|  |
| --- |
|  |
| **Final Document Upload** |
|  |
|  |
| Joshua McGrath - X00129753  Jamie Nolan - X00128813  Lukas Meiliunas - X00139670 |
| **22/04/2018** |

|  |
| --- |
|  |

**Table of Contents**

**Survey Overview** . . . . . . . . 3

**Analysis of results** . . . . . . . . 4

**System Narrative** . . . . . . . . 7

**List of Requirements** . . . . . . . 8

**Use Case Diagram** . . . . . . . . 9

**Use Case Descriptions** . . . . . . . 10

**Class Diagram** . . . . . . . . 21

**Erd**  . . . . . . . . . . 23

**Key of Fields** . . . . . . . . .24

**Input / Output Screens** . . . . . . . 26

**Pseudocode** . . . . . . . . . 40

**Appendix** . . . . . . . . . 44

**Test Case Matrix Finished** . . . . . . . 60

Survey Overview:

We as a group conducted an online survey to over 25 people to find out where the aim for our website should be directed towards. The survey consisted of 10 questions which are relevant to different steps of making this website successful.

· How often do you buy Products online?

· What type of furniture do you typically buy online?

· How do you prefer to buy furniture?

· How much would your budget be?

· Do you use PayPal?

· What is your preferred option, Delivery / Collection?

· Do you prefer buying from Company / Person?

· What do you believe is most important?

· Are you familiar with a review and comment system?

· Have you ever had any bad experiences with online shopping?

Analysis:

The first question of the survey is ‘ How often do you buy products online?’ this question was asked to find out if people are actually interested buying products online and if so how often.

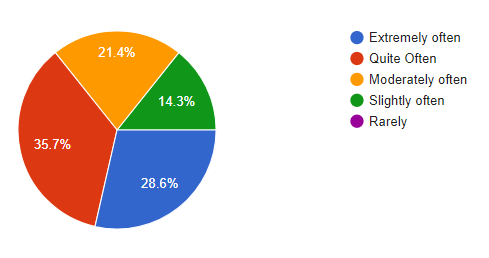


Fig. 1.0

We noticed that from over 15 people 0% of them actually did not use online shopping. This was very surprising to us!

From these unexpected results we asked the participants ‘What type of furniture do you buy online?’ This was probably one of the most important questions in this survey as this was the factor which would later influence our website of what type of furniture to sell.

There was various answers ranging from Bedroom furniture, Decoration, Toilet furniture, Sofa Book casing and lighting. From these results we came up of an idea that the website index page will have various pictures which work as links to different pages leading on to Bedroom, Kitchen, Living room, Toilet, Sitting room.

|  |  |
| --- | --- |
| Fig 1.1 |  |
|  |  |

We were thinking of implementing a ‘PayPal’ service in the transaction process as over 55% of the people who answered this survey used PayPal.

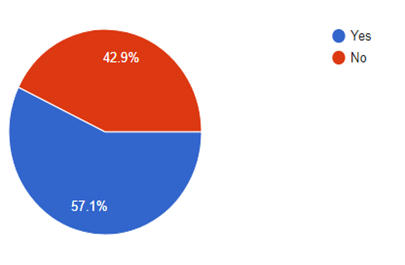


Fig. 1.2

We have already started to research this and we will definitely be implementing this in our website.

We asked ‘What do you think is most important?’ The answers were Security, Trust and privacy.

We asked ‘what would your budget be?’

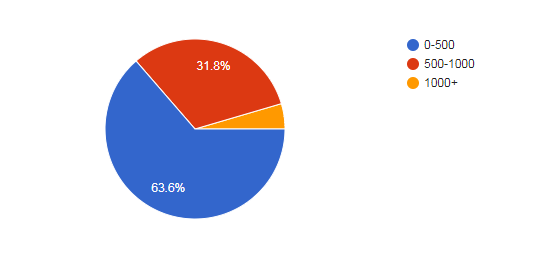


Fig 1.3

The majority of people said their spending would be in between 500 and 1000. This means we will range a big amount of our products on the website off these prices.

System narrative

After gaining the results from the public survey we compiled and put together an analysis of that functions the system should use and have within the web site to reach the users expectations.

The system should allow for creation and editing of a listing of furniture that stores all relevant information regarding the furniture. This listing will be made by the customer or employee with a price that the item owner would like to sell the item for. The system will then add the information to the database and update the in stock amount based upon the listing.

When users first visit the site they will be prompted to sign up the system will provide a form to allow them to enter their name, email address, password, home address, phone number and date of birth. Each users information will be stored in hash form and allocated an ID we will then cross check the info with the ID when a sign in attempt is made. A user will be able to browse search or categorise the searches based on what item they are looking for. Once there account is made they can then make purchases or make listings themselves if they wish of furniture they would like to sell. The user will also the edit or delete their current listings if they wish to do so. The user will also have a cancel order feature if they wish to cancel their order they made.

The administrator account will also have an account that will allow them to moderate, create, cancel and edit furniture listings as well as edit or delete existing accounts. The admin will also be able to generate a report regarding the performance of what has sold and what has not and will show the monthly sales showing the number of company sales and the number of customer-sellers user’s sales.

The employees will also be able to make an account with all the features of the customer account with also the feature to update stock on the website. They also see a 20 percent discount of all listings on the site.

List of requirements

● Add account

● Remove account

● Edit account

● Add listing

● Remove listing

● Edit listing

● Create report

● Apply discount

● Add employee account

● Remove employee account

● Make purchase

● Cancel purchase

● Add to cart

● Remove from cart

● Edit stock available

● View current orders

Use case Diagram

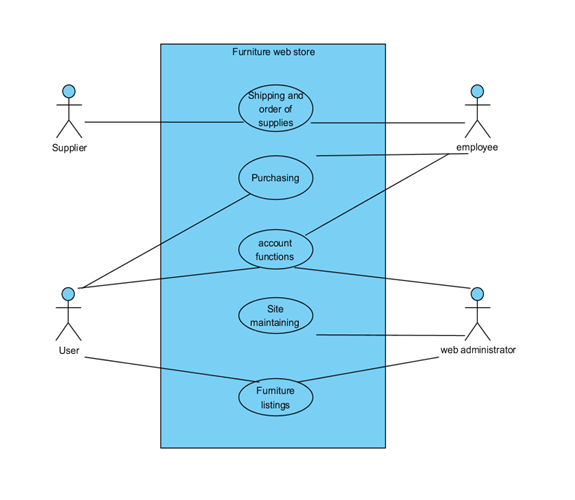


Fig 1.4

Use Case Description:

|  |  |
| --- | --- |
| Use Case: | Purchasing |
| Actor(s): | User |
| Goal: | The goal of the customer is to make a successful online purchase. This is done when the user is logged in and the product is available and is in stock. |
| Overview: | The customer who has already been authenticated and is logged in clicks a button on the website which adds that product to a cart via a p.getId method that selects that specific product and adds it to a cart that is created once they select addProduct. The user is shown the ability to remove add one delete one and empty cart from this point these will all be buttons. The user selects purchase the product is deducted 1 from stock and they are met with the order receipt page that is saved to their accounts. |
| Pre-  Condition: | Website is functioning, user has logged in and item is available. |
| Post-  Condition: | The item is successfully purchased, receipt is created. |
| Successful  Scenario: | 1.User Logs into their account by selecting the log in button and inputting the correct details  2. Browses for the item they want.  3. Customer selects Item and it will then grab the item from the database and add it to the customers cart  4. Payment and shipping information is accepted  6. Stock is depleted by 1 |
| Alternative  Scenario(s): | 4. Payment is accepted  The user does not have the funds payment is declined.  3.User places order for an item  The item is out of stock and the order has failed. |

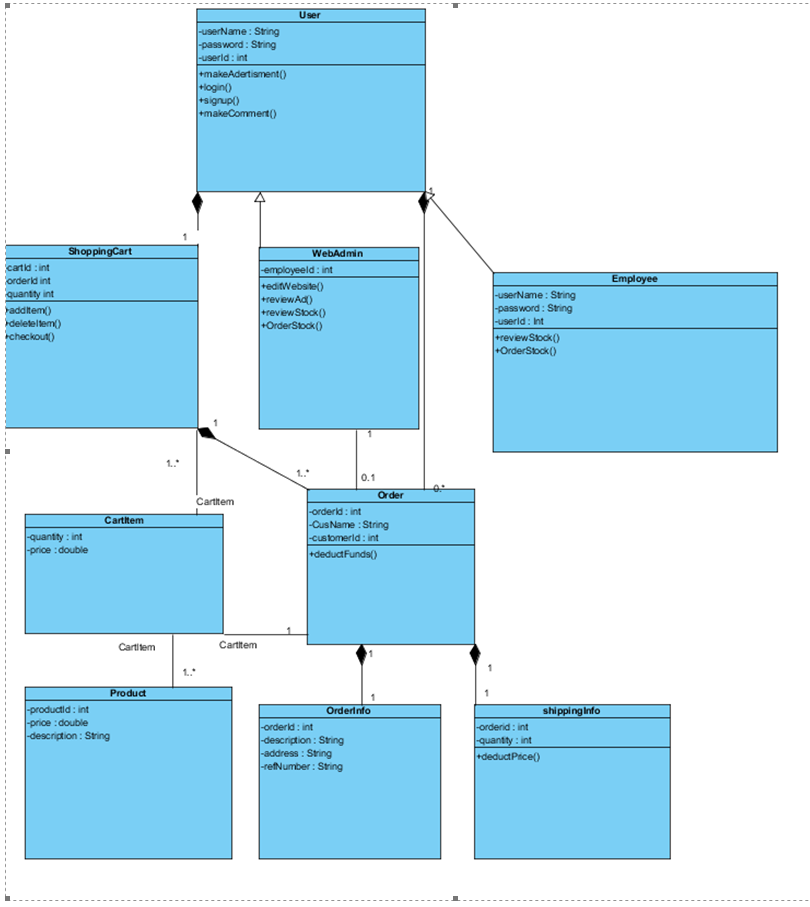
|  |  |
| --- | --- |
| Use Case: | Account Functions |
| Actor(s): | User,admin |
| Goal: | The user can edit their password and view their orders |
| Overview: | The user wants to view orders and change password abilities. |
| Pre-  Condition: | The user is connected to the internet and meets the requirements to sign up and knows the information required to log in that they have set themselves |
| Post-  Condition: | The user accesses the account functionality of the website successfully |
| Successful  Scenario: | 1. The user logs in  2. The login details are successful  3. The user is logged into the website with authentication  4. The user selects from the navbar account details  5. The user has been verified to access this page  6. The user is displayed a webpage where they can edit their password and view their orders. |
| Alternative  Scenario(s): | 2. The user enters details  The user enters incorrect details and login is denied making them unable to access the sites Account functionality |

|  |  |
| --- | --- |
| Use Case: | Site maintaining |
| Actor(s): | Web administrator |
| Goal: | The Web administrator wants to make a change or fix a problem on the website. The Web administrator will get informed of the problem and log into the privileged account to fix this problem. The Web administrator will log in make a change or a fix and then publish the website when the change has been finished. The web admin also wants the ability to edit the design of the website and this is achieved by logging into privileged access that the Web administrator has. |
| Overview: | The Web administrator tweaks the website and fixes problems with the website when a problem or required tweak arises |
| Pre-  Condition: | The Web administrator is logged in and has sufficient privileges to make changes to the website’s code. |
| Post-  Condition: | The website is maintained by the Web administrator |
| Successful  Scenario: | 1. The Website Administrator logs into their privileged account  2. The Website Administrator has logged in successfully and has privileged access to the website  3. The Administrator makes appropriate changes/fixes to the website |
| Alternative  Scenario(s): | 1 The Website Administrator logs into their privileged account  The Website Administrator has forgot their details and login is denied. |

|  |  |
| --- | --- |
| Use Case: | Furniture Listings |
| Actor(s): | User, Web administrator |
| Goal: | The User is applying to create an advertisement on the website and is doing so via the Account functionality of the website they gained from logging into their accounts. The Web administrator will verify with the requirements of an advertisement before accepting the users advertisement. The users advertisement is accepted and published on the website. |
| Overview: | The User wants to put an advertisement on the website and the Web administrator verifies this is suitable for the website and accepts the Advertisement |
| Pre-  Condition: | The User has a product to sell with required information |
| Post-  Condition: | The User advertisement is successfully placed on the website |
| Successful  Scenario: | 1.The User has a product to sell and requests to put it on the website  2.The Web Administrator checks if the advertisement is suitable before publishing it on the website  3.The Advertisement is published |
| Alternative  Scenario(s): | 1. The Web administrator checks if the advertisement is suitable before publishing it on the website  The Product is not suitable and is removed from the website, advertisement failed. |

|  |  |
| --- | --- |
| Use Case: | Shipping and order of supplies |
| Actor(s): | Employee, Supplier |
| Goal: | To Re-Order supplies once the stock has diminished by way of contacting the supplier with information on the stock that needs to be supplied and then purchasing and getting these supplies shipped |
| Overview: | The stock is low/null and is due for a resupply. The Employee will resupply the stock and the stock will become available on the website listing for the user to see |
| Pre-  Condition: | The Employee is aware that the stock is low on a company’s item |
| Post-  Condition: | The Stock is resupplied |
| Successful  Scenario: | 1.The Employee becomes aware that the stock is low  2.The Employee arranges more stock to be bought by contacting the company  3.The stock is purchased and the company starts to ship the stock  3.The items are restocked |
| Alternative  Scenario(s): | 2.The stock manager arranges more stock to be bought  The item is unavailable to get for a time and the stock cannot be re-supplied |

Class diagram:



**Section 1:**

**Overview of Work done**

**Survey Overview - Lukas Meilunas**

**Analysis of results - Lukas Meiliunas**

**System Narrative – Josh McGrath**

**List of Requirements – Josh McGrath**

**Use Case Diagram – Josh McGrath**

**Use Case Description - Jamie Nolan**

**Class Diagram - Jamie Nolan**

**Erd diagram**

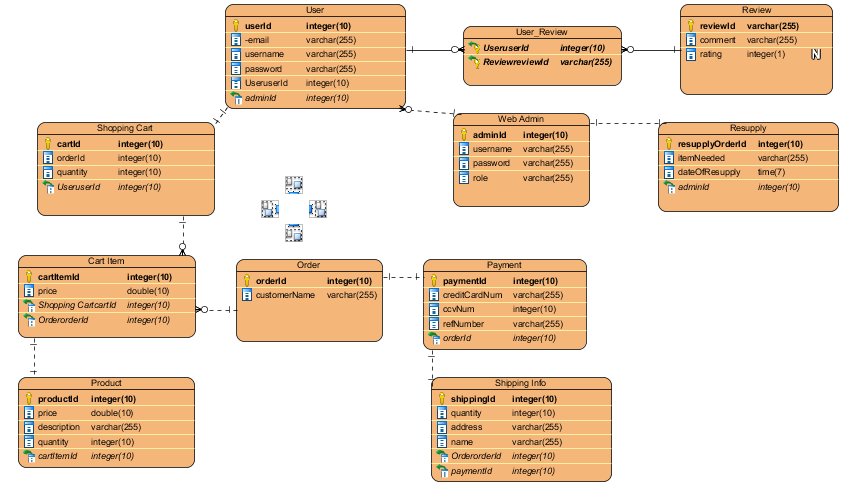
****

Fig 1.6

**KEY OF FIELDS:**

**Int = Integer varchar = Varcharr2(255) double=double String = String**

**PK = Primary Key FK = Foreign Key**

**USER**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| userId PK | Email | Username | Password | cartId FK | adminId FK |
| int | Varchar IS NOT NULL | Varchar IS NOT NULL | Varchar IS NOT NULL | int | int |

**SHOPPING CART**

|  |  |  |  |
| --- | --- | --- | --- |
| cartId PK | orderId | Quantity | userId FK |
| int | number | Int IS NOT NULL | int |

**CART ITEM**

|  |  |  |  |
| --- | --- | --- | --- |
| cartProductId | price | cartId FK | orderId FK |
| int | double | int | int |

**PRODUCT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ProductId PK | Price | description | rating | cartId FK |
| int | double | String | int | int |

**ORDER**

|  |  |  |
| --- | --- | --- |
| orderId PK | custName | shippingId FK |
| int | String | int |

**SHIPPING INFO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| shippingId PK | quantity | address | name | orderId FK |
| Int | Int IS NOT NULL | Varchar2 IS NOT NULL | Varchar2 IS NOT NULL | int |

**ORDER INFO**

|  |  |
| --- | --- |
| orderId PK | custName |
| int | varchar2 |

**WEB ADMIN**

|  |  |  |  |
| --- | --- | --- | --- |
| adminId PK | username | password | role |
| int | String IS NOT NULL | String IS NOT NULL | Varchar2 IS NOT NULL |

**PAYMENT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| paymentId PK | creditCardNum | ccvNum | refNum | orderId FK |
| int | varchar2 IS NOT NULL | varchar2 IS NOT NULL | Varchar2 IS NOT NULL | int |

**REVIEW**

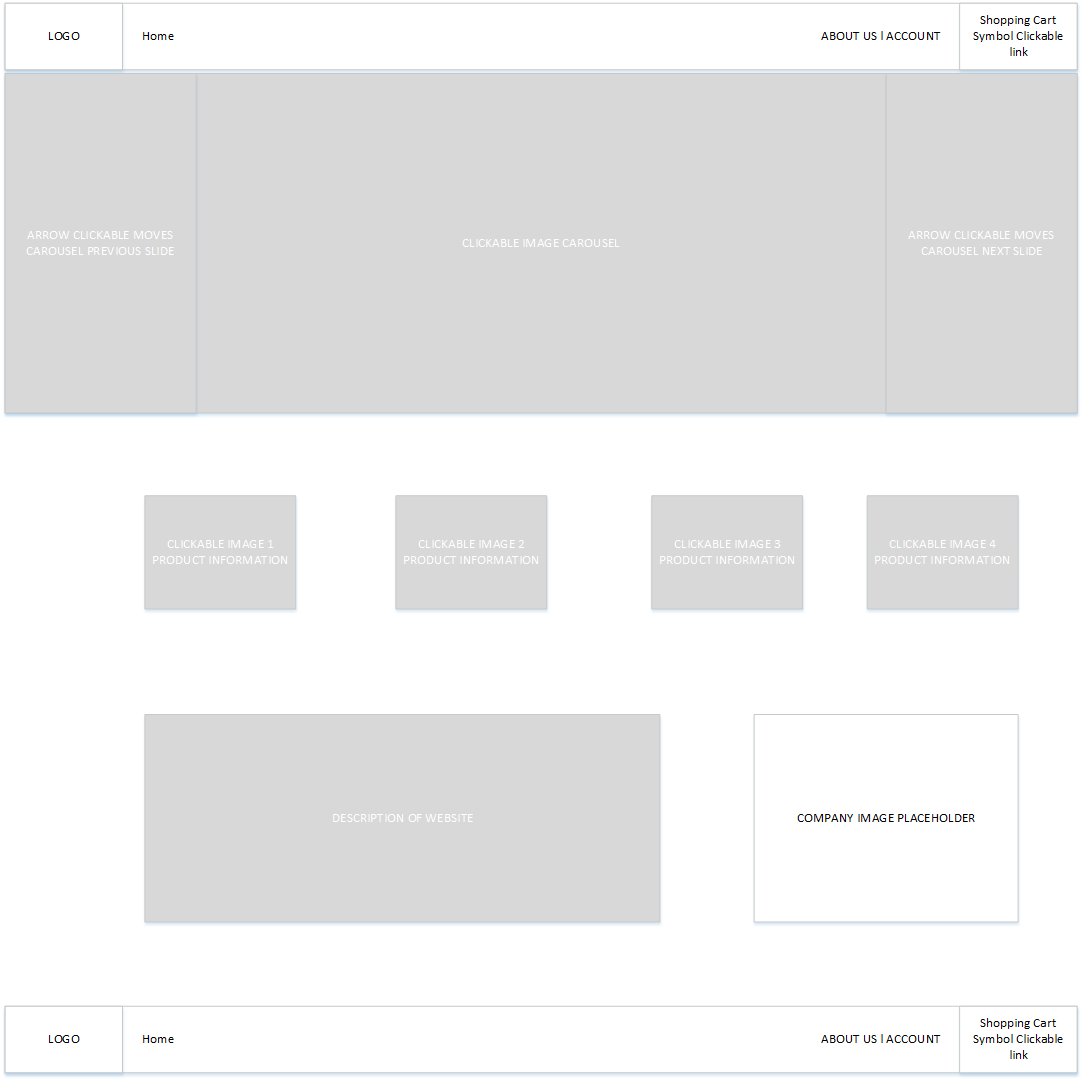
|  |  |  |
| --- | --- | --- |
| reviewId | comment | rating |
| int | varchar2 IS NOT NULL | Int (1) |

**RESUPPLY**

|  |  |  |  |
| --- | --- | --- | --- |
| resupplyOrderId PK | itemNeeded | dateOfResupply | adminId FK |
| int | varchar2 IS NOT NULL | Int (1) | int |

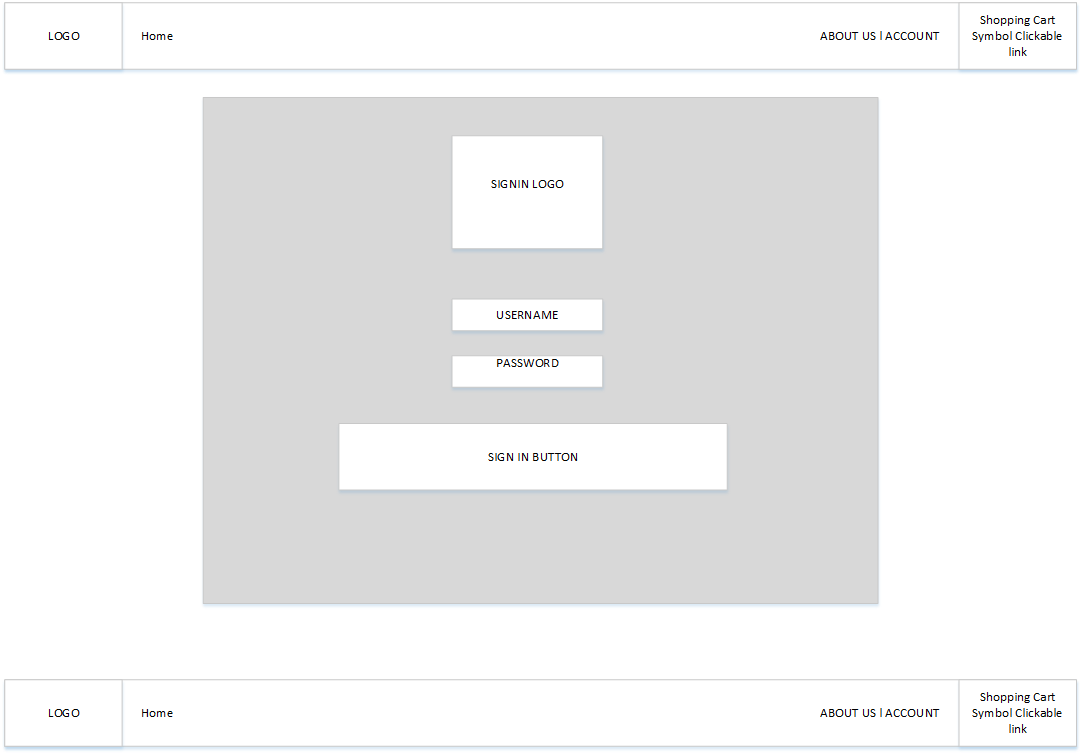
**INPUT/ OUTPUT**

**Home Page**

****Fig 1.7

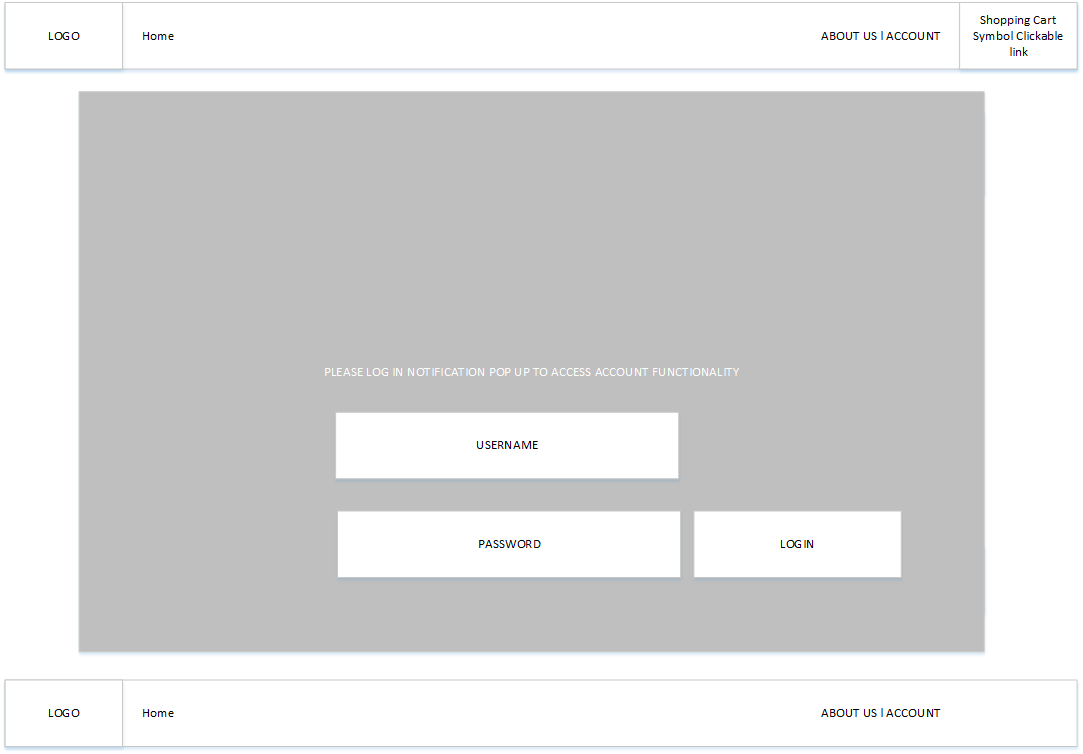
This is our planned design for the index page. The image carousel is to make our website attempt to keep a user on our site while also being able to construct a site such as this. The clickable links on the page include the Navbar home button, About us, Account, Shopping Cart and the clickable images in the middle of the page will bring you to the suitable product page depending on the image the user would’ve clicked.

**LOGIN**

****

The login page is opened when the user selects account but is not logged in and this page is displayed. The login information is entered and it is compared to the information in the database.

**LOGIN POP UP<IF REQUIRED>**

****Fig. 1.8

The account functionality is accessed with a login pop up if the user is not logged in. The user will enter their details and be logged into the website

**LOG IN NOTIFICATION**

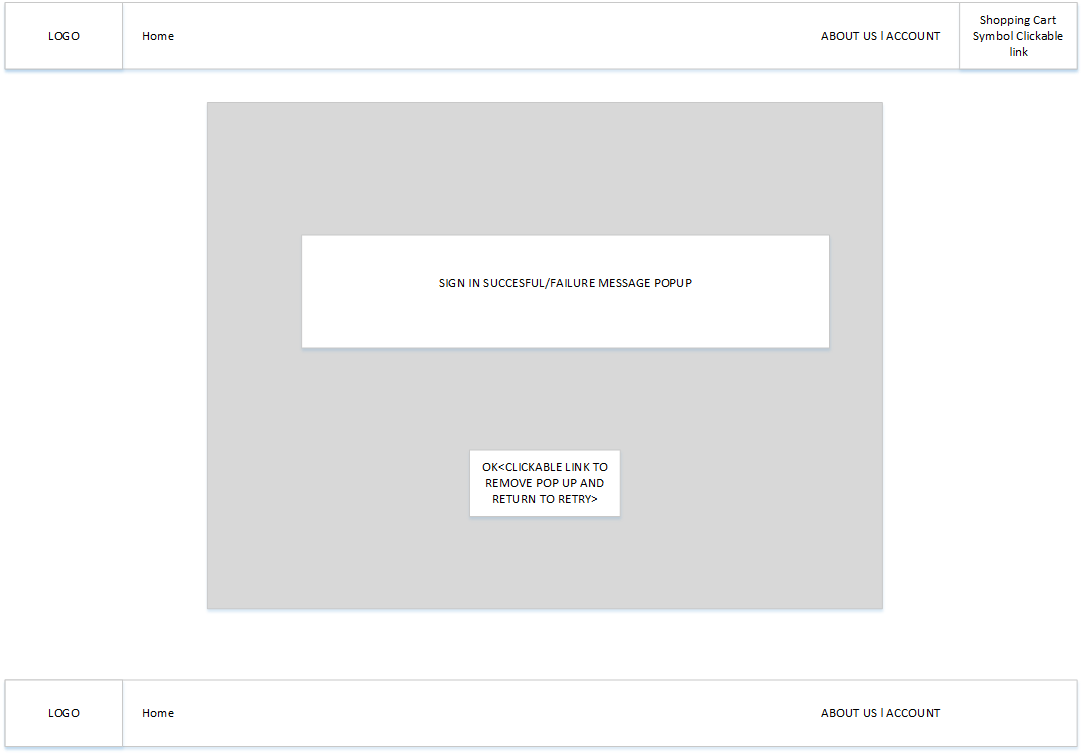


Fig. 1.9

The user would then be displayed a screen if the login was successful or not and the OK button brings the user back to their page.

**ACCOUNT FUNCTIONALITY PAGE**

\*\*SEE APPENDIX

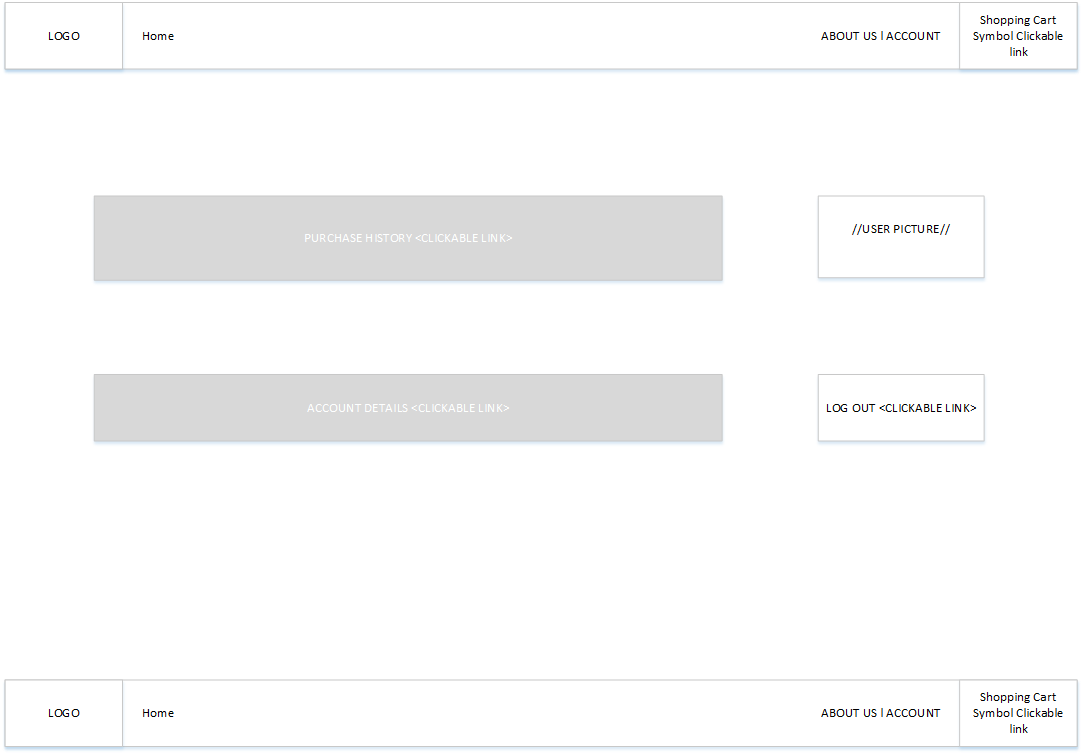


Fig. 2.0

The user can now access their account page. On this page the user can enter their purchase history, account details and log out as well as the nav bars to browse the site.

**ACCOUNT DETAILS EDIT**

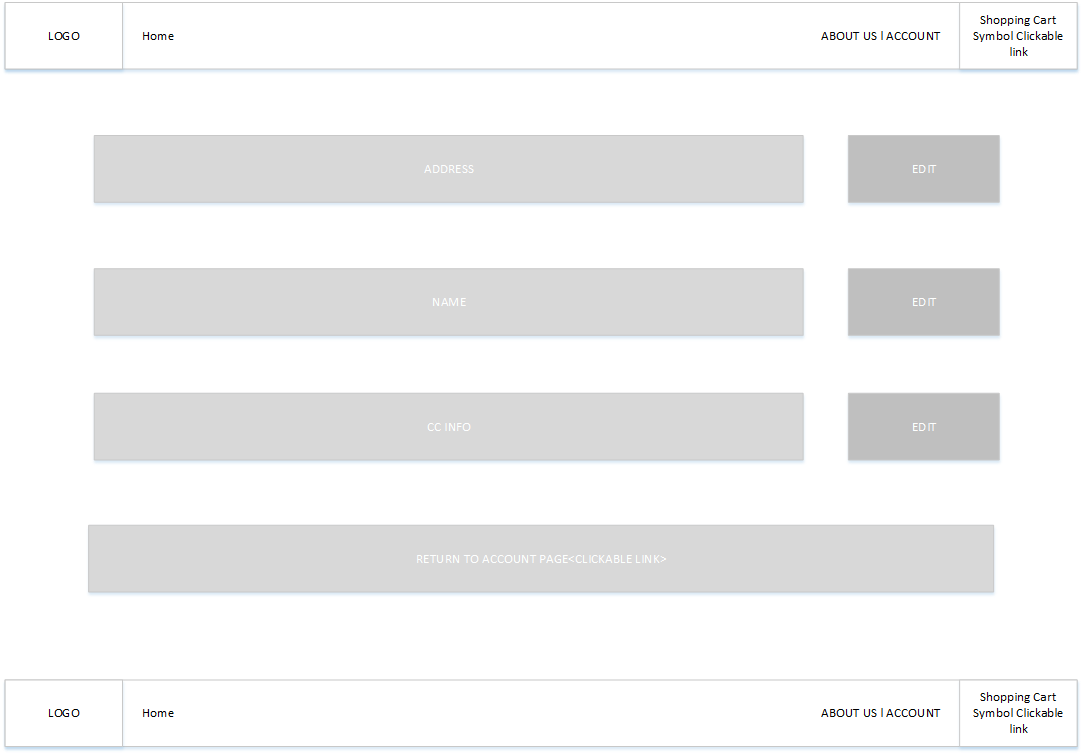
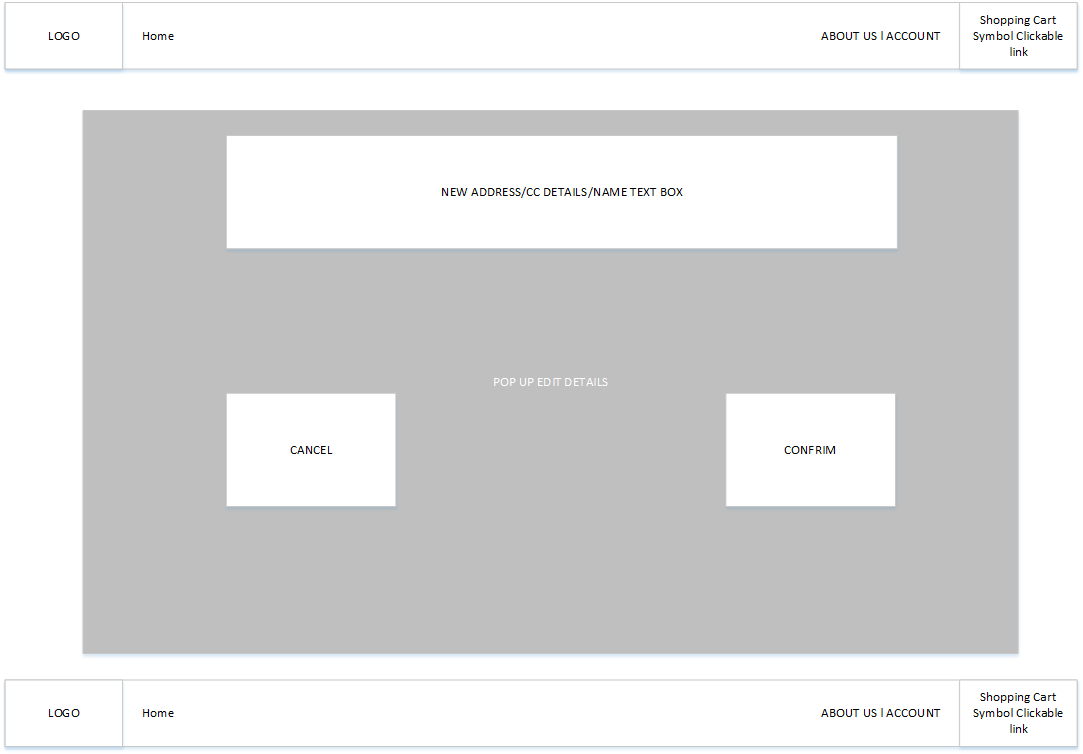


Fig. 2.1

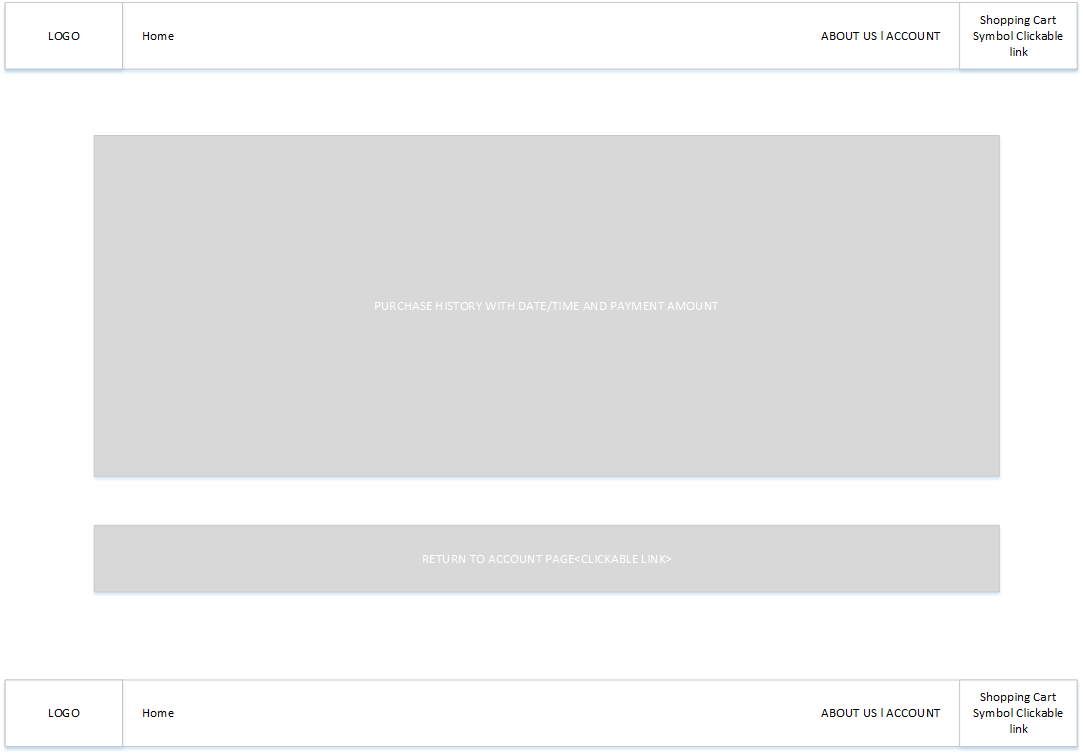
The user has selected account details, this brings them to this page where the user may edit their details by clicking the links and a pop up will ask them to enter their new details.

**ACCOUNT DETAILS EDIT POP UP**



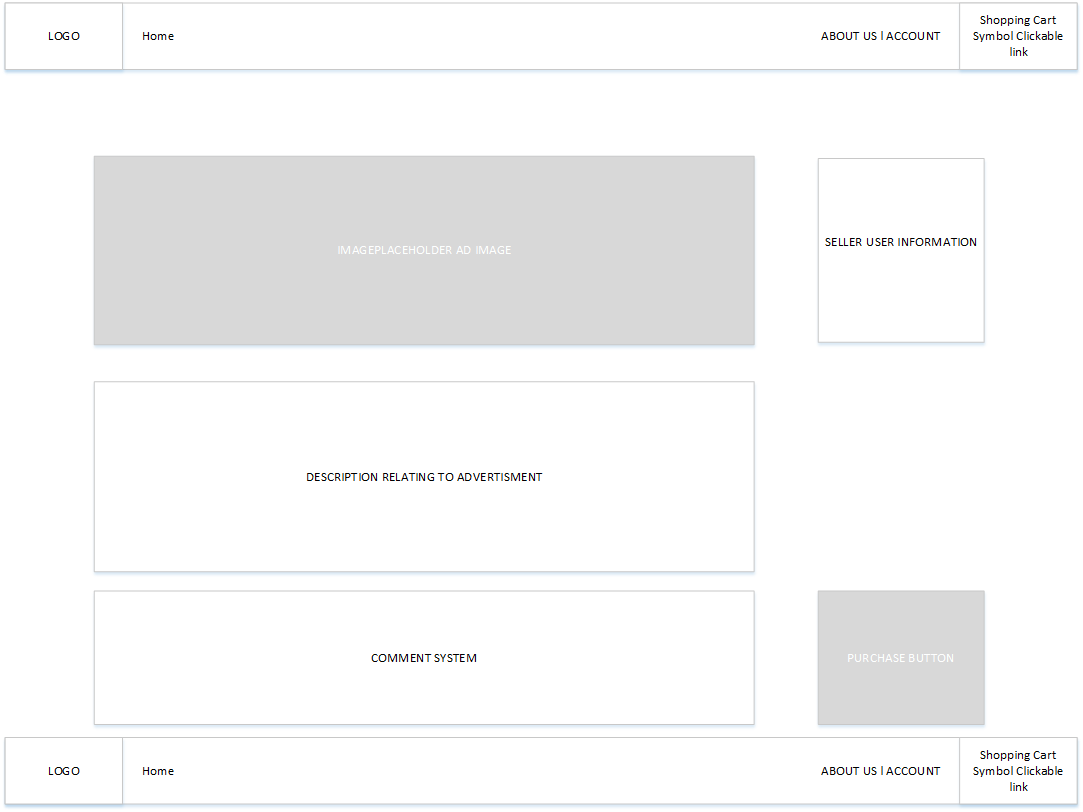
This shall be the pop up that will be displayed to the user in which the user can now input their new details and confirming it or cancelling it will bring the user back to the previous page.

**PURCHASE HISTORY PAGE**



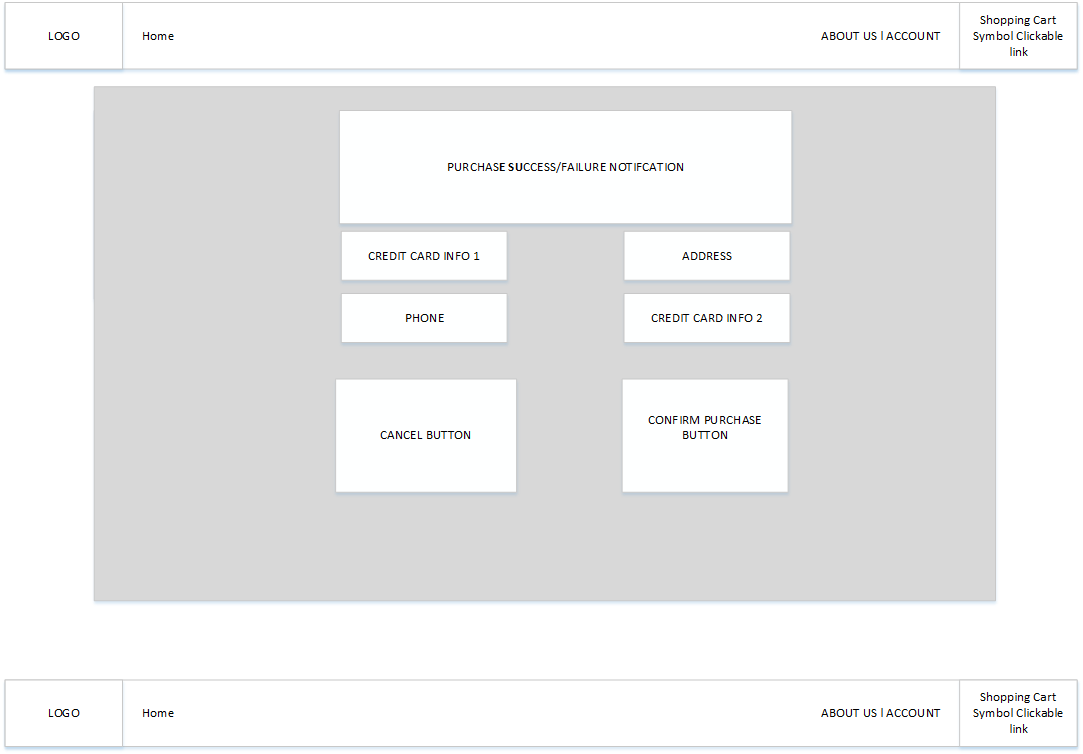
If the user selects the Purchase information option they are brought to this page where they can view their purchase history.

**ADVERTISEMENT**



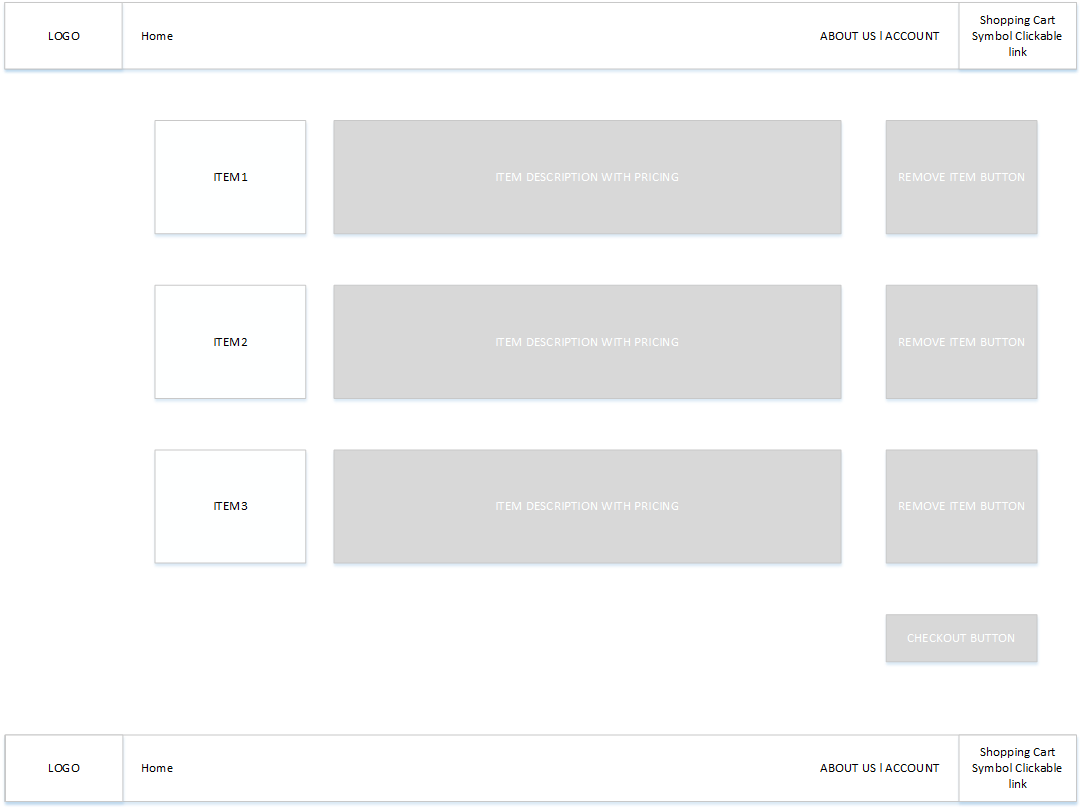
Once the user finds an ad that they like for a product the user clicks into the advertisement. The advertisement page is displayed and the user is shown an image for an ad, the seller information, the description of the product and the purchase button. The user can also make a comment underneath the advertisement where if it is a “user-seller” the user could make an offer in this comment, for example.

**PURCHASE POP UP**



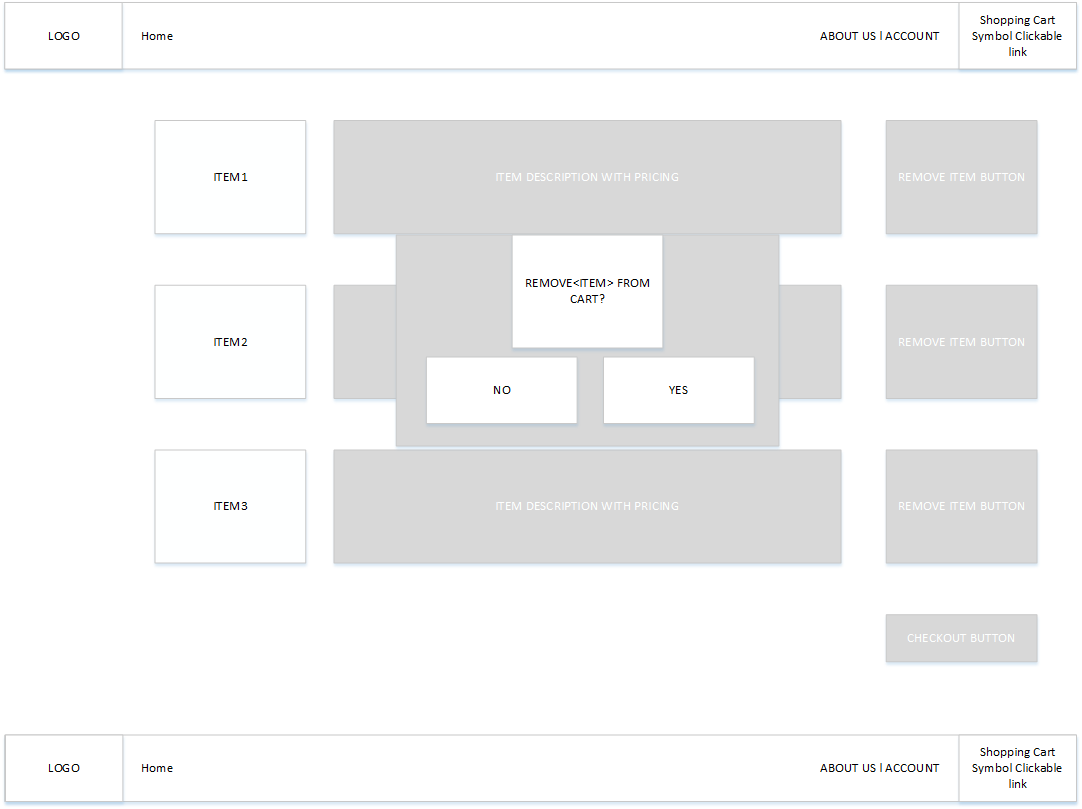
Once the user finds a product that they want they can select the purchase button which prompts them with a pop up where they must enter their details for the purchase of the product. Once the user enters these details and selects the confirm purchase button they will be notified if the purchase was successful or if the purchase was failed. The cancel button closes the popup and cancels the purchase. The purchase, if successful, is then added to the users purchase history. The product is also added to the cart.

**SHOPPING CART**



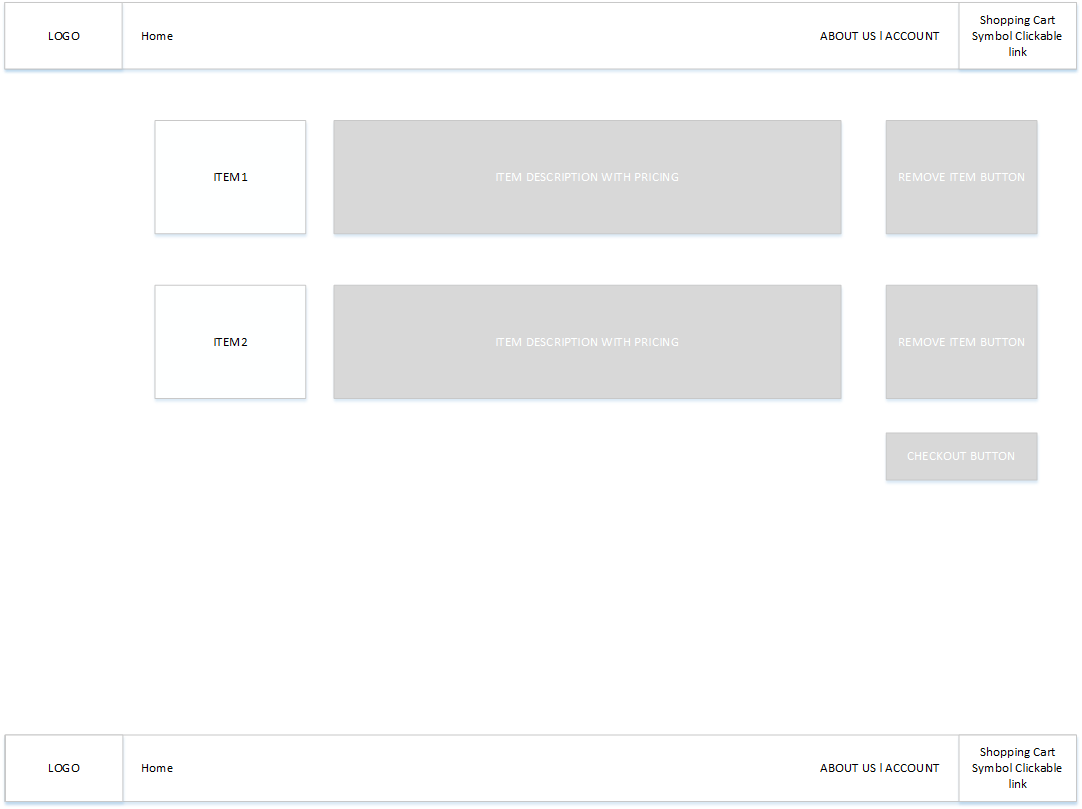
The shopping cart link that is always available in the navbar and footer displays the current items in the logged in users cart. The item image is displayed beside the description/price of the item with a button that will remove the product from the cart. The checkout button pops up the confirm purchase pop up.

**SHOPPING CART ITEM REMOVE**



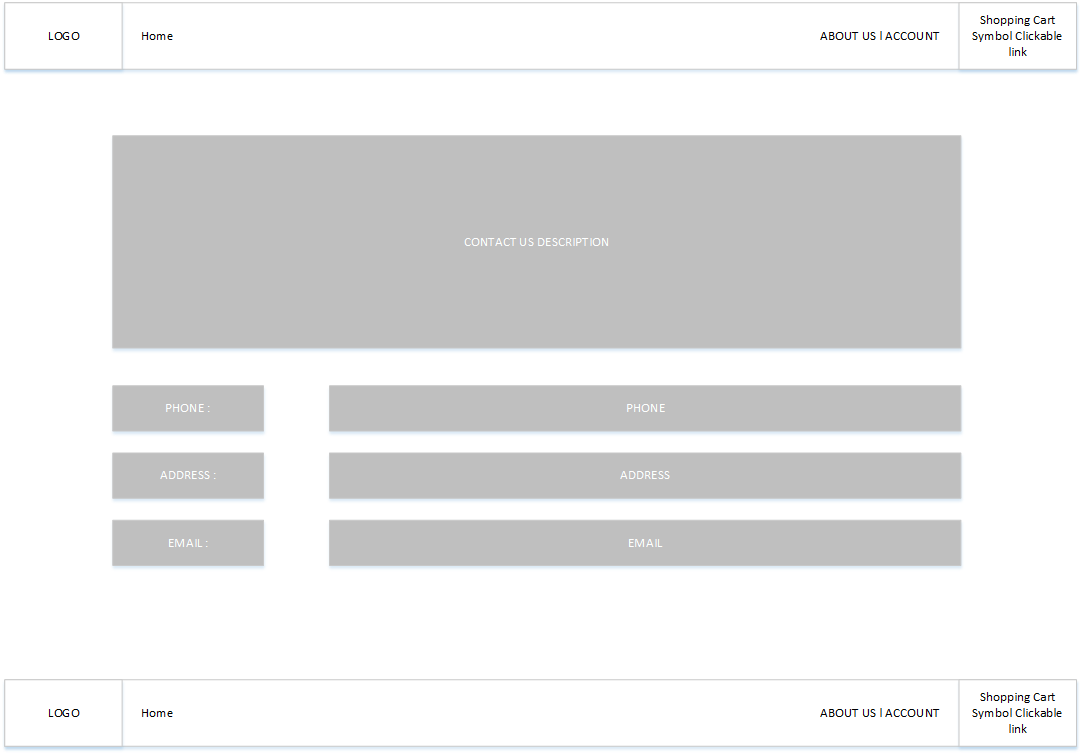
The button remove item is selected and this prompts the user with a pop up to remove the item from the cart as displayed in the design. Clicking yes will remove the product from the cart while no removes the pop up.

SHOPPING CART AFTER REMOVE



As shown here this is the cart after a product is removed with the checkout button moving to the appropriate location to make the design look visually pleasing.

**CONTACT US PAGE**



The contact us page is shown in the design above with the intention of making the page as easy to read as possible. The user can return to the advert section of the site by clicking the home button.

**PSEUDOCODE**

Add User

Output to screen: Enter user first name

input user first name

Output to screen: Enter user last name

input user last name

Output to screen: Enter user address

input user address

Output to screen: Enter user phone

input user phone

Output to screen: Enter user email

if (user email) found

Output to screen: user already exists

else

input user email

user id created

Output to screen: user created successfully

Update user

Output to screen: Update user True/False

input True

Output to screen: Please enter the user id

input user id

If (User id) exists

Output to screen: Please enter which details you wish to update

input details

Output to screen: Please enter new value

input new value

Output to screen: User details updated successfully

else

Output to screen: User does not exist

}

Delete User

Prompt user to enter User id

input id

if(id is found)

then

Prompt the user and ask if they want to delete the User

if(user)

user is removed from system

output to screen: user is deleted

else

exit to user interface

Else

(id not found)

Prompt the user to re-enter details

End if

End if

}

Log In

Prompt User to enter username

Input username

Prompt User to enter password

Input password

if(username && password is found){

Output: you have successfully logged in.

}else{

Output: Incorrect password or username.

Prompt user to input again.

Input Username

Input password

}

Log Out

if(User selects logout function){

Output: Are you sure you want to Log Out?

User presses Yes

User successfully logs out of account.

}else{

User presses no

User is redirected back to home page.

}

Make Comment

if(User selects to make comment){

Prompt user to input sentence

Input message

Prompt user to select rating

Input rating

Output: Review made.

}else {

User not logged in

Comment cannot be made

Prompt user to log in

}

Make Purchase

if(user is logged in && product is in cart){

Prompt user to input credit card details

Input credit cart details

Prompt user to input address

Input address

Output: Product Purchased!

}else{

No product in cart.

Output: no product in your shopping cart!

Output: Add items to cart to make purchase!

User redirected to home screen.

}

Receive Receipt

User successfully purchased product

Output: You have purchased item:

Output product

Output: Shipping address:

Output shippingAddress

Output: Reference number:

Output: referenceNumber

Add to Cart

if(User selects product in stock){

Shopping cart updated

Else{

Output: Product not in stock

}

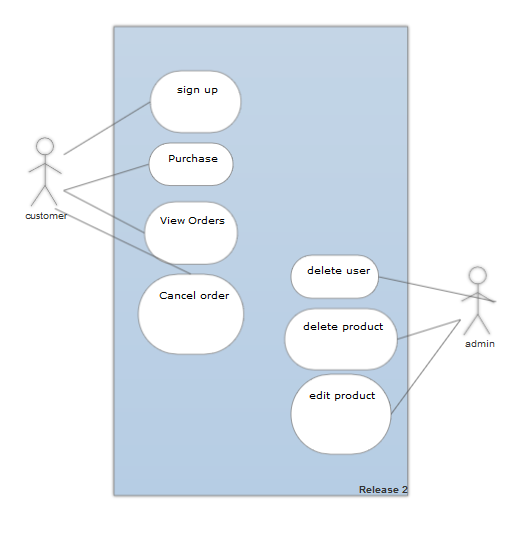
Remove from Cart

User selects product delete

cartItem updated;

**APPENDIX**

**Use Case Diagram - Final Version**

****

**CHANGED USE CASE DESCRIPTIONS TO NEW USE CASE DIAGRAM:**

|  |  |
| --- | --- |
| Use Case: | Sign up |
| Actor(s): | customer |
| Goal: | The goal is to allow the user to sign up so they can use the website with full functionality |
| Overview: | The unregistered user wants to access the sites full functionality and to do so will register an account on the website |
| Pre-  Condition: | User has an email to sign up to the website |
| Post-  Condition: | The website database is up so the users information can be stored |
| Successful  Scenario: | 1. User selects register on the navbar of the website to create an account 2. User inputs suitable details to the website 3. The details are checked to see if they meet constraints e.g is it an email 4. The details are accepted into the database |
| Alternative Scenario | 3 The details are checked to see if they meet constraints e.g is it an email  The email does not have an “@” symbol so the user cannot signup without an email. |

|  |  |
| --- | --- |
| Use Case: | View Orders |
| Actor(s): | customer |
| Goal: | The customer wants to view their orders |
| Overview: | The customer has made a purchase on the website and is wants to view the order receipt which is located in the account page. |
| Pre-  Condition: | Order was successful on a product |
| Post-  Condition: | The order is displayed to the customer |
| Successful  Scenario: | 1. The user has made an order 2. The user selects Account to access their account functionality 3. The user selects view orders 4. The orders are displayed to the user |
| Alternative Scenario | 4. The orders are displayed to the user:  The user has no orders so this fails and no orders are displayed |

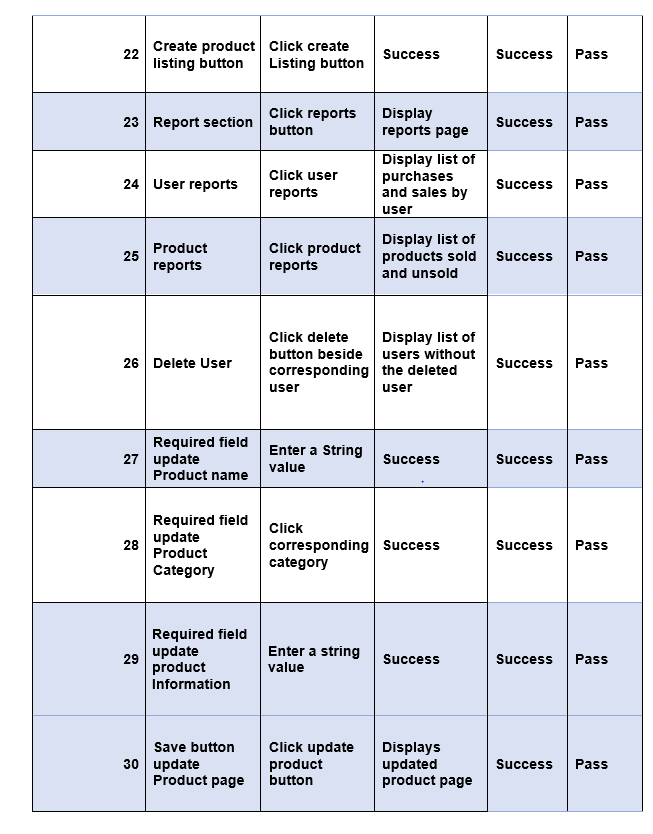
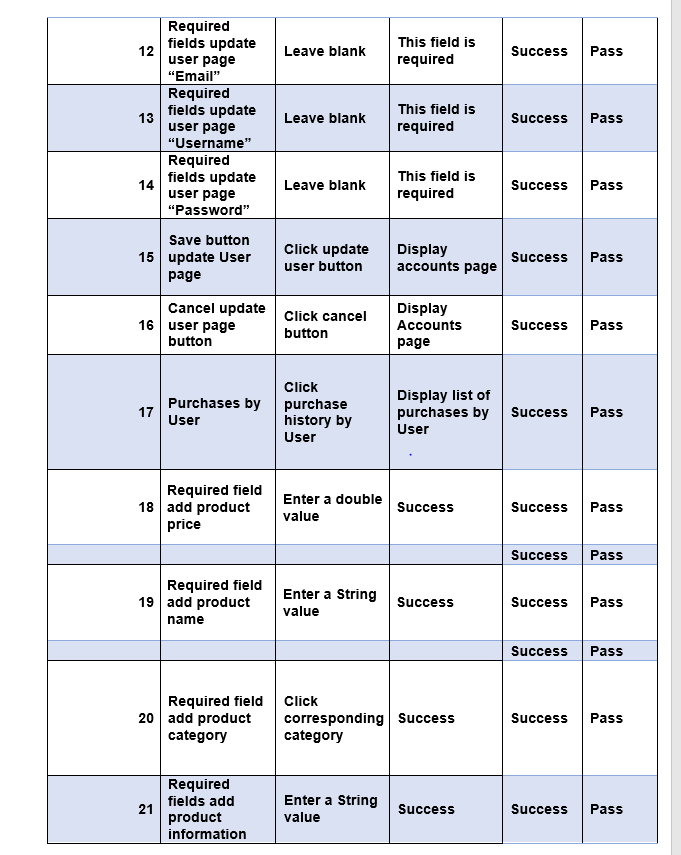
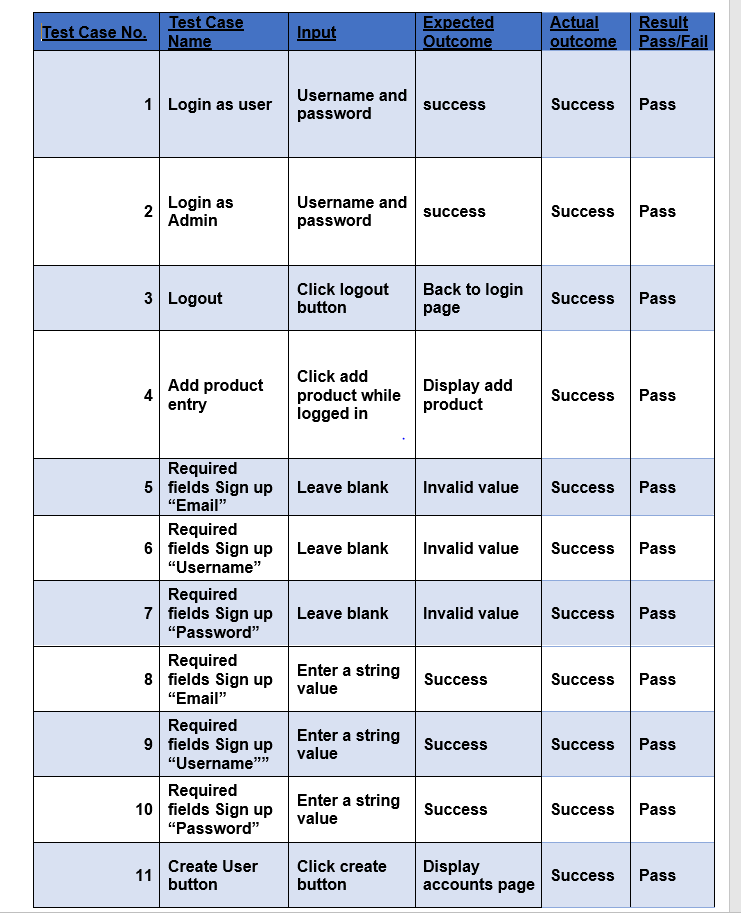
|  |  |
| --- | --- |
| Use Case: | Cancel Order |
| Actor(s): | Customer |
| Goal: | The goal of this is for the Customer to successfully cancel an order which is already pending. |
| Overview: | Customer wants to cancel already pending order. |
| Pre-  Condition: | Customer is logged in. Has purchased items with an order already pending and processing to shipping. |
| Post-  Condition: | The order is successfully cancelled, products put back into stock. |
| Successful  Scenario: | 1. User logs in.  2. Selects the Account page  3. Presses view or cancel orders hyperlink  4. All of the customers orders are displayed in a table  5. Selects to cancel current order  6. Order is successfully cancelled and stock is incremented. |
| Alternative Scenario | 5. Selects to cancel current order  The order was confirmed more than 3 hours ago.  6. Customer is unable to cancel his current order. |

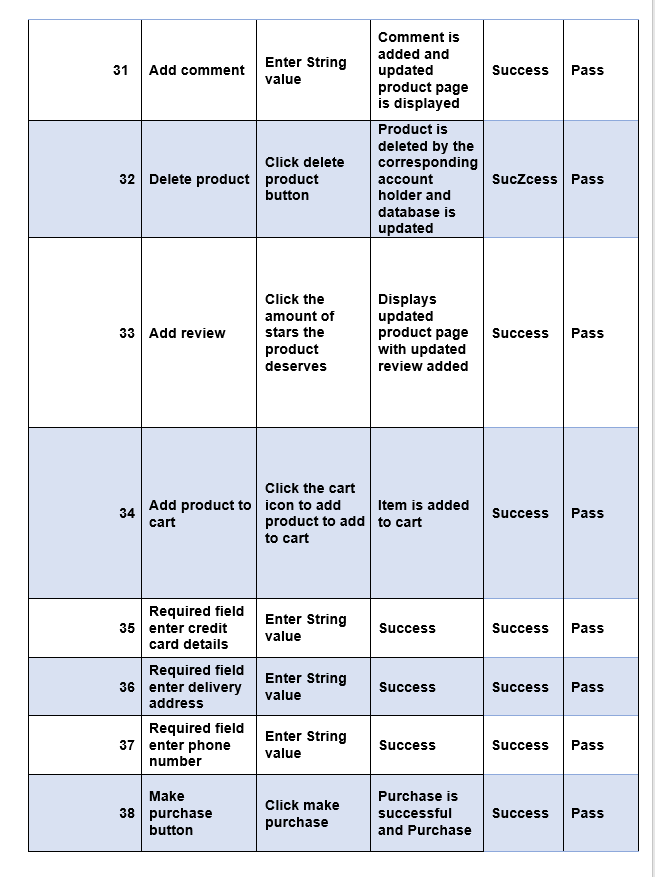
|  |  |
| --- | --- |
| Use Case: | Add Product |
| Actor(s): | Administrator |
| Goal: | The goal of the Administrator is to successfully add a product into the system so it is displayed for sale. |
| Overview: | An Administrator would like to create a new listing (Product) and display it for sale on the website. |
| Pre-  Condition: | User has admin privileges, has all information of the product ready to be put into website. |
| Post-  Condition: | Item is successfully added and is displaying on the product listing page. |
| Successful  Scenario: | 1.User logs in with admin privileges.  2. Selects Admin Control page  3. Selects Add a new Listing.  4. Fills out all details correctly.  5. New Product is added. |
| Alternative Scenario: | 4. Form is not filled out correctly.  5. Form displays error.  6. Details have to be re entered. |

|  |  |
| --- | --- |
| Use Case: | Delete Product |
| Actor(s): | Admin |
| Goal: | The goal is to delete a product as an admin |
| Overview: | There is a product that is no longer on the website and will not be restocked so the admin is going to delete this project from the database |
| Pre-  Condition: | Database is live and the admin is logged in |
| Post-  Condition: | The item exists |
| Successful  Scenario: | 1. The admin logs into an administrator account 2. The admin accesses their admin functionality page 3. Selects the product from a table which lists all the products 4. The admin deletes the product by a button beside the product 5. The product has been deleted |
| Alternative Scenario | 4:The admin deletes the product by a button beside the product  The database is unable to delete the product as it has just been purchased by a user |

|  |  |
| --- | --- |
| Use Case: | Delete user |
| Actor(s): | admin |
| Goal: | The goal is to delete a user from the database |
| Overview: | The admin wants to delete a user for reasons such as an inappropriate name |
| Pre-  Condition: | Database is live and admin is logged in |
| Post-  Condition: | The user has been deleted |
| Successful  Scenario: | 1. The admin logs into their account 2. Accesses admin functionality 3. The admin finds the user which is breaking site rules 4. The admin deletes the user |
| Alternative Scenario: | 4.  The database is down for maintenance and the user cannot be deleted until it is live again |

|  |  |
| --- | --- |
| Use Case: | Change Password |
| Actor(s): | user |
| Goal: | The user wants to change the password they have |
| Overview: | The user wants to change their password to something more memorable and replace it with the current password in the database |
| Pre-  Condition: | The user is logged into their account |
| Post-  Condition: | The user has changed their password and it replaces the old password in the database |
| Successful  Scenario: | 1. The user logs into their account 2. The user accesses their account functionality page 3. Selects change password 4. Inputs a new password 5. The password replaces the old password in the database |
| Alternative Scenario: | 4 The password is not suitable and cannot be changed |





**CHANGED KEY OF FIELDS**

basket (

id bigint auto\_increment not null,

customer\_email varchar(255),

constraint uq\_basket\_customer\_email unique (customer\_email),

constraint pk\_basket primary key (id)

)

category (

id bigint auto\_increment not null,

name varchar(255),

constraint pk\_category primary key (id)

)

comment (

comment\_id bigint auto\_increment not null,

commenting varchar(255),

rating integer not null,

p1\_id bigint,

constraint pk\_comment primary key (comment\_id)

)

order\_item (

id bigint auto\_increment not null,

order\_id bigint,

basket\_id bigint,

product\_id bigint,

quantity integer not null,

price double not null,

constraint pk\_order\_item primary key (id)

)

payment (

id bigint auto\_increment not null,

shipment\_address varchar(255),

cc\_number varchar(255),

ccv varchar(255),

customer\_email varchar(255),

constraint pk\_payment primary key (id)

)

product (

id bigint auto\_increment not null,

name varchar(255),

price double not null,

description varchar(255),

stock integer not null,

category\_id bigint,

constraint pk\_product primary key (id)

)

shop\_order (

id bigint auto\_increment not null,

order\_date timestamp,

customer\_email varchar(255),

constraint pk\_shop\_order primary key (id)

)

user (

role varchar(255),

email varchar(255) not null,

password varchar(255),

first\_name varchar(255),

last\_name varchar(255),

constraint pk\_user primary key (email)

)

**Altered class diagram**

