WIPRO CAPSTONE PROJECT
SHOP FOR HOME
GROUP A – G3

Submitted in the partial fulfillment of the requirements
for Wipro certification
in
JAVA FULL STACK DEVELOPMENT

Submitted by:

Vilupuru Jaswanth Reddy
Ravendra Kumar Baghel
Basavaraj
Ratnakaram Sai Jaya Chandra Raju

ABSTRACT

Shop Free or shop for home ordering Platform is a form of Household e-commerce which allows consumers to directly buy goods or services from a seller over the internet using a web browser. Consumers find a product of interest by visiting the website of the retailer directly or by searching among alternative vendors using a shopping search engine, which displays the same products availability and pricing at different e-retailers. As of 2020, customers can shop online using a range of different Household appliances. Users can login to the website to shop the products for home decoration. It has the features of add to cart, wishlist and make orders from home. Typical online store enables the customer to browse the firms range of products and services.

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

This project Shop for Home has been developed in Angular, Java Spring Boot and MySql. The Shop for Home project is an application which is based on managing the Home-decor and sell the household items online. The main purpose for developing this project in Angular and Java Spring Boot is to manage all the details about home decor item, home decor category, company, order, sales etc. There are two categories of user available in this project. First one is admin and second one is user. Admin can add home-décor item category and can also manage the sales details. This Angular project very helpful for maintaining the sales activity in household item. In this project customer can see the details of home-decor item category etc. Only Admin can edit or delete the details of Home-decor item.

1.1 PURPOSE

The main purpose of this module is to provide all the functionality related to customers. It tracks all the information and details of the customer. We have developed all Category of CRUD (Create, Read, Update and Delete) operations of the customers. This is a role-based module where admin can perform each and every operations on data but the customer will be able to view only his/her data, so access level restrictions has also been implemented on the project. We also provide customized Angular and Java Spring Boot" Projects for beginners.

1.2 User-Requirements:

1. As a user I should be able to login, Logout and Register into the application.
2. As a user I should be able to see the products in different categories.

3. As a user I should be able to sort the products.
4. As a user I should be able to add the products into the shopping cart.
5. As a user I should be able to increase or decrease the quantity added in the cart.
6. As a user I should be able to add “n” number of products in the cart.
7. As a user I should be able to get the Wishlist option where I can add those products which I want but don’t want to order now
8. As a user I should get different discount coupon

1.3 Admin-Requirements:

1. As an Admin I should be able to login, Logout and Register into the application.
2. As an Admin I should be able to perform CRUD on Users.
3. As an Admin I should be able to Perform CRUD on the products.
4. As an Admin I should be able to get bulk upload option to upload a csv for products details
5. As an Admin I should be able to get the stocks.
6. As an Admin I should be able to mail if any stock is less than 10.
7. As an Admin I should be able to get the sales report of a specific duration.
8. As an Admin I should be able to set of users

CHAPTER-2

TECHNOLOGIES REQUIRED

We have developed this project using the below technology

- **HTML**: Page layout has been designed in HTML
- **CSS**: CSS has been used for all the designing part
- **JavaScript**: All the validation task and animations has been developed by JavaScript
- **Java Spring Boot**: All the business and backend API logic has been implemented in Java Spring Boot
- **.SQL**: .SQL files has been used as database for the project
- **Angular:** All the frontend logic has been implemented over the Angular and we used angular CLI for it
- **Visual Studio Code-(VSS**): For Angular IDE, we have used Visual Studio Code
- **STS**: We have used STS (Spring Tool Suite) for developing all spring boot API's
- **Tomcat**: Project will be run over the Tomcat server

2.1 Supported Operating System

We can configure this project on following operating system.

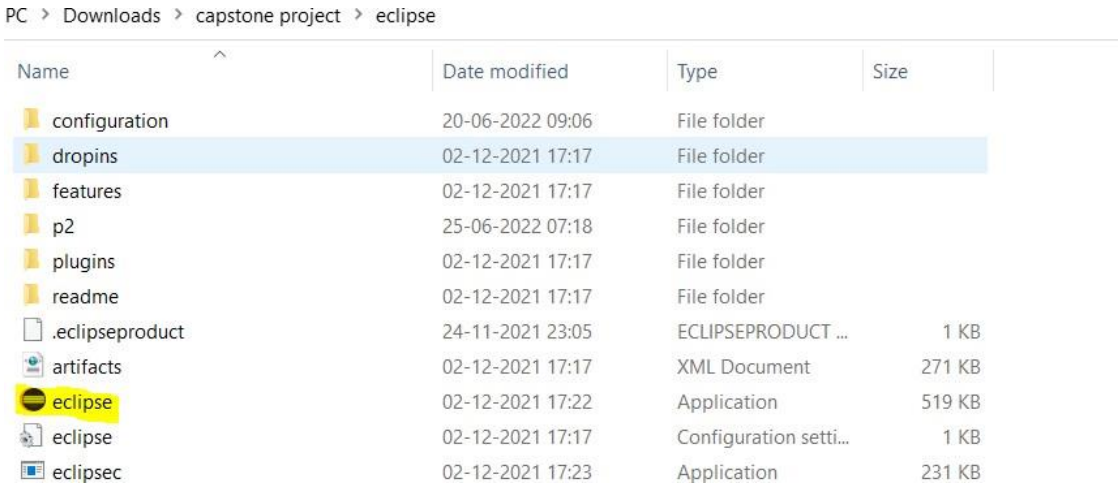
- **Windows**: This project can easily be configured on windows operating system.

CHAPTER-3

INSTALLATION AND INSTRUCTIONS

3.1 Installation Steps:

1. Java 11 jdk -> double click -> proceed with default settings
2. postgresSql -> double click -> proceed with default settings
    a. Search -> pgAdmin after postgres installation completes
3. Vscode -> double click -> proceed with default settings
4. Eclipse -> extract zip file -> run



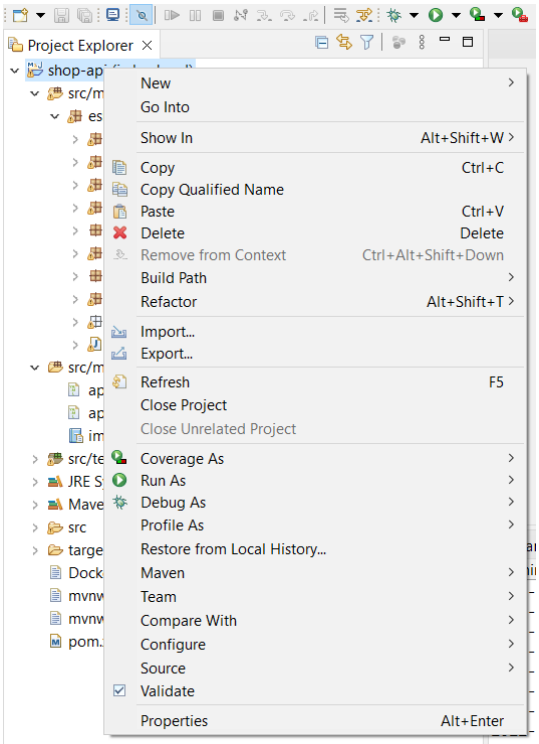
a.

b. double click -> eclipse application -> (pin to task bar if required)

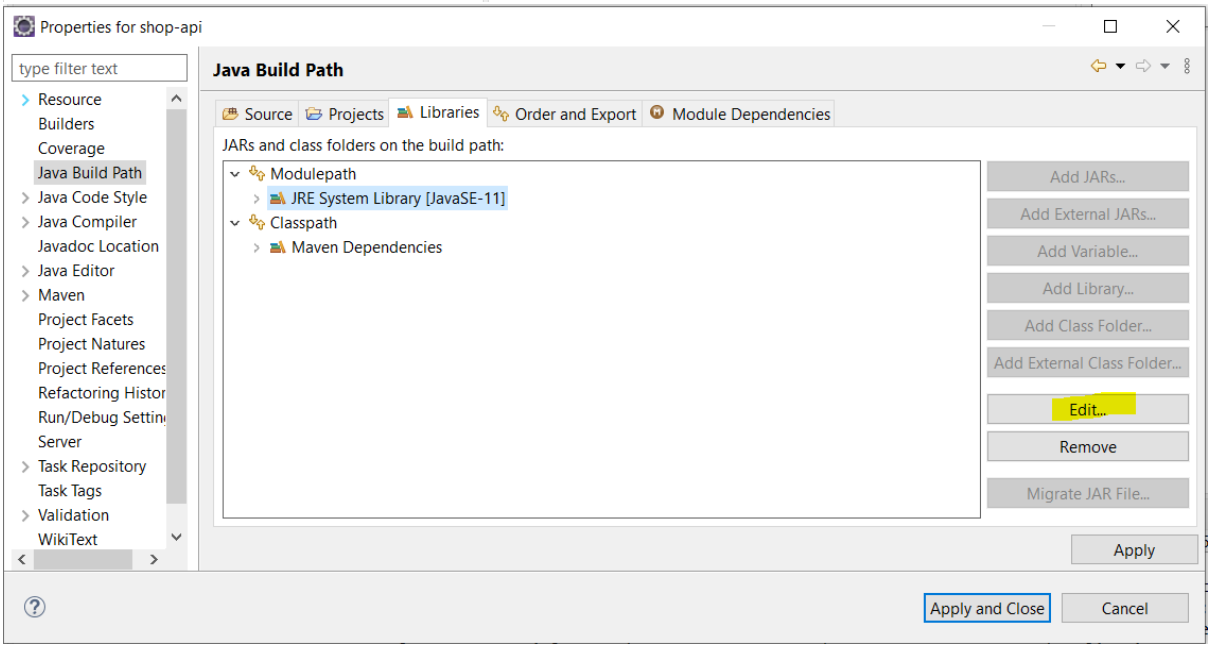
3.2 Import Backend project in Eclipse:

1. Create Eclipse workspace
2. Import Backend Project in workspace
3. Change properties -> right click -> properties

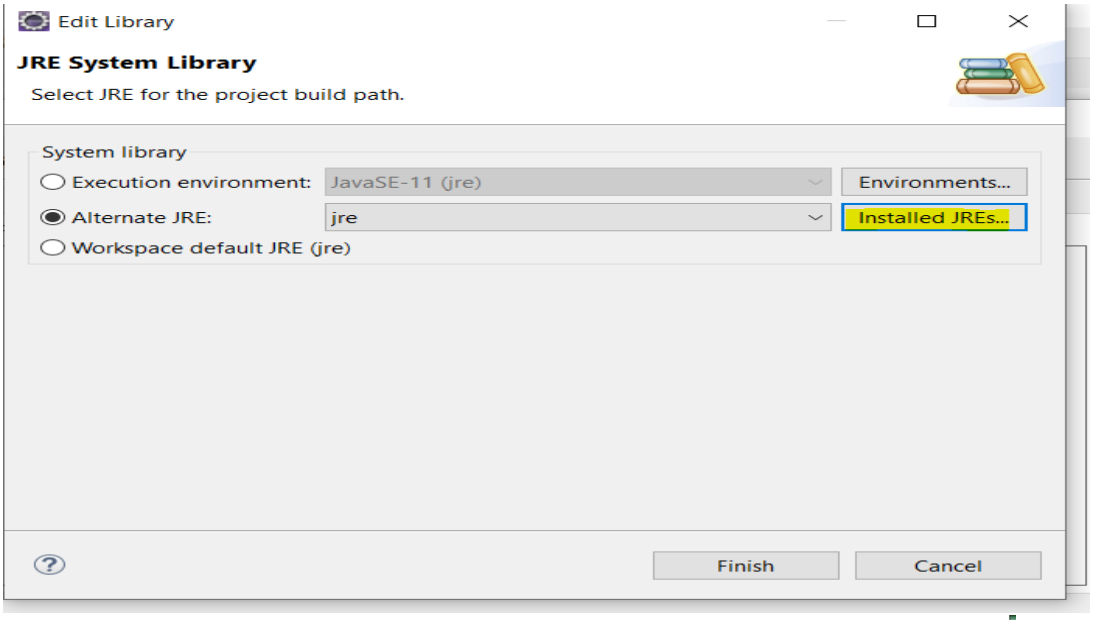
4.


Project Explorer
New
Go Into
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Refactor Alt+Shift+T
Import...
Export...
Refresh F5
Close Project
Close Unrelated Project
Coverage As
Run As
Debug As
Profile As
Restore from Local History...
Maven
Team
Compare With
Configure
Source
Validate
Properties Alt+Enter

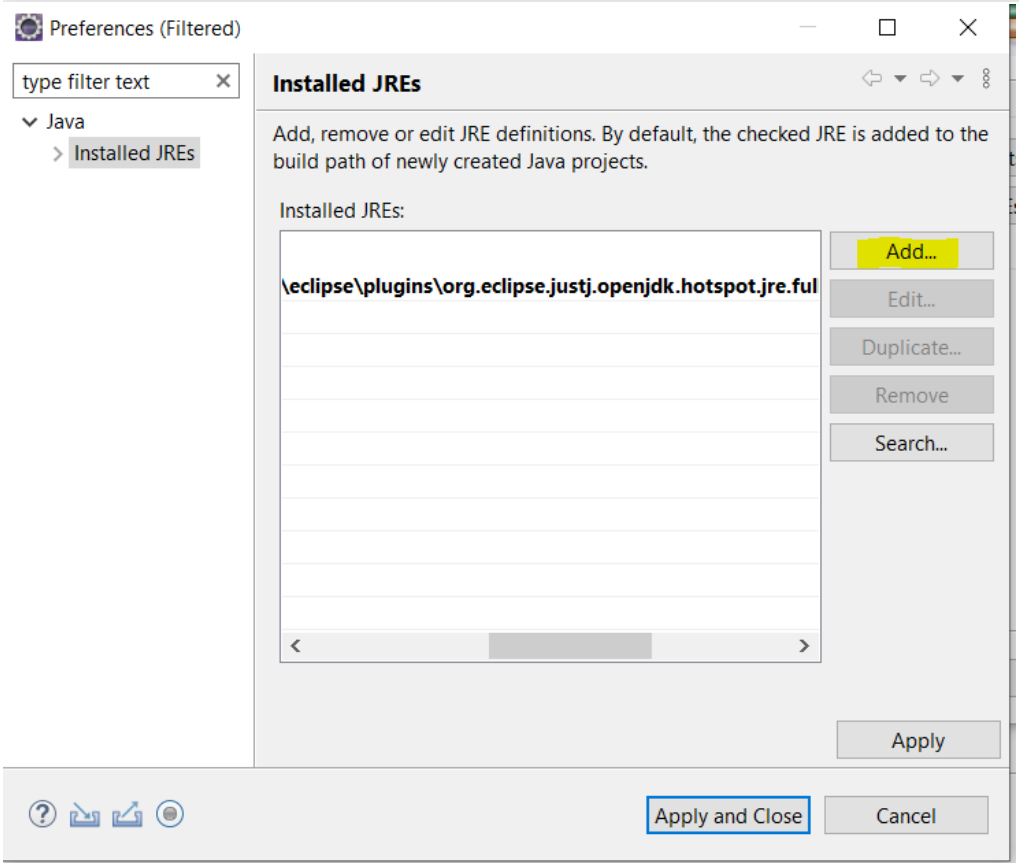
5.

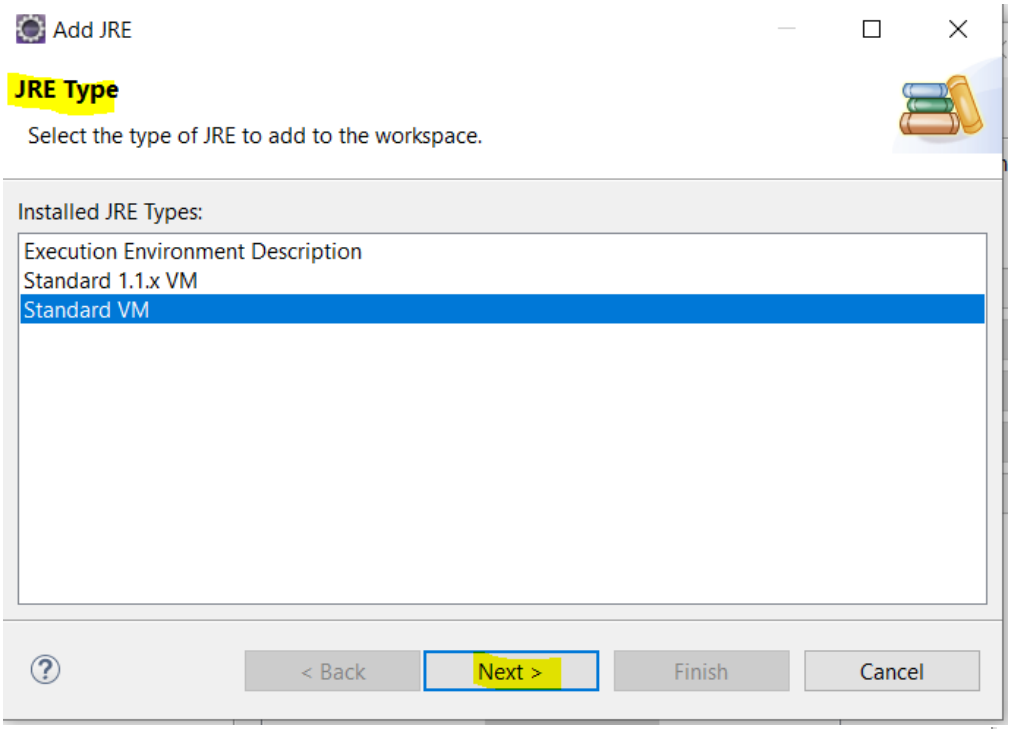

Properties for shop-api
type filter text
Resource
Builders
Coverage
Java Build Path
Java Code Style
Java Compiler
Javadoc Location
Java Editor
Maven
Project Facets
Project Natures
Project References
Refactoring Histor
Run/Debug Settin
Server
Task Repository
Task Tags
Validation
WikiText
Java Build Path
Source
Projects
Libraries
Order and Export
Module Dependencies
JARs and class folders on the build path:
Modulepath
JRE System Library [JavaSE-11]
Classpath
Maven Dependencies
Add JARs...
Add External JARs...
Add Variable...
Add Library...
Add Class Folder...
Add External Class Folder...
Edit...
Remove
Migrate JAR File...
Apply
Apply and Close
Cancel

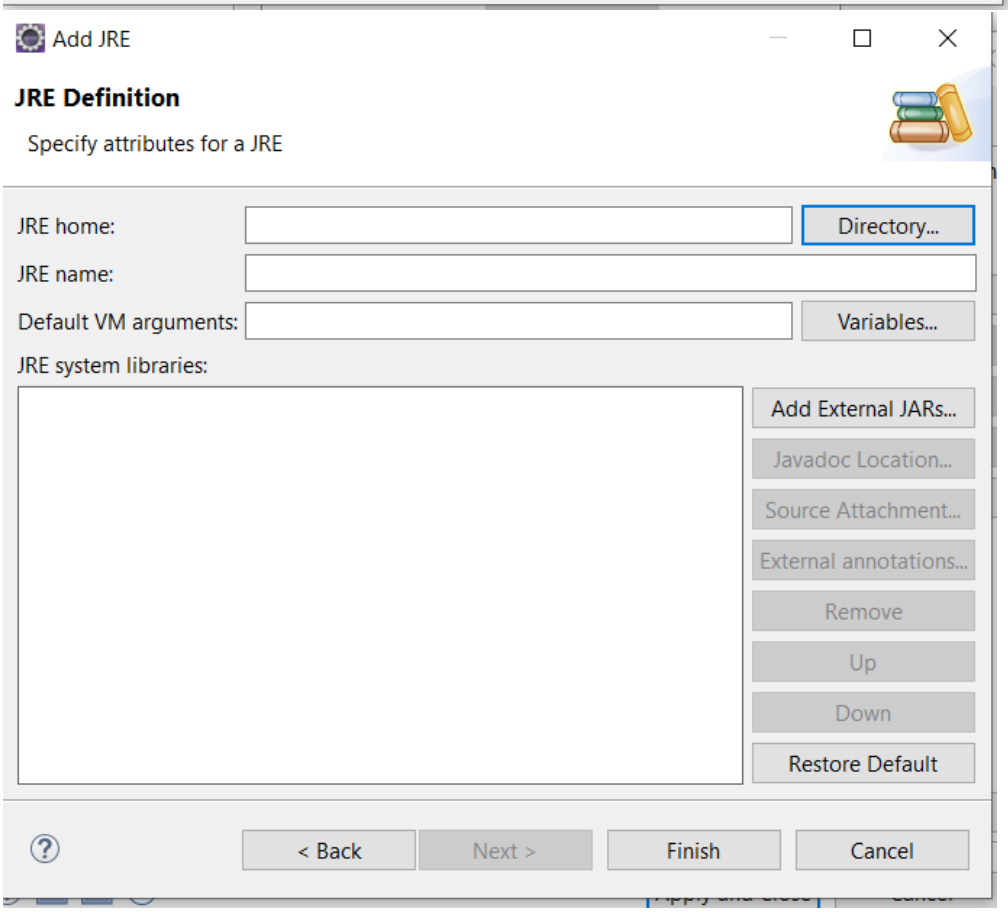
6.


Edit Library
JRE System Library
Select JRE for the project build path.
System library
Execution environment:
JavaSE-11 (jre)
Environments...
Alternate JRE:
jre
Installed JREs...
Workspace default JRE (jre)
Finish
Cancel

7.

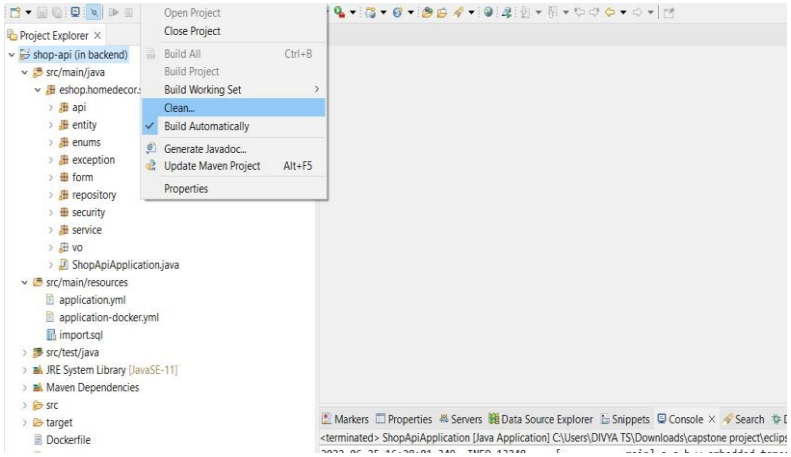

Preferences (Filtered)
type filter text
Java
Installed JREs
Installed JREs
Add, remove or edit JRE definitions. By default, the checked JRE is added to the build path of newly created Java projects.
Installed JREs:
\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.ful
Add...
Edit...
Duplicate...
Remove
Search...
Apply
Apply and Close
Cancel

8. 

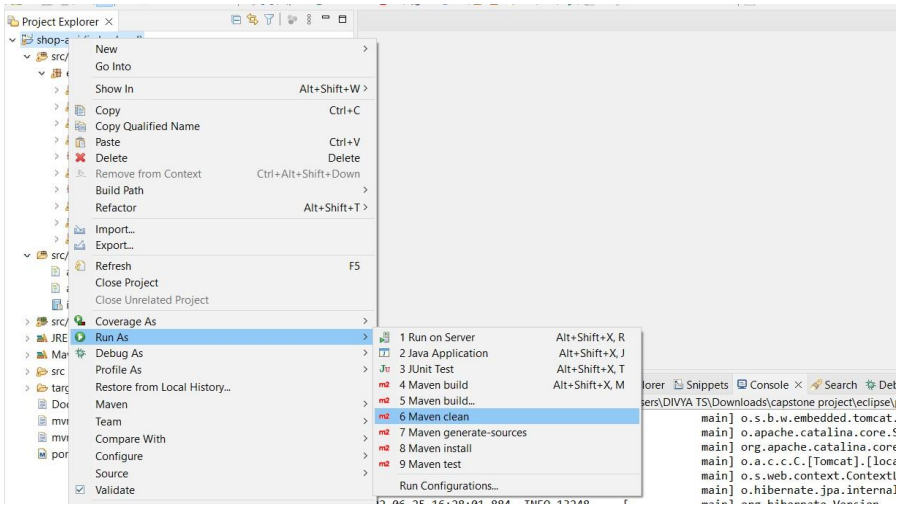
9. 

Select Java directory (eg C:\Program Files\Java\jdk-11.0.14)

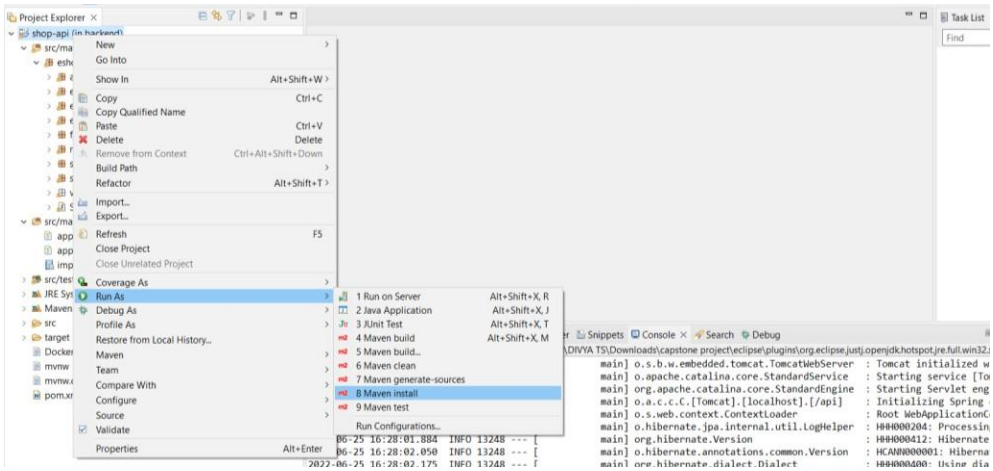
Then Do Project clean and Build.



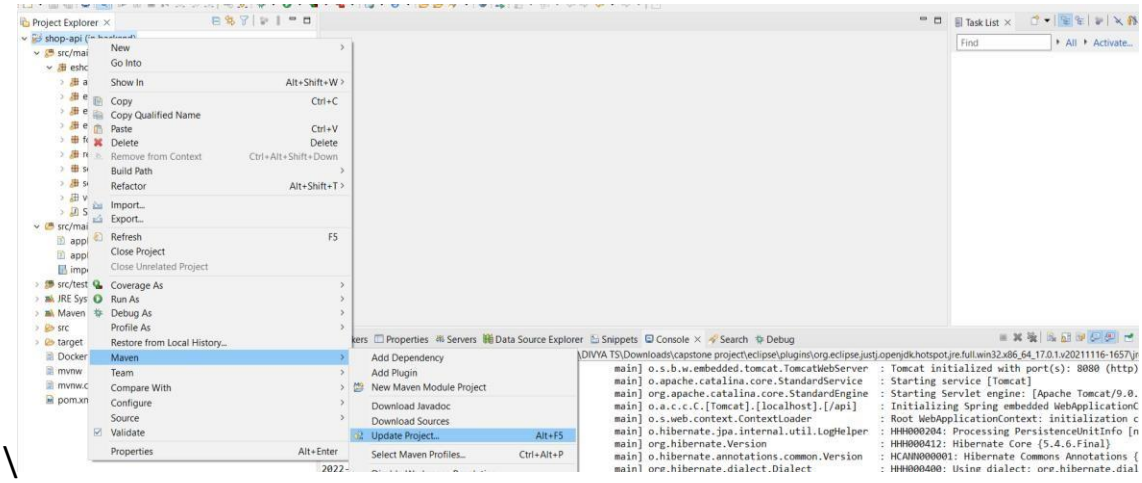
Then do maven clean



Maven install



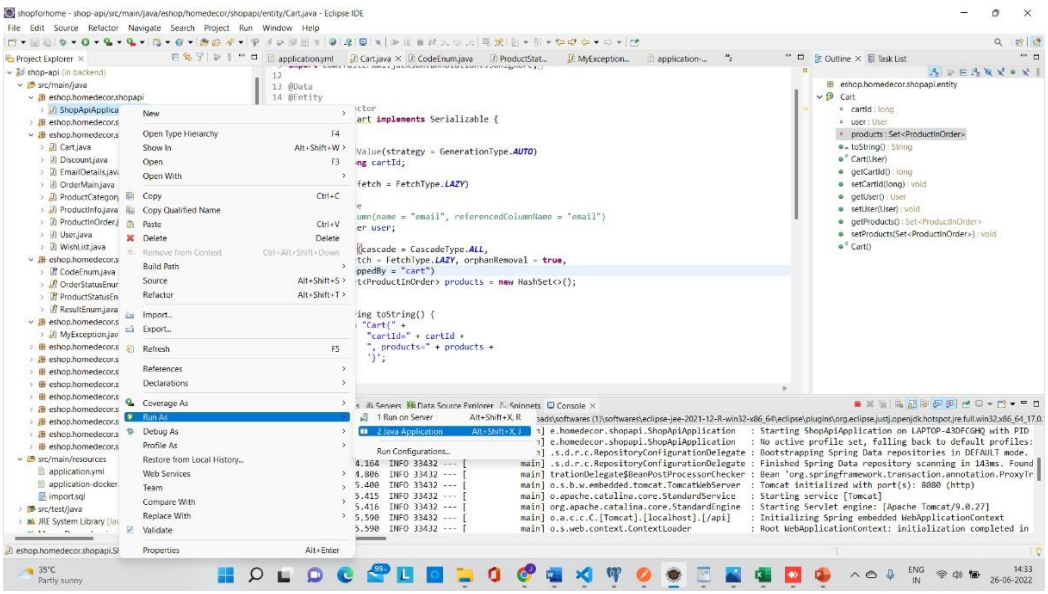
Maven Update



\

Start Server

Before starting application -> create DB and add required tables



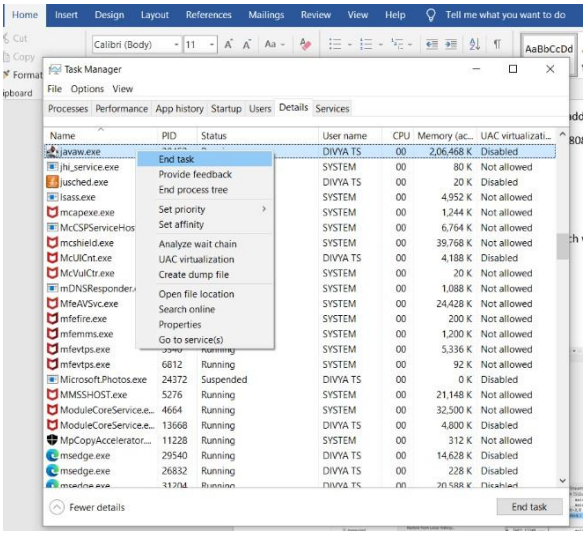
Then start the server (Note : Make sure port 8080 is not used by any other application)

3.3 Known Issues :

Port 8080 is already in use

Fix:

Open TaskManager -> details -> search with j -> javaw.exe / java.exe -> Right Click -> End Task



3.4 Front End

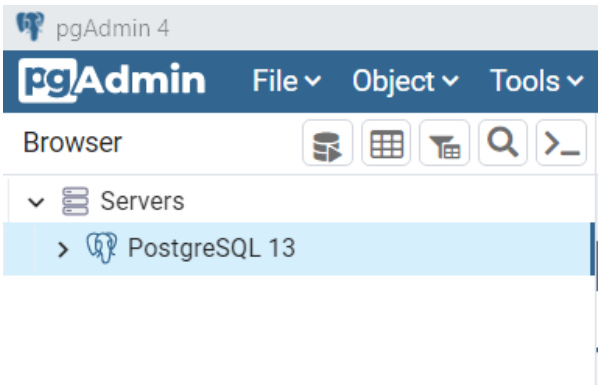
From your local FrontEnd code path -> open cmd

Eg -> {local path}\ecommerce-eshop\frontend

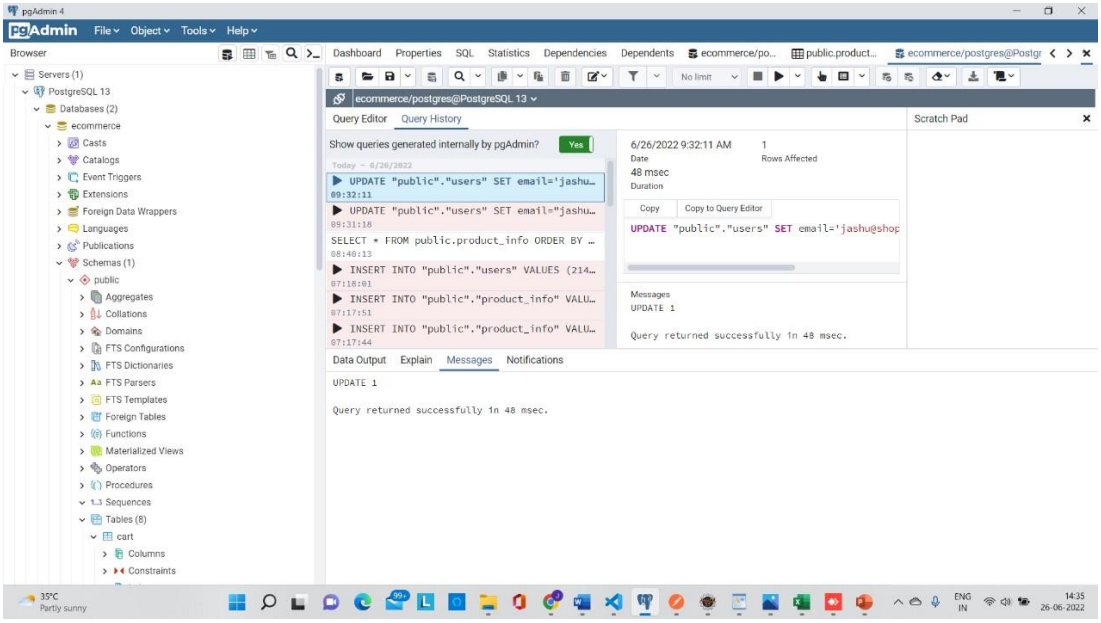
- Run this command -- **code .**
- Run **Npm install** from vs code terminal
- Run npm start
- After the successful compiling you got this link in terminal localhost/4200.
- Open this link in google chrome

3.5 DataBase

- Install postgres SQL
- After installation serach PgAdmin in your computer
- Open that PgAdmin



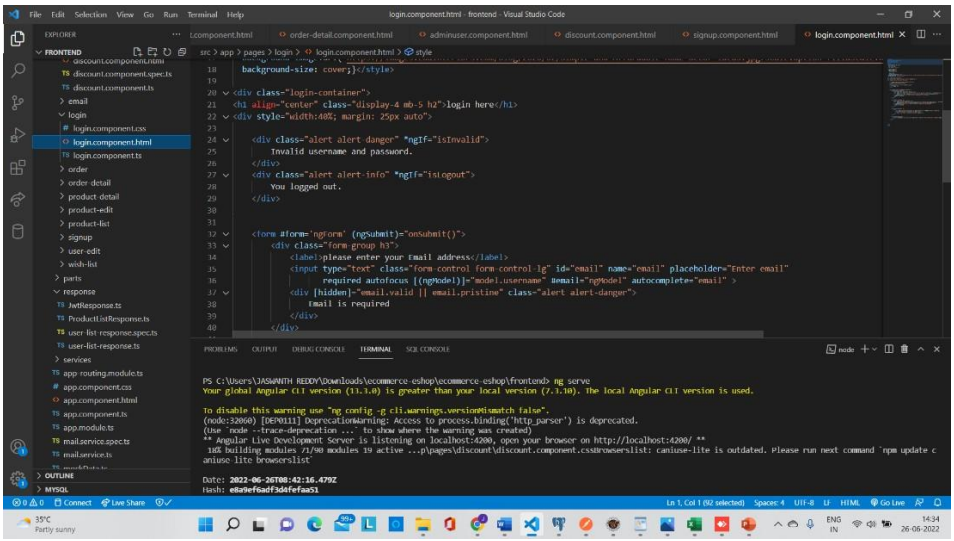
- This type of screen will open.

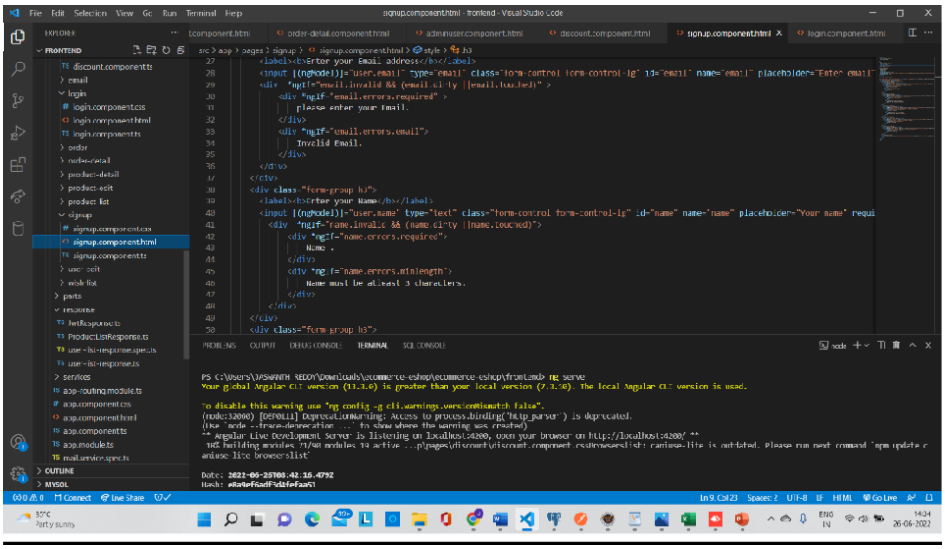


- Open this query editor and You can run the querry from sql file.
- Run the query.

CHAPTER-4

SOURCE CODE

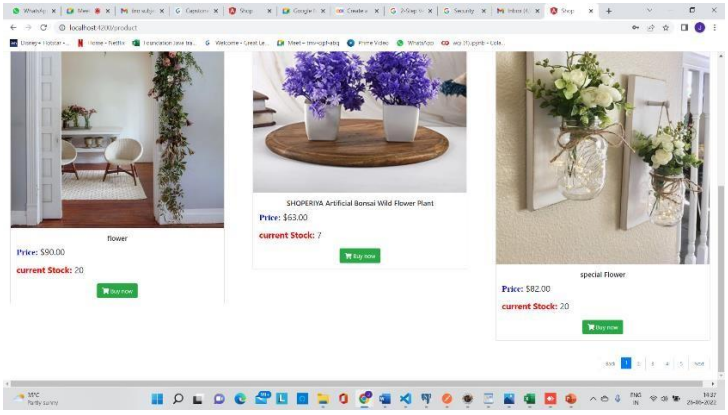


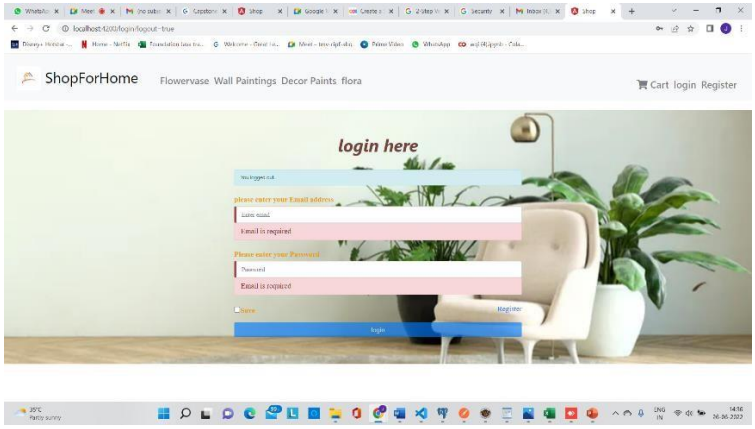


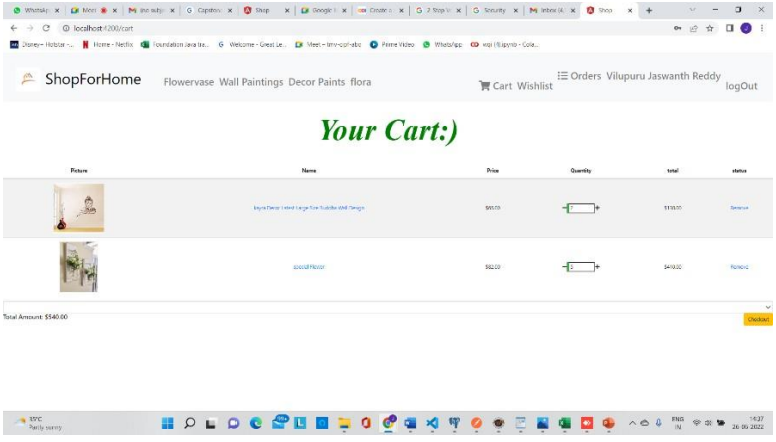
CHAPTER-5

RESULT

The output of the project is shown as in the name of output images.







CHAPTER-6

CONCLUSION

By performing or following the above-mentioned methodology we can conclude that

- We can create a web page using the Angular, Spring boot, Node-JS, Mysql, Eclipse etc..
- In this project, we have created the page of name "**SHOP FOR HOME**" in which a user can buy the house hold items like decorating things and many other.
- We are needed updated or latest version software for smooth working of the project.
- Both frontend and back end needs to run in a frequent manner for to have the page with user interface flexibility.

At the end we can say that web page is created by following the path regulations and shown the output as required.

CHAPTER-7

REFERENCES

Project references are **references to projects that contain assemblies**; you add project references by using the Projects tab of the Reference Manager dialog box. Visual Studio can find an assembly when given a path to the project.