UNIVERSITI TUNKU ABDUL RAHMAN

LEE KONG CHIAN FACULTY OF ENGINEERING AND SCIENCE

UECS2194 WEB APPLICATION DEVELOPMENT

June 2024 Trimester

NO.	STUDENT NAME	STUDENT ID	Practical Group
1.	Kok Yu Hang	2200894	P4
2.	Cheng Wei En	2202205	P4
3.	Chong Rong Quan	2200880	P1
4.	Ong Lit Jay	2300594	P2

Table of Contents

Introduction

For our web application development project in the UECS2194 course, we have created an e-commerce website inspired by the in-game shop Angel 24 from Blue Archive. Our website is designed to sell a variety of instant noodles, instant meals, and snacks. The goal of this project is to demonstrate our ability to develop a dynamic, user-friendly web application that aligns with the course outcomes: constructing static and dynamic web pages using client-side and server-side scripts.

Our website merges the vibrant aesthetic of Angel 24 with the functional design of an e-commerce platform. This combination allows users to browse and purchase items with ease, creating a smooth and enjoyable online shopping experience. The site includes essential features such as a homepage, navigation menu, item listing page, product details page, and a cart/wishlist system. Additionally, we have implemented responsive design and enhanced usability features, including a dynamic search bar, real-time cart updates, and secure payment options, ensuring a seamless experience across all devices.

The development of this web application required the use of a range of technologies, including HTML, CSS, and JavaScript for the frontend, and a MySQL database for managing product and user data. By utilizing these tools, we have successfully built an interactive platform that meets the assignment requirements, showcases our technical abilities, and provides users with a visually appealing and functional online store.

In conclusion, our web application reflects both the technical and creative aspects of web development, demonstrating our ability to build and deploy a fully functional website that integrates both static and dynamic elements as outlined in the course's learning objectives.

Code Snippets with Detailed Explanation

Login page

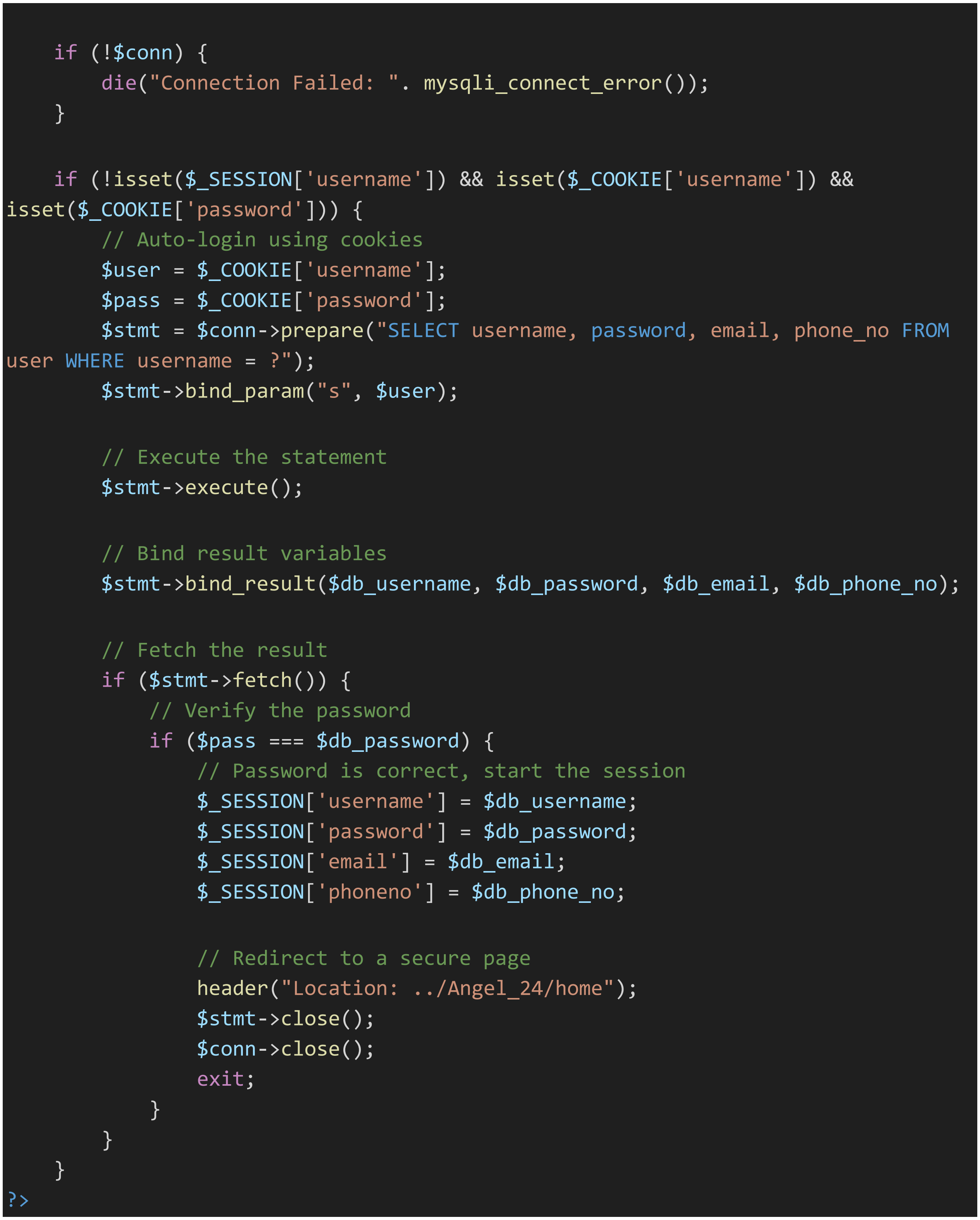
Angel_24/index.php

```
<?php
    session_start();
    $_SESSION['nameErr'] = $_SESSION['phoneErr'] = $_SESSION['imgErr'] = false;
?>

<!DOCTYPE html>
<html>
    <head>
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <link rel="stylesheet" href="style/mystyle2.css">
    </head>
    <body>
        <?php include("includes/checkCookie.php"); ?>
        <?php include("includes/header.php"); ?>

        <video id="wallpaper" autoplay muted loop poster="wallpaper.jpg">
            <source src="wallpaper.mp4" type="video/mp4">
            Your browser does not support HTML5 video.
        </video>

        <?php include("includes/loginForm.php"); ?>
    </body>
</html>
```

Explanation: This is the login page of the website. It will first start the session then call checkCookie.php to check if any $_COOKIE is set before. The video is the video background. Then it will prompt the login form by calling loginForm.php.

Angel_24/includes/checkCookie.php

```
<?php
    if (session_status() === PHP_SESSION_NONE) {
        session_start();
    }

    $conn = mysqli_connect("localhost","root","","angel_24");
```



```php
    if (!$conn) {
        die("Connection Failed: ". mysqli_connect_error());
    }

    if (!isset($_SESSION['username']) && isset($_COOKIE['username']) &&
isset($_COOKIE['password'])) {
        // Auto-login using cookies
        $user = $_COOKIE['username'];
        $pass = $_COOKIE['password'];
        $stmt = $conn->prepare("SELECT username, password, email, phone_no FROM
user WHERE username = ?");
        $stmt->bind_param("s", $user);

        // Execute the statement
        $stmt->execute();

        // Bind result variables
        $stmt->bind_result($db_username, $db_password, $db_email, $db_phone_no);

        // Fetch the result
        if ($stmt->fetch()) {
            // Verify the password
            if ($pass === $db_password) {
                // Password is correct, start the session
                $_SESSION['username'] = $db_username;
                $_SESSION['password'] = $db_password;
                $_SESSION['email'] = $db_email;
                $_SESSION['phoneno'] = $db_phone_no;

                // Redirect to a secure page
                header("Location: ../Angel_24/home");
                $stmt->close();
                $conn->close();
                exit;
            }
        }
    }
?>
```

Explanation: The aim of this code is to check if any $_COOKIE is set before, if is set, then it will check the ‘username’ and ‘password’ that are stored in $_COOKIE and use information stored in $_COOKIE to retrieve user information from the table ‘user’ from database ‘angel_24’. The user information such as ‘username’, ‘password’, ‘email’ and ‘phoneno’ retrieved is then stored in the $_SESSION. After that, the user will be redirected to ‘../Angel_24/home’ which is the home page.

Angel_24/includes/header.php

```
<div id="header">
    <image src="/Angel_24/images/Angel24_icon.png" id="icon"></image>
</div>
```

Explanation: This is the header of the website which only includes the logo of the website.

Angel_24/includes/loginForm.php

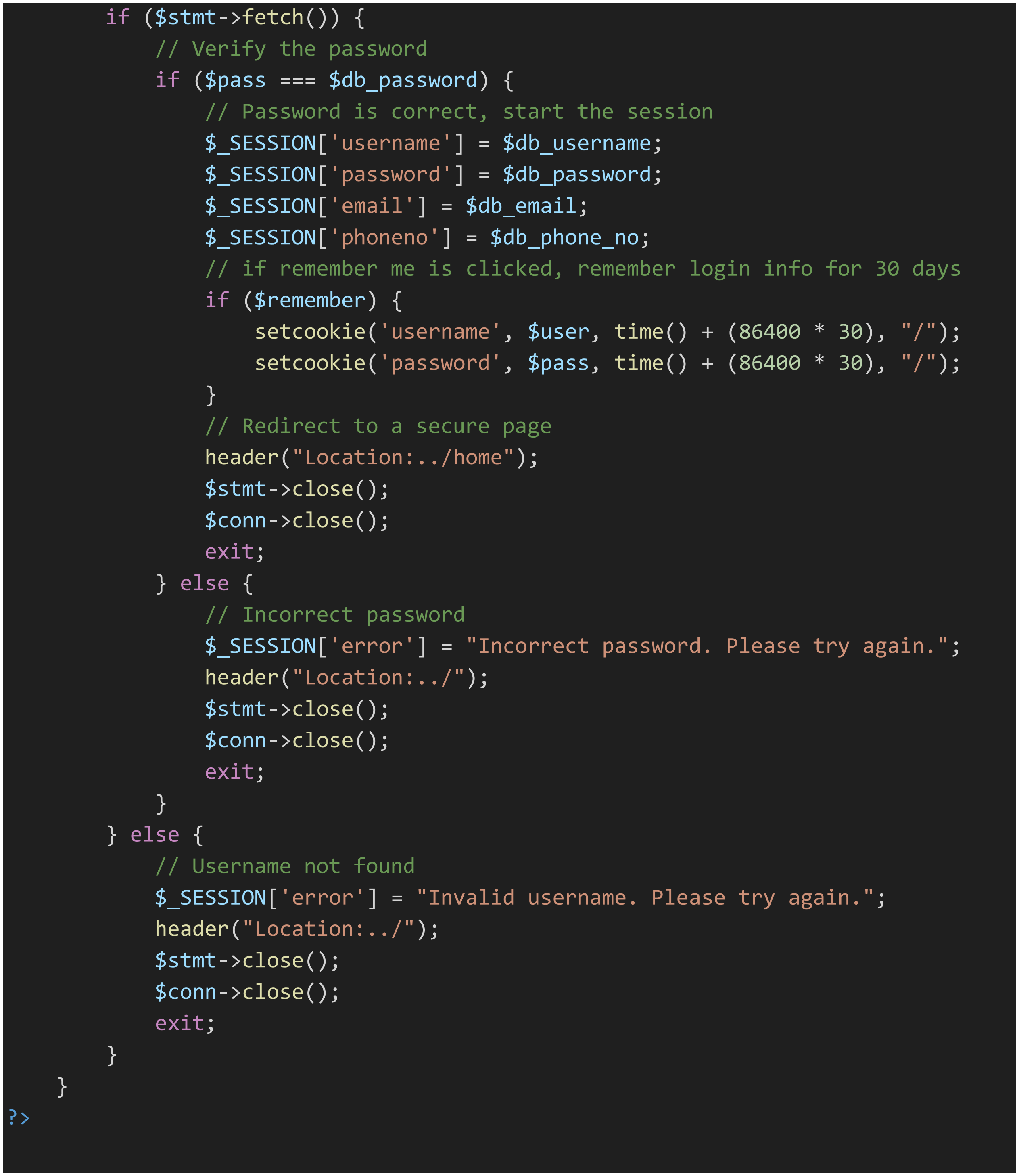
```
<div class="loginForm">
    <h3>Login</h3>
    <!-- Display error messages -->
    <?php
        if (isset($_SESSION['error'])) {
            echo '<p style="color: red;">' . $_SESSION['error'] . '</p>';
            unset($_SESSION['error']); // Clear the error message after
displaying
        }
    ?>
    <form class="formContent" action="includes/login.php" method="POST">
        <div class="container">
            <label for="username">Username: </label>
            <input type="text" name="username" placeholder="Enter Username"
required><br>
            <label for="password">Password: </label>
            <input type="password" name="password" placeholder="Enter Password"
required><br>
            <label id="remember-me-checkbox">
                <input type="checkbox" name="remember" id="remember"> Remember Me
            </label>
            <button type="submit">Login</button>
            <a id="register"
href="../Angel_24/register/register.php">Register</a>
        </div>
    </form>
```

```
</div>
```

Explanation: This code will show user the login form. At the beginning, the code will first check if any $_SESSION['error'] is set, if yes, then it will show the error message then unset the $_SESSION['error']. Then is the login form which receives 3 inputs, 'username', 'password' and 'remember me' checkbox. After user submit the login form, it will send the input to the login.php by using method 'POST' to verify user login details. There are also a 'register' button under the 'login' button which will redirect the user to register page.

Angel_24/includes/login.php

```
<?php
    session_start();

    $conn = mysqli_connect("localhost","root","","angel_24");

    if (!$conn) {
        die("Connection Failed: ". mysqli_connect_error());
    }

    if($_SERVER['REQUEST_METHOD'] === 'POST') {
        // Get the posted form data
        $user = $_POST['username'];
        $pass = $_POST['password'];
        $remember = isset($_POST['remember']);

        // Prepare and bind
        $stmt = $conn->prepare("SELECT username, password, email, phone_no FROM user WHERE BINARY username = ?");
        $stmt->bind_param("s", $user);

        // Execute the statement
        $stmt->execute();

        // Bind result variables
        $stmt->bind_result($db_username, $db_password, $db_email, $db_phone_no);
        echo "";

        // Fetch the result
```



```php
        if ($stmt->fetch()) {
            // Verify the password
            if ($pass === $db_password) {
                // Password is correct, start the session
                $_SESSION['username'] = $db_username;
                $_SESSION['password'] = $db_password;
                $_SESSION['email'] = $db_email;
                $_SESSION['phoneno'] = $db_phone_no;
                // if remember me is clicked, remember login info for 30 days
                if ($remember) {
                    setcookie('username', $user, time() + (86400 * 30), "/");
                    setcookie('password', $pass, time() + (86400 * 30), "/");
                }
                // Redirect to a secure page
                header("Location:../home");
                $stmt->close();
                $conn->close();
                exit;
            } else {
                // Incorrect password
                $_SESSION['error'] = "Incorrect password. Please try again.";
                header("Location:../");
                $stmt->close();
                $conn->close();
                exit;
            }
        } else {
            // Username not found
            $_SESSION['error'] = "Invalid username. Please try again.";
            header("Location:../");
            $stmt->close();
            $conn->close();
            exit;
        }
    }
?>
```

Explanation: This code will get the data from $_POST, which are ‘username’, ‘password’ and check is the ‘remember me’ checkbox is checked. Then, it will connect to database to get the user data for validation purposes. If the username and password is correct, then that user details will be set to the $_SESSION and if ‘remember-me’ checkbox is checked, then the ‘username’ and ‘password’ will be set to $_COOKIE, then the user will be redirected to home page. If the ‘password’ is incorrect, the code will set error messages to the $_SESSION[‘error’] and redirect to login page. If the ‘username’ is not found, the code will set error messages to the $_SESSION[‘error’] and redirect to login page.

Register page

Angel_24/register/register.php

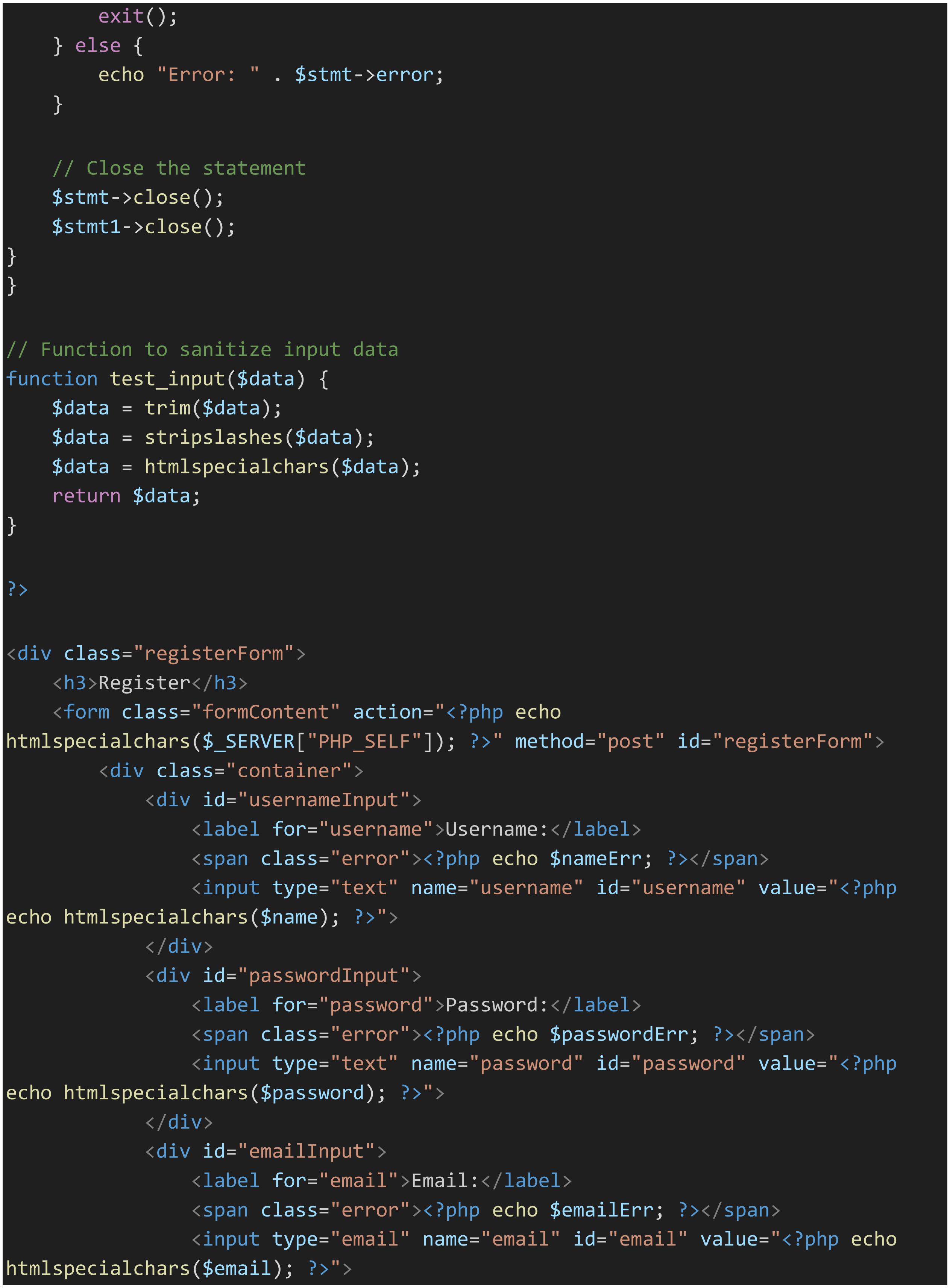
```
<?php
    session_start();
?>

<!DOCTYPE html>
<html>
    <head>
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <link rel="stylesheet" href="../style/mystyle2.css">
    </head>
    <body>
        <?php include("../includes/header.php"); ?>

        <video id="wallpaper" autoplay muted loop poster="../wallpaper.jpeg">
            <source src="../wallpaper.mp4" type="video/mp4">
            Your browser does not support HTML5 video.
        </video>

        <?php include("registerForm.php"); ?>

    </body>
</html>
```

Explanation: The code structure of the register page is similar to login page, but the difference is that it shows register form instead of login form.

Angel_24/register/registerForm.php

```
<link rel="stylesheet" href="../style/mystyle2.css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
```

```php
<?php

// Define variables and set to empty values
$nameErr = $passwordErr = $emailErr = $phoneErr = "";
$name = $password = $email = $phone = "";


// Connect to database
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "angel_24";
$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect_error) {
    die("Connection failed: ". $conn->connect_error);
}


// Check if form is submitted
if($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Validate name
    if (empty($_POST['username'])) {
        $nameErr = 'Name is required';
    } else {
        $name = test_input($_POST['username']);

        // Check if username already exists
        $stmt = $conn->prepare("SELECT username FROM user WHERE username = ?");
        $stmt->bind_param("s", $name);
        $stmt->execute();
        $stmt->store_result();

        if ($stmt->num_rows > 0) {
            $nameErr = 'Username already taken';
        }
        $stmt->close();
    }

    if (empty($_POST['password'])) {
        $passwordErr = 'Password is required';
    } else {
        $password = test_input($_POST['password']);
    }

    // Validate email
    if (empty($_POST['email'])) {
        $emailErr = "Email is required";
    } else {
        $email = test_input($_POST['email']);
        // Check if email address is well-formed
        if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
            $emailErr = "Invalid email format";
        }
    }

    // Validate phone number
    if (empty($_POST['phone'])) {
        $phoneErr = "Phone number is required";
    } else {
        $phone = test_input($_POST['phone']);
        //Check if phone number is valid
        if (!preg_match("/^\d{9,11}$/", $phone)) {
            $phoneErr = "Invalid phone number format. No '-'";
        }
    }



    // Check if there are no errors
   if (empty($nameErr) && empty($passwordErr) && empty($emailErr) &&
empty($phoneErr)) {
    // Prepare and bind the SQL statement
    $stmt = $conn->prepare("INSERT INTO user (username, password, email,
phone_no) VALUES (?, ?, ?, ?)");
    $stmt->bind_param("ssss", $name, $password, $email, $phone);

    $stmt1 = $conn->prepare("INSERT INTO profile_pic (username) VALUES (?)");
    $stmt1->bind_param("s", $name);

    // Execute the statement
    if ($stmt->execute() && $stmt1->execute()) {
        // Redirect to home page
        header("Location:../");
```



```
        exit();
    } else {
        echo "Error: " . $stmt->error;
    }

    // Close the statement
    $stmt->close();
    $stmt1->close();
}
}

// Function to sanitize input data
function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}

?>

<div class="registerForm">
    <h3>Register</h3>
    <form class="formContent" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post" id="registerForm">
        <div class="container">
            <div id="usernameInput">
                <label for="username">Username:</label>
                <span class="error"><?php echo $nameErr; ?></span>
                <input type="text" name="username" id="username" value="<?php
echo htmlspecialchars($name); ?>">
            </div>
            <div id="passwordInput">
                <label for="password">Password:</label>
                <span class="error"><?php echo $passwordErr; ?></span>
                <input type="text" name="password" id="password" value="<?php
echo htmlspecialchars($password); ?>">
            </div>
            <div id="emailInput">
                <label for="email">Email:</label>
                <span class="error"><?php echo $emailErr; ?></span>
                <input type="email" name="email" id="email" value="<?php echo
htmlspecialchars($email); ?>">
```

```
            </div>
            <div id="phoneInput">
                <label for="phone">Phone:</label>
                <span class="error"><?php echo $phoneErr; ?></span>
                <input type="tel" name="phone" id="phone" value="<?php echo
htmlspecialchars($phone); ?>">
            </div>
            <br>
            <button type="submit" name="submit" id="submit">Submit</button>
            <a href="../index.php" class="return-btn"><i class="fa fa-arrow-
left"></i> Back</a>
        </div>
    </form>
</div>
```

Explanation: This code included two parts, one is for error validation, and one is the register form itself. The error validation part's code will first set variables to "", then connect to the database. The code will check the input so that all input is not empty or else it will set error messages to related error variables such as $nameErr. If username is set, it will retrieve all username from the table 'user' and check if the username is taken by other users or not. If username is taken, then username error will set to the $nameErr. This code will check the email format, if format is incorrect, then email error will set to the $emailErr. The code will then check phone number format to make sure that only number is entered, no '-' between and length must between 9 to 11 digits, else it will set phone number format error messages to $phoneErr. If there are no any error messages is set, the code will then insert the user data into the 'user' table and username into the 'profile_pic' table, then redirect to login page. The register form part receive input from user such as username, password, email address and phone number. It will show error messages if any input's format is incorrect.

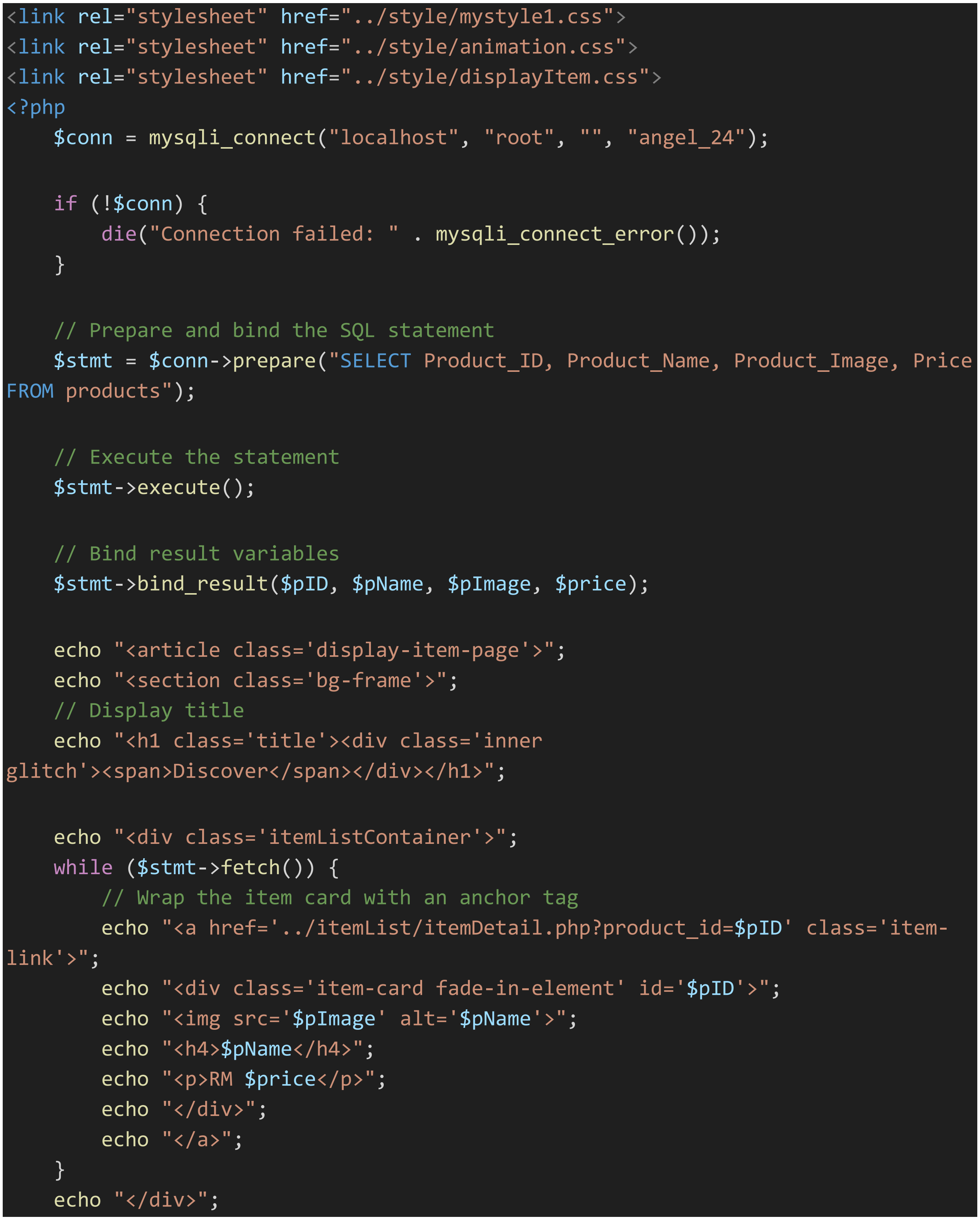
Home page

Angel_24/home/index.php

```
<?php
session_start();
// Check if the user is not logged in
if(!isset($_SESSION['username'])) {
    header('Location: ../index.php');
    exit();
}
?>

<!DOCTYPE html>
<html>
    <head>
        <title>Angel 24</title>
        <link rel="stylesheet" href="../style/mystyle1.css">
        <link rel="stylesheet" href="../style/animation.css">
        <meta name="viewport" content="width=device-width, initial-scale=1">
    </head>
    <body>
        <?php include("../includes/header.php"); ?>
        <?php include("../includes/navigation.php"); ?>
        <div class="top-section-container fade-in">
            <div class="welcome-container">
            <video id="wallpaper" autoplay muted loop
poster="../images/Sanctum_Tower_1.png">
                <source src="../home_background.mp4" type="video/mp4">
                Your browser does not support HTML5 video.
            </video>
            <div class="dot-background"></div>
            <h1>Welcome To Angel_24, <?php echo
htmlspecialchars($_SESSION['username']); ?></h1>
            </div>
        </div>

        <div class="content-container fade-in-element">
            <?php include('displayAll.php'); ?>
        </div>
        <script src="../javascript/animation.js"></script>

    </body>
</html>
```

Explanation: This code will first check is the $_SESSION['username'] is set, if not then it will redirect user to login page. The home page has a header, navigation bar, top-section-container and content-container. 'top-section-container' has a welcome-container inside which will show the welcome message and a loop video background. In the content-container, the code will call displayAll.php to display all items in the database. The animation.js is included to add animation to all item-cards in displayAll.php.

Angel_24/home/displayAll.php



```php
<link rel="stylesheet" href="../style/mystyle1.css">
<link rel="stylesheet" href="../style/animation.css">
<link rel="stylesheet" href="../style/displayItem.css">
<?php
    $conn = mysqli_connect("localhost", "root", "", "angel_24");

    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());
    }

    // Prepare and bind the SQL statement
    $stmt = $conn->prepare("SELECT Product_ID, Product_Name, Product_Image, Price FROM products");

    // Execute the statement
    $stmt->execute();

    // Bind result variables
    $stmt->bind_result($pID, $pName, $pImage, $price);

    echo "<article class='display-item-page'>";
    echo "<section class='bg-frame'>";
    // Display title
    echo "<h1 class='title'><div class='inner glitch'><span>Discover</span></div></h1>";

    echo "<div class='itemListContainer'>";
    while ($stmt->fetch()) {
        // Wrap the item card with an anchor tag
        echo "<a href='../itemList/itemDetail.php?product_id=$pID' class='item-link'>";
        echo "<div class='item-card fade-in-element' id='$pID'>";
        echo "<img src='$pImage' alt='$pName'>";
        echo "<h4>$pName</h4>";
        echo "<p>RM $price</p>";
        echo "</div>";
        echo "</a>";
    }
    echo "</div>";
    echo "<br><br>";
    echo "</section>";
    echo "</article>";

    // Close connections
    $stmt->close();
    $conn->close();

?>
```

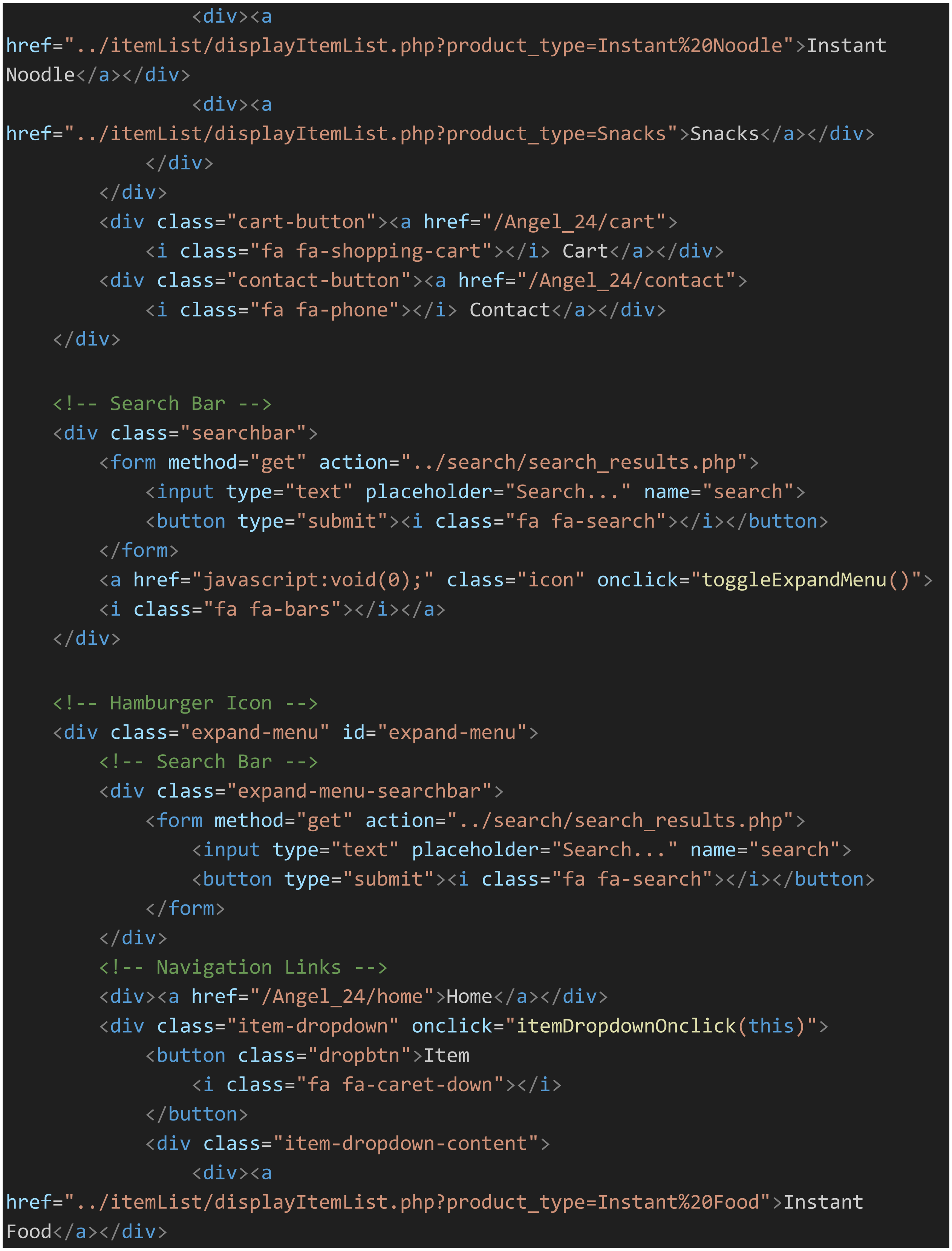
Explanation: This code will retrieve all products information (Product_ID, Product_Name, Product_Image, Price) from ‘products’ table. Then it will loop repeatedly to generate item-card and will stop looping when there is no more products information. Each item-card is covered by a <a> which will open the item’s details page.

Navigation bar

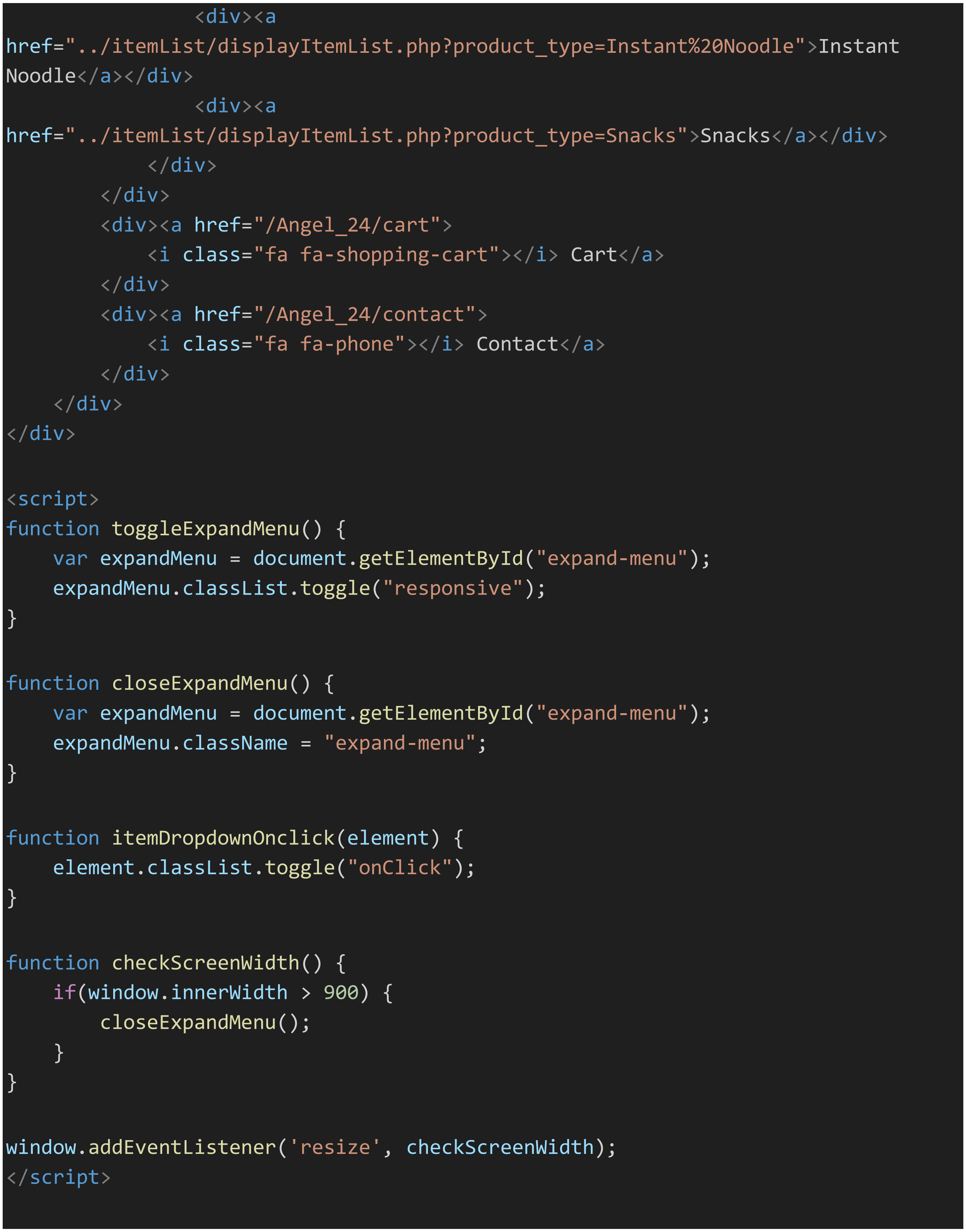
Angel_24/includes/navigation.php

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css">
<link rel="stylesheet" href="../style/navigationStyle.css">
<link rel="stylesheet" href="../style/profile.css">
<meta name="viewport" content="width=device-width, initial-scale=1">

<?php
// Fetch the profile picture from the database
$conn = new mysqli("localhost", "root", "", "angel_24");
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

if (session_status() === PHP_SESSION_NONE) {
    session_start();
}

$logged_in_user = $_SESSION['username'];
```

```
$stmt = $conn->prepare("SELECT image_data, image_type FROM profile_pic WHERE username = ?");
$stmt->bind_param("s", $logged_in_user);
$stmt->execute();
$stmt->bind_result($imgData, $imgType);
$stmt->fetch();
$stmt->close();
$conn->close();

// Convert the binary image data to base64
if ($imgData) {
    $base64Image = 'data:' . $imgType . ';base64,' . base64_encode($imgData);
} else {
    // Use a default image if no profile picture is uploaded
    $base64Image = '../images/profile/profile-pic.png';  // Path to default profile image
}
?>

<div class="top_nav" id="topNav">
    <div class="nav-left">
        <div class="profile-dropdown">
            <div class="profile-btn" style="background-image: url('<?php echo $base64Image; ?>');" onmouseover="closeExpandMenu()"></div>
            <div class="profile-menu">
                <a href="../profile/"><i class="fas fa-user-edit"></i>Edit Profile</a>
                <a href="../includes/logout.php"><i class="fas fa-sign-out-alt"></i>Log out</a>
            </div>


        </div>
        <!-- Navigation Links -->
        <div class="home-button"><a href="/Angel_24/home">Home</a></div>
        <div class="item-dropdown">
            <button class="dropbtn">Item
                <i class="fa fa-caret-down"></i>
            </button>
            <div class="item-dropdown-content">
                <div><a href="../itemList/displayItemList.php?product_type=Instant%20Food">Instant Food</a></div>
```



```
                <div><a
href="../itemList/displayItemList.php?product_type=Instant%20Noodle">Instant
Noodle</a></div>
                <div><a
href="../itemList/displayItemList.php?product_type=Snacks">Snacks</a></div>
            </div>
        </div>
        <div class="cart-button"><a href="/Angel_24/cart">
            <i class="fa fa-shopping-cart"></i> Cart</a></div>
        <div class="contact-button"><a href="/Angel_24/contact">
            <i class="fa fa-phone"></i> Contact</a></div>
    </div>

    <!-- Search Bar -->
    <div class="searchbar">
        <form method="get" action="../search/search_results.php">
            <input type="text" placeholder="Search..." name="search">
            <button type="submit"><i class="fa fa-search"></i></button>
        </form>
        <a href="javascript:void(0);" class="icon" onclick="toggleExpandMenu()">
        <i class="fa fa-bars"></i></a>
    </div>

    <!-- Hamburger Icon -->
    <div class="expand-menu" id="expand-menu">
        <!-- Search Bar -->
        <div class="expand-menu-searchbar">
            <form method="get" action="../search/search_results.php">
                <input type="text" placeholder="Search..." name="search">
                <button type="submit"><i class="fa fa-search"></i></button>
            </form>
        </div>
        <!-- Navigation Links -->
        <div><a href="/Angel_24/home">Home</a></div>
        <div class="item-dropdown" onclick="itemDropdownOnclick(this)">
            <button class="dropbtn">Item
                <i class="fa fa-caret-down"></i>
            </button>
            <div class="item-dropdown-content">
                <div><a
href="../itemList/displayItemList.php?product_type=Instant%20Food">Instant
Food</a></div>
```



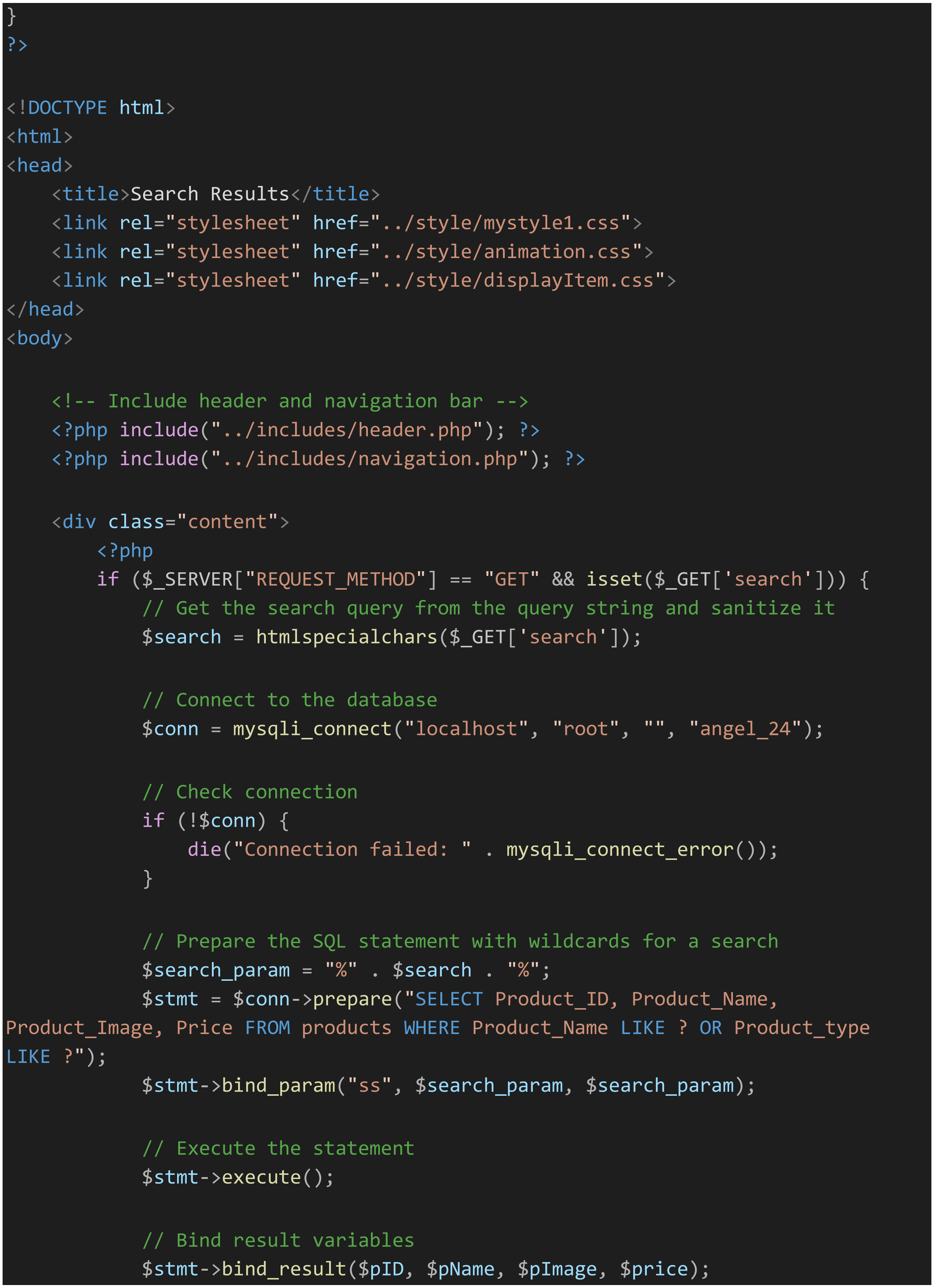
```
                <div><a href="../itemList/displayItemList.php?product_type=Instant%20Noodle">Instant Noodle</a></div>
                <div><a href="../itemList/displayItemList.php?product_type=Snacks">Snacks</a></div>
            </div>
        </div>
        <div><a href="/Angel_24/cart">
            <i class="fa fa-shopping-cart"></i> Cart</a>
        </div>
        <div><a href="/Angel_24/contact">
            <i class="fa fa-phone"></i> Contact</a>
        </div>
    </div>
</div>

<script>
function toggleExpandMenu() {
    var expandMenu = document.getElementById("expand-menu");
    expandMenu.classList.toggle("responsive");
}

function closeExpandMenu() {
    var expandMenu = document.getElementById("expand-menu");
    expandMenu.className = "expand-menu";
}

function itemDropdownOnclick(element) {
    element.classList.toggle("onClick");
}

function checkScreenWidth() {
    if(window.innerWidth > 900) {
        closeExpandMenu();
    }
}

window.addEventListener('resize', checkScreenWidth);
</script>
```

Explanation: This code will first check if any session starts before, if no, then start $_SESSION. The php script in this code will first retrieve ‘image_data’ and ‘image_type’ from ‘profile_pic’ table based on the $_SESSION[‘username’]. $imgType contains the MIME type of the image. If $imgData is not empty, the image in $imgData is encoded in base64 format using base64_encode() function and store in $base64Image, else, $base64Image will set to the path to default profile pic. For the navigation bar part, there are 3 main divisions under top-nav division, which are nav-left, searchbar and expand-menu. In nav-left, it includes the link to edit profile page, logout, home page, item type page, cart page and contact us page. The $base64Image is used in profile-btn division to show user profile pic. In searchbar, when user enters some words, the form will submit the data by using method get to search_result.php and it will show a list of related items. There is a hidden link which is the hamburger icon will call toggleExpandMenu() function when user click on it. The expand-menu division includes the same options in navigation bar, but it is hidden when the screen width is large. When screen width is getting smaller, the original options in navigation bar except profile-btn division will set to display hidden, and a hamburger icon will be shown at the right end of the navigation bar. Only when the user clicks on navigation bar, it will display the expand-menu. The following is the javascript script which have 4 functions, toggleExpandMenu(), closeExpandMenu(), itemDropdownOnclick() and checkScreenWidth(). Function toggleExpandMenu() will add ‘responsive’ class to the expand-menu division. closeExpandMenu() will set the expand-menu division to ‘expand-menu’ which remove the ‘responsive’ class. ItemDropdownOnclick() will add ‘onClick’ class to referred element, in this case the item-dropdown division inside the expand-menu division. The last function, checkScreenWidth() will check the screen width and if screen width is bigger than 900, it will call closeExpandMenu() function. Window.addEventListener(‘resize’, checkScreenWidth); will always call checkScreenWidth() function when user resize the window.

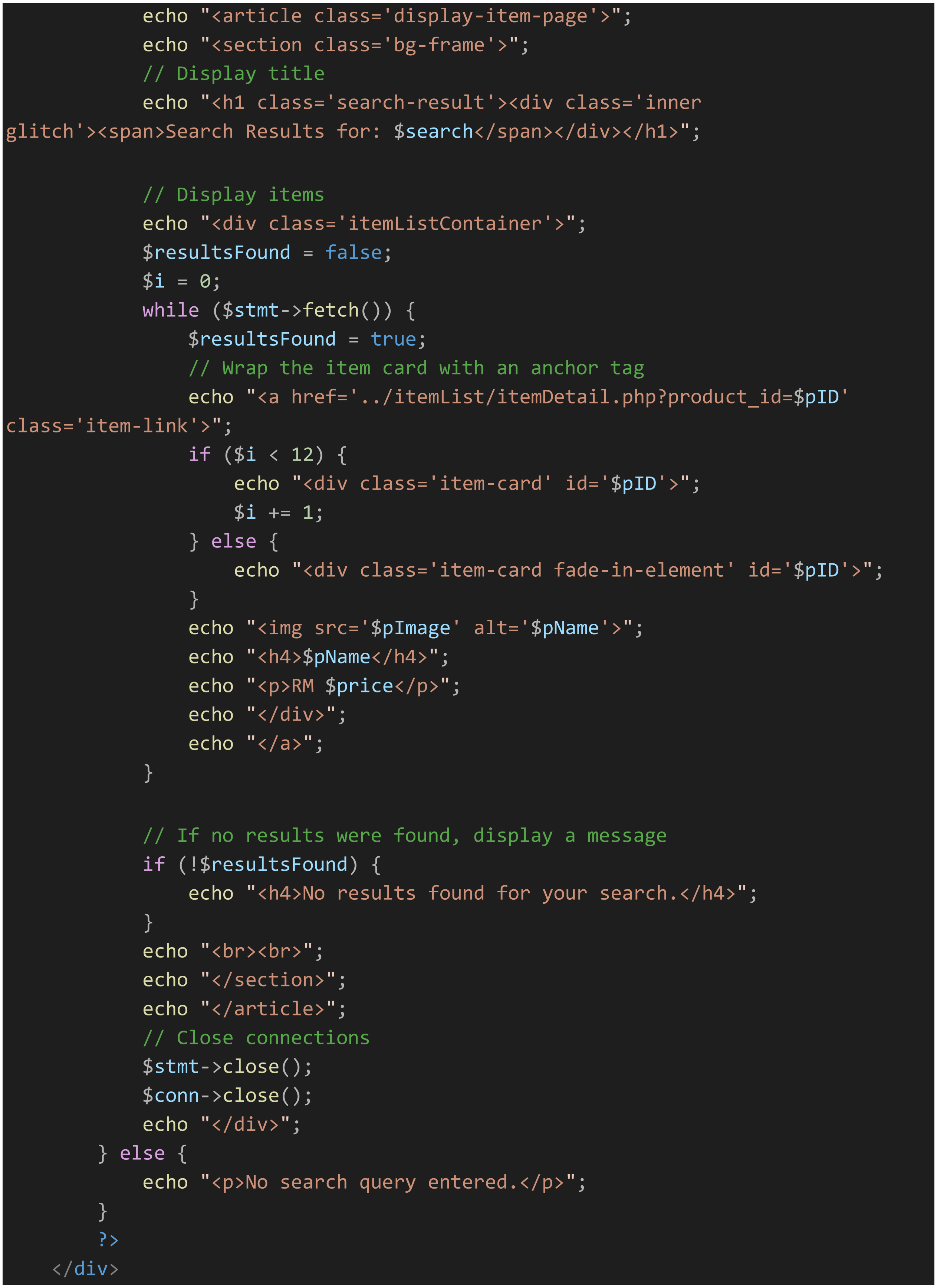
Search bar

Angel_24/search/search_results.php

```
<?php
session_start();
//Check if the user is not logged in
if(!isset($_SESSION['username'])) {
    header('Location: ../index.php');
    exit();
```



```
}
?>

<!DOCTYPE html>
<html>
<head>
    <title>Search Results</title>
    <link rel="stylesheet" href="../style/mystyle1.css">
    <link rel="stylesheet" href="../style/animation.css">
    <link rel="stylesheet" href="../style/displayItem.css">
</head>
<body>

    <!-- Include header and navigation bar -->
    <?php include("../includes/header.php"); ?>
    <?php include("../includes/navigation.php"); ?>

    <div class="content">
        <?php
        if ($_SERVER["REQUEST_METHOD"] == "GET" && isset($_GET['search'])) {
            // Get the search query from the query string and sanitize it
            $search = htmlspecialchars($_GET['search']);

            // Connect to the database
            $conn = mysqli_connect("localhost", "root", "", "angel_24");

            // Check connection
            if (!$conn) {
                die("Connection failed: " . mysqli_connect_error());
            }

            // Prepare the SQL statement with wildcards for a search
            $search_param = "%" . $search . "%";
            $stmt = $conn->prepare("SELECT Product_ID, Product_Name,
Product_Image, Price FROM products WHERE Product_Name LIKE ? OR Product_type
LIKE ?");
            $stmt->bind_param("ss", $search_param, $search_param);

            // Execute the statement
            $stmt->execute();

            // Bind result variables
            $stmt->bind_result($pID, $pName, $pImage, $price);
```



```
            echo "<article class='display-item-page'>";
            echo "<section class='bg-frame'>";
            // Display title
            echo "<h1 class='search-result'><div class='inner
glitch'><span>Search Results for: $search</span></div></h1>";

            // Display items
            echo "<div class='itemListContainer'>";
            $resultsFound = false;
            $i = 0;
            while ($stmt->fetch()) {
                $resultsFound = true;
                // Wrap the item card with an anchor tag
                echo "<a href='../itemList/itemDetail.php?product_id=$pID'
class='item-link'>";
                if ($i < 12) {
                    echo "<div class='item-card' id='$pID'>";
                    $i += 1;
                } else {
                    echo "<div class='item-card fade-in-element' id='$pID'>";
                }
                echo "<img src='$pImage' alt='$pName'>";
                echo "<h4>$pName</h4>";
                echo "<p>RM $price</p>";
                echo "</div>";
                echo "</a>";
            }

            // If no results were found, display a message
            if (!$resultsFound) {
                echo "<h4>No results found for your search.</h4>";
            }
            echo "<br><br>";
            echo "</section>";
            echo "</article>";
            // Close connections
            $stmt->close();
            $conn->close();
            echo "</div>";
        } else {
            echo "<p>No search query entered.</p>";
        }
        ?>
    </div>
    <script src="../javascript/animation.js"></script>
</body>
</html>
```

Explanation :

After the search bar form is submitted, the script starts with a session with “session_start()” and checks if the user is logged in by verifying if there is a username in the session with “$_SESSION[‘username’]. If the user is not logged in, they would be redirected to the login page “Angel24/index.php”. This page implements external CSS stylesheets “mystyle1.css”, “animation.css” and “displayItem.css”, it also includes shared elements such as header and navigation with “include()” functions. The request method is checked if its “GET” and if the query string contains the “product_name”, “htmlspecialchars()” is also used to sanitize the “product_name”. The script then connects to MYSQL database “angel_24” using “mysqli_connect()”, an error message will be displayed if the connection fails. A SQL query is prepared using a wildcard search “LIKE ?” to search for matching products in the “products” table based on the “product_name”, the query uses prepared statements “$stmt->prepare()” to prevent SQL injection attacks. The results are displayed in a structured format, in similar fashion to “itemList\displayItemList.php”. A message will be displayed if there are no matching products. The prepared statement and database connection will then be properly closed using “close()” . “animation.js” is included at the end of the file to bring a more dynamic and engaging user experience.

Contact page

Angel_24/contact/index.php

```
<?php
session_start();
//Check if the user is not logged in
if(!isset($_SESSION['username'])) {
header('Location: ../index.php');
exit();
}
?>


<!DOCTYPE html>
<html>
<head>
<title>Angel 24 - Contact Us</title>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
```

```
<link rel="stylesheet" href="../style/mystyle1.css"> <!-- Link to your external CSS -->
<link rel="stylesheet" href="../style/contactForm.css">
</head>
<body>
<?php include("../includes/header.php"); ?>
<?php include("../includes/navigation.php"); ?>
<div class="contact-page">
<div class="contact-container">
<h1>Contact Us</h1>
<p>If you have any questions or need further assistance, feel free to reach out to us through the form
below.</p>

<div class="contact-info">
<p><i class="fa fa-map-marker"></i> 12 Shopping Lane, E-commerce City, EC 56789</p>
<p><i class="fa fa-phone"></i> +123 456 7890</p>
<p><i class="fa fa-envelope"></i> support@angel24.com</p>
</div>

<form action="submit_contact_form.php" class="contact-form" method="POST">
<label for="name">Name:</label>
<input type="text" id="name" class="name-input" name="name" required>

<label for="email">Email:</label>
<input type="email" id="email" name="email" required>

<label for="message">Message:</label>
<textarea id="message" name="message" rows="6" required></textarea>

<button type="submit" class="submit-btn">Submit</button>
</form>
</div>
</div>
</body>
</html>
```

Explanation: The file includes the header and navigation elements by using include() functions or header.php and navigation.php, ensuring that these elements are consistent across multiple pages. External stylesheets are included for consistent styling (mystyle1.css), ensuring that the visual layout is responsive and engaging for users. This page asks the user to fill in the contact form including the Name and email address, after clicking submit, the submit_contact_form.php will start running the backend process.

Angel_24/contact/submit_contact_form.php

```php
<?php
// Database connection
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "angel_24";

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection
if ($conn->connect_error) {
die("Connection failed: " . $conn->connect_error);
}

if ($_SERVER["REQUEST_METHOD"] == "POST") {
$name = $conn->real_escape_string($_POST['name']);
$email = $conn->real_escape_string($_POST['email']);
$message = $conn->real_escape_string($_POST['message']);

$sql = "INSERT INTO contact_messages (name, email, message) VALUES ('$name', '$email', '$message')";

if ($conn->query($sql) === TRUE) {
// Redirect to the thank you page after successful form submission
header('Location: thank_you.php');
exit();
} else {
echo "Error: " . $sql . "<br>" . $conn->error;
}
}


$conn->close();
?>
```

Explanation: Once the user submits the form, submit_contact_form.php handles the backend processing. When the form is submitted, the $_POST method captures the input fields: name, email, and message. These inputs are sanitized using the htmlspecialchars() function to prevent malicious code injection. The script then connect to MySQL database angel_24 using mysqli_connect(). If the connection is successful, it prepares an SQL query to insert the sanitized contact information into the contact_messages table. Prepared statements ($stmt->prepare()) are used to prevent SQL injection by binding the parameters securely and inserting the values into table. Upon successful form submission, the user is redirected to the thank_you.php page using the header() function. If any errors occur, a failure message is displayed.

Angel_24/contact/Thank_you.php

```
<!DOCTYPE html>
<html>
<head>
<title>Thank You!</title>
<link rel="stylesheet" href="../style/mystyle1.css"> <!-- Link to your stylesheet -->
<style>
/* Add custom styles here or in your mystyle1.css file */

body {
font-family: Arial, sans-serif;
margin: 0;
padding: 0;
background-color: #f4f4f4;
}

.thank-you-page {
padding-top: 50px;
```

```css
min-height: calc(100vh - 50px);
background-color: rgba(239, 239, 240, 0.5);
}

.thank-you-container {
width: 50%;
margin: 0px auto;
padding: 20px;
background-color: #fff;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
border-radius: 8px;
text-align: center;
}

h1 {
color: #333;
}

p {
color: #666;
margin-bottom: 30px;
}

.return-link {
display: inline-block;
padding: 10px 20px;
background-color: #007BFF;
color: white;
text-decoration: none;
border-radius: 5px;
transition: background-color 0.3s ease;
}

.return-link:hover {
background-color: #0056b3;
}

@media (max-width: 768px) {
.thank-you-container {
```

```
width: 80%;
}
}
</style>
</head>
<body>
<?php include("../includes/header.php"); ?>
<?php include("../includes/navigation.php"); ?>
<div class="thank-you-page">
<div class="thank-you-container">
<h1>Thank You for Your Message!</h1>
<p>We appreciate you contacting us. One of our customer service colleagues will get back in touch with
you soon! Have a great day!</p>
<a href="../home/index.php" class="return-link">Return to Home</a>
</div>
</div>
</body>
</html>
```

Explanation: It includes external styles(mystyle1.css) for consistent visual styling. The layout includes a message thanking the user for contacting customer serviice, and it mentions that someone will be in touch with them soon.The page contains a simple, visually appealing container (thank-you-container) to display the message and a button(return-link) that redirects users back to the homepage(../home/index.php). The button has hover effects for better user interaction, with styles managed through CSS.

Angel_24/contact/admin.php

```
<?php
// Database connection
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "angel_24";

$conn = new mysqli($servername, $username, $password, $dbname);
```

```
// Check connection
if ($conn->connect_error) {
die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT id, name, email, message, submitted_at FROM contact_messages ORDER BY submitted_at DESC";
$result = $conn->query($sql);
?>

<!DOCTYPE html>
<html>
<head>
<title>Admin - View Messages</title>
<link rel="stylesheet" href="../style/adminstyle.css"> <!-- Link to your admin stylesheet -->
<style>
/* Add custom styles here or in your adminstyle.css file */

body {
font-family: Arial, sans-serif;
margin: 0;
padding: 20px;
background-color: #f4f4f4;
}

h1 {
text-align: center;
color: #333;
margin-bottom: 20px;
}

table {
width: 100%;
border-collapse: collapse;
margin: 0 auto;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
background-color: #fff;
border-radius: 8px;
overflow: hidden;
}

th, td {
padding: 10px;
text-align: left;
border-bottom: 1px solid #ddd;
}

th {
background-color: #007BFF;
color: white;
}

tr:nth-child(even) {
background-color: #f9f9f9;
}

tr:hover {
background-color: #f1f1f1;
}

p {
text-align: center;
color: #666;
}

</style>
</head>
<body>
<h1>Contact Form Submissions</h1>
<?php if ($result->num_rows > 0): ?>
<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Email</th>
<th>Message</th>
```

```
<th>Date Submitted</th>
</tr>
</thead>
<tbody>
<?php while($row = $result->fetch_assoc()): ?>
<tr>
<td><?php echo $row["id"]; ?></td>
<td><?php echo $row["name"]; ?></td>
<td><?php echo $row["email"]; ?></td>
<td><?php echo $row["message"]; ?></td>
<td><?php echo $row["submitted_at"]; ?></td>
</tr>
<?php endwhile; ?>
</tbody>
</table>
<?php else: ?>
<p>No messages found.</p>
<?php endif; ?>

<?php $conn->close(); ?>
</body>
</html>
```

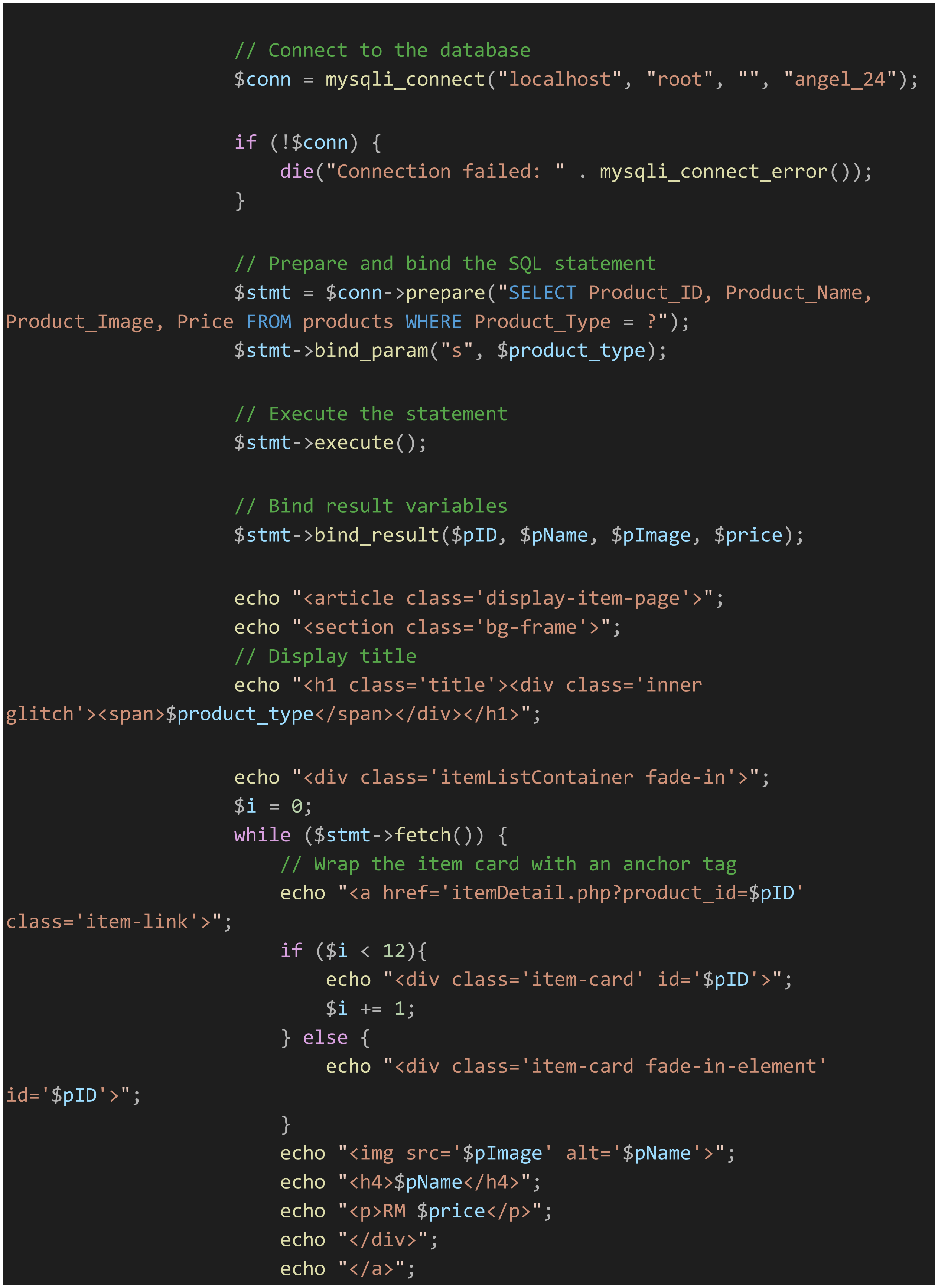
Explanation: It starts by connecting to the MySQL database angel_24 using the mysqli_connect() function. The connection parameters, such as server name , username , and password,are defined , and error message is displayed if the connection fails using $conn->connnect error. A SQL query is prepared to retrieve the ID, name , email, message and submissions, ordered by submission date in descending order.This is achieved using SQL SELECT query and stored in the $result variable.If there are results ($result->num_rows >0), the submissions are displayed in a structured HTML table , where each row corresponds to a submission.The table , where each row corresponds to a submission.The table and tr elements are styled with CSS to improve readability and aesthetics.The file also uses an external stylesheet(adminstyle.css) to manage additional

Styles and include inline styles for the layout of the page. If there are no submissions (else), a message is displayed saying, “No messages found.”The MySQL connection is closed using $conn->close().

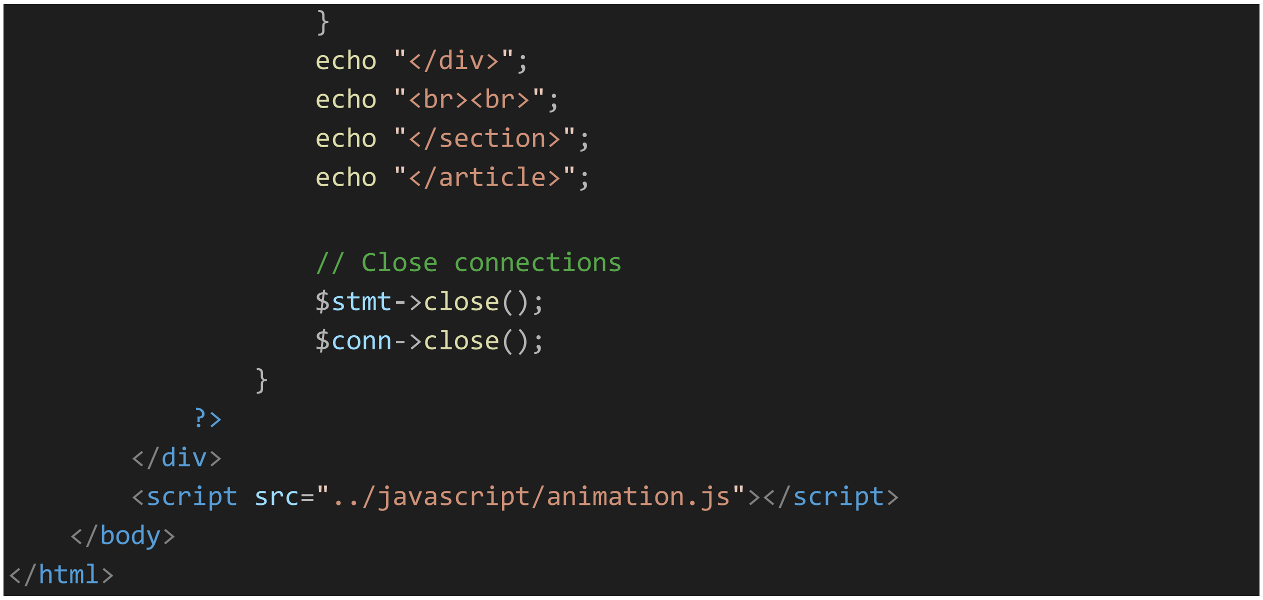
Item list by category

Angel_24/itemList/displayItemList.php

```
<?php
session_start();
//Check if the user is not logged in
if(!isset($_SESSION['username'])) {
    header('Location: ../index.php');
    exit();
}
?>

<!DOCTYPE html>
<html>
    <head>
        <title>Item List</title>
        <link rel="stylesheet" href="../style/mystyle1.css">
        <link rel="stylesheet" href="../style/animation.css">
        <link rel="stylesheet" href="../style/displayItem.css">
    </head>
    <body>

        <?php include("../includes/header.php"); ?>
        <?php include("../includes/navigation.php"); ?>

        <div>
            <?php
                if ($_SERVER["REQUEST_METHOD"] == "GET" &&
isset($_GET['product_type'])) {
                    // Get the product type from the query string
                    $product_type = $_GET['product_type'];

                    // Ensure the variable is safe to use
                    $product_type = htmlspecialchars($product_type);
```



```
                    // Connect to the database
                    $conn = mysqli_connect("localhost", "root", "", "angel_24");

                    if (!$conn) {
                        die("Connection failed: " . mysqli_connect_error());
                    }

                    // Prepare and bind the SQL statement
                    $stmt = $conn->prepare("SELECT Product_ID, Product_Name,
Product_Image, Price FROM products WHERE Product_Type = ?");
                    $stmt->bind_param("s", $product_type);

                    // Execute the statement
                    $stmt->execute();

                    // Bind result variables
                    $stmt->bind_result($pID, $pName, $pImage, $price);

                    echo "<article class='display-item-page'>";
                    echo "<section class='bg-frame'>";
                    // Display title
                    echo "<h1 class='title'><div class='inner
glitch'><span>$product_type</span></div></h1>";

                    echo "<div class='itemListContainer fade-in'>";
                    $i = 0;
                    while ($stmt->fetch()) {
                        // Wrap the item card with an anchor tag
                        echo "<a href='itemDetail.php?product_id=$pID'
class='item-link'>";
                        if ($i < 12){
                            echo "<div class='item-card' id='$pID'>";
                            $i += 1;
                        } else {
                            echo "<div class='item-card fade-in-element'
id='$pID'>";
                        }
                        echo "<img src='$pImage' alt='$pName'>";
                        echo "<h4>$pName</h4>";
                        echo "<p>RM $price</p>";
                        echo "</div>";
                        echo "</a>";
```



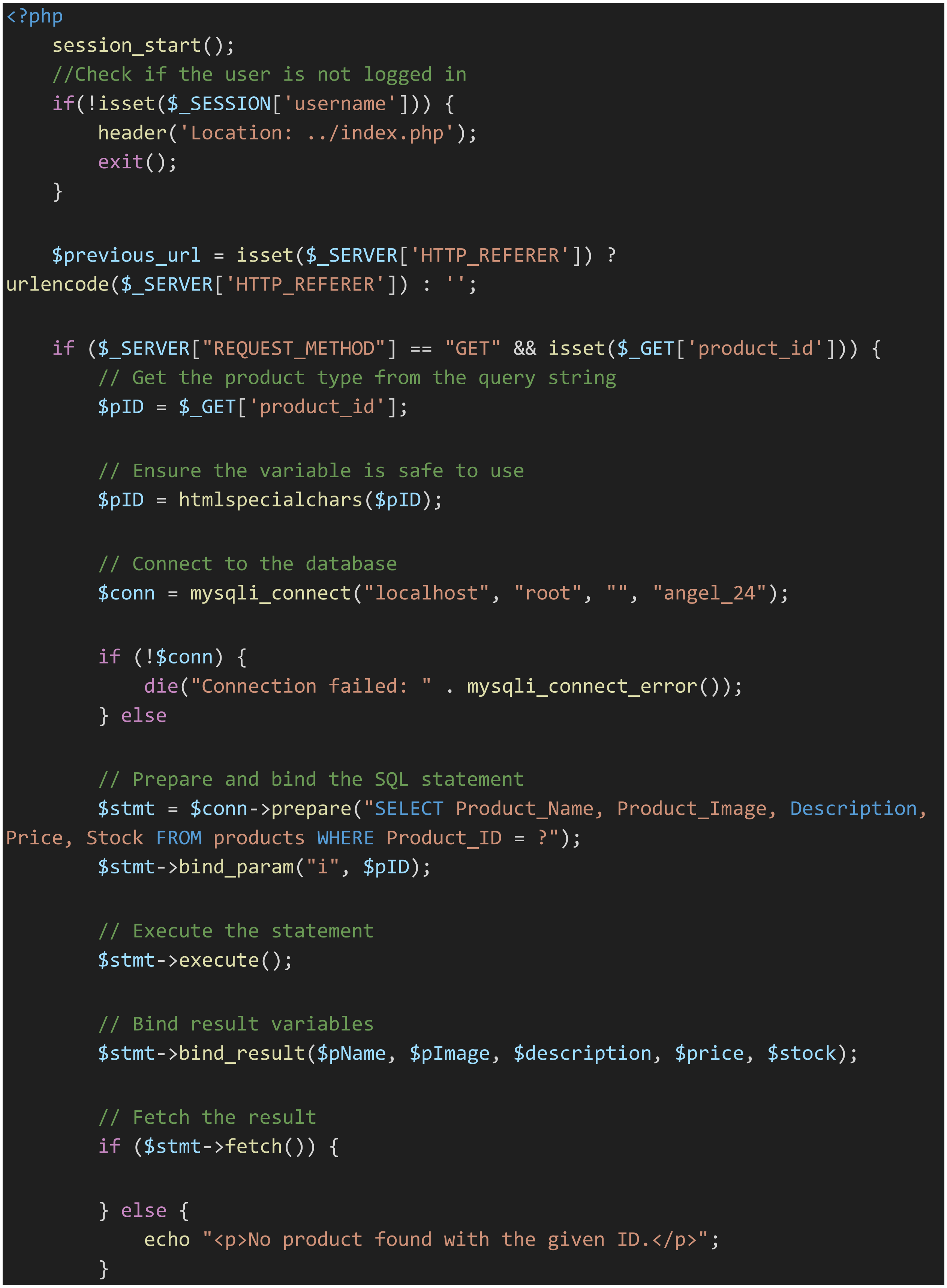
```
                    }
                    echo "</div>";
                    echo "<br><br>";
                    echo "</section>";
                    echo "</article>";

                    // Close connections
                    $stmt->close();
                    $conn->close();
                }
            ?>
        </div>
        <script src="../javascript/animation.js"></script>
    </body>
</html>
```

Explanation :

After the user clicks on one of the categories in the items' dropdown list, the script starts with a session with "session_start()" and checks if the user is logged in by verifying if there is a username in the session with "$_SESSION['username']. If the user is not logged in, they would be redirected to the login page "Angel24/index.php". This page implements external CSS stylesheets "mystyle1.css", "animation.css" and "displayItem.css", it also includes shared elements such as header and navigation with "include()" functions. The request method is checked if its "GET" and if the query string contains the "product_type", "htmlspecialchars()" is also used to sanitize the "product_type" The script then connects to MYSQL database "angel_24" using "mysqli_connect()", an error message will be displayed if the connection fails. A SQL query is prepared using a wildcard search "LIKE ?" to search for matching products in the "products" table based on the "product_type", the query uses prepared statements "$stmt->prepare()" to prevent SQL injection attacks. The results are displayed in a structured format, each product is wrapped in a link that points to the details page "itemDetail.php?product_id=$pID" and shown in an item card with its image, name and price. The first 12 products are displayed normally; anything beyond that is given the class ".fade_in_element" defined by "animation.js" and "animation.css". The prepared statement and database connection will then be properly closed using "close()" . "animation.js" is included at the end of the file to bring a more dynamic and engaging user experience.

Item details

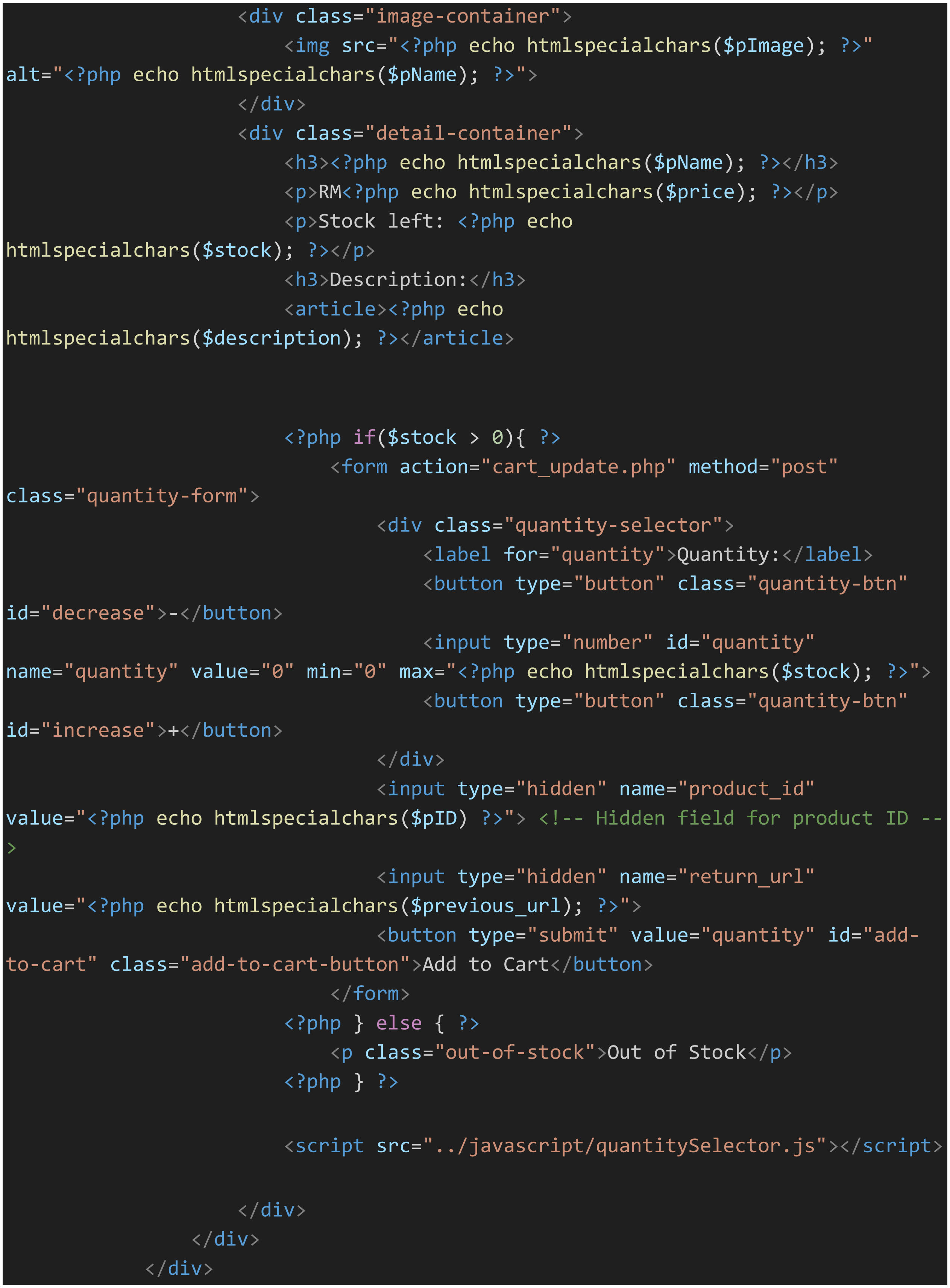
Angel_24/itemList/itemDetail.php



```
<?php
    session_start();
    //Check if the user is not logged in
    if(!isset($_SESSION['username'])) {
        header('Location: ../index.php');
        exit();
    }

    $previous_url = isset($_SERVER['HTTP_REFERER']) ?
urlencode($_SERVER['HTTP_REFERER']) : '';

    if ($_SERVER["REQUEST_METHOD"] == "GET" && isset($_GET['product_id'])) {
        // Get the product type from the query string
        $pID = $_GET['product_id'];

        // Ensure the variable is safe to use
        $pID = htmlspecialchars($pID);

        // Connect to the database
        $conn = mysqli_connect("localhost", "root", "", "angel_24");

        if (!$conn) {
            die("Connection failed: " . mysqli_connect_error());
        } else

        // Prepare and bind the SQL statement
        $stmt = $conn->prepare("SELECT Product_Name, Product_Image, Description,
Price, Stock FROM products WHERE Product_ID = ?");
        $stmt->bind_param("i", $pID);

        // Execute the statement
        $stmt->execute();

        // Bind result variables
        $stmt->bind_result($pName, $pImage, $description, $price, $stock);

        // Fetch the result
        if ($stmt->fetch()) {

        } else {
            echo "<p>No product found with the given ID.</p>";
        }
        $stmt -> close();
        $conn -> close();


    }
?>
```

Explanation: This php first check if user is login. If user is not login yet, it redirects user to login page. $_SERVER['HTTP_REFERER'] is a global array in php that hold the previous page URL. By using urlencode() function, $previous_url stored the link of user previous page if exists. If POST is requested and product_id is retrieved by GET method, $pID is set as product id and sanitized. Database is then connected with variables “localhost”, “root”, “” and “angel_24” and error message is displayed if connection error occurs. A SQL statement is then prepared and executed to retrieve selected product information. Another SQL statement is prepared and executed to assign product information to corresponding variables. If the product id is not found in database, error message is displayed. SQL statement and connection to database is then closed as best practice.

```
<!DOCTYPE html>
<html>
    <head>
        <title><?php echo htmlspecialchars($pName); ?></title>
        <link rel="stylesheet" href="../style/mystyle1.css">
        <link rel="stylesheet" href="../style/itemDetail.css">
        <link rel="stylesheet" href="../style/previousButton.css">
        <link rel="stylesheet" href="../style/quantitySelector.css">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <style>
        </style>
        </head>
    <body>
        <?php include("../includes/header.php"); ?>
        <?php include("../includes/navigation.php"); ?>
        <div class="item-detail-page">
            <button onclick="history.back()" class="previous-button"><i class="fa
fa-arrow-left"></i></button>
            <div class="item-detail-title"><span>Item Details</span></div>
            <div class="item-description-container">
                <div class="image-detail-container">
```



```
                    <div class="image-container">
                        <img src="<?php echo htmlspecialchars($pImage); ?>"
alt="<?php echo htmlspecialchars($pName); ?>">
                    </div>
                    <div class="detail-container">
                        <h3><?php echo htmlspecialchars($pName); ?></h3>
                        <p>RM<?php echo htmlspecialchars($price); ?></p>
                        <p>Stock left: <?php echo
htmlspecialchars($stock); ?></p>
                        <h3>Description:</h3>
                        <article><?php echo
htmlspecialchars($description); ?></article>



                        <?php if($stock > 0){ ?>
                            <form action="cart_update.php" method="post"
class="quantity-form">
                                <div class="quantity-selector">
                                    <label for="quantity">Quantity:</label>
                                    <button type="button" class="quantity-btn"
id="decrease">-</button>
                                    <input type="number" id="quantity"
name="quantity" value="0" min="0" max="<?php echo htmlspecialchars($stock); ?>">
                                    <button type="button" class="quantity-btn"
id="increase">+</button>
                                </div>
                                <input type="hidden" name="product_id"
value="<?php echo htmlspecialchars($pID) ?>"> <!-- Hidden field for product ID --
>
                                <input type="hidden" name="return_url"
value="<?php echo htmlspecialchars($previous_url); ?>">
                                <button type="submit" value="quantity" id="add-
to-cart" class="add-to-cart-button">Add to Cart</button>
                            </form>
                        <?php } else { ?>
                            <p class="out-of-stock">Out of Stock</p>
                        <?php } ?>

                        <script src="../javascript/quantitySelector.js"></script>

                    </div>
                </div>
            </div>
```

```
        </div>
    </body>
</html>
```

Explanation: All necessary css, header and navigation bar are included. The product name is used as title. A button is created and placed at top left corner below navigation bar which allows user redirects to previous page.

Inside image container, product image is displayed with alternative of product name if image cannot display. Inside detail container, product information such as name, price, stock number and description are shown. If the product is still in stock, quantity-form that executes cart.update.php on action is displayed. The form contains 2 buttons that allow user to decrease or increase the quantity of products to purchase, user can also enter the desired quantity directly in quantity number field. If the selected product is out of stock, a message of “Out of Stock” is displayed instead of form. QuantitySelector.js is included.

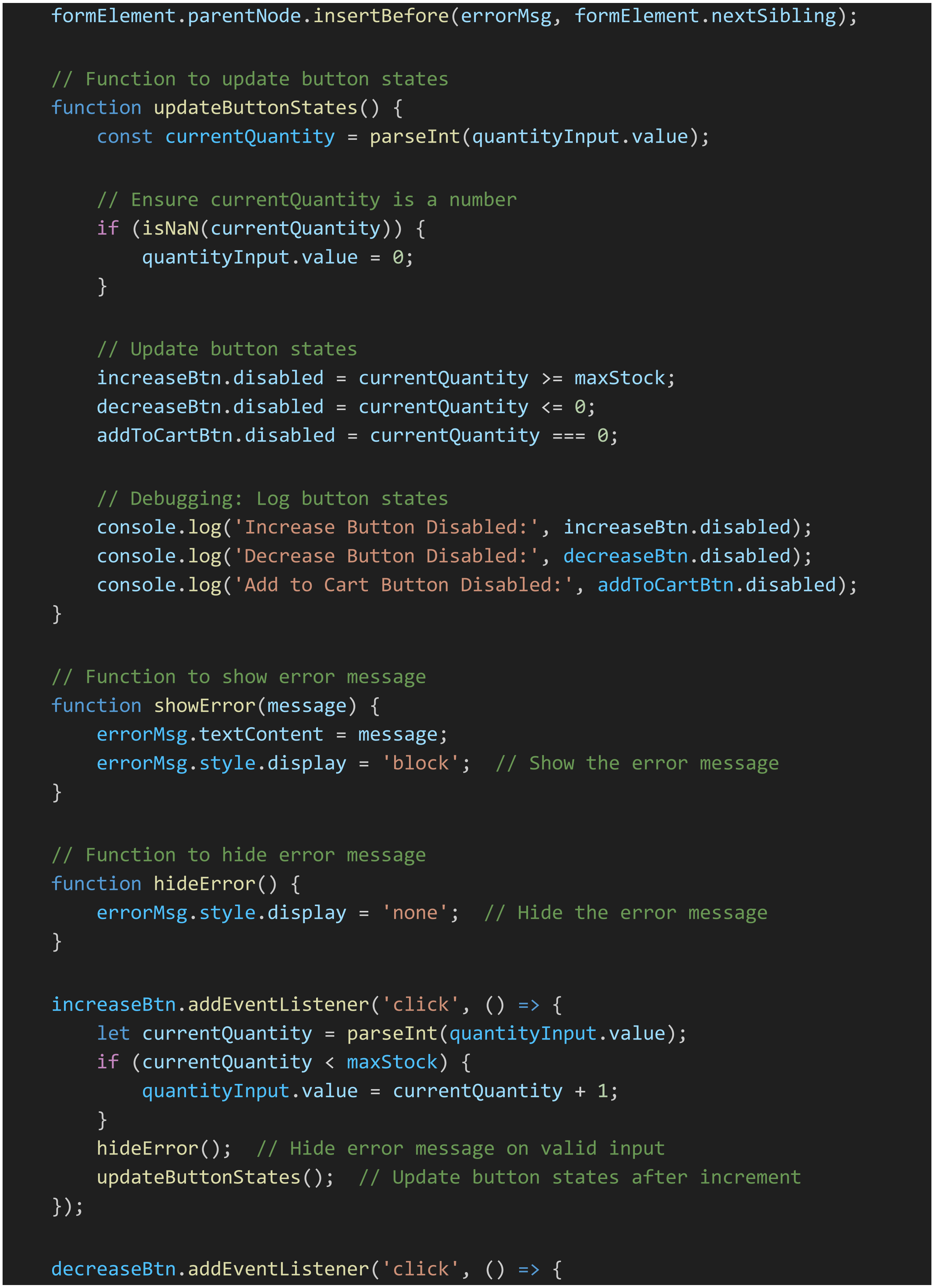
Angel_24/itemList/cart_update.php

```
<?php
session_start(); // Start the session

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Retrieve the form data
    $product_id = isset($_POST['product_id']) ?
htmlspecialchars($_POST['product_id']) : '';
    $quantity = isset($_POST['quantity']) ? intval($_POST['quantity']) : 0;


    // Validate the quantity
    if ($quantity < 0) {
        $quantity = 0; // Ensure the quantity is non-negative
    }

    // Initialize the cart in session if it does not exist
    if (!isset($_SESSION['cart'])) {
        $_SESSION['cart'] = [];
    }

    // Check if the product is already in the session cart
    if (isset($_SESSION['cart'][$product_id])) {
        // If it exists, add the new quantity to the existing quantity
        $_SESSION['cart'][$product_id] += $quantity;
    } else {
        // If it doesn't exist, add it to the cart with the new quantity
        $_SESSION['cart'][$product_id] = $quantity;
    }

    // No database actions are needed as we're not storing the cart in the
database

} else {
    echo "Invalid request.";
}

// Back to return URL
$return_url = isset($_POST["return_url"]) ? urldecode($_POST["return_url"]) : '';
// Return URL
header('Location: ' . $return_url);
?>
```

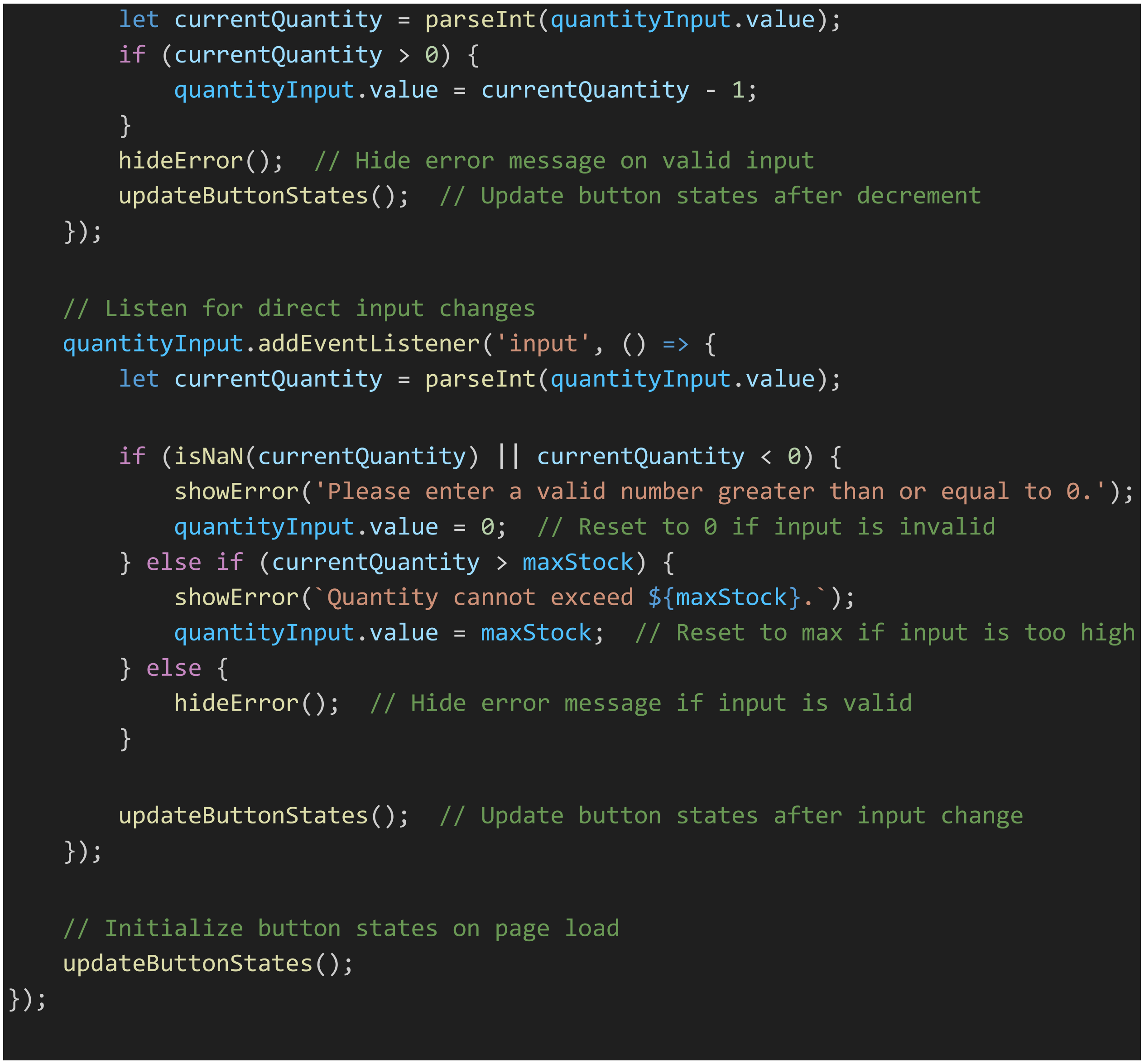
Explanation: Once POST method is requested (submit button in quantity-form), this php script retrieves the product id and the quantity to purchase. It set the quantity to 0 if it is negative value and initialize cart to empty array if cart doesn't exist. Then, quantity of selected product in cart increases if exists. Else, product_id with its quantity is added into cart. If POST method is not requested, it displays error message. Lastly, user is redirected to previous page.

Angel_24/javascript/quantitySelector.js

```
document.addEventListener('DOMContentLoaded', () => {
    const quantityInput = document.getElementById('quantity');
    const increaseBtn = document.getElementById('increase');
    const decreaseBtn = document.getElementById('decrease');
    const addToCartBtn = document.getElementById('add-to-cart');
    const formElement = document.querySelector('.quantity-form');

    // Ensure the quantity input has a max attribute and parse it
    const maxStock = parseInt(quantityInput.getAttribute('max'));

    // Create an element to show error messages
    const errorMsg = document.createElement('p');
    errorMsg.className = 'error-message';
    errorMsg.style.color = 'red';  // Make error message red for visibility
    errorMsg.style.display = 'none';  // Hide error message by default
```



```
    formElement.parentNode.insertBefore(errorMsg, formElement.nextSibling);

    // Function to update button states
    function updateButtonStates() {
        const currentQuantity = parseInt(quantityInput.value);

        // Ensure currentQuantity is a number
        if (isNaN(currentQuantity)) {
            quantityInput.value = 0;
        }

        // Update button states
        increaseBtn.disabled = currentQuantity >= maxStock;
        decreaseBtn.disabled = currentQuantity <= 0;
        addToCartBtn.disabled = currentQuantity === 0;

        // Debugging: Log button states
        console.log('Increase Button Disabled:', increaseBtn.disabled);
        console.log('Decrease Button Disabled:', decreaseBtn.disabled);
        console.log('Add to Cart Button Disabled:', addToCartBtn.disabled);
    }

    // Function to show error message
    function showError(message) {
        errorMsg.textContent = message;
        errorMsg.style.display = 'block';  // Show the error message
    }

    // Function to hide error message
    function hideError() {
        errorMsg.style.display = 'none';  // Hide the error message
    }

    increaseBtn.addEventListener('click', () => {
        let currentQuantity = parseInt(quantityInput.value);
        if (currentQuantity < maxStock) {
            quantityInput.value = currentQuantity + 1;
        }
        hideError();  // Hide error message on valid input
        updateButtonStates();  // Update button states after increment
    });

    decreaseBtn.addEventListener('click', () => {
```



```
        let currentQuantity = parseInt(quantityInput.value);
        if (currentQuantity > 0) {
            quantityInput.value = currentQuantity - 1;
        }
        hideError();  // Hide error message on valid input
        updateButtonStates();  // Update button states after decrement
    });

    // Listen for direct input changes
    quantityInput.addEventListener('input', () => {
        let currentQuantity = parseInt(quantityInput.value);

        if (isNaN(currentQuantity) || currentQuantity < 0) {
            showError('Please enter a valid number greater than or equal to 0.');
            quantityInput.value = 0;  // Reset to 0 if input is invalid
        } else if (currentQuantity > maxStock) {
            showError(`Quantity cannot exceed ${maxStock}.`);
            quantityInput.value = maxStock;  // Reset to max if input is too high
        } else {
            hideError();  // Hide error message if input is valid
        }

        updateButtonStates();  // Update button states after input change
    });

    // Initialize button states on page load
    updateButtonStates();
});
```

Explanation: This js script first set a maximum number for quantity input. Then it creates an element to display error message and set its color to red and hidden by default.

In updateButtonStates, it retrieves the quantity input value and set the value to 0 if it is not an number. IsNaN() is a method that returns false when number passed and return true when other data types are passed. Then, it disables increase button when input quantity value larger than or equals to max, disables decrease button when input quantity smaller than or equals to 0 and disables add to cart button when input quantity value equals to 0.

In showError, it assign message passed to error message and display error message.

In hideError, the error message is set to hidden.

IncreaseBtn is added an event listener that allows it to increase current quantity by one if current quantity doesn’t exceed max quantity, hide error message and update its states on click.

decreaseBtn is added an event listener that allows it to decrease current quantity by one if current quantity larger than 0, hide error message and update its states on click.

QuantityInput is added an event listener that when value input, it validates if it is a number and if input value is negative value. If input data is not a number or is a negative value, display error message and set currentQuantity back to 0. If value input larger than max, error message is displayed and currentQuantity is set to maximum quantity. If no error occurs, it sets error message to hidden. Lastly, it updates 2 buttons’ state.

When page onload, this js script execute updateButtonStates function.

Cart page

Angel_24/cart/index.php

```
<?php
session_start();
//Check if the user is not logged in
if(!isset($_SESSION['username'])) {
    header('Location: ../index.php');
    exit();
}
?>

<!DOCTYPE html>
<html>
<head>
    <title>Cart</title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
    <link rel="stylesheet" href="../style/mystyle1.css">
    <link rel="stylesheet" href="../style/displayItem.css">
    <link rel="stylesheet" href="../style/cart.css">
    <link rel="stylesheet" href="../style/animation.css">
    <link rel="stylesheet" href="../style/quantitySelector.css">
</head>
<body>
    <?php include("../includes/header.php"); ?>
    <?php include("../includes/navigation.php"); ?>
    <article class="cart-page">
        <section class="bg-frame">
        <h1 class="title"><div class="inner glitch"><span>Cart</span></div></h1>

        <div class='cart-item'>
            <?php include("list_cart.php"); ?>
        </div>
        </section>
    </article>

</body>
</html>
```

Explanation: this is the very first page of cart page. At beginning, it check if user is login. If no, user is redirected to login page. Else, all necessary css, header and navigation bar are included. Inside cart-item section, list_cart.php is included.

Angel_24/cart/list_cart.php

```
<?php
    if (session_status() === PHP_SESSION_NONE) {
        session_start();
    }
    $current_url = urlencode($url = "http://" . $_SERVER['HTTP_HOST'] .
$_SERVER['REQUEST_URI']);

    // Connect to the database
    $conn = mysqli_connect("localhost", "root", "", "angel_24");
    $no_quantity = true;

    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());
    }

    // Check if the cart session variable exists
    if (isset($_SESSION['cart']) && !empty($_SESSION['cart'])) {
        // Iterate over the cart items
        foreach ($_SESSION['cart'] as $pID => $quantity) {
            if ($quantity > 0) {
                $no_quantity = false;
                // Fetch item details from the database
                $stmt = $conn->prepare("SELECT Product_Name, Product_Image,
Price, Stock FROM products WHERE Product_ID = ?");
                $stmt->bind_param("i", $pID);
                $stmt->execute();
                $stmt->bind_result($pName, $pImage, $price, $stock);
                $stmt->fetch();
                $stmt->close();

                // Display item details
                echo "<div class='product-row'>";
                echo "<img src='" . htmlspecialchars($pImage) . "' alt='" .
htmlspecialchars($pName) . "' >";
                echo "<div class='name-price-container'>";
                echo "<p>" . htmlspecialchars($pName) . "</p>";
                echo "<p>RM " . htmlspecialchars($price) . "</p>";
                echo "</div>";

                // Unique IDs for each form and element
                $quantityInputId = "quantity_" . htmlspecialchars($pID);

                // Quantity form with JavaScript to auto-submit
                echo '<form action="quantity_update.php" method="post"
name="form_' . htmlspecialchars($pID) . '" class="quantity-form">';
                echo '<label for="' . $quantityInputId . '">Quantity:</label>';
                echo '<div class="quantity-selector">';
                echo '<button type="button" class="quantity-btn"
onclick="adjustQuantity(' . htmlspecialchars($pID) . ', -1)">-</button>';
                echo '<input type="number" id="' . $quantityInputId . '"
name="quantity" value="' . htmlspecialchars($quantity) . '" min="0" max="' .
htmlspecialchars($stock) . '" onchange="this.form.submit()">';
                echo '<button type="button" class="quantity-btn"
onclick="adjustQuantity(' . htmlspecialchars($pID) . ', 1)">+</button>';
                echo '</div>';
                echo '<input type="hidden" name="product_id" value="' .
htmlspecialchars($pID) . '">';
                echo '<input type="hidden" name="return_url" value="' .
$current_url . '">';
                echo '</form>';
                echo "</div>";
            }
        }
        if (!$no_quantity) {
            echo '<div class="checkout-container">';
            echo '<a href="checkout.php" class="checkout-button">Proceed to
Checkout</a>';
            echo '</div>';
        }
    }
    if ($no_quantity) {
        echo "<div class='cart-empty-container'>";
        echo "<p>Your cart is empty.</p>";
        echo "<img src='../images/sad_momoi.png'>";
        echo "</div>";
    }

    // Close the database connection
    $conn->close();
?>
```

Explanation: At beginning. php script here start session if session haven't started. It then save current page url and connect to database. Script is terminated and error message is displayed if error occurs connected to databse. If cart session variables exists, product details are fetch from database, assigned to variables and displayed. Then it give every form and element a unique ID, and displaying form contents. If there's product in cart, a checkout button is shown and it allows user to proceed to payment. If there's no product in cart, a picture and message are shown to tell user cart is empty.

```
<script>
function adjustQuantity(productId, change) {
    let input = document.getElementById('quantity_' + productId);
    let currentQuantity = parseInt(input.value) || 0;
    let newQuantity = currentQuantity + change;
    let maxQuantity = parseInt(input.max) || Infinity;

    if (newQuantity >= 0 && newQuantity <= maxQuantity) {
        input.value = newQuantity;
        input.dispatchEvent(new Event('change')); // Trigger change event to
submit the form
    }
}
</script>
```

Explanation: js script here adjust the quantity of product based on parameters passed if the updated newQuantity is in valid range (>=0 && <=maxQuantity). If it does, assign newQuantity to input.value and trigger change event to submit the form.

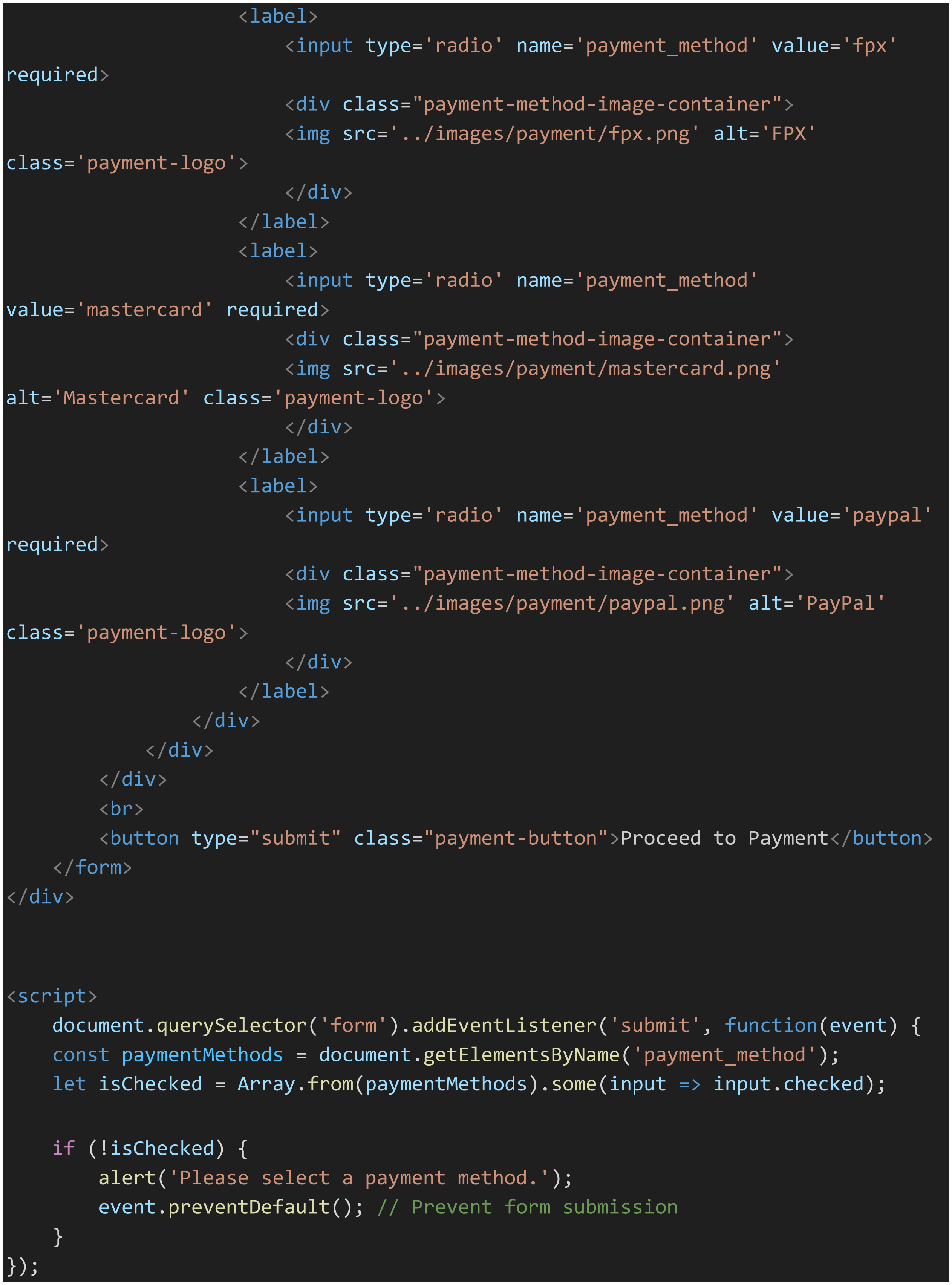
Angel_24/cart/checkout.php

```
<?php
session_start();
//Check if the user is not logged in
if(!isset($_SESSION['username'])) {
    header('Location: ../index.php');
    exit();
}
?>

<!DOCTYPE html>
<html>
    <head>
        <title>Check Out</title>
        <link rel="stylesheet" href="../style/mystyle1.css">
        <link rel="stylesheet" href="../style/previousButton.css">
        <link rel="stylesheet" href="../style/animation.css">
    </head>
    <body>
        <?php include("../includes/header.php"); ?>
        <?php include("../includes/navigation.php"); ?>

        <article class="checkout-page">
            <section class="bg-frame">
                <button onclick="history.back()" class="previous-button"><i
class="fa fa-arrow-left"></i></button>
                <h1 class="title"><div class="inner glitch"><span>Check
Out</span></div></h1>
            <?php include("shippingForm.php"); ?>
            </section>
        </article>

    </body>
</html>
```

Explanation: this php file check if user is login. If no, it redirects user to login page. Else, it includes all necessary css, header and navigation bar. It then include shippingForm inside

Angel_24/cart/shippingForm.php

```
<link rel="stylesheet" href="../style/form.css">

<div class="shipping-form-container">
    <form class="shipping-form" action="paymentSuccessful.php" method="post" onsubmit="return validatePaymentMethod();">
        <div class="shipping-form-grid">
            <input type="text" placeholder="First Name" name="first_name" class="two-column" required>
            <input type="text" placeholder="Last Name" name="last_name" class="two-column" required>
        </div>
        <input type="text" placeholder="Street Address" name="street" required><br>
        <input type="text" placeholder="Apt / Suite / Unit (Optional)" name="unit"><br>
        <div class="shipping-form-grid">
            <input type="text" placeholder="City" name="city" class="two-column" required>
            <input type="text" placeholder="State" name="state" class="two-column" required>
        </div>
        <input type="text" placeholder="Postcode" name="postcode" required>
        <?php include("list_checkout_item.php"); ?>

        <!-- Add payment method selection -->
        <div class='payment-method-container'>
            <div class='payment-options'>
            <h4>Select Payment Method:</h4>
                <div>
                    <label>
                        <input type='radio' name='payment_method' value='tng' required>
                        <div class="payment-method-image-container">
                            <img src='../images/payment/tng.png' alt='tng' class='payment-logo'>
                        </div>
                    </label>
```



```
                    <label>
                        <input type='radio' name='payment_method' value='fpx'
required>
                        <div class="payment-method-image-container">
                        <img src='../images/payment/fpx.png' alt='FPX'
class='payment-logo'>
                        </div>
                    </label>
                    <label>
                        <input type='radio' name='payment_method'
value='mastercard' required>
                        <div class="payment-method-image-container">
                        <img src='../images/payment/mastercard.png'
alt='Mastercard' class='payment-logo'>
                        </div>
                    </label>
                    <label>
                        <input type='radio' name='payment_method' value='paypal'
required>
                        <div class="payment-method-image-container">
                        <img src='../images/payment/paypal.png' alt='PayPal'
class='payment-logo'>
                        </div>
                    </label>
                </div>
            </div>
        </div>
        <br>
        <button type="submit" class="payment-button">Proceed to Payment</button>
    </form>
</div>



<script>
    document.querySelector('form').addEventListener('submit', function(event) {
    const paymentMethods = document.getElementsByName('payment_method');
    let isChecked = Array.from(paymentMethods).some(input => input.checked);

    if (!isChecked) {
        alert('Please select a payment method.');
        event.preventDefault(); // Prevent form submission
    }
});
```

```

</script>
```

Explanation: When user proceeds to check out, the shipping form is displayed, and user must fill in all data field except for “unit” field and choose their payment method before they can check out. list_checkout_item.php is included to display the item selected by user.The contents of shipping form grid allows user to input data while contents of payment-method-container display 4 radio buttons representing 4 payment methods. If all required data are filled and payment method is selected, click on payment-button redirects user to paymentSuccessful.php

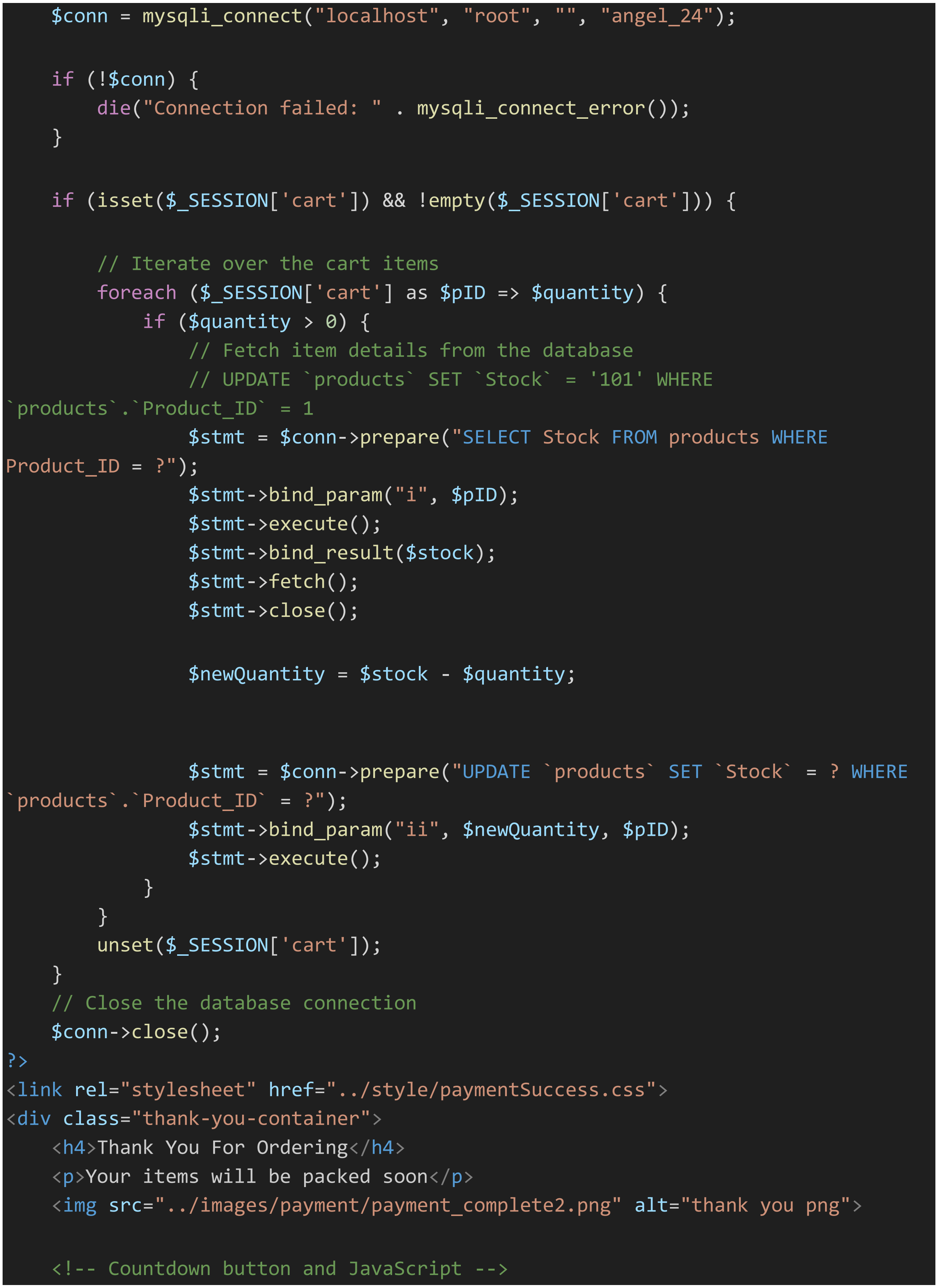
Angel_24/cart/list_checkout_item.php

```
<link rel="stylesheet" href="../style/checkout_item.css">
<?php
    if (session_status() === PHP_SESSION_NONE) {
        session_start();
    }

    // Connect to the database
    $conn = mysqli_connect("localhost", "root", "", "angel_24");

    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());
    }

    echo "<div class='checkout-item'>";
    echo "<h4>Item: </h4>";
    if (isset($_SESSION['cart']) && !empty($_SESSION['cart'])) {
        $total_price = 0;

        // Iterate over the cart items
        foreach ($_SESSION['cart'] as $pID => $quantity) {
            if ($quantity > 0) {
                $no_quantity = true;
                // Fetch item details from the database
                $stmt = $conn->prepare("SELECT Product_Name, Product_Image, 
Price, Stock FROM products WHERE Product_ID = ?");
                $stmt->bind_param("i", $pID);
                $stmt->execute();
                $stmt->bind_result($pName, $pImage, $price, $stock);
                $stmt->fetch();
                $stmt->close();

                // Display item details
                echo "<div class='product-row'>";
                    echo "<img src='" . htmlspecialchars($pImage) . "' alt='" .
htmlspecialchars($pName) . "' >";
                    echo "<div class='name-price-container'>";
                        echo "<p>".htmlspecialchars($pName)."</p>";
                        echo "<p>Quantity: ".htmlspecialchars($quantity)."</p>";
                    echo "</div>";
                    echo "<div class='price-container'>";
                        echo "<p>RM " . htmlspecialchars(number_format($price *
$quantity, 2)) . "</p>";
                    echo "</div>";
                echo "</div>";
                $total_price += $price * $quantity;
            }
        }
        echo "<div class='total-price-container'>";
            echo "<p class='total-price'>TOTAL: RM ".
htmlspecialchars(number_format($total_price, 2)) . "</p>";
        echo "</div>";
    }
    if (!$no_quantity) {
        echo "<p>Your cart is empty.</p>";
    }
    echo "</div>";

    // Close the database connection
    $conn->close();
?>
```

Explanation: this php script connect to database. If there's cart item session variable is set and not empty, it fetch all the cart items and display their information. If there is no cart item, a message is displayed informing user the cart is empty.

Angel_24/cart/paymentSuccessful.php

```
<?php
    session_start();

    // Connect to the database
```



```
    $conn = mysqli_connect("localhost", "root", "", "angel_24");

    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());
    }

    if (isset($_SESSION['cart']) && !empty($_SESSION['cart'])) {

        // Iterate over the cart items
        foreach ($_SESSION['cart'] as $pID => $quantity) {
            if ($quantity > 0) {
                // Fetch item details from the database
                // UPDATE `products` SET `Stock` = '101' WHERE
`products`.`Product_ID` = 1
                $stmt = $conn->prepare("SELECT Stock FROM products WHERE
Product_ID = ?");
                $stmt->bind_param("i", $pID);
                $stmt->execute();
                $stmt->bind_result($stock);
                $stmt->fetch();
                $stmt->close();

                $newQuantity = $stock - $quantity;


                $stmt = $conn->prepare("UPDATE `products` SET `Stock` = ? WHERE
`products`.`Product_ID` = ?");
                $stmt->bind_param("ii", $newQuantity, $pID);
                $stmt->execute();
            }
        }
        unset($_SESSION['cart']);
    }
    // Close the database connection
    $conn->close();
?>
<link rel="stylesheet" href="../style/paymentSuccess.css">
<div class="thank-you-container">
    <h4>Thank You For Ordering</h4>
    <p>Your items will be packed soon</p>
    <img src="../images/payment/payment_complete2.png" alt="thank you png">

    <!-- Countdown button and JavaScript -->
    <p id="countdown">Returning to homepage in 10 seconds...</p>
    <button id="returnBtn" onclick="redirectToHomepage()">Return to Homepage</button>
</div>

<script>
    // Countdown timer
    var countdown = 10;
    var countdownElement = document.getElementById('countdown');
    var countdownInterval = setInterval(function() {
        countdown--;
        countdownElement.textContent = 'Returning to homepage in ' + countdown + ' seconds...';
        if (countdown <= 0) {
            clearInterval(countdownInterval);
            redirectToHomepage();
        }
    }, 1000);

    // Function to redirect to homepage
    function redirectToHomepage() {
        window.location.href = '../home/index.php'; // Change 'index.php' to your actual homepage file
    }
</script>
```

Explanation: the php script connect to database, update the data is database and unset the cart item session variable. Then, it displays a picture and message informing the payment is successful. The js script set a variable, countdown as 10. Then, it use setInterval() function to decrease countdown by 1 every second. When countdown is less than or equal to 0, countdownInterval() is cleared and execute redirectToHomepage() function.RedirectToHomepage() function redirects user to the home page of Angel_24 website.

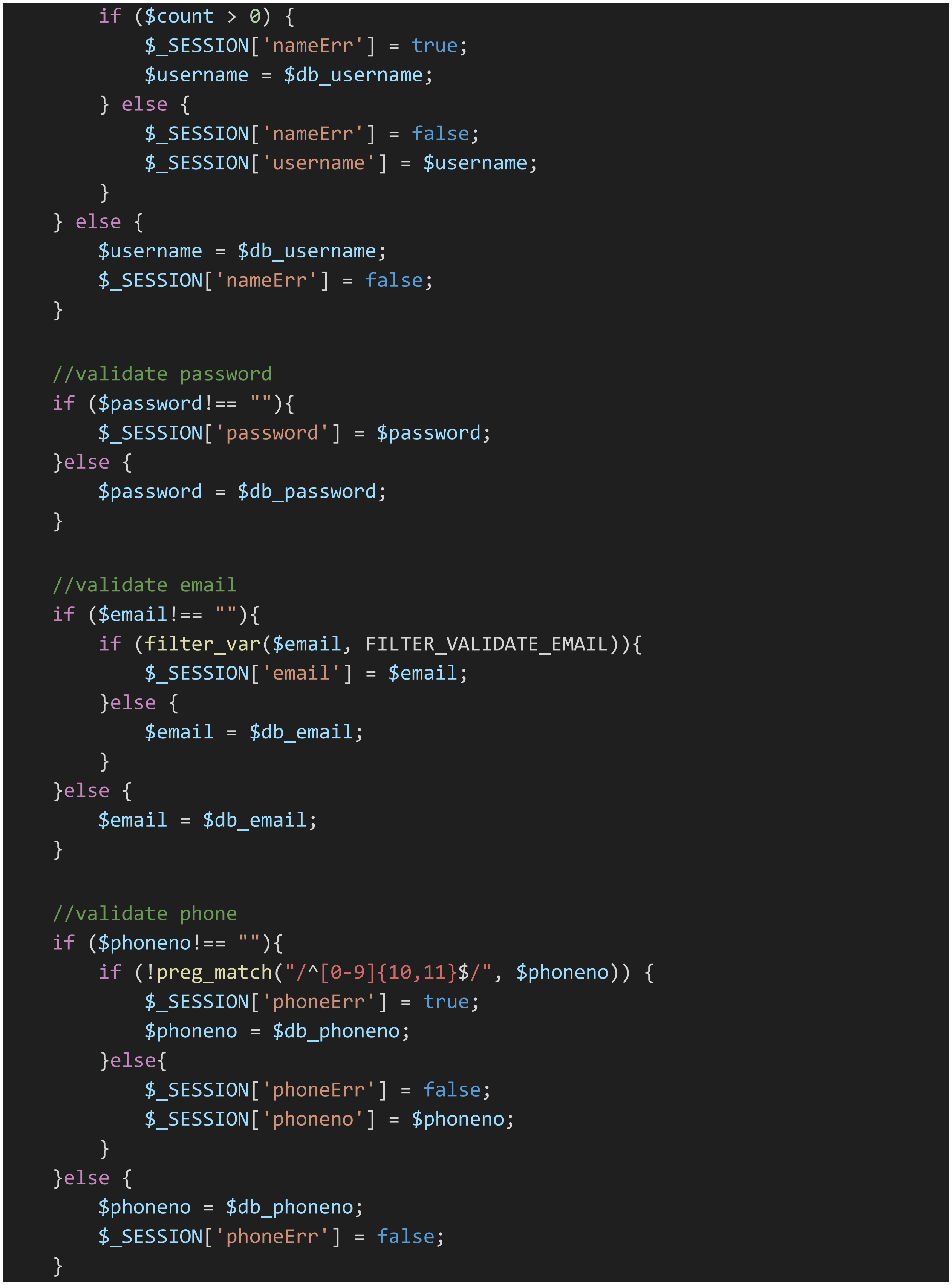
Edit profile page

Angel_24/profile/index.php

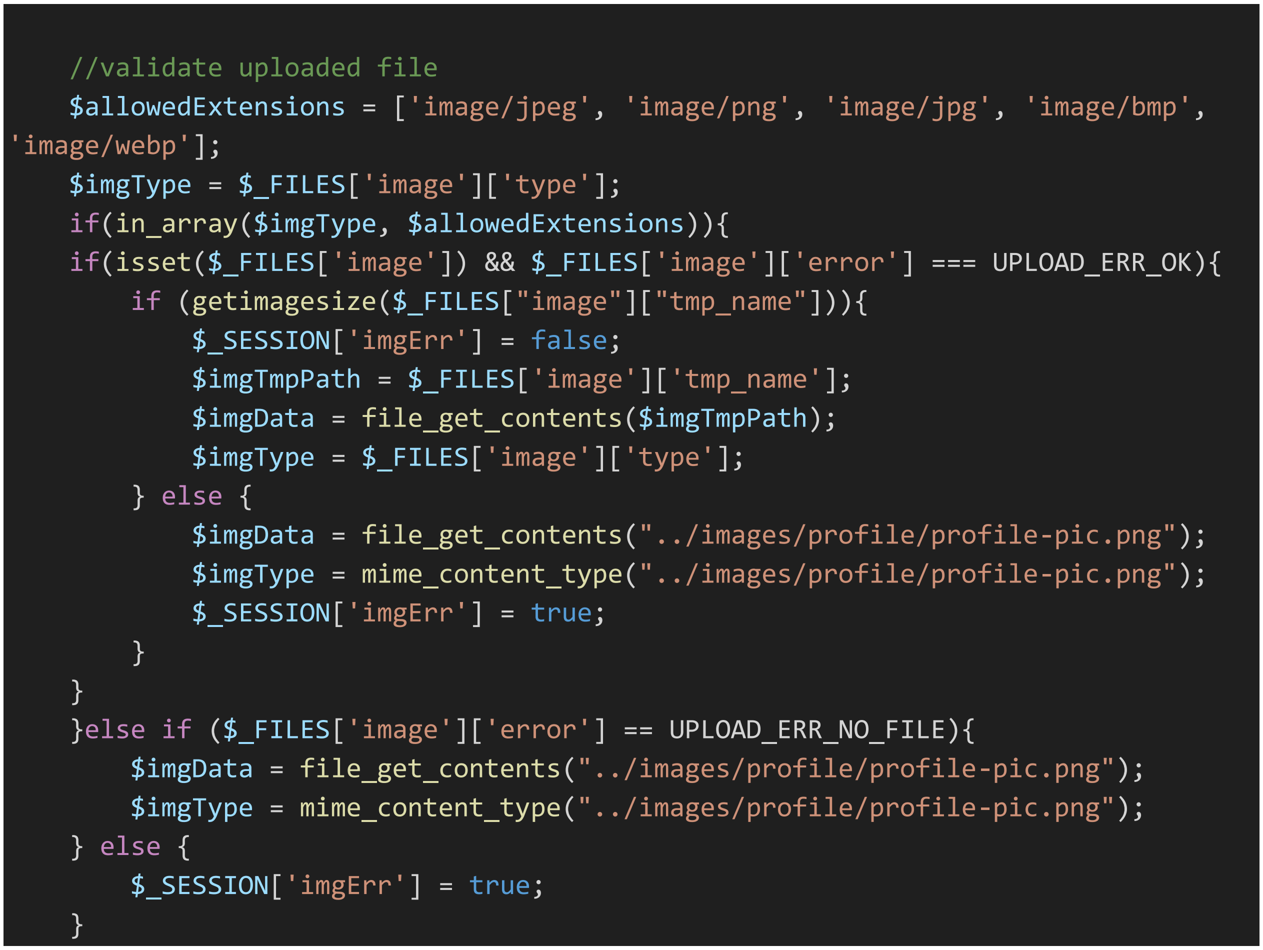
```
<?php
if (session_status() === PHP_SESSION_NONE) {
    session_start();
}
// Database connection
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "angel_24";

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
```

Explanation: This part of code set $servername, $username, $password, $dbname as “localhost”, “root”, “” and “angel_24” and connect to database using those variables. If there is error connected to the database, it stops the scripts and displays error message using die() function.

```
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Get the variables from the form
    $username = $_POST['name'];
    $password = $_POST['password'];
    $email = $_POST['email'];
    $phoneno = $_POST['phoneno'];

    $db_username = $_SESSION['username'];
    $db_password = $_SESSION['password'];
    $db_email = $_SESSION['email'];
    $db_phoneno = $_SESSION['phoneno'];

    // validate username
    if ($username !== "" && $username !== $db_username) {
        $stmt_check = $conn->prepare("SELECT COUNT(*) FROM user WHERE username
= ?");
        $stmt_check->bind_param("s", $username);
        $stmt_check->execute();
        $stmt_check->bind_result($count);
        $stmt_check->fetch();
        $stmt_check->close();
```



```php
        if ($count > 0) {
            $_SESSION['nameErr'] = true;
            $username = $db_username;
        } else {
            $_SESSION['nameErr'] = false;
            $_SESSION['username'] = $username;
        }
    } else {
        $username = $db_username;
        $_SESSION['nameErr'] = false;
    }

    //validate password
    if ($password!== ""){
        $_SESSION['password'] = $password;
    }else {
        $password = $db_password;
    }

    //validate email
    if ($email!== ""){
        if (filter_var($email, FILTER_VALIDATE_EMAIL)){
            $_SESSION['email'] = $email;
        }else {
            $email = $db_email;
        }
    }else {
        $email = $db_email;
    }

    //validate phone
    if ($phoneno!== ""){
        if (!preg_match("/^[0-9]{10,11}$/", $phoneno)) {
            $_SESSION['phoneErr'] = true;
            $phoneno = $db_phoneno;
        }else{
            $_SESSION['phoneErr'] = false;
            $_SESSION['phoneno'] = $phoneno;
        }
    }else {
        $phoneno = $db_phoneno;
        $_SESSION['phoneErr'] = false;
    }
```



```
    //validate uploaded file
    $allowedExtensions = ['image/jpeg', 'image/png', 'image/jpg', 'image/bmp',
'image/webp'];
    $imgType = $_FILES['image']['type'];
    if(in_array($imgType, $allowedExtensions)){
    if(isset($_FILES['image']) && $_FILES['image']['error'] === UPLOAD_ERR_OK){
        if (getimagesize($_FILES["image"]["tmp_name"])){
            $_SESSION['imgErr'] = false;
            $imgTmpPath = $_FILES['image']['tmp_name'];
            $imgData = file_get_contents($imgTmpPath);
            $imgType = $_FILES['image']['type'];
        } else {
            $imgData = file_get_contents("../images/profile/profile-pic.png");
            $imgType = mime_content_type("../images/profile/profile-pic.png");
            $_SESSION['imgErr'] = true;
        }
    }
    }else if ($_FILES['image']['error'] == UPLOAD_ERR_NO_FILE){
        $imgData = file_get_contents("../images/profile/profile-pic.png");
        $imgType = mime_content_type("../images/profile/profile-pic.png");
    } else {
        $_SESSION['imgErr'] = true;
    }
```

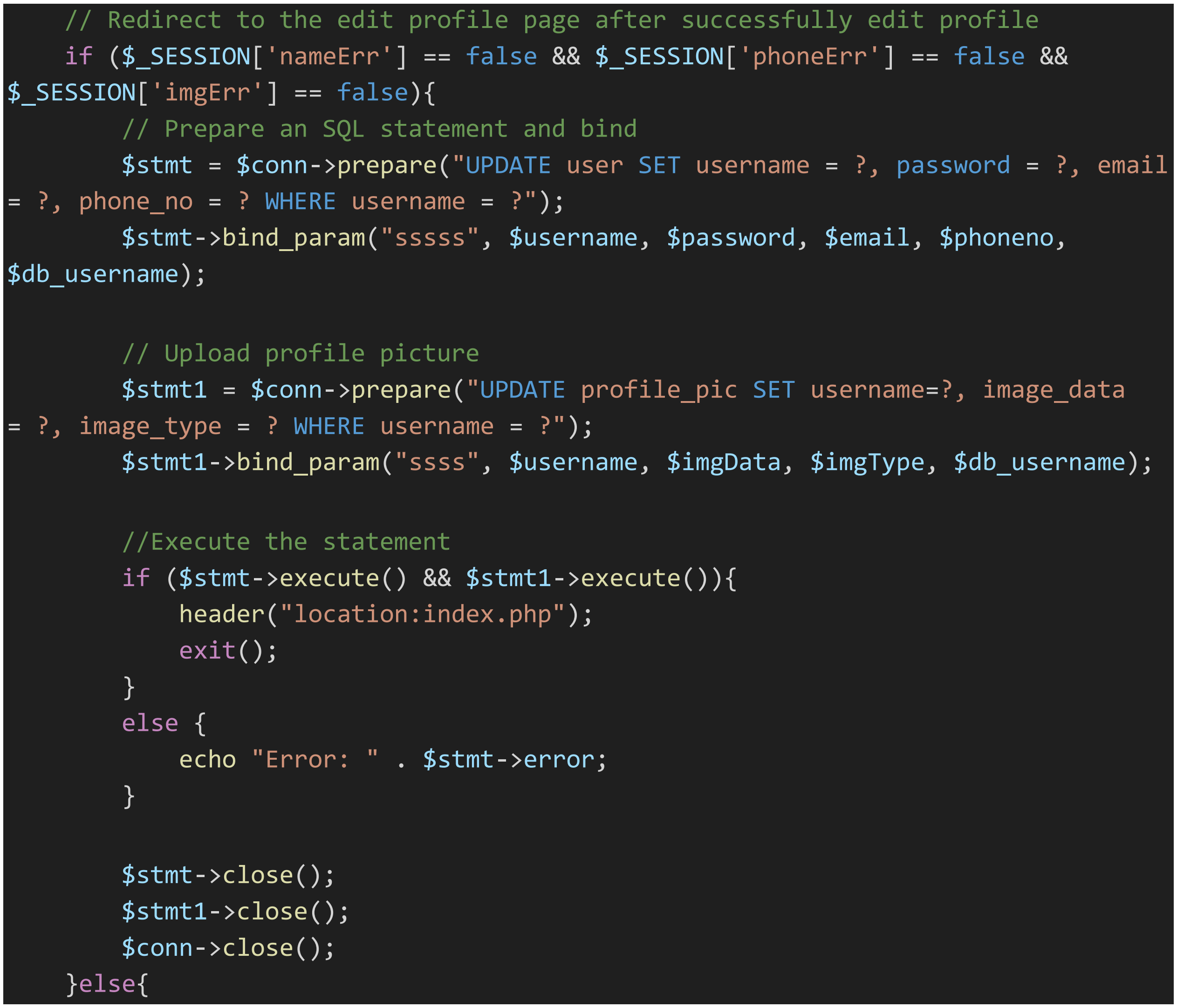
Explanation: When POST request is made, the following script starts executing. Variable from form are assigned to $username, $password, $email and $phoneno. Session current username, password, email and phoneno are assigned to $db_username, $db_password, $db_email and $db_phoneno. Then it start validating username. If username field left empty, username is set to current $db_username. If username existed in database, true is assigned to $_SESSION['nameErr']. If no error found, $_SESSION[‘username'] is set to $username and $_SESSION['nameErr'] is set to false.

During password validation, $password is set to $db_password if the password field is left empty. Else, $_SESSION[‘password’] is set to $password.

During email validation, $email is set to $db_email if the email field is left empty. Else, it checks if $email entered has a correct format using filter_var() with predefined php filter FILTER_VALIDATE_EMAIL. If format incorrect, $email is set to $db_email. Else, $_SESSION[‘email’] is set to $email.

During phone number validation, $phoneno is set to $db_phoneno if the phoneno field is left empty. Else, it check $phoneno using preg_match() function. "/^[0-9]{10,11}$/" validate if $phoeno is consists of number of 10 or 11 digits of number 0-9. If $phoneno is valid, $_SESSION['phoneErr'] is set to false and $_SESSION['phoneno'] is set to $phoneno. Else, $_SESSION['phoneErr'] is set to true and $phoneno is set to $db_phoneno.

During file validation, an array of extensions is created and the extension of the file uploaded is retrieved. If the file uploaded is either .jpeg, .jpg, .png, .bmp, .webp, then it checks if the upload is successful. If there is no error with uploading, it checks if that image is a real image. If true, $_SESSION['imgErr'] is set to false, image data is saved as $imgData using file_get_contents() function. If false, the profile picture is set to default profile picture and $_SESSION['imgErr'] is set to true. If no file is uploaded, profile picture is set to default profile picture. If the file type uploaded is not in array of extensions, $_SESSION['imgErr'] is set to true.



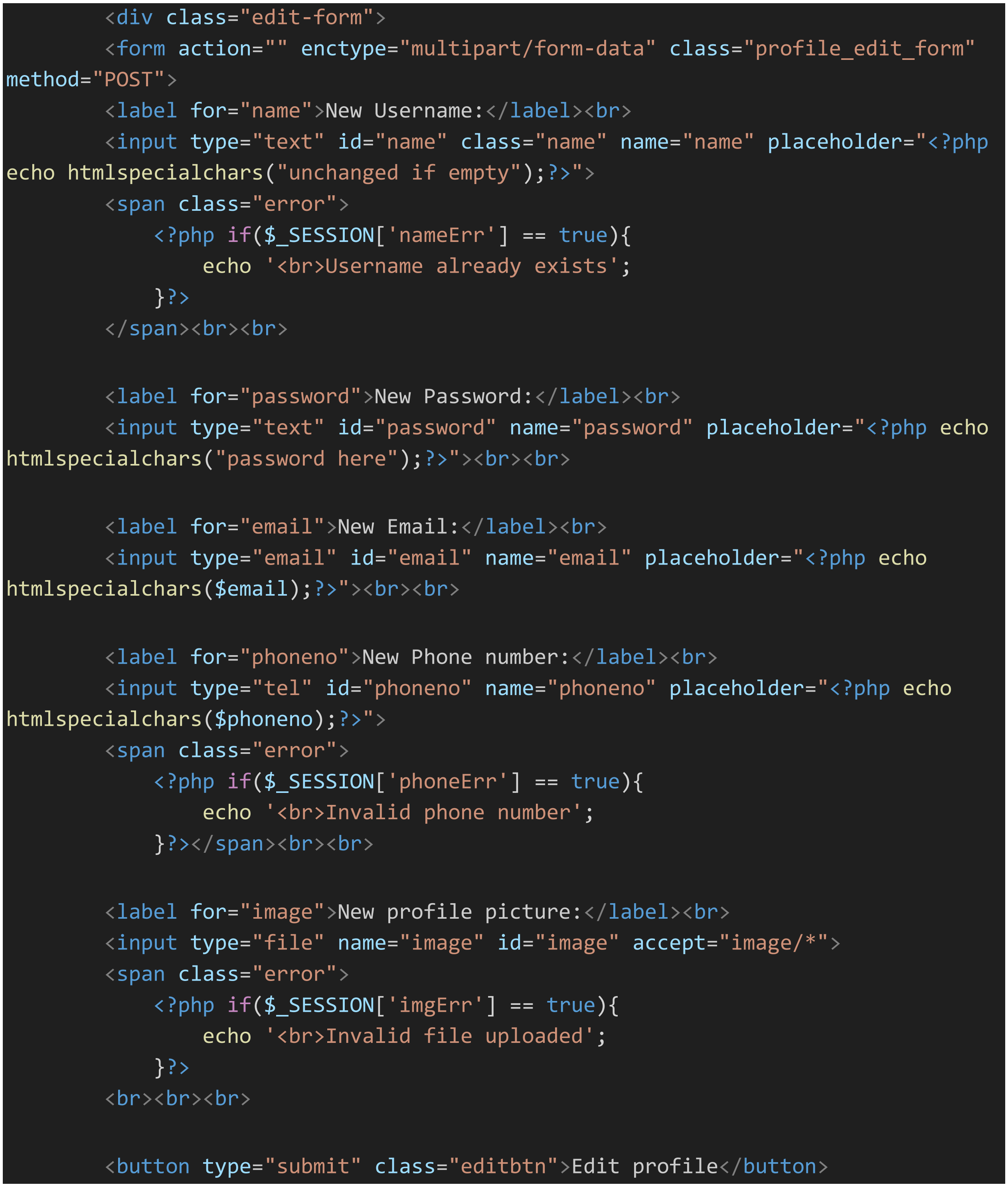
```
    // Redirect to the edit profile page after successfully edit profile
    if ($_SESSION['nameErr'] == false && $_SESSION['phoneErr'] == false &&
$_SESSION['imgErr'] == false){
        // Prepare an SQL statement and bind
        $stmt = $conn->prepare("UPDATE user SET username = ?, password = ?, email
= ?, phone_no = ? WHERE username = ?");
        $stmt->bind_param("sssss", $username, $password, $email, $phoneno,
$db_username);

        // Upload profile picture
        $stmt1 = $conn->prepare("UPDATE profile_pic SET username=?, image_data
= ?, image_type = ? WHERE username = ?");
        $stmt1->bind_param("ssss", $username, $imgData, $imgType, $db_username);

        //Execute the statement
        if ($stmt->execute() && $stmt1->execute()){
            header("location:index.php");
            exit();
        }
        else {
            echo "Error: " . $stmt->error;
        }

        $stmt->close();
        $stmt1->close();
        $conn->close();
    }else{
    }
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>
```

Explanation: If all session error variables are false, variables are bind to the prepare statements. If both statements are execute correctly, user is redirected to current page and script is terminated. Else, error message is displayed.

A function test_input($data) is created to clean and sanitize the data before further processing.

```
<!DOCTYPE html>
<html>
    <head>
        <title>Edit profile</title>
        <link rel="stylesheet" href="../style/mystyle1.css">
        <link rel="stylesheet" href="../style/edit_profile.css">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
    <body>
        <?php include("../includes/header.php"); ?>
        <?php include("../includes/navigation.php"); ?>
        <div class="present-profile">
            <img src="../includes/profile_pic.php" alt="Image from Database"><br>
            <?php
            $username = $_SESSION['username'];
            $password = $_SESSION['password'];
            $email = $_SESSION['email'];
            $phoneno = $_SESSION['phoneno'];
            echo "Username: ". $username. "<br>";
            echo "Password: ". $password. "<br>";
            echo "Email: ". $email. "<br>";
            echo "Phone number: ". $phoneno. "<br>";
            ?>
        </div>
```

Explanation: All the necessary css, header and navagation bar are included. Inside present-profile, user’s current information which are username, password, email and phone number are displayed.



```
        <div class="edit-form">
        <form action="" enctype="multipart/form-data" class="profile_edit_form"
method="POST">
        <label for="name">New Username:</label><br>
        <input type="text" id="name" class="name" name="name" placeholder="<?php
echo htmlspecialchars("unchanged if empty");?>">
        <span class="error">
            <?php if($_SESSION['nameErr'] == true){
                echo '<br>Username already exists';
            }?>
        </span><br><br>

        <label for="password">New Password:</label><br>
        <input type="text" id="password" name="password" placeholder="<?php echo
htmlspecialchars("password here");?>"><br><br>

        <label for="email">New Email:</label><br>
        <input type="email" id="email" name="email" placeholder="<?php echo
htmlspecialchars($email);?>"><br><br>

        <label for="phoneno">New Phone number:</label><br>
        <input type="tel" id="phoneno" name="phoneno" placeholder="<?php echo
htmlspecialchars($phoneno);?>">
        <span class="error">
            <?php if($_SESSION['phoneErr'] == true){
                echo '<br>Invalid phone number';
            }?></span><br><br>

        <label for="image">New profile picture:</label><br>
        <input type="file" name="image" id="image" accept="image/*">
        <span class="error">
            <?php if($_SESSION['imgErr'] == true){
                echo '<br>Invalid file uploaded';
            }?>
        <br><br><br>

        <button type="submit" class="editbtn">Edit profile</button>
```

```
    </form>
    </div>
    </body>
</html>
```

Explanation: an edit-form is created to contain all form elements. The form is created with an attribute of enctype="multipart/form-data" to retrieve the file uploaded by user. 4 text fields and 1 file upload button are created to retrieve user's input. 3 spans are created to display error messages to user when invalid input is entered.

Log out

Angel_24/includes/logout.php

```
<?php
    session_start();
    session_unset();
    session_destroy();
    if (isset($_COOKIE['username'])) {
        setcookie('username', '', time() - 3600, "/");
    }
    if (isset($_COOKIE['password'])) {
        setcookie('password', '', time() - 3600, "/");
    }
    header('Location: ../index.php');
    exit();
?>
```

Explaination: In this function, all the session variables are unset by session_unset() and the session is destroyed by session_destroy(). This function then checks if username and password are set. If they do, remove them by setting expiration time to an hour ago. After that, this method redirects user to../index.php which is login page and terminates itself by exit().

Conclusion

In conclusion, our web application development project successfully demonstrates the integration of essential web development skills, combining client-side and server-side technologies to create a functional e-commerce platform. Inspired by the in-game shop Angel 24 from Blue Archive, we have developed a visually appealing and user-friendly website that allows customers to browse and purchase instant noodles, meals, and snacks with ease. Through this project, we applied key concepts from the course, such as constructing static pages, building dynamic, event-driven features, and utilizing a MySQL database for backend management.

The design and functionality of the website were crafted to provide an optimal user experience, with responsive features that adapt to various screen sizes, making it accessible from any device. By incorporating key e-commerce functionalities like a navigation menu, product listing, shopping cart, and secure payment options, we ensured the website meets modern web application standards.

This project not only allowed us to hone our technical skills but also emphasized the importance of creating intuitive and visually engaging interfaces. The development process highlighted the critical balance between creativity and functionality, demonstrating how web applications can be both aesthetically pleasing and efficient. As a result, our web application meets the project objectives and provides a foundation for future improvements in web development practices.

Workload Summary

Member	Workload
Kok Yu Hang	**Code**: Most css in style folder, all js in javascript folder, checkCookie.php, login and register page, navigation bar, display item page and item details page. **Code explanation**: Login Page, Register Page, Home Page, Navigation bar.
Cheng Wei En	**Code**: home page, cart page, list_cart.php, edit profile page, edit_profile.css. **Code explanation**: Item details, cart page, Edit profile page, Log out
Chong Rong Quan	**Code**: Search bar function, checkout page, payment success page, database SQL code. **Code explanation**: Search bar, Item list by category
Ong Lit Jay	**Code**: contact us page and admin page. **Code explanation**: Contact page

References

CSS animation inspiration:

ブルーアーカイブ×ASMR. (2024). DLsite. https://www.dlsite.com/modpub/lp/home/BlueArchiveASMR/

CSS effect:

Beautiful CSS box-shadow examples - CSS Scan. (n.d.). Getcssscan.com. https://getcssscan.com/css-box-shadow-examples

CSS icon:

W3.CSS Icons. (n.d.). Www.w3schools.com. https://www.w3schools.com/w3css/w3css_icons.asp

Font used:

Google Fonts: Kanit. (n.d.). Google Fonts. https://fonts.google.com/specimen/Kanit

M PLUS 1 - Google Fonts. (2024). Google Fonts. https://fonts.google.com/specimen/M+PLUS+1?query=m+plus

List of Sources Cited

Products:

Page, H. (2017). Lay’s Classic Potato Chips, 10oz (Pack of 4) Ukraine | Ubuy. Ubuy Ukraine. https://www.ubuy.com.ua/en/product/1PQY0TPBU-lays-potato-chips-classic-10-ounce-pack-of-4

(2024). Lazada.com.my. https://www.lazada.com.my/products/doritos-tortilla-chips-nacho-cheese-65g-i3526979905.html

Page, H. (2017). Buy RITZ Original Crackers, 10.3 oz Online Malaysia | Ubuy. Ubuy Malaysia. https://www.ubuy.com.my/en/product/46B67BS-nabisco-ritz-crackers-10-3-oz

Orville Redenbacher’s Popcorn - Ready-to-Eat Cinema Butter (12 pack with a total of 12 150g sachets) : Amazon.ca: Grocery & Gourmet Food. (2024). Amazon.ca. https://www.amazon.ca/Orville-Redenbacher-Popcorn-Ready-Eat/dp/B00FPCW6GO

Amazon.com : PLANTERS Salted Peanuts, 6 oz. bag (12-Pack) : Grocery & Gourmet Food. (2024). Amazon.com. https://www.amazon.com/PLANTERS-Salted-Peanuts-oz-12-Pack/dp/B0BYKX76RL

Nature ValleyTM Crunchy Granola Bars Oats 'N Honey (Double Bar) (18 ct) 1.49 oz. (2024). Dev.generalmillsfoodservice.com. https://www.generalmillsfoodservice.com/products/category/snacks/bars/nature-valley/crunchy-granola-bars-oats-n-honey-1-49oz

(2024). Ebayimg.com. https://i.ebayimg.com/images/g/RHkAAOSwdEZgxkh0/s-l1600.webp

(2024). Ebayimg.com. https://i.ebayimg.com/images/g/1fIAAOSwolphVghP/s-l1600.webp

Page, H. (2017). Jack Links Beef Jerky, Original Flavor, 7-Pack - Malaysia | Ubuy. Ubuy Malaysia. https://www.ubuy.com.my/en/product/79ESOCKDW-7pack-jack-links-beef-jerky-original-1-2-pounder-bag-flavorful-meat-snack-10g-of-protein-and-80-calories-made-with-premium-beef-96-fat

(2024). Ebayimg.com. https://i.ebayimg.com/images/g/h84AAOSwJY5i2YDn/s-l960.webp

CHEETOS® Crunchy Cheese Flavored Snacks | Cheetos. (2019). CHEETOS® Crunchy Cheese Flavored Snacks | Cheetos. Cheetos.com. https://www.cheetos.com/products/cheetos-crunchy-cheese-flavored-snacks

Kellogg's® Rice Krispies Treats® Original Bars - SmartLabelTM. (2022). Kelloggs.com. https://smartlabel.kelloggs.com/Product/Index/00038000265013

Pringles Sour Cream and Onion Potato Crisps 102g. (2020). Jaya Grocer | Online Store KL. https://klec.jayagrocer.com/products/pringles-sour-cream-and-onion-potato-crisps-107g

KIND | Healthy Snacks | Wholesome Granola Bars & Clusters. (n.d.). KIND Snacks. https://www.kindsnacks.com/healthy-grains-bars/dark-chocolate-chunk-M18054.html

Sun-Maid Raisins (Canister) - Teik Senn Malaysia. (2023, February 14). Teik Senn Malaysia. https://teiksenn.com/product/sun-maid-raisins-canister/

Chocolate Chip. (n.d.). Clif Bar. https://shop.clifbar.com/products/clif-bar-chocolate-chip

Page, H. (2017). Goldfish Cheddar Cheese Baked Snack Crackers - 6.6 Malaysia | Ubuy. Ubuy Malaysia. https://www.ubuy.com.my/en/product/7FI80I-pepperidge-farm-goldfish-cheddar-crackers-6-6-oz-bag

Page, H. (2017). Buy Welch’s Mixed Fruit Snacks, 8 Oz Bag Online Malaysia | Ubuy. Ubuy Malaysia. https://www.ubuy.com.my/en/product/4MFJX9VAW-welch-s-fruit-snacks-mixed-fruit-bag-8-oz

(2022). Quakeroats.com. https://www.quakeroats.com/products/oat-snacks/chewy-dipps/chocolate-chip

Amazon.com. (2014). Amazon.com. https://www.amazon.com/Pepperidge-Farm-Milano-Cookies-Chocolate/dp/B00BUY621Q

(2022). Heb.com. https://www.heb.com/product-detail/maruchan-chicken-flavor-ramen-noodle-soup/133649

NISSIN FOODS Top Ramen Beef Flavor Cup Noodles - Beef - 2.25 oz - 12 / Carton | Bundle of 10 Cartons. (2024). Walmart.com. https://www.walmart.com/ip/NISSIN-FOODS-Top-Ramen-Beef-Flavor-Cup-Noodles-Beef-2-25-oz-12-Carton-Bundle-of-10-Cartons/1372163704

Kuyastindahan.co.uk, 2024, www.kuyastindahan.co.uk/images/indomie-mi-goreng-original-fried-noodles-80g-p1157-3012_image.jpg.

and, H. (2024). *Amazon.co.jp: 10 x Heinzmak and Cheese Original 205GM : Food ...* Www.google.com. https://images.app.goo.gl/5kpkGnHwKWNQdhqBA

Campbell's Condensed Soup Chicken Noodle - 284 ml | Valumart. (2024). *Campbell’s Condensed Soup Chicken Noodle - 284 ml | Valumart*. Www.google.com. https://images.app.goo.gl/s7ZY3fL3RY2w8DJWA

HORMEL. (2024). *HORMEL COMPLEATS Meatloaf & Gravy With ... - Amazon.com*. Www.google.com. https://images.app.goo.gl/qsh9u5ykB9JgCTB47

Kitchen, A. (2024). *Amy's Kitchen - Amy's Indian Palak Paneer*. Www.google.com. https://images.app.goo.gl/be28v7Q63ts9PN368

Lipton. (2024). *Lipton Cup-a-Soup Instant Soup For a Warm Cup of Soup Chicken ...* Www.google.com. https://images.app.goo.gl/QWCXpJidvw5jejX17

Market, F. (2024). *365 by Whole Foods Market, Organic Microwave Popcorn, 21 Ounce ...* Www.google.com. https://images.app.goo.gl/Zy2L4SWMymRedGL59

Maruchan Yakisoba Teriyaki Beef Flavor Home-style Japanese Noodles. (2024). *Maruchan Yakisoba Teriyaki Beef Flavor Home-style Japanese Noodles ...* Www.google.com. https://images.app.goo.gl/dgPYiYjMnxsRA5uXA

Mix, S. (2024). *Chicken Noodle Soup Mix - Bear Creek*. Www.google.com. https://images.app.goo.gl/1K7tPh3BttEhrvRcA

oz. (2024). *Chef Boyardee Beef Ravioli, 15 oz [24-Cans] | eBay*. Www.google.com. https://images.app.goo.gl/P8zZJfVH8mbMfe4F8 oz ,. (2024). *Progresso Chicken Noodle Soup - 19 oz, Pack of 3 Malaysia | Ubuy*. Www.google.com. https://images.app.goo.gl/LfkoGiPCdrHL2Lu27

Ready, O. (2024). *Uncle Ben's Original Ready Rice Basmati Rice, Easy Side Dish, 8.5 OZ P*. Www.google.com. https://images.app.goo.gl/h27rvnuVARjJvAGb6

Rotini, A. (2024). *Prego Ready Meals Creamy Three Cheese Alfredo Rotini - 4 Pack in ...* Www.google.com. https://images.app.goo.gl/mvXffcXyWYJu156c7

Shells, V. (2024). *Velveeta Shells and Cheese Macaroni and Cheese Cups Easy ...* Www.google.com. https://images.app.goo.gl/8nAkrqYea21GjgAd8

Side, N. (2024). *Teriyaki Noodles Side | Knorr US*. Www.google.com. https://images.app.goo.gl/RByUh3naJG3GE5ny6

Weis. (2024). *Shop | Weis Markets*. Www.google.com. https://images.app.goo.gl/r4akxACfPwcjS4st9

(2024a). Kuyastindahan.co.uk. https://www.kuyastindahan.co.uk/images/indomie-mi-goreng-original-fried-noodles-80g-p1157-3012_image.jpg

(2024b). Mawolatraders.com. https://www.mawolatraders.com/wp-content/uploads/2021/03/12029994.jpg

(2024c). Media-Amazon.com. https://m.media-amazon.com/images/I/A1nq41GRyvL._AC_SL1500_.jpg

(2024d). Woolworths.media. https://cdn0.woolworths.media/content/wowproductimages/large/032230.jpg

(2024e). Albertsons-Media.com. https://images.albertsons-media.com/is/image/ABS/960297476?

(2024f). Media-Amazon.com. https://m.media-amazon.com/images/I/A1JHLg2Z-fL._SL1500_.jpg

(2024g). Shopify.com. https://cdn.shopify.com/s/files/1/0362/8158/1707/products/6169629adc9d10eb6b1668c1fb67fa8b_04-13-20_10-40-33_grande.jpg?v=1587177225

(2024h). Sdlcdn.com. https://n3.sdlcdn.com/imgs/b/c/d/MAGGI-2-Minute-Noodles-Masala-SDL421975270-4-1a414.jpg

(2024i). Ssl-Images-Amazon.com. https://images-na.ssl-images-amazon.com/images/I/81kFdSChhKL.jpg

(2024j). Jcook.pl. https://jcook.pl/userdata/public/gfx/1265/Makaron-instant-Ramen-Jin-Mild-CUP-marki-Ottogi-65g.jpg

(2024k). Intercityshops.com. https://www.intercityshops.com/wp-content/uploads/2020/11/s-l640.jpg

(2024l). App.goo.gl. https://images.app.goo.gl/jX4rJK73KQ4NeMGh7

(2024m). App.goo.gl. https://images.app.goo.gl/EH4bJYF6UqwUbr7t9

(2024n). App.goo.gl. https://images.app.goo.gl/AB5H2MviPWCrnEFi8

(2024o). App.goo.gl. https://images.app.goo.gl/vBMNqcJYdFpmgzZi7

(2024p). Walmartimages.com. https://i5.walmartimages.com/asr/aee4c493-3375-4298-ace5-3581c29271ee.3ec667fbdbfca7a8fcd98443a5660cea.jpeg

(2024q). App.goo.gl. https://images.app.goo.gl/wiyLutxCbQvsxU528

Payment:

Facebook. (2022). Facebook.com. https://www.facebook.com/photo.php?fbid=182899750776689&id=103914295341902&set=p.182899750776689

BH, B. (2020, February 10). Touch 'n Go eWallet guna kod QR DuitNow. Berita Harian. https://www.bharian.com.my/bisnes/teknologi/2020/02/654415/touch-n-go-ewallet-guna-kod-qr-duitnow

(2024). Wikimedia.org. https://upload.wikimedia.org/wikipedia/commons/a/a4/Paypal_2014_logo.png

(2024). Wikimedia.org. https://upload.wikimedia.org/wikipedia/commons/2/2a/Mastercard-logo.svg

admin. (2020, September 11). FPX Logo - Dita Selia. Dita Selia. https://www.ditaselia.my/payment-gateway/fpx-logo/

Other images:

(2024). Nocookie.net. https://static.wikia.nocookie.net/blue-archive/images/3/39/Sanctum_Tower_1.png/revision/latest/scale-to-width-down/1000?cb=20220204064732

Blue Archive Stickers. (2023). SigStick. https://www.sigstick.com/pack/J4AagQnWMPrpcgB9S4Iu

(2024). Dlsite.com. https://www.dlsite.com/modpub/lp/home/BlueArchiveASMR/img/kv_dot.png

(2024). Dlsite.com. https://www.dlsite.com/modpub/lp/home/BlueArchiveASMR/img/bg.png

(2024). Dlsite.com. https://www.dlsite.com/modpub/lp/home/BlueArchiveASMR/img/bg_frame_lb.png

(2024). Dlsite.com. https://www.dlsite.com/modpub/lp/home/BlueArchiveASMR/img/bg_frame_lt.png

(2024). Dlsite.com. https://www.dlsite.com/modpub/lp/home/BlueArchiveASMR/img/bg_frame_rb.png

(2024). Dlsite.com. https://www.dlsite.com/modpub/lp/home/BlueArchiveASMR/img/bg_frame_rt.png

(2024). Dlsite.com. https://www.dlsite.com/modpub/lp/home/BlueArchiveASMR/img/newrelease_headline.png

(2024). Dlsite.com. https://www.dlsite.com/modpub/lp/home/BlueArchiveASMR/img/title_bg.png