

WIPRO CAPSTONE PROJECT SHOP FOR HOME C1 - GROUP B - G4

Submitted in the partial fulfilment of the requirements
for Wipro certification

JAVA FULL STACK DEVELOPMENT

Submitted by

Adarsh V S

Phani Kumar Talapaneni

Govardhan sai Allu

Seera Sai Siva Srinivas

Kondapu Nagendharreddy

Harshit Chaudhary

under the guidance of
Mr. Kurra Kiran Kumar



ABSTRACT

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 CONENTS

CHAPTER-1: INTRODUCTION

1.1 PURPOSE

1.2 USER REQUIREMENTS

1.3 ADMIN REQUIREMENTS

CHAPTER-2: TECHNOLOGIES REQUIRED

2.1 SUPPORTED OPERATING SYSTEM

2.2 FEATURES OF THE APPLICATION

CHAPTER-3: SOFTWARE USED

3.1 FRONTEND

3.2 BACKEND

3.3 DATABASE

CHAPTER-4: RESULTS

CHAPTER-5: CONCLUSION

CHAPTER-6: REFERENCE

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 operation 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
- **BOOTSTRAP:** It has been used for 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.

2.2 Features of the application:

1. Registration
2. Login
3. Searching
4. Filtering
5. Sorting
6. Dynamic data
7. Responsive and compatible with different devices

CHAPTER-3

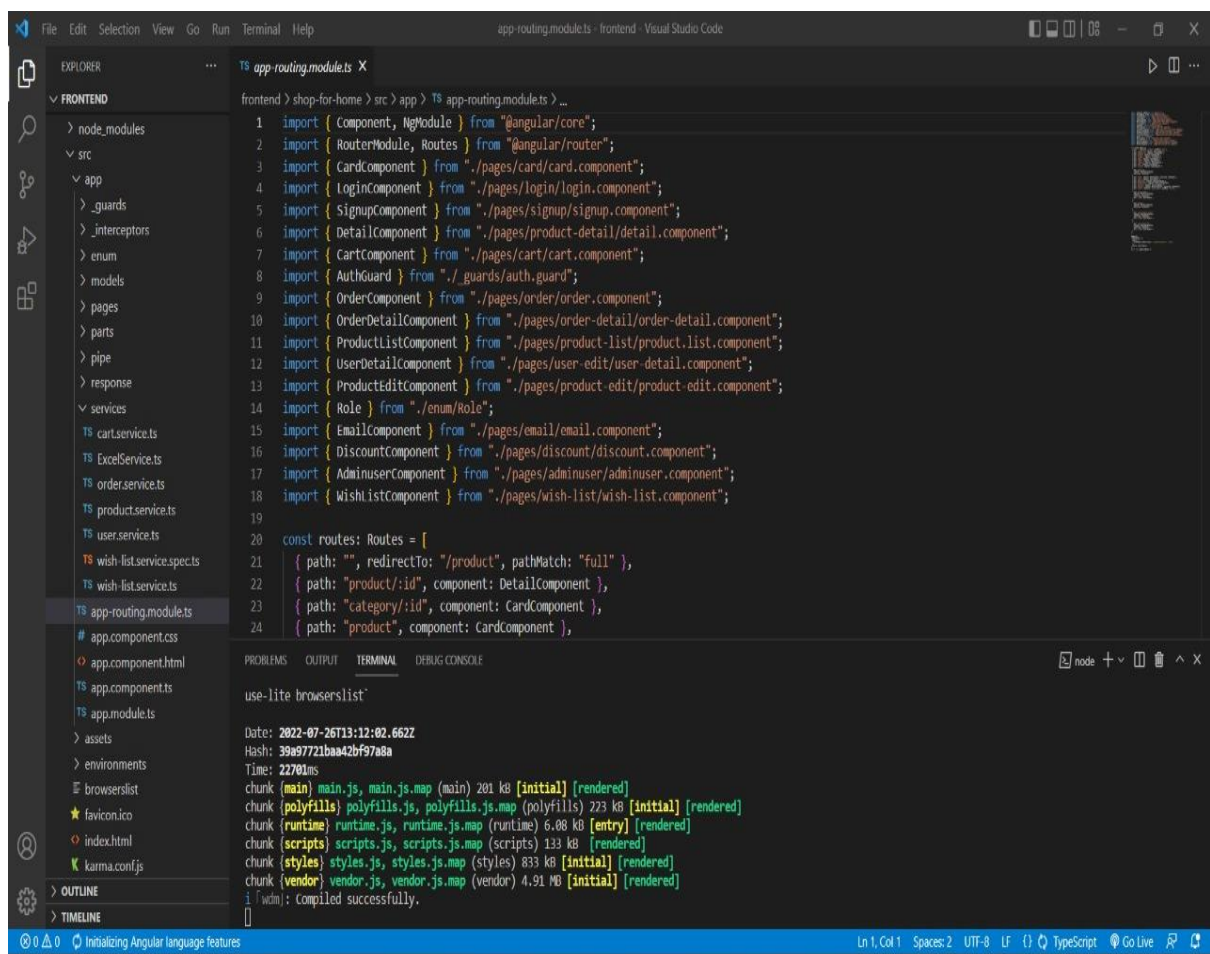
SOFTWARE USED

3.1 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



The screenshot shows the Visual Studio Code interface with the Explorer, Search, and Run and Debug views. The Explorer view shows the file structure of the frontend code, including the app-routing.module.ts file. The Search view shows the results of a search for "app-routing.module.ts". The Run and Debug view shows the terminal output, which includes the command "use-lite browserslist" and the output of the "ng serve" command, showing the application is running on localhost:4200.

```
1 import { Component, NgModule } from "@angular/core";
2 import { RouterModule, Routes } from "@angular/router";
3 import { CardComponent } from "../pages/card/card.component";
4 import { LoginComponent } from "../pages/login/login.component";
5 import { SignupComponent } from "../pages/signup/signup.component";
6 import { DetailComponent } from "../pages/product-detail/detail.component";
7 import { CartComponent } from "../pages/cart/cart.component";
8 import { AuthGuard } from "../guards/auth.guard";
9 import { OrderComponent } from "../pages/order/order.component";
10 import { OrderDetailComponent } from "../pages/order-detail/order-detail.component";
11 import { ProductListComponent } from "../pages/product-list/product-list.component";
12 import { UserDetailComponent } from "../pages/user-edit/user-detail.component";
13 import { ProductEditComponent } from "../pages/product-edit/product-edit.component";
14 import { Role } from "../enum/role";
15 import { EmailComponent } from "../pages/email/email.component";
16 import { DiscountComponent } from "../pages/discount/discount.component";
17 import { AdminuserComponent } from "../pages/adminuser/adminuser.component";
18 import { WishListComponent } from "../pages/wish-list/wish-list.component";
19
20 const routes: Routes = [
21   { path: "", redirectTo: "/product", pathMatch: "full" },
22   { path: "product/:id", component: DetailComponent },
23   { path: "category/:id", component: CardComponent },
24   { path: "product", component: CardComponent },
25 ]
```

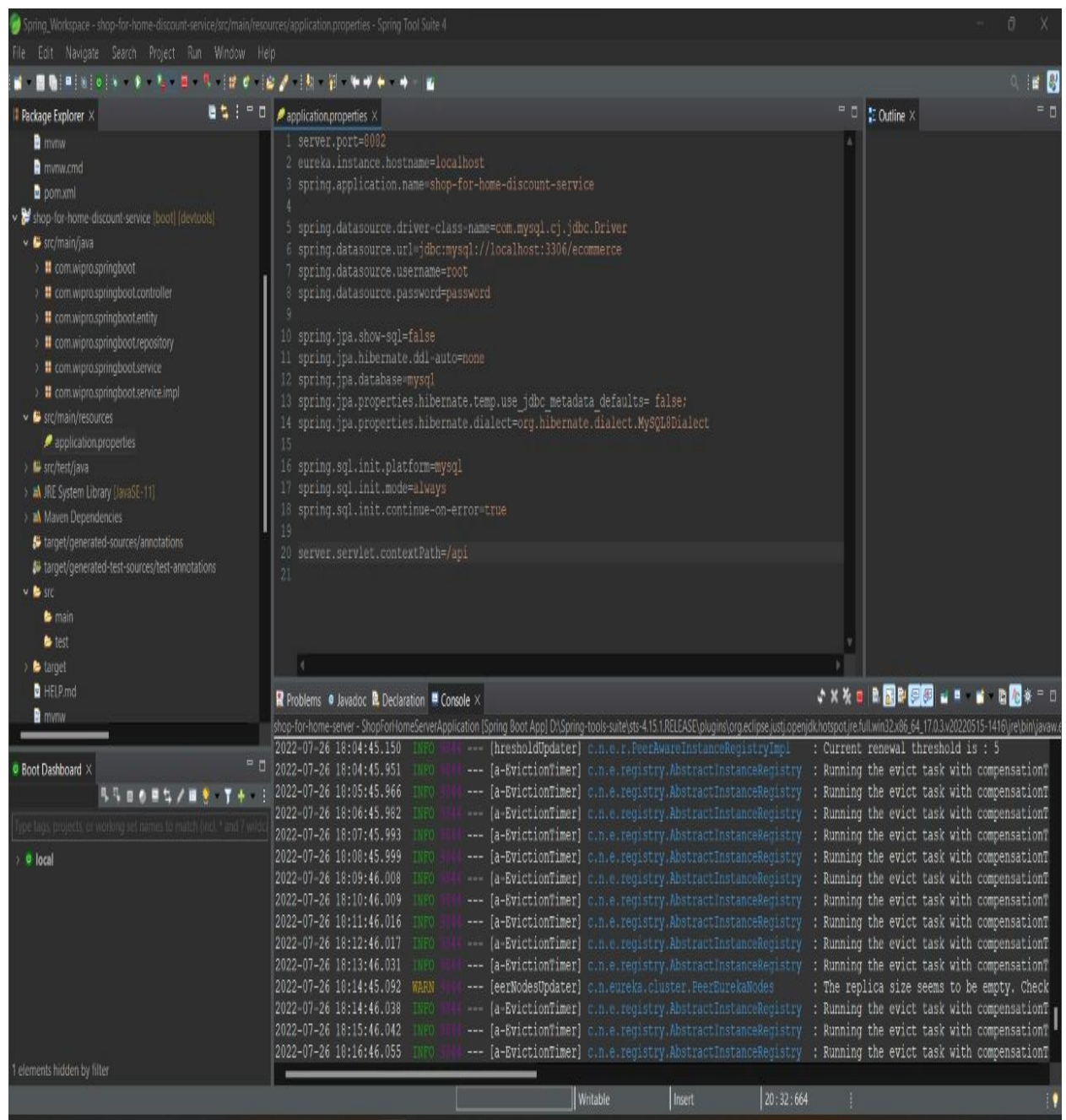
use-lite browserslist

Date: 2022-07-26T13:12:02.662Z
Hash: 39a97721baa42bf97a8a
Time: 22701ms

chunk (main) main.js, main.js.map (main) 201 kB [initial] [rendered]
chunk (polyfills) polyfills.js, polyfills.js.map (polyfills) 223 kB [initial] [rendered]
chunk (runtime) runtime.js, runtime.js.map (runtime) 6.08 kB [entry] [rendered]
chunk (scripts) scripts.js, scripts.js.map (scripts) 133 kB [rendered]
chunk (styles) styles.js, styles.js.map (styles) 833 kB [initial] [rendered]
chunk (vendor) vendor.js, vendor.js.map (vendor) 4.91 MB [initial] [rendered]
i (win): Compiled successfully.

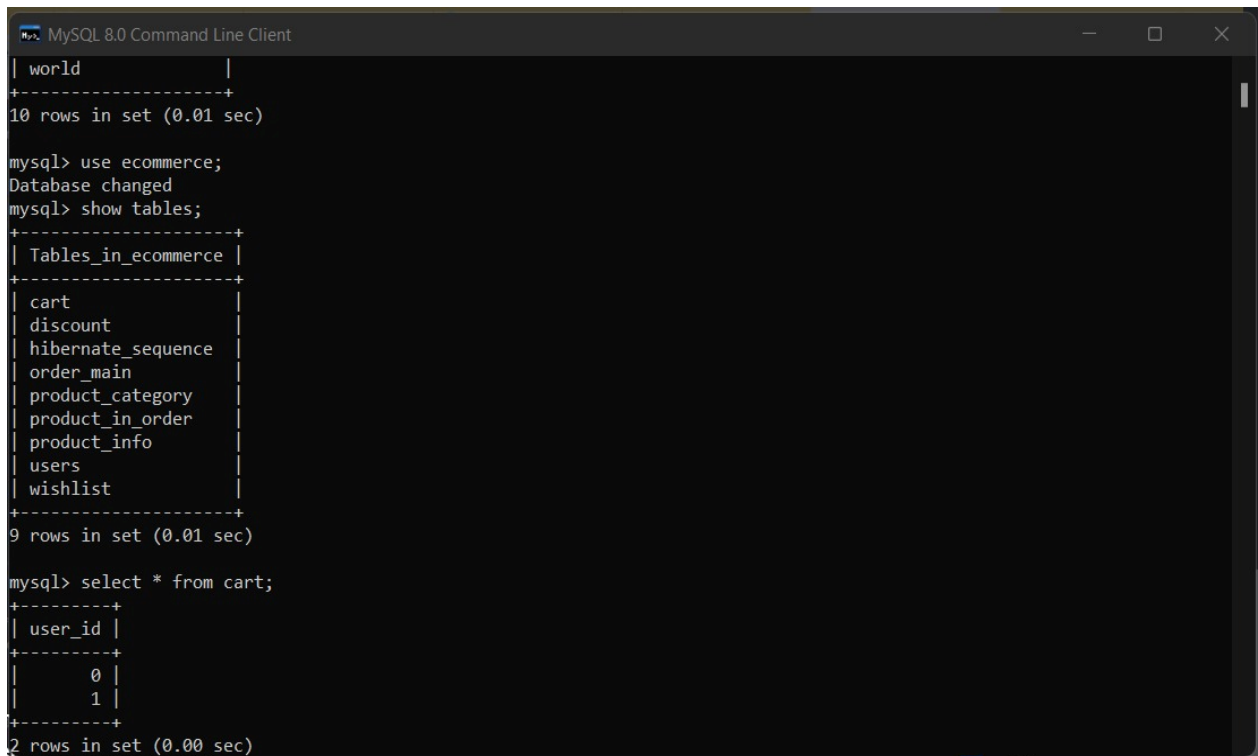
3.2 Back End

Spring Tool Suite (STS) is a java IDE tailored for developing Spring-based enterprise applications. It is easier, faster, and more convenient. And most importantly it is based on Eclipse IDE. STS is free, open-source, and powered by VMware.



3.3 Database

- Install MySQL
- After installation search MySQL command line in your computer
- Open that command line

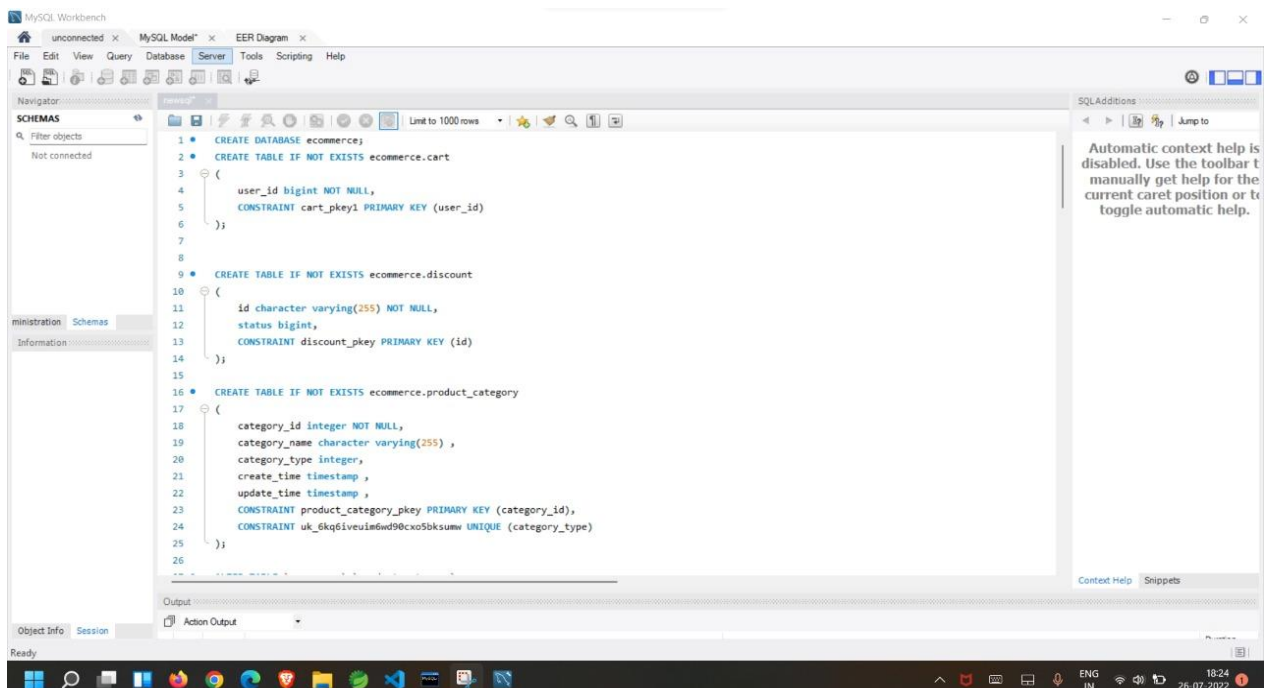


```
MySQL 8.0 Command Line Client
| world |
+-----+
10 rows in set (0.01 sec)

mysql> use ecommerce;
Database changed
mysql> show tables;
+-----+
| Tables_in_ecommerce |
+-----+
| cart                  |
| discount              |
| hibernate_sequence    |
| order_main            |
| product_category      |
| product_in_order      |
| product_info          |
| users                 |
| wishlist              |
+-----+
9 rows in set (0.01 sec)

mysql> select * from cart;
+-----+
| user_id |
+-----+
| 0       |
| 1       |
+-----+
2 rows in set (0.00 sec)
```

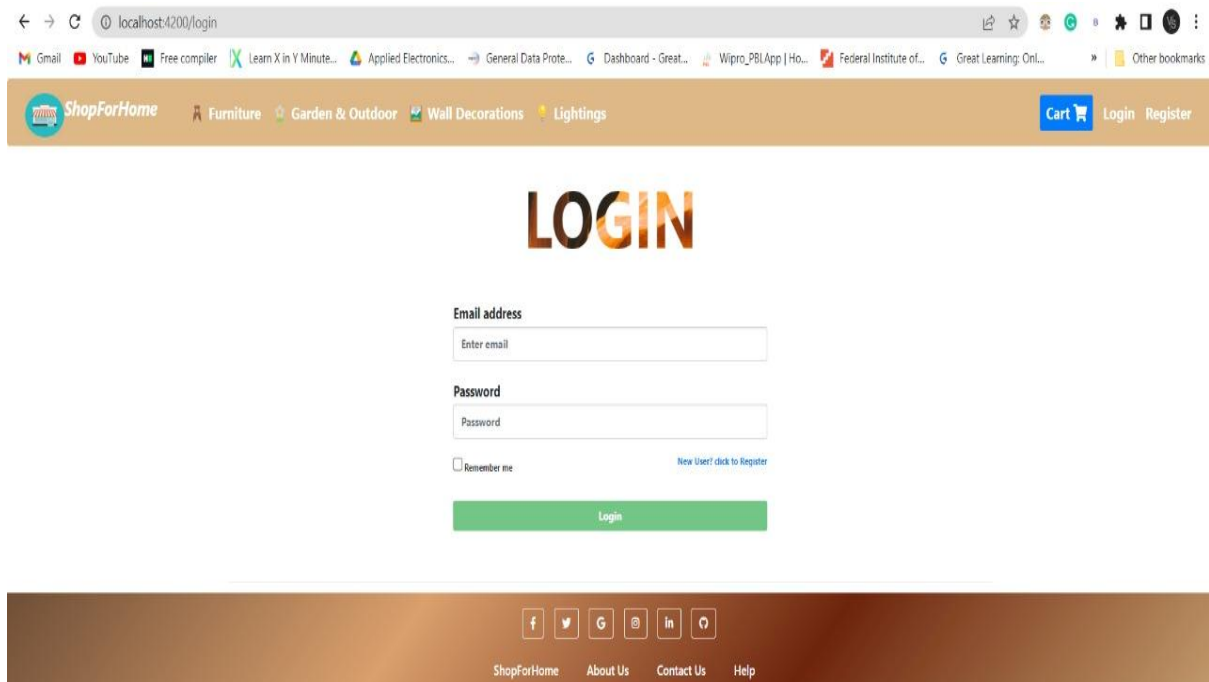
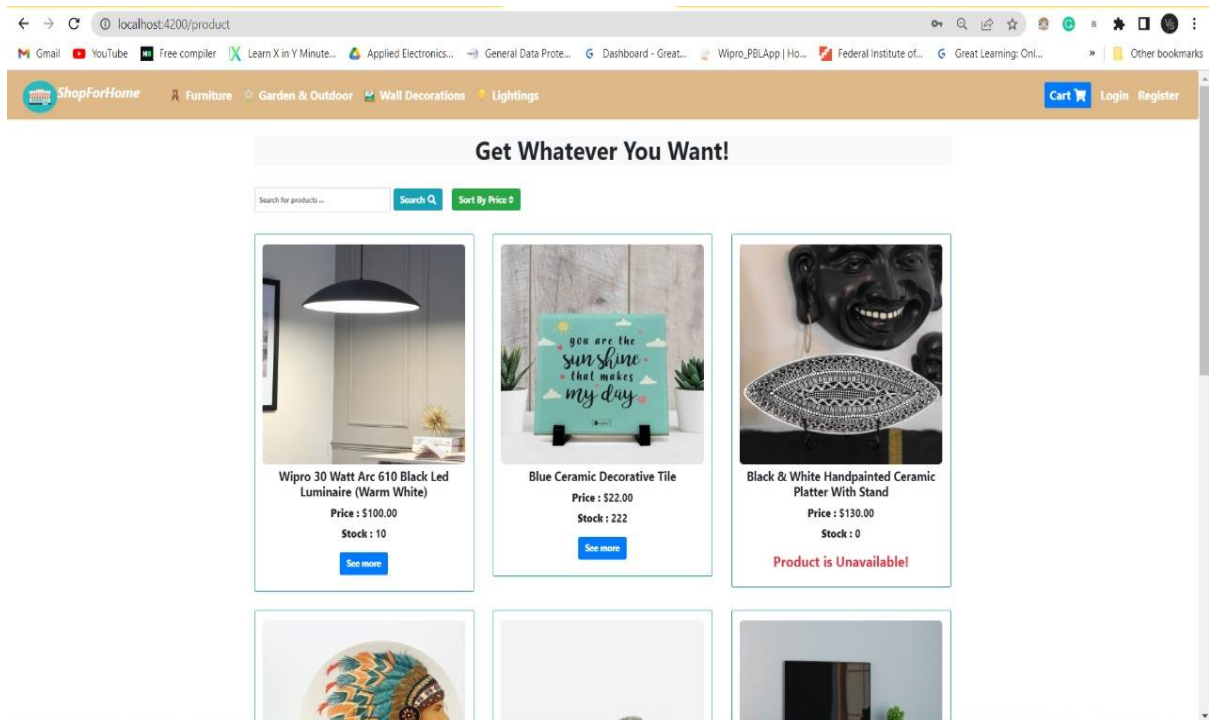
➤ MySQL Workbench



CHAPTER-4

RESULTS

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



Products

Add Product +





Choose File No file chosen

Upload Products

Download Report

Search for products ...

Search Q

Photo	Code	Name	Type	Description	Price	Stock	Status	Action	Mail
	B0011	Study Table	Books	Computer Table with Drawers	\$30.00	95	Available	Edit Remove	Email
	B0012	Dining Table	Books	Wooden 4 Seater Dining Table Set	\$20.00	195	Available	Edit Remove	Email
	B0013	Fabric Chair	Books	Colourful Fabric Chair	\$10.00	9	Available	Edit Remove	Email
	B0014	Console Table	Books	Rectanqular Wooden Console Table	\$30.00	199	Available	Edit	Email

Orders

Order #	Customer Name	Customer Email	Customer Phone	Shipping Address	Total	Order Date	Status	Action
3	Adam	abc@gmail.com	0987654321	weqweggd	\$30.00	Jul 25, 2022	Finished	Show




Previous 1 Next

[Gmail](#)
[YouTube](#)
[free compiler](#)
[Learn X in Y Minute...](#)
[Applied Electronics...](#)
[General Data Prote...](#)
[Dashboard - Great...](#)
[Wipro_PBLApp | Ho...](#)
[Federal Institute of...](#)
[Great Learning: Onl...](#)
[Other bookmarks](#)

[ShopForHome](#)
[Furniture](#)
[Garden & Outdoor](#)
[Wall Decorations](#)
[Lightings](#)

[Cart](#)
[Login](#)
[Register](#)

My Cart

Photo	Name	Price	Quantity	Subtotal	Action
	Wipro 30 Watt Arc 610 Black Led Luminaire (Warm White)	\$100.00	<input type="text" value="1"/>	\$100.00	Remove
	Console Table	\$30.00	<input type="text" value="3"/>	\$90.00	Remove
	Fabric Chair	\$10.00	<input type="text" value="2"/>	\$20.00	Remove

Total: \$210.00

[Checkout](#)

[f](#)
[t](#)
[G](#)
[@](#)
[in](#)
[v](#)



[ShopForHome](#)
[About Us](#)
[Contact Us](#)
[Help](#)

[Gmail](#)
[YouTube](#)
[free compiler](#)
[Learn X in Y Minute...](#)
[Applied Electronics...](#)
[General Data Prote...](#)
[Dashboard - Great...](#)
[Wipro_PBLApp | Ho...](#)
[Federal Institute of...](#)
[Great Learning: Onl...](#)
[Other bookmarks](#)

[ShopForHome](#)
[Furniture](#)
[Garden & Outdoor](#)
[Wall Decorations](#)
[Lightings](#)

[Cart](#)
[Wishlist](#)
[Orders](#)
[user1](#)
[Logout](#)

My WishList

Photo	Name	Price	Stock	Action
	Center Table	\$30.00	199	Remove
	Console Table	\$30.00	199	Remove

[f](#)
[t](#)
[G](#)
[@](#)
[in](#)
[v](#)

[ShopForHome](#)
[About Us](#)
[Contact Us](#)
[Help](#)

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, 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 need updated or latest version software for smooth working of the project.
- Both frontend and backend need to run in a frequent manner to 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.

