e-shopping app

frontend ---🡪 angular application

backend -🡪 spring boot with micro service concept to develop the rest api

the communication between front end and backend is rest api

course-3-shopping-app

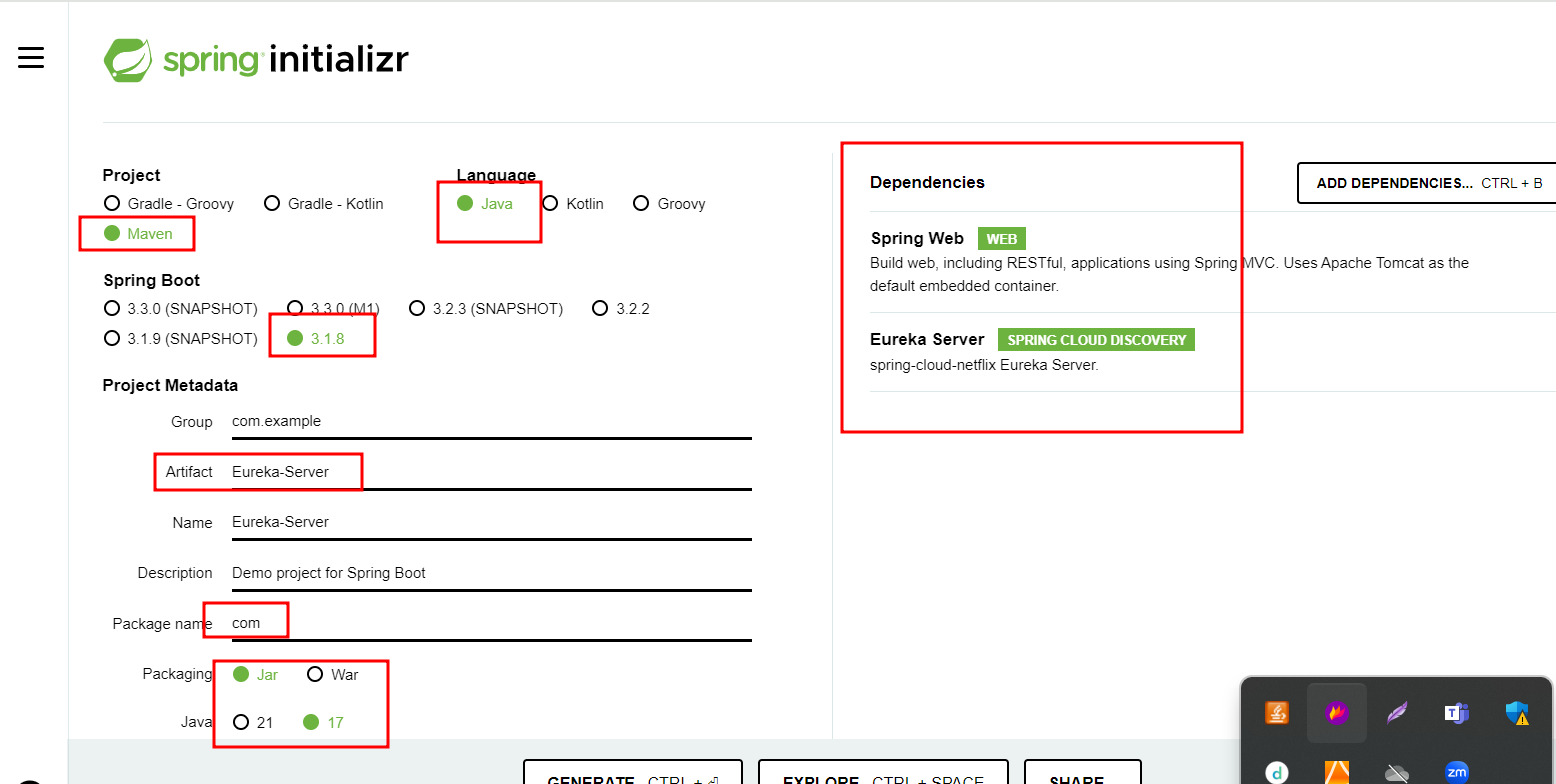
frontend backend folder

backend

1st

Project Eureka – Server

With starter web and eureka server : port number 8761



application.properties

server.port=8761

eureka.client.register-with-eureka=false

eureka.client.fetch-registry=false

2nd database caltech\_course3\_eshopping\_app

Project login-app (admin and customer with signIn and signUp)

With starter port 8181 with database

Web

Eureka client

Jpa

Mysql

Dev tool

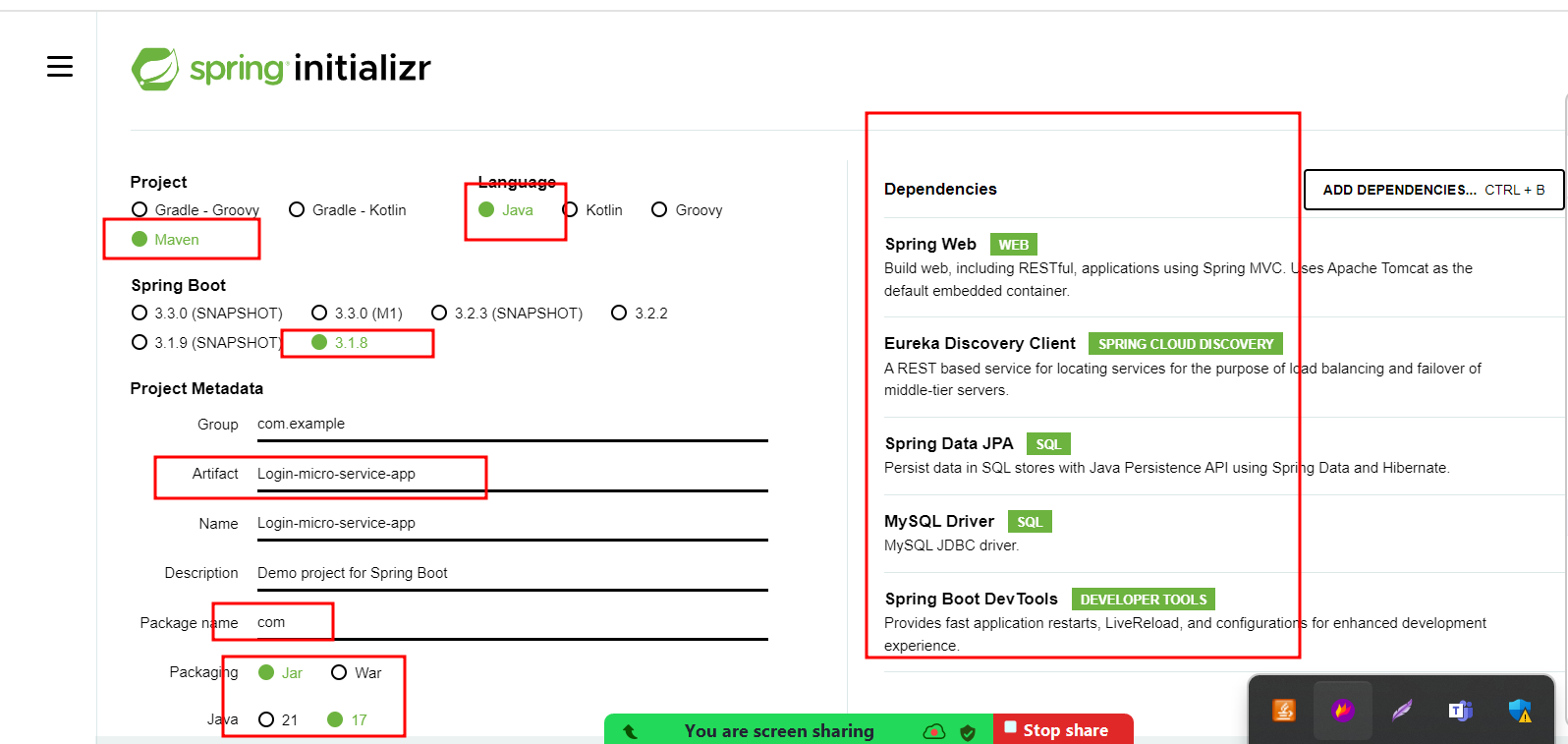
2nd rest api

emailId, password and typeofuser

Admin user only Signin and for customer SignIn and SignUp

SignIn <http://localhost:8181/login/signIn> post with json data

SignUp <http://localhost:8181/login/signUp> post with json data



3rd database caltech\_course3\_eshopping\_app

Project product\_app

With starter port 8282 with database

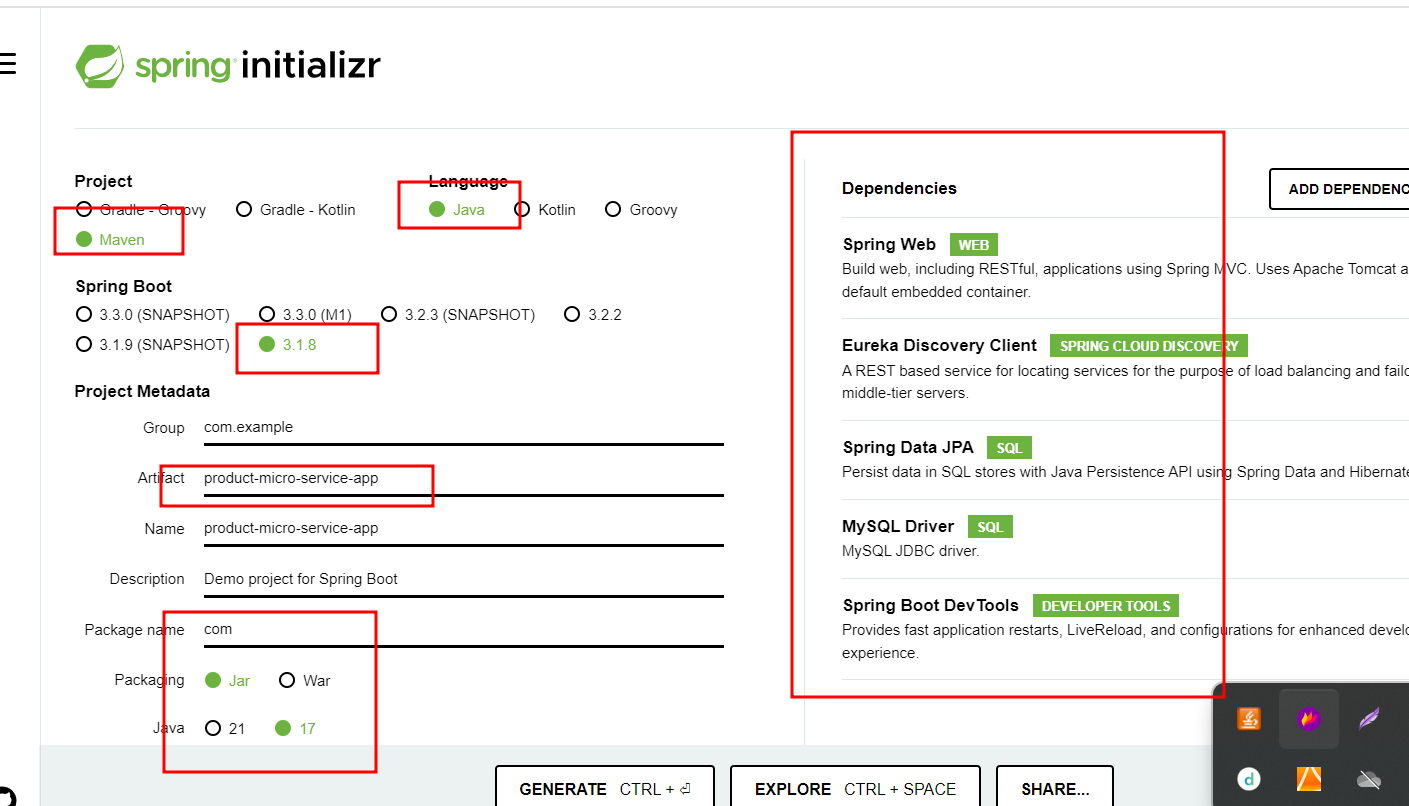
Web

Eureka client

Jpa

Mysql

Dev tool



<http://localhost:8282/product/addProduct>

<http://localhost:8282/product/viewProduct>

<http://localhost:8282/product/deleteProduct>

<http://localhost:8282/product/updateProduct>

Frontend

ng new e-shopping-app

routing -🡪 yes

styling -🡪 css

with bootstrap features.

ng g c login component

ng g c signup component

ng g class login model class

ng g s login service class

ng g c dashboard component not using

ng g c adminhome component

we need to allow to all call rest api of product ie

addProduct, deleteProduct, viewProduct and updateProduct

ng g c productbyadmin

add product, deleteproduct, updateproduct retrieveproduct

ng g c customerhome component

viewProduct

ng g c productbycustomer

retrieveproduct

ng g class product

ng g s product

product model product entity product table

angular model class 🡨-🡪 Java bean or entity class 🡨🡪 Table in mysql

Variable name and column name must match

Cors : Cross Origin Resource Sharing

Two domain or server are communicating with each other.

4200 8181

Web server tomcat server

Backend technologies we need to allow to access resources.

WSDL : Web Service description language SOAP web service

WADL Web Application description language rest web service