

**Term Project**

**[ COMP743 – Web Development]**

**Diploma in Software Development Level 7**

**[Rodrigo Adati]** [**17018158c@cornell.ac.nz**](mailto:17018158c@cornell.ac.nz)

**[Adrian Del Rosario] 16097908c**[**@cornell.ac.nz**](mailto:17018054c@cornell.ac.nz)

**[Ivy Rose Valdez]** [**17018201c@cornell.ac.nz**](mailto:17018201c@cornell.ac.nz)

Contents

[1. Business Description 4](#_Toc491819810)

[2. Scope 4](#_Toc491819811)

[3. Use case 5](#_Toc491819812)

[A. Use Case Diagram 5](#_Toc491819813)

[B. Use Case Description 5](#_Toc491819814)

[TASK 2- DESIGN AND DEVELOPMENT OF STATIC WEBSITE 10](#_Toc491819815)

[1. Application Design 10](#_Toc491819816)

[TASK 3 – DATABASE DESIGN & SQL FUNCTIONS 13](#_Toc491819818)

[1. ERD Diagram 13](#_Toc491819819)

[2. Data Base Scripts 14](#_Toc491819820)

[3. Data base Functions 19](#_Toc491819821)

[TASK 4 – WEB APPLICATION DEVELOPMENT USING PHP 22](#_Toc491819822)

# Business Description

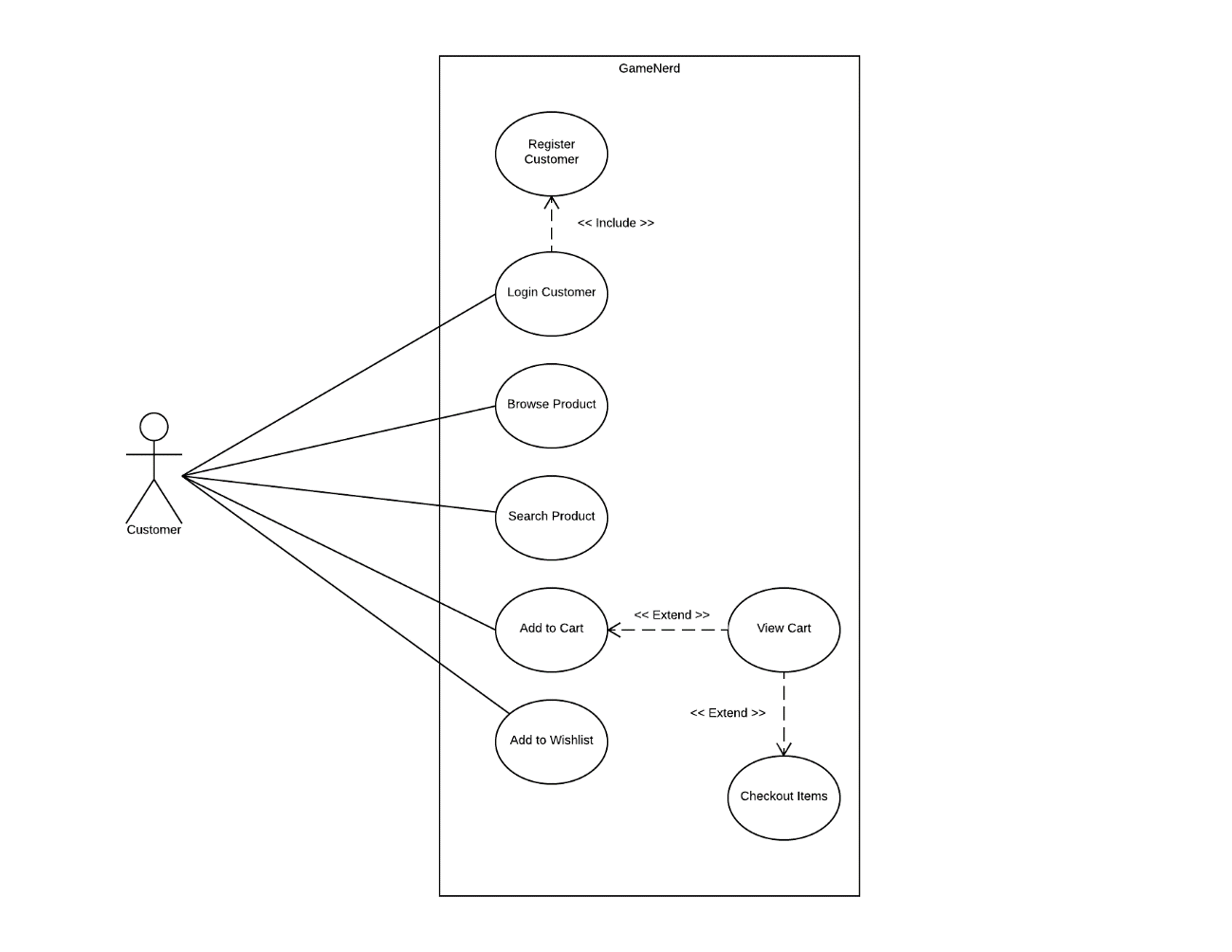
Gamenerd is an e-commerce website specializing in selling consoles and PC gaming related merchandise. This allows a customer to search and buy items from the website without the need of going to a physical store. The purpose of the project is to make it convenient for customers that have interest into gaming, to purchase game related items.

# Scope

For the scope of the project the customer will be able to search for products, save products to the cart and wishlist, create an account to manage their information as well as purchases made in the website.

# Use case

## Use Case Diagram



## Use Case Description

|  |  |
| --- | --- |
| Use Case Name | Register Customer |
| Triggers | The customer creates an account to have access to the system. |
| Basic Course of Events | 1. The customer clicks on the Sign In link. 2. The customer clicks on the “Not a member? Signup” link 3. The system displays the signup form. 4. The customer enters the first name. 5. The customer enters the last name. 6. The customer enters the email 7. The customer enters the username 8. The customer enters the password 9. The customer enters the confirm password 10. The customer enters the address 11. The customer enters the city 12. The customer enters the state 13. The customer enters the postal code 14. The client clicks on the Create Account button. 15. The system adds the customer information to the database. |
| Exceptions Flow | 4a. If the customer did not input the first name, the system displays the message “Fields should not be empty”.  5a. If the customer did not input the last name, the system displays the message “Fields should not be empty”.  6a. If the customer did not input the email, the system displays the message “Fields should not be empty”.  7a. If the customer did not input the username, the system displays the message “Fields should not be empty”.  8a. If the customer did not input the password, the system displays the message “Fields should not be empty”.  9a. If the customer`s input is not the same as the password, the system displays the message “Password does not match”.  10a. If the customer did not input the address, the system displays the message “Fields should not be empty”.  11a. If the customer did not input the city, the system displays the message “Fields should not be empty”.  12a. If the customer did not input the state, the system displays the message “Fields should not be empty”.  13a. If the customer did not input the postal code, the system displays the message “Fields should not be empty”. |
| Post-Conditions | **Successful Completion**  The customer account is created.  **Failure Condition**  The system do not register the customer. |

|  |  |
| --- | --- |
| Use Case Name | Login Customer |
| Triggers | The customer needs to login into the system. |
| Basic Course of Events | 1. The customer clicks on the Sign In link. 2. The system displays the login form. 3. The customer enters the username. 4. The customer enters the password. 5. The client clicks on the Sign In button. 6. The system grants the customer access to the system. |
| Exceptions Flow | 3a. If the customer username does not exist the system displays the message “No user found”.  4a. If the customer password is not correct the system displays the message “No user found”. |
| Post-Conditions | **Successful Completion**  The customer have access to the system.  **Failure Condition**  The system do not grant access to the customer. |

|  |  |
| --- | --- |
| Use Case Name | Search Product |
| Triggers | The customer needs to find products in the system |
| Basic Course of Events | 1. The customer input the name of the product. 2. The customer clicks the search button. 3. The system displays a list of products. |
| Exceptions Flow | 1a. If the customer do not input the name of the product the system displays the message “Please input a filter”. |
| Post-Conditions | **Successful Completion**  System returns a list of products. |

|  |  |
| --- | --- |
| Use Case Name | Browse Product |
| Triggers | The customer needs to browse products in the system |
| Basic Course of Events | 1. The customer clicks on the department. 2. The customer clicks the category link. 3. The system displays a list of products. |
| Post-Conditions | **Successful Completion**  System returns a list of products. |

|  |  |
| --- | --- |
| Use Case Name | Add to cart |
| Triggers | The customer adds a product to the cart |
| Basic Course of Events | 1. The customer clicks on a product. 2. The customer clicks the Add to cart button. 3. The system adds the product to the cart. |
| Post-Conditions | **Successful Completion**  System adds the product to the cart. |

|  |  |
| --- | --- |
| Use Case Name | Add to Wishlist |
| Triggers | The customer adds a product to the wishlist |
| Basic Course of Events | 1. The customer clicks on a product. 2. The customer clicks the Add to wishlist button. 3. The system adds the product to the wishlist. |
| Post-Conditions | **Successful Completion**  System adds the product to the wishlist. |

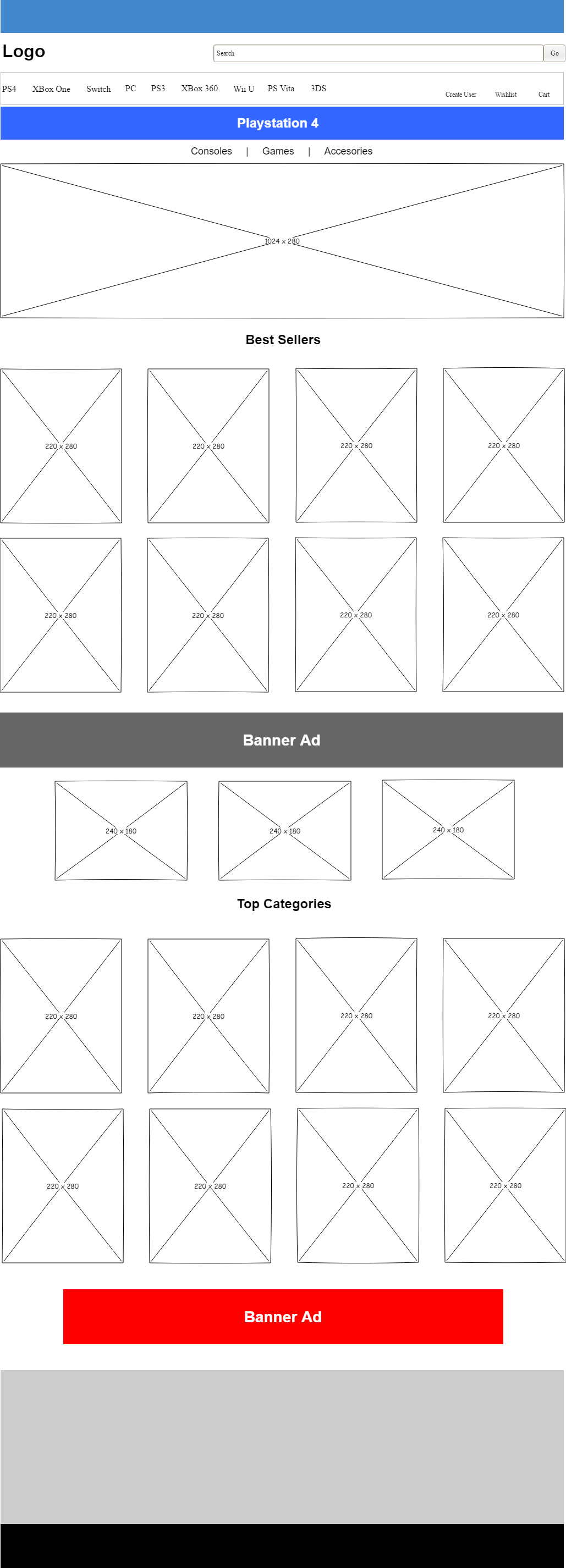
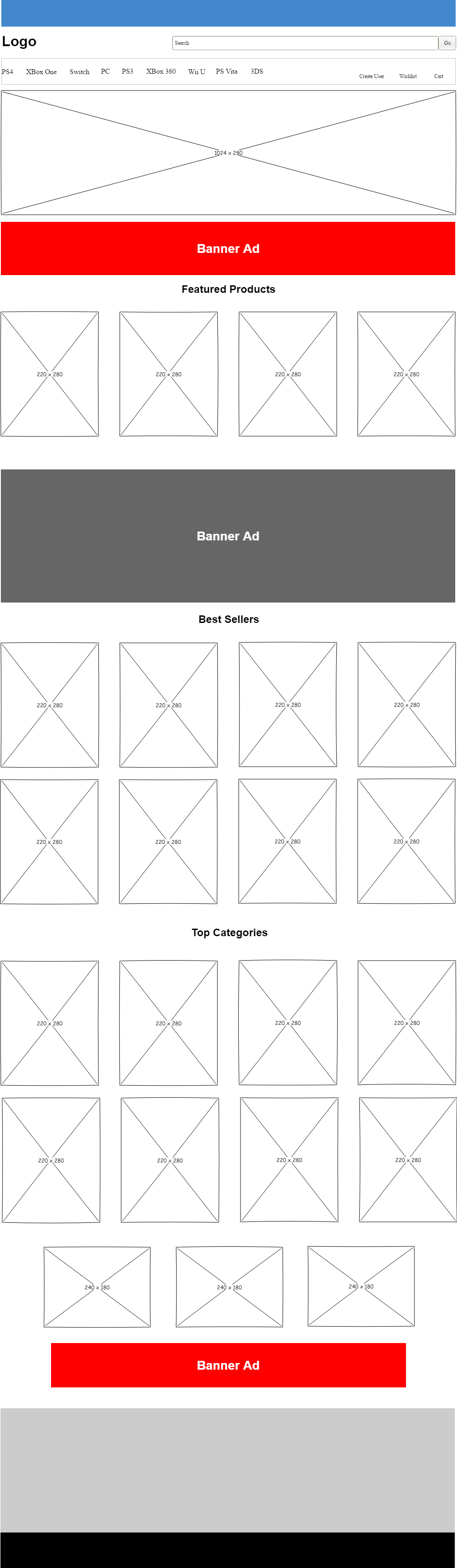
|  |  |
| --- | --- |
| Use Case Name | View Cart |
| Triggers | The customer needs to view the products from the cart |
| Basic Course of Events | 1. The customer clicks on the cart link. 2. The system displays the products from the cart. |
| Post-Conditions | **Successful Completion**  System displays the product from the cart. |

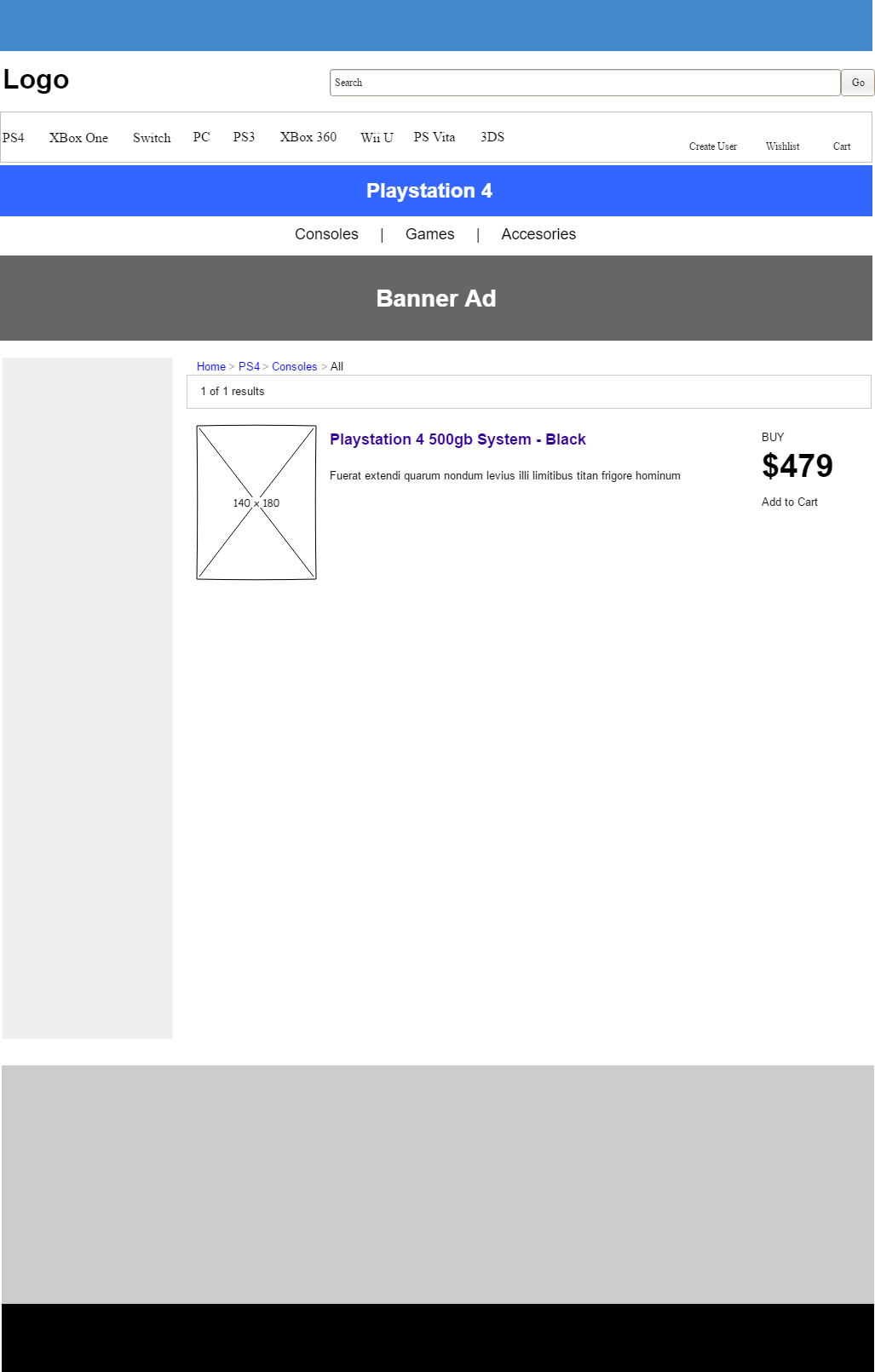
|  |  |
| --- | --- |
| Use Case Name | Checkout Item |
| Triggers | The customer wants to finalize the purchase |
| Basic Course of Events | 1. The customer clicks on the cart link. 2. The system displays the products from the cart. 3. The customer clicks the Finish button |
| Post-Conditions | **Successful Completion**  System process the customer request. |

# TASK 2- DESIGN AND DEVELOPMENT OF STATIC WEBSITE

# Application Design

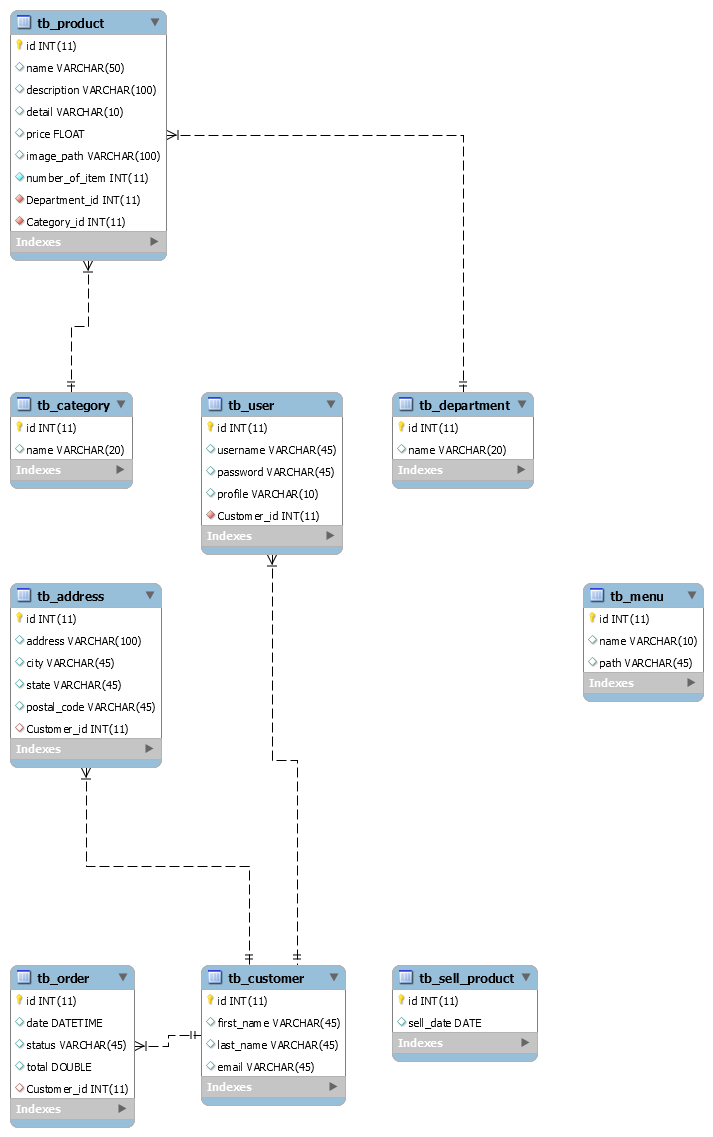
## 





# TASK 3 – DATABASE DESIGN & SQL FUNCTIONS

## ERD Diagram

****

## Data Base Scripts

CREATE SCHEMA IF NOT EXISTS `comp743` DEFAULT CHARACTER SET utf8 ;

USE `comp743` ;

-- -----------------------------------------------------

-- Table `comp743`.`tb\_customer`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `comp743`.`tb\_customer` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`first\_name` VARCHAR(45) NULL DEFAULT NULL,

`last\_name` VARCHAR(45) NULL DEFAULT NULL,

`email` VARCHAR(45) NULL DEFAULT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

AUTO\_INCREMENT = 27

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `comp743`.`tb\_address`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `comp743`.`tb\_address` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`address` VARCHAR(100) NULL DEFAULT NULL,

`city` VARCHAR(45) NULL DEFAULT NULL,

`state` VARCHAR(45) NULL DEFAULT NULL,

`postal\_code` VARCHAR(45) NULL DEFAULT NULL,

`Customer\_id` INT(11) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `Customer\_id\_idx` (`Customer\_id` ASC),

CONSTRAINT `Customer\_id`

FOREIGN KEY (`Customer\_id`)

REFERENCES `comp743`.`tb\_customer` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

AUTO\_INCREMENT = 21

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `comp743`.`tb\_category`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `comp743`.`tb\_category` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

AUTO\_INCREMENT = 4

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `comp743`.`tb\_department`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `comp743`.`tb\_department` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(20) NULL DEFAULT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

AUTO\_INCREMENT = 10

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `comp743`.`tb\_menu`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `comp743`.`tb\_menu` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(10) NULL DEFAULT NULL,

`path` VARCHAR(45) NULL DEFAULT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

AUTO\_INCREMENT = 10

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `comp743`.`tb\_order`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `comp743`.`tb\_order` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`date` DATETIME NULL DEFAULT NULL,

`status` VARCHAR(45) NULL DEFAULT NULL,

`total` DOUBLE NULL DEFAULT NULL,

`Customer\_id` INT(11) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `id\_idx` (`Customer\_id` ASC),

CONSTRAINT `id`

FOREIGN KEY (`Customer\_id`)

REFERENCES `comp743`.`tb\_customer` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

AUTO\_INCREMENT = 12

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `comp743`.`tb\_product`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `comp743`.`tb\_product` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(50) NULL DEFAULT NULL,

`description` VARCHAR(100) NULL DEFAULT NULL,

`detail` VARCHAR(10) NULL DEFAULT NULL,

`price` FLOAT NULL DEFAULT NULL,

`image\_path` VARCHAR(100) NULL DEFAULT NULL,

`number\_of\_item` INT(11) NOT NULL DEFAULT '1',

`Department\_id` INT(11) NOT NULL,

`Category\_id` INT(11) NOT NULL,

PRIMARY KEY (`id`),

INDEX `fk\_Product\_Department1\_idx` (`Department\_id` ASC),

INDEX `fk\_Product\_Category1\_idx` (`Category\_id` ASC),

CONSTRAINT `fk\_Product\_Category1`

FOREIGN KEY (`Category\_id`)

REFERENCES `comp743`.`tb\_category` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_Product\_Department1`

FOREIGN KEY (`Department\_id`)

REFERENCES `comp743`.`tb\_department` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

AUTO\_INCREMENT = 43

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `comp743`.`tb\_sell\_product`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `comp743`.`tb\_sell\_product` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`sell\_date` DATE NULL DEFAULT NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------

-- Table `comp743`.`tb\_user`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `comp743`.`tb\_user` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`username` VARCHAR(45) NULL DEFAULT NULL,

`password` VARCHAR(45) NULL DEFAULT NULL,

`profile` VARCHAR(10) NULL DEFAULT NULL,

`Customer\_id` INT(11) NOT NULL,

PRIMARY KEY (`id`),

INDEX `fk\_User\_Customer\_idx` (`Customer\_id` ASC),

CONSTRAINT `fk\_User\_Customer`

FOREIGN KEY (`Customer\_id`)

REFERENCES `comp743`.`tb\_customer` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB

AUTO\_INCREMENT = 22

DEFAULT CHARACTER SET = utf8;

## Data base Functions

|  |  |
| --- | --- |
| **Name** | Search products |
| **Function** | SELECT \* FROM tb\_product LIMIT 8 |
| **Description** | Search the products limiting the number of items to 8 |

|  |  |
| --- | --- |
| **Name** | Search Menu |
| **Function** | SELECT \* FROM tb\_menu |
| **Description** | Search the available menus for the system`s menu |

|  |  |
| --- | --- |
| **Name** | Search products by name |
| **Function** | SELECT \* FROM tb\_product WHERE name like CONCAT('$search','%'); |
| **Description** | Search the products, filtering by their name |

|  |  |
| --- | --- |
| **Name** | Check user |
| **Function** | SELECT \* FROM tb\_user WHERE username='$username' AND password='$password' |
| **Description** | Check if the user exists |

|  |  |
| --- | --- |
| **Name** | Change Password |
| **Function** | UPDATE tb\_user  SET password=:password  WHERE Customer\_id=:Customer\_id; |
| **Description** | Change the user password |

|  |  |
| --- | --- |
| **Name** | Get product by category and department |
| **Function** | SELECT tb\_product.\* FROM tb\_product  inner join tb\_department  on tb\_product.Department\_id = tb\_department.id  inner join tb\_category  on tb\_product.Category\_id = tb\_category.id  WHERE tb\_department.name='$departmentName' AND tb\_category.name='$categoryName' |
| **Description** | Get the products from the department and category |

|  |  |
| --- | --- |
| **Name** | Get products by department |
| **Function** | SELECT tb\_product.\* FROM tb\_product  inner join tb\_department  on tb\_product.Department\_id = tb\_department.id  WHERE tb\_department.name='$departmentName' LIMIT 8 |
| **Description** | Get products filtered by the department name |

|  |  |
| --- | --- |
| **Name** | Add customer |
| **Function** | INSERT INTO tb\_customer (first\_name,last\_name,email)  VALUES (:first\_name,:last\_name,:email);  SELECT @pk := last\_insert\_id();  INSERT INTO tb\_address (address, city, state, postal\_code, Customer\_id)  VALUES (:address, :city, :state, :postal\_code, @pk);  INSERT INTO tb\_user (username, password, profile, Customer\_id)  VALUES (:username, :password, :profile, @pk); |
| **Description** | Add customer to the data base |

|  |  |
| --- | --- |
| **Name** | Get customer by id |
| **Function** | SELECT \* FROM tb\_customer WHERE id='$id' |
| **Description** | Get the customer filtered by the id |

|  |  |
| --- | --- |
| **Name** | Get address |
| **Function** | SELECT \* FROM tb\_address WHERE Customer\_id='$id' |
| **Description** | Get the address of the customer |

|  |  |
| --- | --- |
| **Name** | Update customer |
| **Function** | UPDATE tb\_customer  SET first\_name = :first\_name,  last\_name = :last\_name,  email = :email  WHERE id = :id |
| **Description** | Updated customer information |

|  |  |
| --- | --- |
| **Name** | Update address |
| **Function** | UPDATE tb\_address SET  address = :address,  city = :city,  state = :state,  postal\_code = :postal\_code  WHERE id = :id |
| **Description** | Updated customer address |

|  |  |
| --- | --- |
| **Name** | Get order |
| **Function** | SELECT \* FROM tb\_order WHERE Customer\_id = '$Customer\_id' |
| **Description** | Get the order list of the employee |

|  |  |
| --- | --- |
| **Name** | Add order |
| **Function** | INSERT INTO tb\_order (date, status, total, Customer\_id)  VALUES (NOW(),:status,:total, :Customer\_id); |
| **Description** | Add customer order to the database |

# TASK 4 – WEB APPLICATION DEVELOPMENT USING PHP

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | |
|
| **Test Scenario 01** | | Given I am a user, when I click the Sign-up link, then I will be able to create my sign in account. | | | | | |
| **Test Case 01** | | **Pre-Condition: Launch the website then click the Sign in link.** | | | | | |
| **Test Steps** | | **Data** | **Expected Result** | **Test Result Browser:Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Click the Sign-up link on the Sign-in page. |  | Sign-up page should be displayed. | **Passed** |  |
| 2 | Populate all the information required. |  | Textfield should be populated. | **Passed** |  |
| 3 | Click the Sign up button. |  | Sign-in account should be created successfully. | **Passed** |  |
| **Test Scenario 02** | | Given I am a user , when I click the Sign in link , then I will be able to login using my sign-in account. | | | | | |
| **Test Case 02** | | **Pre-Condition: Create a sign-in account.** | | | | | |
| **Test Steps** | | **Data** | **Expected Result** | **Test Result Browser: Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Launch the website. |  | Website should be launched. | **Passed** |  |
| 2 | Click the Sign in link. |  | Sign-in page should be displayed. | **Passed** |  |
|  |  | 3 | Populate the Username and Password. |  | Username and Password should be populated. | **Passed** |  |
|  |  | 4 | Click the Sign in button. |  | Should sign-in successfully. | **Passed** |  |
| **Test Scenario 03** | | Given I am a user, when I navigate to navigation menu and click per menu, then I will be able to see all the designated products and their product description. | | | | | |
| **Test Case 03** | | **Pre-Condition: NA** | | | | | |
| **Test Steps** | | **Data** | **Expected Result** | **Test Result Browser: Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Launch the website. | PS4 XBOX ONE SWITCH PC PS3 XBOX 360 Wii U  PS Vista 3DS | Website should be launched. | **Passed** |  |
| 2 | Navigate to navigation menu. | Navigation menu should be displayed. | **Passed** |  |
| 3 | Click the menu <Data> | List of Products with price should be displayed once the menu<Data> has been clicked. | **Passed** |  |
| 4 | Click the product | Product description should be displayed. | **Passed** |  |
| **Test Scenario 04** | | Given I am a user , when I add a product to the cart, then I will be able to see the product in the cart for placing the order. | | | | | |
| **Test Case 04** | | **Pre-Condition: User should sign-in and select the product to purchase.** | | | | | |
| **Test Steps** | |  | **Expected Result** | **Test Result Browser: Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Click the "Add to Cart" button once product has been clicked. |  | Product should be added successfully to the cart. | **Passed** |  |
| 2 | Click the "Cart" link |  | Purchased product details should be displayed. | **Passed** |  |
| 3 | Click the "Finish" button |  | Order should be placed successfully. | **Passed** |  |
| **Test Scenario 05** | | Given I am a user , when I add a product to the wishlist, then I will be able to see the product to my wishlist and add to the cart and place the order. | | | | | |
| **Test Case 05** | | **Pre-Condition: User should sign-in and select the product to purchase.** | | | | | |
| **Test Steps** | |  | **Expected Result** | **Test Result Browser: Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Click the "Wishlist" icon once product has been clicked. |  | Product should be added successfully to the wishlist. | **Passed** |  |
| 2 | Click the "Wishlist" link |  | Product details should be displayed. | **Passed** |  |
| 3 | Click the "Cart" icon once you already want to purchase the product. |  | Product should be added successfully to the cart. | **Passed** |  |
| 4 | Click the "Cart" link |  | Purchased product details should be displayed. | **Passed** |  |
| 5 | Click the "Finish" button |  | Order should be placed successfully. | **Passed** |  |
| **Test Scenario 06** | | Given I am a user , when I enter text in the serch field and click the search icon, then I will be able to see the terms that match the search I have entered | | | | | |
| **Test Case 06** | | **Pre-Condition: NA** | | | | | |
| **Test Steps** | |  | **Expected Result** | **Test Result Browser: Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Launch the website. |  | Product should be added successfully to the wishlist. | **Passed** |  |
| 2 | Enter text in the search field |  | Search field should be populated. | **Passed** |  |
| 3 | Click the search icon |  | Terms that match the search you have entered should be displayed. | **Passed** |  |
| **Test Scenario 07** | | Given I am a user , when I click the account icon and navigate to My Account ,then I will be able to edit my account. | | | | | |
| **Test Case 07** | | **Pre-Condition: User should sign-in.** | | | | | |
| **Test Steps** | |  | **Expected Result** | **Test Result  Browser: Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Click the account icon |  | Account page is displayed. | **Passed** |  |
| 2 | Click the "My Account" in the account menu |  | Edit account modal is displayed with your personal data registered. | **Passed** |  |
| 3 | Edit your first name, last name and email address. |  | Updated personal data<account> should be displayed. | **Passed** |  |
| 4 | Click the "Save Information" button |  | Pop up message displayed, personal data have been updated automatically. | **Passed** |  |
| **Test Scenario 08** | | Given I am a user , when I click the account icon and navigate to Main Address ,then I will be able to edit my address. | | | | | |
| **Test Case 08** | | **Pre-Condition: User should sign-in.** | | | | | |
| **Test Steps** | |  | **Expected Result** | **Test Result Browser: Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Click the account icon |  | Account page is displayed. | **Passed** |  |
| 2 | Click the "Main Address" in the account menu |  | Main Address modal is displayed with your personal data registered. | **Passed** |  |
| 3 | Edit your address, city, state and postal code |  | Updated personal data<address> should be displayed. | **Passed** |  |
| 4 | Click the "Save Information" button |  | Pop up message displayed, personal data have been updated automatically. | **Passed** |  |
| **Test Scenario 09** | | Given I am a user , when I click the account icon and navigate to change password ,then I will be able to update my password. | | | | | |
| **Test Case 09** | | **Pre-Condition: User should sign-in.** | | | | | |
| **Test Steps** | |  | **Expected Result** | **Test Result Browser: Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Click the account icon |  | Account page is displayed. | **Passed** |  |
| 2 | Click the "Change Password" in the account menu |  | Change Password modal is displayed. | **Passed** |  |
| 3 | Enter your new password |  | Updated password should be entered | **Passed** |  |
| 4 | Click the "Save Information" button |  | Pop up message displayed, password has been updated automatically. | **Passed** |  |
| 4 | Click the "Save Information" button |  | Pop up message displayed, password has been updated automatically. | **Passed** |  |
| **Test Scenario 10** | | Given I am a user , when I click the account icon and navigate to My Orders ,then I will be able to see the purchased items. | | | | | |
| **Test Case 10** | | **Pre-Condition: User should sign-in and has been purchased items successfully.** | | | | | |
| **Test Steps** | |  | **Expected Result** | **Test Result Browser: Google** | **Comment** |
|
| Test Steps | Execution Details |
| 1 | Click the account icon |  | Account page is displayed. | **Passed** |  |
| 2 | Click the "My Orders" in the account menu |  | My Orders modal is displayed with your purchased items. | **Passed** |  |