



Capstone Project Report On - **SHOP FOR HOME**

Team Members:

1. Priyanka Dandupati
2. Beera Hema Sundar Chinna Rao
3. Nandini Pothuraju
4. Jayasree Nallamothe

Mentor: Parth Shukla

Stream: JAVA-GCP

Partner: Great Learning

PROBLEM STATEMENT:

Shop For Home is a popular Store in the market for shopping the home décor stuff. Due to Covid-19, all offline shopping stopped. So, the store wants to move to the cloud platforms and wants its own web application.

COMPONENTS OF THE PROJECT:

Any full-stack application can be divided into three parts that are mentioned below

- The Frontend
- The Backend
- The Database

Core concepts used in the project:

1. Front End:

- HTML.
- CSS3
- Bootstrap4.
- Type Script.
- Angular,
- Angular Material for View.

2. Git:

To connect and push files from the local system to GitHub.

3. GitHub:

To store the application code and track its versions

4. Back End:

Eclipse IDE, Java Programming, Searching and Sorting Spring Boot Dev Tools, Spring Web and Spring Data JPA.

Operating System:

- Windows 7/8/10/11
- Linux distros
- MacOS X or later.

Processor:

- Intel or AMD dual core x 86 processor.

RAM:

- 8 GB or above.

Hard disk:

- 1000 MB of free disk space or more.

Features of the application:

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

Recommended technologies:

1. Database management: SQL.
2. Backend logic: Spring Boot, Java programming.
3. Frontend development: Angular, Bootstrap, HTML/CSS, and Java script
4. DevOps and production technologies: Git, GitHub.

STUDY OF THE SYSTEM:

Modules:

The system after a careful analysis has been identified to be presented with the following modules and roles.

The modules involved are:

- Admin
- Users

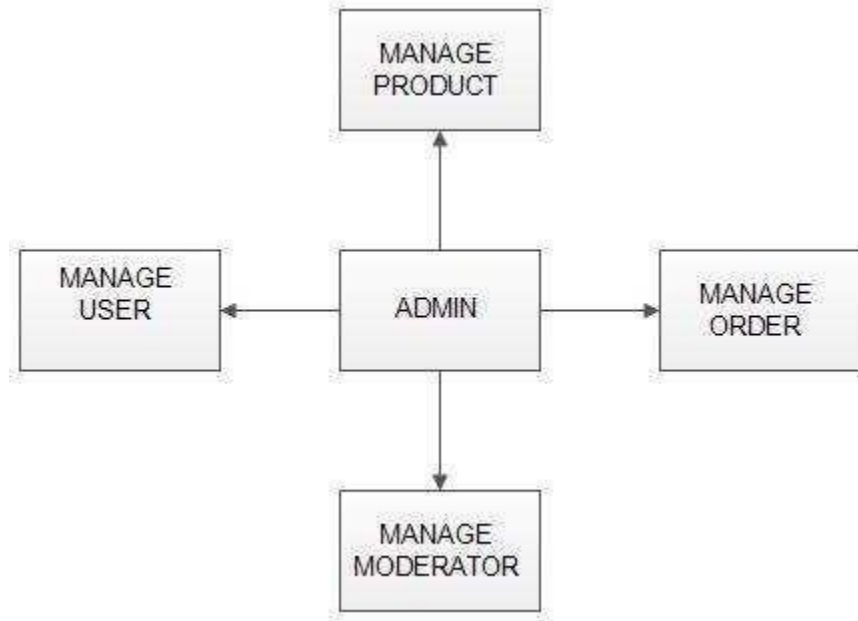
Admin Portal:

The administrator is the super user of this application. Only the admin has access to this admin page. Admin may be the owner of the shop. The administrator has all the information about all the users and about all products.

The admin user should be able to: Add or remove different products to or from the application to build a rich product line Edit product item details like name, price, discount, and offers to keep it aligned to the current prices Enable or disable the product items.

This module is divided into different sub-modules.

1. Manage Moderators
 2. Manage Products
 3. Manage Users
 4. Manage Orders
-



Admin module

Manage Moderator's:

- **Add Moderator:**

Only the admin is having the privilege to add a moderator. A moderator can be considered as staff who manages the orders or owner of a group of products.

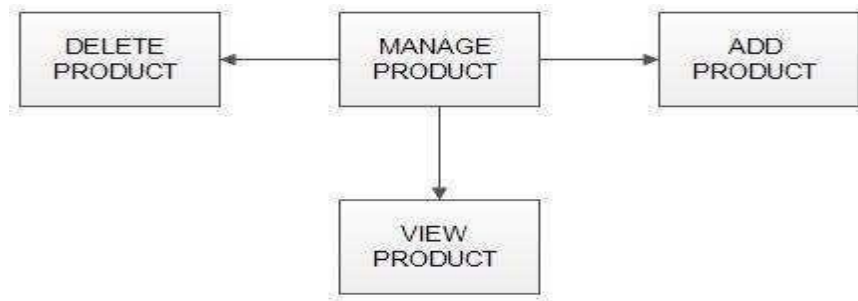
- **Remove Moderator:**

Admin has the privilege to delete a moderator who was added.

- **Search moderator:**

All existing moderators can be viewed by the administrator as a list. If there are a number of moderators and the admin needs to find one of them, the admin can search for a moderator by name.

Manage Products:



Manage Products

- **Add Products:**

The shopping cart project contains different kinds of products. The products can be classified into different categories by name. Admin can add new products into the existing system with all its details including an image.

- **Delete Products:**

Administrators can delete the products based on the stock of that particular product.

- **Search products:**

Admin will have a list view of all the existing products. He can also search for a particular product by name.

Manage Users:

- **View Users:**

The admin will have a list view of all the users registered in the system. Admin can view all the details of each user in the list except the password.

- **Add Users:**

Admin has privileges to add a user directly by providing the details.

Manage Orders:



Manage Orders

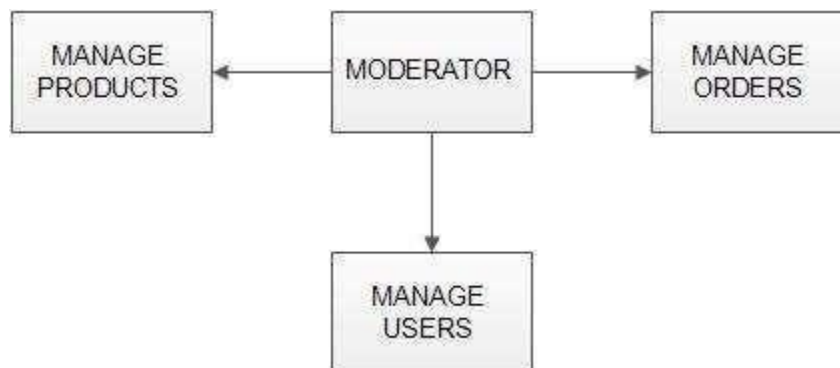
- **View Order:**

Administrators can view the Orders which are generated by the users. He can verify the details of the purchase.

- **Delete Order:**

Admin can delete orders from the orders list when the product is taken for delivery.

Moderators:



Moderator Module

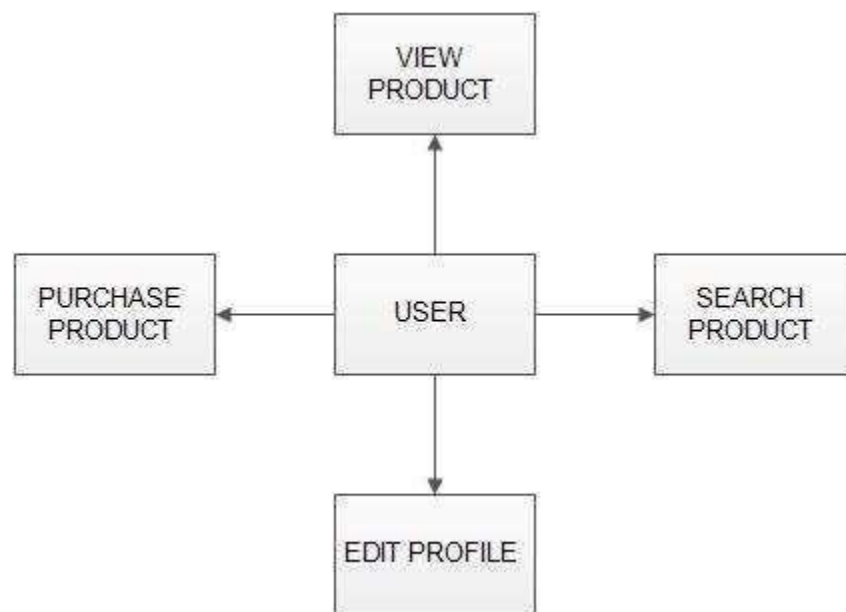
A moderator is considered a staff who can manage orders for the time being. As a future update moderator may give the facility to add and manage his own products. Moderators can reduce the workload of admin. Now moderator has all the privileges an admin has

except managing other moderators. He can add products and users. He can also check the orders and edit his profile.

- Manage products
- Manage users
- Manage orders

Users Portal:

It deals with user activities. The end-user should be able to: Sign-in to the application to maintain a record of activities and Search for product items based on the search keyword. Add all the selected product items to a cart and customize the purchase at the end Perform a seamless payment process Get an order summary details page once the payment is complete



User Module

Registration:

A new user will have to register in the system by providing essential details in order to view the products in the system. The admin must accept a new user by unblocking him.

- **Login:**

A user must login with his user name and password to the system after registration.

- **View Products:**

Users can view the list of products based on their names after successful login. A detailed description of a particular product with product name, product details, product image, and price can be viewed by users.

- **Search Product:**

Users can search for a particular product in the list by name.

- **Add to cart:**

The user can add the desired product to his cart by clicking add to cart option on the product. He can view his cart by clicking on the cart button. All products added by the cart can be viewed in the cart. Users can remove an item from the cart by clicking remove.

- **Submit Cart:**

After confirming the items in the cart the user can submit the cart by providing a delivery address. On successful submitting the cart will become empty.

- **History:**

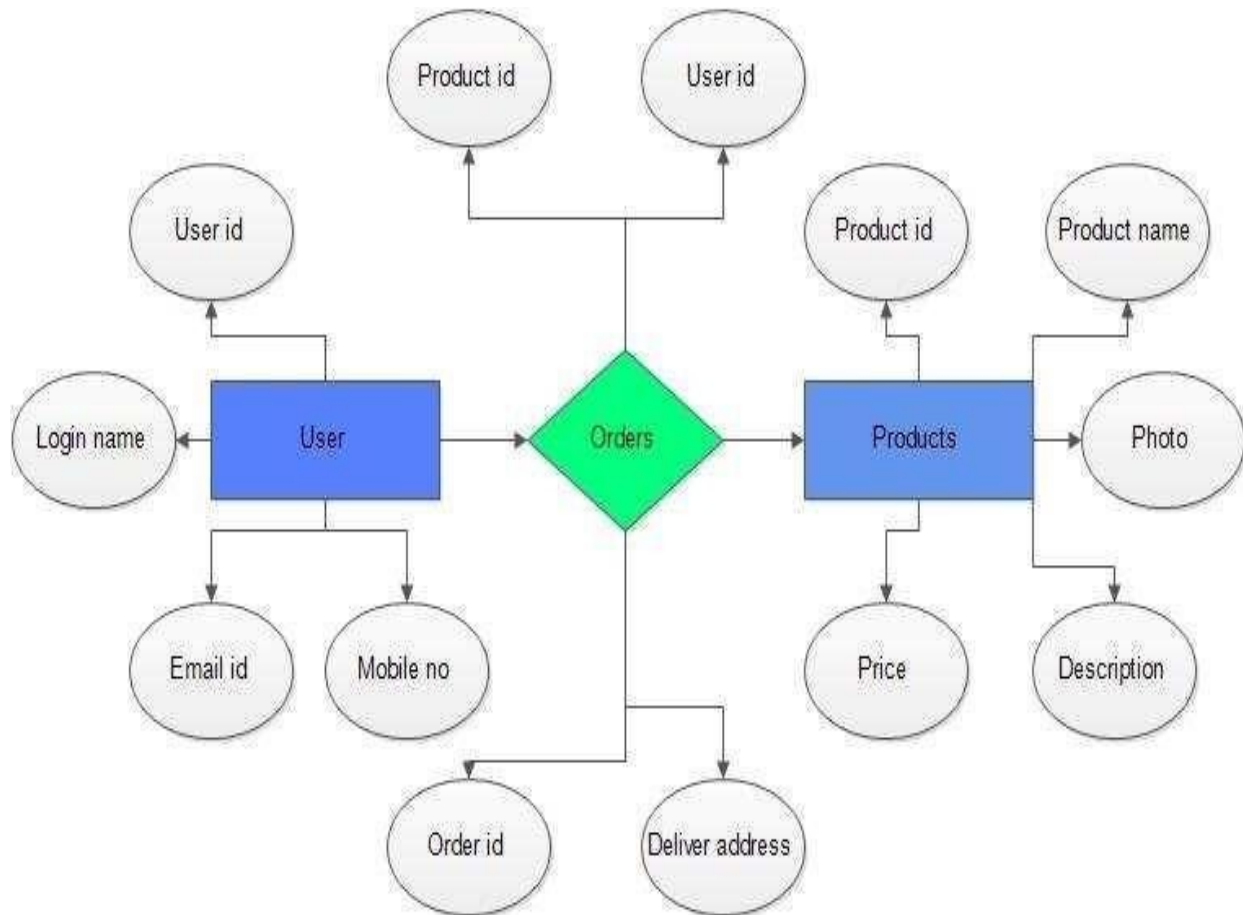
In the history, the user will have a view of pending orders.

- **Edit Profile:**

The user can view and edit the profile.

DATA FLOW DIAGRAM:

A Data Flow Diagram (DFD) is a structured analysis and design tool that can be used for flowcharting. A DFD is a network that describes the flow of data and the processes that change or transform the data throughout a system. This network is constructed by using a set of symbols that do not imply any physical implementation. It has the purpose of the clarifying system requirements and identifying major transformations. So it is the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. DFD can be considered to be an abstraction of the logic of an information-oriented or a process-oriented system flow-chart. For these reasons, DFD's are often referred to as logical data flow diagrams.



Project Users and Admin Stories:

The project is planned to be completed in 2 sprints. Tasks assumed to be completed in the sprint are: Creating the flow of the application and Initializing the Git repository to track changes as development progresses. Writing the Java program to fulfill the requirements of the project.

User Stories:

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 Wish-list 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 coupons.

Admin Stories:

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 a bulk upload option to upload a CSV for product 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 the discount coupons for the specific set of user.

Sprint-I Objectives:

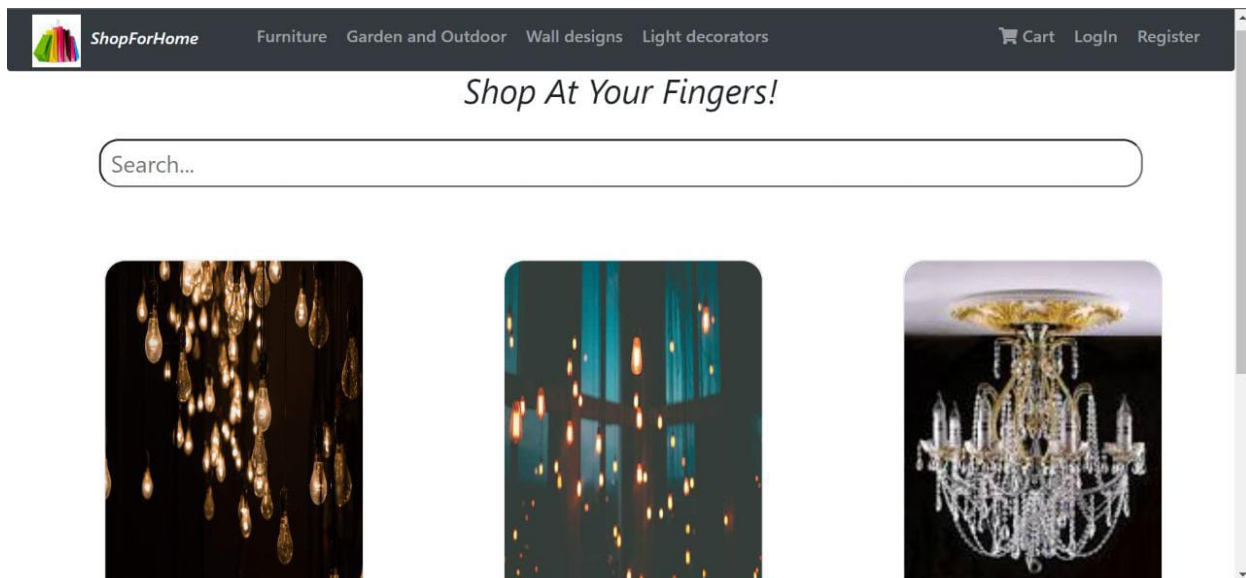
1. Create a Git repository
 2. Create database schema (all tables along with their relationships)
 3. Create entities in Spring
 4. Create Micro service based structure.
 5. CRUD on Users and Products
 6. Create an e-Commerce Template in Angular (Static only) to hold images and product lists.
 7. Develop Search Functionality in Angular
 8. Bulk upload implementation.
-

Sprint-II Objectives:

1. Cart option
2. Stock and Sales reports
3. Create Data Transfer objects
4. Create repository
5. Create Service layer logic
6. Create a Controller to direct the rest API
7. Create Discount Micro Service

Screenshots:


Home Page:



← → ↺

localhost 4200/product

🔍 📄 ☆ 🏠 📱




bulbs

Description: bulbs

Price: \$10.00

Stock: 6

Get It!




lightings

Description: lights

Price: \$20.00

Stock: 195

Get It!



Chandelier

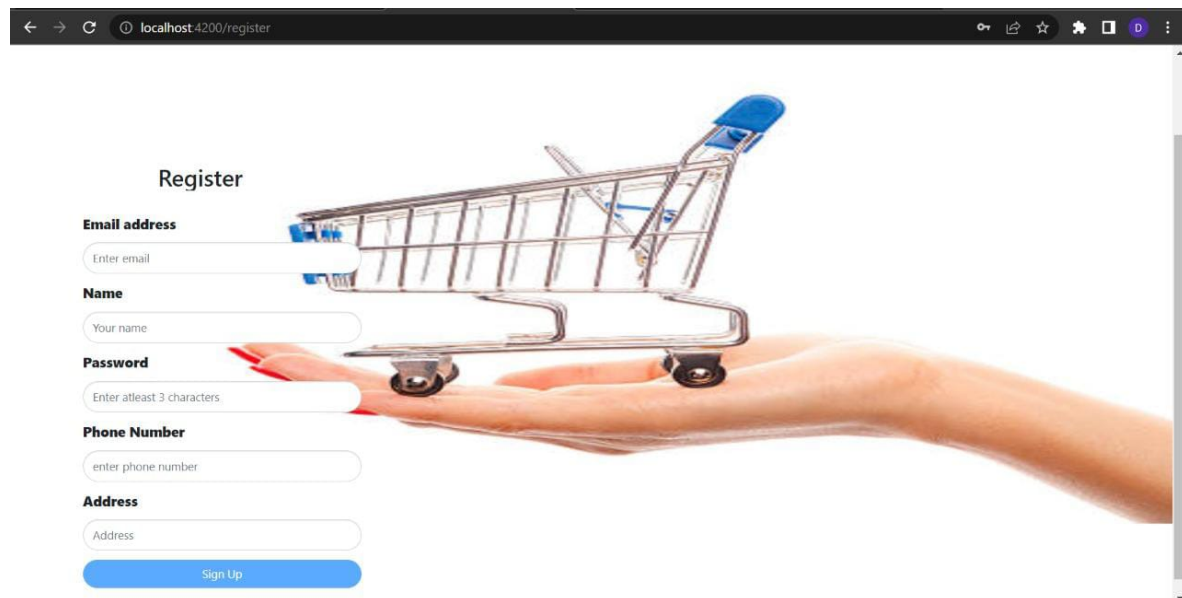
Description: chandelier

Price: \$1,000.00

Stock: 10

Get It!

Sign Up Page:



Register

Email address


Name

Password

Phone Number

Address

Sign In Page:



Login

Email address

Password


☐ Remember Me

← → ↻


localhost:4200/category/0

🔍 📁 ⭐ ⚙️ 📺 📄


Furnitures



Study Table
Description: Computer Table with Drawers
Price: \$30.00
Stock: 95



Dining Table
Description: Wooden 4 Seater Dining Table Set
Price: \$20.00
Stock: 195






Fabric Chair
Description: Colourful Fabric Chair
Price: \$10.00
Stock: 200

← → ↻

localhost:4200/category/1

🔍 📁 ⭐ ⚙️ 📺 📄

Garden & Outdoor




← → ↻ localhost:4200/category/2


🔍 📄 ☆ ⚙️ 🖨️ 🌐

Wall Decorations


Search...



Metal Leaf Art
Description: Iron Titan Leaf Metal Wall Art
Price: \$10.00



Flower Wall Art
Description: Iron Double Ring In Flower Wall Art
Price: \$13.00




Rose Flower Art
Description: Iron Red And White Flower Metal Wall Art
Price: \$22.00

← → ↻ localhost:4200/category/3


🔍 📄 ☆ ⚙️ 🖨️ 🌐

Lightings


Search...



bulbs




lightings






Chandelier

Orders Page:

← → ↺ localhost:4200/order

ShopForHome

Furniture Garden and Outdoor Wall designs Light decorators

 Cart  Wishlist  Orders kusu Sign Out

Orders

Order #	Customer Name	Customer Email	Customer phone	Shipping Address	Total	Order Data	Status	Action
2147483651	kusuma	saipriya122301@gmail.com	9876542350	main road,Tekkali	\$22.00	Aug 14, 2022	Canceled	Show
2147483650	kusuma	saipriya122301@gmail.com	9876542350	main road,Tekkali	\$40.00	Aug 14, 2022	Canceled	Show


Previous

1




Next

My Cart Page:


← → ↺ localhost:4200/cart

ShopForHome

Furniture Garden and Outdoor Wall designs Light decorators

 Cart  Wishlist  Orders kusu Sign Out

My Cart

Photo	Name	Price	Quantity	Subtotal	Action
	lightings	\$20.00	<div><div>-</div><div>3</div><div>+</div></div>	\$100.00	Remove

Select Coupon

COUPON APPLIED :

Total: \$100.00

Checkout

My Wishlist Page:

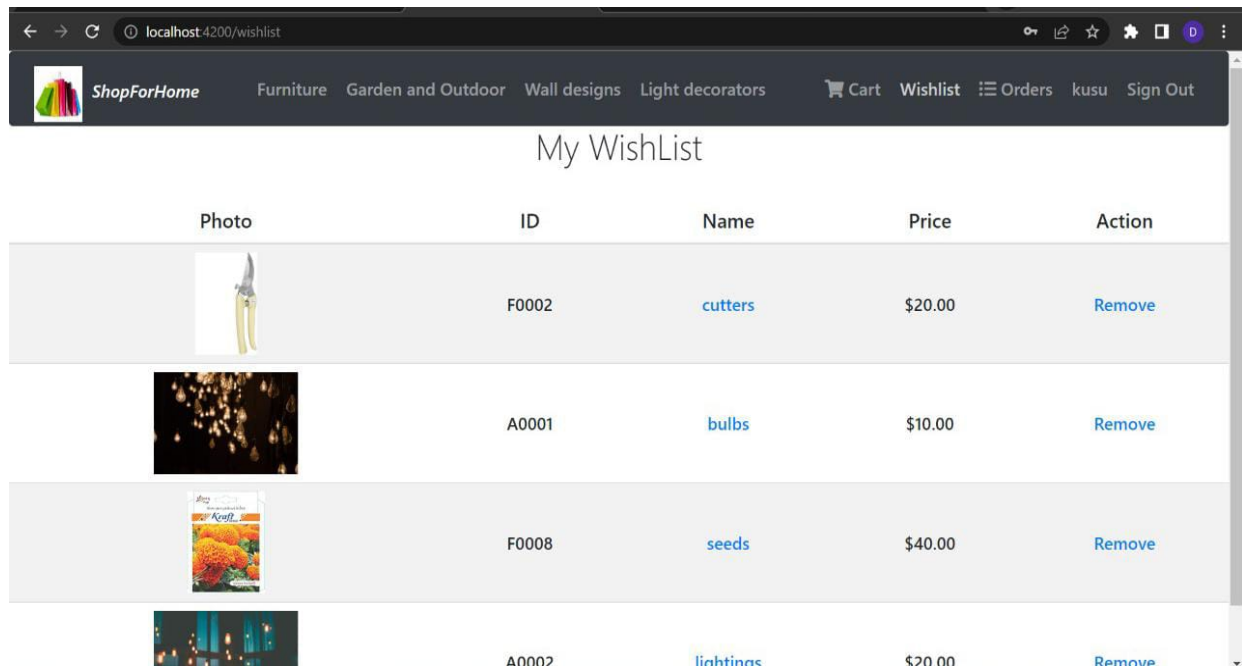






Photo	ID	Name	Price	Action
	F0002	cutters	\$20.00	Remove
	A0001	bulbs	\$10.00	Remove
	F0008	seeds	\$40.00	Remove
	A0002	lightings	\$20.00	Remove

Admin Sign In Page:



ShopForHome Furniture Garden and Outdoor Wall designs Light decorators Cart Login Register

Login

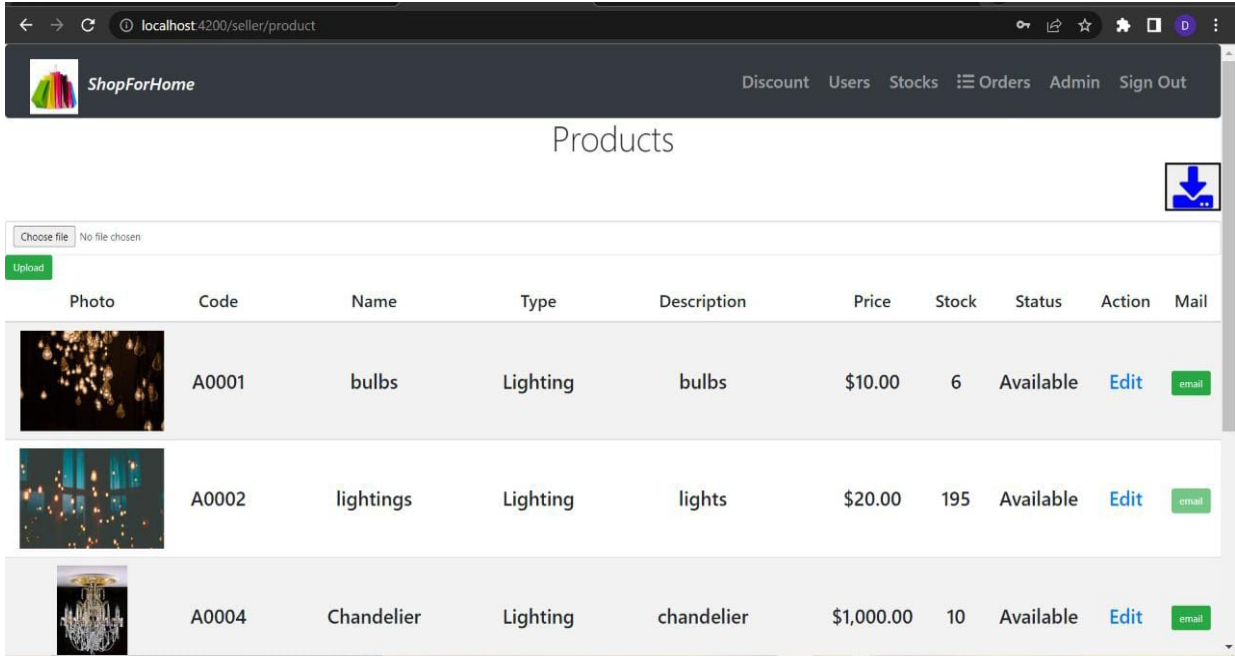
Email address
admin@eshop.com

Password




☐ Remember Me

[Sign In](#)

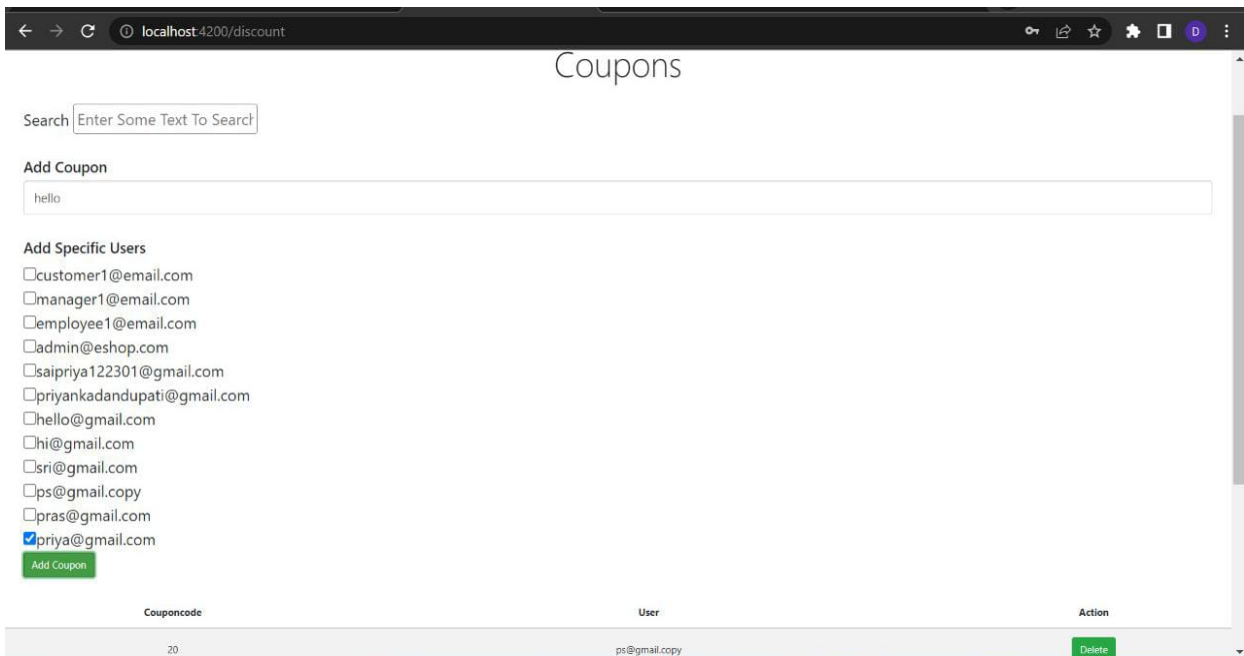
Products Page from Admin login:



The screenshot shows the 'Products' page of the ShopForHome admin interface. The browser address bar indicates the URL is localhost:4200/seller/product. The top navigation bar includes links for Discount, Users, Stocks, Orders, Admin, and Sign Out. The main heading is 'Products', and there is a download icon in the top right corner. Below the heading is a file upload section with a 'Choose file' button and an 'Upload' button. The main content is a table with columns: Photo, Code, Name, Type, Description, Price, Stock, Status, Action, and Mail. The table contains three rows of product data.

Photo	Code	Name	Type	Description	Price	Stock	Status	Action	Mail
	A0001	bulbs	Lighting	bulbs	\$10.00	6	Available	Edit	email
	A0002	lightings	Lighting	lights	\$20.00	195	Available	Edit	email
	A0004	Chandelier	Lighting	chandelier	\$1,000.00	10	Available	Edit	email

Coupons available:



The screenshot shows the 'Coupons' page of the ShopForHome admin interface. The browser address bar indicates the URL is localhost:4200/discount. The main heading is 'Coupons'. Below the heading is a search bar with the placeholder text 'Enter Some Text To Search'. There is an 'Add Coupon' section with a text input field containing 'hello'. Below this is an 'Add Specific Users' section with a list of email addresses and checkboxes. The 'priya@gmail.com' checkbox is checked. At the bottom of this section is an 'Add Coupon' button. Below the coupon form is a table with columns: Couponcode, User, and Action. The table contains one row of data.

Search

Add Coupon

Add Specific Users

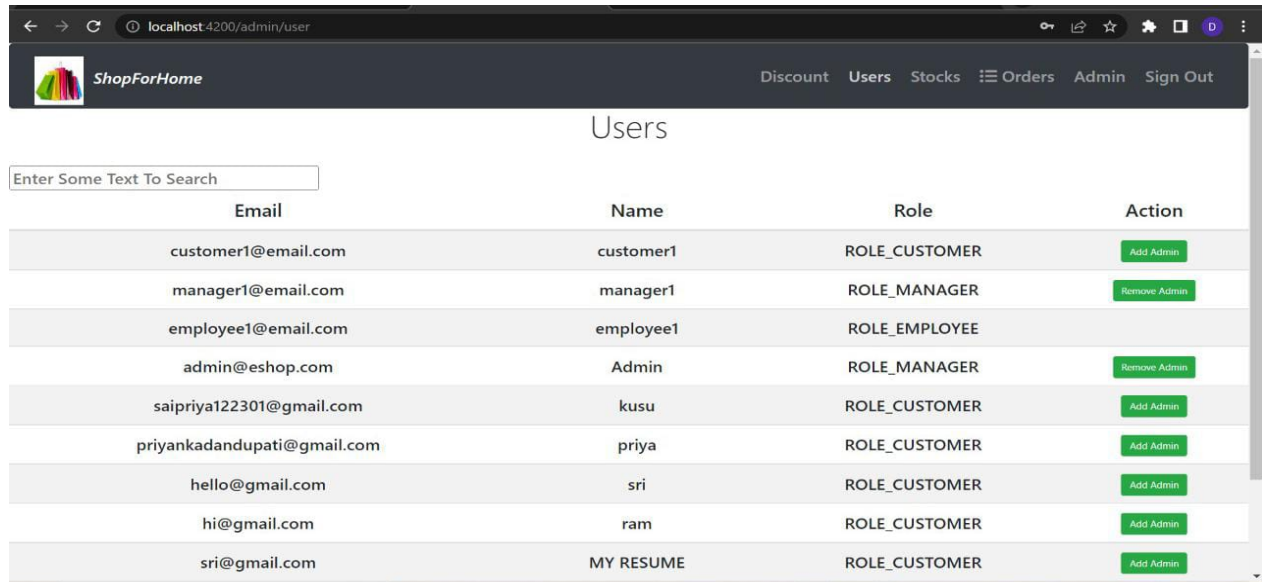
- ☐ customer1@email.com
- ☐ manager1@email.com
- ☐ employee1@email.com
- ☐ admin@eshop.com
- ☐ saipriya122301@gmail.com
- ☐ priyankadandupati@gmail.com
- ☐ hello@gmail.com
- ☐ hi@gmail.com
- ☐ sri@gmail.com
- ☐ ps@gmail.com
- ☐ pras@gmail.com
- ☒ priya@gmail.com

[Add Coupon](#)

Couponcode	User	Action
20	ps@gmail.com	Delete

Users Page:

- It is visible only to the Admin that who were using the application.



ShopForHome

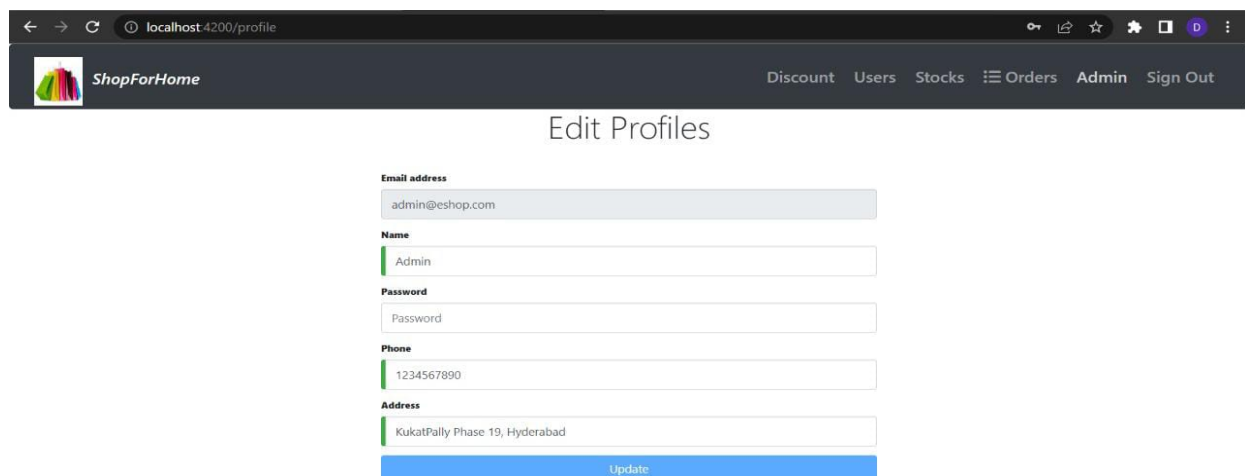
Discount Users Stocks Orders Admin Sign Out

Users

Enter Some Text To Search

Email	Name	Role	Action
customer1@email.com	customer1	ROLE_CUSTOMER	<button>Add Admin</button>
manager1@email.com	manager1	ROLE_MANAGER	<button>Remove Admin</button>
employee1@email.com	employee1	ROLE_EMPLOYEE	
admin@eshop.com	Admin	ROLE_MANAGER	<button>Remove Admin</button>
saipriya122301@gmail.com	kusu	ROLE_CUSTOMER	<button>Add Admin</button>
priyankadandupati@gmail.com	priya	ROLE_CUSTOMER	<button>Add Admin</button>
hello@gmail.com	sri	ROLE_CUSTOMER	<button>Add Admin</button>
hi@gmail.com	ram	ROLE_CUSTOMER	<button>Add Admin</button>
sri@gmail.com	MY RESUME	ROLE_CUSTOMER	<button>Add Admin</button>

Edit Profiles Page:



ShopForHome

Discount Users Stocks Orders Admin Sign Out

Edit Profiles

Email address

admin@eshop.com

Name

Admin

Password

Password

Phone

1234567890

Address

KukatPally Phase 19, Hyderabad

Update

These are all visible to us on the server such as

- <https://localhost:4200/>

Here all the frontend applications get's run by using a integrated terminal and an in-build server in a browser.

For any application or a website we need frontend, backend, database where

- Frontend is the display of the website to our naked eye. It is done with the help of the HTML, CSS, .js etc.,
- Backend is the internal function of the website or application that all it's functioning depends.
- Database is the pre determined data of that given website or an application here in our project all the products available are the pre determined data.

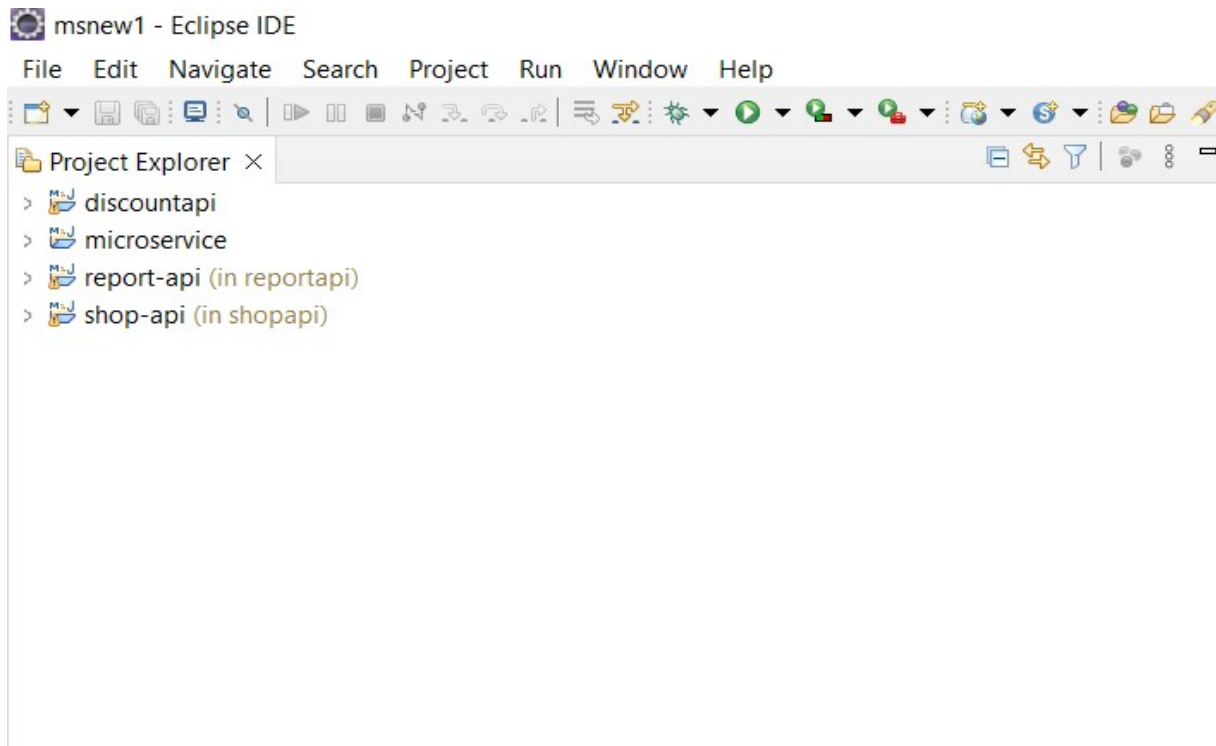
All this operation and functionality is done with the help of the Angular, Js, MySQL.

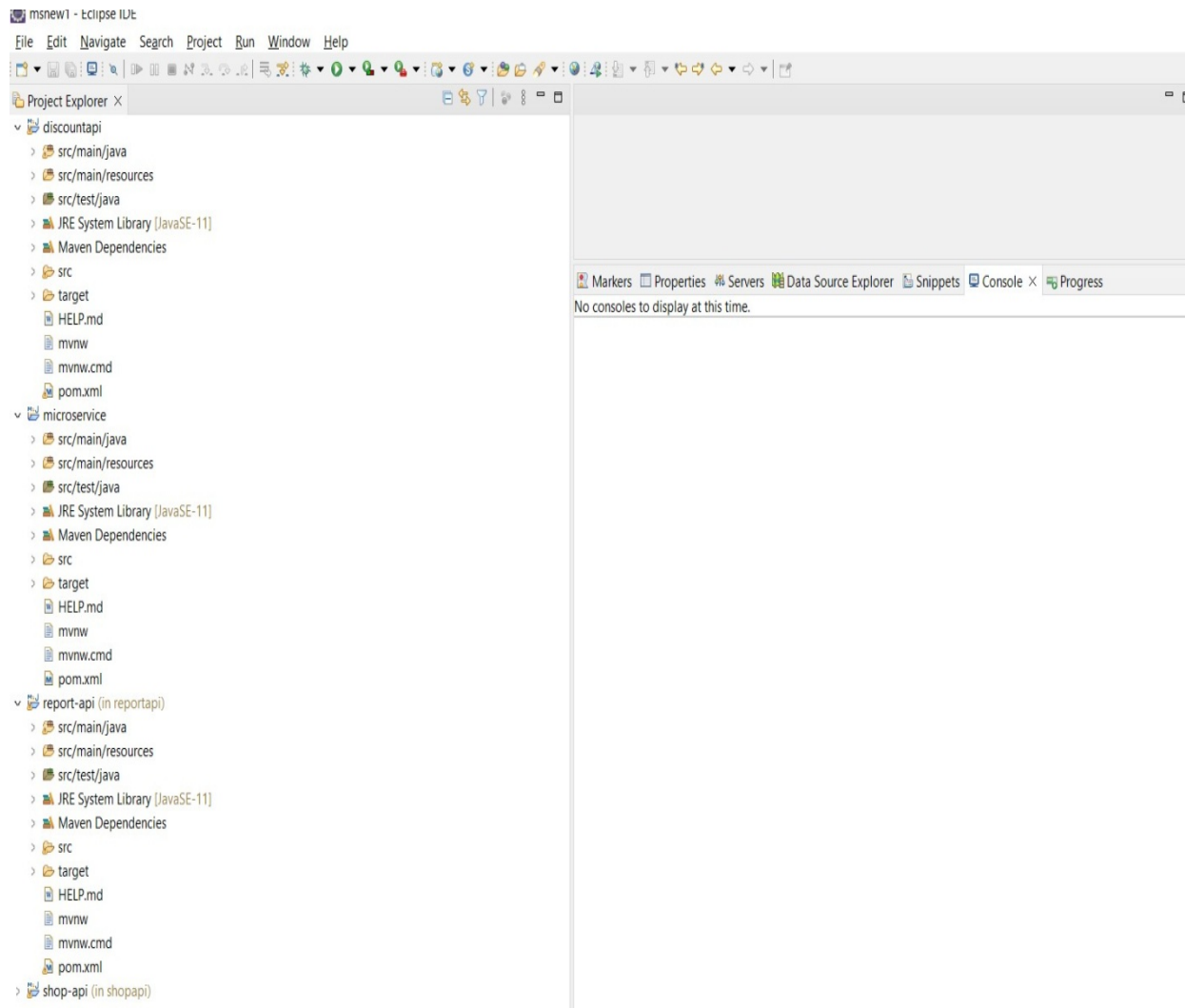
Data Base Structures:

- It is run by using Eclipse or Java Tool Suite application.

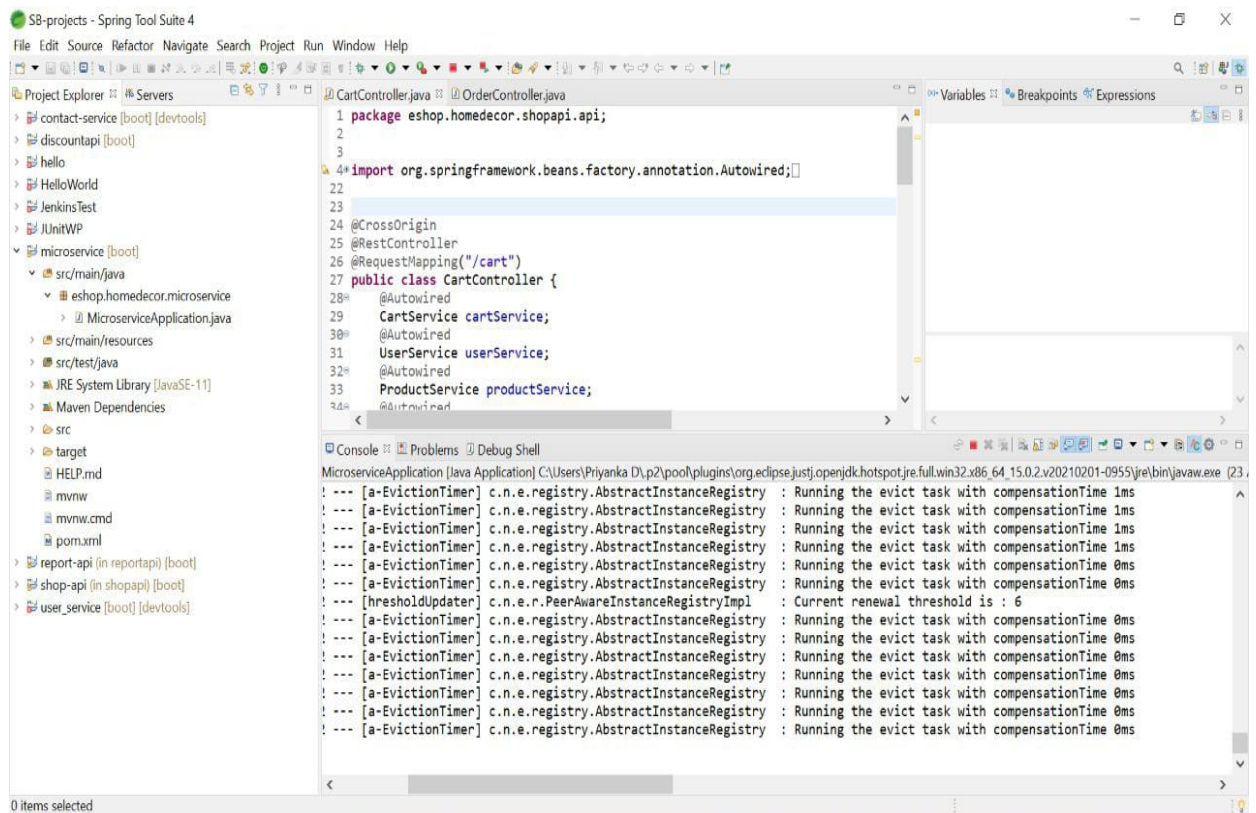
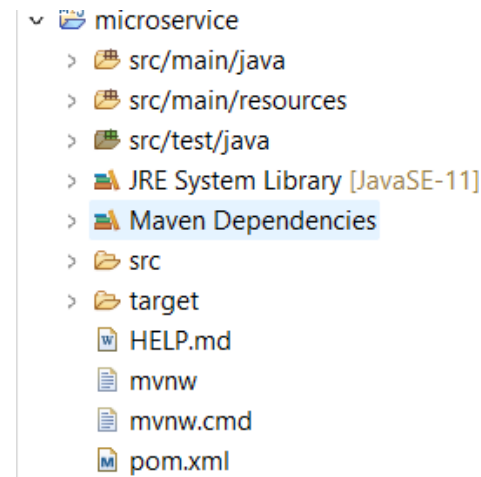
In this backend function of our project we have four main applications they are

1. microservice
2. shop-api
3. discountapi
4. shop-api

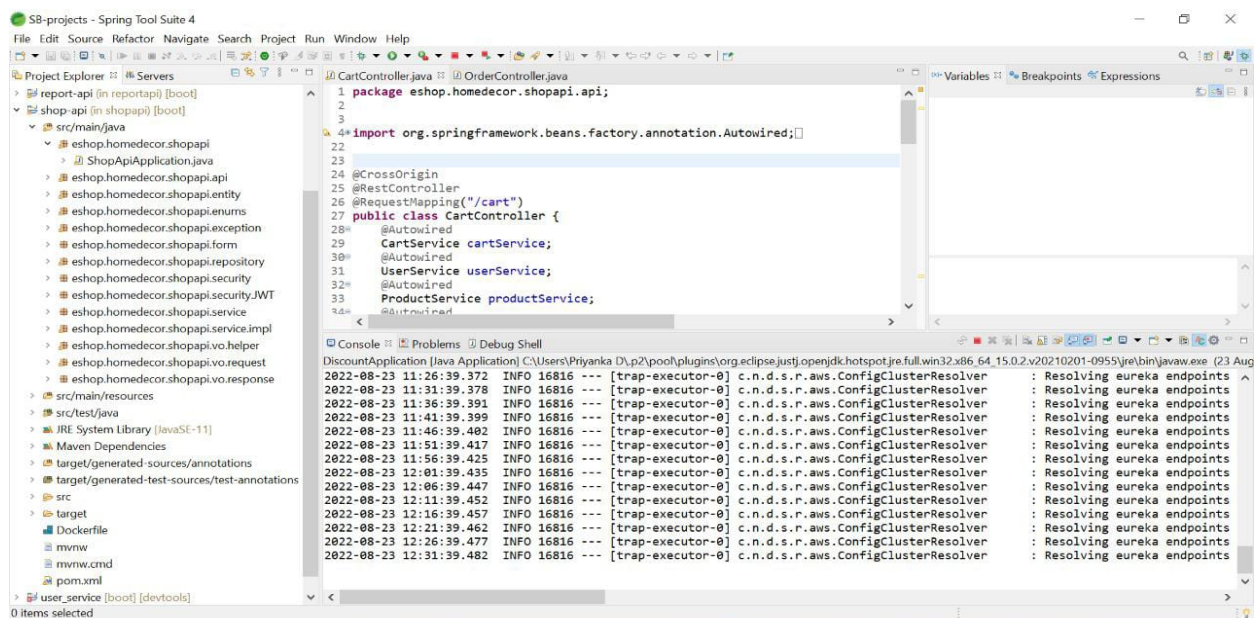
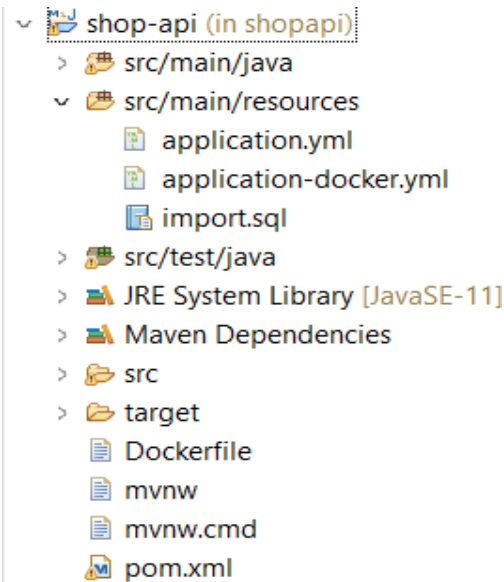




Micro Service:



Shop Micro Service:



Discount Micro Service:

The screenshot displays the Spring Tool Suite 4 IDE interface. The Project Explorer on the left shows the project structure for 'discountapi' and 'microservice'. The main editor window shows the 'CartController.java' file with the following code:

```
1 package eshop.homedecor.shopapi.api;
2
3
4 import org.springframework.beans.factory.annotation.Autowired;
5
6 @CrossOrigin
7 @RestController
8 @RequestMapping("/cart")
9 public class CartController {
10     @Autowired
11     CartService cartService;
12     @Autowired
13     UserService userService;
14     @Autowired
15     ProductService productService;
16 }
```

The Console window at the bottom shows the following output:

```
ShopApiApplication [Java Application] C:\Users\Priyanka D\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_15.0.2.v20210201-0955\jre\bin\javaw.exe (23 Aug
Hibernate: select cart0_.user_id as user_id1_0_0_ from cart cart0_ where cart0_.user_id=?
Hibernate: select count(user0_.id) as col_0_0_ from users user0_
Hibernate: select count(user0_.id) as col_0_0_ from users user0_
2022-08-23 12:21:13.392 INFO 3160 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints v
Hibernate: select user0_.id as id1_6_, user0_.active as active2_6_, user0_.address as address3_6_, user0_.email as email4_6_, us
Hibernate: select cart0_.user_id as user_id1_0_0_ from cart cart0_ where cart0_.user_id=?
Hibernate: select productinfo0_.product_id as product_1_4_, productinfo0_.category_type as category2_4_, productinfo0_.create_time
Hibernate: select count(productinfo0_.product_id) as col_0_0_ from product_info productinfo0_
Hibernate: select user0_.id as id1_6_, user0_.active as active2_6_, user0_.address as address3_6_, user0_.email as email4_6_, us
Hibernate: select cart0_.user_id as user_id1_0_0_ from cart cart0_ where cart0_.user_id=?
Hibernate: select user0_.id as id1_6_, user0_.active as active2_6_, user0_.address as address3_6_, user0_.email as email4_6_, us
Hibernate: select cart0_.user_id as user_id1_0_0_ from cart cart0_ where cart0_.user_id=?
2022-08-23 12:26:13.401 INFO 3160 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints v
2022-08-23 12:31:13.407 INFO 3160 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints v
```


Report Micro Service:

- report-api (in reportapi)
 - src/main/java
 - src/main/resources
 - src/test/java
 - JRE System Library [JavaSE-11]
 - Maven Dependencies
 - src
 - target
 - HELP.md
 - mvnw
 - mvnw.cmd
 - pom.xml

SB-projects - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

Project Explorer Servers

- contact-service [boot] [devtools]
- discountapi [boot]
- hello
- HelloWorld
- JenkinsTest
- JUnitWP
- microservice [boot]
- report-api (in reportapi) [boot]
 - src/main/java
 - eshop.homedecor.report
 - ReportApplication.java
 - eshop.homedecor.reportapi
 - eshop.homedecor.report.entity
 - eshop.homedecor.report.repository
 - eshop.homedecor.report.response
 - eshop.homedecor.report.service.impl
 - src/main/resources
 - src/test/java
 - JRE System Library [JavaSE-11]
 - Maven Dependencies
 - target/generated-sources/annotations
 - target/generated-test-sources/test-annotations
 - src
 - target
 - HELP.md
 - mvnw
 - mvnw.cmd
 - pom.xml
 - shop-api (in shopapi) [boot]
 - user_service [boot] [devtools]

Console Problems Debug Shell

DiscountApplication [Java Application] C:\Users\Priyanka D\p2\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_15.0.2.v20210201-0955\jre\bin\javaw.exe (23 Aug 2022-08-23 11:26:39.372 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 11:31:39.378 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 11:36:39.391 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 11:41:39.399 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 11:46:39.402 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 11:51:39.417 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 11:56:39.425 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 12:01:39.435 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 12:06:39.447 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 12:11:39.452 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 12:16:39.457 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 12:21:39.462 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 12:26:39.477 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

2022-08-23 12:31:39.482 INFO 16816 --- [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints

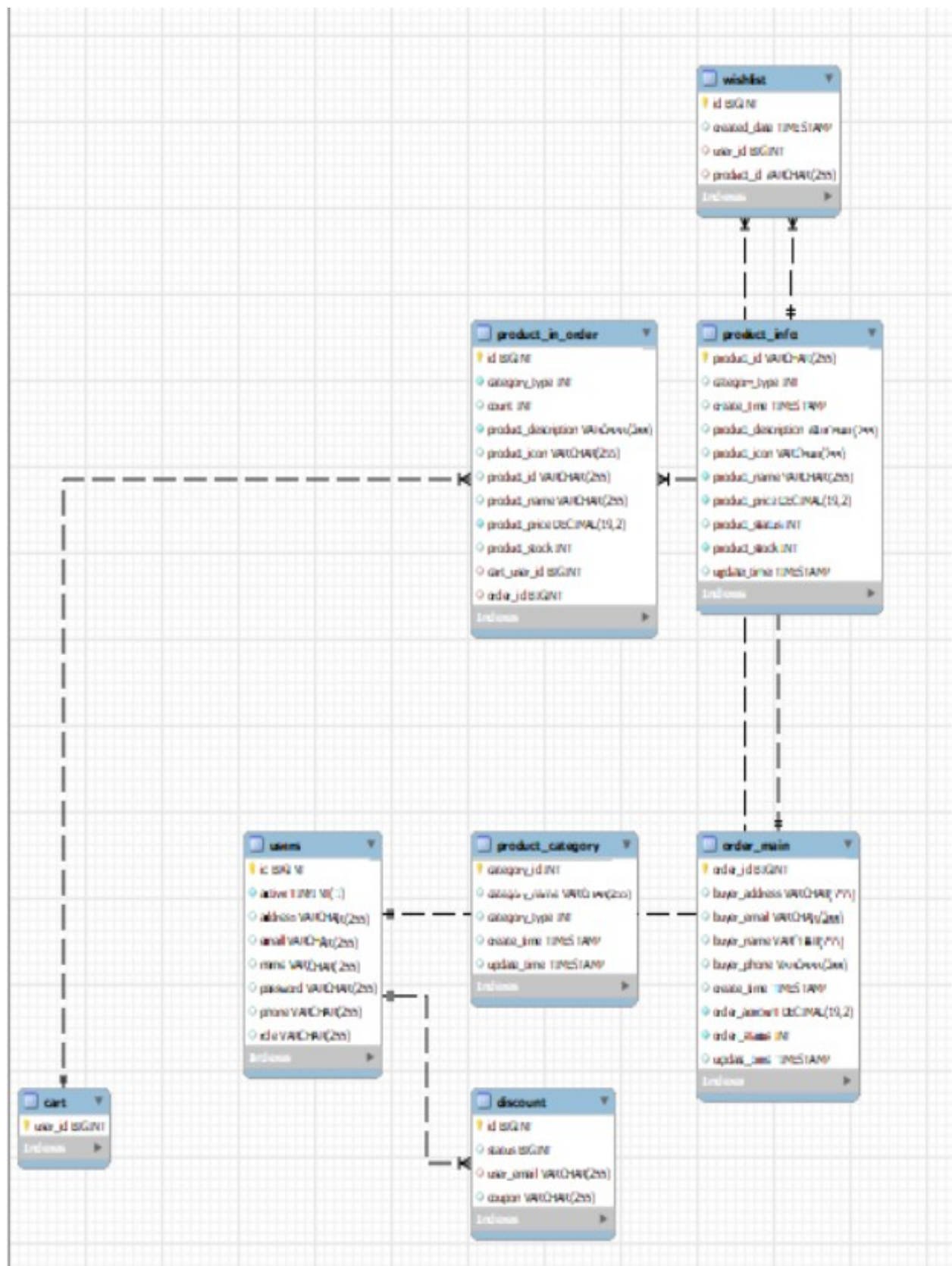
0 items selected

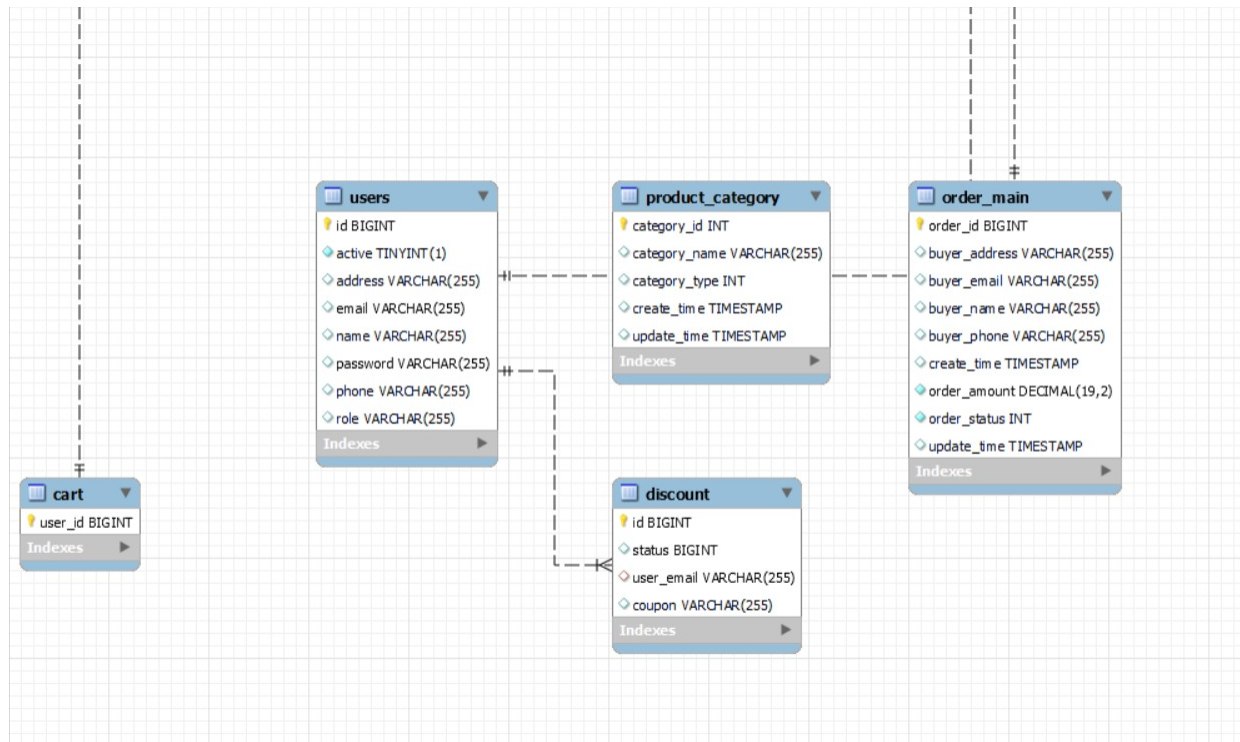
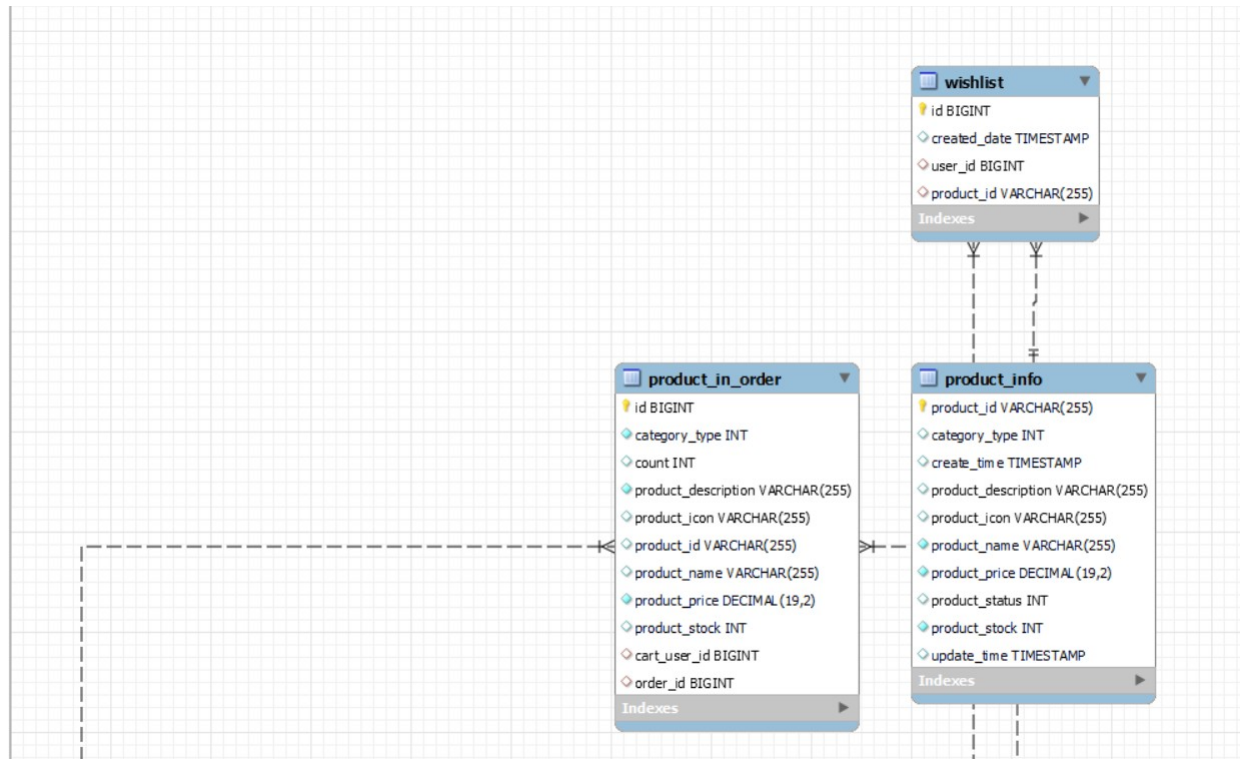
Data Base tables:

- MySQL

The screenshot displays a MySQL database management interface. On the left, the 'SCHEMAS' pane shows a tree view of the database structure. The 'ecommerce' schema is expanded, revealing a list of tables: 'cart', 'discount', 'order_main', 'product_category', 'product_in_order', 'product_info', 'users', and 'wishlist'. The 'product_info' table is currently selected. The main window shows the SQL query 'SELECT * FROM ecommerce.product_info;' and its results in a grid format. The results table has five columns: 'product_id', 'category_type', 'create_time', and 'product_description'. It contains eight rows of data, including product IDs B0011 through B0015 and C0011 through C0012, with their respective category types, creation times, and descriptions.

product_id	category_type	create_time	product_description
B0011	0	2022-07-14 08:04:01	Computer Table with Drawers
B0012	0	2022-07-14 08:04:01	Wooden 4 Seater Dining Table Set
B0013	0	2022-07-14 08:04:01	Colourful Fabric Chair
B0014	0	2022-07-14 08:04:01	Rectangular Wooden Console Table
B0015	0	2022-07-14 08:04:01	Wooden Center Table
C0011	2	2022-07-14 08:04:01	Iron Titan Leaf Metal Wall Art
C0012	2	2022-07-14 08:04:01	Iron Double Ring In Flower Wall Art





Unique Selling Points of the Application

1. The application is designed to keep on running and taking user inputs even after exceptions occur. To terminate the application, an appropriate option needs to be selected.
2. With this application, one can easily access the product items as their own by adding a cart.
3. Total price is also determined by its own.
4. Any user can sign up and there are no restrictions.
5. Online payment is also accepted.

Conclusion

At the end of the project we have created a fully functional e-commerce website for furniture and Home decors.
