INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT (IACSD), AKURDI, PUNE

Documentation On

Style-Hub Online Shopping App

PG-DAC March 2023

Submitted By:

Group No: 10

Roll No.	Name:
233013	Ankit Lipate
233095	Siddhant Moon

Mrs. Sonali Mogal
Project Guide

Mr. Rohit Puranik
Center Coordinator

ABSTRACT

Online Shopping is a web based application intended for online retailers. The main objective of this application is to make it interactive and its ease of use. It would make searching, viewing and selection of a product easier. It contains a sophisticated search engine for users to search for products specific to their needs. The search engine provides an easy and convenient way to search for products where a user can Search for a product interactively and the search engine would refine the products available based on the user's input. The user can then view the complete specification of each product. They can also view the product reviews and also write their own reviews. The main emphasis lies in providing a user friendly search engine for effectively showing the desired results and its drag and drop behavior.

ACKNOWLEDGEMENT

I take this occasion to thank God, almighty for blessing us with his grace and taking our endeavor to a successful culmination. I extend my sincere and heartfelt thanks to our esteemed guide, **Mrs. Sonali Mogal** for providing me with the right guidance and advice at the crucial juncture stand for showing me the right way. I extend my sincere thanks to our respected **Centre Co-Ordinator Mr. Rohit Puranik**, for allowing us to use the facilities available. I would like to thank the other faculty members also, at this occasion. Last but not the least, I would like to thank my friends and family for the support and encouragement they have given me during the course of our work.

Table of Contents

LIST OF FIGURES

INTRODUCTION

Shopping has long been considered a recreational activity by many. Shopping online is no exception. The goal of this application is to develop a web based interface for online retailers. The system would be easy to use and hence make the shopping experience pleasant for the users. The goal of this application is

• To develop an easy to use web based interface where users can search for products, view a complete description of the products and order the products.

• A search engine that provides an easy and convenient way to search for products specific to their needs. The search engine would list a set of products based on the search term and the user can further filter the list based on various parameters.

• A React enabled website with the latest React controls giving attractive and interactive look to the web pages and prevents the annoying post backs.

• A user can view the complete specification of the product along with various images and also view the customer reviews of the product. They can also write their own reviews.

Need of the application

There are large numbers of commercial Online Shopping websites offering large number of products tailored to meet the shopping interests of large number of customers. These online marketplaces have thousands of products listed under various categories.

Problem:

The basic problems with the existing systems are the non-interactive environment they provide to the users.

• The use of traditional user interfaces which make continuous post backs to the server; each post back makes a call to the server, gets the response and then

refreshes the entire web form to display the result. This scenario adds an extra trade off causing a delay in displaying the results

• A search engine that would display the results without allowing the users to further filter the results based on various parameters.

• Use of traditional and non-user friendly interfaces that are hard to use

Solution:

• The motive of this Online Shopping Web Application is to allow the user to play with the search tool and create different combinatorial search criteria to perform exhaustive search.

• Making the application React enabled gets rid of these unnecessary delays letting the user to perform exhaustive search. The users of this application can easily feel the difference between the React empowered user interfaces vs. traditional user interfaces.

• Provide Interactive interface through which a user can interact with different areas of application easily.

PROJECT OBJECTIVE

The purpose of the project is to build an application program to reduce the manual work for managing the Shopping, Internet, Products, Payment. It tracks all the details about the Payment, Bills, Customer.

PROJECT OVERVIEW

This system involves its own database to be maintained. As the information or details about the products are stored in the database (like RDBMS) for the server-side functionalities. The Server process is for dealing with the customer's detail and the items that are shipped to different locations based on the addresses provided by the customers.

The application design contains two modules one is for the customers who wish to buy the articles. And another is for the store owners who maintain and updates the information regarding the articles and about the customers. The end-user to use this product are the common people for whom the application is to be hosted on the web and the admin maintains the database.

The application that is deployed on the customer's database, the information regarding the items is highlighted and forwarded from the database for the customer (front view) based on the choice through the menu list and based on all these searches and transactions the database of all the products is updated at the end of each transaction. The entries for products, into the application, can be made through various screens designed for various levels of users. As soon as the authorized personnel feeds the relevant data into the system, several reports are generated based on the security policy used.

PROJECT SCOPE

The current system can be extended to allow the users to create accounts and Buy products as they require. 7The users could subscribe for price alerts which would enable them to receive messages when price for products fall below a particular level. The current system is confined only to the shopping cart process. It can be extended to have an easy to use check out process. Users can have multiple shipping and billing information saved. During checkout they can use the drag and drop feature to select shipping and billing information.

STUDY OF THE SYSTEM

MODULES:

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

The modules involved are:

- Admin
- Customers
- Delivery Person

Admin:

The Admin is the super user of this application. Only Admin have access to this Admin page. Admin is the owner of one of the Applications . The Admin has all the

information about the other Customers and about all work staff.

This module is divided into different sub modules.

1. Add Category
2. Add Product
3. All Orders
4. Delete Products
5. Delete User
6. Assign Order Deliver

UML Diagram

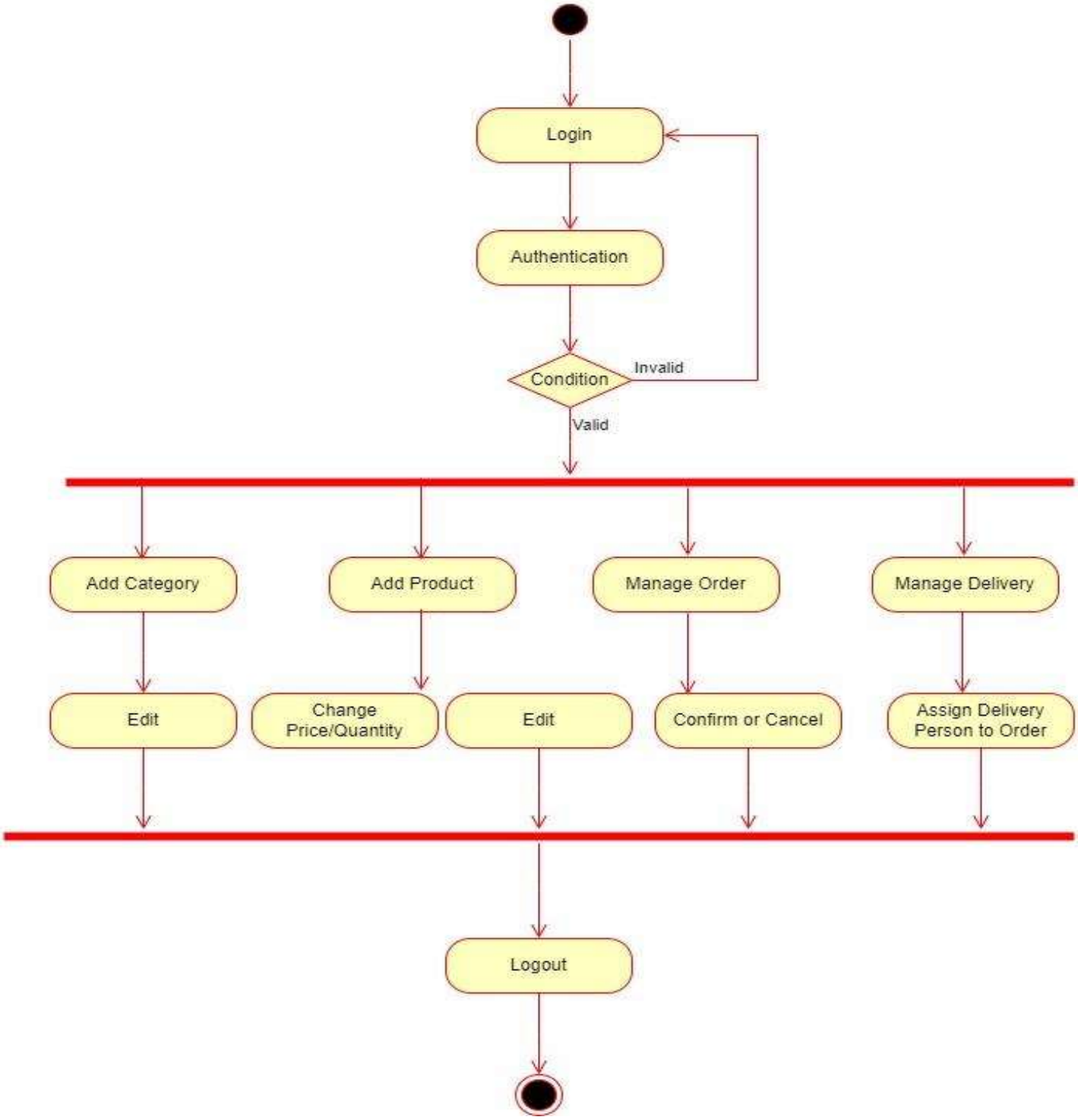


Figure 1 Admin Activity Diagram

- **Add Category**

  Clothing may consist of different types, so it is necessary to create categories to filter them accordingly.

- **Add Product**

  Admin can add the product based on its Category.

- **Manage Order**

  Admin have to manage the Orders ordered by their customers.

- **Manage Delivery**

  The Admin have managed the Deliveries according to customers Order, order date. Admin is also responsible for delivery according to customer order.

- **Change Price/Quantity**

  Admin can change the price/quantity of the products as per requirement.

- **Confirm and Cancel**

  Admin are allowed to cancel and confirm the Orders.

- **Assign delivery person to Orders**

  Only Admin can assign the delivery to the delivery person.

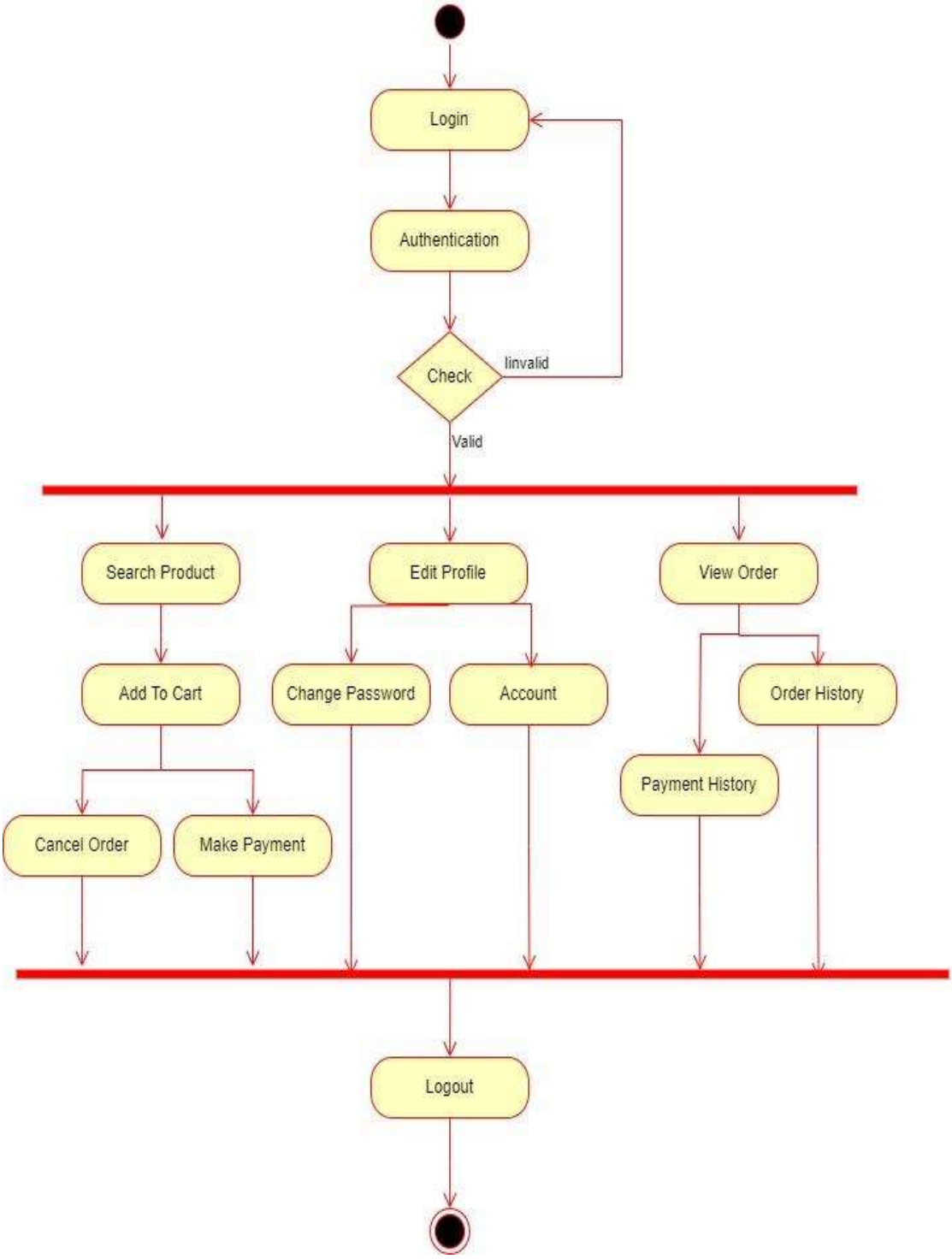


Figure 2 Customers Activity Diagram

- **Search Products**

  Customers are allowed to search the products as per their will

- **Edit Profile**

  Edit option is given to the customers to edit their profile

- **View Orders**

  Customers can view their orders as they may have added to the cart

- **Add to Cart**

  Customers can add the product to their cart

- **Change Password**

  Customer can change the password of login to the application.

- **Account**

  Customer can place the order in cart and check it out and make payment with debit card.

- **Order History**

  Customer can check the Order history which they order early.

- **Payment History**

  Customer can check the payment history for any product which they placed any order.

- **Cancel Order**

  Customer can cancel order if they want to do that or any product is not like but within a week.

- **Make Payment**

  Customer can make payment after add product in cart with valid card name, cvv and 16-digit card number.

SYSTEM ANALYSIS

As the goal of the application is ease of use and to provide an interactive interface, extensive research has been done to gain an insight into the needs and behaviors of various users. The working of the application is made convenient and easy to use for the end user.

Users can be classified into two types based on their knowledge of the products that suit their needs. They can be classified as users who know about the product that would satisfy their needs and who have to figure out the product that would satisfy their needs. Users who know about the product should be able to find the product easily with the click of a button. Users who have to figure out the product that would satisfy their needs could use a search term to find a list of products and then should be able to filter the results based on various parameters like product type, manufacturer, price range, platform supported etc.

SYSTEM REQUIREMENT SPECIFICATION

Document: Software Requirement Specification Document

Title: Software Requirement Specification for Online Clothes Shopping (Style Hub).

Objective (Purpose):

The online shopping System for products (Clothes) Web Application is intended to provide complete solution for Consumers. It will enable customers to browse through virtual shop and purchase products online without visiting the shop physically.

Scope:

Customer will be able to place and cancel order. The System will show the order history for each customer. Customer will be able to see his/her order status. The ecommerce website will allow customers to browse, search, and purchase clothes. It will also provide inventory management and order processing functionalities.

Requirements:

Functional Requirement:

User Account

Here are the detailed functionalities of each user type in an online clothes shopping store

Guest User:

A guest user is someone who visits the website without registering. The following are the functionalities of a guest user

Browse products: A guest user can browse through the products available on the website.

Register: A guest user can register to become a registered user and gain access to additional functionalities

Registered User:

A registered user is someone who has created an account on the website. The following are the functionalities of a registered user:

Buy products: A registered user can purchase products from the website.

Add products to cart: A registered user can add products to the shopping cart and proceed to checkout.

Review products: A registered user can leave reviews for products they have purchased.

View order history: A registered user can view their order history and track the status of their orders.

Admin:

The admin is responsible for managing the website and ensuring that everything runs smoothly.

The following are the functionalities of an admin:

Add and delete categories: The admin can create new categories for products and delete existing categories.

Add and delete products: The admin can add new products to the website and delete existing products.

Delete user: The admin can delete a user's account if necessary.

Assign delivery personnel to orders: The admin can assign a delivery person to deliver an order to a customer.

Delivery Person:

The delivery person is responsible for delivering products to customers. The following are the functionalities of a delivery person

Register: A delivery person can register on the website.

Update delivery status: A delivery person can update the status of a delivery (e.g., processing, on the way, Delivered) in the system.

Registration and creation of user profile

Guest User Registration:

A guest user can register to become a registered user by clicking on the "Register" button or link on the website. The following steps are involved in registering as a guest user Enter personal details: The user will be asked to enter their personal details, such as their name, email, phone number, address, and password.

Admin User Creation:

Creating an admin user account involves the following steps:

Access admin panel: The website owner or authorized personnel will need to access the admin panel of the website.

Create admin account: From the admin panel, the user can create an admin account by entering their personal details, such as their name, email address, and password.

Assign admin role: Once the admin account has been created, the user will need to assign the admin role to the account in order to access the admin functionalities.

Delivery Person Registration:

Registering as a delivery person involves the following steps:

Click on "Register": The delivery person clicks on the "Register" button or link on the website.

Enter personal details: The delivery person will be asked to enter their personal details, such as their name, email address, and password.

Access delivery functionalities: Once the delivery person has registered, they can access the delivery

functionalities such as updating the delivery status of orders.

Browsing/Ordering/Reviewing Products

As a registered user of an online clothes shopping store, you can enjoy a seamless

shopping experience from the comfort of your own home. Here is a step-by-step guide on how to browse, order and review products on an online clothes shopping store

Browsing Products:

Once you are logged into your account, you can start browsing the store's product catalog. You can browse products by categories, such as shirts, pants, t-shirts, trousers, jeans, etc. Alternatively, you can use the search bar to find specific products. Each product will have a product description, price, images, and customer reviews.

Adding Products to Cart:

Once you have found the product you wish to purchase, click on the 'Add to Cart' button. You can continue browsing and adding products to your cart. Your cart will keep track of all the products

you have added, and you can view it by clicking on the cart icon.

Checkout:

When you are ready to make your purchase, go to your cart and click on the 'Checkout' button. You will be taken to a page where you can review your order, add/remove products, and enter your shipping and payment details. You may also be prompted to select a delivery option.

Order Confirmation:

After you have placed your order, you will receive an order confirmation email. The store may also provide you with a tracking number so you can track your package's delivery status.

Reviewing Products:

After you have received your product, you can log back into your account and leave a review.

This helps other customers make informed purchasing decisions. You can rate the product, leave a

comment, and upload photos.

Overall, an online clothes shopping store provides a convenient way to shop for clothes. As a

registered user, you can enjoy the benefits of faster checkout, order tracking, and the ability to leave

reviews.

View Order History

As a registered user of an online clothes shopping store, you can easily view your order history to keep track of all the products you have ordered in the past. Here is a step-by-step guide on how to view your ordered products history:

Login to Your Account:

To view your order history, you need to log in to your account on the online clothes shopping store. If you have not created an account, you may need to create one using your email address and a password.

Navigate to Order History:

Once you are logged in, you can navigate to your order history by clicking on the "My Orders" tab on the website. This will take you to a page where you can view all your past orders.

Select the Order:

In your order history, you will see a list of all your past orders. You can click on the order you wish to view to see the details of that order. This will include information such as the date of the order, the products ordered, the order status, and the order total.

View Product Details:

To view the details of the products ordered, click on the product name or image. This will take you to a page with more information about the product, such as the product description, price, and images.

NON FUNCTIONAL REQUIREMENTS

Performance

Number of Concurrent Users:

Online clothes shopping Store shall be able to handle at least 1000 transactions/orders per Ordering/Browsing of Products:

The system is susceptible to any temporary server failure since it uses the strong feature of Spring Boot and Spring Data JPA. Hence the browsing and Ordering of products will be continued even if the sever gets disconnected in between the session

Other Requirements:

Hardware Interfaces

The SPMS is expected to function on Intel PIII 900 MHz Processor equivalent or above, 128 MB RAM, 20 GB HDD.

Software Interfaces

The SPMS shall work on MS Windows operating systems family (MS Windows 98, MS Windows NT Workstation, MS Windows 2000, MS Windows XP). It configures to work with Oracle database. This System works on Apache Tomcat server. It uses browser IE 5.0 & above. It uses IIS 5.0 server.

SYSTEM DESIGN

System design is the solution for the creation of a new system. This phase focuses on the detailed implementation of the feasible system. Its emphasis on translating design. Specifications to performance specification. System design has two phases of development.

- Logical Design
- Physical Design

During logical design phase the analyst describes inputs (sources), outputs(destinations), databases (data sores) and procedures (data flows) all in a format that meets the user requirements. The analyst also specifies the needs of the user at a level that virtually determines the information flow in and out of the system and the data resources. Here the logical design is done through data flow diagrams and database design. The physical design is followed by physical design or coding. Physical design produces the working system by defining the design specifications, which specify exactly what the candidate system must do. The programmers write the necessary programs that accept input from the user, perform necessary processing on accepted data and produce the required report on a hard copy or display it on the screen.

INPUT AND OUTPUT DESIGN

INPUT DESIGN:

Input design is the link that ties the information system into the world of its users. The input design involves determining the inputs, validating the data, minimizing the data entry and providing a multi-user facility. Inaccurate inputs are the most common cause of errors in data processing. Errors entered by the data entry operators can be controlled by input design. The user-originated inputs are converted to a computer-based format in the input design. Input data are collected and organized into groups of similar data. Once identified, the appropriate input media are selected for processing. All the input data are validated and if any data violates any conditions, the user is warned by a message. If the data satisfies all the conditions, it is transferred to the appropriate tables in the database. In this project the customer details are to be entered at the time of registration. A page is designed for this purpose which is user friendly and easy to use. The design is done such that users get appropriate messages when exceptions occur.

OUTPUT DESIGN:

Computer output is the most important and direct source of information to the user. Output design is a very important phase since the output needs to be in an efficient manner. Efficient and intelligible output design improves the system relationship with the user and helps in decision making. Allowing the user to view the sample screen is important because the user is the ultimate judge of the quality of output. The output module of this system is the selected notifications.

DATABASE DESIGN

DATABASE

Databases are the storehouses of data used in the software systems. The data is stored in tables inside the database. Several tables are created for the manipulation of the data for the system. Two essential settings for a database are

- Primary key - the field that is unique for all the record occurrences
- Foreign key - the field used to set relation between tables Normalization is a technique to avoid redundancy in the tables.

SYSTEM TOOLS

The various system tools that have been used in developing both the front end and the back end of the project are being discussed in this chapter.

FRONT END:

React is a library which is developed by Facebook are utilized to implement the frontend. React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.

BACKEND:

The back end is implemented using MySQL which is used to design databases.

MySQL:

MySQL is the world's second most widely used open-source relational database management system (RDBMS). The SQL phrase stands for Structured Query Language.

Spring-Boot:

This is used to connect MYSQL and fetch data from database and store the data in database. The Spring Framework is an application framework and inversion of control container for the Java platform. The framework's core features can be used by any Java application, but there are extensions for building web applications on top of the Java EE (Enterprise Edition) platform. Although the framework does not impose any specific programming model, it has become popular in the Java community as an addition to the Enterprise JavaBeans (EJB) model. The Spring Framework is Open-source Framework.

0 Level DFD

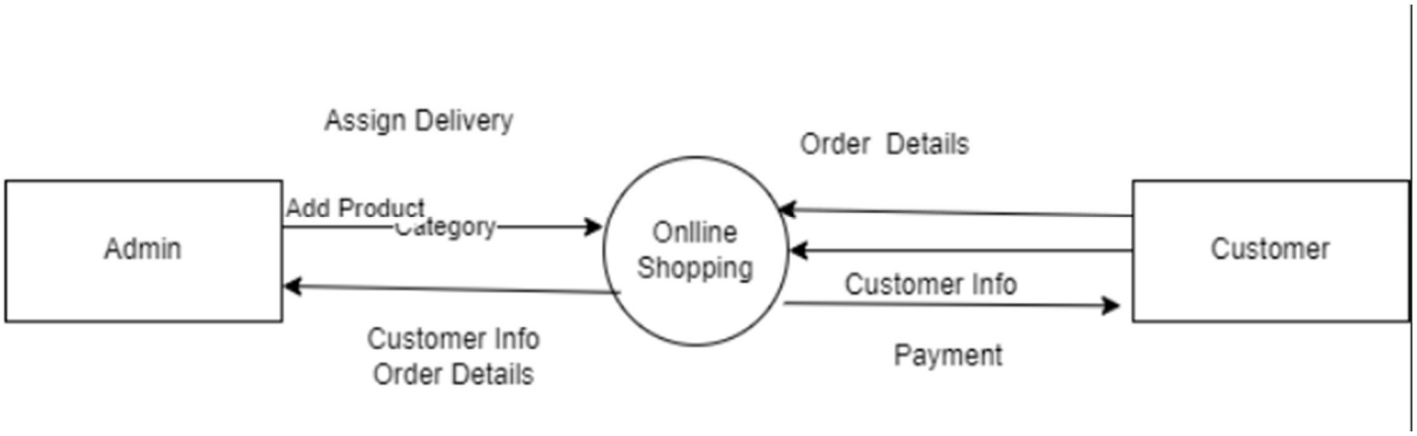


Figure 3 0 Level DFD

1 Level DFD for Customer

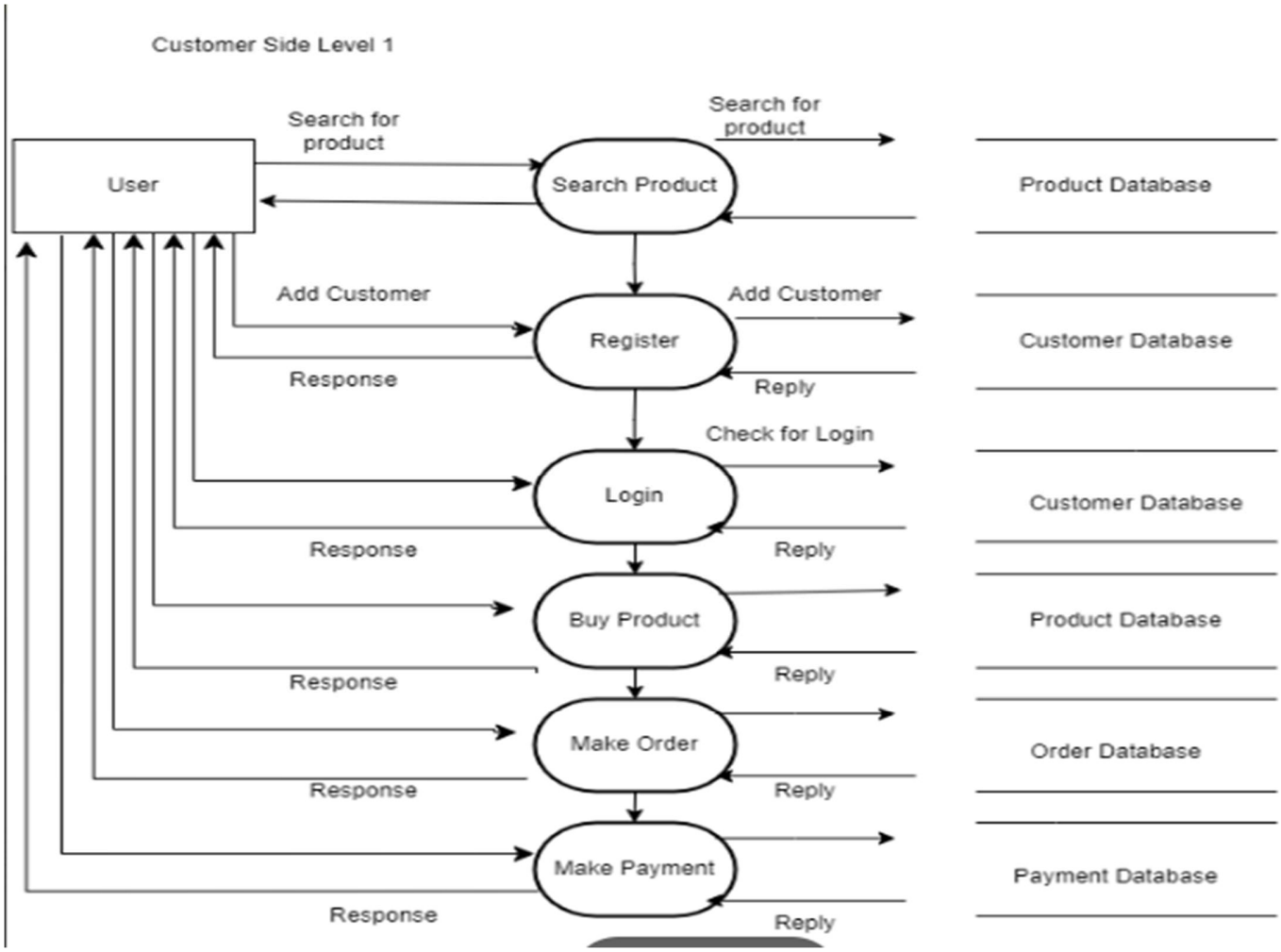


Figure 4 1 Level DFD for Customer

1 Level DFD for Admin

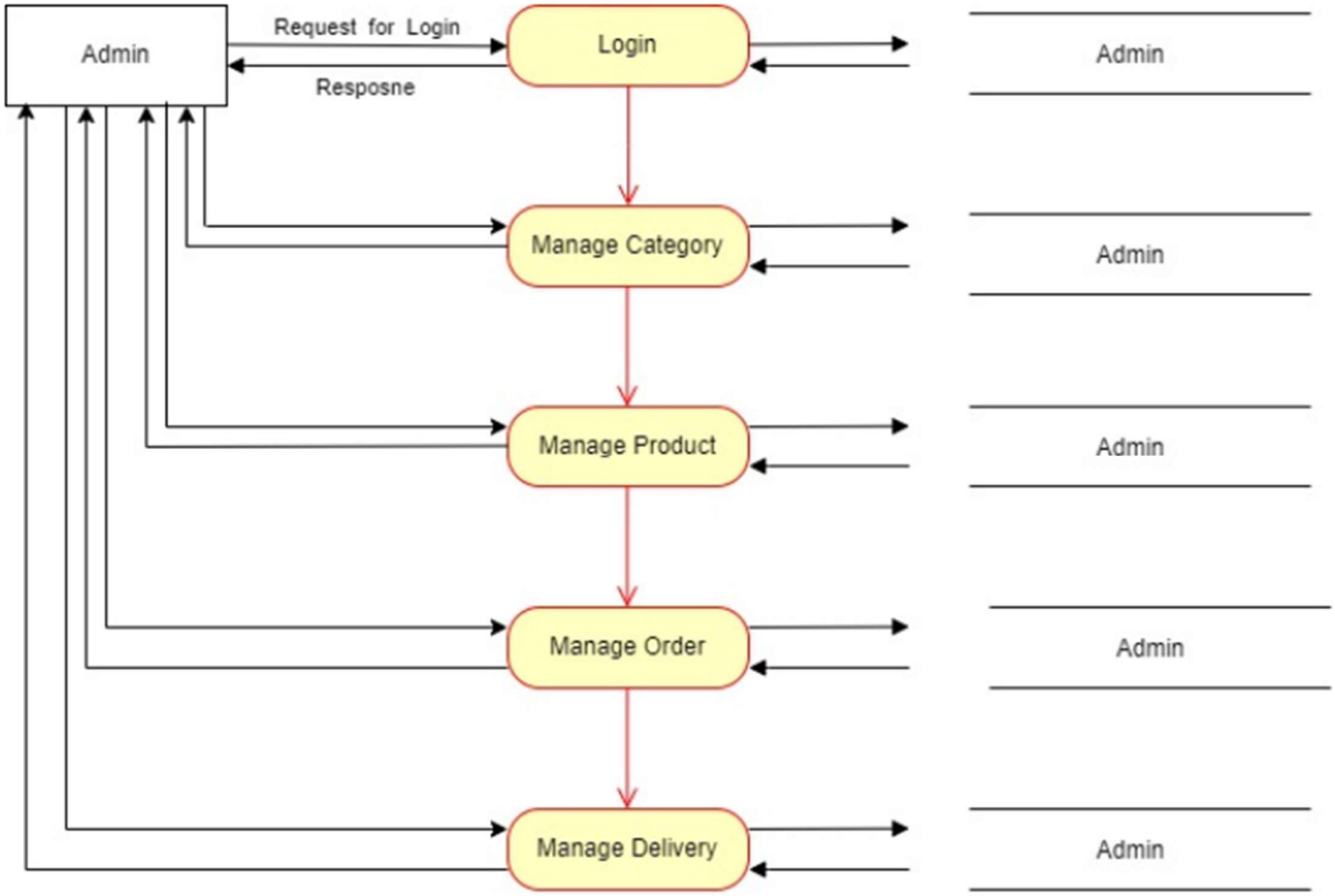


Figure 5 1 Level DFD for Admin

E-R Diagram:

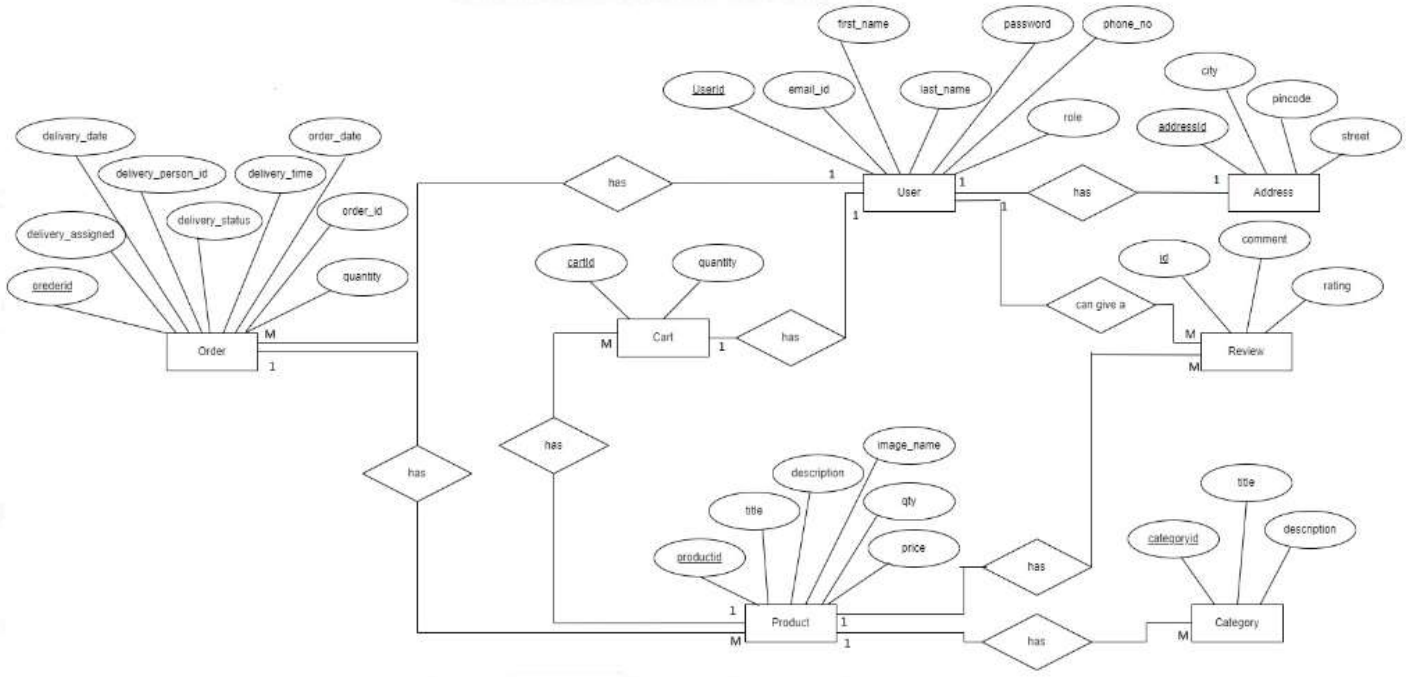


Figure 6 E-R Diagram

Class Diagram

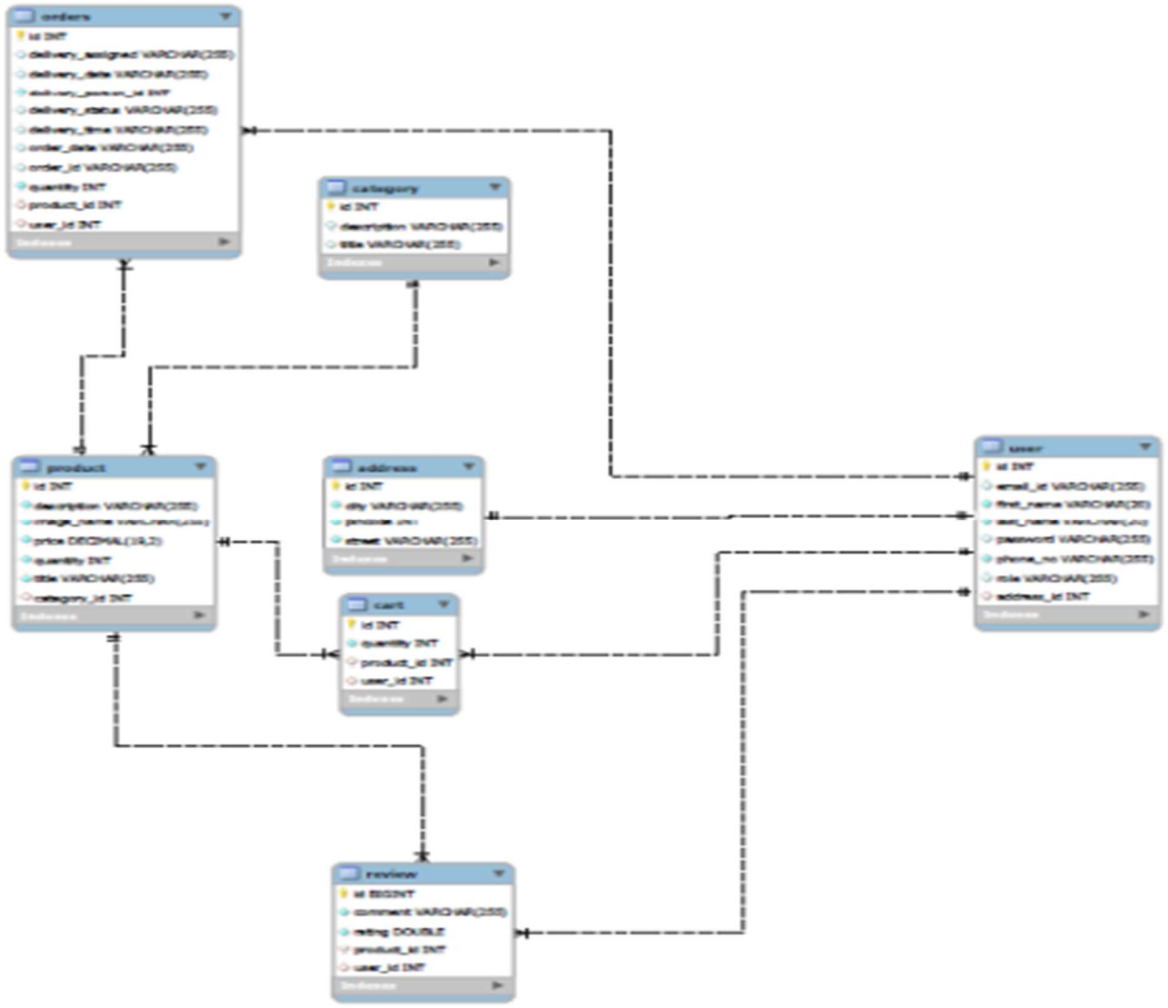


Figure 7 Class Diagram

Sequence Diagram:

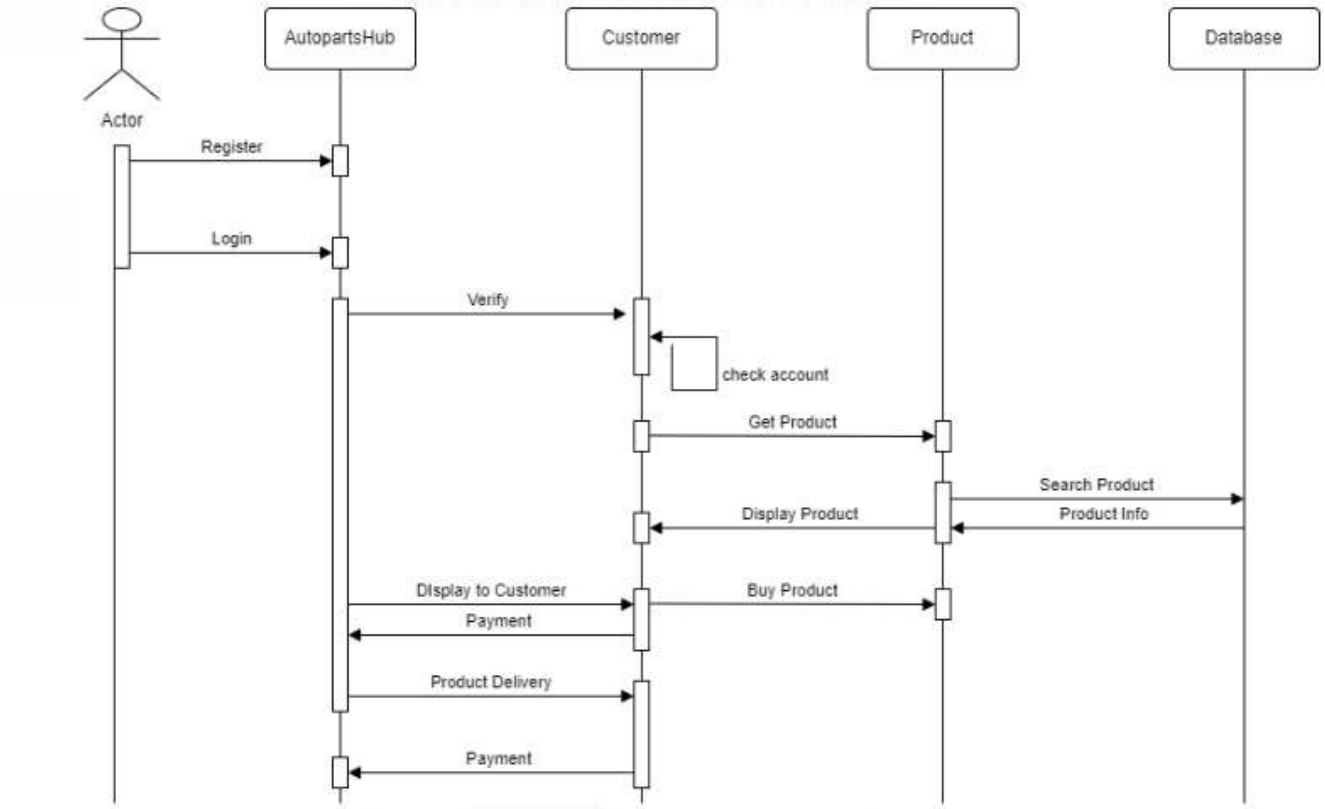


Figure 8 Sequence Diagram

TABLE STRUCTURE:

Tables:

```
+---------------------+
| Tables_in_stylehub  |
+---------------------+
| address             |
| cart                |
| category            |
| orders              |
| product             |
| review              |
| user                |
+---------------------+
```

Address:

```
+---------+--------------+------+-----+---------+----------------+
| Field   | Type         | Null | Key | Default | Extra          |
+---------+--------------+------+-----+---------+----------------+
| id      | int          | NO   | PRI | NULL    | auto_increment |
| city    | varchar(255) | NO   |     | NULL    |                |
| pincode | int          | NO   |     | NULL    |                |
| street  | varchar(255) | NO   |     | NULL    |                |
+---------+--------------+------+-----+---------+----------------+
```

Cart :

```
mysql> desc cart;
+------------+------+------+-----+---------+----------------+
| Field      | Type | Null | Key | Default | Extra          |
+------------+------+------+-----+---------+----------------+
| id         | int  | NO   | PRI | NULL    | auto_increment |
| quantity   | int  | NO   |     | NULL    |                |
| product_id | int  | YES  | MUL | NULL    |                |
| user_id    | int  | YES  | MUL | NULL    |                |
+------------+------+------+-----+---------+----------------+
4 rows in set (0.00 sec)
```

Category:

```
mysql> desc category;
+-------------+--------------+------+-----+---------+----------------+
| Field       | Type         | Null | Key | Default | Extra          |
+-------------+--------------+------+-----+---------+----------------+
| id          | int          | NO   | PRI | NULL    | auto_increment |
| description | varchar(255) | YES  |     | NULL    |                |
| title       | varchar(255) | YES  |     | NULL    |                |
+-------------+--------------+------+-----+---------+----------------+
3 rows in set (0.00 sec)
```

Orders:

```
mysql> desc orders;
+-------------------+--------------+------+-----+---------+----------------+
| Field             | Type         | Null | Key | Default | Extra          |
+-------------------+--------------+------+-----+---------+----------------+
| id                | int          | NO   | PRI | NULL    | auto_increment |
| delivery_assigned | varchar(255) | YES  |     | NULL    |                |
| delivery_date     | varchar(255) | YES  |     | NULL    |                |
| delivery_person_id| int          | NO   |     | NULL    |                |
| delivery_status   | varchar(255) | YES  |     | NULL    |                |
| delivery_time     | varchar(255) | YES  |     | NULL    |                |
| order_date        | varchar(255) | YES  |     | NULL    |                |
| order_id          | varchar(255) | YES  |     | NULL    |                |
| quantity          | int          | NO   |     | NULL    |                |
| product_id        | int          | YES  | MUL | NULL    |                |
| user_id           | int          | YES  | MUL | NULL    |                |
+-------------------+--------------+------+-----+---------+----------------+
11 rows in set (0.00 sec)
```

Product:

```
mysql> desc product;
+-------------+---------------+------+-----+---------+----------------+
| Field       | Type          | Null | Key | Default | Extra          |
+-------------+---------------+------+-----+---------+----------------+
| id          | int           | NO   | PRI | NULL    | auto_increment |
| description | varchar(255)  | NO   |     | NULL    |                |
| image_name  | varchar(255)  | NO   |     | NULL    |                |
| price       | decimal(19,2) | NO   |     | NULL    |                |
| quantity    | int           | NO   |     | NULL    |                |
| title       | varchar(255)  | NO   |     | NULL    |                |
| category_id | int           | YES  | MUL | NULL    |                |
+-------------+---------------+------+-----+---------+----------------+
7 rows in set (0.00 sec)
```

Review:

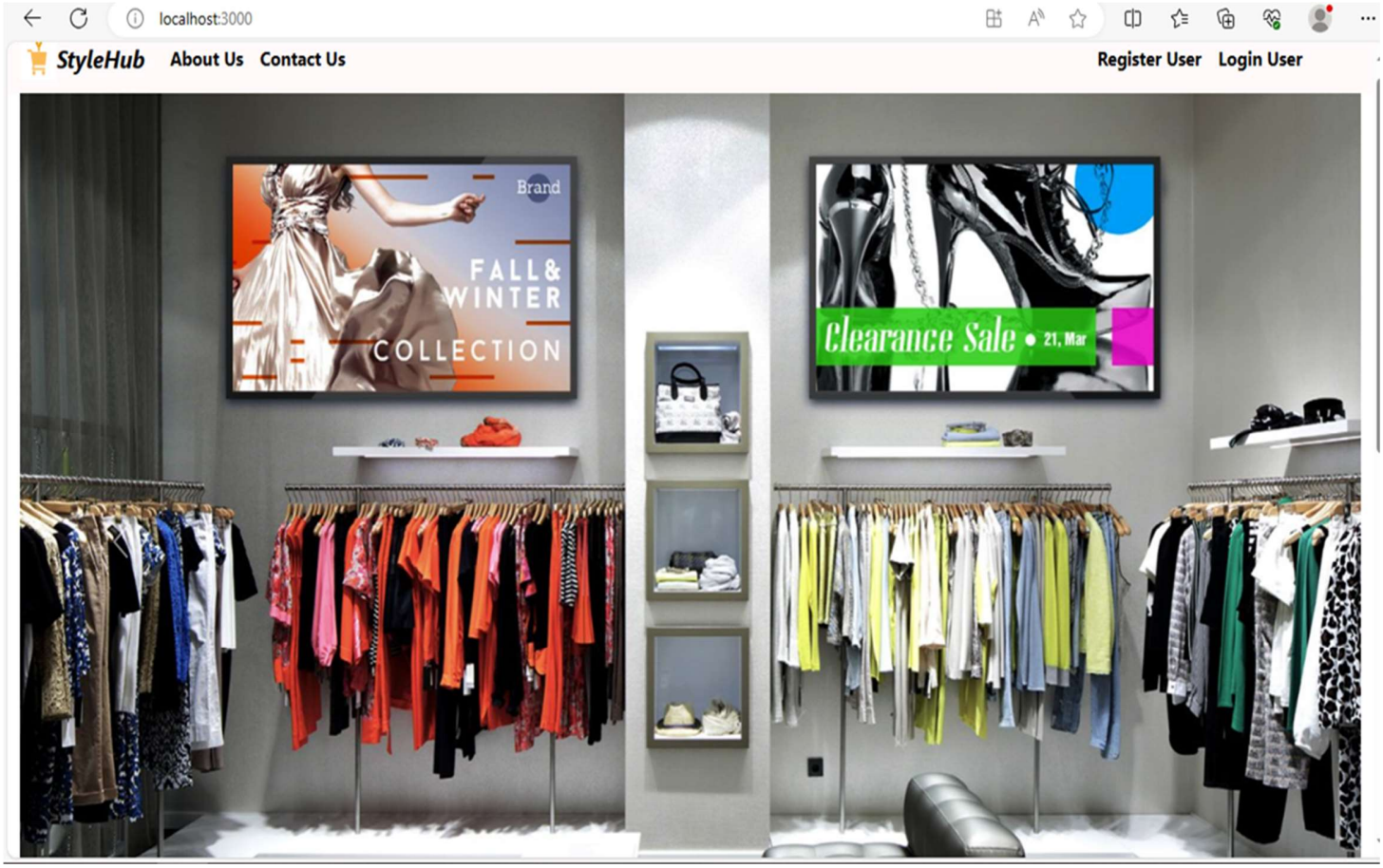
```
mysql> desc review;
+------------+--------------+------+-----+---------+----------------+
| Field      | Type         | Null | Key | Default | Extra          |
+------------+--------------+------+-----+---------+----------------+
| id         | bigint       | NO   | PRI | NULL    | auto_increment |
| comment    | varchar(255) | NO   |     | NULL    |                |
| rating     | double       | NO   |     | NULL    |                |
| product_id | int          | YES  | MUL | NULL    |                |
| user_id    | int          | YES  | MUL | NULL    |                |
+------------+--------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)
```

User :

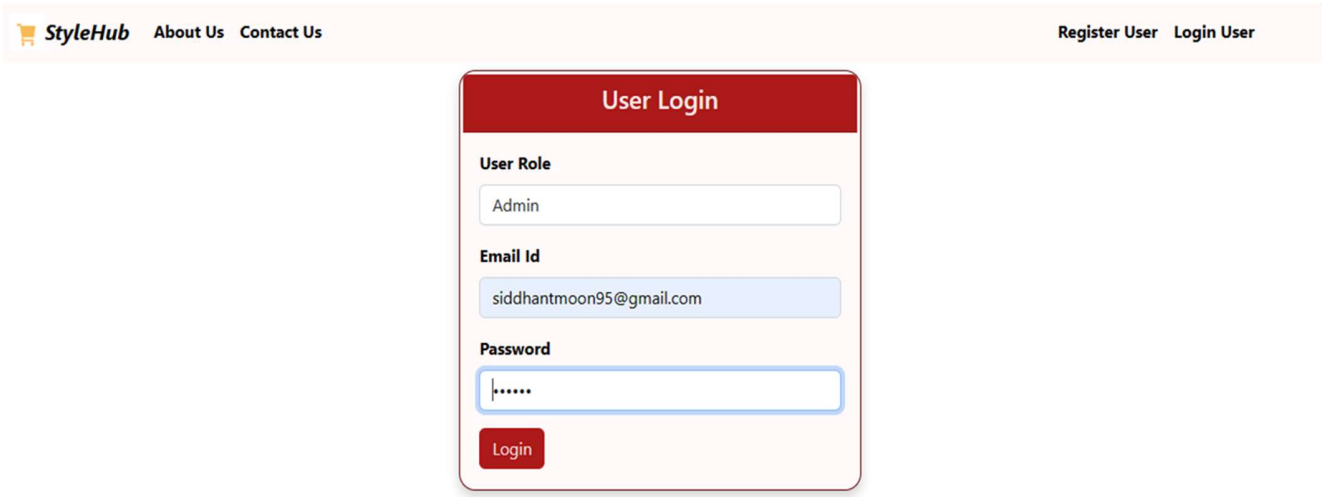
```
mysql> desc user;
+------------+--------------+------+-----+---------+----------------+
| Field      | Type         | Null | Key | Default | Extra          |
+------------+--------------+------+-----+---------+----------------+
| id         | int          | NO   | PRI | NULL    | auto_increment |
| email_id   | varchar(255) | NO   | UNI | NULL    |                |
| first_name | varchar(20)  | NO   |     | NULL    |                |
| last_name  | varchar(20)  | NO   |     | NULL    |                |
| password   | varchar(255) | NO   |     | NULL    |                |
| phone_no   | varchar(255) | NO   |     | NULL    |                |
| role       | varchar(255) | YES  |     | NULL    |                |
| address_id | int          | YES  | MUL | NULL    |                |
+------------+--------------+------+-----+---------+----------------+
8 rows in set (0.00 sec)
```

PROJECT DIAGRAMS

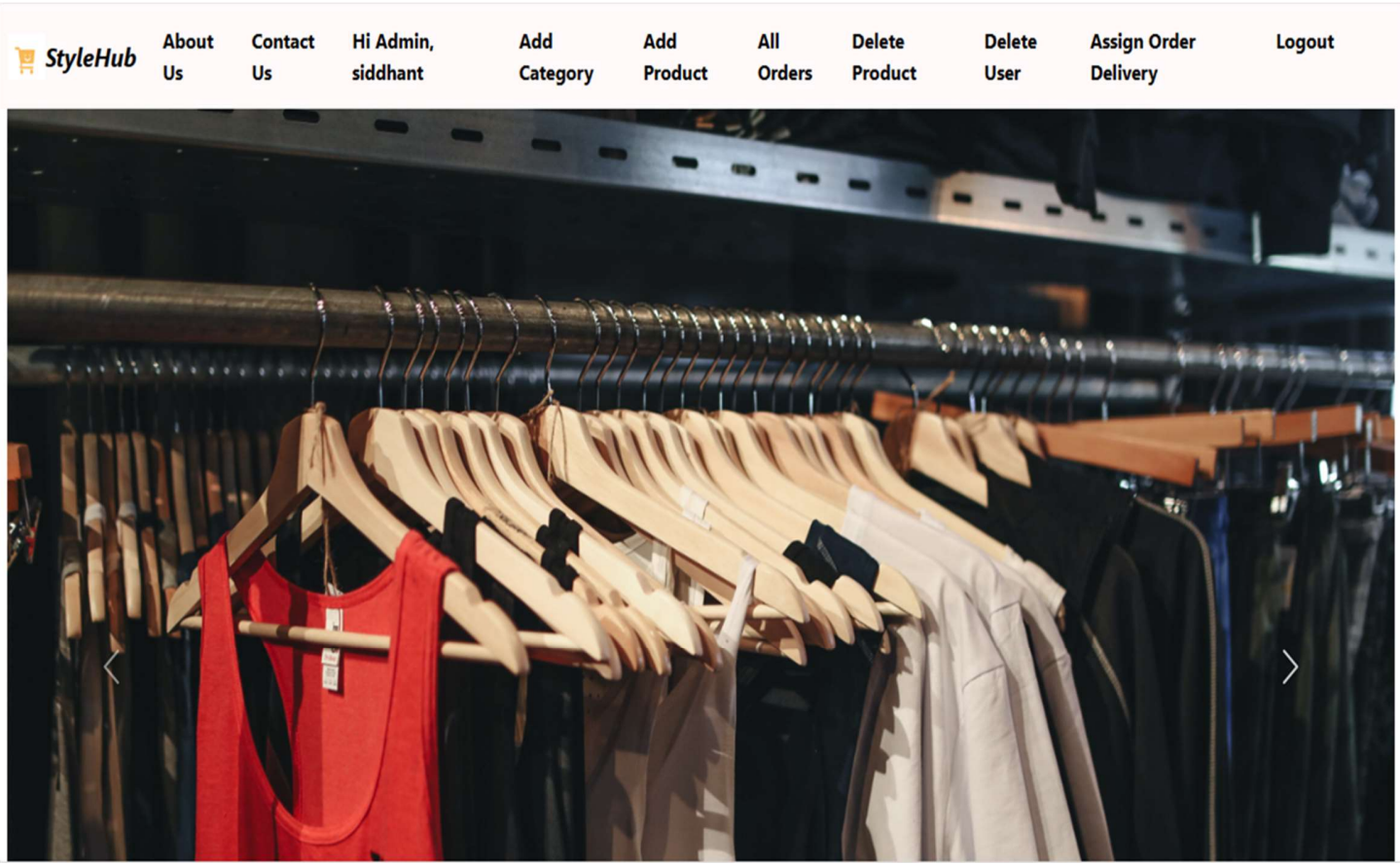
Home Page



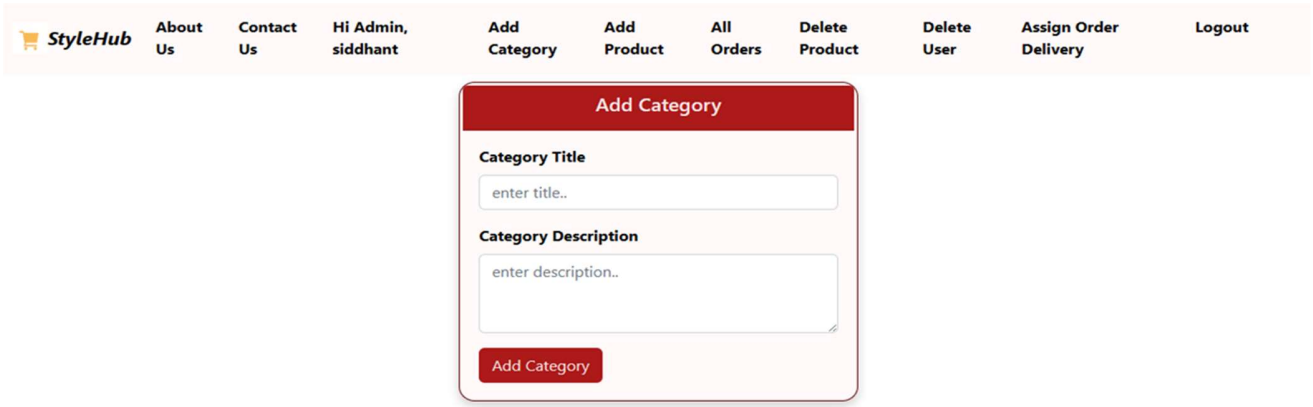
Admin Login



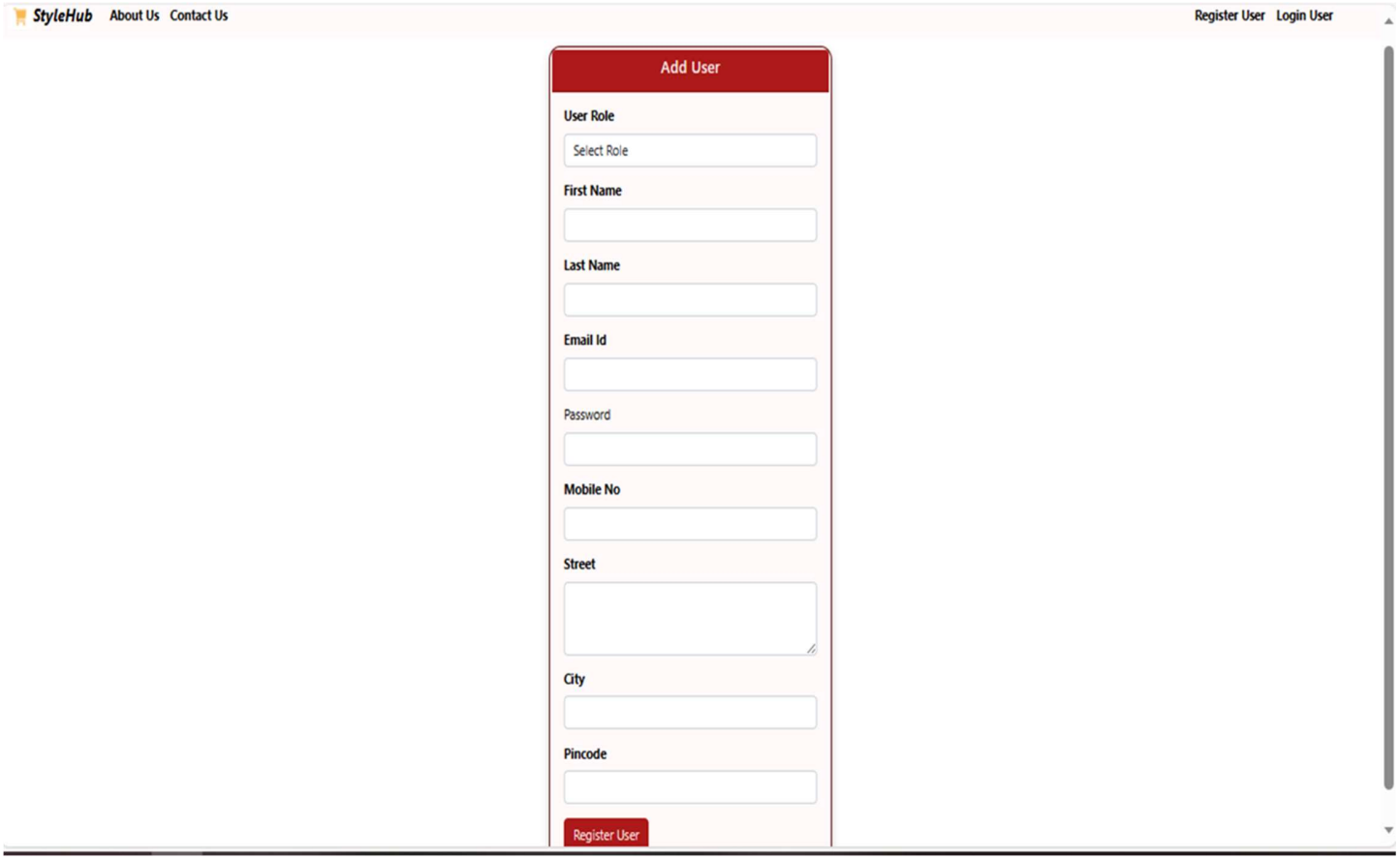
Admin Functionalities



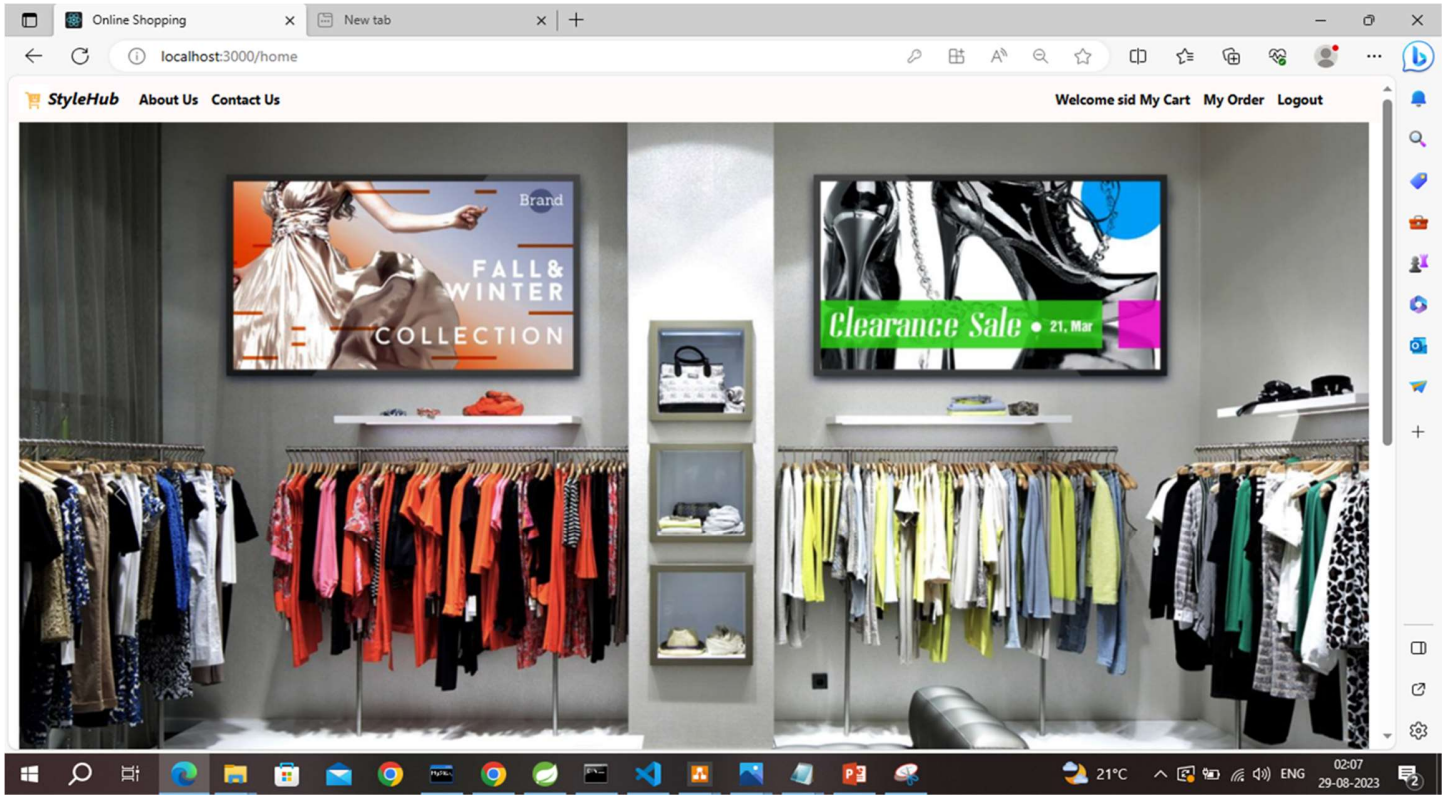
Add Features



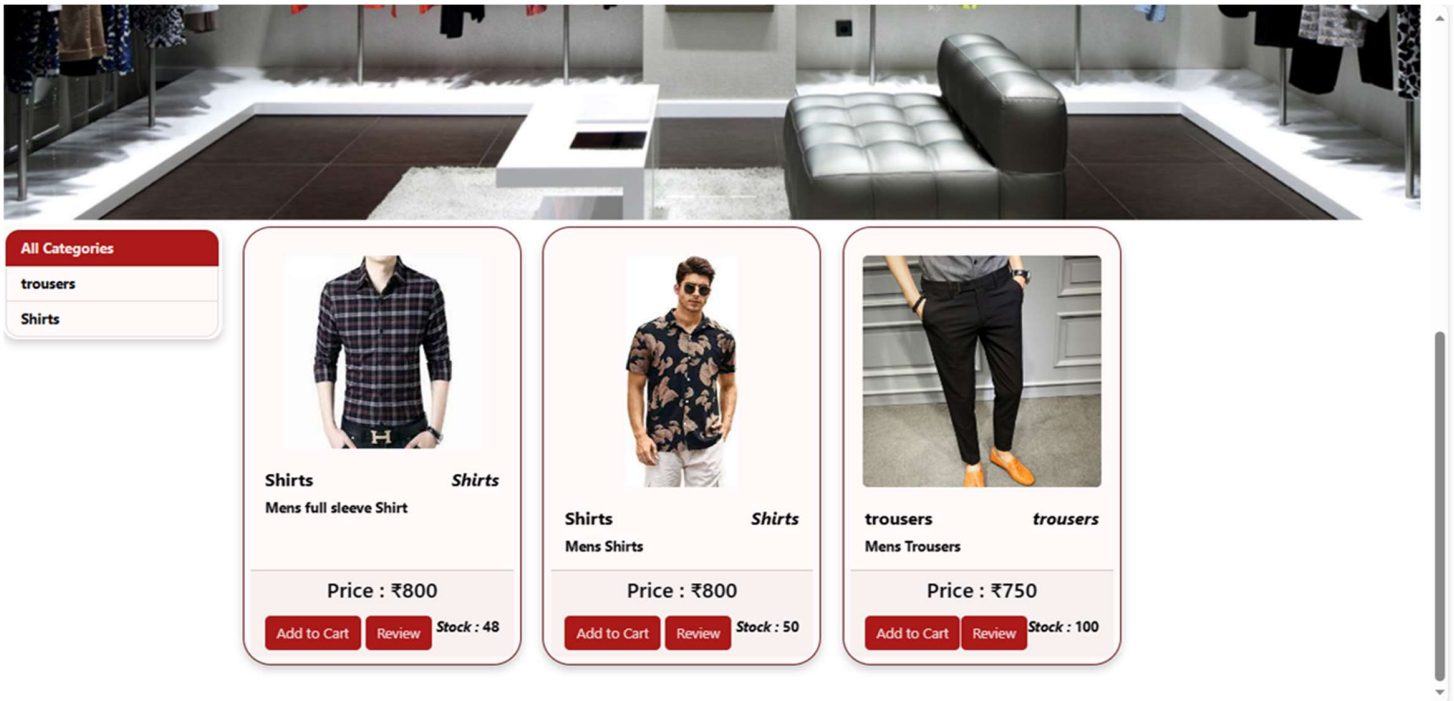
Signup



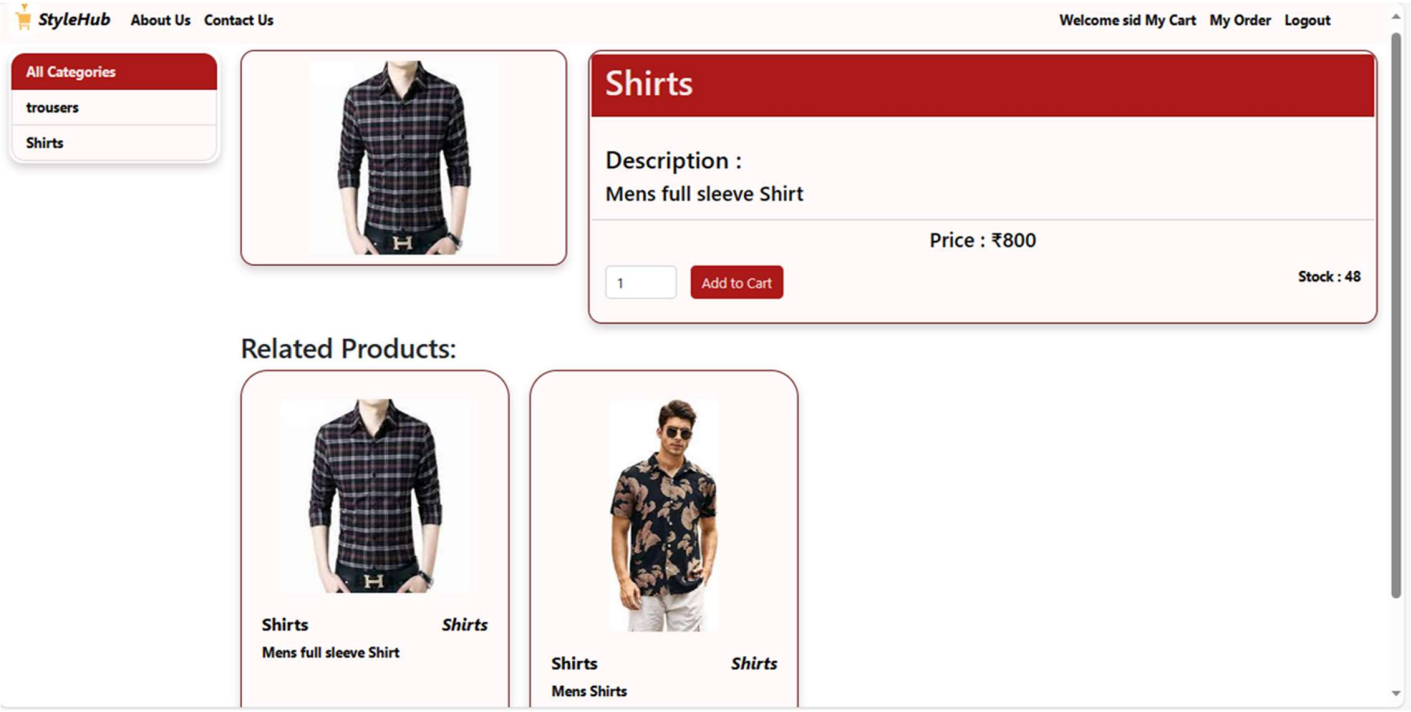
Sign In Successful



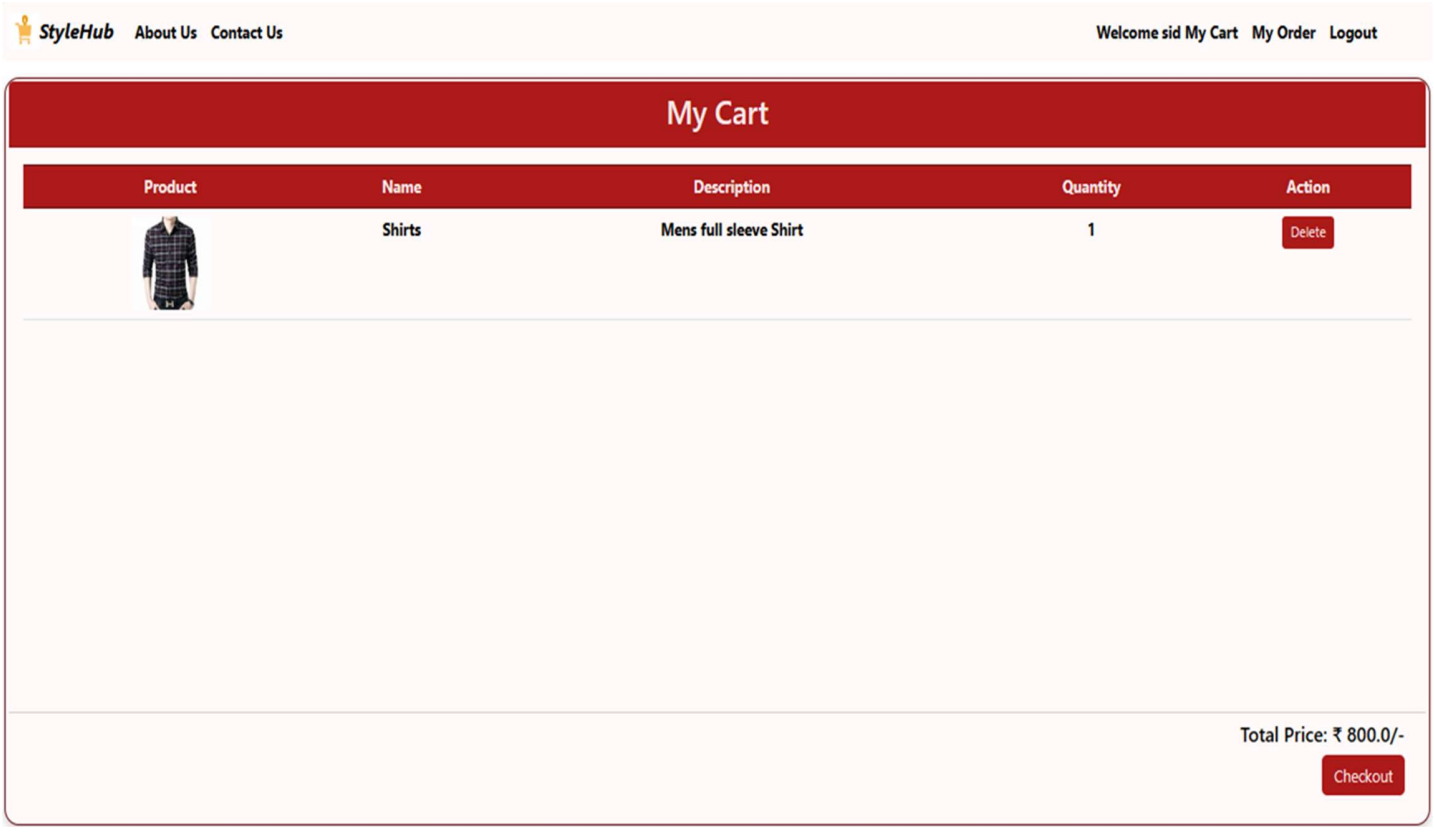
All Categories



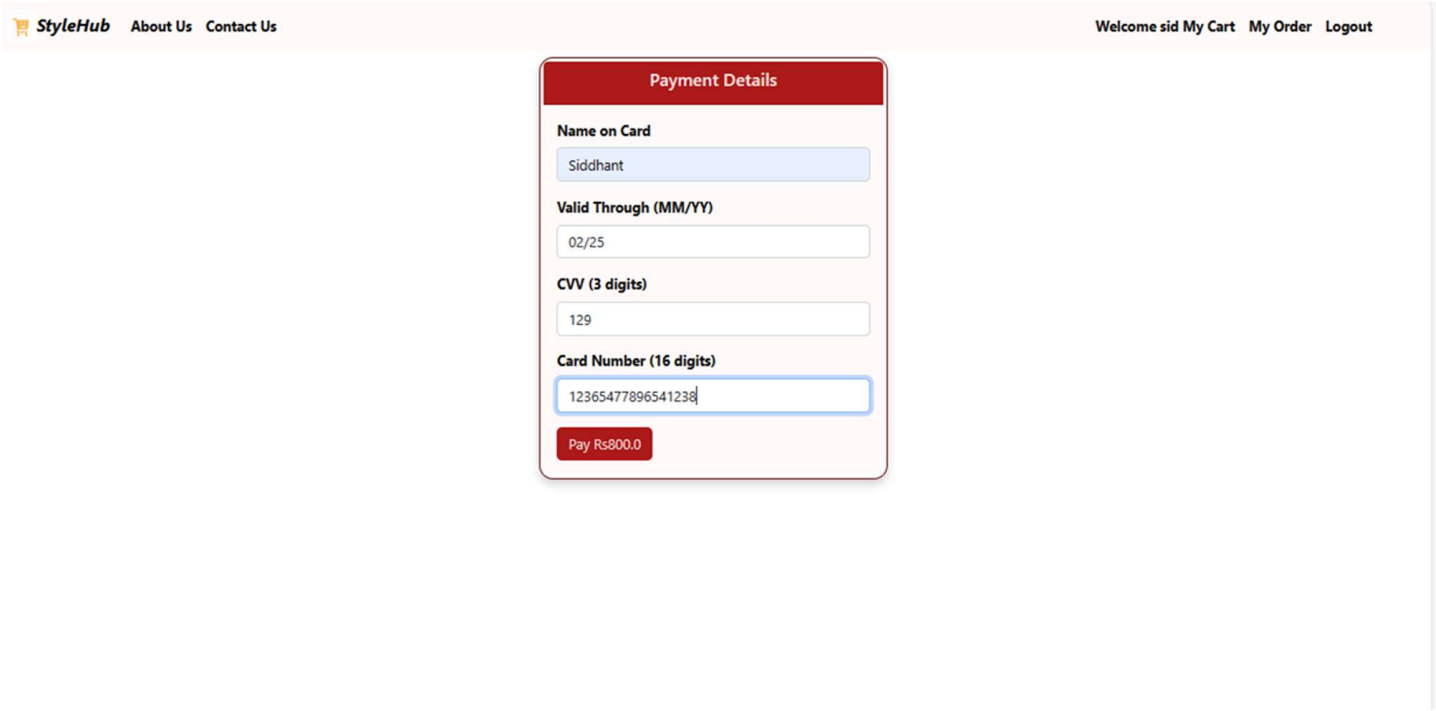
Add to Cart



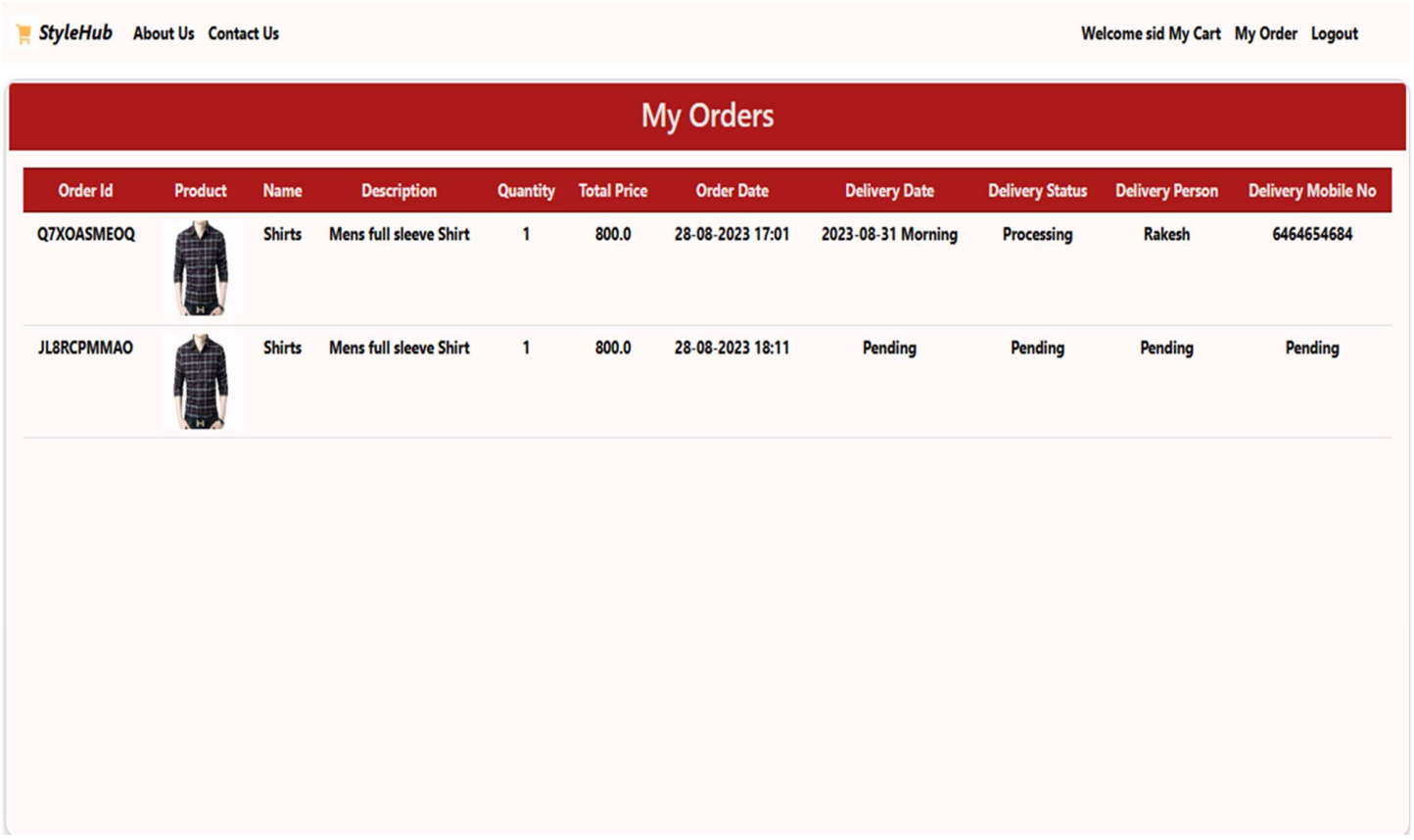
My Cart



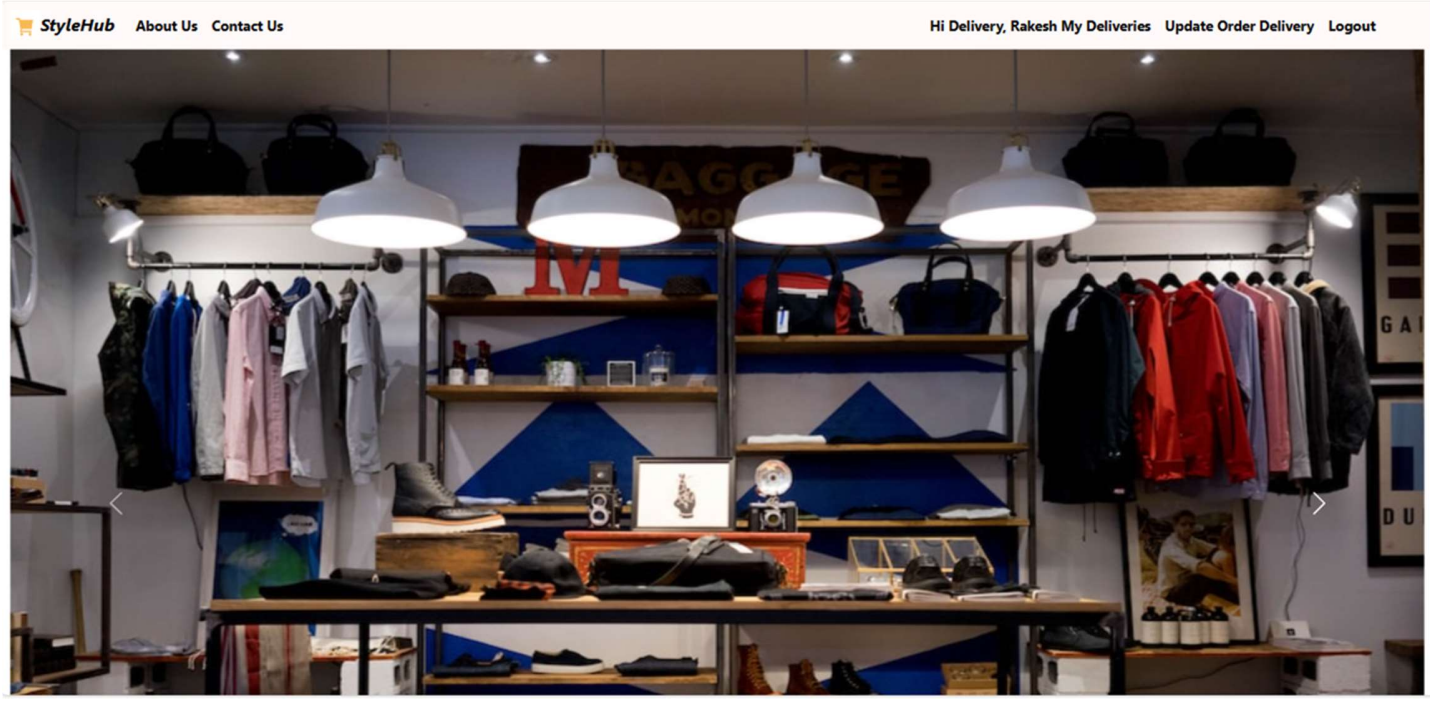
Payment Details



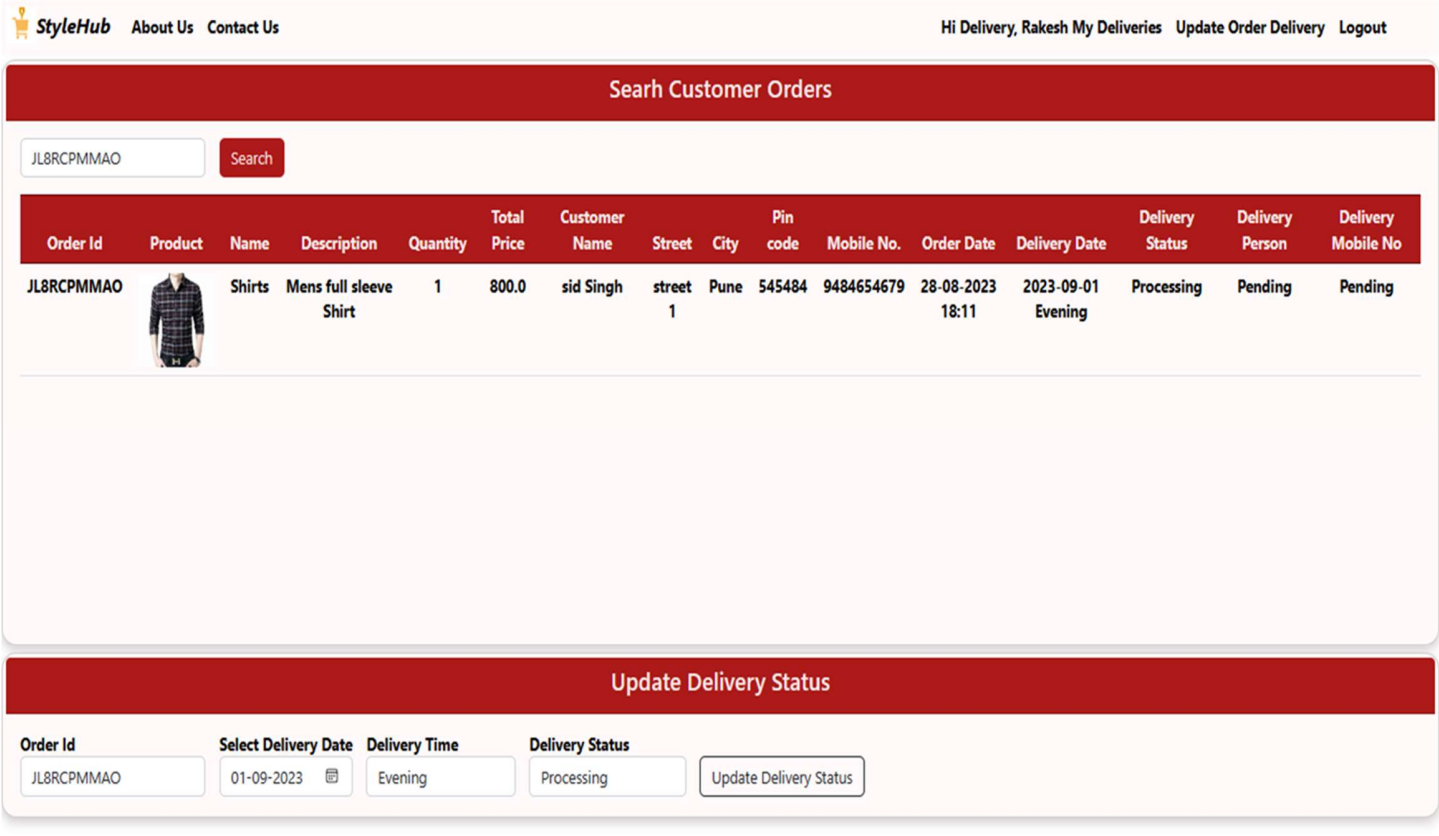
My Order



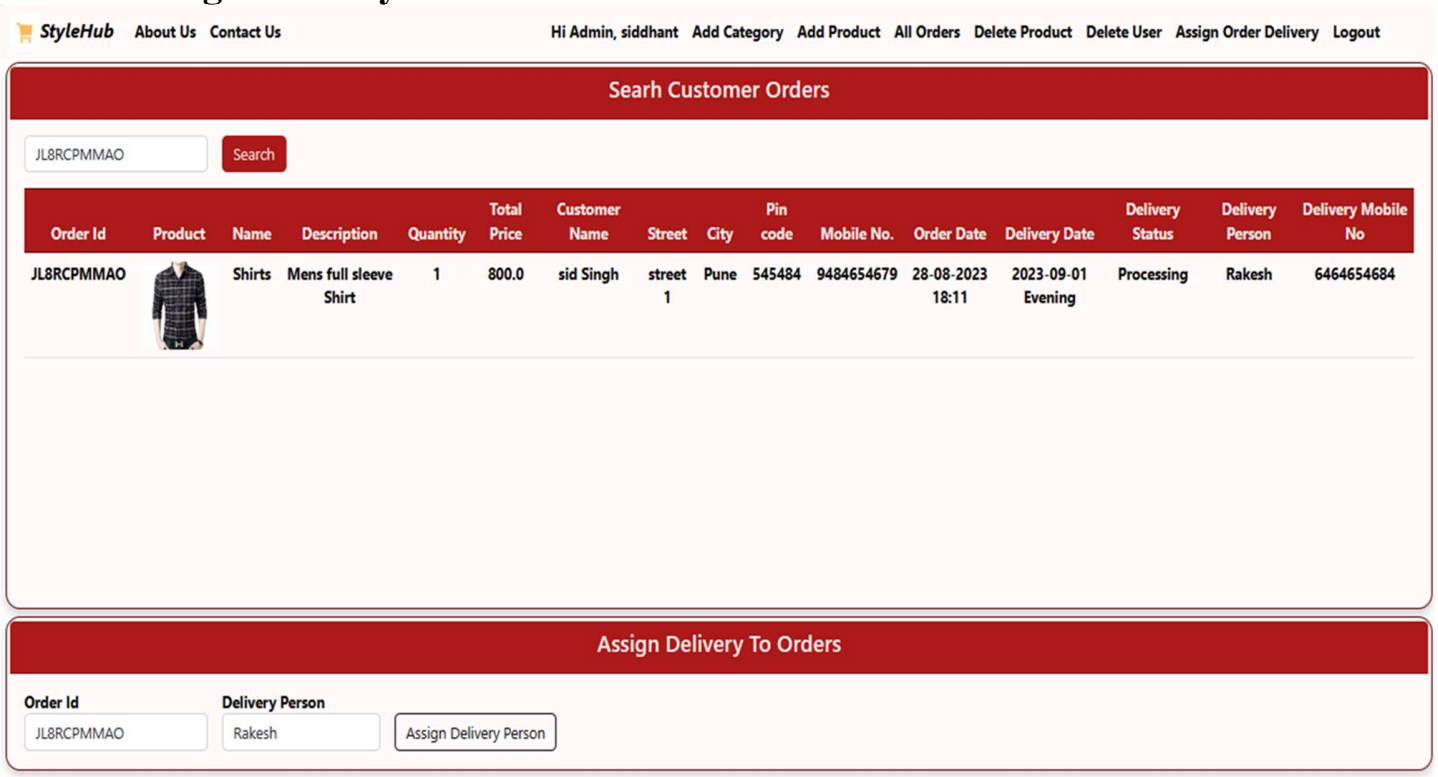
Login as a delivery Person



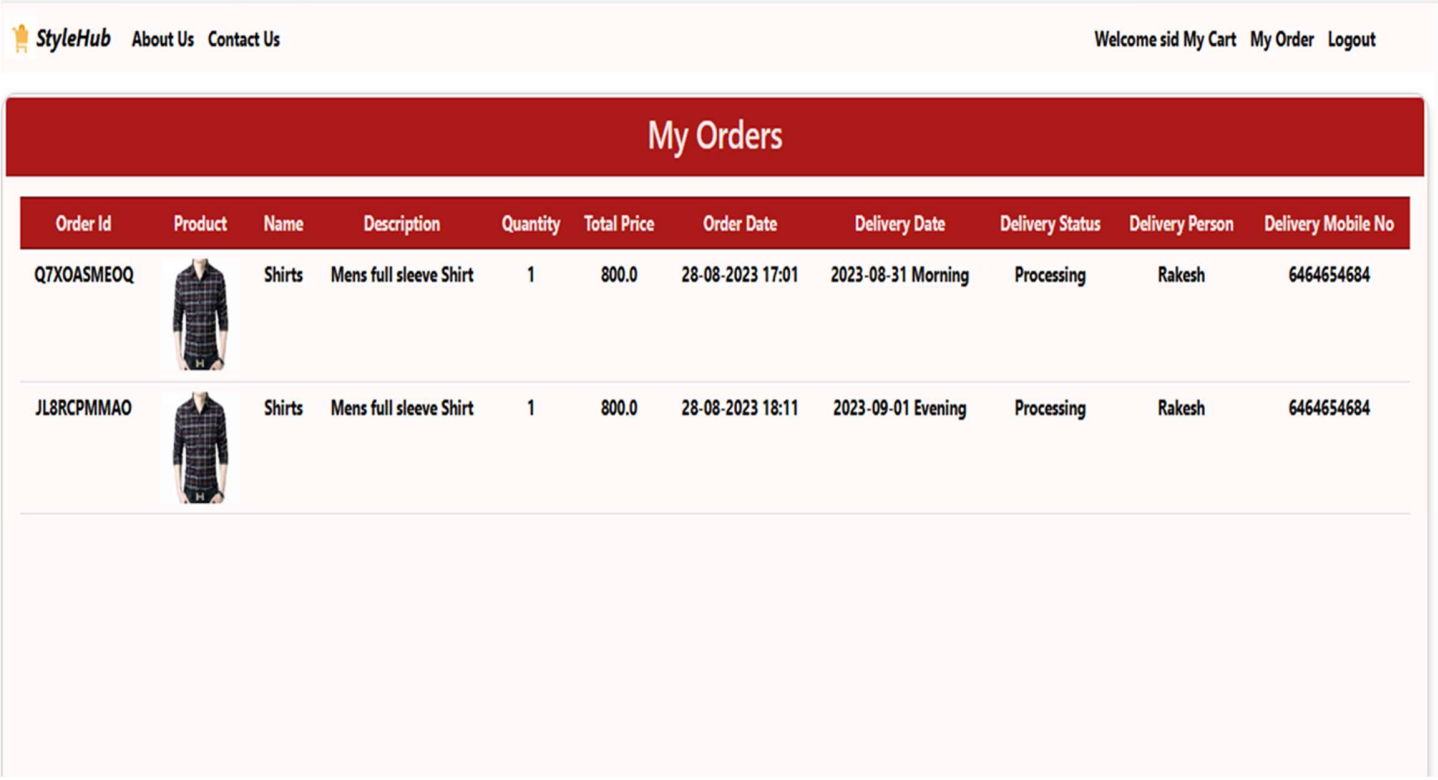
Search Customer Order



Assign delivery to Order



Tracking My Orders



CONCLUSION

The project entitled Online shopping application was completed successfully.

The system has been developed with much care and free of errors and at the same time it is efficient and less time consuming. The purpose of this project was to develop a web application and an android application for purchasing items from a shop.

This project helped us in gaining valuable information and practical knowledge on several topics like designing web pages using React.js, usage of responsive templates, designing of android applications, and management of databases using MySQL. The entire system is secured. Also, the project helped us understand about the development phases of a project and software development life cycle. We learned how to test different features of a project.

This project has given us great satisfaction in having designed an application which can be implemented to any nearby shops or branded shops selling various kinds of products by simple modifications.

There is a scope for further development in our project to a great extent. A number of features can be added to this system in future like providing payment gateway for maintenance payment and maintain their records in dashboard. Another feature we wished to implement was displaying notice board without login to system with the first page of “know about society”. Also wish to add each member profile with images store in data as well as flat owners with their family members information in the system. These features could have implemented unless the time did not limit us.

REFERENCES

[1] JavaScript Enlightenment, Cody Lindley-First Edition, based on JavaScript 1.5, ECMA-262, Edition

[2] Mc Graw Hill's, Java: The complete reference 7thEdition, Herbert Scheldt

[3] Complete CSS Guide, Maxine Sherrin and John Allsopp-O'ReillyMedia; September 2012

ONLINE REFERENCE

[1] www.Google.com

[2] www.w3school.com

[3] www.javatpoint.com

[4] www.stackoverflow.com

[5] www.reactbootstrap.com

[6] www.doc.spring.io/spring-framework.com

[7] www.geekforgeeks.com

[8] www.microsoftdoc.com

[9] Learn Code with Durgesh

[10] Learn Programming Yourself