Task 3

Elisea Mizzi
PHP

1.How my database was set up.

   a. PHP: Used for scripting on the server side.
   b. MySQL: Database administration used for storing information.
   c. Bootstrap: This was used to help me ensure that I would have an interface that is both visually appealing and responsive.
   d. PHP was used to facilitate my communication with the MySQL database where information would be stored.
   e. MySQL was used as a data storage platform, guaranteeing effective management of user and product-related data.

2.Techniques used to manipulate data in my database through my web application.

   a. Create: Added functionality for managing user profiles, registering accounts, and adding new products.
   b. Read: displayed user profile, product details, and alcohol goods.
   c. Edit: allows users to change the quantity of products in their shopping carts and their profiles.
   d. Remove: The application allowed admin users to add and remove items.
   e. To guarantee efficient data management across the application, the application's codebase was smoothly updated, deleted, and added functionality allowed changes to be saved to data storage platform and edited as needed.

3. How a virtual server was set up locally on my device to mimic a live server.

   a. Xampp: An all-in-one development environment that combined PHP and MySQL on a local server.
   b. Xampp offered a practical and easy-to-use method for configuring a local server. The integration of Apache, MySQL, and PHP via Xampp guaranteed alignment within the server.

4. Techniques used to build a dynamic web application.

   a. includes/: contained functions necessary for the operation of the application, database scripts, and configuration files.
   b. style.css: provide a well-structured CSS for aesthetically pleasing and consistent design.
   c.   index.php: functioned as the home page, dynamically showing alcohol goods according to data from the database.
   d.   details.php: provided comprehensive details about a chosen product, improving the user's online shopping experience.
   e.   README.md: Detailed instructions to help developers comprehend and manage the programme.

 Carefully considered folder structure was designed to maintain code organisation and foster developer communication. Using a separate stylesheet (style.css), the programme could have a unified and visually appealing appearance throughout. Key pages such as index.php and product_detail.php were designed specifically to draw attention to the dynamic aspects of the online application.

5.Include Test Cases following the process detailed in your IPO chart in Task 1.

Input	Expected output	Actual output	Function	Comment
Home button	Takes the user back to the home page	Takes the user back to the home page	Pass	Easily accessible
Hamburger menu	Is shown and hidden when selected by the user. Lists categories underneath each other.	The user presses a drop down button called products. Once the user presses it again it is hidden.	pass	Changed the menu from a hamburger menu to a dropdown menu. Making it look neater to users.
Product Category	Selection items are listed underneath when pressed	Items are listed	Pass	The dropdown menu makes the design feel a bit neater.
Shopping Cart	Items the user has selected go to the cart where the user can edit the amount as well as delete these items.	Items appear in cart user has the option to edit or remove items.	Pass	
Sign up button	User enters requested date. This data is then sent over to php myadmin for future use, to allow users to only login.	User data is sent to php my admin users can login in the future.	Pass	
Login button	User information is saved when they	When login button is pressed users are	Pass	

	sign up, allowing them to enter information that was created during the sign up stage.	directed to the homepage.		
Professional view	Only an admin member is allowed to view this page. Once the admin uses his email and password he is directed to the admin homepage.	When the admin enters the professional view with his admin account he can edit and arrange the elements of the website as needed.	Pass	Takes the admin user which is logged in with their account information. To be able to make any necessary adjustments needed.
Submit Application	Information is then stored to php myadmin for future use.	Information is stored in php myadmin.	Pass	
Add quantity	User is allowed to add amount they would need before adding it to the cart where the amount total would show.	Amount selected is viewed in cart and is editable.	Pass	
Add to cart	Once the user selects the more information button he will see the option to adjust the quantity amount as well as add the item to the cart. The item will be added to the	Items listed are added to php myadmin.	Pass	

	cart as well as to the php myadmin row.			
Edit amount in cart	Once the user presses the cart he will have the option to edit item quantity. Once the button is pressed it will be edited according to the new item amount from the user to php myadmin.	Amount is updated	Pass	
Delete product in cart	Once the user presses the cart he will have the option to delete item. Once the button is pressed it will be removed from php myadmin.	Item is removed from php myadmin.	Pass	
Edit user profile	From the professional view, admins are able to select the profile tab where they will have the options to edit their profile such as name, surname and email.	User is able to edit their name, surname or email and it will save to phpMyAdm in when the user presses save.	Pass	
Role Management	From the professional view, admins are able to select the	Role Manageme nt tab is opened.	Pass	

	role management tab where they will have the options to add, edit and delete role.			
Product Management	From the professional view, admins are able to select the product management tab where they will have the options to add, edit and delete products.	Product Management tab is open.	Pass	
Add role	From the professional view, admins are able to add roles for the user. Once it is added the admin can save the changes to php my admin.	Changes are added to php my admin	Pass	
Save Role	From the professional view, admins are able to edit the user role from the role management. Once it is edit the admin can save the changes to php my admin.	Changes are saved to php myadmin	Pass	
Delete role	From the professional view, admins are able to	User is deleted from	Pass	

	delete users from the website as well as php my admin removing them completely from the system.	phpMyAdmin.		
Save product	From the professional view, admins are able to add products from the website as well as php my admin adding them into the system.	A new product is added to php myadmin.	Pass	
Delete product	From the professional view, admins are able to remove products from the website as well as php my admin deleting them from the system completely.	The product was removed from php myadmin	Pass	