**Database Project Report: Clothing Store**

|  |  |  |
| --- | --- | --- |
| Group Member Roll Number | Group Member Section | Group Member Name |
| 21K-3225 | BCS-5K | Usman Rasheed |
| 21K-4924 | BCS-5K | Muneeb Ali |

# Proposal:

The project aims to replicate the experience of browsing and shopping on a clothing store website, like Outfitters or Khaadi. Users can create an account, log in, select clothing categories, choose specific items, indicate preferences such as size, color, and quantity, add chosen items to their cart, input shipping details for delivery, and ultimately view a receipt summarizing their selections.

The goal is to simulate a seamless shopping experience, mirroring the process individuals undergo when purchasing clothing online. This involves user registration, personalized item selection, cart management, and checkout with shipping information, ending with a receipt being generated.  
  
  
**The functions that the project would include would be:**

A **Registration page** for the user

A **Login page** for the user

A Search Bar to search for clothes by their name

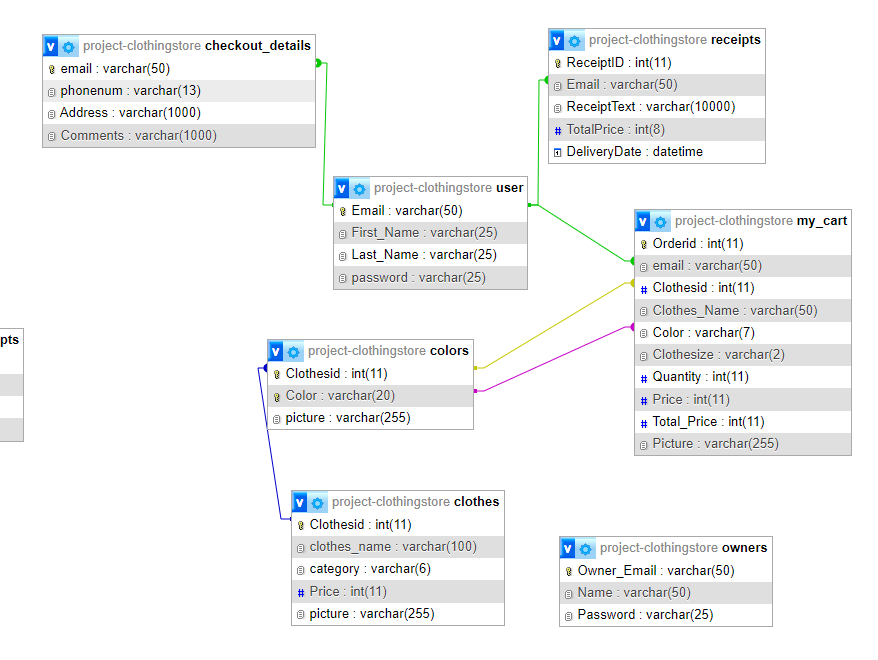
* A **Homepage** to view all other pages
* A page for each **Category** available (Men, Women, Kids, Winter) that Displays the clothes
* Clothes should come in Different **Color Variations**
* A **Product Detail Page**, where you select the details of the clothe you decide to buy
* A **Cart** option where you can view what you’ve selected, You should be able to **Delete from Cart and Proceed to Checkout**
* A **Checkout Form** for inputting your user details
* A **Receipt** option to view your order details
* An **Owner Registration, Login, and Homepage.**
* Owner can **View User Detail, Receipt Details And Clothing Details**
* Owner can **Insert Clothes**
* Owner Can **Delete Clothes**

**Technologies Used:** HTML (& CSS), JavaScript, PHP, MySQL

Normalization:   
  
In our Project Normalization was required towards a single table: The Clothes Table, Said table was broken into two different tables Clothes, and Colors. The Diagram below shows the normalization for the Clothes Table.  
  
  
  


FD1 = (ClothesID -> Name, Category, Price, Picture)

FD2 = (ClothesID, Color -> Picture)

ER Diagram:  
  
  
User to Checkout\_Details: 1 to 1

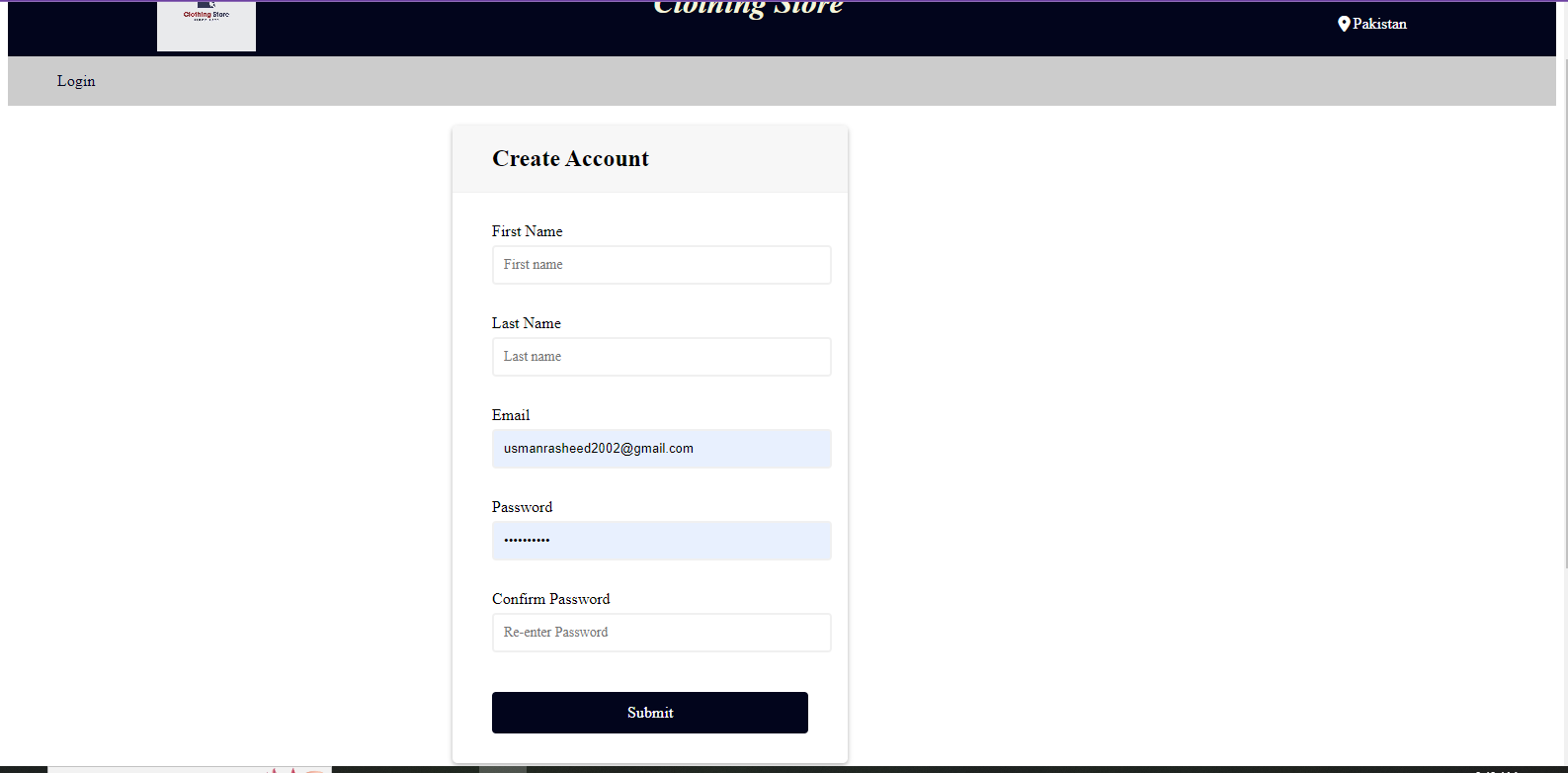
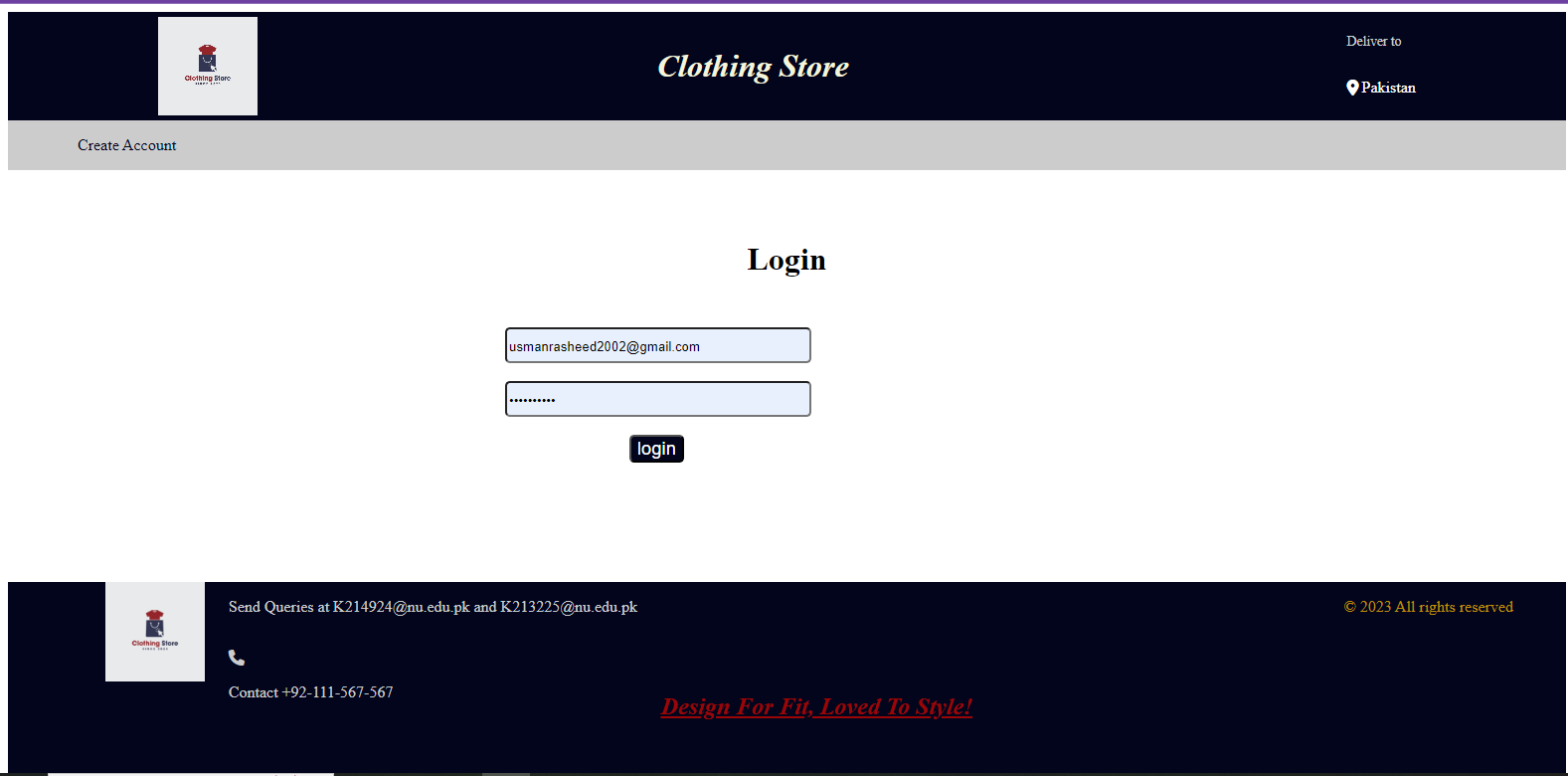
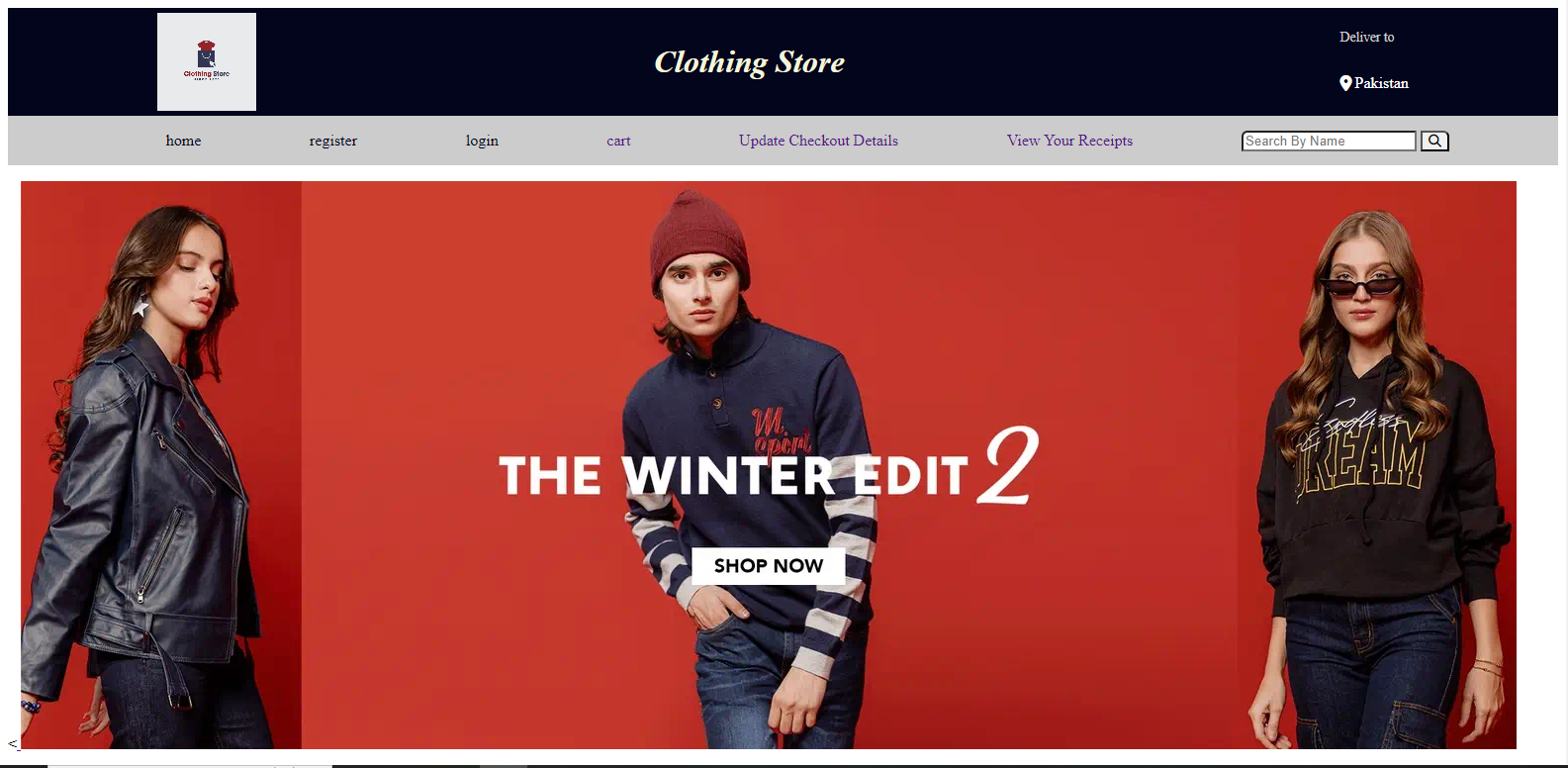
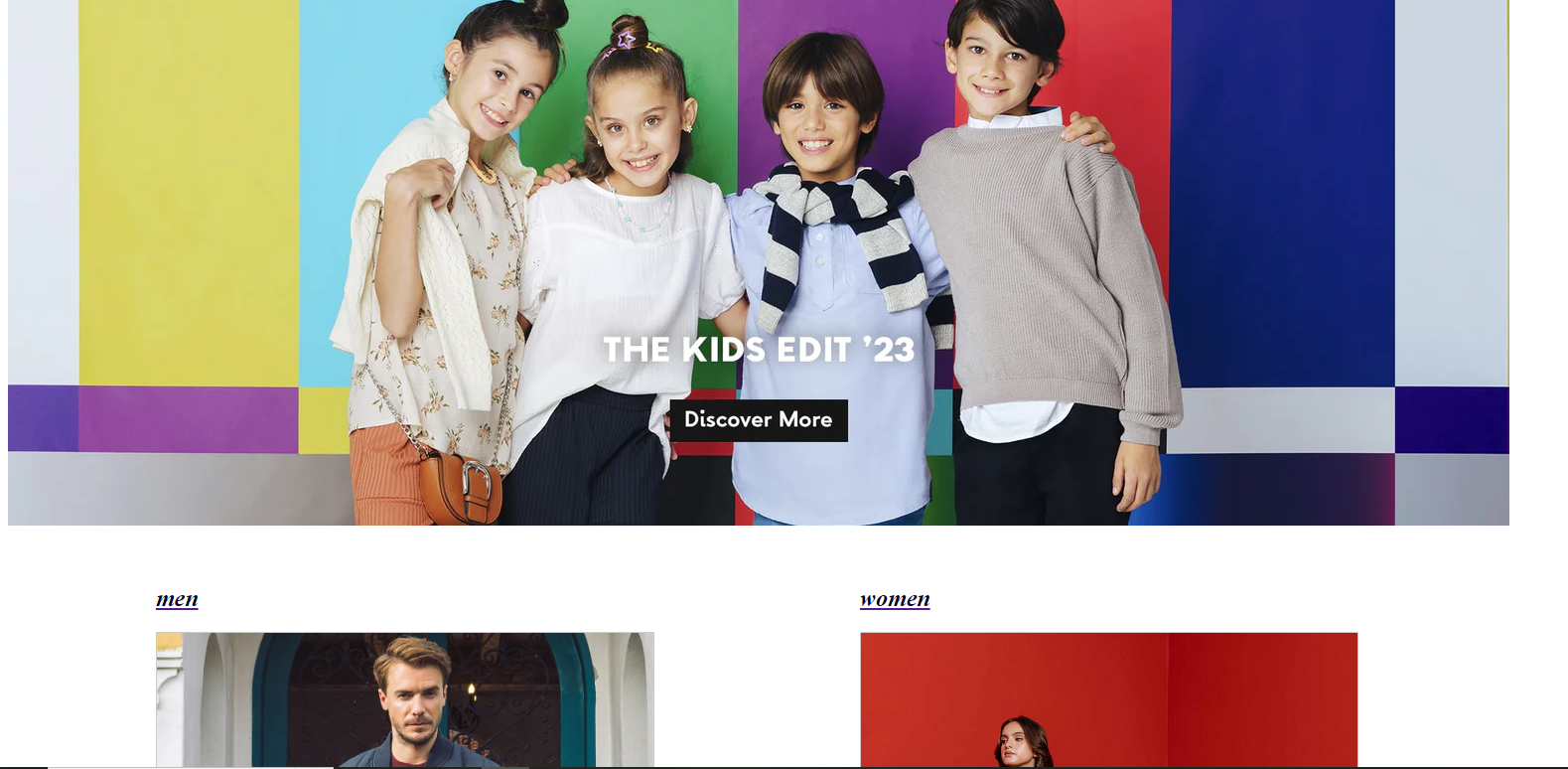
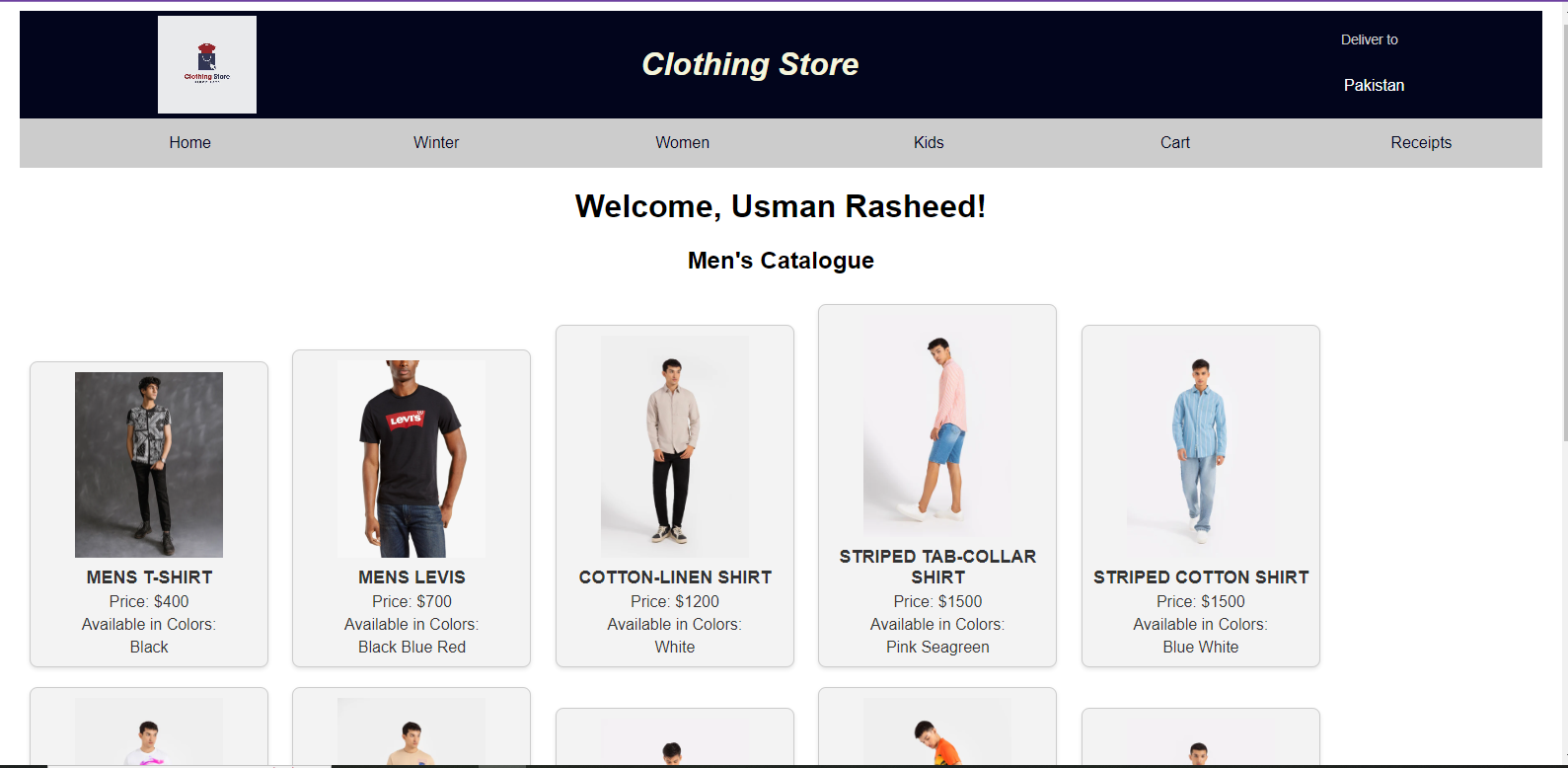
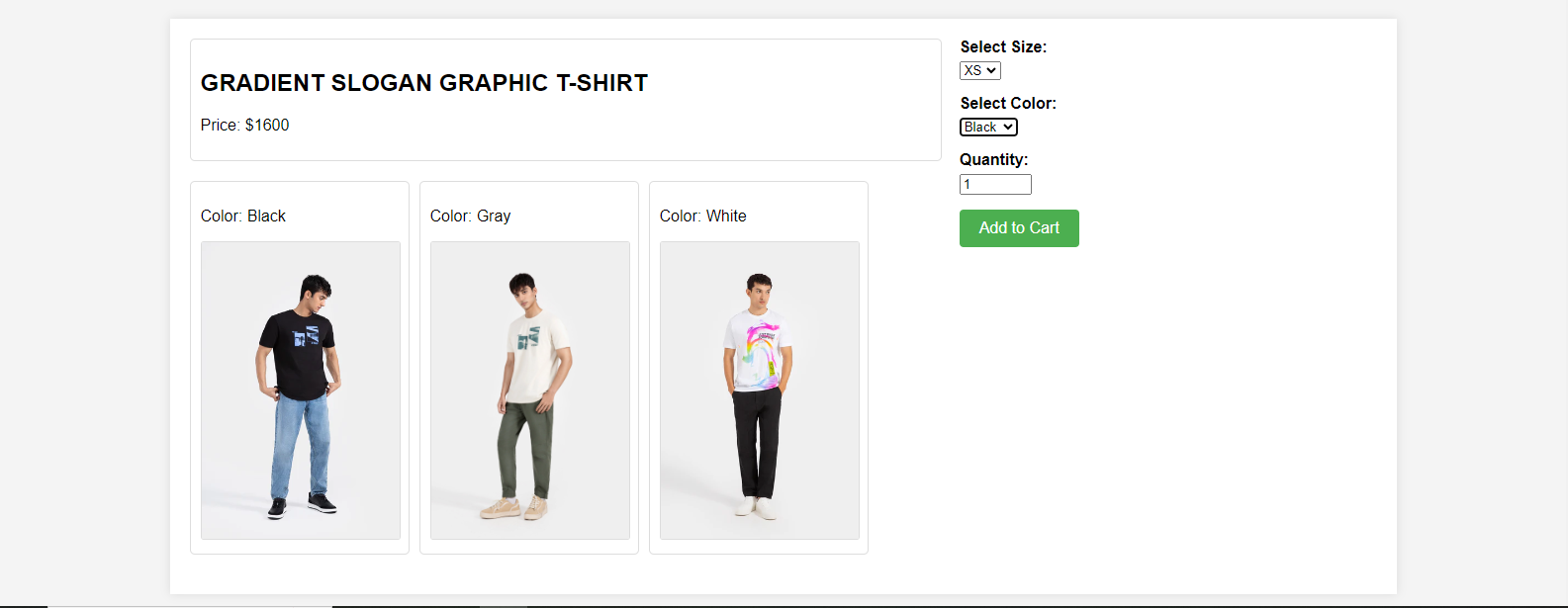
User to My\_Cart: 1 to N

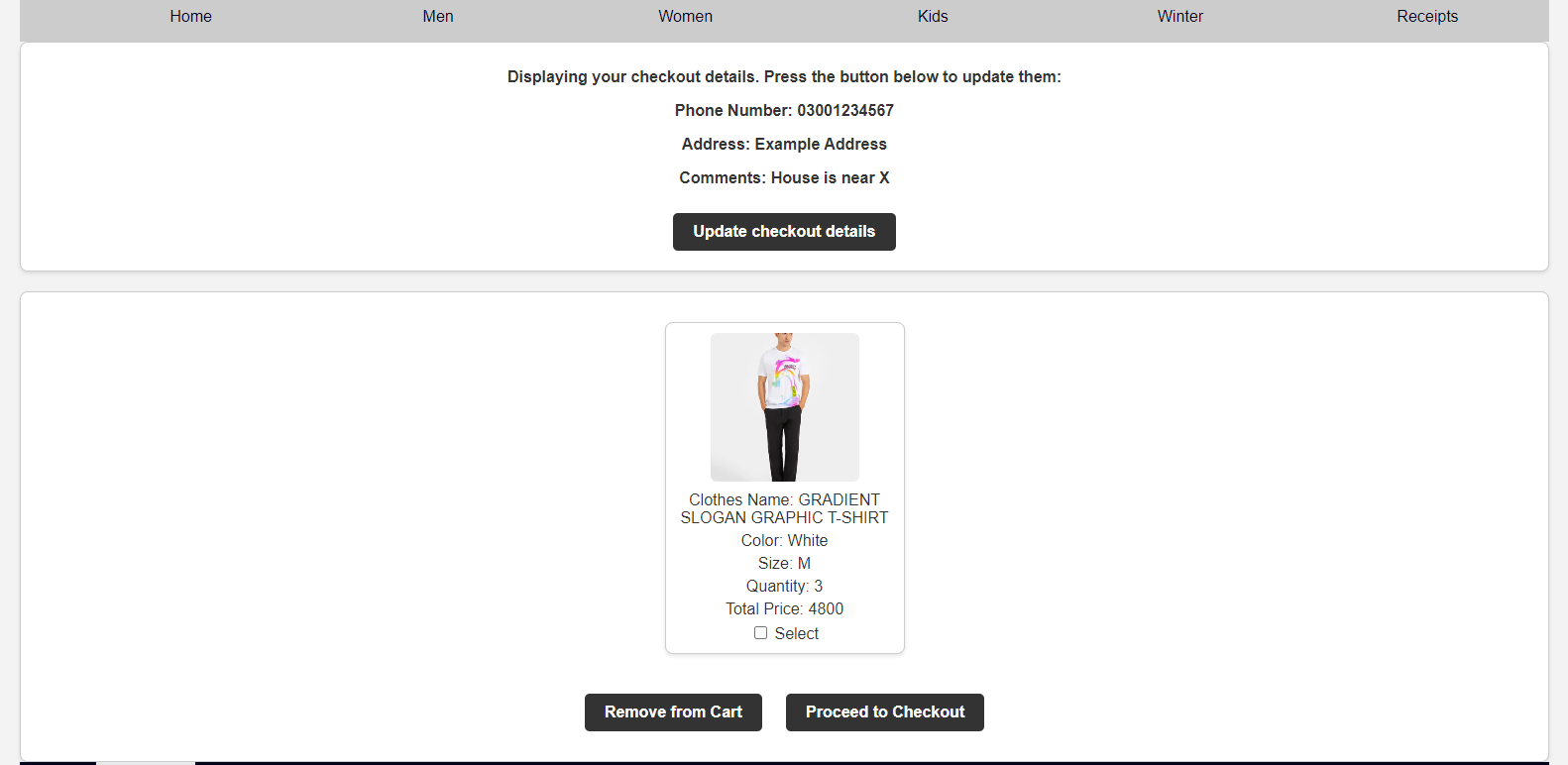
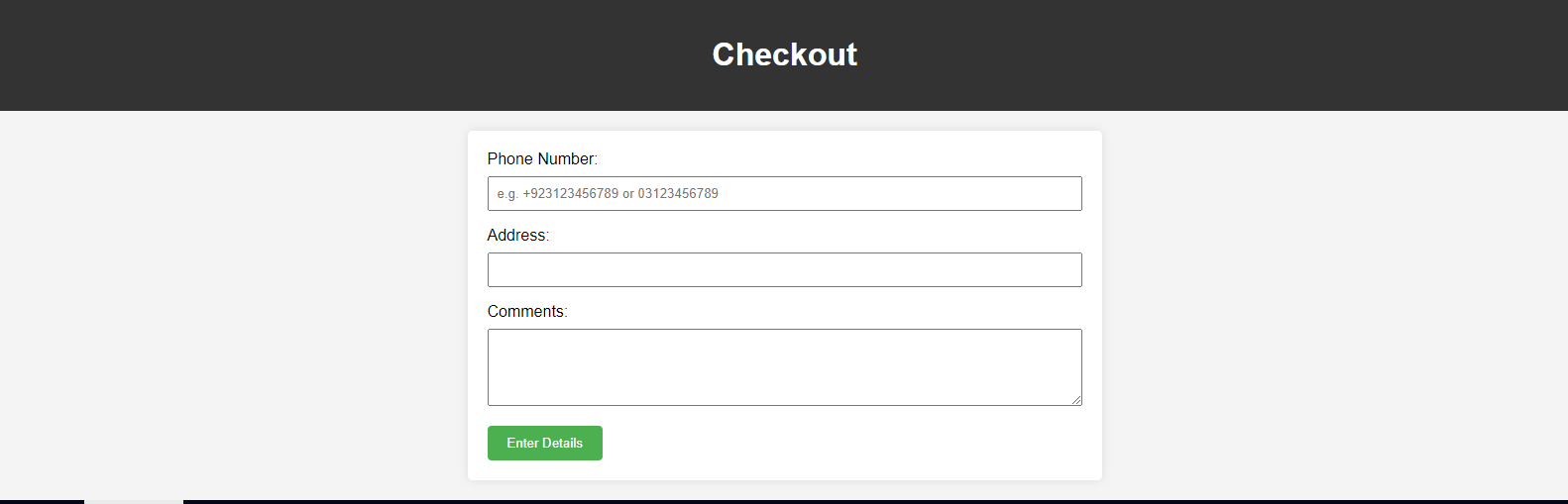
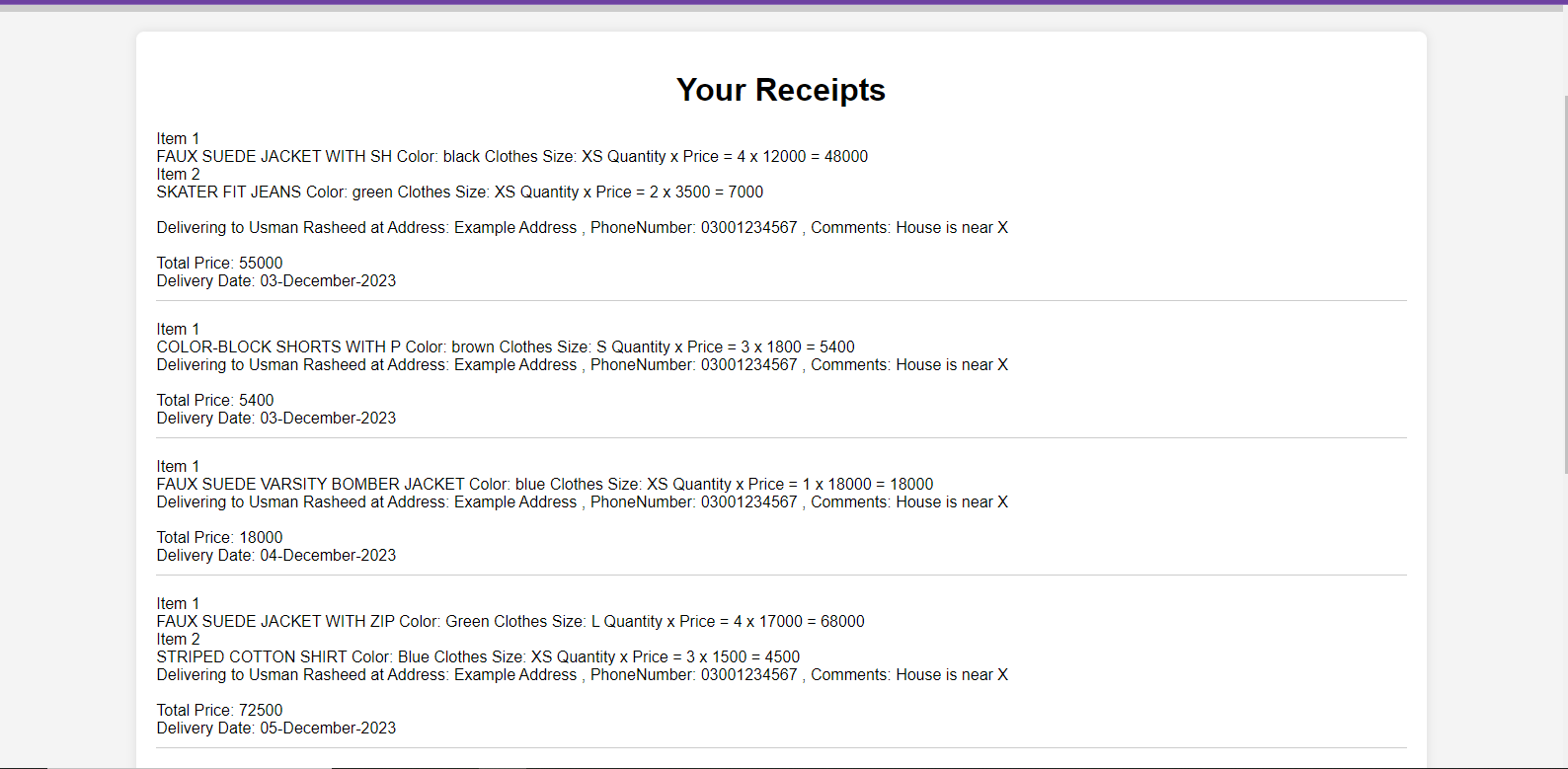
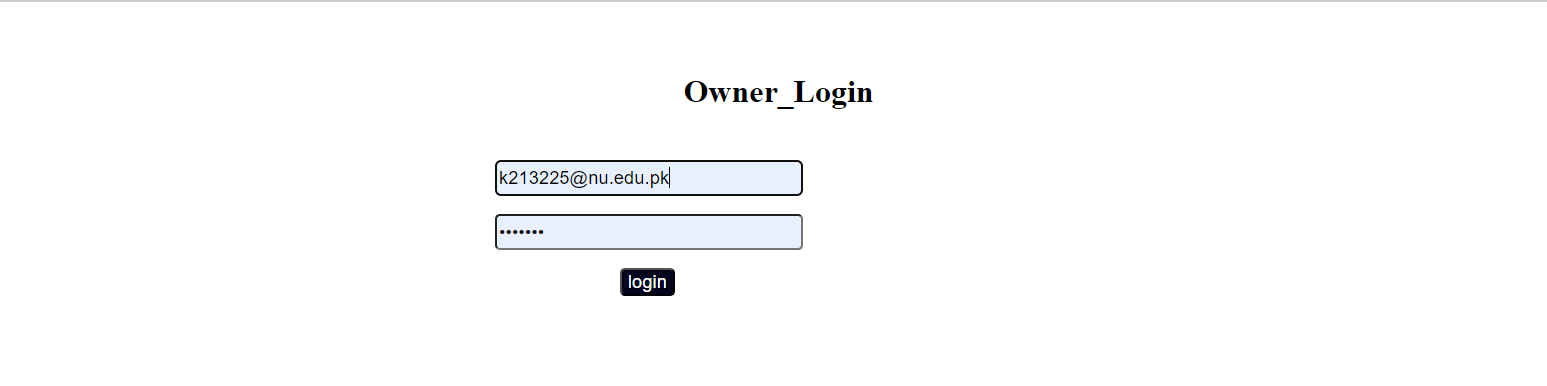
User to Receipts: 1 to N

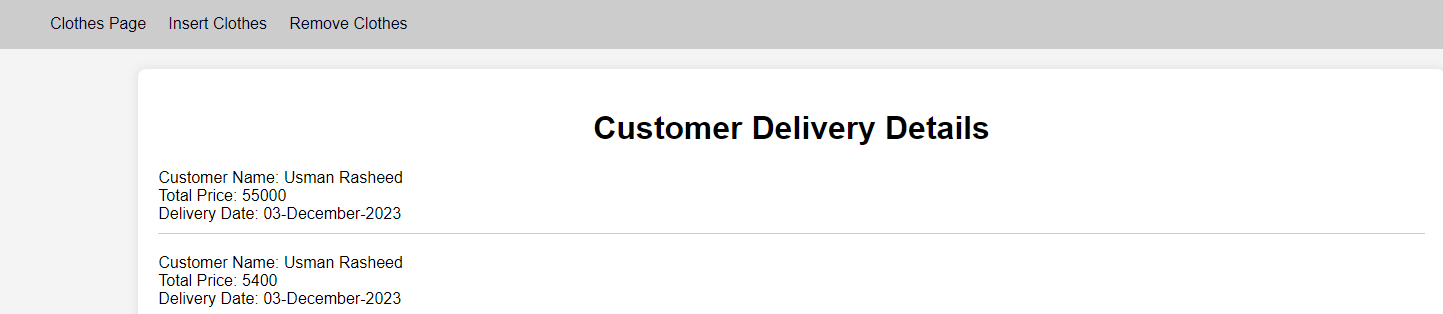
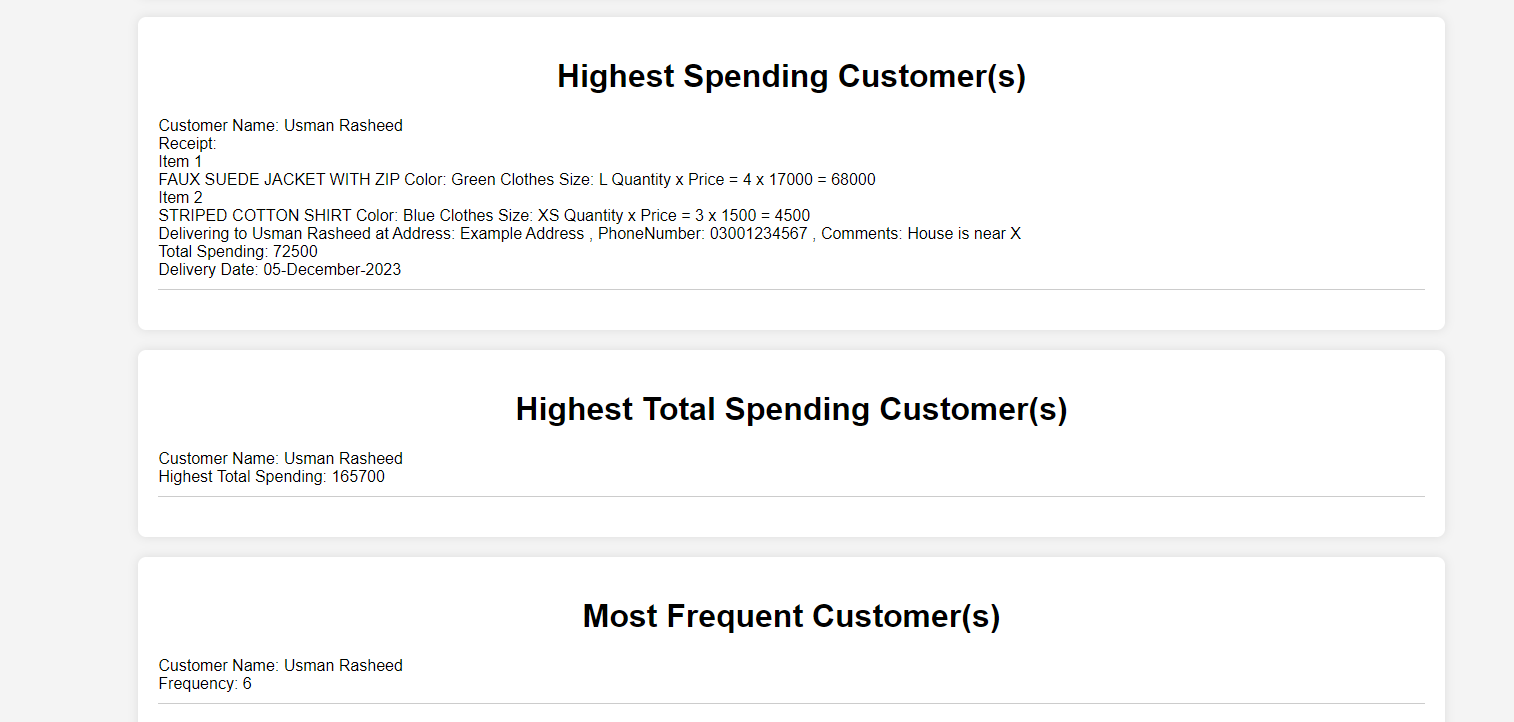
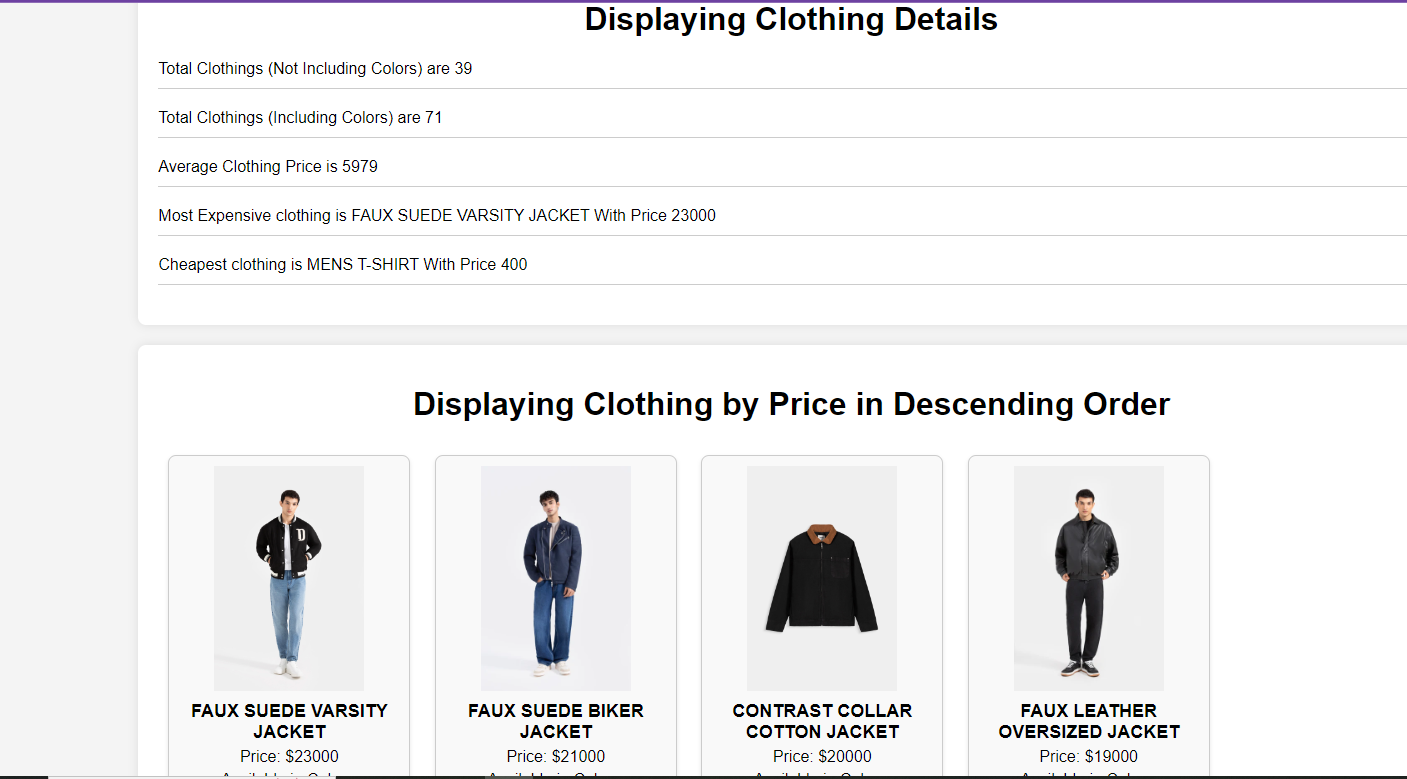
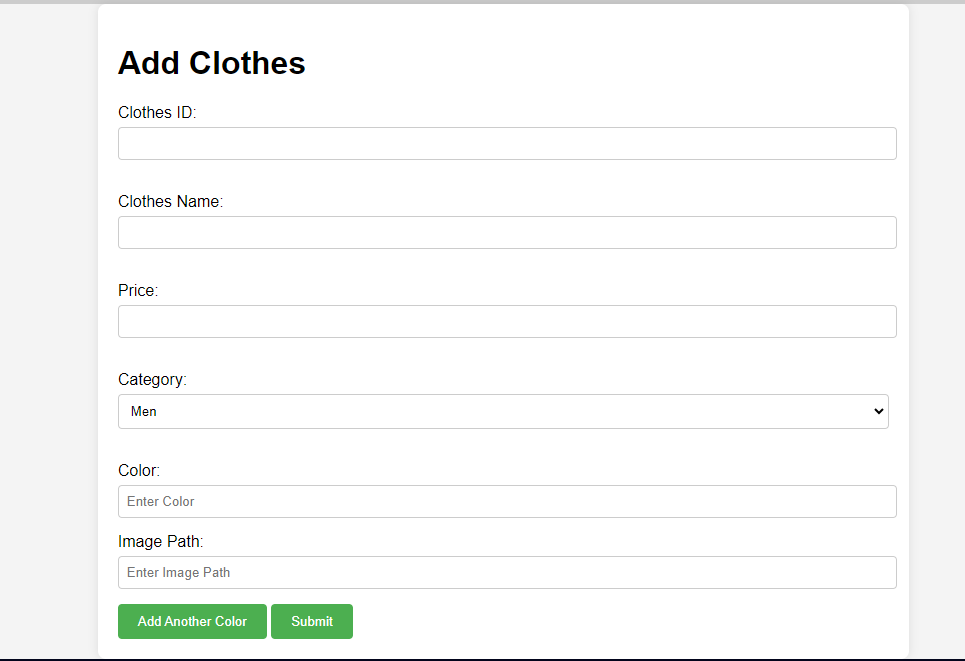
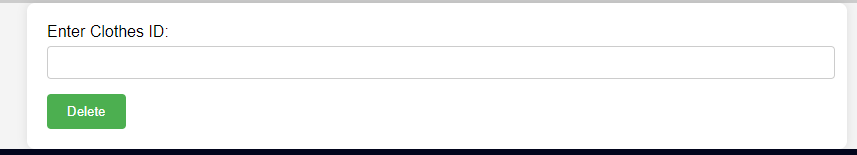
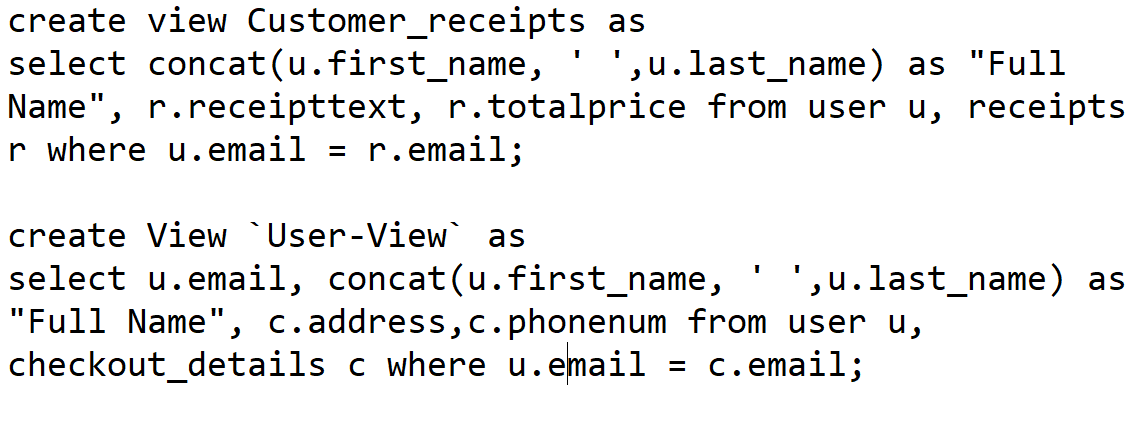
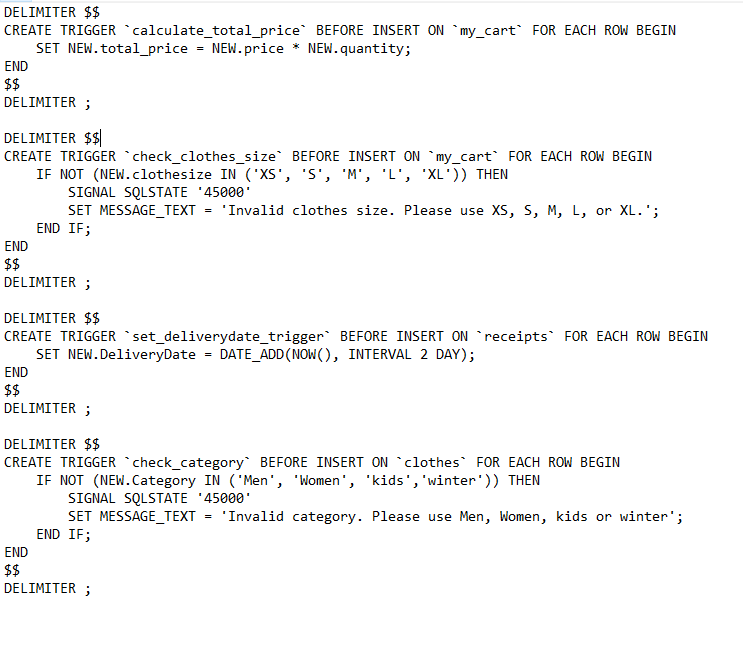
Clothes to Colors: 1 to N

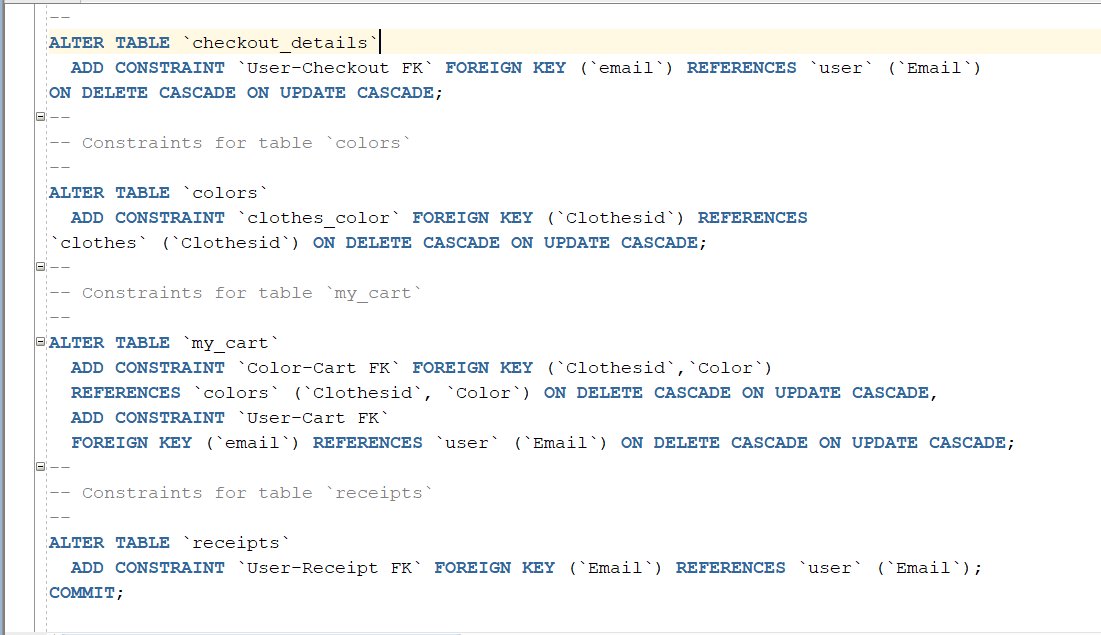
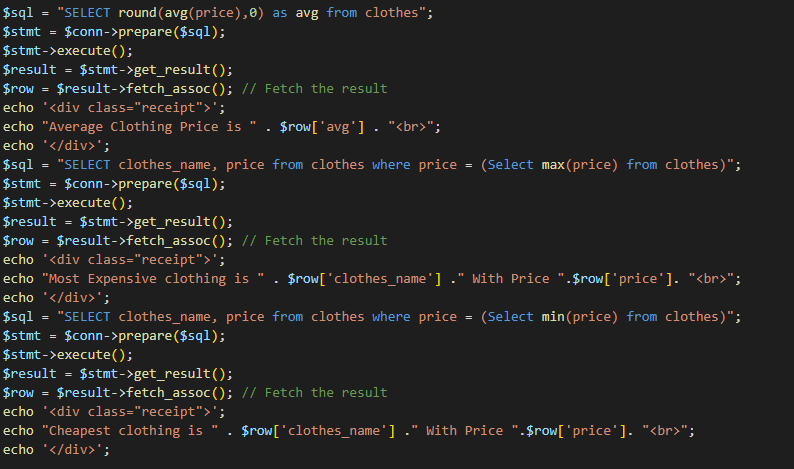
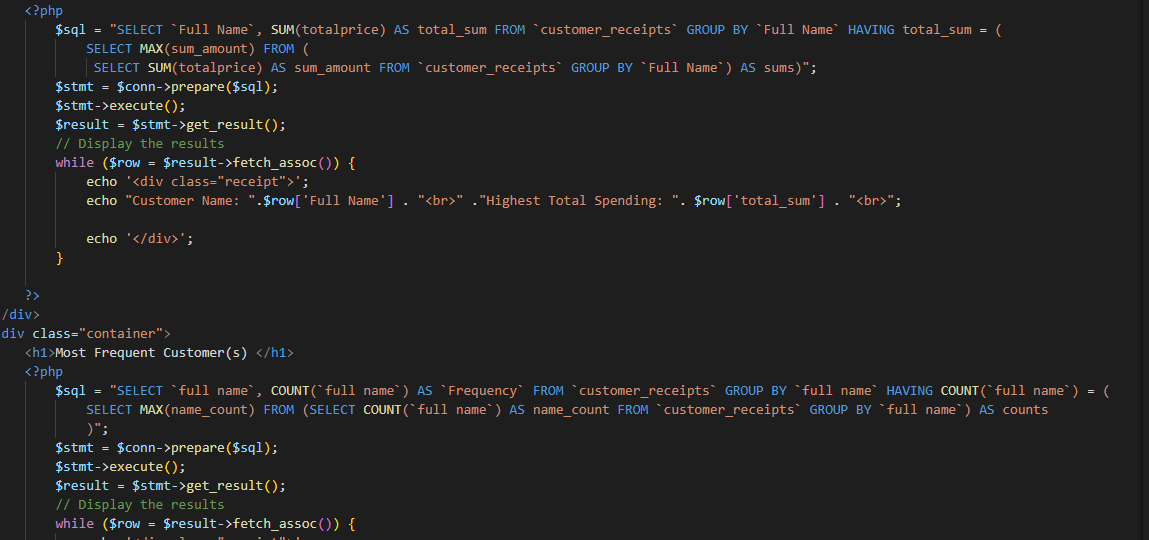
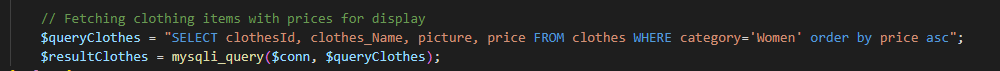
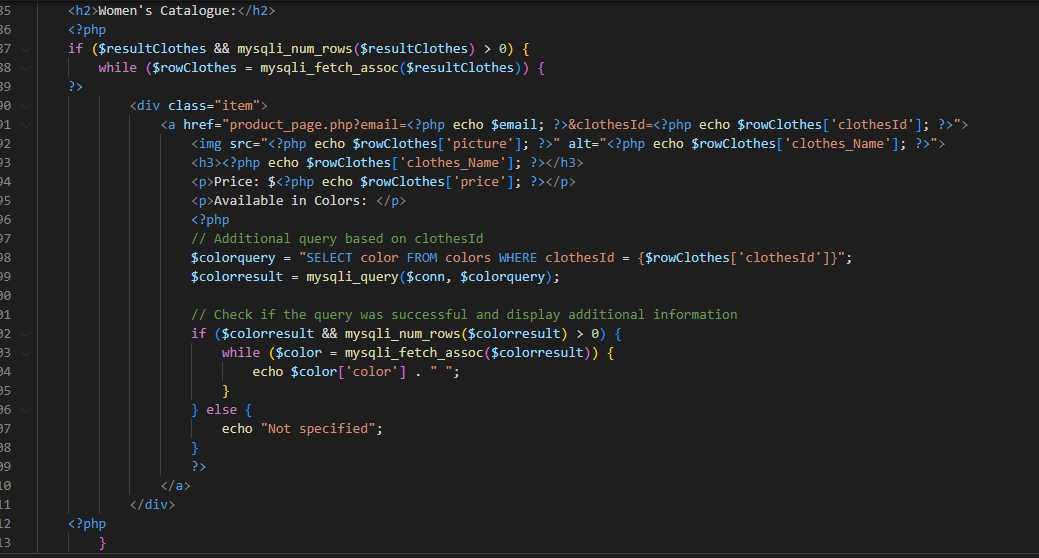
Colors to My\_Cart: N to M

Entities in the Project: User, Clothes, Cart, Receipts, Owner

Tables in the Project: User, Receipts, Checkout\_details, Clothes, Colors, My\_cart, owners.  
  
Website Flow:  
  
  
  
The user registers first  
  
  
Then they login  
  
  
  
  
The user can then go to any page they prefer, or search clothing by name, let’s go to men for this example:  
  
  
The User name is displayed at the top and then clothing’s are displayed with a picture, the colors they are available in and the price in ascending order  
  
  
The User will then be sent to a product page where they can view product details, along with the clothing images. The user can select the color, the size and the quantity of their item.

  
  
The user can now view all their cart items, and their total price. Note that the total price is price x quantity, User checkout details are also displayed, if the user hadn’t entered checkout details they would be sent to a checkout page instead to the cart. User can update their checkout details whenever they want before proceeding to checkout.  
Remove from cart will remove selected element from cart. While proceed to checkout will create a receipt.  
  
  
  
This is the checkout page if the user proceeds to update checkout details  
  
  
  
Finally a receipt is generated. Our Receipt at the bottom is the order we just checked out, it will be delivered 2 days from now (6 December). Since this is a simulation all delivery dates are 2 days after the current time.  
  
  
For the owner side the owner must first login.

  
  
The owner can then view customer and receipt details, highest spending, highest total spending, most frequent, receipts yet to be delivered, receipts that are delivered.  
  
  
They can then go to the Clothes page where they can view clothes details and then the picture, colors, name and price in descending order  
  
  
Owner can add clothes, with as many colors they like.  
  
  
Owner can delete clothe by providing the clothes id.  
  
  
  
  
  
Implementation Phase:  
  
**Use of Joins and Views:**  
  
  
  
The views are also using the built in function: concat();  
  
Use of Triggers:  
  
  
The trigger set\_deliverydate is also using the built in function: NOW()

**Use of Constraints:**  
  
  
  
  
**Other Concepts of database includes:**   
Use of Subqueries, Use of having function, use of group by and use of order by  
  
**Use of other built in functions used include:**Count, Sum, Average, Round, Maximum and Minimum  
  
Below are some examples (There are many more):  
  
  
  
Other concepts utilized include dynamically displaying in html through PHP (Here is an example, there are many more)  
  
  
  
  
  
**User Inputs (For User):** Registration input (placed in database), Login Information(Fetched from database), Select Category or Search (Display clothes and colors accordingly), Select Clothe (Display clothe with color, and color images accordingly), Add clothe to cart by inputting size, color and quantity (Store Clothes in cart database), Enter/Update Checkout details (Insert into or update checkout\_details table), View Cart (Display User Cart), Delete Cart Items (Deletes items from cart), Proceed to Checkout(Places cart items into receipts table, and deletes from cart), View Receipts (Display from receipts table for User).

**User Inputs (for Owner):** Input Login Details (Fetched from database), View user details, receipt details and various other details through built in functions (Display the two views created, also display the relevant function). Display Clothes and Colors, and other various details through built in functions (Display Clothes, their relevant colors, and the built in functions), Insert Clothes and Colors (Insert into clothes and colors the relevant data), Delete Clothes (Delete clothes and colors from the respected tables)