Online Shopping System

Project report (CA3) submitted in fulfilment of the requirements for the Degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

Shivam Nath

12018682

SUBJECT

INT220 - SERVER-SIDE SCRIPTING

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab (India)

November 2023

Table of Contents

INTRODUCTION

The digital age has ushered in a transformation in the way we approach commerce and consumerism. With the advent of the internet, the retail landscape has undergone a seismic shift, giving birth to the Online Shopping System - a virtual marketplace that has reshaped the very essence of retailing. In a world where convenience, accessibility, and efficiency are paramount, online shopping has emerged as a game-changer, impacting the lives of consumers and businesses alike.

This report delves into the multifaceted world of the Online Shopping System, seeking to explore its evolution, impact, and the dynamics that make it an integral part of modern life. We will examine the historical backdrop, technological underpinnings, key players, and the implications for consumers and businesses as we navigate this intricate web of e-commerce.

As we embark on this journey, it is essential to recognize that the online shopping ecosystem encompasses a vast array of platforms, from massive marketplaces like Amazon and Alibaba to niche boutiques specializing in artisanal goods. It caters to the diverse needs of consumers, from the tech-savvy millennials who crave instant gratification to the more traditional shoppers who seek the comfort of familiar retail experiences.

TECHNOLOGIES USED

This section encapsulates the array of technologies that have been harnessed in the development and construction of the Online Shopping System. Within this section, you can detail the various technologies—PHP, PhpMyAdmin, HTML, CSS, and Bootstrap—explaining their individual roles and collective contributions to the system's functionality and user interface.

1. PHP (Hypertext Preprocessor):

At the core of the Heritage Online Shopping System is PHP, a server-side scripting language renowned for its versatility and efficiency. PHP facilitates dynamic content generation,enabling seamless interaction between the user interface and the server. Its usage in this project ensures the smooth execution of server-side operations, making real-time data processing, such as booking validations and user authentication, possible.

2. MySQL:

MySQL is a popular open-source relational database management system, widely recognized for its reliability and performance. In the Online Shopping System, MySQL is used as the database backend to store and manage critical data such as Cart Information, product details, Customers details. Key aspects of MySQL's contribution include:

Data Storage: MySQL efficiently stores and organizes all the essential data related to Customers,Products, Prices, Number of Product , admin and Users along with Users Type in structured tables.

Data Retrieval: It provides powerful SQL querying capabilities, enabling the system to retrieve and present data to users as needed.

Data Integrity: MySQL ensures data integrity and security, enforcing constraints and permissions to prevent unauthorized access or data corruption.

Scalability: MySQL's scalability supports the growth of the Student Project Management system, making it capable of handling elections of varying sizes.

3. Bootstrap:

Employing Bootstrap, a front-end framework, has notably elevated the visual aesthetics and responsiveness of the ACI Online Shopping System. By leveraging Bootstrap's ready-to-use templates andadaptable grid system, crafting a user interface that is both visually appealing and mobile- friendly has become effortlessly achievable. This results in a uniform and immersive user experience, accessible across a spectrum of devices, thereby enhancing the application's overallusability.

4. PhpMyAdmin

PhpMyAdmin functions as the go-to database management tool for this project, providing an intuitive interface to interact with the MySQL database. This tool significantly simplifies the process of administering the database, enabling the smooth handling of user and reservation data storage, retrieval, and organization. Its role in maintaining data security, integrity, and facilitating seamless communication between the application and the database is pivotal to the project's success.

5. XAMPP:

XAMPP, standing for cross-platform, Apache, MySQL, PHP, and Perl, stands as an open-source solution for web server management, playing a pivotal role in the development of the Online Voting System through:

Unified Local Environment: XAMPP integrates essential components like the Apache server, MySQL database, PHP, and other vital tools into a singular package. This amalgamation facilitates developers in establishing and evaluating the system on their personal machines.

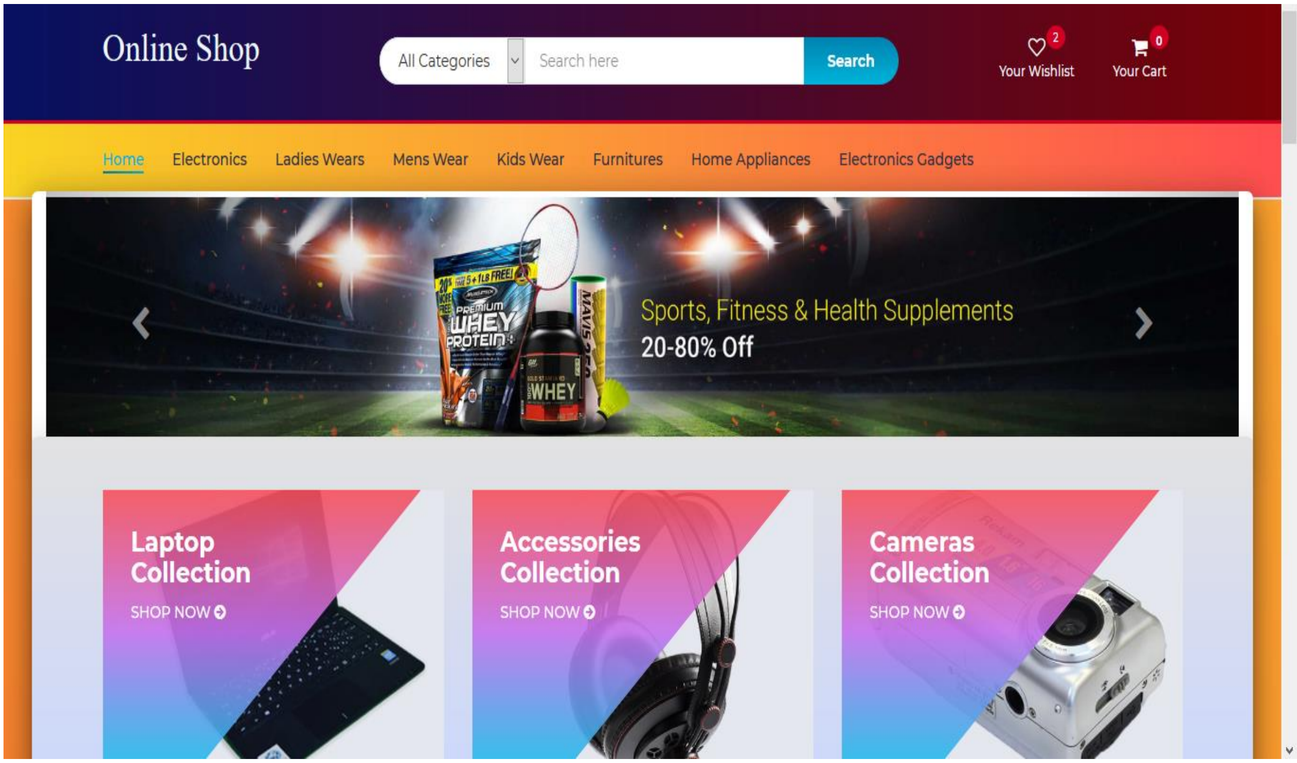
Testing and Debugging: XAMPP serves as a supportive platform for system evaluation, enabling thorough testing and debugging procedures. Its function ensures that the Online Shopping system operates seamlessly before it is deployed onto a live server.

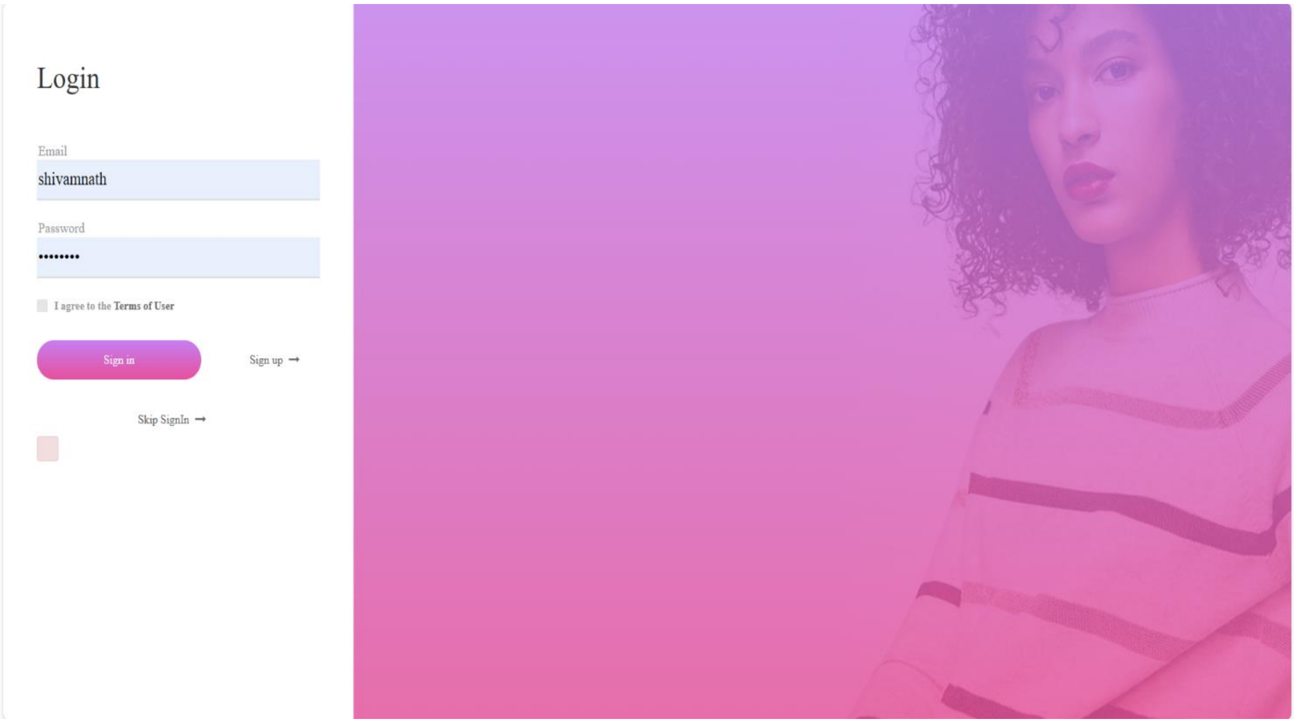
Seamless Migration: The convenience of transitioning the system from the local XAMPP environment to a production server is a noteworthy advantage. This simplifies the process of making the system live once it's thoroughly tested and ready for deployment.

6. JavaScript:

JavaScript, a flexible scripting language, is utilized to boost the dynamism and user responsiveness within the Online Shopping system. Its implementation enables real-time content updates, eliminating the need for entire page reloads and creating a smoother, more immersive user journey. Furthermore, it plays a crucial role in validating user inputs directly on the client side, ensuring data precision and bolstering the overall trustworthiness of information entered by users.

SNAPSHOTS


Online Shop
All Categories
Search here
Search
2
Your Wishlist
0
Your Cart
Home
Electronics
Ladies Wears
Mens Wear
Kids Wear
Furnitures
Home Appliances
Electronics Gadgets
Sports, Fitness & Health Supplements
20-80% Off
Laptop Collection
SHOP NOW
Accessories Collection
SHOP NOW
Cameras Collection
SHOP NOW



Login

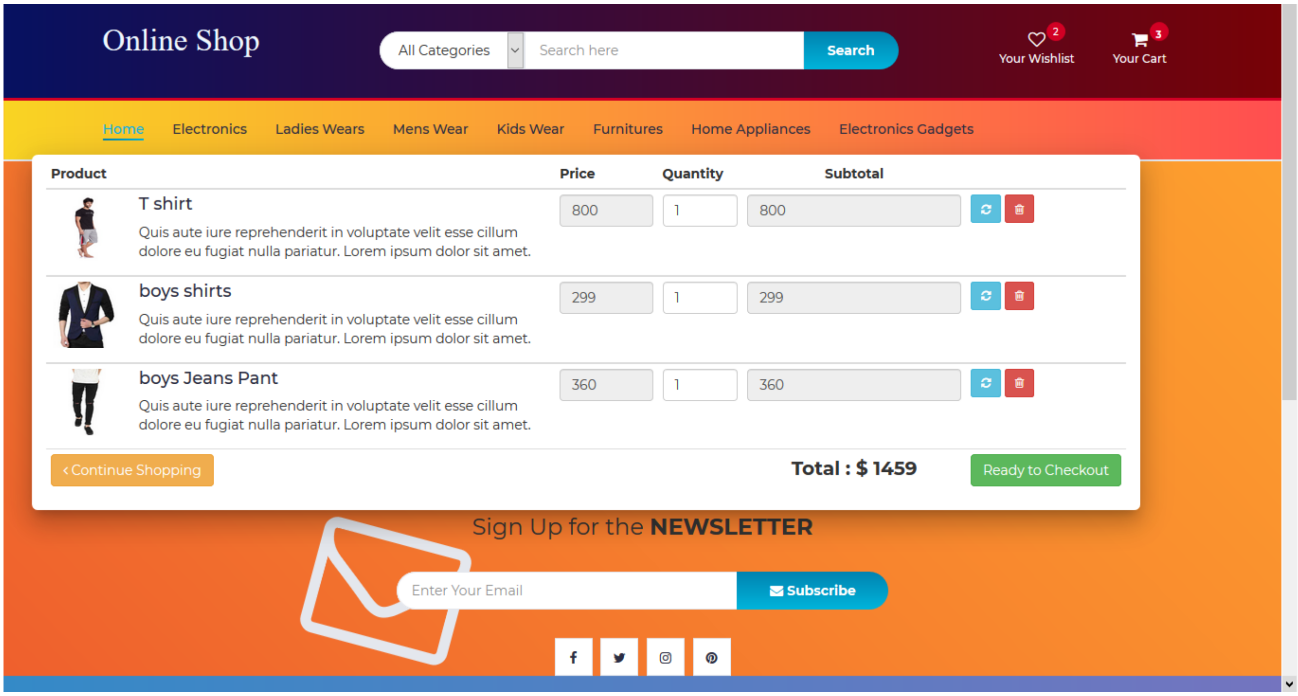
Email

shivamnath

Password

••••••••

□ I agree to the **Terms of User**



Product	Price	Quantity	Subtotal
T shirt Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Lorem ipsum dolor sit amet.	800	1	800
boys shirts Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Lorem ipsum dolor sit amet.	299	1	299
boys Jeans Pant Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Lorem ipsum dolor sit amet.	360	1	360

Total : $ 1459

Sign Up for the **NEWSLETTER**

Key Features:

1. **User-Friendly Interface:** The system should have an intuitive and user-friendly interface, making it easy for customers to browse products, add them to the cart, and complete transactions.

2. **Search and Navigation:** Efficient search functionality and intuitive navigation menus help users find products quickly, with filters and sorting options to refine search results.

3. **Product Listings:** Comprehensive product listings with high-quality images, detailed descriptions, pricing, and customer reviews provide valuable information for users.

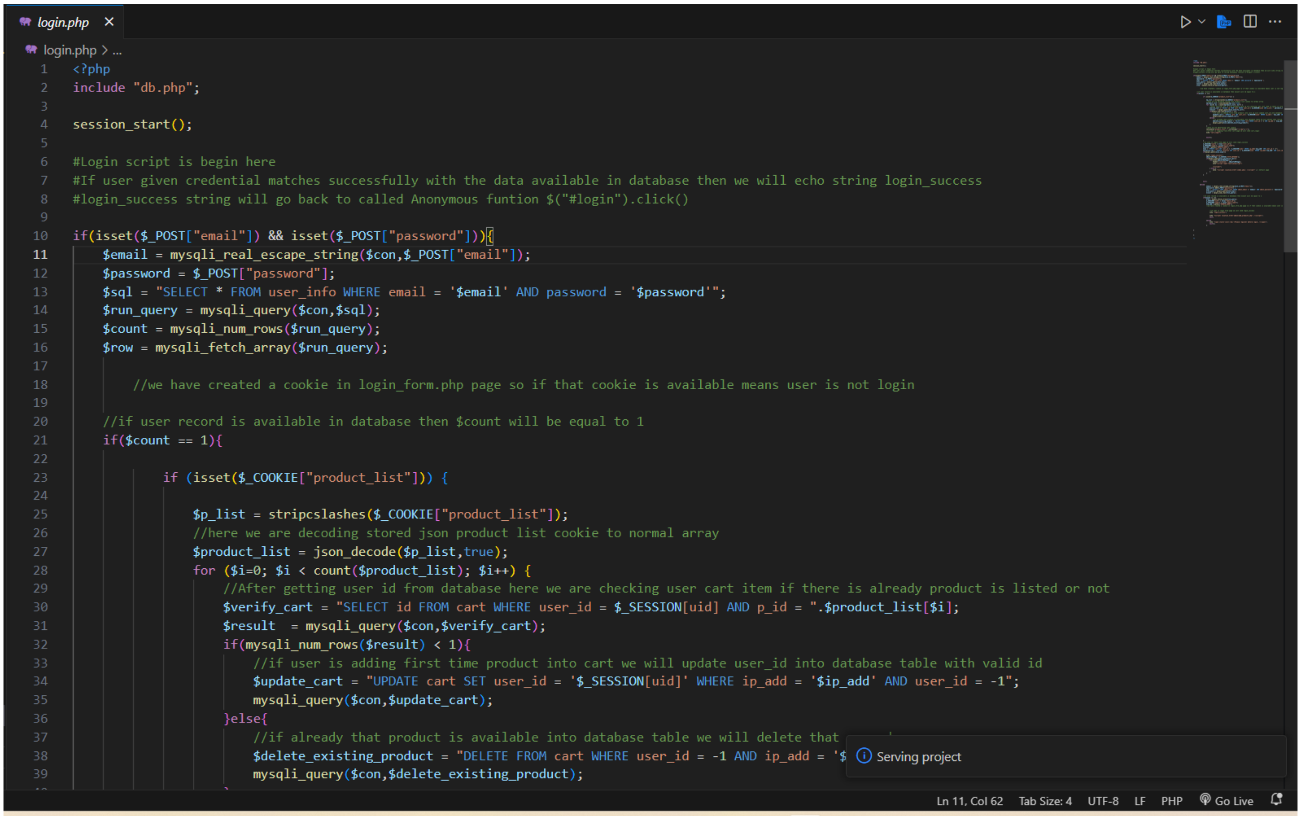
4. **Shopping Cart:** A well-designed shopping cart allows users to review and modify their selected items, estimate shipping costs, and proceed to checkout.

5. **Secure Payment Processing:** Secure payment gateways and multiple payment options (credit cards, digital wallets, etc.) instill trust in users and ensure the safety of their financial information.

6. **Order Tracking:** Order tracking and real-time updates on shipping and delivery status keep customers informed about their purchases.

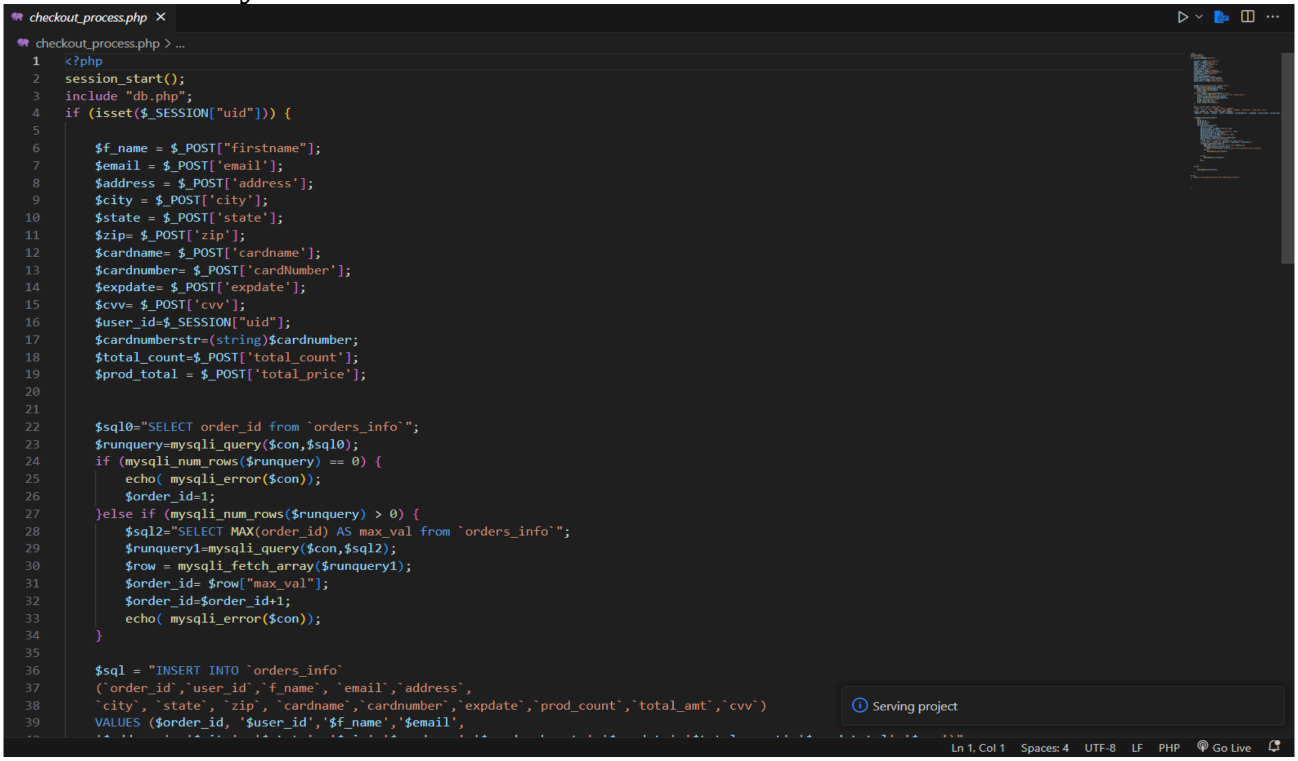
7. **Wishlist and Favorites:** Users can save items to Wishlist or favorites for future reference, facilitating repeat purchases and reducing cart abandonment.

Main Layout



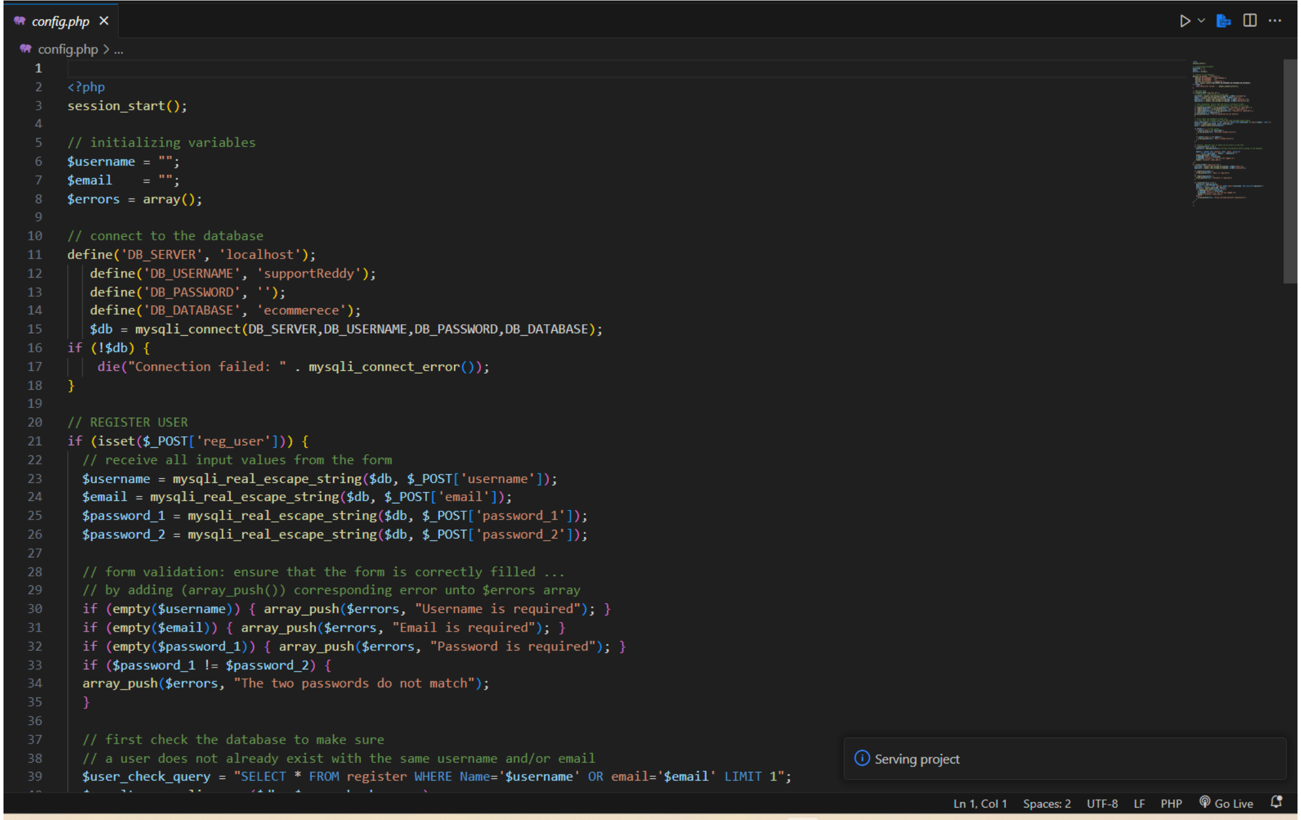
```php
<?php
include "db.php";

session_start();

#Login script is begin here
#If user given credential matches successfully with the data available in database then we will echo string login_success
#login_success string will go back to called Anonymous funtion $("#login").click()

if(isset($_POST["email"]) && isset($_POST["password"])){
    $email = mysqli_real_escape_string($con,$_POST["email"]);
    $password = $_POST["password"];
    $sql = "SELECT * FROM user_info WHERE email = '$email' AND password = '$password'";
    $run_query = mysqli_query($con,$sql);
    $count = mysqli_num_rows($run_query);
    $row = mysqli_fetch_array($run_query);

        //we have created a cookie in login_form.php page so if that cookie is available means user is not login

    //if user record is available in database then $count will be equal to 1
    if($count == 1){

            if (isset($_COOKIE["product_list"])) {

                $p_list = stripcslashes($_COOKIE["product_list"]);
                //here we are decoding stored json product list cookie to normal array
                $product_list = json_decode($p_list,true);
                for ($i=0; $i < count($product_list); $i++) {
                    //After getting user id from database here we are checking user cart item if there is already product is listed or not
                    $verify_cart = "SELECT id FROM cart WHERE user_id = $_SESSION[uid] AND p_id = ".$product_list[$i];
                    $result  = mysqli_query($con,$verify_cart);
                    if(mysqli_num_rows($result) < 1){
                        //if user is adding first time product into cart we will update user_id into database table with valid id
                        $update_cart = "UPDATE cart SET user_id = '$_SESSION[uid]' WHERE ip_add = '$ip_add' AND user_id = -1";
                        mysqli_query($con,$update_cart);
                    }else{
                        //if already that product is available into database table we will delete that
                        $delete_existing_product = "DELETE FROM cart WHERE user_id = -1 AND ip_add = '$
                        mysqli_query($con,$delete_existing_product);
```

Checkout Layout



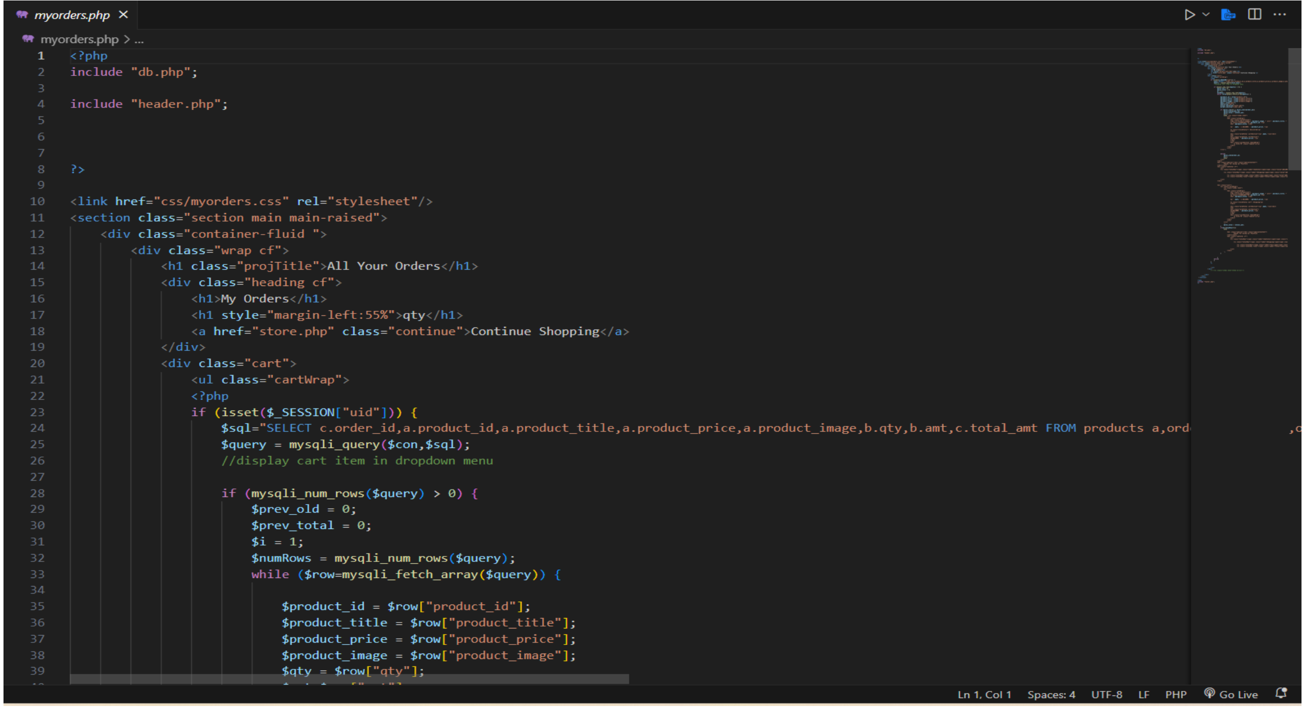
```php
<?php
session_start();
include "db.php";
if (isset($_SESSION["uid"])) {

    $f_name = $_POST["firstname"];
    $email = $_POST['email'];
    $address = $_POST['address'];
    $city = $_POST['city'];
    $state = $_POST['state'];
    $zip= $_POST['zip'];
    $cardname= $_POST['cardname'];
    $cardnumber= $_POST['cardNumber'];
    $expdate= $_POST['expdate'];
    $cvv= $_POST['cvv'];
    $user_id=$_SESSION["uid"];
    $cardnumberstr=(string)$cardnumber;
    $total_count=$_POST['total_count'];
    $prod_total = $_POST['total_price'];


    $sql0="SELECT order_id from `orders_info`";
    $runquery=mysqli_query($con,$sql0);
    if (mysqli_num_rows($runquery) == 0) {
        echo( mysqli_error($con));
        $order_id=1;
    }else if (mysqli_num_rows($runquery) > 0) {
        $sql2="SELECT MAX(order_id) AS max_val from `orders_info`";
        $runquery1=mysqli_query($con,$sql2);
        $row = mysqli_fetch_array($runquery1);
        $order_id= $row["max_val"];
        $order_id=$order_id+1;
        echo( mysqli_error($con));
    }

    $sql = "INSERT INTO `orders_info`
    (`order_id`,`user_id`,`f_name`, `email`,`address`,
    `city`, `state`, `zip`, `cardname`,`cardnumber`,`expdate`,`prod_count`,`total_amt`,`cvv`)
    VALUES ($order_id, '$user_id','$f_name','$email',
```

Config Layout



```php

<?php
session_start();

// initializing variables
$username = "";
$email    = "";
$errors = array();

// connect to the database
define('DB_SERVER', 'localhost');
  define('DB_USERNAME', 'supportReddy');
  define('DB_PASSWORD', '');
  define('DB_DATABASE', 'ecommerece');
  $db = mysqli_connect(DB_SERVER,DB_USERNAME,DB_PASSWORD,DB_DATABASE);
if (!$db) {
   die("Connection failed: " . mysqli_connect_error());
}

// REGISTER USER
if (isset($_POST['reg_user'])) {
  // receive all input values from the form
  $username = mysqli_real_escape_string($db, $_POST['username']);
  $email = mysqli_real_escape_string($db, $_POST['email']);
  $password_1 = mysqli_real_escape_string($db, $_POST['password_1']);
  $password_2 = mysqli_real_escape_string($db, $_POST['password_2']);

  // form validation: ensure that the form is correctly filled ...
  // by adding (array_push()) corresponding error unto $errors array
  if (empty($username)) { array_push($errors, "Username is required"); }
  if (empty($email)) { array_push($errors, "Email is required"); }
  if (empty($password_1)) { array_push($errors, "Password is required"); }
  if ($password_1 != $password_2) {
  array_push($errors, "The two passwords do not match");
  }

  // first check the database to make sure
  // a user does not already exist with the same username and/or email
  $user_check_query = "SELECT * FROM register WHERE Name='$username' OR email='$email' LIMIT 1";
```

My Order Layout



```php
<?php
include "db.php";

include "header.php";



?>

<link href="css/myorders.css" rel="stylesheet"/>
<section class="section main main-raised">
    <div class="container-fluid ">
        <div class="wrap cf">
            <h1 class="projTitle">All Your Orders</h1>
            <div class="heading cf">
                <h1>My Orders</h1>
                <h1 style="margin-left:55%">qty</h1>
                <a href="store.php" class="continue">Continue Shopping</a>
            </div>
            <div class="cart">
                <ul class="cartWrap">
                <?php
                if (isset($_SESSION["uid"])) {
                    $sql="SELECT c.order_id,a.product_id,a.product_title,a.product_price,a.product_image,b.qty,b.amt,c.total_amt FROM products a,ord
                    $query = mysqli_query($con,$sql);
                    //display cart item in dropdown menu

                    if (mysqli_num_rows($query) > 0) {
                        $prev_old = 0;
                        $prev_total = 0;
                        $i = 1;
                        $numRows = mysqli_num_rows($query);
                        while ($row=mysqli_fetch_array($query)) {

                            $product_id = $row["product_id"];
                            $product_title = $row["product_title"];
                            $product_price = $row["product_price"];
                            $product_image = $row["product_image"];
                            $qty = $row["qty"];
```

DATABASE

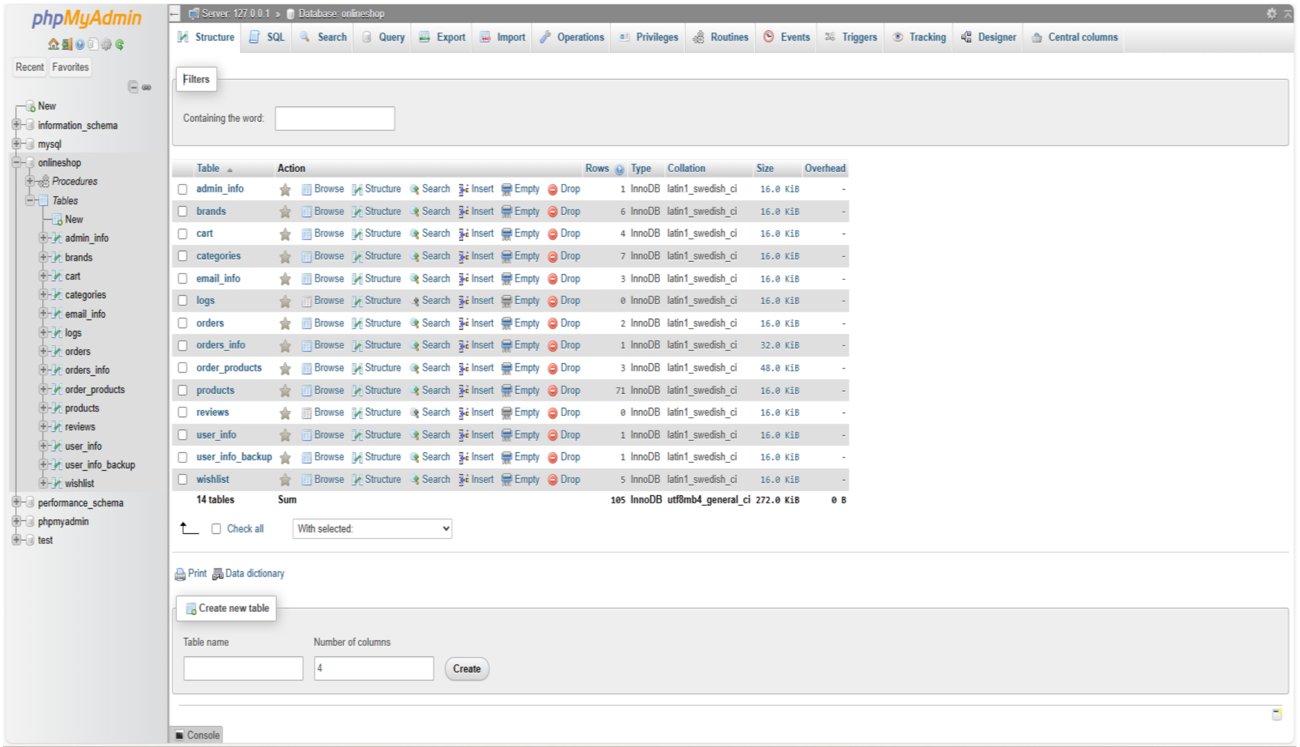
A well-structured and efficient database is essential to store and manage various types of data, including product information, user accounts, orders, and more. You can design a database using a relational database management system (RDBMS) like MySQL, PostgreSQL, or Microsoft SQL Server. Below, I'll provide an online shopping system:

1. **User Information Table:**

- User ID (Primary Key)
- Username
- Email
- Password (Hashed)
- Name
- Address
- Phone Number
- Payment Information (encrypted)

2. **Product Catalog Table:**

- Product ID (Primary Key)
- Product Name
- Description
- Category
- Price
- Stock Quantity
- Manufacturer
- Product Image URLs



Containing the word:

	Table	Action							Rows	Type	Collation	Size	Overhead
□	admin_info	☆	Browse	Structure	Search	Insert	Empty	Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	brands	☆	Browse	Structure	Search	Insert	Empty	Drop	6	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	cart	☆	Browse	Structure	Search	Insert	Empty	Drop	4	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	categories	☆	Browse	Structure	Search	Insert	Empty	Drop	7	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	email_info	☆	Browse	Structure	Search	Insert	Empty	Drop	3	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	logs	☆	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	orders	☆	Browse	Structure	Search	Insert	Empty	Drop	2	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	orders_info	☆	Browse	Structure	Search	Insert	Empty	Drop	1	InnoDB	latin1_swedish_ci	32.0 KiB	-
□	order_products	☆	Browse	Structure	Search	Insert	Empty	Drop	3	InnoDB	latin1_swedish_ci	48.0 KiB	-
□	products	☆	Browse	Structure	Search	Insert	Empty	Drop	71	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	reviews	☆	Browse	Structure	Search	Insert	Empty	Drop	0	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	user_info	☆	Browse	Structure	Search	Insert	Empty	Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	user_info_backup	☆	Browse	Structure	Search	Insert	Empty	Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	wishlist	☆	Browse	Structure	Search	Insert	Empty	Drop	5	InnoDB	latin1_swedish_ci	16.0 KiB	-
	14 tables	Sum							105	InnoDB	utf8mb4_general_ci	272.0 KiB	0 B

□ Check all   With selected:

Print   Data dictionary

Create new table

Table name:

Number of columns: 4

Create

PROJECT LINK

GitHub Link:

https://github.com/shivamnath2001/12018682_PHP.git

BIBLIOGRAPHY

1. PHP Documentation:

URL: [PHP Official Documentation] (https://www.php.net/manual/en/)

2. Bootstrap Documentation:

URL: [Bootstrap Documentation] (https://getbootstrap.com/docs/4.0/)

3. MySQL Documentation:

URL: [MySQL Documentation] (https://dev.mysql.com/doc/)

4. PhpMyAdmin Documentation:

URL: [PhpMyAdmin Documentation] (https://docs.phpmyadmin.net/)

5. JavaScript Documentation:

URL: [MDN JavaScript Documentation] (https://developer.mozilla.org/en-US/docs/Web/JavaScript)

Thank You