**Room Reservation System:**

‘rrs’ database has created with create-drop ddl property.

A screen shot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

Eureka service registry with api-gateway, config-server, property-app, booking-app, payment-app

A screenshot of a computer

Description automatically generated

HTTP Request using api-gateway(port:9013):-

1. POST URL: to create the property record into the database from property-app microservice.

A screenshot of a computer

Description automatically generated

2. GET URL: to fetch the all properties details from database.

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

3. GET Request : to fetch the property detail based on ID

URL: http://localhost:9013/property/1

A screenshot of a computer

Description automatically generated

4. HTTP GET: to fetch the property by location.

URL: <http://localhost:9013/property/location/bangalore>

A screenshot of a computer

Description automatically generated

5. Booking property from booking-app microservice.

HTTP POST request: to book the properties

URL: http://localhost:9013/book

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

* **Circuit breaker Failure Case into Booking-app microservice:**

If any downsteam api are down or having any error then while making api call from the booking microservices it will automatically open the circuit breaker to handle this fall back scenario.

Currently there is not issue and circuit breakers state is closed.

HTTP GET Request: <http://localhost:9110/actuator/circuitbreakers>

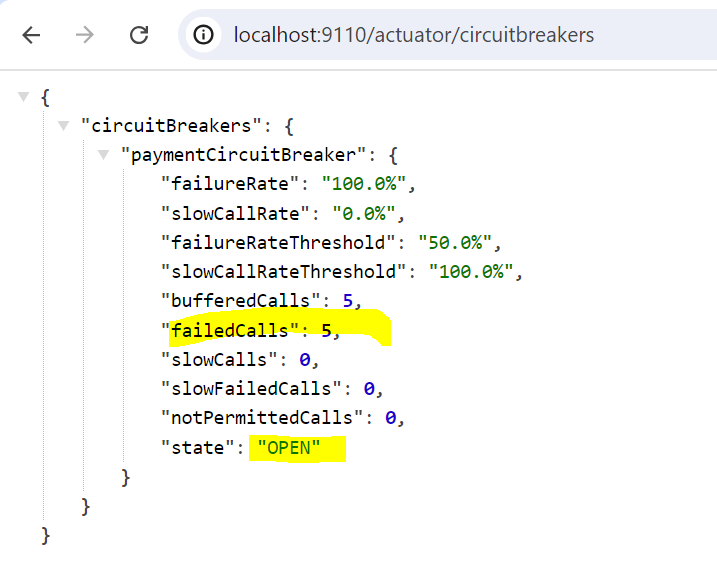
A screenshot of a computer program

Description automatically generated

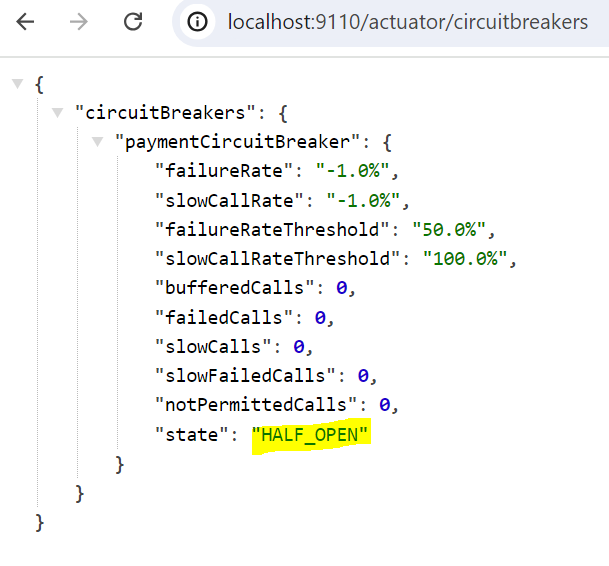
Make the 5 consecutive failed request with wrong property id. It will open the circuit breaker. From open to half open automatically transition will take place after 5 sec(as per config)

A screen shot of a computer program

Description automatically generated



Automatically circuit breaker state will transition from OPEN to Half-open within 5sec.



**Zipkin:**

We can use zipkin for Distributed tracing.

As shown below, for the post request of booking a property, these are the internal calls that are being made to other microservices and we can trace them and their respective timings using zipkin.

A screenshot of a computer

Description automatically generated

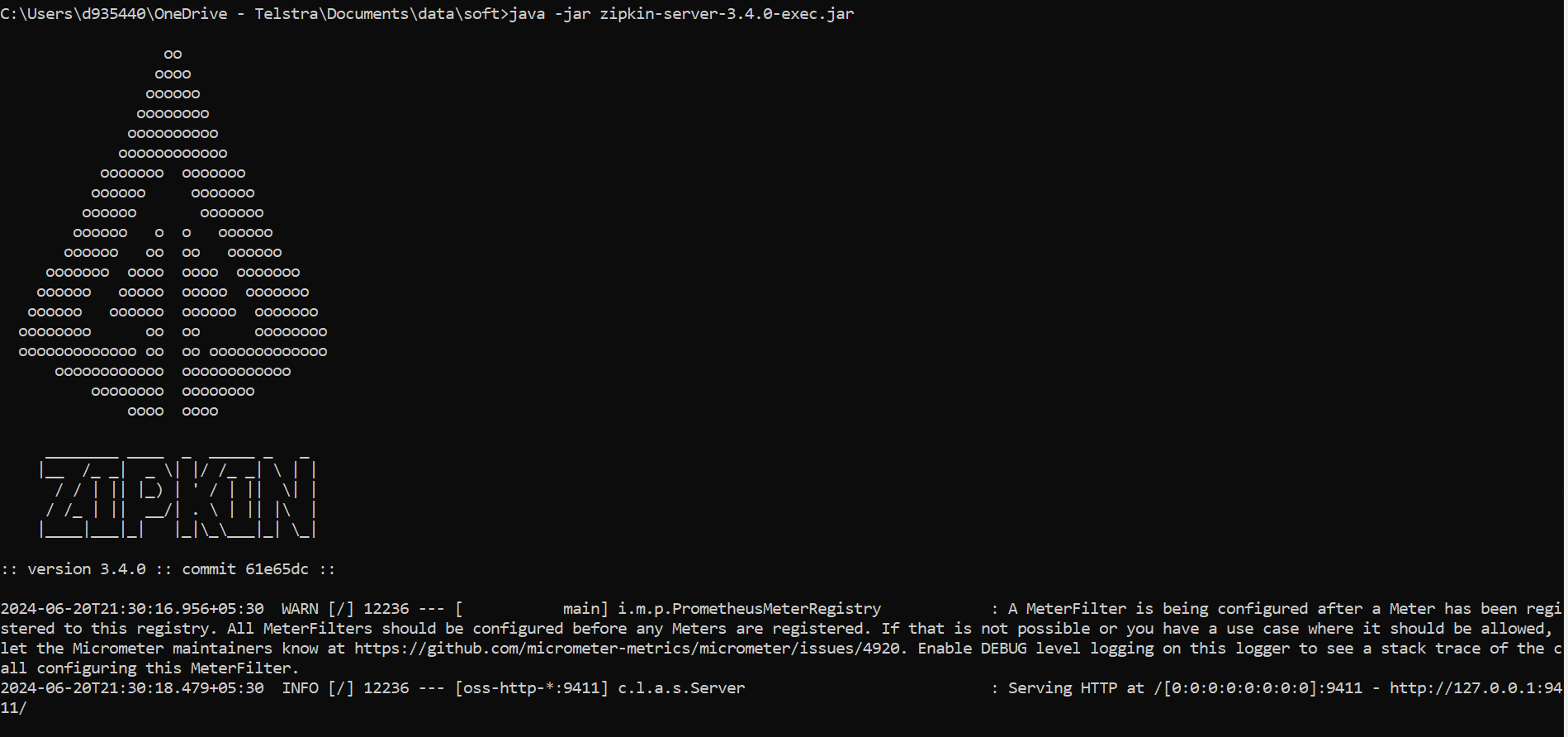
A screenshot of a computer

Description automatically generated

Steps to download and run the zipkin in local system.

1. Download the zipkin <https://zipkin.io/pages/quickstart.html>
2. Run using - java -jar zipkin-server-3.4.0-exec.jar

Zipkin is running on default port 9411

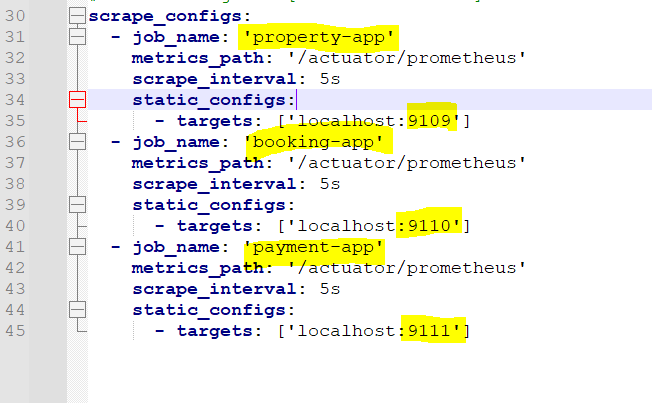


**Prometheus:**

An open-source monitoring system with a dimensional data model, flexible query language, efficient time series database and modern alerting approach

We are using Prometheus for Monitoring using Prometheus:

Number of calls made per microservice, and their response status etc can be monitored by Prometheus which can be configured using prometheus.yml as show below.



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

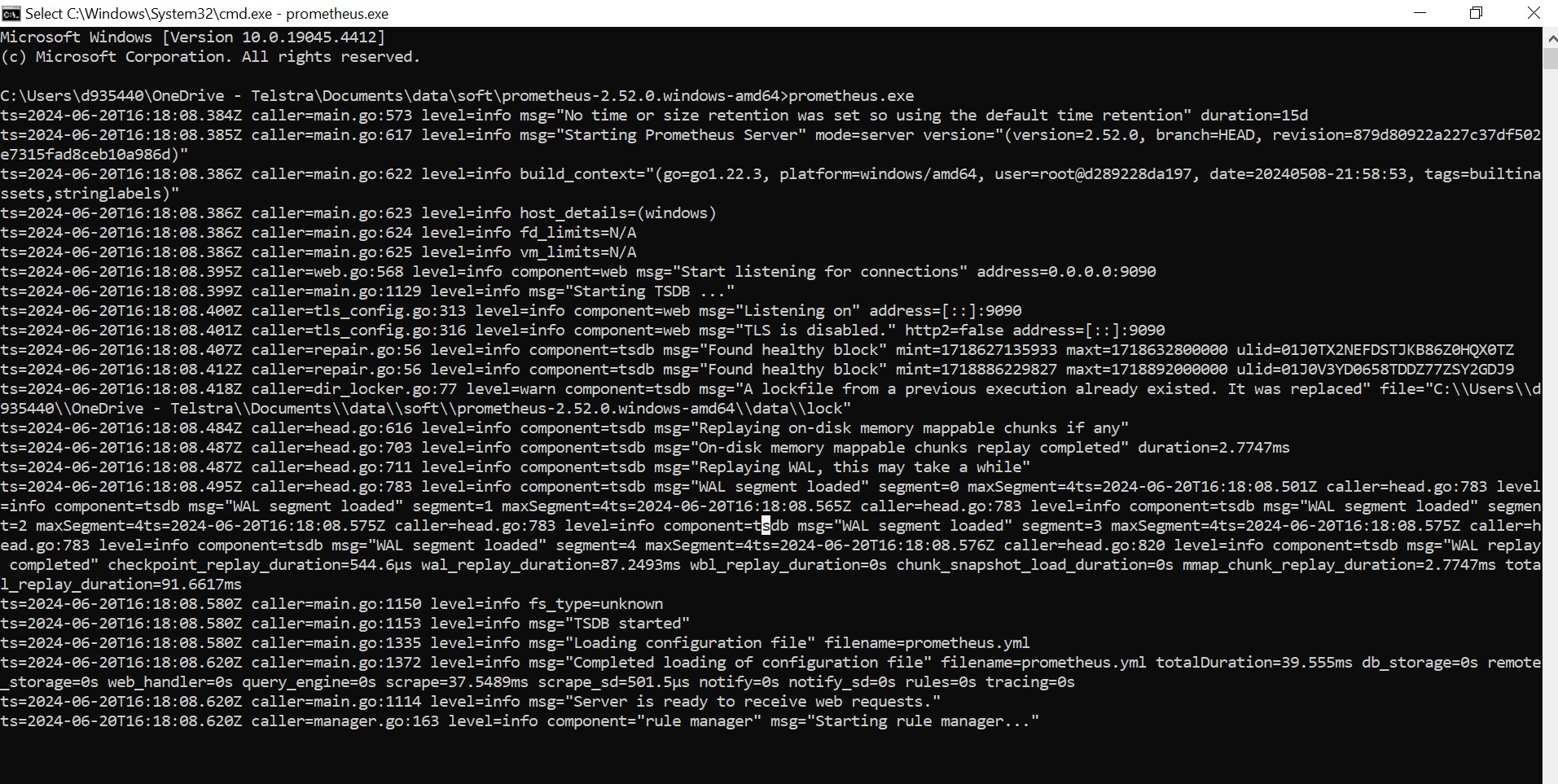
Download and run the Prometheus into local system:

1. Download from <https://prometheus.io/download/>

A screenshot of a computer

Description automatically generated

2. extract the zip file and run the Prometheus.exe



**Swagger documentation:**

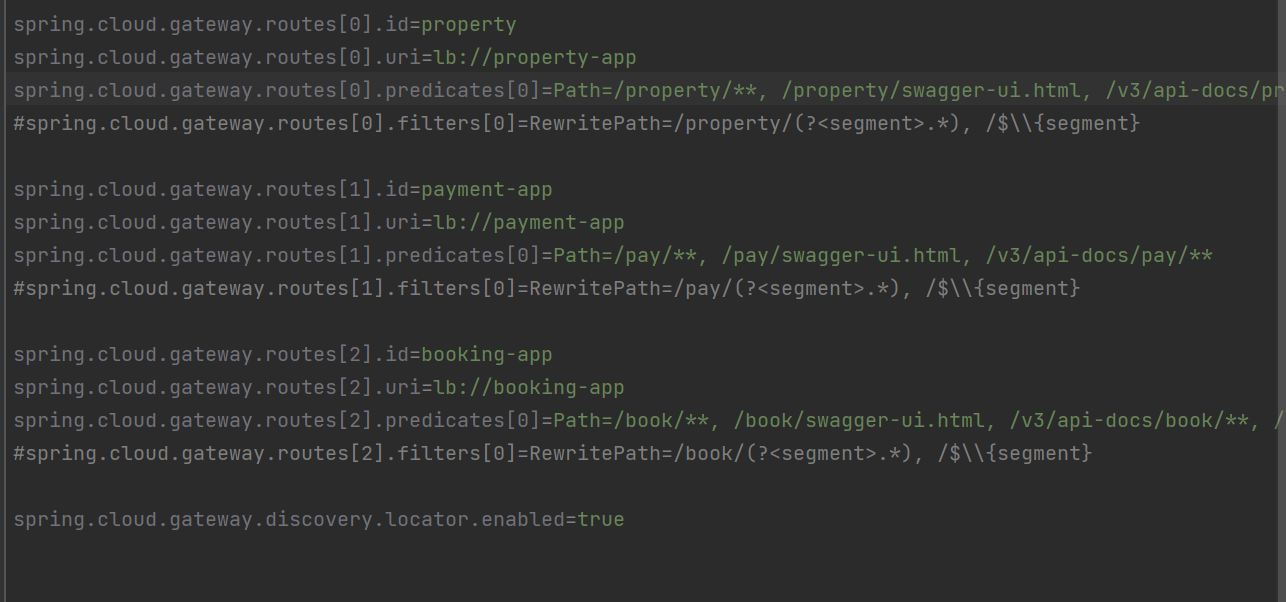
Documentation using Swagger api

All the api’s exposed and its features can be viewed in swagger documentation and even we can try out testing the rest api from here.

We need to add the dependency into the pom.xml to use the swagger api.

A screen shot of a computer program

Description automatically generated



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**UNIT TEST CASES:**

Test cases are included for microservices at the controller level as shown below:

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated