** 2023-2024...
Module:MSc Research Project.....
Lecturer: Diego Lugones.....
Submission Due Date:12/08/2024.....
Project Title: Enhancing Microservice Performance: A Hybrid Model Combining Service Discovery and Circuit Breaker Patterns for Optimized Performance.....
Word Count: **Page Count:**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Ritika
Date: 12/08/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	

Penalty Applied (if applicable):	
----------------------------------	--

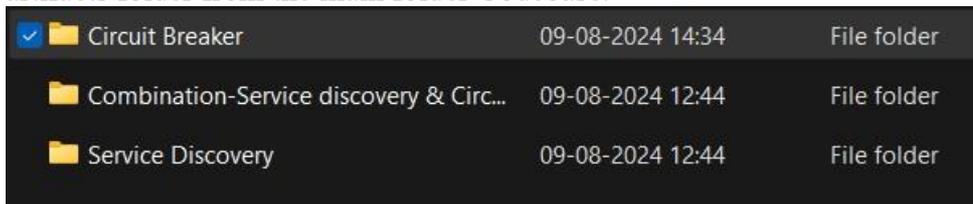
Configuration Manual

Ritika

Student ID: 22208691

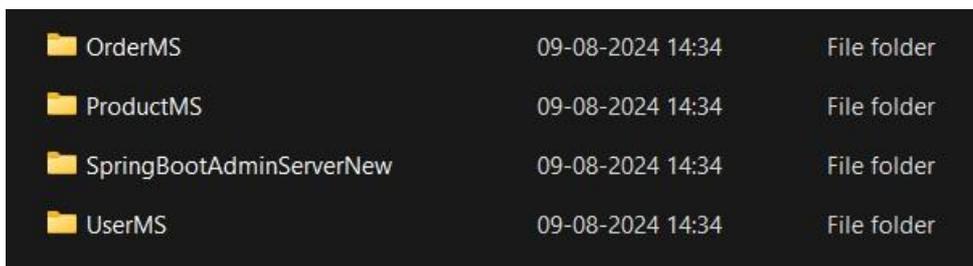
1. Installation of Circuit Breaker - Spring Boot based project.

- a. At First Step, extract the project folders named as Circuit Breaker code base from artifacts folder from the main folder Codebase.



<input checked="" type="checkbox"/>	Circuit Breaker	09-08-2024 14:34	File folder
<input type="checkbox"/>	Combination-Service discovery & Circ...	09-08-2024 12:44	File folder
<input type="checkbox"/>	Service Discovery	09-08-2024 12:44	File folder

- b. Then, clone the Circuit Breaker backend code project named Circuit Breaker with contents including OrderMS, ProductMS and UserMS. SpringBootAdminServiceNew is the external dependency that provides metrics related to health of the services.



<input type="checkbox"/>	OrderMS	09-08-2024 14:34	File folder
<input type="checkbox"/>	ProductMS	09-08-2024 14:34	File folder
<input type="checkbox"/>	SpringBootAdminServerNew	09-08-2024 14:34	File folder
<input type="checkbox"/>	UserMS	09-08-2024 14:34	File folder

- c. The functionality for adding any product to cart in the shopping backend application can be installed by importing the project course in Spring Boot Framework or Eclipse.
- d. The projects can be imported as an **existing maven project** the IDE. Now run the command **mvn clean install** in a way to clean the project and reinstall all the dependencies of the project from the repository.

```

[INFO] --- resources:3.1.0:testResources (default-testResources) @ OrderMS ---
[INFO] Not copying test resources
[INFO]
[INFO] --- compiler:3.8.1:testCompile (default-testCompile) @ OrderMS ---
[INFO] Not compiling test sources
[INFO]
[INFO] --- surefire:2.22.2:test (default-test) @ OrderMS ---
[INFO] Tests are skipped.
[INFO]
[INFO] --- jar:3.2.0:jar (default-jar) @ OrderMS ---
[INFO] Building jar: C:\Users\SURAJ LUMBA\Documents\RitikaDocs\Shopping\OrderMS\target\OrderMS-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- spring-boot:2.3.2.RELEASE:repackage (repackage) @ OrderMS ---
[INFO] Replacing main artifact with repackaged archive
[INFO]
[INFO] --- install:2.5.2:install (default-install) @ OrderMS ---
[INFO] Installing C:\Users\SURAJ LUMBA\Documents\RitikaDocs\Shopping\OrderMS\target\OrderMS-0.0.1-SNAPSHOT.jar to C:\Users\SURAJ LUMBA\Documents\RitikaDocs\Shopping\OrderMS\pom.xml to C:\Users\SURAJ LUMBA\.m2\repository\com
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO]
[INFO] Total time: 9.592 s
[INFO] Finished at: 2024-08-09T14:55:24+01:00
[INFO] -----

```

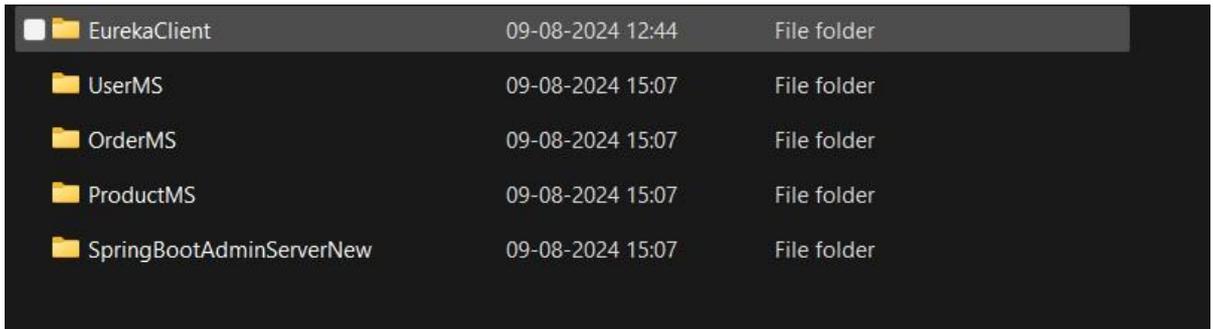
- e. Similarly, mvn clean install can be done for all the services.
- f. After this, we will right click on all the files and **SpringBoot Application**.
- g. After this, the applications will run and will be up for the availability of the microservice. The spring boot application will be running on the ports **5100,5200,5300**. We will get the below shown logs on Spring Admin once the application will be up.



- h. And then through postman we can send request to the services and make use of the applications.

2. Installation of Service Discovery - Spring Boot based project.

- a. Similarly, clone the Service Discovery backend code project named Service Discovery with contents including OrderMS, ProductMS and UserMS. SpringBootAdminServiceNew is the external dependency that provides metrics related to health of the services.
- b. Import all the services with Eureka Client with all services registered as eureka servers.



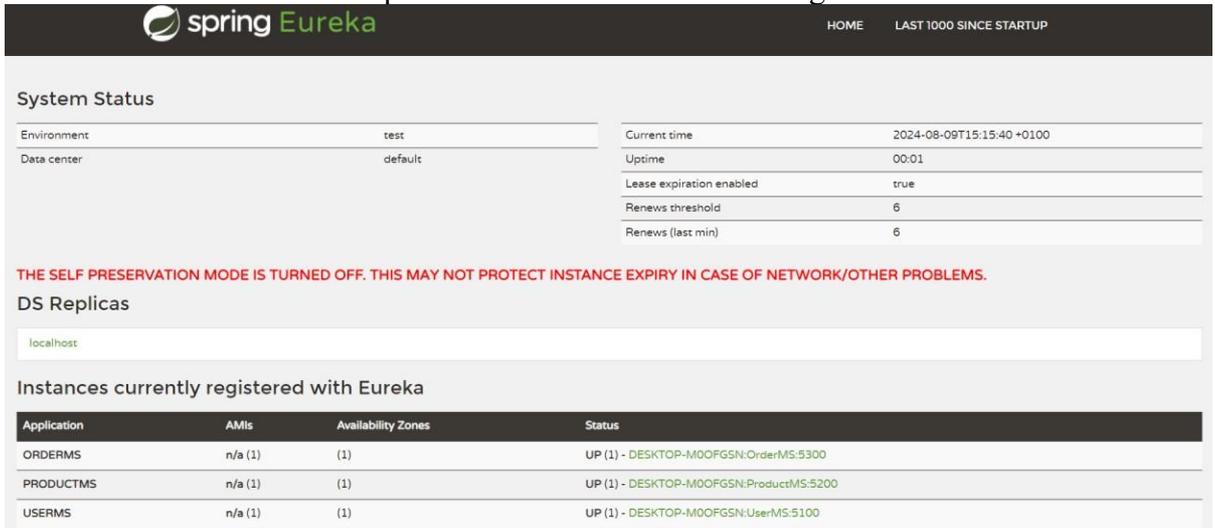
- c. Similarly, mvn clean install can be done for all the services.
- d. After this, we will right click on all the files and **Run as SpringBoot Application**.
- e. After this, the applications will run and will be up for the availability of the microservice. The spring boot application will be running on the ports **5100,5200,5300**. We will get the below shown logs on Spring Admin once the application will be up.
- f. In this we also run Eureka Client as Spring Boot Application.

```

10] e.s.EurekaServerInitializerConfiguration : Started Eureka Server
in] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8761 (http) with context path ''
in] .s.c.n.e.s.EurekaAutoServiceRegistration : Updating port to 8761
in] com.example.EurekaClientApplication      : Started EurekaClientApplication in 7.786 seconds (JVM running for 9.189)
.1] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring DispatcherServlet 'dispatcherServlet'
.1] o.s.web.servlet.DispatcherServlet        : Initializing Servlet 'dispatcherServlet'
.1] o.s.web.servlet.DispatcherServlet        : Completed initialization in 8 ms
-1] c.n.e.registry.AbstractInstanceRegistry  : DS: Registry: lease doesn't exist, registering resource: ORDERMS - DESKTOP-M00FGSN:OrderMS:5300
-1] c.n.eureka.resources.InstanceResource    : Not Found (Renew): ORDERMS - DESKTOP-M00FGSN:OrderMS:5300
-2] c.n.e.registry.AbstractInstanceRegistry  : Registered instance ORDERMS/DESKTOP-M00FGSN:OrderMS:5300 with status UP (replication=false)
-5] c.n.e.registry.AbstractInstanceRegistry  : Registered instance ORDERMS/DESKTOP-M00FGSN:OrderMS:5300 with status UP (replication=true)

```

- g. Eureka Client is available on port **8761** as shown in above image.



- h. The above image is the eureka registry dashboard with services registered.

3. Installation of Combination- Service Discovery & Circuit Breaker - Spring Boot based project.

- a. As followed in above steps we can install all the services and import it in our IDE.
- b. We can use Combination- Service Discovery & Circuit Breaker folder to access all the projects.