Day 10 : 25 Sep 2024



To create micro service architecture application using spring boot. Spring boot provided spring cloud modules. They provided us Server ie Eureka Server. Eureka Server is a type of registry or server which allow us to deploy more than one micro service projects. Eureka Server handle all configuration details about service, provide load balance feature, it check health of the application etc.

To implicit micro service architecture we need minimum two projects

1 Eureka Server

2. Eureka Client 1 8181

3. Eureka Client 2 8282

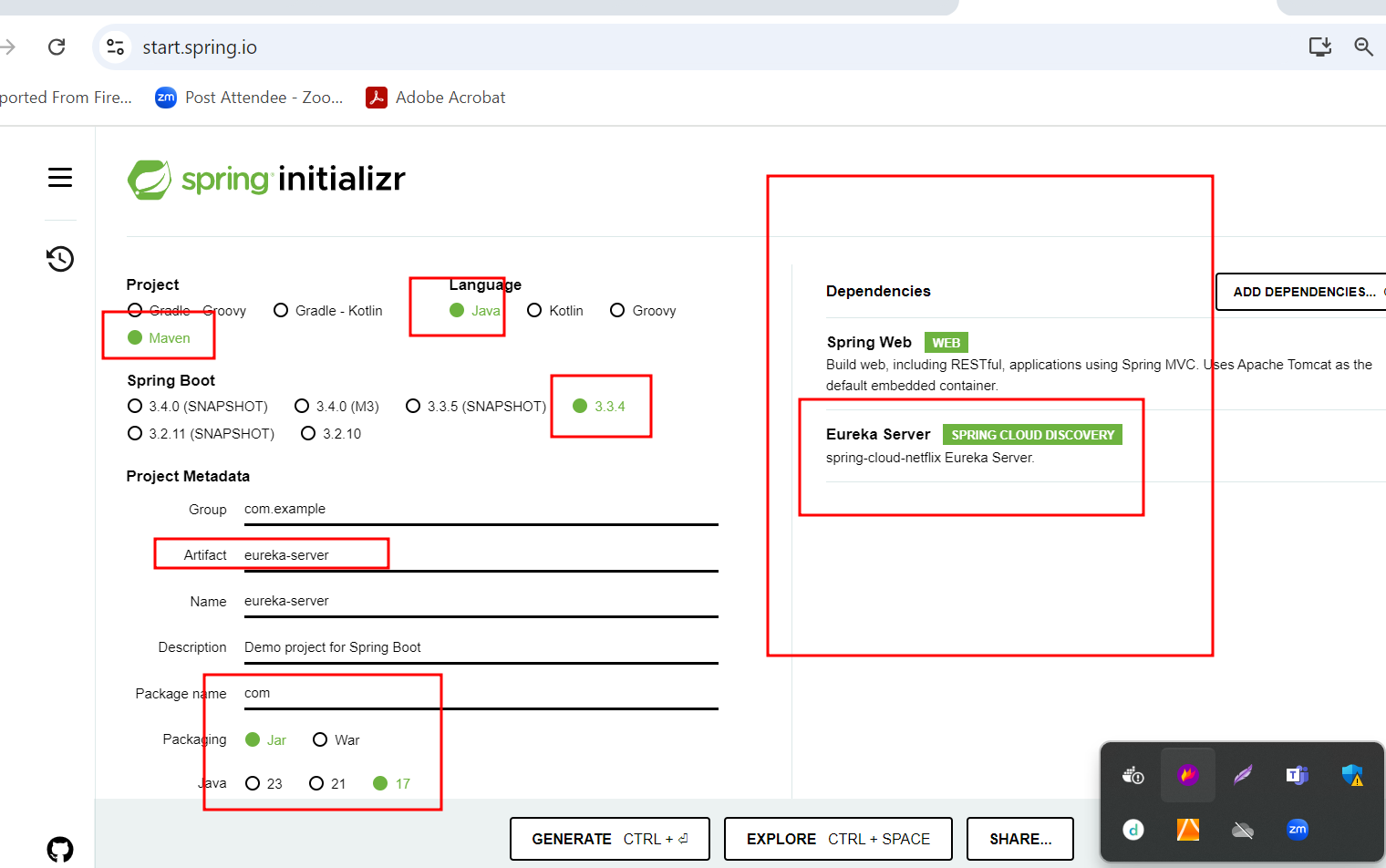
4. Eureka Client 3 8383

5. Eureka Client 4 8484

Eureka Server :

Web Starter

Eureka Starter



By default eureka server run on default port number 8761 port.

But we added web starter so eureka server run on port number 8080.

But in every micro service project we need mention eureka server port number. Because every micro service with eureka client application search eureka on port number 8761.

1. Account Micro Service 8383

Web Starter -🡪 end point

Eureka Client --🡪 help us to deploy project on Eureka Server

Jpa starter -🡪 orm features

My sql connector -🡪 mysql database

Dev tools --🡪 refresh the project

Lombok -🡪 to avoid setter and getter methods

Database 🡪 mysql

Post method : create account (accno(PK), name,amount,emailid(unique)) etc

Put method : withdrawn

Put method : deposit

Get method : using acno get amount

Get method : using emailid get the accno

1. Gpay, Paypal, Paytm etc 8484

Web starter 🡪 end point

Eureka Client --🡪 help us to deploy project on Eureka Server

Jpa starter -🡪 orm features

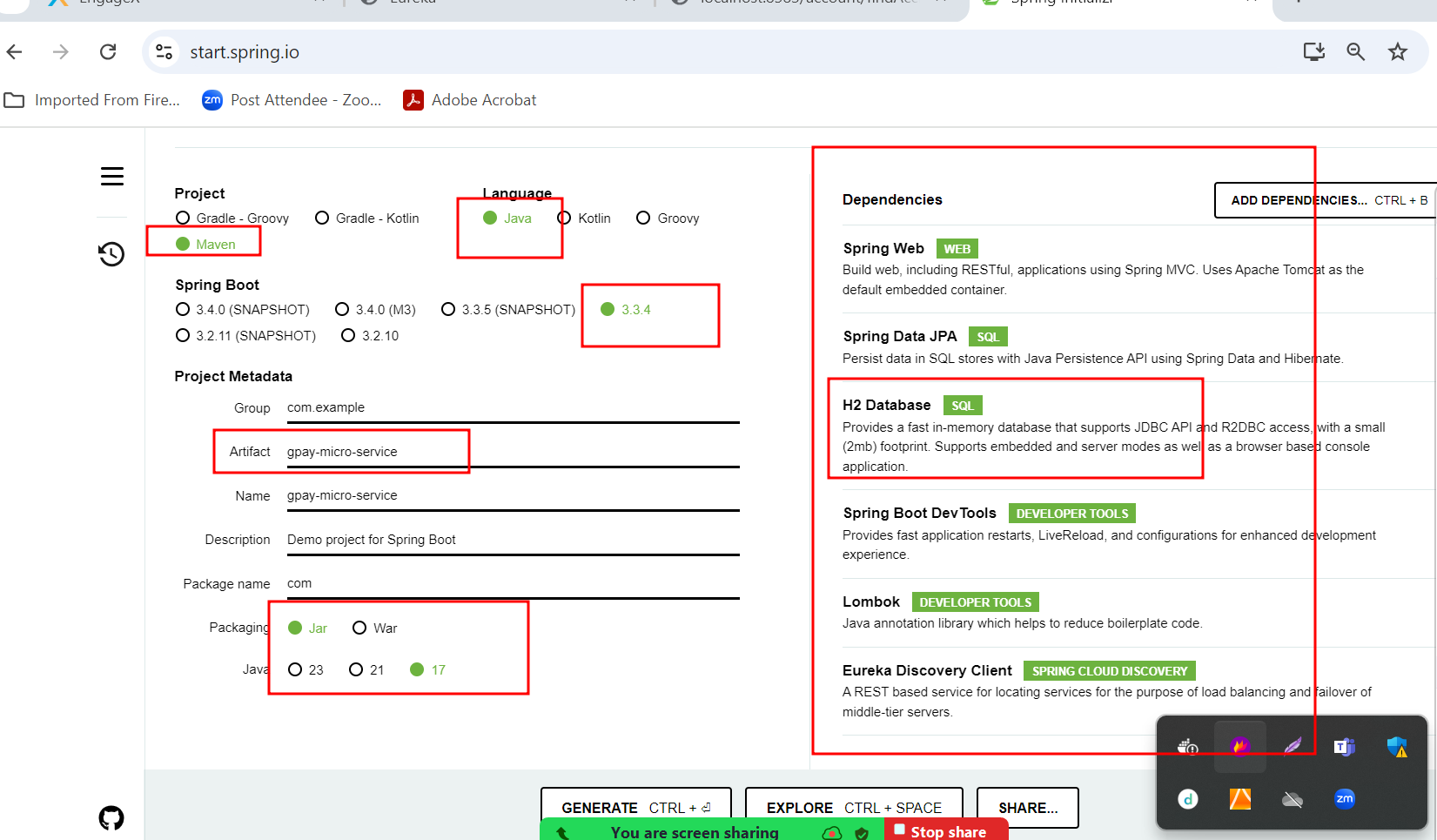
H2 connector 🡪 h2 in memory database.

Dev tools --🡪 refresh the project

Lombok -🡪 to avoid setter and getter methods

Database 🡪 h2

Post method -🡪while creating gpay,(gpaid auto increment, accno) from client side we ask emailid using emaild we interact with account micro service to get the account details. If present we create gpay account



In Spring boot we can communicate to another rest api or micro service using

One of the way RestTemplate , WebClient or FeignClient

Eureka Server (server machine )

198.166.45.32

Account Micro Service Gpay Micro Service

198.134.56.78 :8383 197.23.456.23:8484

We invoke account micro service through Gpay micro service using RestTemplate

PayPal micro service which test Account Micro service or Gpay Micro service using

Feign Client.

Feign Client is type of starter provided by spring boot which help to invoke rest api in declarative manner.

This micro service run on port number 8585 to get balance amount using accno number.

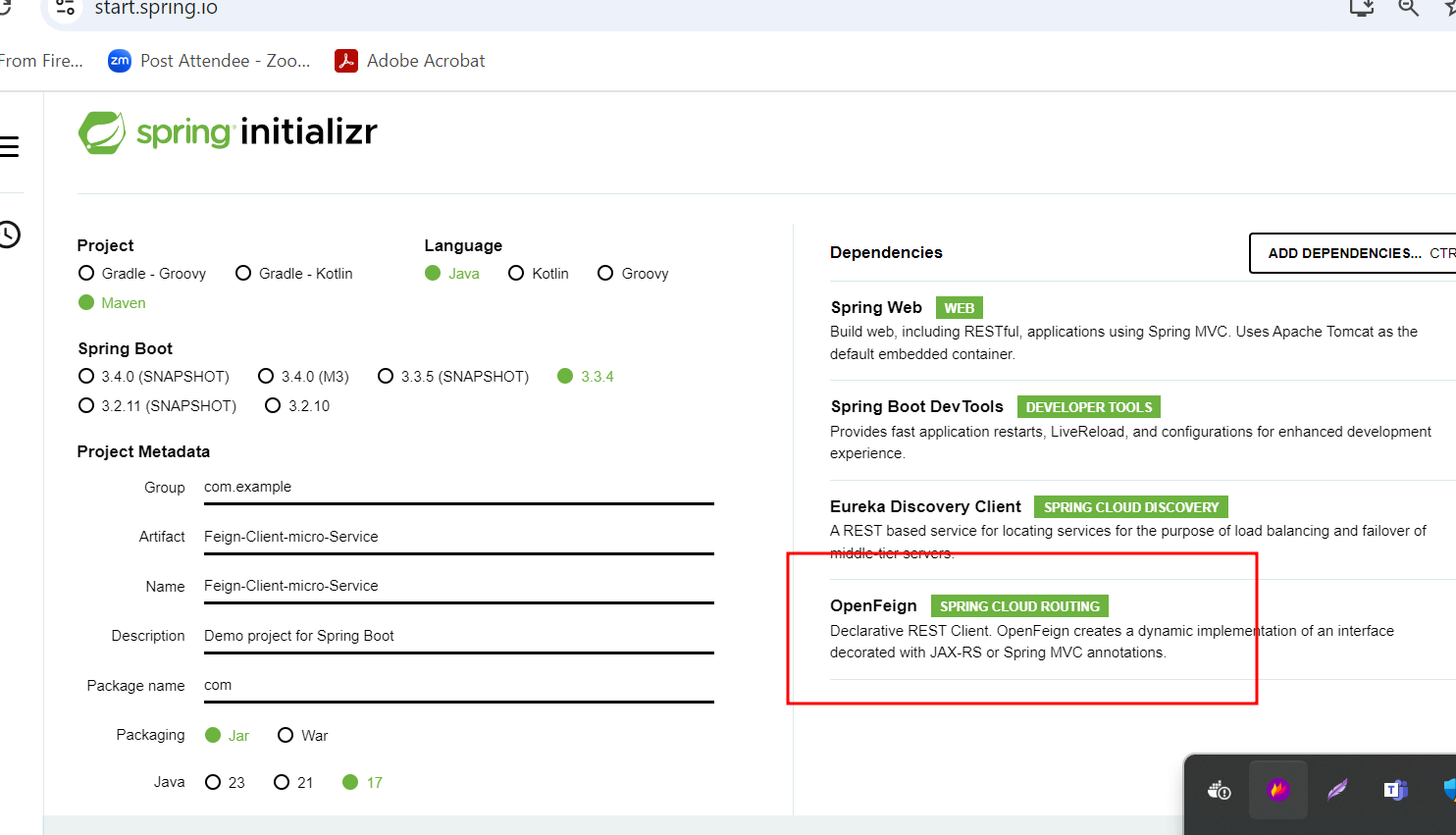
Starter

Web starter

Feign client starter

Dev tools

Eureka Client



Problem Statement

**Background of the problem statement:  
Sporty Shoes is a company that manufactures and sells sports shoes. They have a walk-in store, and now, they wish to launch their e-commerce portal sportyshoes.com.  
   
You’re asked to develop a prototype of the application. It will be then presented to the relevant stakeholders for budget approval. Your manager has set up a meeting where you’re asked to do the following:   
● Presenting the specification document which has the product’s capabilities, appearance, and user interactions ● Setting up Git and GitHub account to store and track your enhancements of the prototype  ● Explaining the Java concepts used in the project  ● Discussing the generic features of the product: ● There will be an admin to manage the website. An administrator login will be required to access the admin**

**Sporty Shoes  E-App**

**Thymeleaf starter View -🡪Thymeleaf**

**Web starter --🡪@Controller**

**MysqL connector 🡪 MySQL Database**

**JPA Starter -🡪 Spring JPA Data**

**Devtools -🡪**

**Two type of user**

Login entity emailid(PK), password and type of user etc.

Login.html

Two types of user

Admin : for admin no signup

[admin@gmail.com](mailto:admin@gmail.com) and password admin@123

admin can do Sign in

if success open admin home page.

Add/delete/update/view Sport shoe information

SID, brand, type of shoe, price, qty, image etc

View order information for all customer

They can filter the product using branch, price etc.

Customer or normal user for customer signup as well as signin

After signin customer open the home page.

View the Shoe information using image

Price

You need to place the order for the specific product or more than one product.

Customer can view its own order.

inside main method

@autowired

LoginRepository loginRepository

@PostConstruct

public void init() {

login ll = new Login([admin@gmail.com,”admin@123](mailto:admin@gmail.com,”admin@123)”);

loginRepository.save(ll);

}

Push this project in your personal public github account.

1st Document :

Short description about your projects and your remote repository URL.

2nd document

Screen short for your working project.