E-COMMERCE APPLICATION ON IBM CLOUD FOUNDRY

PHASE 4-DEVELOPMENT PART-2

Team Members:

Lathika D (211521104078) [Team Leader]

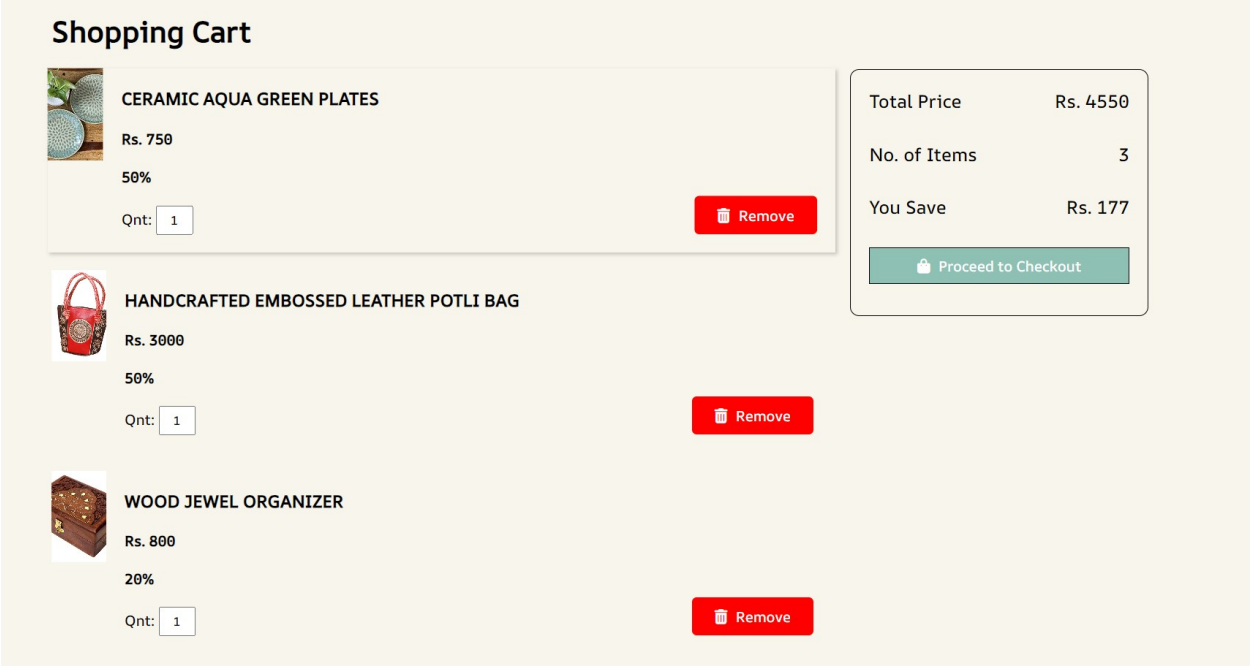
Jaya Varsha R(211521104059)

Jebil R Sharon(211521104061)

Pavithra N(211521104104)

Jayapriya T(211521104058)

CART IMPLEMENTATION



Static Data

This component will include somewhat static data that the customer needs only to retrieve while interacting with a shopping cart.

The data is stored in the following types of tables:

- product table
- discount table
- user table

Session Data

This is the most important component of the shopping cart database where all the live interactions (session details) are stored when the client is interacting with the shopping cart.

- shopping_session table
- cart_item table

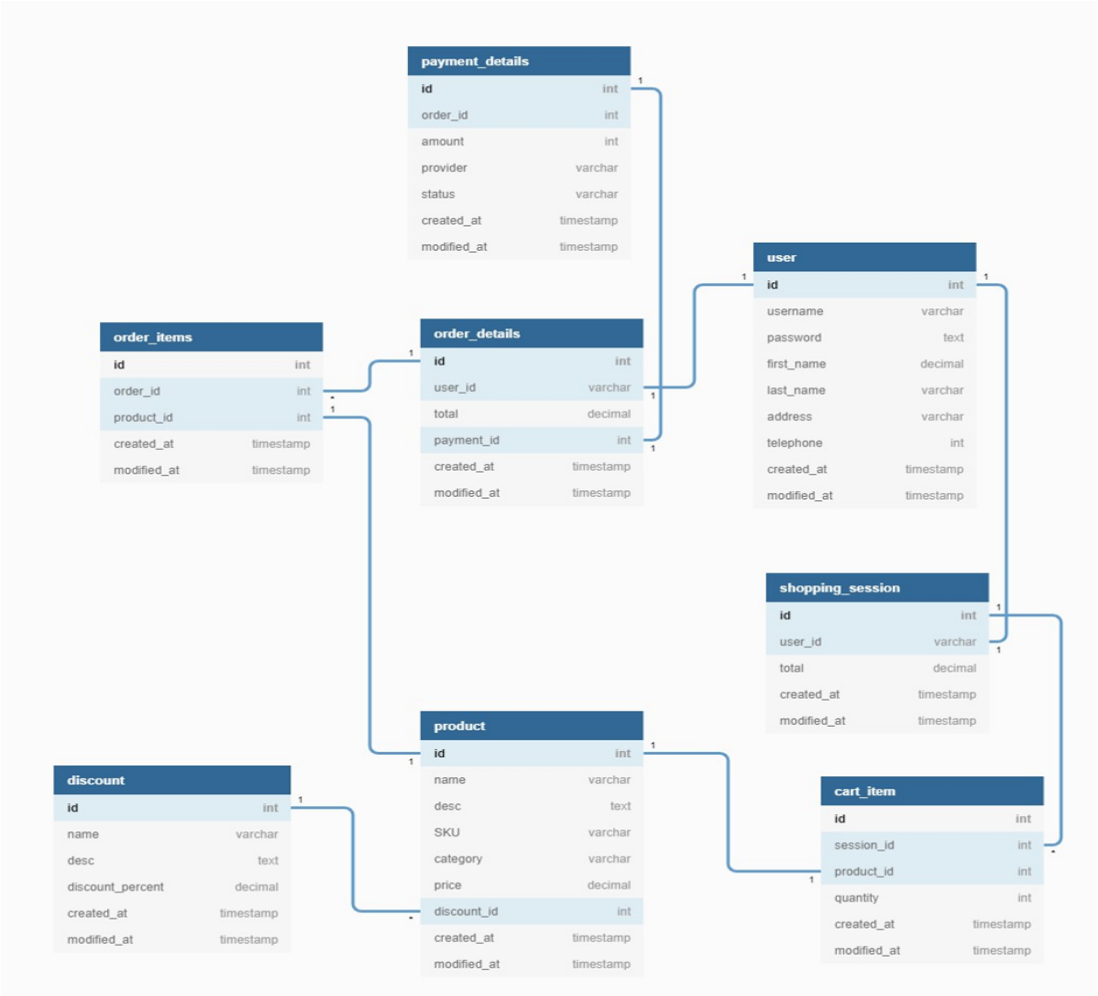
Processed Data

Once the customer completes a transaction, we need to permanently store the order information by moving the Session Data into permanent storage.

Additionally, we need to store the payment details.

- order_details table
- order_items table
- payment_details table

TABLE RELATIONSHIPS IN THE DATABASE:



FILE STRUCTURE FOR CART:

The shopping cart software example has the following file structure. The below list has the file names and their responsibility.

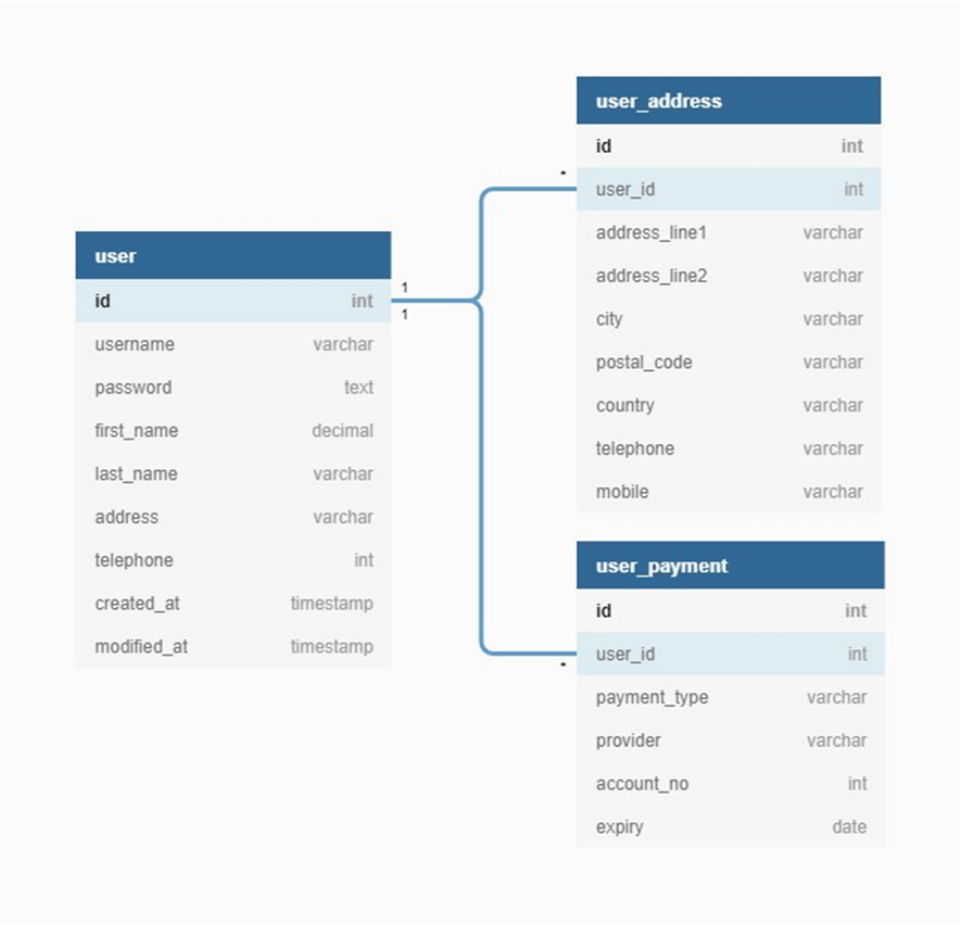
- dbcontroller.php – a generic database layer to help with DAO functions. It also manages the database connection.
- index.php – to display the product gallery for the shopping cart.
- style.css – to showcase products for the shopping cart. The styles are minimal and cross-browser compatible.
- tblproduct.sql – contains SQL script with the product table structure and the data.
- product-images – a folder that contains the product images. I have used these images to show the product gallery.

EXPANDING THE SCOPE OF THE DATABASE:

Shopping cart databases are only a single part of a vast e-commerce experience. This section will briefly explain how to extend the database to cover additional functionalities by introducing new tables and fields to the existing database.

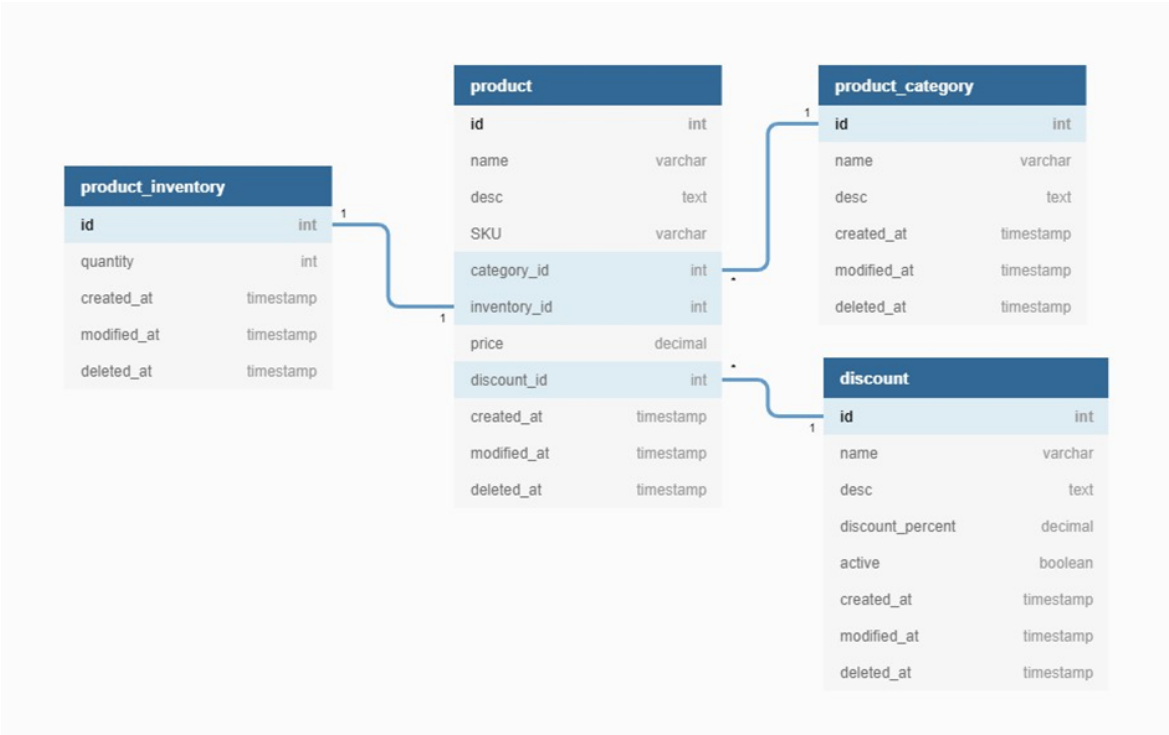
USER DETAILS:

A user table can be extended with other tables such as user_address and user_payment to store user preferences. Thus, it enables a smooth shopping experience by providing the stored details of the user for a faster checkout process.



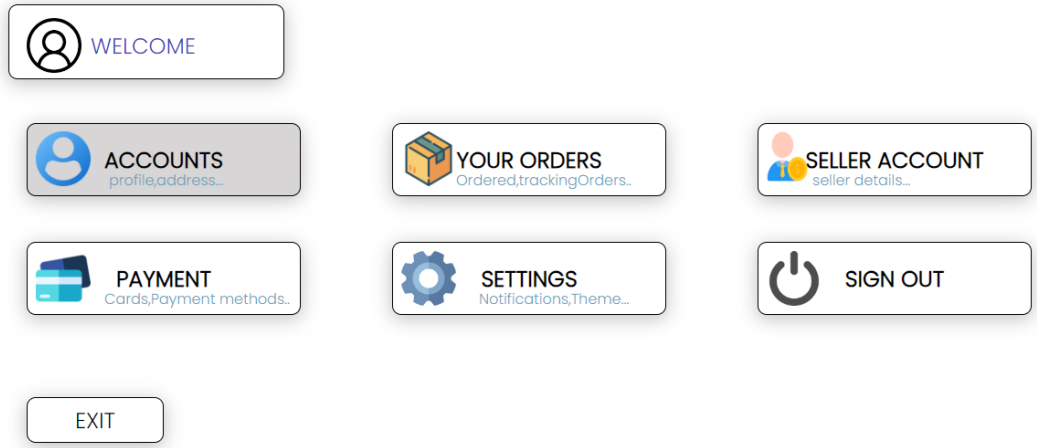
PRODUCT DETAILS:

Combining additional tables like inventory and category to the products table enables us to expand the functionality of product management. This is a key consideration when expanding the e-commerce platform to integrate PIM functionalities.

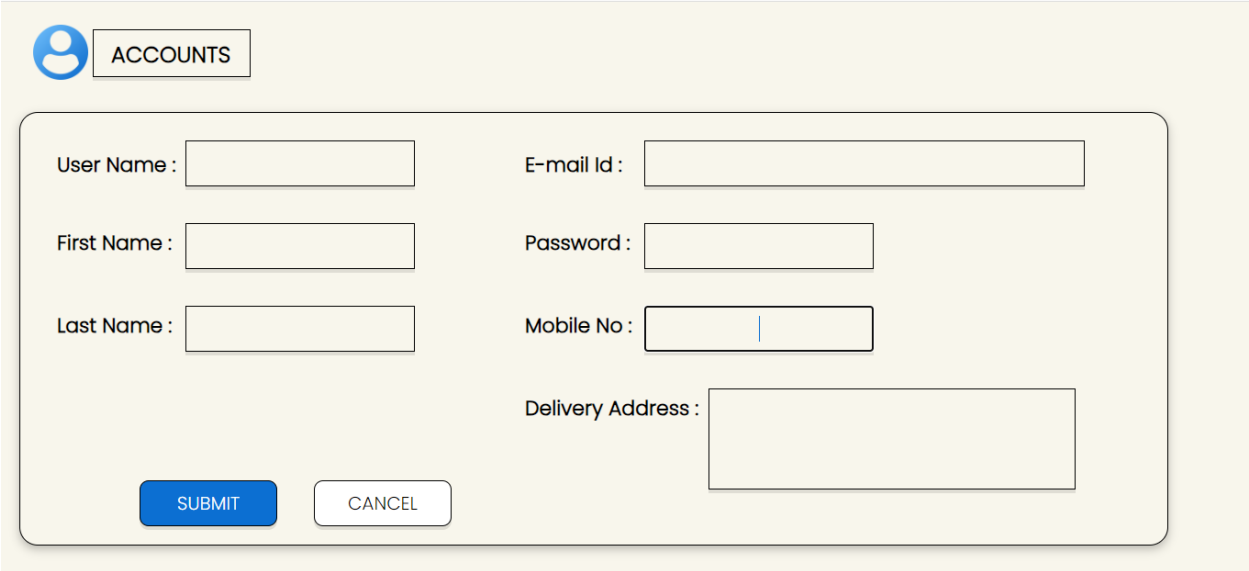


User Account Page

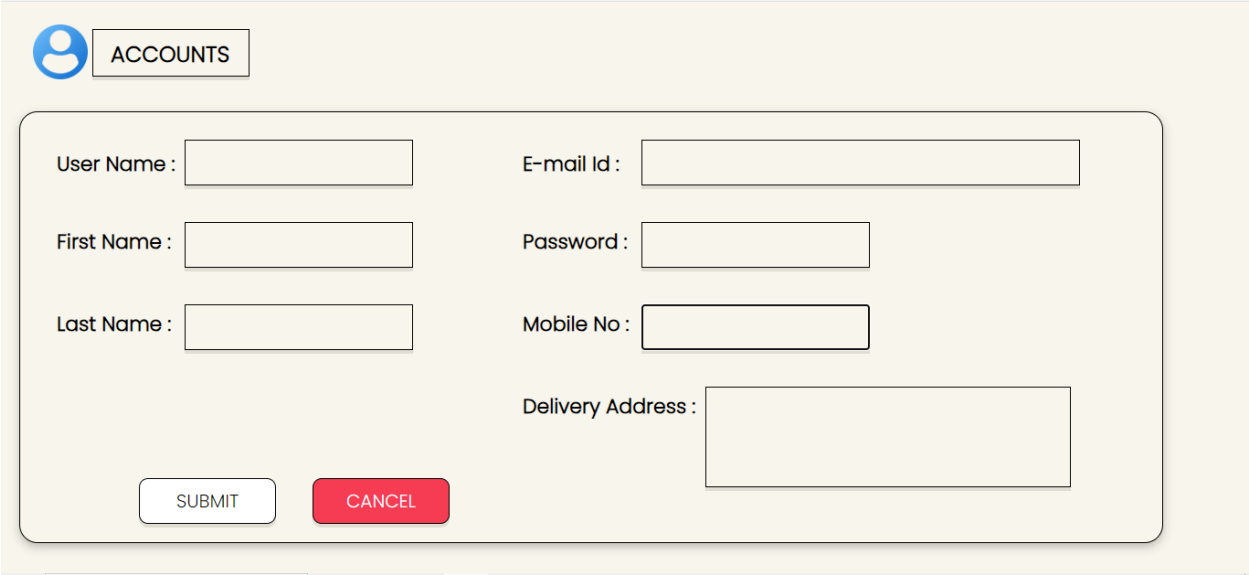
The account page consists of the details of the account holder.This page is given with various icons, on clicking on each icon a corresponding webpage is opened.



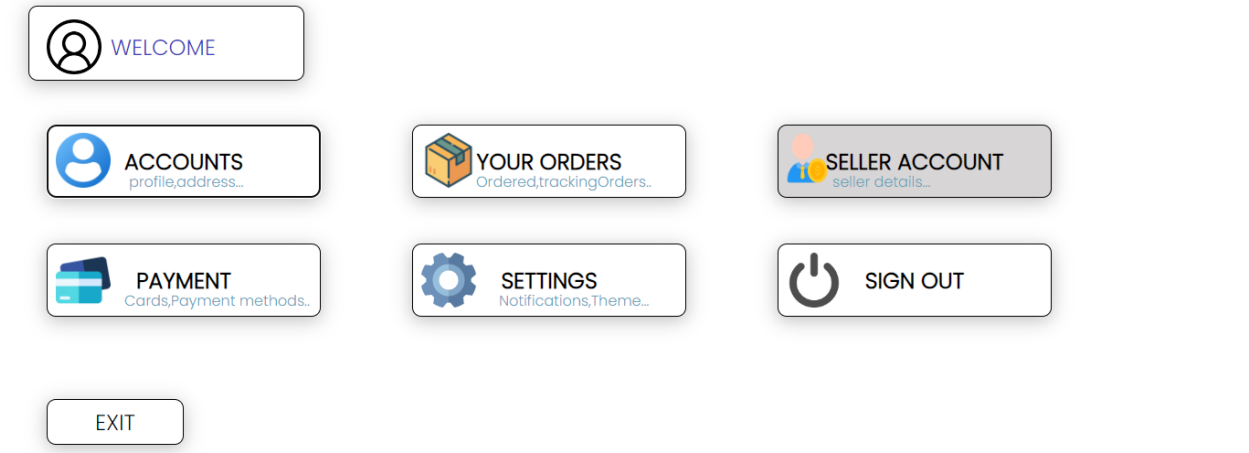
The webpage is designed in a way that it could get the data from the user. Once the data is entered by the user and the user clicks on the submit button the data is stored in a database. The data collected are stored using php and sql codes.



The webpage is also given with a button for canceling the submission process. This button prevents the user from not storing their data into a database.

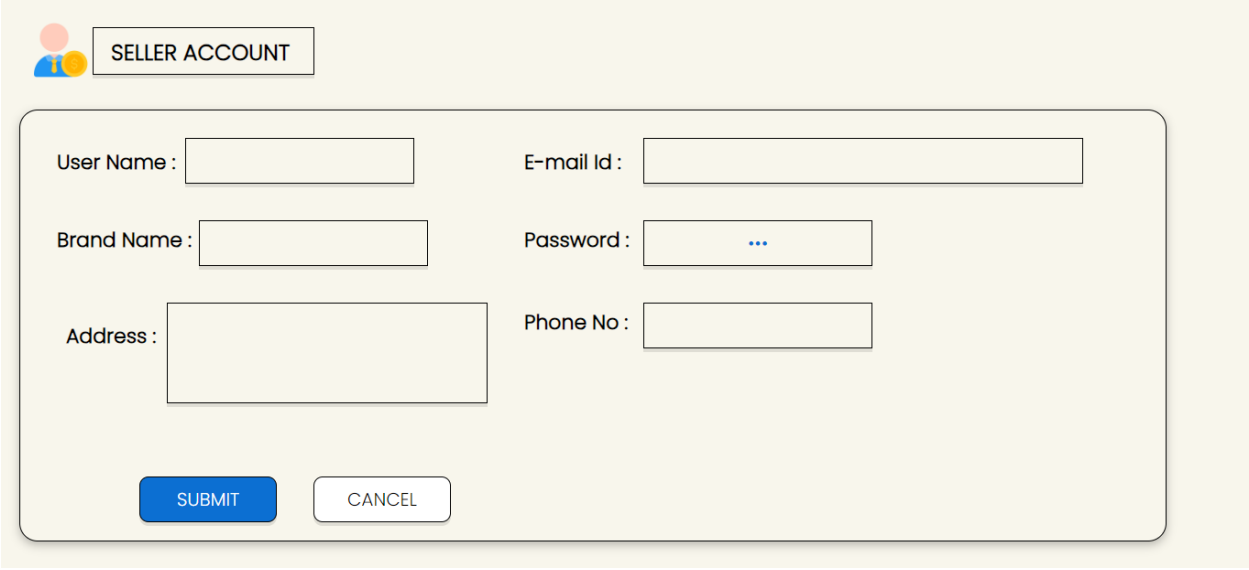


Seller Account Page



The user account page also provides a signup page for a seller. A seller could upload any of the products he/ she wishes to sell. When clicked on the seller account icon, it directs it to the seller account page.

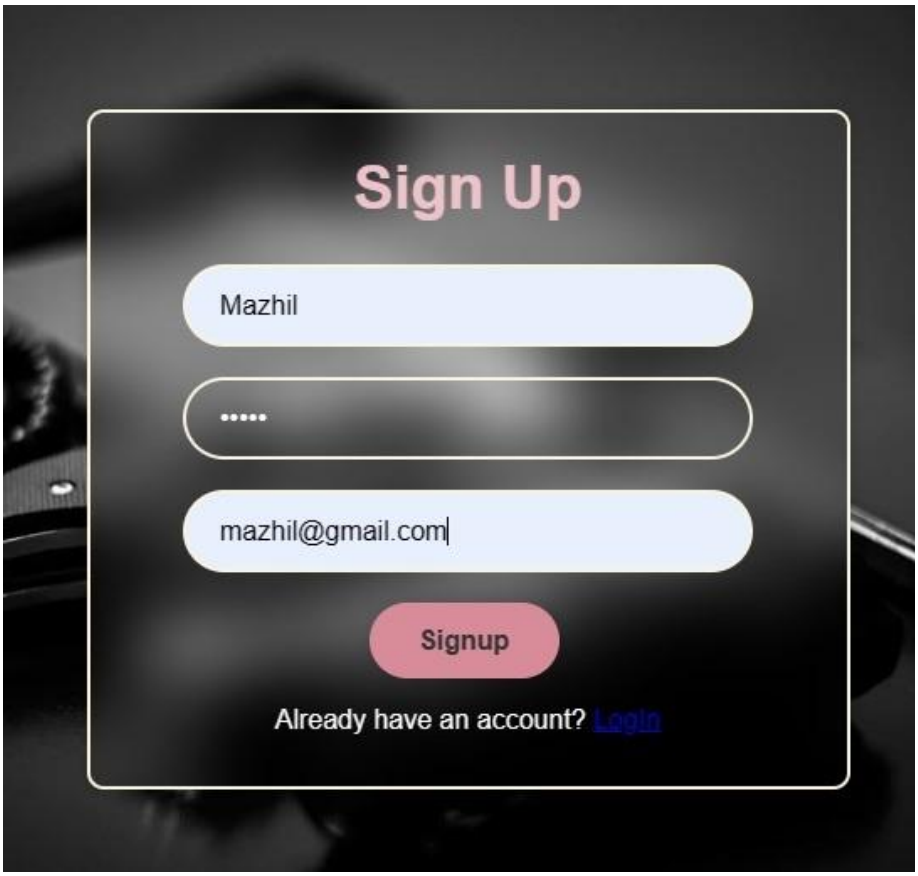
The seller account page is similar to the user's account page. Not only it looks similar to the user's account pge but it also works in the same way. It gets the information from a seller and stores it in a database.



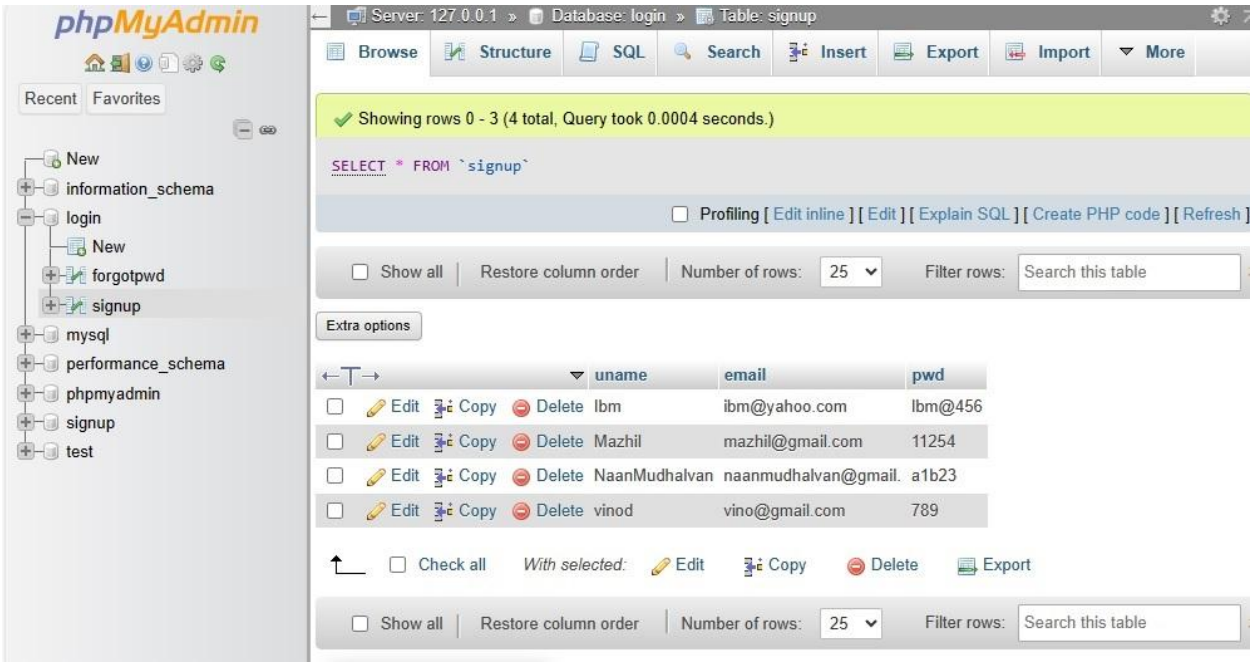
Two different databases are used to store the data of the user account page and the seller account page. These databases are connected with the web pages using php codes and are updated using sql language.

Signup page

The signup page is the entry point for a web application. To make the application available to everyone the data entered in the signup page is stored in a database.



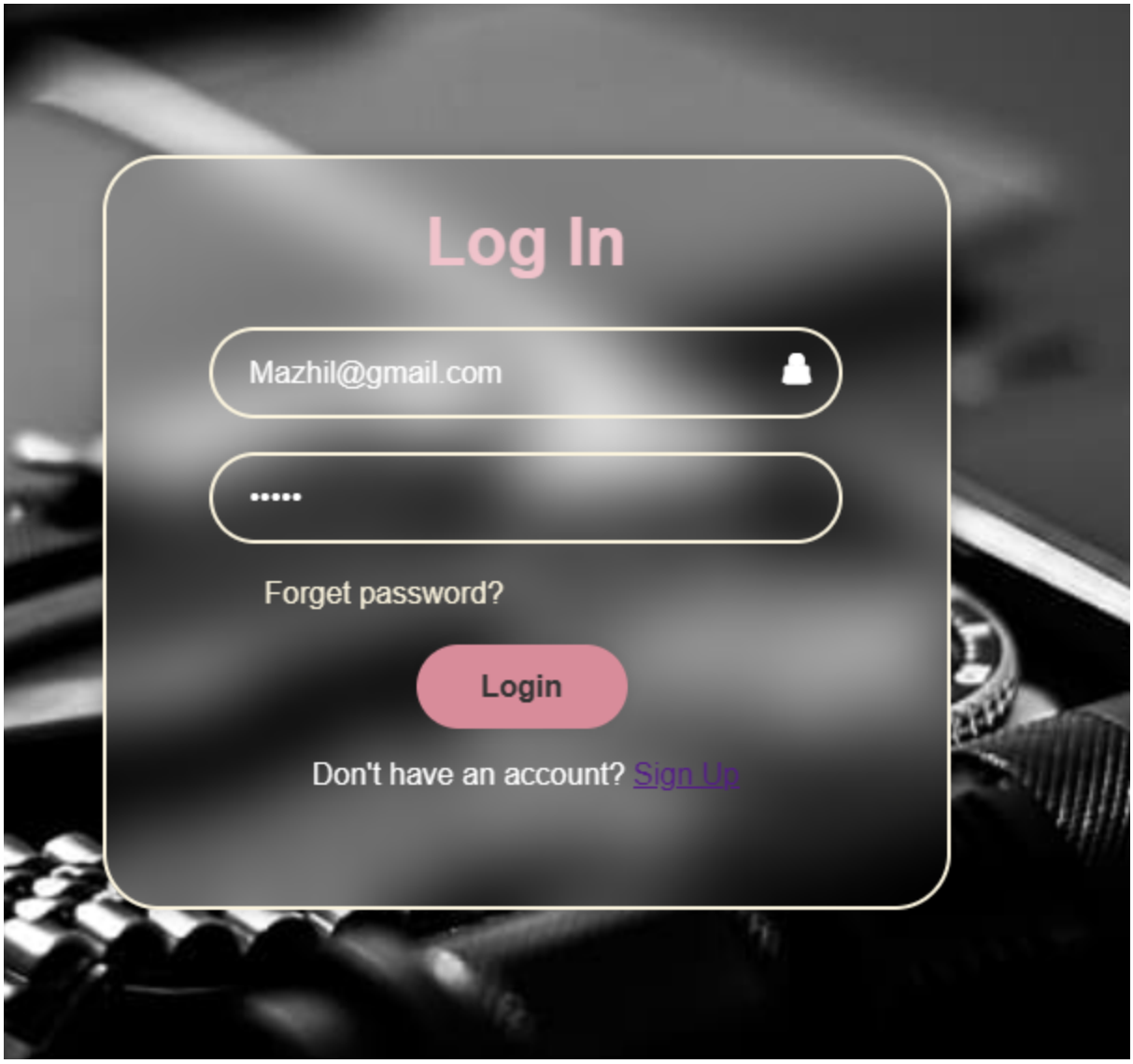
The database works in such a way that when the submit button is clicked in the signup page the data entered is automatically stored in the database in the form of tables.



The email id is marked as a primary key to identify every account uniquely. The user cannot sign up using the same email id more than one time. If tried then there appears a message saying “Someone already registered using this email”.

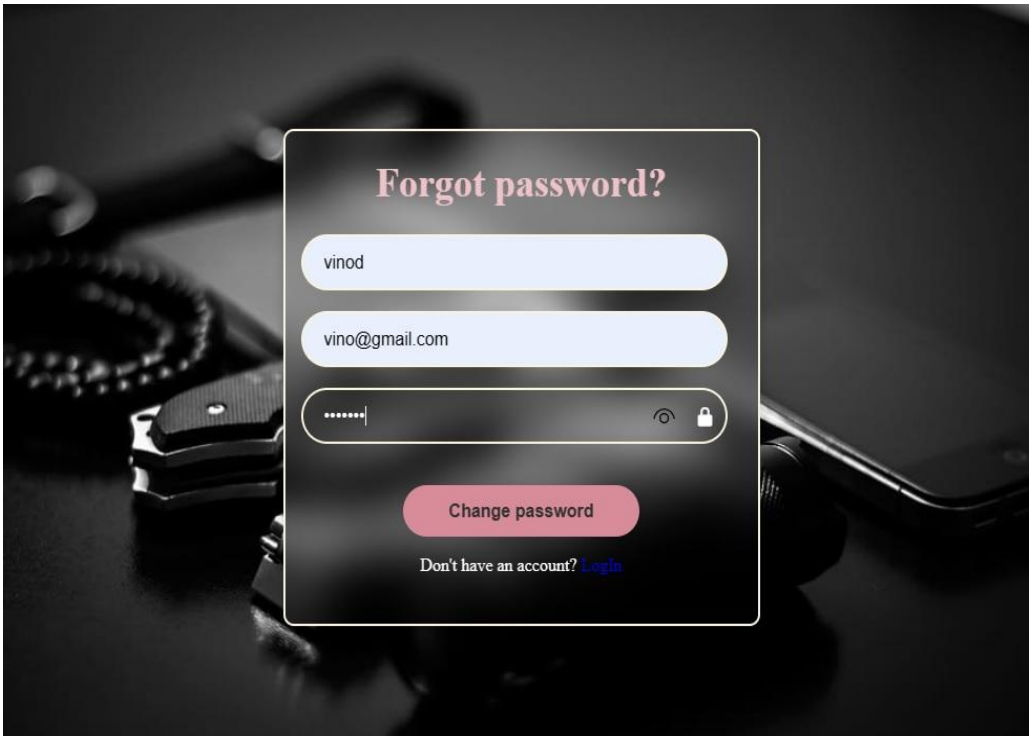
Login page

Based on the database created on phpmyadmin for the sign up page. The login page uses the same data to cross verify the credentials. If the data matches exactly, the user is granted permission into the application. If the data is mismatched with what is entered in the database they won't be granted access.

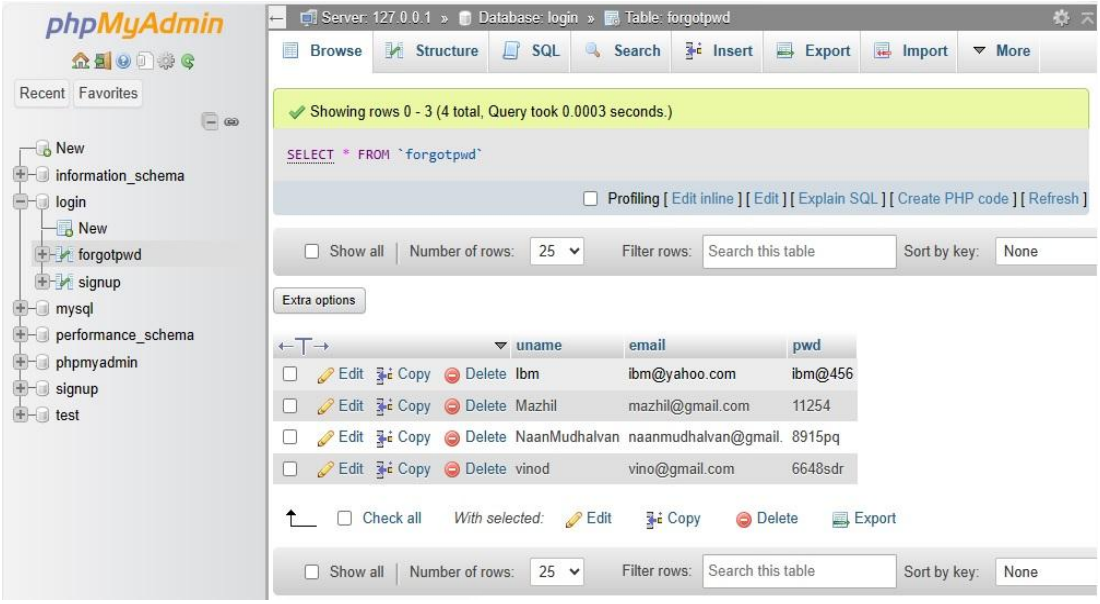


Forgot password page

The forgot password page is an important element in an application. This facilitates users to login into their accounts even when they don't remember their password. The forgot password page also has to store its data into a database.

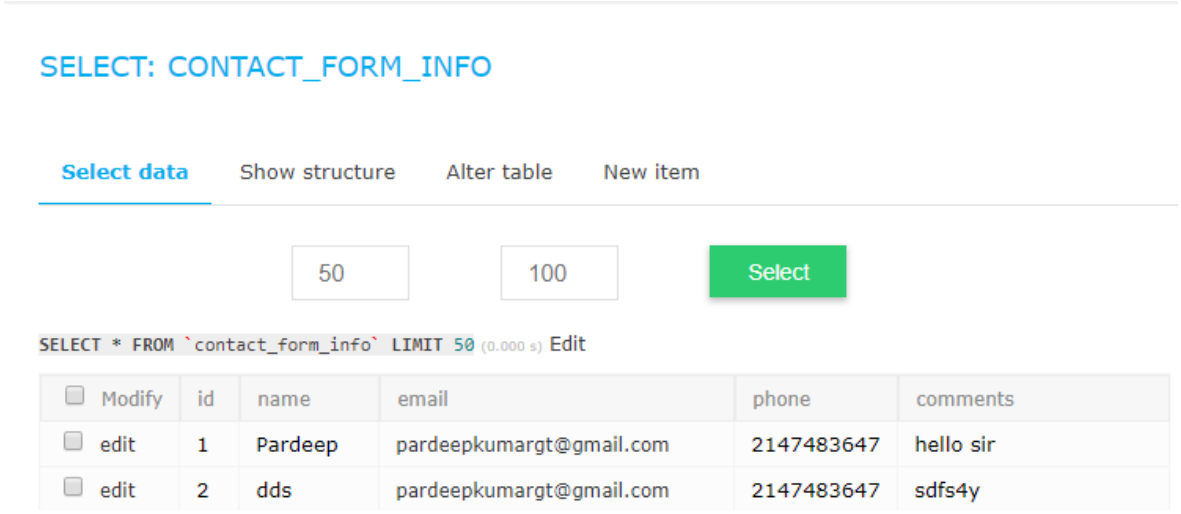


The database used for the signup page is used here in storing the changes made by this page. When data is entered into a forgot password page and it is submitted to the database of the signup page it checks for the user name , email id are the same and then the corresponding password is updated.



CONTACT US PAGE

A contact form is a form where the user /client can post any query to the admin/server. The user leaves a comment and that comment is stored in the server-side database for review. After a successful review the admin contact and reply to that user.



The contact form stores the information about the details of the user and their queries in the application. It stores the information given in the text fields and stores it to the database when given the "Submit" button. In PHP, you can validate the form data and send the entered message to an intended email address.