**Spring boot microservice example**

Spring boot microservice example with Eureka Server + Eureka Client + Spring Cloud API Gateway + OAuth2.0 + Circuit Breaker + Resilience4J + FeignClient + RestTemplate

* **Eureka Server** : Eureka service registry
* **Spring Cloud API Gateway**: API Gateway which is responsible to route the request to specific microservice
* **Spring OAuth2.0**: Authentication service and responsible to secure the end points.
  + Eureka Client
  + Generate OAuth token
  + Validate the OAuth token
* **User Service**: User microservice with a basic feature
  + Eureka Client
  + OAuth2.0 Client
  + Supports RestTemplate Client
  + Supports Resilience4J circuit breaker
  + Use of Resilience4J circuit breaker with RestTemplate
  + Swagger Document - <http://localhost:8181/swagger-ui/index.html>
* **Order Service**: Order microservice with a basic feature
  + Eureka Client
  + OAuth2.0 Client
  + Supports FeignClient Client
  + Supports Resilience4J circuit breaker
  + Use of Resilience4J circuit breaker with FeignClient
  + Swagger Document - <http://localhost:8282/swagger-ui/index.html>
* **Notification Service**: Notification microservice with basic feature
  + Eureka Client
  + OAuth2.0 Client
  + Swagger Document - <http://localhost:8383/swagger-ui/index.html>

**Checkout repository**

> git clone https://github.com/subhashlamba/spring-microservices.git

> cd spring-boot-microservices-example

**Step 1: Start all services**

**1.1 For windows:**

mvn clean install -f .\spring-boot-cloud-eureka-server\pom.xml

mvn clean install -f .\spring-boot-cloud-zuul-routing\pom.xml

mvn clean install -f .\spring-boot-cloud-eureka-user-service\pom.xml

mvn clean install -f .\spring-boot-cloud-eureka-order-service\pom.xml

mvn clean install -f .\spring-boot-cloud-eureka-notification-service\pom.xml

mvn clean install -f .\spring-boot-cloud-authentication-service\pom.xml

START "Server" java -jar spring-boot-cloud-eureka-server/target/eureka-server.jar

START "API Gateway" java -jar spring-boot-cloud-zuul-routing/target/zuul-api-gateway.jar --server.port=8088

START "User Service" java -jar spring-boot-cloud-eureka-user-service/target/user-service.jar --server.port=8181

START "Order Service" java -jar spring-boot-cloud-eureka-order-service/target/order-service.jar --server.port=8282

START "Notification Service" java -jar spring-boot-cloud-eureka-notification-service/target/notification-service.jar --server.port=8383

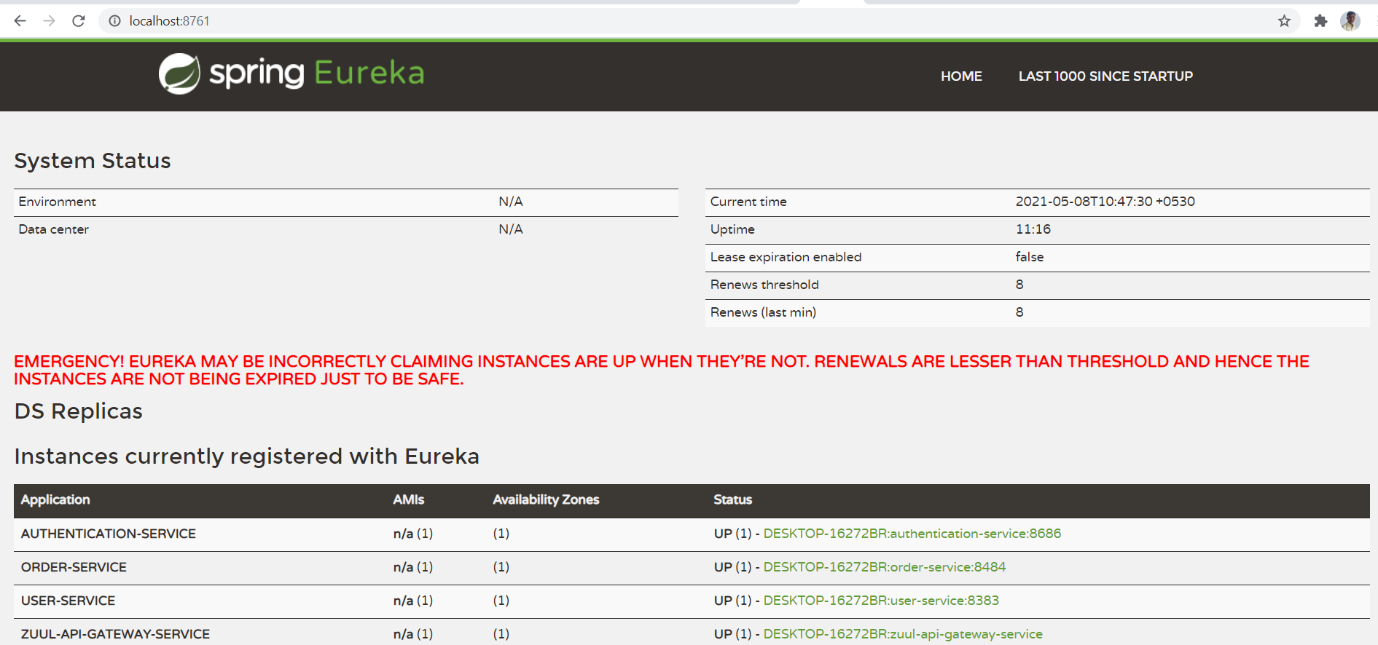
START "Authentication Service" java -jar spring-boot-cloud-authentication-service/target/authentication-service.jar --server.port=8484

**Step 2: Check Eureka Server**

Eureka server is running 8761 port, Now let's open it. Where we can check following instances are running:

* 1 instance of API Gateway service
* 1 instance of User service
* 1 instance of Order service
* 1 instance of Authentication service
* 1 instance of Notification service

**Eureka server :**[**http://localhost:8761/**](http://localhost:8761/)

[](https://github.com/ygravikiran/FORKED_spring-boot-microservice-example/blob/main/document/eureka-server.PNG)

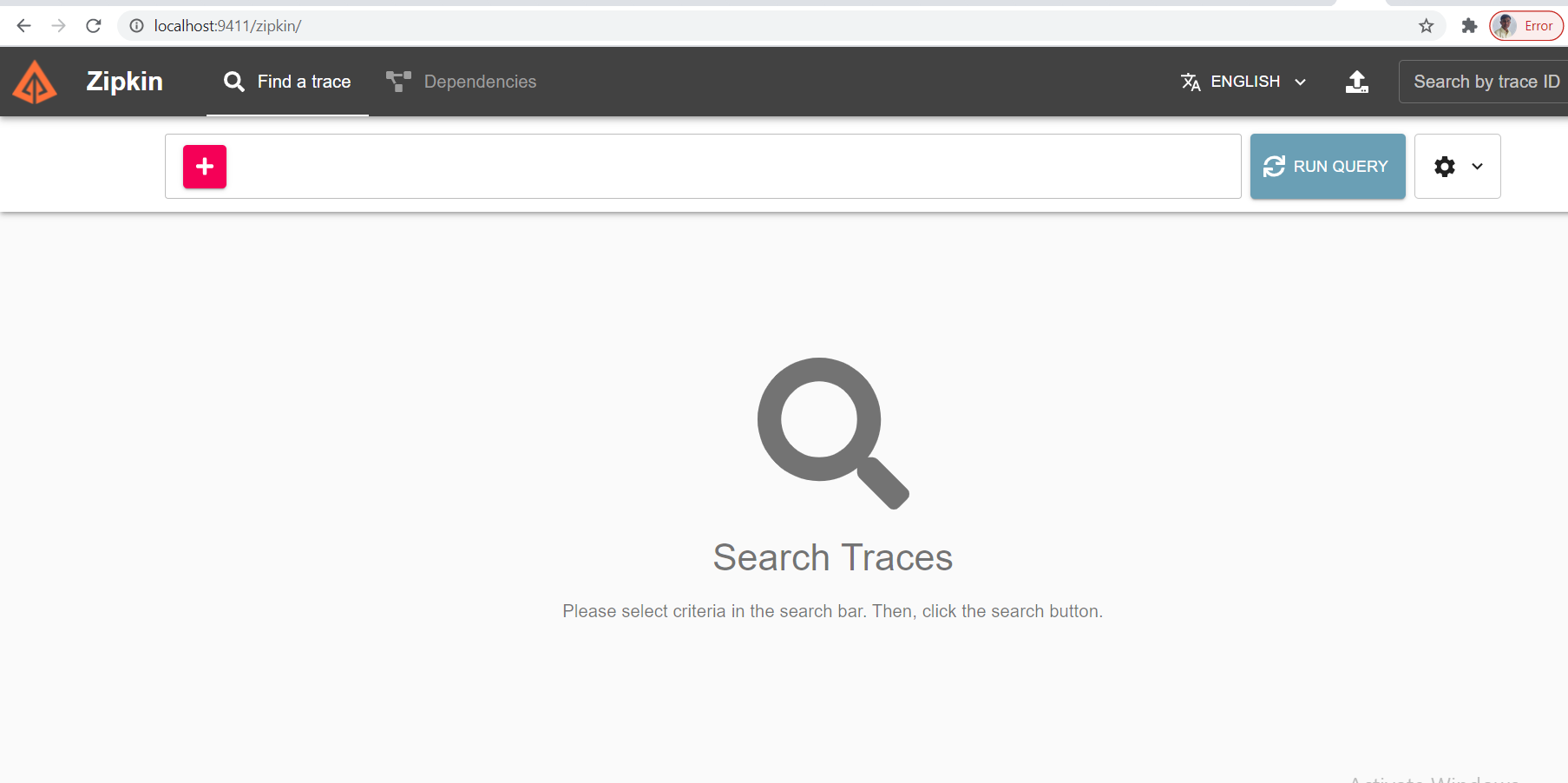
**Step 3: Configure Zipkin Server (Optional)**

1. Download the [https://zipkin.io/pages/quickstart] Download zipkin

2. Start zipkin using following command:

java -jar zipkin-server-2.23.2-exec.jar

3. Open zipkin server : [http://localhost:9411/](http://localhost:9411/)

[](https://github.com/ygravikiran/FORKED_spring-boot-microservice-example/blob/main/document/Zipkin-Server.PNG)

**Step 4: Generate OAuth2.0 token**

**4.1 Using curl**

curl -X POST \

http://localhost:8080/oauth/token \

-H 'authorization: Basic amF2YWRldmVsb3BlcnpvbmU6c2VjcmV0' \

-H 'cache-control: no-cache' \

-H 'content-type: multipart/form-data; boundary=----WebKitFormBoundary7MA4YWxkTrZu0gW' \

-H 'postman-token: d2629b30-a7cf-fa72-df64-d71118afb549' \

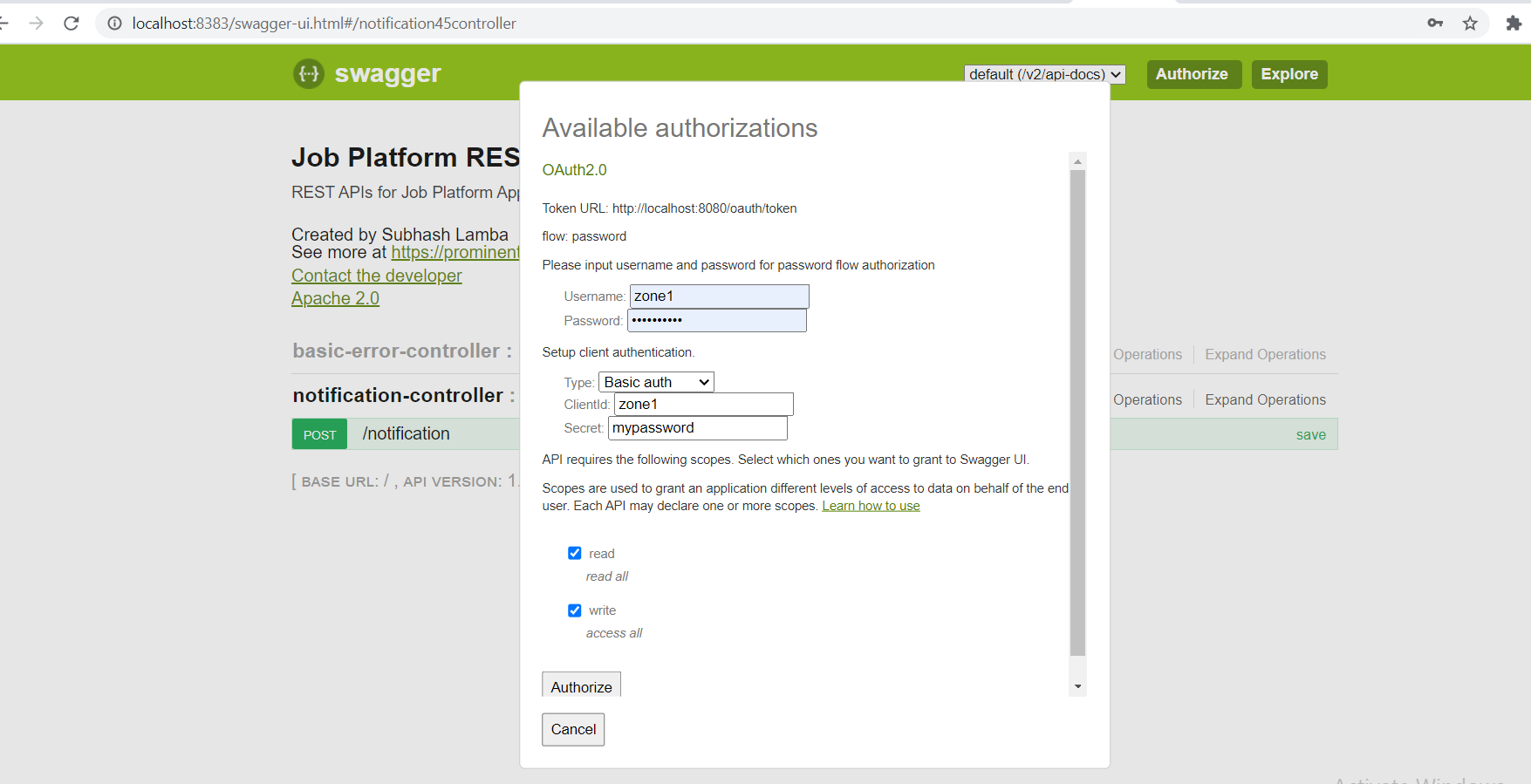
-F grant\_type=password \

-F username=zone1 \

-F password=mypassword

**4.2 Using swagger**

In the swagger UI, There is a "Autorize" button on the top, click on it it will open the following popup:

[](https://github.com/ygravikiran/FORKED_spring-boot-microservice-example/blob/main/document/Auth2-0-Generate-Token.PNG)

Use the following information

Username: zone1

Password: mypassword

ClientId: javadeveloperzone

Secret: secret

Output

{

"access\_token": "imdUX2\_t\_WQLSTUlaLBTjVyHUTg",

"token\_type": "bearer",

"refresh\_token": "zLufOQtLQO1u-8JP7KN64Dsc3wc",

"expires\_in": 522,

"scope": "read write"

}

**Step 5: Create user using user microservice**

**5.1 Using curl**

curl -X POST \

http://localhost:8080/user/ \

-H 'authorization: Bearer imdUX2\_t\_WQLSTUlaLBTjVyHUTg' \

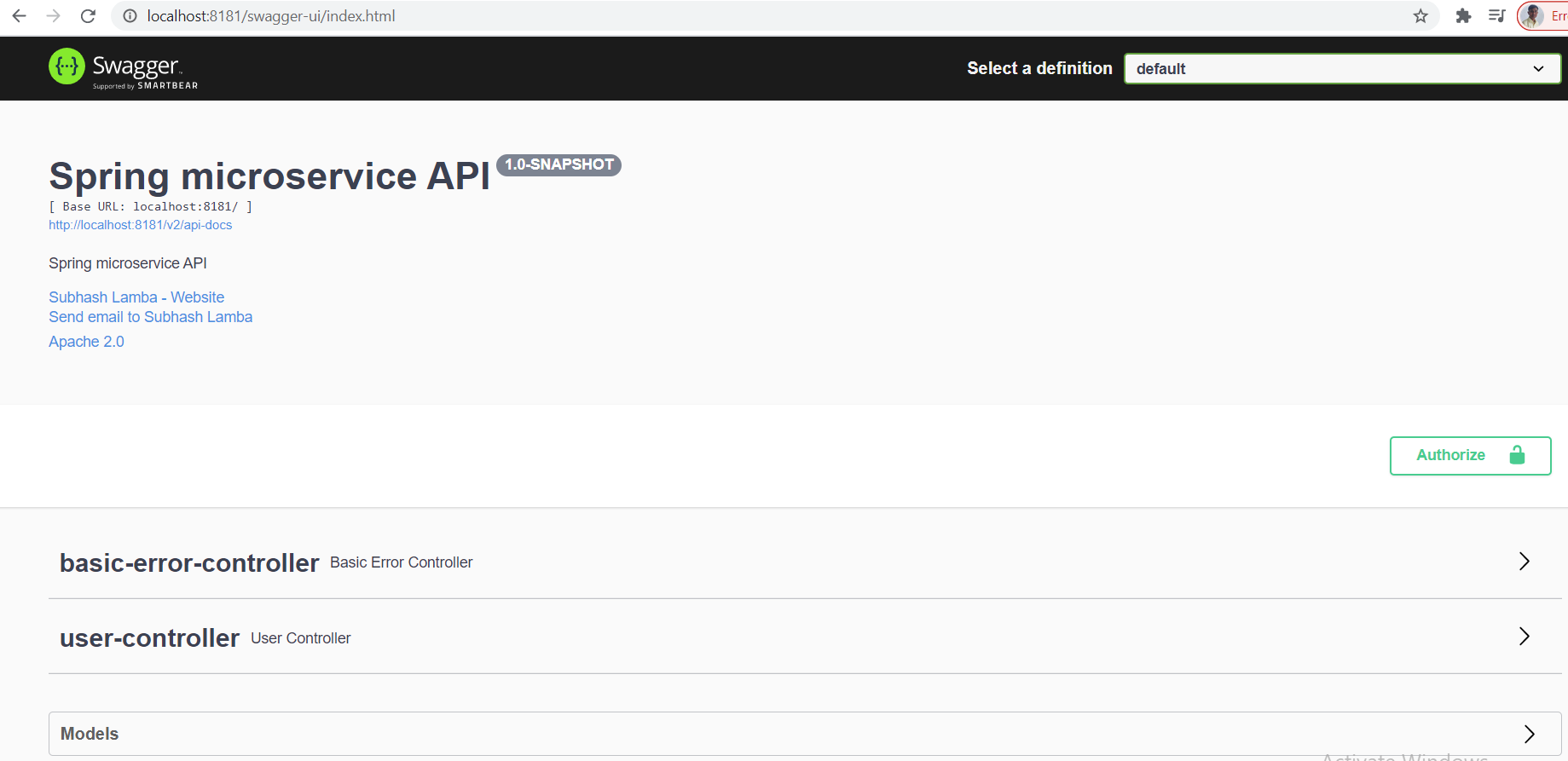
-H 'cache-control: no-cache' \

-H 'content-type: application/json' \

-H 'postman-token: d73bb63f-969c-4081-1dee-77fa2b0f1f5a' \

-d '{"firstName":"subhash", "lastName": "Lamba"}'

**5.2 Using Swagger UI**

[](https://github.com/ygravikiran/FORKED_spring-boot-microservice-example/blob/main/document/User-Microservice-Swagger.PNG)

**Step 6: Create order using order microservice**

**6.1 Using curl**

curl -X POST \

http://localhost:8080/order/ \

-H 'authorization: Bearer DF977lxKiBzdWKTkZsE7XevqK40' \

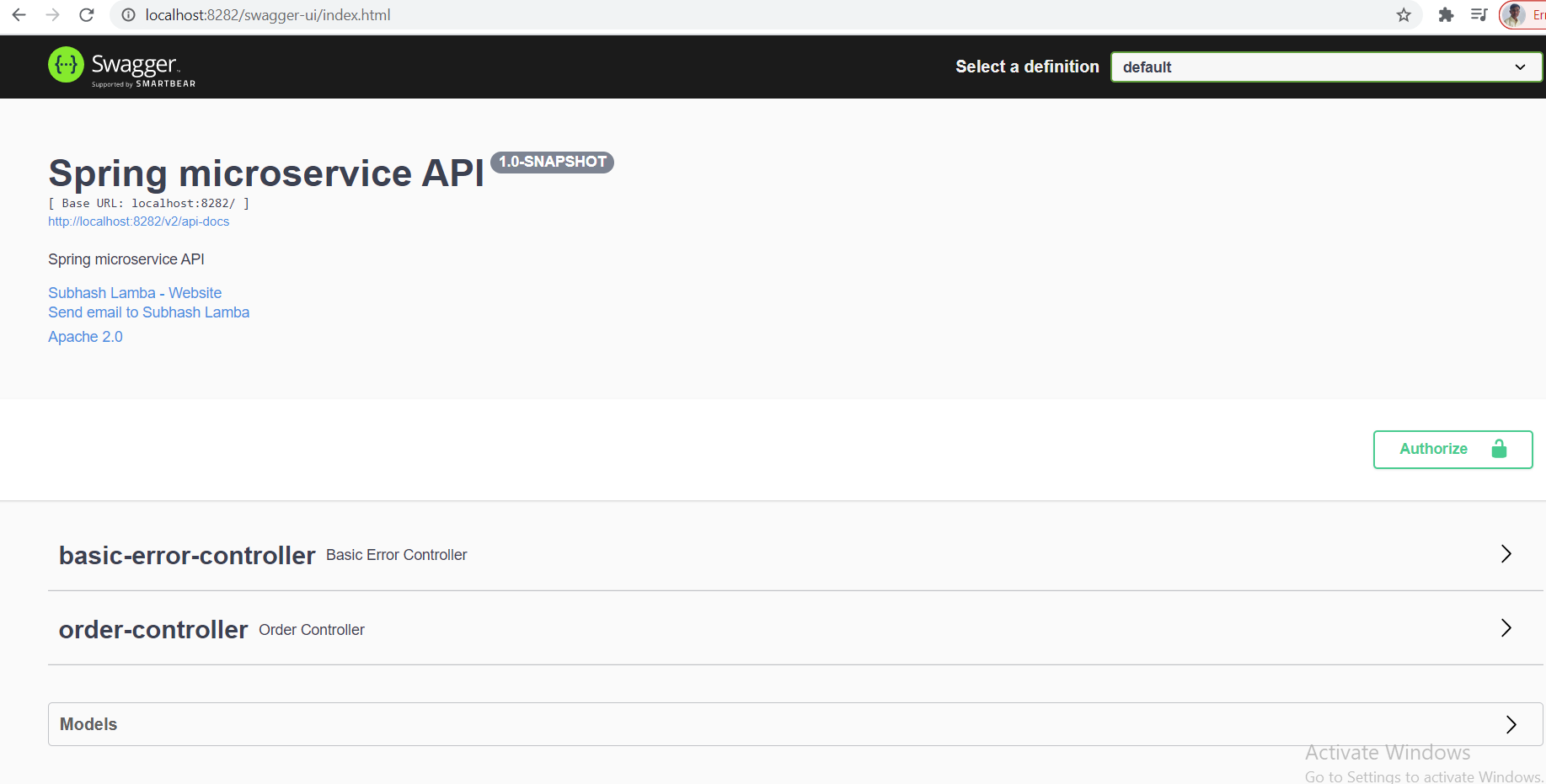
-H 'cache-control: no-cache' \

-H 'content-type: application/json' \

-H 'postman-token: 2256b83e-30dc-3b8e-26f0-8d13c0dcfd44' \

-d '{"userId":1, "orderDate": "2021-01-01"}'

**6.2 Using Swagger UI**

[](https://github.com/ygravikiran/FORKED_spring-boot-microservice-example/blob/main/document/Order-Microservice-Swagger.PNG)

**Step 7: Get user and order using user microservice**

**7.1 Using curl**

curl -X GET \

http://localhost:8080/user/1 \

-H 'authorization: Bearer DF977lxKiBzdWKTkZsE7XevqK40' \

-H 'cache-control: no-cache' \

-H 'content-type: application/json' \

-H 'postman-token: 5163fb23-f54f-c793-9faa-e9df54eab9d5' \

-d '{"userId":1, "orderDate": "2021-01-01"}'

{

"user": {

"id": 1,

"firstName": "subhash",

"lastName": "Lamba"

},

"orders": [

{

"id": 5,

"orderDate": "2021-01-01T00:00:00.000+00:00",

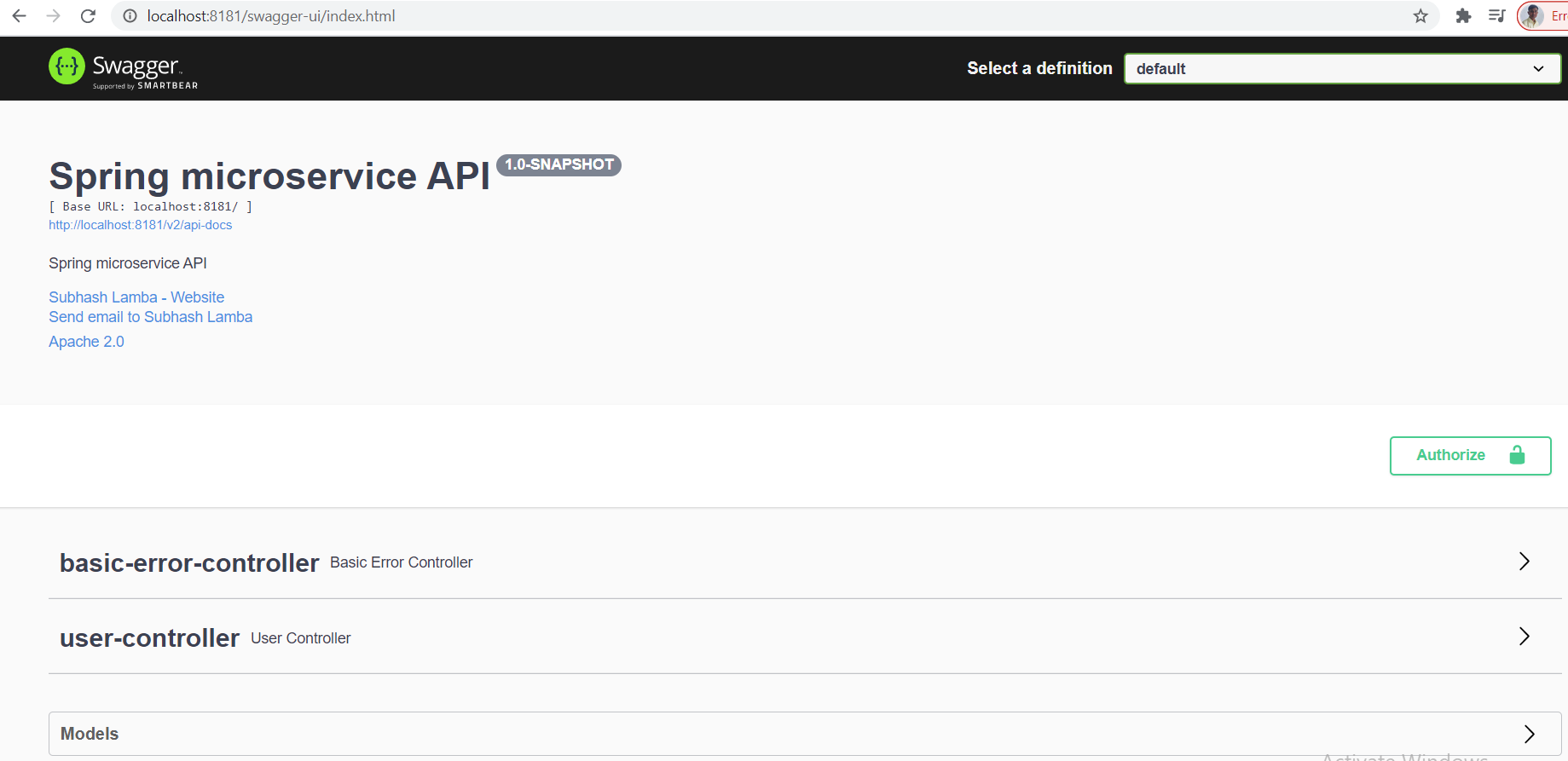
"userId": 1

}

]

}

**7.1 Using Swagger UI**

[](https://github.com/ygravikiran/FORKED_spring-boot-microservice-example/blob/main/document/User-Microservice-Swagger.PNG)