### **FPT ACADEMY INTERNATIONAL**



**FPT – APTECH COMPUTER EDUCATION**

Sneakers Online

|  |  |  |  |
| --- | --- | --- | --- |
| **Center Name:** | FPT – Aptech – HCM2 | | |
|  | Address:590 Cach Mang Thang Tam,  District 3, Ho Chi Minh City | | |
| **Instructor:** | Mr. Hoang Duc Quang | | |
| **Class:** | T1.2008.M0 | | |
| **Group:** | 01 | | |
| **Group members** | Name  Nguyen Son Ngoc | | Student ID  Student1273498 | |
|  | Nguyen Anh Tai | | Student1270670 | |
|  | Nguyen Ha Vy  Phan Thanh Nam | | Student1273518  Student1273834 | |
|  |  | |  | |
| **Project Duration** | Start date | 02nd June 2021 | | |
|  | End date | 02nd July 2021 | | |

**This is to certify that**

**Mr./Ms.:**

Nguyen Son Ngoc

Nguyen Anh Tai

Nguyen Ha Vy

Phan Thanh Nam

Has successfully designed & developed

Project: Sneakers Online

**Submitted by:**

Nguyen Son Ngoc

**Date of issue:** June 05th 2021

**Authorized Signature:**

CONTENT

[REVIEW 1 4](#_Toc74517359)

[1. ACKNOWLEDGE 5](#_Toc74517360)

[2. INTRODUCTION 5](#_Toc74517361)

[3. PROBLEM DEFINITION 6](#_Toc74517362)

[4. CUSTOMER'S REQUIREMENTS SPECIFICATIONS (CRS) 6](#_Toc74517363)

[CLIENT: Sneakers Online 6](#_Toc74517364)

[5. TASK SHEET REVIEW 1 10](#_Toc74517365)

[REVIEW 2 11](#_Toc74517366)

[6. USECASE 12](#_Toc74517367)

[7. DATA FLOW DIAGRAM (DFD) 38](#_Toc74517368)

[8. ENTITY PROPERTIES 43](#_Toc74517369)

[9. ENTITY RELATIONSHIP DIAGRAM (ERD) 46](#_Toc74517370)

[10. TABLE DESIGN 48](#_Toc74517371)

[11. TASK SHEET REVIEW 2 55](#_Toc74517372)

[REVIEW 3 56](#_Toc74517373)

[12. SITE MAP 57](#_Toc74517374)

[13. ORGANIC STORE DESIGN 59](#_Toc74517375)

[14. task sheet review 3 83](#_Toc74517376)

# REVIEW 1

## ACKNOWLEDGE

Our group would like to devote this chapter to express our sincere appreciation to all those who offered us valuable assistance during in carrying out this Project.

Firstly, we owe a debt of gratitude to our dedicated instructor, Mr. Hoang Duc Quang, whose suggestions and encouragement helped us to overcome not only technological challenges but also psychological tension. Without him, our work could not complete successfully.

Secondly, we highly appreciate the Project Team at the Head Office, which has provided us an excellent opportunity to apply what we learned into practice. We are certain that the experiences we gained from this project will facilitate our future career greatly.

Thirdly, we would also like to express my gratitude to all the staff members of my center for not only providing me with the opportunity to work with them on this project, but also for their support and encouragement throughout the process.

Finally yet importantly, we wish to say a personal thank you to all our dear classmates and team members who are always willing to support us openly and enthusiastically.

## INTRODUCTION

Sneakers may just be one of the most important pieces of sports equipment you can buy for your young athlete.  So it's important to give thought to what goes on your youngster's feet.

Whether you like it or not, often a kid's image is tied up in the specific brand of sneaker they wear and how much the shoes cost. But according to the American Academy of Pediatric Sports Medicine, for good foot health, the condition of the shoe is often more important than price or brand status.

More over, we are in an era where information technology makes buying and selling easier than ever. People can order an expensive phone to a daily kitchenware and everything is completed electronically and interactively in real- time. Sneakers is also out of this cycle. Therefore, we are developing a website named **Sneakers Online** to bring good products to more customers.

**Sneakers Online** is for an organization who wants to develop an online store where their users can be a person who does not have much time to go to the sneakers store or person has shopping online habit or just simply they stay too far from standalone store. It will provide its user the following features.

For customer:

* Customer can register account.
* Customer can login, login with facebook, google.
* Customer can manage their account.
* If customer forgot password, they can reset password, (using their email).
* Customer can view any products follow category such as Addidas, Nike, Puma, Converse.
* Customer can search products by brand, function and price.
* Customer can see products’ introduction before buying.
* Customer can buy products after adding products to cart or adding products to wishlist.
* Customer can view cart, update cart.
* Customer can payment online or payment on delivery.
* Customer can check ordered information.
* Customer can chat with shop by chatbox.
* Customer can interact with website by giving question, feedback on contact us function.
* Customer can check out.

## PROBLEM DEFINITION

After analyzing the project specifications of Sneakers Online, to fulfil customer’s requirements properly, we came to conclusion that in order to achieve the best result, we should deal with following queries:

### From the view point of visitors

* A homepage will show the main products, suitable logo, display the logo on all the pages in the site.
* A menu that will include all the functions implemented in the website.
* Customer can buy products, add product to cart, add product to wishlist, view cart and check out after login their account.

### From the view point of the Administrator

* A login page will grant access to Sneakers Online’s admin.
* Admin can view, add, update, and delete products.
* Admin can view, add, update, delete, find, lock, and unlock customer account.
* Admin can manage news and orders.

## CUSTOMER'S REQUIREMENTS SPECIFICATIONS (CRS)

## CLIENT**:** Sneakers Online

### **BUSINESS/PROJECT OBJECTIVE:**

The project is to design an online website for selling sneakers products, for instant: Addidas, Nike, Puma, Converse.

#### **Input to the system:**

**Sneakers Online’s administrators need to provide:**

* Information required in order creating account to access to the admin system.
* Products are going to be shown on the website.

**Sneakers Online’s customers need to provide:**

* Personal information to buy products.

#### **Output from the system:**

**Sneakers Online’s administrators will be able to:**

* Login/Logout to the admin system.
* Manage (view, add, update, delete) products, price.
* Manage (view, add, update, delete, find, lock, unlock) customer account.
* Manage (view, add, update, delete) news.
* Manage (view, update) products, order.

**Sneakers Online’s guide will be able to:**

* Register customer account.
* Buy products after add information.
* View products that they have chosen.

**Sneakers Online’s customer will be able to:**

* Register account.
* Login/Logout to the account.
* Search products they want to buy by name, price.
* Add product to cart and checkout.
* View purchase history.

### SCOPE OF WORK

The application should perform the following functions:

#### Login**:**

* There are 1 types of account in the application (“admin”). Administrator will login as “admin”.
* For “admin”: can view the products id, name, add new, update, delete.
* For “admin”: can view the customer id, name, add new, update, delete, lock, unlock.
* For “admin”: can view and manage view.
* For “admin”: can view and manage order.

#### **Buying products:**

* When customer choose a product and the selected product will be added to the cart.
* Customer can buy products at shopping cart after pressing the “Checkout” button and fill in the information.

#### **Database:**

* The information about products, customers, news, orders will be stored in the database.
* The information from the contact us page.
* The information from the about us page.

### HARDWARE/ SOFTWARE REQUIREMENTS:

#### **Hardware**

##### **Web Server**

|  |  |
| --- | --- |
| **Processor** | Intel Core I5 or higher. |
| **Memory** | 4 GB RAM or greater. |
| **Modem/ADSL** | Internet access is required. |

##### **Client**

|  |  |
| --- | --- |
| **Processor** | Intel Core I3 or higher. |
| **Memory** | 2 GB RAM or greater. |
| **Monitor** | Super VGA (1024x768) or higher resolution. |
| **Modem/ADSL** | Internet access is required. |

#### **Software**

##### **Web Server**

|  |  |
| --- | --- |
| **Operation System** | Window 7 or later. |
| **Browser** | Google Chrome version 35. |
| **Database** | MySQL. |
| **Software** | Internet Information Service |
| PHP - Laravel framework |

##### **Client**

|  |  |
| --- | --- |
| **Operation System** | Window 7 or higher. |
| **Browser** | Google Chrome version 35. |

#### **Development software**

* Microsoft Windows 10.
* MySQL.
* Visual Studio Code.
* Google Chrome.
* XAMPP.

#### **Technology**

* PHP.
* JQuery.
* Bootstrap.
* HTML/CSS

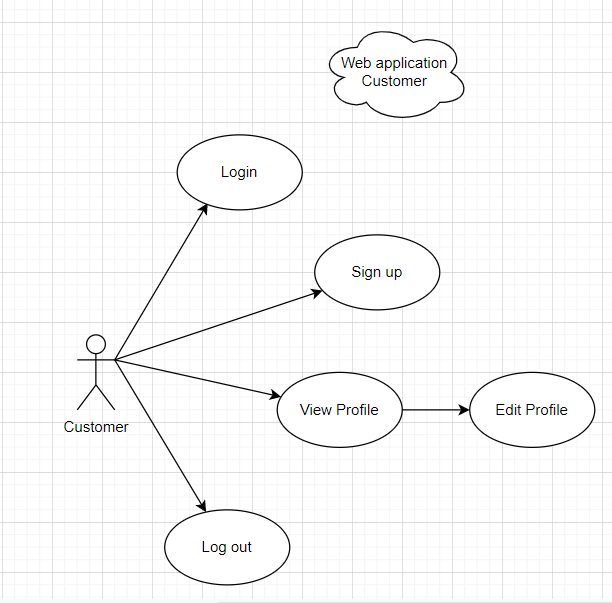
## TASK SHEET REVIEW 1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Project Ref. No.: Sneakers Online | | Project Name | Activity Plan Prepared By | Date of Preparation of Activity Plan: | | | | |
| Sr./No. | Task | Actual Start Date | Actual Days | Team member name | Status |
| 1 | Acknowledgment | Sneakers Online | All Team | 02-Jun-21 | 1 | Ngoc | Completed |
| 2 | Introduction | 02-Jun-21 | 1 | Ngoc | Completed |
| 3 | Problem Definition | 02-Jun-21 | 1 | Ngoc | Completed |
| 4 | Customer Requirement Specification (CRS) | 02-Jun-21 | 3 | Ngoc | Completed |
| 5 | Scope of Work | 02-Jun-21 | 2 | Vy | Completed |
| 6 | Hardware/Software Requirement | 02-Jun-21 | 2 | Tai | Completed |
| 7 | Task Sheet | 02-Jun-21 | 1 | Nam | Completed |

|  |  |
| --- | --- |
| Date: 05 Jun 2021 | |
| Signature of Instructor:  **Mr. Hoang Duc Quang** | Signature of Team Leader:  **Nguyen Ha Vy** |

# REVIEW 2

## USECASE



### CUSTOMER

#### LOGIN AS CUSTOMER

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Login | |
| **Actors** | Customer | |
| **Description** | Customer who has registered an account can login | |
| **Requirements** | Customer provides username and password | |
| **Pre-conditions** | Customer must register an account | |
| **Post-conditions** | Success: Customer is logged in to website | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Customer clicks ‘Login’ button on Home Page.  3. Actor inputs Email and Password, then click the ‘Login’ button. | 2. System redirects to Login Page with the following controls:  - “Email” text field  - “Password” text field  - “Login” button  4. System checks the information.  5. System redirects to Customer’s dashboard. |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor inputs invalid email and password. | System redirects to Login page with the following controls:  - “Email” text field  - “Password” text field  - “Login” button  System shows message: “Invalid email and password”. |

#### REGISTER AS CUSTOMER:

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Register as Customer | |
| **Actors** | Customer | |
| **Description** | Customer who has registered an account can login | |
| **Requirements** | Customer registers an account on the website and becomes a member | |
| **Pre-conditions** | Customer provides their register information | |
| **Post-conditions** | Success: Customer creates a User account in this website | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Customer clicks “Register” button on Home Page.  3. Customer inputs register’s information and click “Register” button. | 2. System redirects to Register page and displays Registration form with the following controls:  - “Username” text field  - “Password” text field  - “Confirm Password” text field  - “Email” text field  - “Register” button  4. System validates the information.  5. System inserts the account into database  6. System show success message. |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| - Actor left “Username” text field blank.  - Username has already existed.  - Actor left “Password” text field blank.  - Actor left “Confirm Password” text field blank.  - Password and confirm password don’t match.  - Actor left “Email” text field blank.  - Email has already existed.  - Actor left “Email” text field blank or input an invalid email address. | System displays message to customer:  - “Username cannot be blank”.  - “This Username has already existed”.  - “Password cannot be blank”.  - “Confirm Password cannot be blank”.  - “Confirm Password does not match password.”.  - “Email cannot be blank”.  - “This Email has already existed”.  - “Email is invalid”. |

#### VIEW PROFILE

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | View profile | |
| **Actors** | Customer | |
| **Description** | Customer who want to view their profile | |
| **Requirements** | Customer provides their personal information | |
| **Pre-conditions** | Customer is logged in | |
| **Post-conditions** | Success: Actor can view their profile | |
| Fail: Actor cannot view their profile | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks “My Account” button on Home Page. | 2. System redirects to Account page and displays Account form with the following controls:  -“Username” text field  - “Fullname” text field  - “Address” text field  - “Email” text field |

#### EDIT PROFILE:

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Edit profile | |
| **Actors** | Customer | |
| **Description** | Customer who want to update their profile | |
| **Requirements** | Customer provides their personal information | |
| **Pre-conditions** | Customer is logged in | |
| **Post-conditions** | Success: Actor can edit their profile | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks “Account” button on Home Page.  3. Actor inputs personal information and clicks “Submit”. | 2. System redirects to Account page and displays Account form with the following controls:  -“Username” text field  - “Fullname” text field  - “Address” text field  - “Email” text field  - “Password” text field  - “Confirm password” text field  - “Save” button  4. System validates password then sends to database.  5. System shows message: “Your account has been update successfully”. |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor loggin an account which does not exist in database. | System shows message: “Invalid account”. |

#### LOGOUT

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Logout | |
| **Actors** | Customer | |
| **Description** | Customer clicks on “Logout” button and their session is terminated. | |
| **Requirements** | N/A | |
| **Pre-conditions** | 1. Customer is logged in  2. Customer no longer wants to be logged in | |
| **Post-conditions** | Success: Customer can logout | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor clicks “Log out” button on Home Page. | 2. The system logs the actor out and invalidates the session.  3. The system redirects to the default logout page. |
| **Exceptions** | Actor’s actions: | System’s responses: |
| 1. Actor cannot logout | System shows message: “Not logged in”. |

### CART



#### SEARCH PRODUCTS

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Search product | |
| **Actors** | Registered Customer/ New Customer | |
| **Description** | Customer can search product by category, price. | |
| **Requirements** | Customer provides search phrase | |
| **Pre-conditions** | N/A | |
| **Post-conditions** | Success: Show product that matches search phrase | |
| Fail: No product is shown | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor inputs search phrase.  3. Actor clicks product’s name in list or view every product that matches search phrase. | 2. System shows list of products that match the search phrase. |

#### VIEW PRODUCT DETAIL

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | View product | |
| **Actors** | Registered Customer/ New Customer | |
| **Description** | Customer can view product detail | |
| **Requirements** | Customer select product to show product’s detail | |
| **Pre-conditions** | N/A | |
| **Post-conditions** | Success: Display product detail matching the selected product | |
| Fail: No product is shown | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor select product | 2. System shows product’s detail matching selected product |

#### CREATE CART

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Create cart | |
| **Actors** | Registered customer | |
| **Description** | Customer can add product to their cart | |
| **Requirements** | Customer must choose at least one product | |
| **Pre-conditions** | Customer must log in before adding product | |
| **Post-conditions** | Success: Actor add product to cart successfully | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses a product and click “Add to cart”. | 2. System adds selected product to cart by id of this product. |
| **Exceptions** | **Actor’s action** | **System’s response** |
| 1. Actor inputs more products than there are in stock. | 2. System shows messages:  - “Product out of stock”. |

#### VIEW CART

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | View cart | |
| **Actors** | Registered customer | |
| **Description** | Customer can view their cart | |
| **Requirements** | N/A | |
| **Pre-conditions** | Customer must log in before view cart | |
| **Post-conditions** | Success: Customer can view their cart | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor click on “Add to Cart” button or “Cart” icon. | 2. System redirects to current “Cart” page.  3. System shows cart and added products. |
| **Alternative flow** | 1. Actor clicks “x”– Close button- on table  3. Actor change size of a product.  5. Actor clicks “x” – close button – in the right side of each product. | 2. System redirects to current page.  4. System update size of this product into cart.  6. Actor change quantity of a product.  7. System removes this product from cart. |

#### CHECKOUT

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Check out and Place order | |
| **Actors** | Registered customer | |
| **Description** | Customer can order products | |
| **Requirements** | Customer must has at least 1 product in cart | |
| **Pre-conditions** | Customer must log in before adding product | |
| **Post-conditions** | Success: Actor order successfully | |
| Fail: Refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor click “Check out” or “My Cart” at dropdown menu list or “Add to Cart” button or “Cart” icon.  3. Actor clicks “Proceed to Check out” button then clicks “Confirm Order”.  5. Actor inputs order information then click “Delivery Details”.  7.Actor click “Payment Method”.  9. Actor inputs Card owner’s name, card number and security code if actor choose payment by credit card. | 2. System shows cart and added products.  4. System shows all products in cart with calculated total price.  6. System shows following controls:  - “Customer name” text field.  - “Phone” text field.  - “Address” text field.  8. Systems shows Payment tab type according to type of payment actors choose.  10. System validates information. |
| **Alternative flow** | 1. Actor clicks “X” button on each product.  3. Actors clicks “Home” button. | 2. System removes product from cart.  4. System redirect to Home pages. |
| **Exceptions** | **Actor’s action** | **System’s response** |
| 1. Actor inputs more products than there are in stock.  2. Actor left “Billing name” text field blank.  3. Actor left “Address” text field blank.  4. Actor left “Delivery date” blank.  5. Actor left “Delivery time frame” blank.  6. Actor left “Payment” dropdown blank.  7. Actor inputs invalid card information. | 8. System shows messages:  - “Product out of stock”.  - “Order name must not be blank”.  “Address must not be blank”.  “Must choose a delivery date”.  “Must choose a delivery time frame”.  “Must choose a payment method”.  “Invalid card”. |

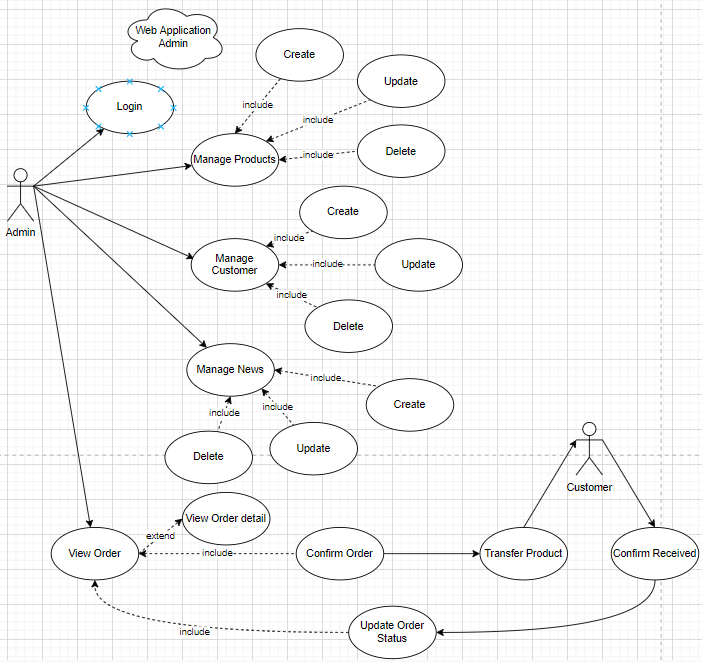
#### VIEW ORDER

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | View order | |
| **Actors** | Registered customer | |
| **Description** | Customer can view their order history | |
| **Requirements** | N/A | |
| **Pre-conditions** | Customer must log in to website | |
| **Post-conditions** | Success: Customer can view their order. | |
| Fail: Customer cannot see their order. | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses “My Account” icon on Menu  3. Actor chooses “Order History and Details” tag. | 2. System redirects to Account page.  4. System displays a list of existing orders and their status.  5. System displays all order’s detail and its status. |

#### CANCEL ORDER

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Cancel order | |
| **Actors** | Registered customer | |
| **Description** | Customer can cancel their order | |
| **Requirements** | Customer provides order’s id to cancel | |
| **Pre-conditions** | Customer must log in to website and only can cancel when this order has not been confirmed by admin | |
| **Post-conditions** | Success: Cancel successfully. | |
| Fail: Order cannot cancel. | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses “Account” icon on Menu  3. Actor chooses “Order History and Details” tag.  5. Actor selects the order that they want to cancel in “Details” and clicks “Cancel” Button. | 2. System redirects to Account page.  4. System redirects to “Purchase history” page and displays a list of existing orders.  6. System validates inputs by id of order.  7. System change order status in database.  8. System displays message: successful for error. |

### ADMIN



#### LOGIN

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Login | |
| **Actor** | Admin | |
| **Description** | When admin wants to manage products, news, views orders and orders detail | |
| **Requirements** | Admin provides username and password | |
| **Pre-conditions** | Admin must be exist on database | |
| **Post-conditions** | Success: Admin is logged in to website admin | |
| Fail: refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor go to ‘Login Admin’ page .  3. Actor inputs Username and Password, then click the ‘Login’ button. | 2. System shows with the following controls:  - 'Username' text field  - 'Password' text field  - 'Login' button  4. System validates the information.  5. System redirects to Admin website |
| **Exceptions** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor inputs invalid email and password | 2.System shows message: “Invalid username or password”.  3.System redirects to Login Admin page with the following controls:  - ‘Username’ text field  - ‘Password’ text field  - ‘Login’ button. |

#### Manage PRODUCTS

##### Create new Product

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Create new product | |
| **Actor** | Admin | |
| **Description** | When admin wants to create a new product | |
| **Requirements** | Actor provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: new product is added into the database | |
| Fail: refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses Product Tab on menu.  3. Actor clicks ‘Add New Product’ button.  5. Actor inputs information to create new product | 2. System redirects to ‘ All Products’ page and display a list of existing product.  4. System displays a form with the following fields:  - Product Name  - Product Price  - Brand Name  - Special  - Image  - Quantity: + Size Product  - Content  6. System validates inputs.  7. System adds product into database.  8. System displays confirmation message or error message. |

##### Update Product

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Update product’s detail | |
| **Actor** | Admin | |
| **Description** | When admin wants to update product | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: product is updated in the database | |
| Fail: refill information | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses Product Tab on menu.  3. Actor clicks on ‘Edit Product’ on a product.  5. Actor clicks on ‘Add Stock’ on a product.  7. Actor chooses the criteria they want to update and input information. | 2. System redirects to ‘All Products’ page and display a list of existing product.  4. System displays a form with the following fields:  - Product Name  - Product Price  - Brand Name  - Special  - Image  - Quantity: + Size Product  - Content  6. System displays a form with the following fields:  - Size product  - Already Available  - Stock You Want To Add  8. System validates inputs.  9. System update product into database.  10. System displays confirmation message or error message.  11. System redirects to product’s detail page. |

##### Delete Product

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Delete Product | |
| **Actor** | Admin | |
| **Description** | When admin wants to delete product from the database | |
| **Requirements** | Admin provides product’s information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: product is delete in the database | |
| Fail: product is not delete from the database | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses Product Tab on menu.  3. Actor clicks ‘Delete’ button. | 2. System redirects to ‘All Products’ page and display a list of existing product.  4. System delete product in the database  5. System displays confirmation or error. |

### NEWS

#### Create new News

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Create News | |
| **Actor** | Admin | |
| **Description** | When admin wants to create a News | |
| **Requirements** | Admin provides New content | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: News content is added into the website | |
| Fail: admin can’t create News | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses News Tab on menu.  3. Actor clicks ‘Add New Content’ button.  5. Actor inputs information to create news | 2. System redirects to ‘News’ page and display a list of News.  4. System displays a form with the following fields:  - Title  - Content  - Image  6. System validates inputs.  7. System adds News into website. |

#### Update News

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Update News on the website | |
| **Actor** | Admin | |
| **Description** | When admin wants to update news information | |
| **Requirements** | Admin provides News Content | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: News is updated in the website | |
| Fail: admin can’t update News | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses News Tab on menu.  3. Actor clicks ‘Edit’ button.  5. Actor chooses the criteria they want to update and input information. | 2. System redirects to ‘News’ page and display a list of News.  4. System displays a form with the following fields.  - Title  - Content  - Image  - Old Image  6. System validates inputs.  7. System update News Content to website.  8. System redirects to Manage Blog. |

#### Delete News

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Delete News | |
| **Actor** | Admin | |
| **Description** | When admin wants to delete News from the website | |
| **Requirements** | Admin provides News Content | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: product is delete in the website | |
| Fail: product is not delete from the website | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses “Manage Post” on menu.  3. Actor clicks ‘Delete’ button. | 2. System redirects to ‘News List’ page and display a list of News.  4. System delete News in the website  5. System displays confirmation or error. |

### ORDER

#### View Order

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | View Order | |
| **Actor** | Admin | |
| **Description** | When admin wants to know the order of the product | |
| **Requirements** | Admin has at least 1 order on website. | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: product is shown in the website | |
| Fail: Page View Order cannot see from the website | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses View Order on menu.  3. Actor clicks ‘View’ button. | 2. System redirects to ‘Order List’ page and display a list of Order.  4. System show Order in the website.  5. System displays confirmation or error. |

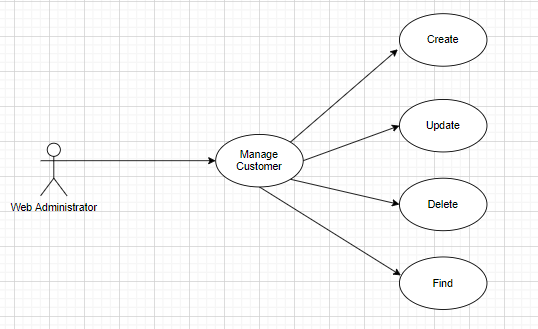
#### View Order Detail

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | View Order Detail | |
| **Actor** | Admin | |
| **Description** | When admin wants to see View Order Detail from the website | |
| **Requirements** | Admin has at least 1 order on website. | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: Admin can view Order Detail on the website | |
| Fail: Admin cant’ view the Order Detail on the website | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses View Order on menu.  3. Actor clicks ‘Detail’ button. | 2. System redirects to ‘Order List’ page and display a list of Order.  4. System show Order Details in the website  5. System displays confirmation or error. |

#### Update Order Status

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Update Order Status | |
| **Actor** | Admin | |
| **Description** | When admin wants to update Order Status from the website | |
| **Requirements** | Admin has at least 1 order on website. | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: Admin can update Order Status on the website | |
| Fail: Admin can’t update Order Status on the website | |
| **Basic flow** | **Actor’s actions:** | **System’s responses:** |
| 1. Actor chooses View Order Tab on menu.  3. Actor clicks ‘Delete’ button. | 2. System redirects to ‘Order List’ page and display a list of Order.  4. System update Order Status in the website  5. System displays confirmation or error. |

### WEBSITE ADMINISTRATOR



#### CREATE CUSTOMER

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use case name** | Create Customer | |
| **Actor** | Admin | |
| **Description** | When admin wants to create customer | |
| **Requirements** | Actor provides customer information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-conditions** | Success: new Customer is added into the database | |
| Fail: refill information | |
| **Basic flow** | **Actor’s action** | **System’s responses:** |
| 1. Actor chooses Customer Tab on menu.  3. Actor clicks ‘Add New Customer’ button.  5. Actor inputs information to create Customer. | 2. System redirects to ‘Customer’ page and display a list of existing Customers.  4. System displays a form with the following fields:  - username  - password  - email  6. System validates inputs.  7. System adds new customer into database.  8. System displays confirmation message or error message. |

#### UPDATE CUSTOMER

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Update Customer | |
| **Actors** | Admin | |
| **Description** | When admin wants to Update customer | |
| **Requirements** | Actor provides customer information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-condition** | Success: Customer is update into the database | |
| Fail: refill information | |
| **Basic flow** | **Actor’s action** | **System’s responses:** |
| 1. Actor chooses Customer Tab on menu.  3. Actor clicks on ‘View Details’ on a Customer  5. Actor chooses the criteria they want to update and input information. | 2. System redirects to ‘Customer’ page and display a list of existing Customers.  4. System displays a form with the following fields:  - username  - password  - email  6. System validates inputs.  7. System update customer into database.  8. System displays confirmation message or error message.  9. System redirects to customer’s detail page. |

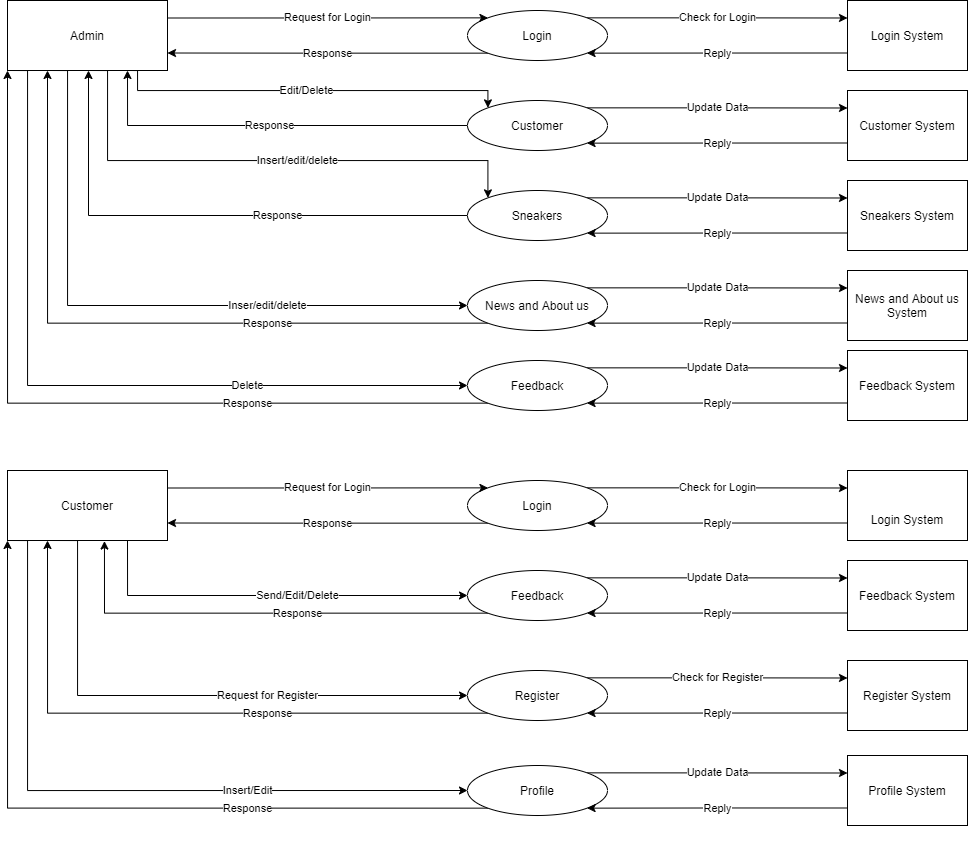
#### DELETE CUSTOMER

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Delete Customer | |
| **Actors** | Admin | |
| **Description** | When admin wants to delete customer | |
| **Requirements** | Actor provides customer information | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-condition** | Success: Customer is delete from the database | |
| Fail: Customer is not deleted from the database | |
| **Basic flow** | **Actor’s action** | **System’s responses:** |
| 1. Actor chooses Customer Tab on menu.  3. Actor clicks ‘Delete’ button. | 2. System redirects to ‘Customer’ page and display a list of existing Customers.  4. System delete customer in the database.  5. System displays confirmation message or error message. |

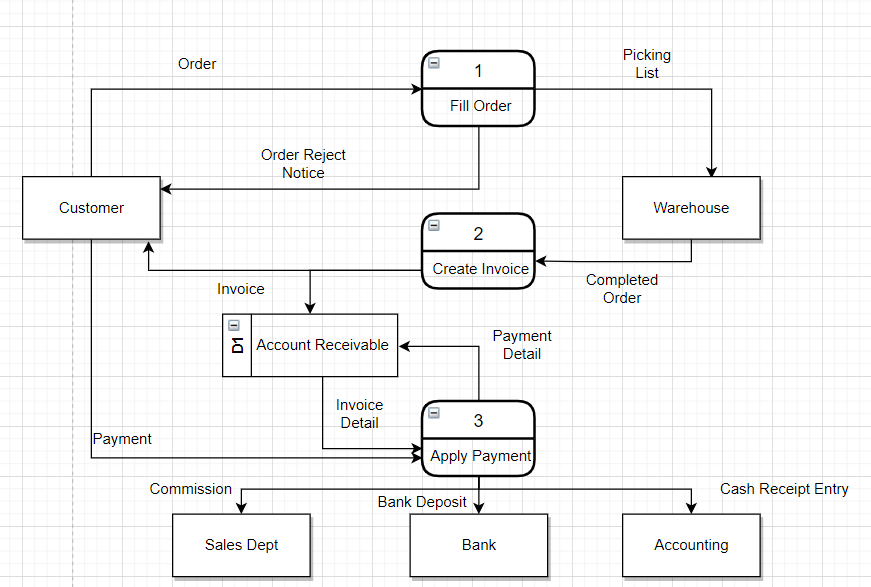
#### FIND CUSTOMER

|  |  |  |
| --- | --- | --- |
| **Author** |  | |
| **Use Case Name** | Find Customer | |
| **Actors** | Admin | |
| **Description** | When admin wants to find customer by Username | |
| **Requirements** | Actor provides Customer’s Username | |
| **Pre-conditions** | Actor is logged in admin | |
| **Post-condition** | Success: Show customer that matches Username | |
| Fail: No customer is shown | |
| **Basic flow** | **Actor’s action** | **System’s responses:** |
| 1 Actor inputs Username .  3. Actor clicks customer’s title in list or view, which matches the Username . | 2. System show list of customer that match the Username . |

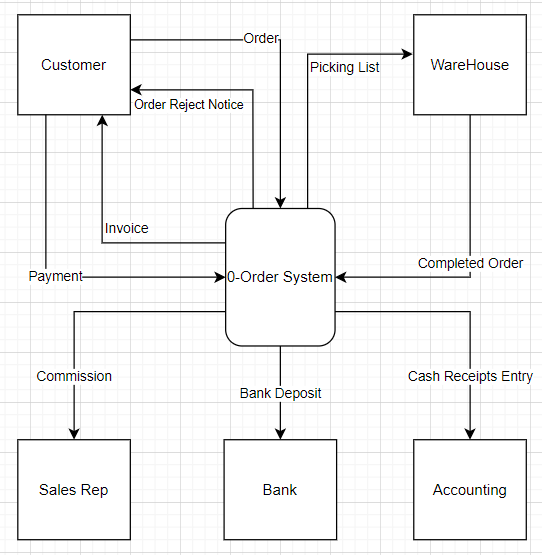
## DATA FLOW DIAGRAM (DFD)

****

### LEVEL 0

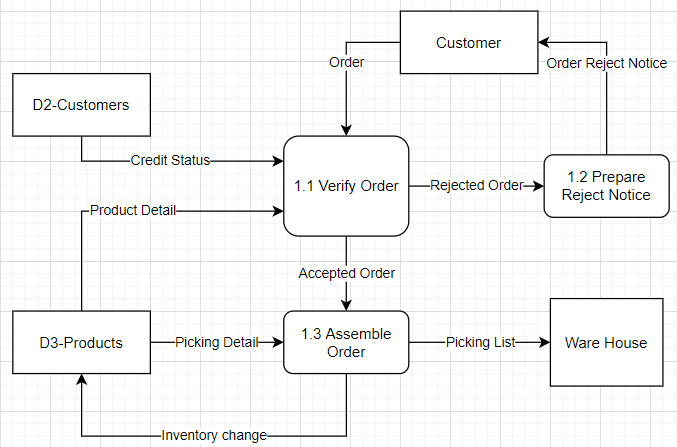


#### Order Sytem

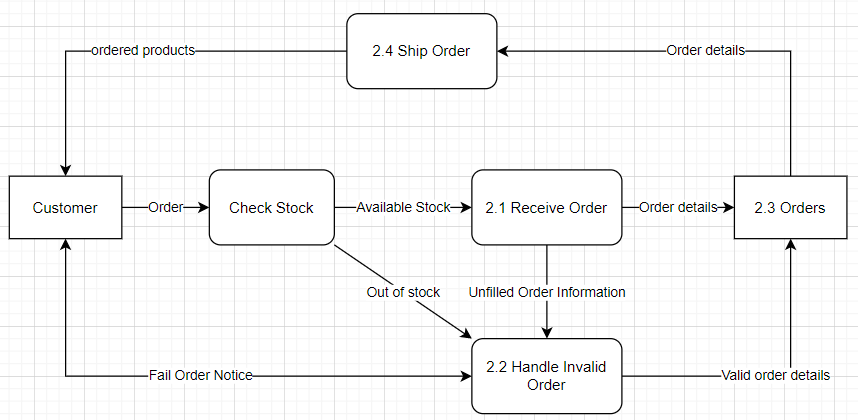


### LEVEL 1

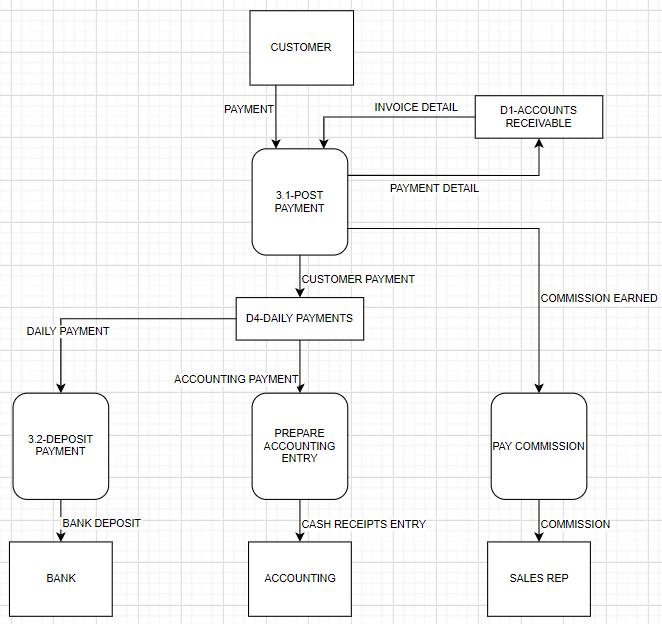
#### 1.FILL ORDER



#### 2.CREATE INVOICE

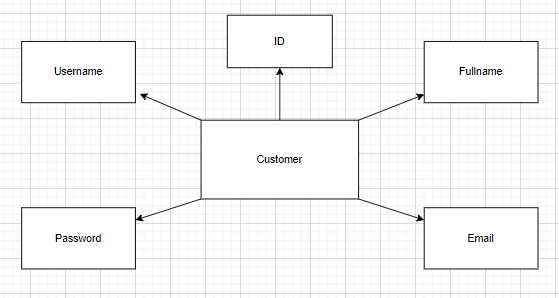


#### 3.APPLY PAYMENT

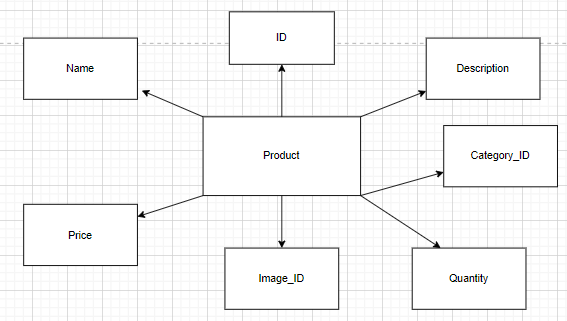


## ENTITY PROPERTIES

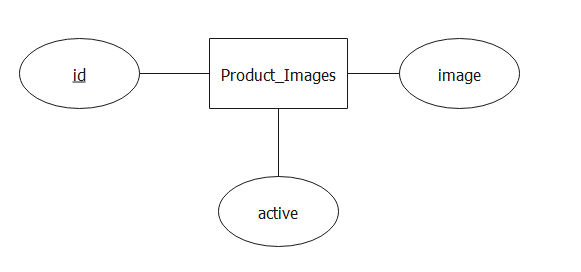
### Customer



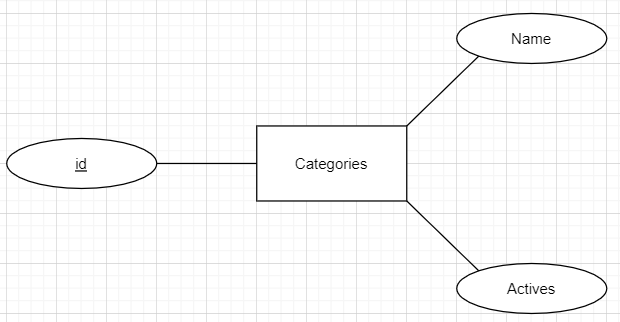
### Product



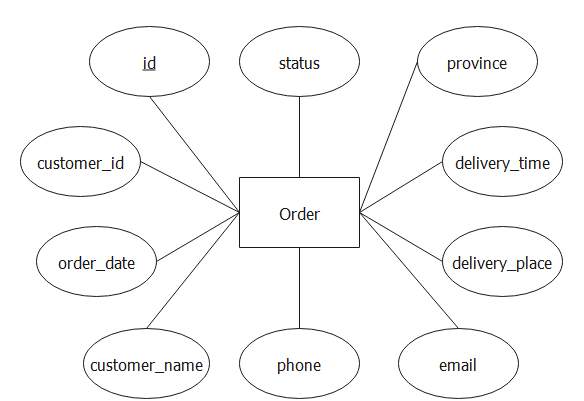
### Product\_Image

****

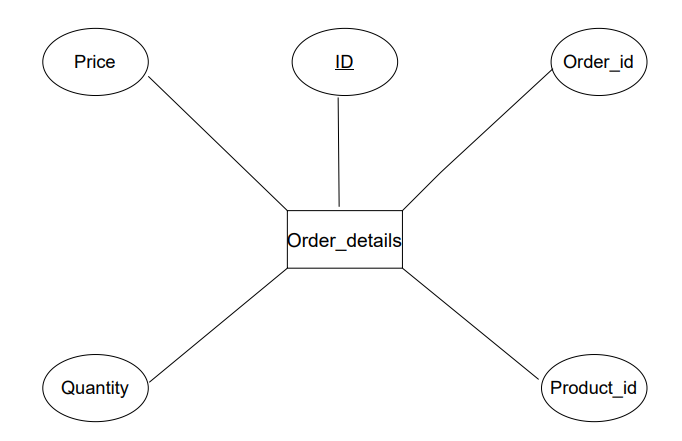
### Category



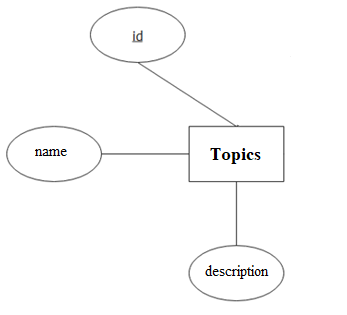
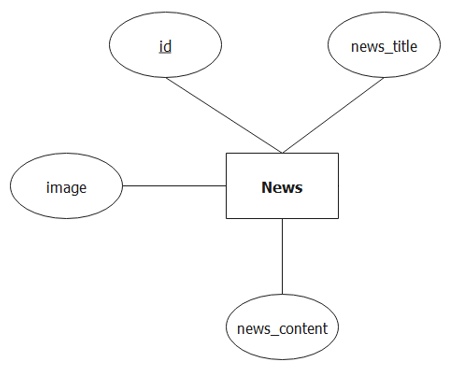
### Order

****

### Order\_Detail



### News and Topics

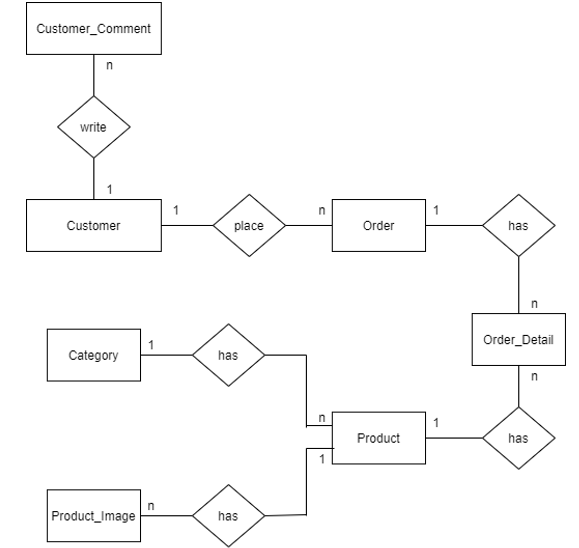


### Admin

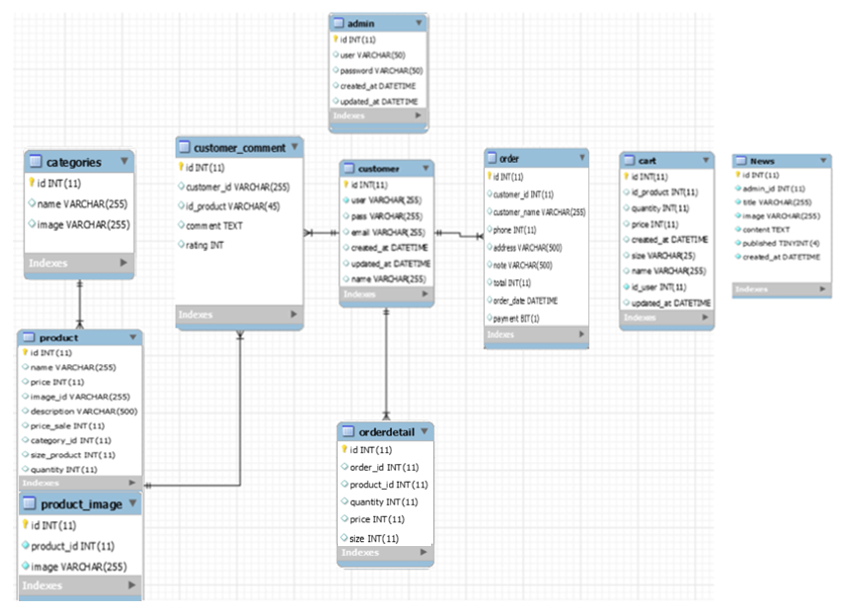


## ENTITY RELATIONSHIP DIAGRAM (ERD)

### LOGICAL ERD



### PHYSICAL ERD



## TABLE DESIGN

### Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table: Admin | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | Id | INT(11)  Auto-increment | NOT NULL  PRIMARY KEY | Store the ID of the admin |
| 2 | username | VARCHAR(50) | NOT NULL | Store the user name of the admin |
| 3 | password | VARCHAR(50) | NOT NULL | Store the password of the admin |
| 4 | Created\_at | DATE TIME | NOT NULL | Store the date time at the time of create admin |
| 5 | Updated\_at | DATE TIME | NOT NULL | Store the date time at the time of update admin |

### Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table: Customer | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | Id | INT(11)  Auto-increment | NOT NULL  PRIMARY KEY | Store the ID of the customer |
| 2 | name | VARCHAR(255) | NOT NULL | Store the full name of the customer |
| 3 | username | VARCHAR(255) | NOT NULL | Store the user name of the customer |
| 4 | password | VARCHAR(255) | NOT NULL | Store the password of the customer |
| 5 | email | VARCHAR(255) | NOT NULL | Store the Email of the customer |
| 6 | Created\_at | DATE TIME | NOT NULL | Store the date time at the time of create admin |
| 7 | Updated\_at | DATE TIME | NOT NULL | Store the date time at the time of update admin |

### Customer\_Comment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table: Customer\_comments | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | id | INT(11)  Auto-increment | NOT NULL  PRIMARY KEY | Store the ID of customer comment |
| 2 | customer\_id | INT(11) | NOT NULL  FOREIGN KEY | Store the ID of customer |
| 3 | product\_id | VARCHAR(100) | NOT NULL | Store the ID of product |
| 4 | comment | DATE | NOT NULL | Store comment of customer |
| 5 | rating | INT | NOT NULL | Store rating of customer |

### Product

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: products** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | id | INT(11)  Auto-increment | NOT NULL  PRIMARY KEY | Store the ID of the product. |
| 2 | category\_id | INT(11) | NOT NULL  FOREIGN KEY | Store the ID of the category |
| 3 | image\_id | INT(11) | NOT NULL  FOREIGN KEY | Store the ID of the image. |
| 4 | name | VARCHAR(255) | NOT NULL | Stores the name of product |
| 5 | desciption | TEXT | NOT NULL | Stores the desciption of product |
| 6 | price | INT(11) | NOT NULL | Stores the price of product. |
| 7 | price\_sale | INT(11) | NOT NULL | Stores the price promotion of product. |
| 8 | quantity | INT(11) | NOT NULL | Stores the quantity of product. |

### Product\_Image

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: Products\_Images** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | id | INT(11)  Auto-increment | NOT NULL  PRIMARY KEY | Store the ID of the product\_image |
| 2 | product\_id | INT(11) | NOT NULL  FOREIGN KEY | Store the ID of the product |
| 3 | image | VARCHAR(255) | NOT NULL | Store the Image of the product |

### Category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: Categories** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | id | INT(11)  Auto-increment | NOT NULL  PRIMARY KEY | Store the ID of the product\_image |
| 2 | name | VARCHAR(255) | NOT NULL | Store the name of the sale |
| 3 | image | VARCHAR(255) | NOT NULL | Store the Image of the category |

### Order

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table: Orders | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | id | INT(11)  Auto-increment | NOT NULL  PRIMARY KEY | Store the ID of the order. |
| 2 | customer\_id | INT(11) | NOT NULL  FOREIGN KEY | Store the ID of customer. |
| 3 | customer\_name | VARCHAR(255) | NOT NULL | Store customer name. |
| 4 | order\_date | DATE TIME | NOT NULL | Store order date. |
| 5 | phone | INT(11) | NOT NULL | Store phone number of receiver. |
| 6 | address | VARCHAR(100) | NOT NULL | Store address of receiver |
| 7 | note | VARCHAR(500) | NOT NULL | Store note of receiver. |
| 8 | total | INT(11) | NOT NULL | Stores the total price of order. |
| 9 | payment | BIT(1) | NOT NULL | Store the payment of receiver. |

### Order\_Detail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: Order\_details** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | id | INT(11)  Auto-increment | NOT NULL  PRIMARY KEY | Store the ID of the order details |
| 2 | order\_id | INT(11) | NOT NULL  FOREIGN KEY | Store the ID of the order |
| 3 | product\_id | INT(11) | NOT NULL  FOREIGN KEY | Stores the ID of the product |
| 4 | quantity | INT(11) | NOT NULL | Stores the quantity of product |
| 5 | price | INT(11) | NOT NULL | Stores the price of product |
| 6 | size | INT(11) | NOT NULL | Stores the size of product |

### News

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: news** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | id | INT(11)  Auto-increment | NOT NULL PRIMARY KEY | Stores the ID of News |
| 2 | admin\_id | INT(11) | NOT NULL  FOREIGN KEY | Stores the ID of admin posted News. |
| 3 | news\_title | VARCHAR(255) | NOT NULL | Stores the Title of News |
| 4 | news\_content | VARCHAR(255) | NOT NULL | Stores the Content of News |
| 5 | image | VARCHAR(255) | NOT NULL | Stores the Image of News |

### Cart

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table: Cart** | | | | |
| No. | Column Name | Data Type | Constraint | Description |
| 1 | id | INT(11)  Auto-increment | NOT NULL  PRIMARY KEY | Store the ID of the Cart |
| 2 | product\_id | INT(11) | NOT NULL  FOREIGN KEY | Stores the ID of the product |
| 3 | quantity | INT(11) | NOT NULL | Stores the quantity of product |
| 4 | price | INT(11) | NOT NULL | Stores the price of product |
| 5 | size | INT(11) | NOT NULL | Stores the size of product |
| 6 | Created\_at | DATE TIME | NOT NULL | Store the date time at the time of create cart |
| 7 | Updated\_at | DATE TIME | NOT NULL | Store the date time at the time of update cart |

## TASK SHEET REVIEW 2

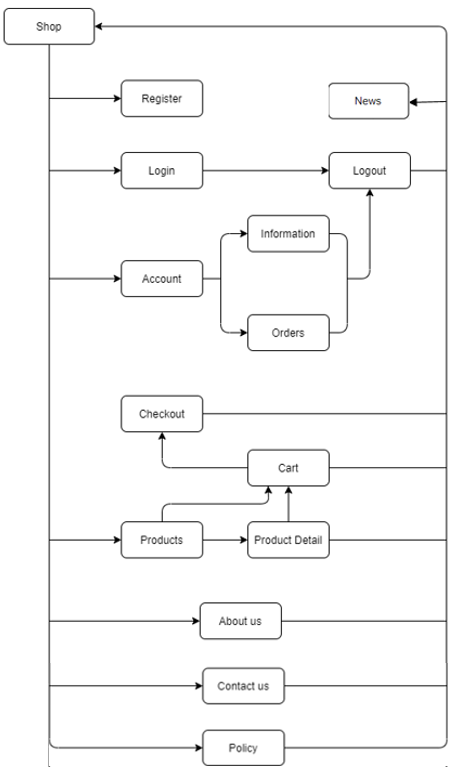
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Project Ref. No.: Sneakers Online | | Project Name | Activity Plan Prepared By | Date of Preparation of Activity Plan: | | | |
| Sr./No. | Task | Actual Start Date | Actual Days | Team member name | Status |
| 1 | Use Case | Sneakers Online | Chi | 7-Jun-21 | 3 | Ngọc | Completed |
| 2 | Data Flow Diagram (DFD) | 9-Jun-21 | 2 | Ngọc | Completed |
| 3 | Entity Relationship Diagram (ERD) | 7-Jun-21 | 3 | Ngọc  Vy | Completed |
| 4 | Entity Properties | 8-Jun-21 | 2 | Tài  Nam | Completed |
| 5 | Table Design | 9-Jun-21 | 1 | Vy | Completed |

|  |  |
| --- | --- |
| Date: 10 Jun 2020 | |
| Signature of Instructor:  Mr. Hoang Duc Quang | Signature of Team Leader:  Nguyen Ha Vy |

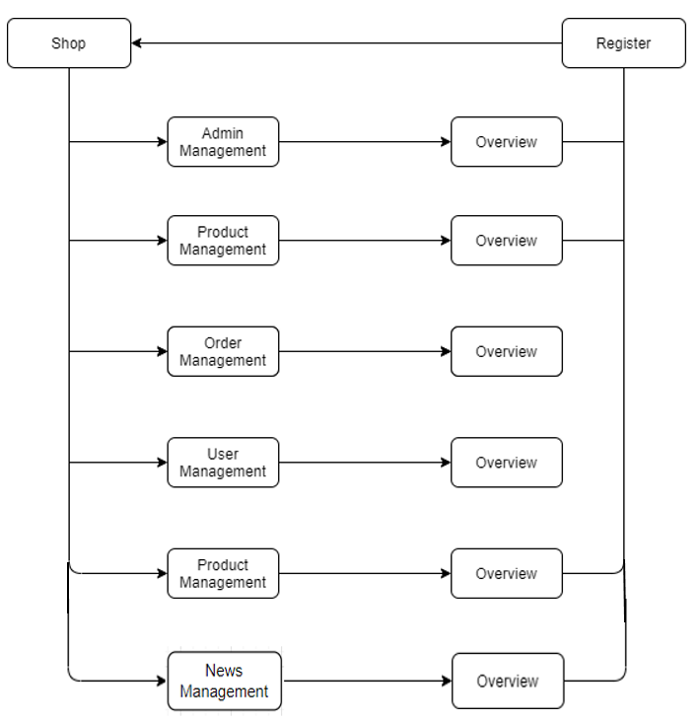
# REVIEW 3

## SITE MAP

### Customer’s Site Map

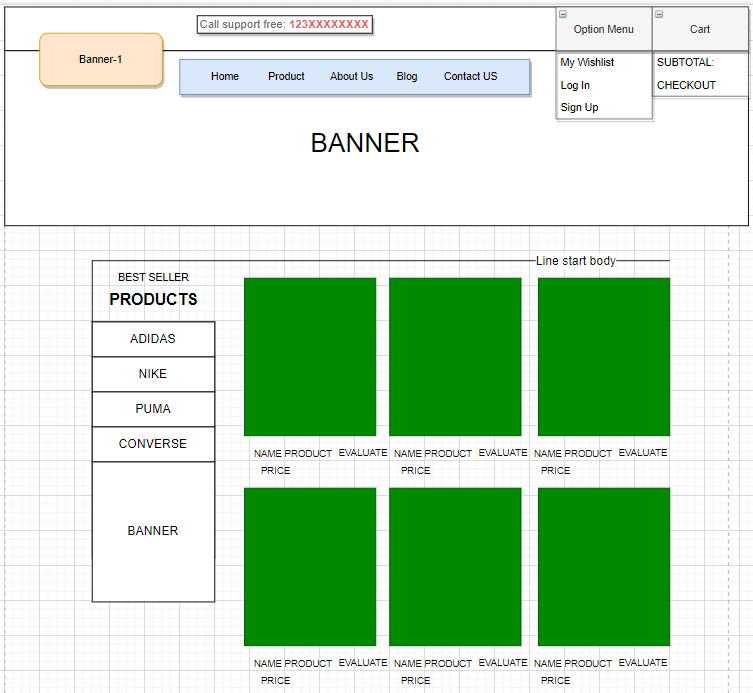


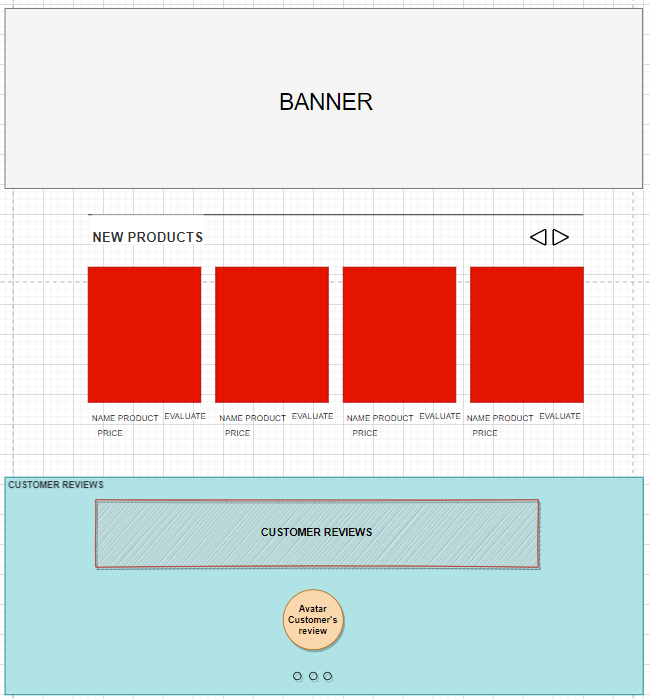
### Admin’s Site Map

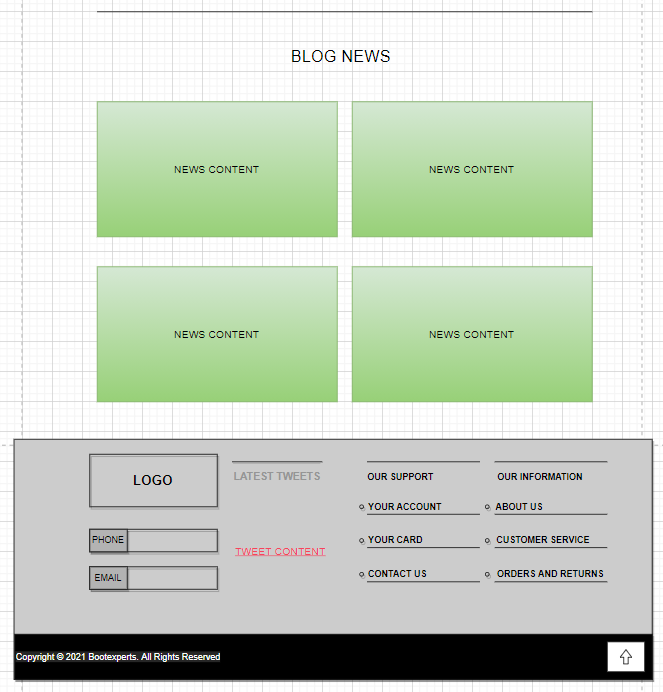


## SNEAKERS ONLINE DESIGN

### HomePage







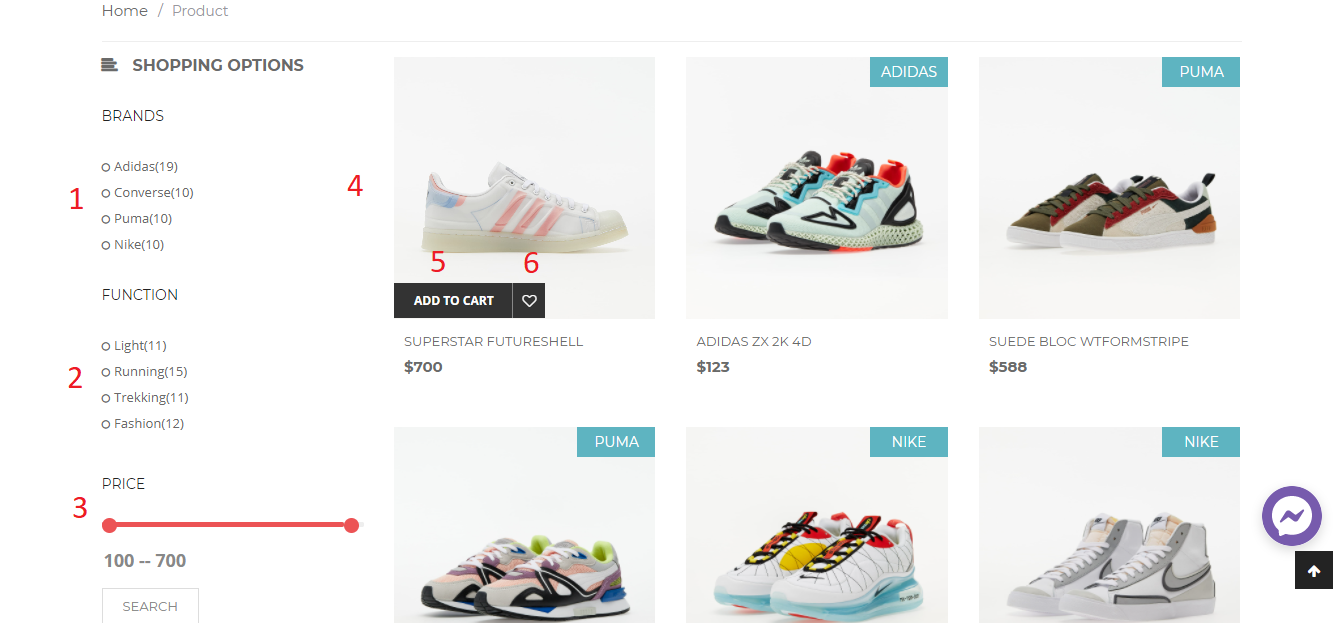
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Logo | Link |  | Click | Go to Home page. |
| 2 | Home | Link |  | Click | Go to Home page. |
| 3 | Products | Link |  | Click | Go to Products page. |
| 4 | About Us | Link |  | Click | Go to About Us page. |
| 5 | Blog | Link |  | Click | Go to Blog News page. |
| 6 | Contact us | Link |  | Click | Go to Contact us page. |
| 7 | My Wishlist | Link |  | Click | Go to wishlist page. |
| 8 | Login | Link |  | Click | Log in to existing account or register new account. |
| 9 | Sign Up | Link |  | Click | Register new account. |
| 10 | Cart | Button |  | Click | Go to Cart Page. |
| 11 | SubTotal | Text |  |  | Show total price of products in cart. |
| 12 | Check out | Button |  | Click | Go to Check out Page. |
| 13 | Side bar  (category:   * Addidas. * Nike. * Puma. * Converse. | Button |  | Click | Show products follow category. |
| 14 | Best Seller Products | Text |  |  | Show best seller product and is image . |
| 15 | New products | text |  |  | Show new product and is image. |
| 16 | Customer Review | Text |  |  | Show avatar and customer’s comment. |
| 17 | Blog News | Link |  | Click | Go to Blog News Page. |
| 18 | Footer (information) | Link |  | Click | Go to information for Sneakers Online. |
| 19 | Footer (product information) | Link |  | Click | Go to Question and Answer page, Privacy page, Term of use page. |
| 20 | Footer(Your Account) | Link |  | Click | Go to Account page or Login page. |
| 21 | Footer(Contact us) | Link |  | Click | Go to Contact Us page. |
| 22 | Footer(About us) | Link |  | Click | Go to About Us page. |

New product



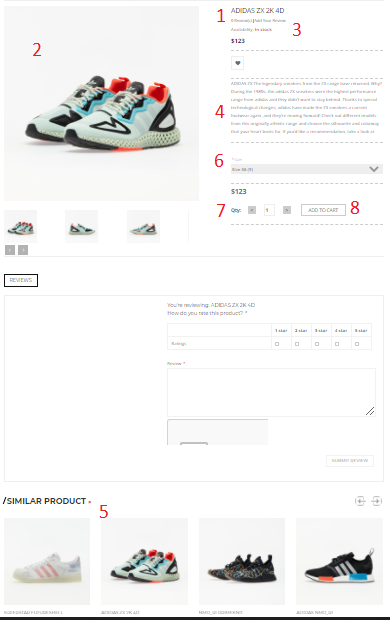
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | New Products | Text |  |  | Show New product and is image . |
| 2 | Image | Link |  | Click | Show image of products and go to product detail if click it. |
| 3 | Side bar  (category:   * Addidas. * Nike. * Puma. * Converse. | Button |  | Click | Show products follow category. |

### Product Page



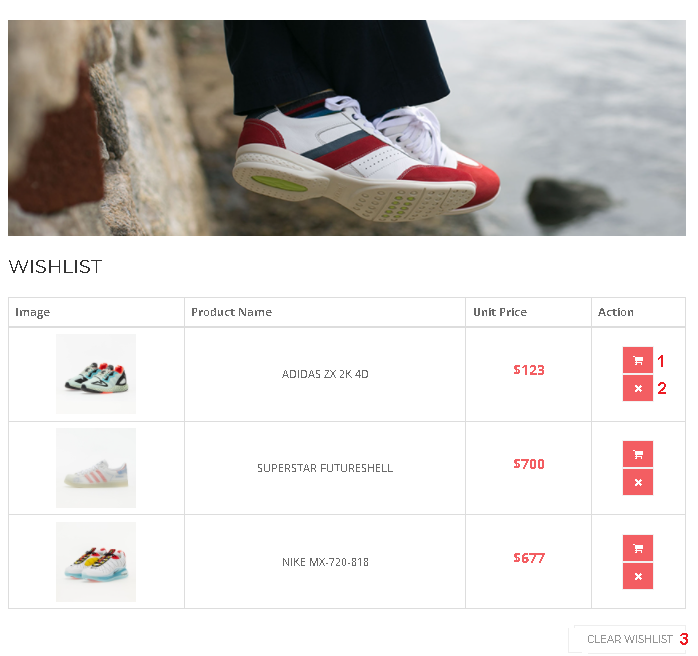
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Search by Brands | Button |  | Click | Show products follow category. |
| 2 | Search by Function | Button |  | Click | Show products follow function. |
| 3 | Search by price | Button |  | Click | View products by price. |
| 4 | Image | link |  | click | See product detail |
| 5 | Add to cart | button |  | click | Add product to cart |
| 6 | Add to wishlist | Button |  | Click | Add product to wishlist |

### Product detail



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Product name | Text |  |  | Show name of product. |
| 2 | Image | Text |  |  | Show image of named product. |
| 3 | Rating | Image |  | Click | View rate of product. |
| 4 | Description | Text |  |  | Show description of product. |
| 5 | Similar product | Image |  |  | Show similar product. |
| 6 | Size | Input | Available Option. |  | Choose a size for product. |
| 7 | Quantity | Input | Number. |  | Add quantity for product. |
| 8 | Add to Cart | Button |  | Click | Add product to cart and redirect to Cart page. |

### View Wishlist



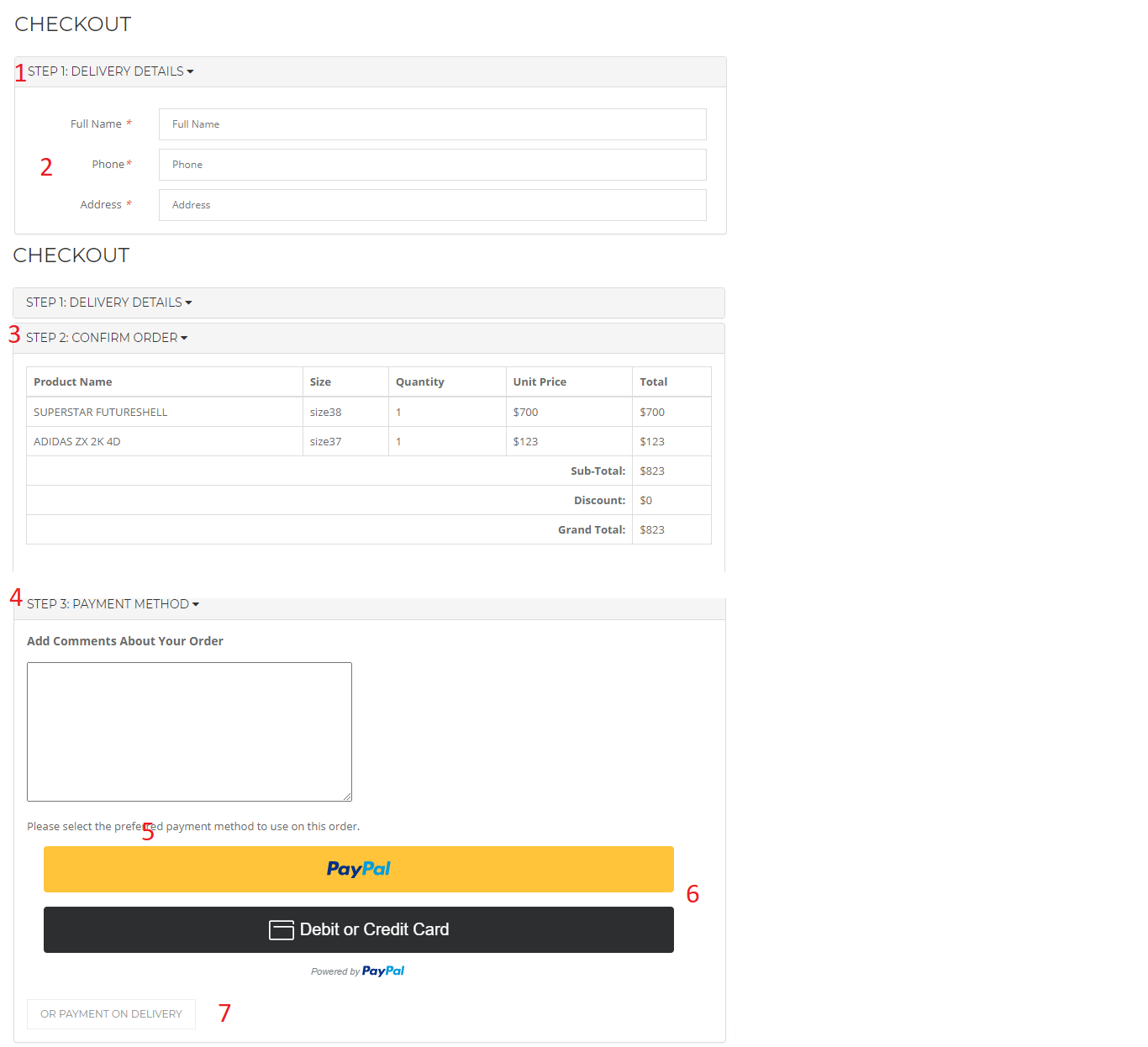
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Add to Cart | Button |  | Click | Add product to cart. |
| 2 | Remove button | Button |  | Click | Remove selected product. |
| 3 | Continue | Button |  | Click | Go to Product page. |

### View cart



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Remove button | Button |  | Click | Remove selected product. |
| 2 | Check out | Button |  | Click | Go to check out page. |
| 3 | Table (Move) | Button |  | Click | Add selected product to Wishlist. |
| 4 | Table (Qty) | Input field | Number | Add | Add quantity of selected products. |
| 5 | Discount | Input field | text | Add | Add Discount Code |
| 6 | Clear Shopping Cart | Button |  | Click | Clear all product in cart. |
| 7 | Update Shopping Cart | Button |  | Click | Update product in cart after edit quantity. |
| 8 | Continue Shopping | Button |  | Click | Go to Product page. |
| 9 | Discount | Button |  | Click | Send discount code |

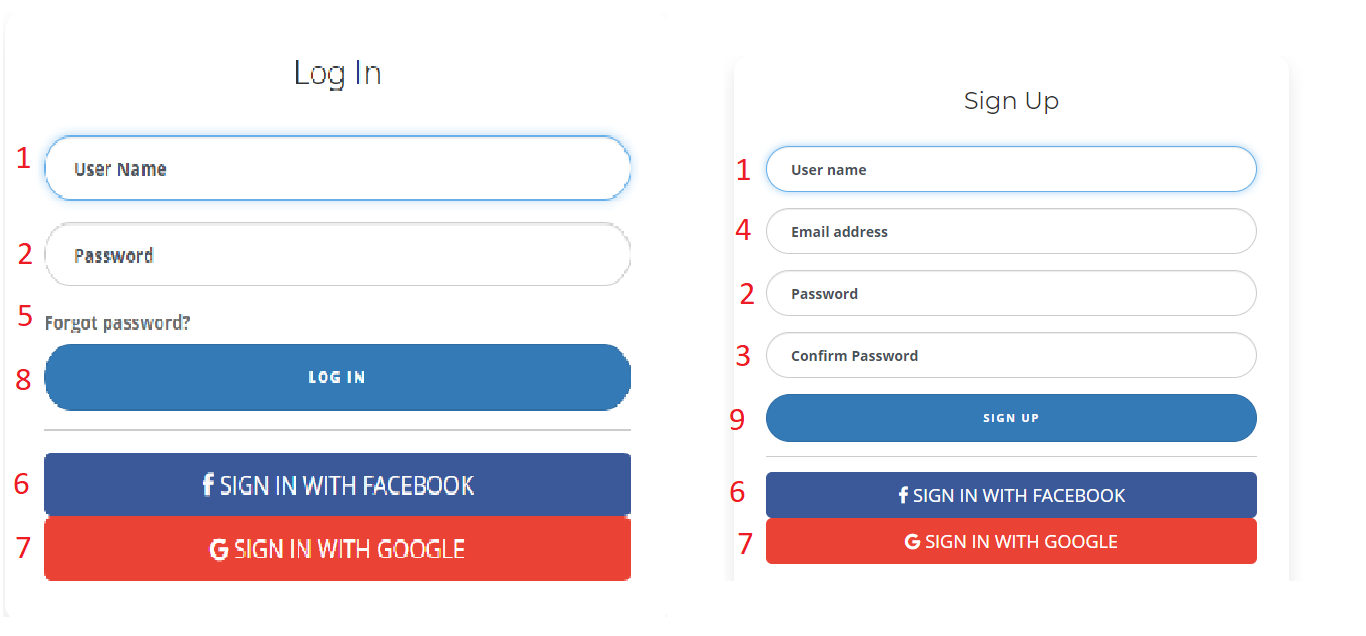
### Checkout



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Delivery Detail | Button |  | Click | Show Form information of buyer. |
| 2 | -Full Name  -Phone  -Address | Input | Text  Number  Text |  | Fill information of buyer in Form. |
| 3 | Confirm Order | Button |  | Click | Show information of Order and Total price. |
| 4 | Payment Method | Button |  | Click | Show payment method. |
| 5 | Comment Order | Input | Text |  | Say something about order |
| 6 | Payment online | Button |  | Click | Payment online |
| 7 | Payment on delivery | Button |  | Click | Payment on delivery |

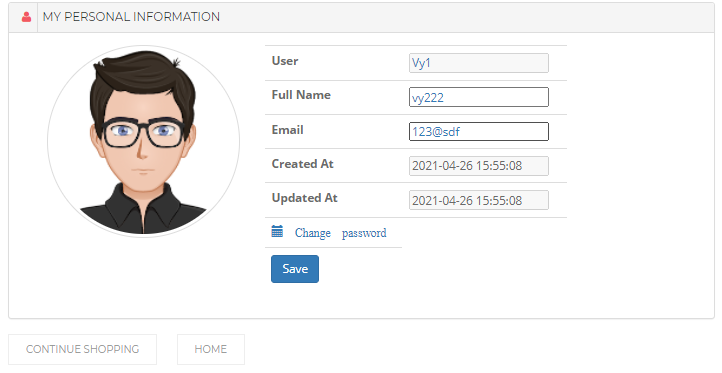
### Customer

#### Customer Login & Register



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Username | Text | Not null, not number |  | Customer’s name |
| 2 | Password | text | Not null, not number |  | Customer’s password |
| 3 | Confirm Password | text | Not null, not number |  | Re-Enter Customer’s password |
| 4 | Email Address | text | Not null, must be email (have .com and @) |  | Customer’s email |
| 5 | Forgot password | button |  | Click | Go to Forgot password page |
| 6 | Sign in with Facebook | Button |  |  | Open a pop up to sign in with facebook. |
| 7 | Sign in with Google | Button |  |  | Open a pop up to sign in with google. |
| 8 | Login | Button |  | Click | Link to index page |
| 9 | Register | Button |  | Click | Link to index page |
|  | | | | | |

Customer information



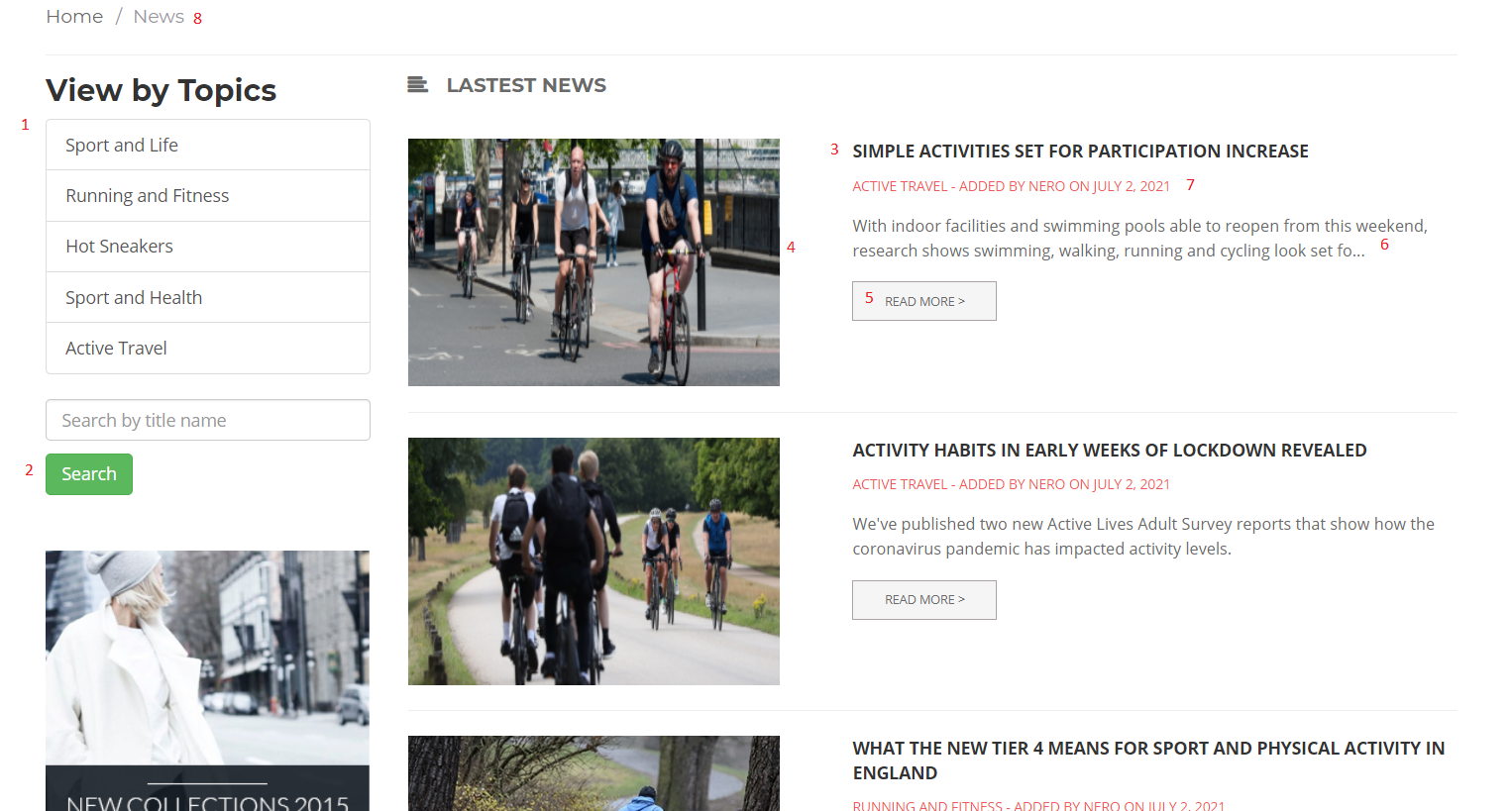
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Username | Null | Not null, not number |  | Customer’s name |
| 2 | Fullname | Null | Not null, not number |  | Customer’s full name |
| 3 | Email Address | Null | Not null, must be email (have .com and @) |  | Customer’s email |
| 4 | Created\_at | Text |  |  | Show date time when create a customer’s account. |
| 5 | Updated\_at | Text |  |  | Show date time when update a customer’s account. |
| 6 | Change password | Button | Not null | Click | Change Customer’s password |
| 7 | Save | Button | Not null | Click | Complete updating Customer’s account |
|  | | | | | |

About US



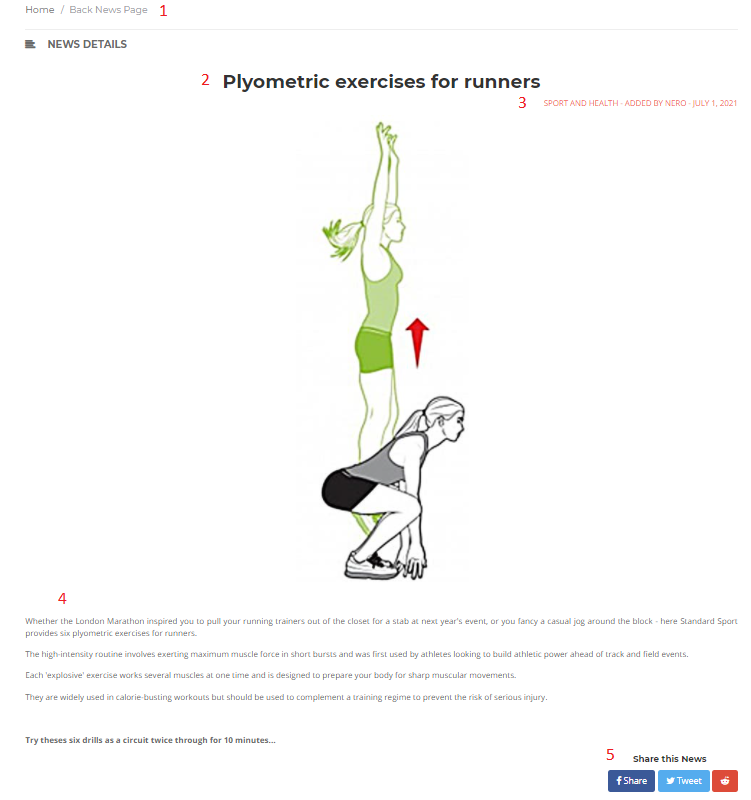
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | About Us | Text |  |  | Show introduce information about Sneaker Online shop. |
| 2 | Meet the team | Text |  |  | Show information and avatar about team’s members. |
|  | | | | | |

#### **NEWS**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Topics | Button |  | Click | Show News follow topics. |
| 2 | Search | Button |  | Click | Search News by title name. |
| 3 | Title | Link |  | Click | Go to Blog News Detail. |
| 4 | Image | Link |  | Click | Go to Blog News Detail. |
| 5 | Read more | Button |  | Click | Go to Blog News Detail. |
| 6 | Preview content | Text |  |  | Show preview content |
| 7 | Admin post/Date News post | Text |  |  | Show name Admin post and date of posting News |
| 8 | News | Link |  | Click | Refresh News page |

#### **BLOG NEWS DTAILS**



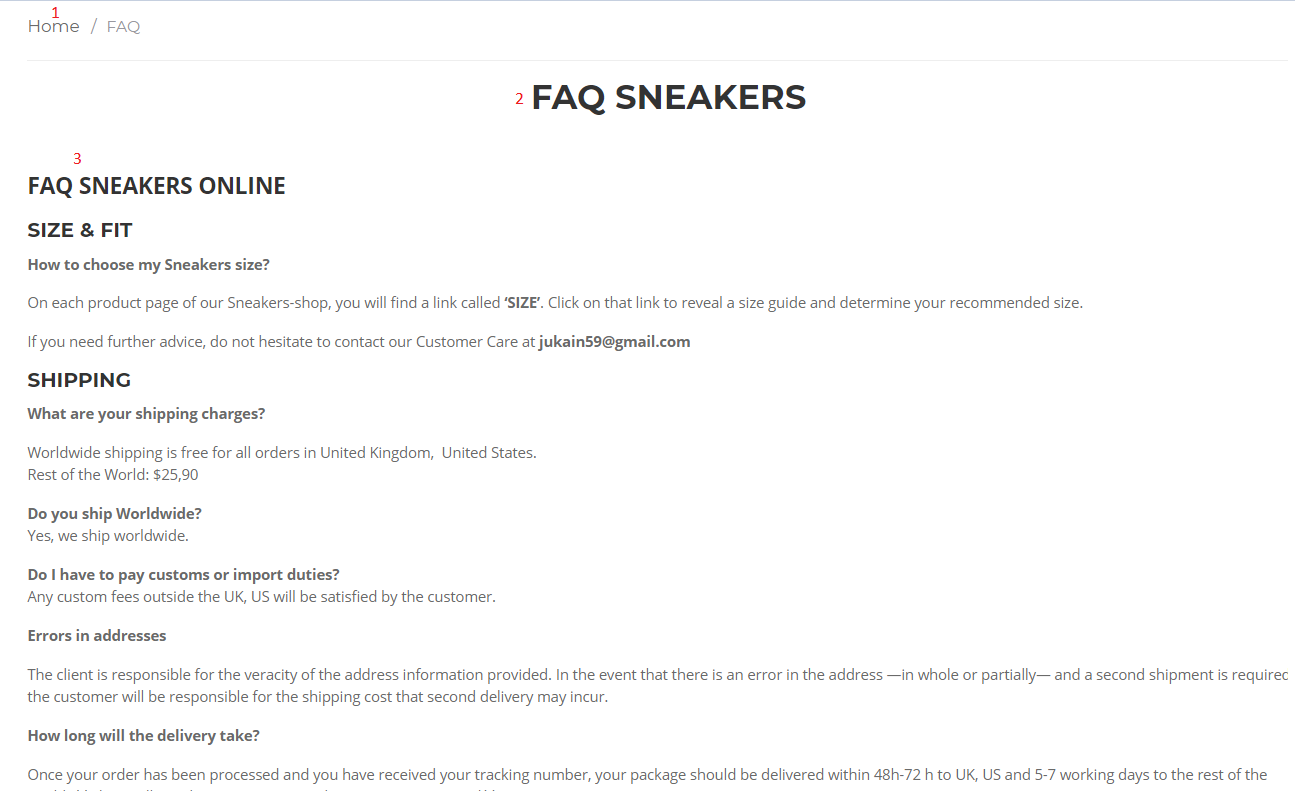
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Back News Page | Link |  | Click | Redirect to main News Page |
| 2 | Title | Text |  |  | News Title |
| 3 | Admin post/Date News post | Text |  |  | Show name Admin post and date of posting News |
| 4 | News Content | Text |  |  | Show Blog News‘s content. |
| 5 | Share this News | Button |  | Click | Share current News to socials network. |

#### **Order History and Details**



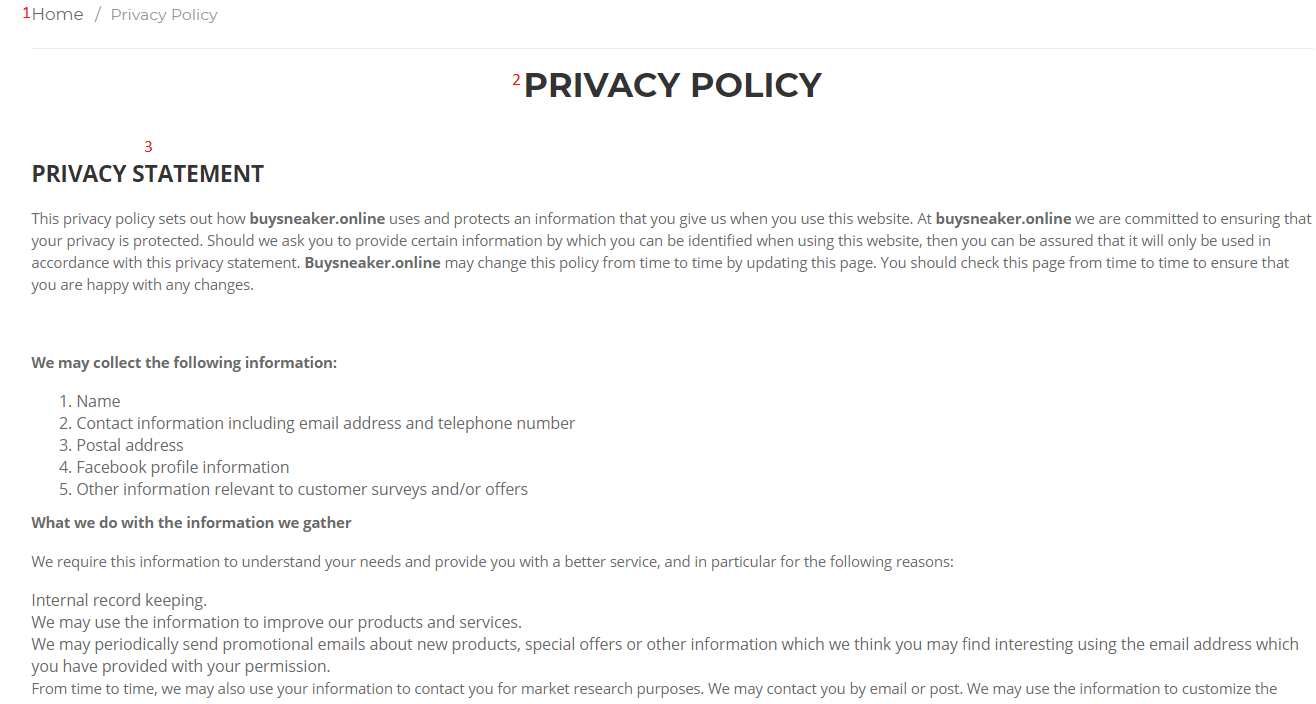
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Order History and Details | NULL |  |  | Show information about order. |
| 2 | Time | Text |  |  | Show time when order was created. |
| 3 | Total | Text |  |  | Show total price of order |
| 4 | Process | Text |  |  | Show status of order |
| 5 | Detail | Text |  |  | Show Order details |
| 6 | Cancel | Button |  | Click | Cancel order |

#### **FAQs**



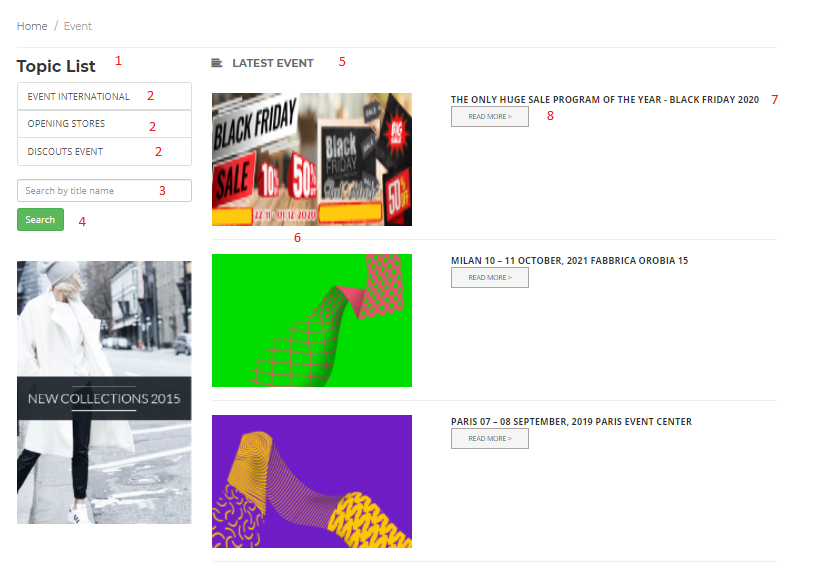
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Home | Link |  | Click | Redirect to Home Page |
| 2 | FAQ Sneakers | Text |  |  | FAQ Title |
| 3 | FAQ content | Text |  |  | Content of FAQ. |

#### **Policy**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Home | Link |  | Click | Redirect to Home Page |
| 2 | Privacy Policy | Text |  |  | Privacy Title |
| 3 | Privacy content | Text |  |  | Content of Policy |

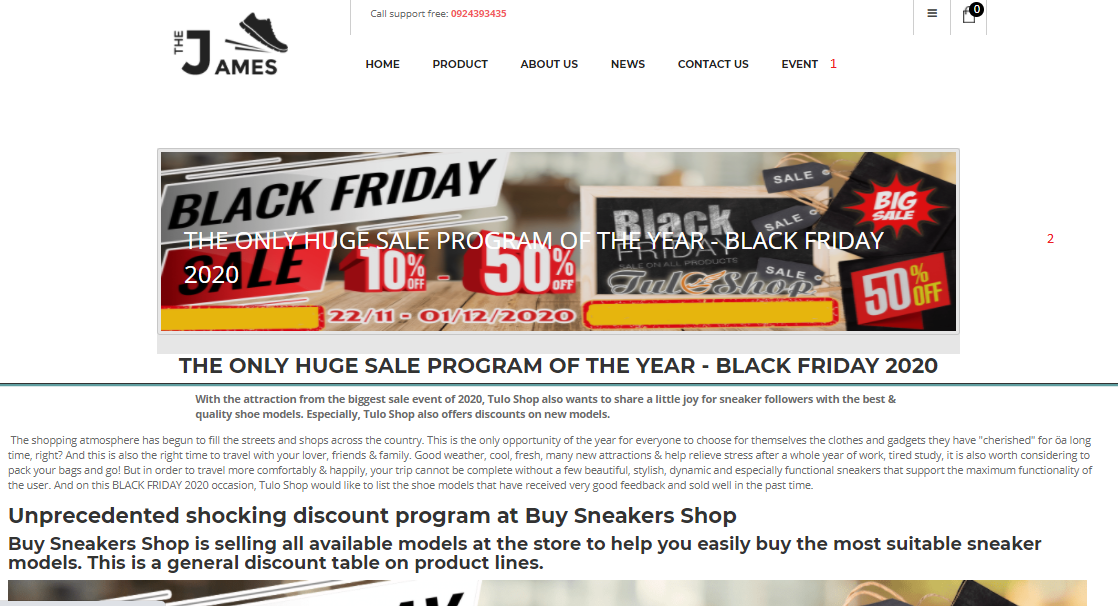
#### **Event**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Topic List |  | No | Topic Event |
| 2 | Name Topic |  | Click | Summarize the topics event |
| 3 | Search by title name |  | Input | Requied Input Value |
| 4 | Search |  | Click | Click to Search |
| 5 | Latest Event |  | Click | Click Will Wiew All New Event |
| 6 | Image Event |  | Click | Wiew Image of Event |
| 7 | Name Event |  | Click | Name event name and clickable |
| 8 | Read More |  | Click | Click here and see post event |
|  |  |  |  |  |

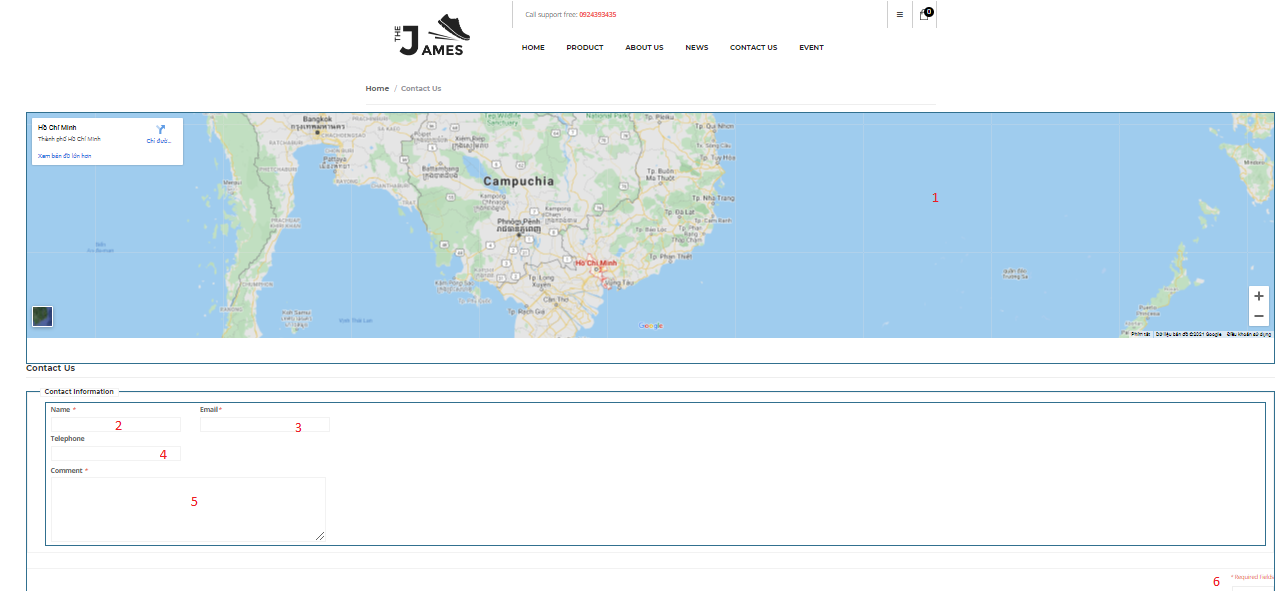
#### **Event Detail**

Clicking on any article will navigate to the detailed article



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Categoty |  | Click | Back Page Event |
| 2 | Post Event |  | Not Event | Only Event |

#### **Contact Us**

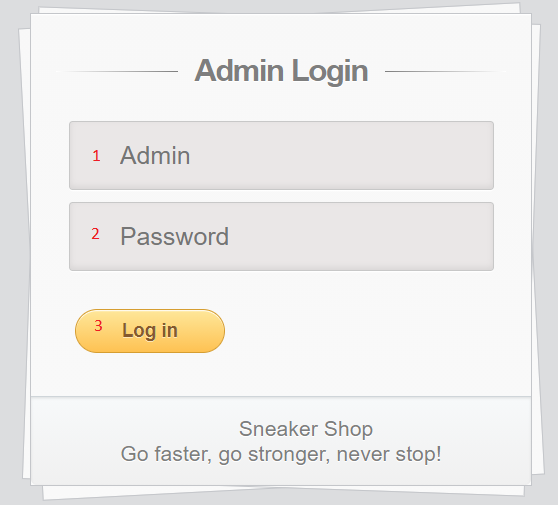
******

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Type** | **Description** |
| 1 | Map |  | Zoom or double click | Link | Wiew Location Store |
| 2 | Name |  | Fill name | Text | Customer Name |
| 3 | Email |  | Fill mail | Text | Customer Mail |
| 4 | Telephone |  | Fill Telephonr | number | Customer Telephone |
| 5 | Comment |  | Fill Comment | Text | Customer Comment |
| 6 | Submit |  | Click | Click | Sent Form |

### **Admin**

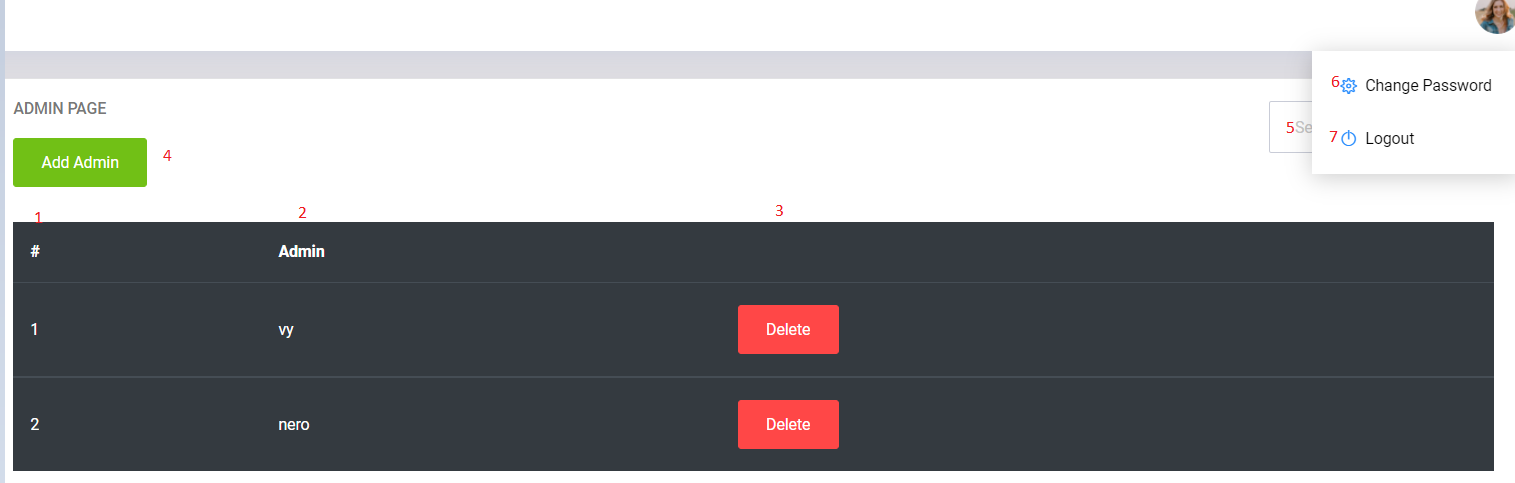
#### You must first login before entering the site

**Note:** Because a member can only have an admin account. So, admin must remember his/her password by writing to a note or save on cloud, google document. We don’t make a function “Forgot Password” because of security web and we apply this as same as Bitcoin Wallet security : You can’t access when you forgot. However, admin can still have back his/her account by contact to an database admin who manage data of website, he/she can access to database where store data information of admin and get back password. The database manager must be reliable.



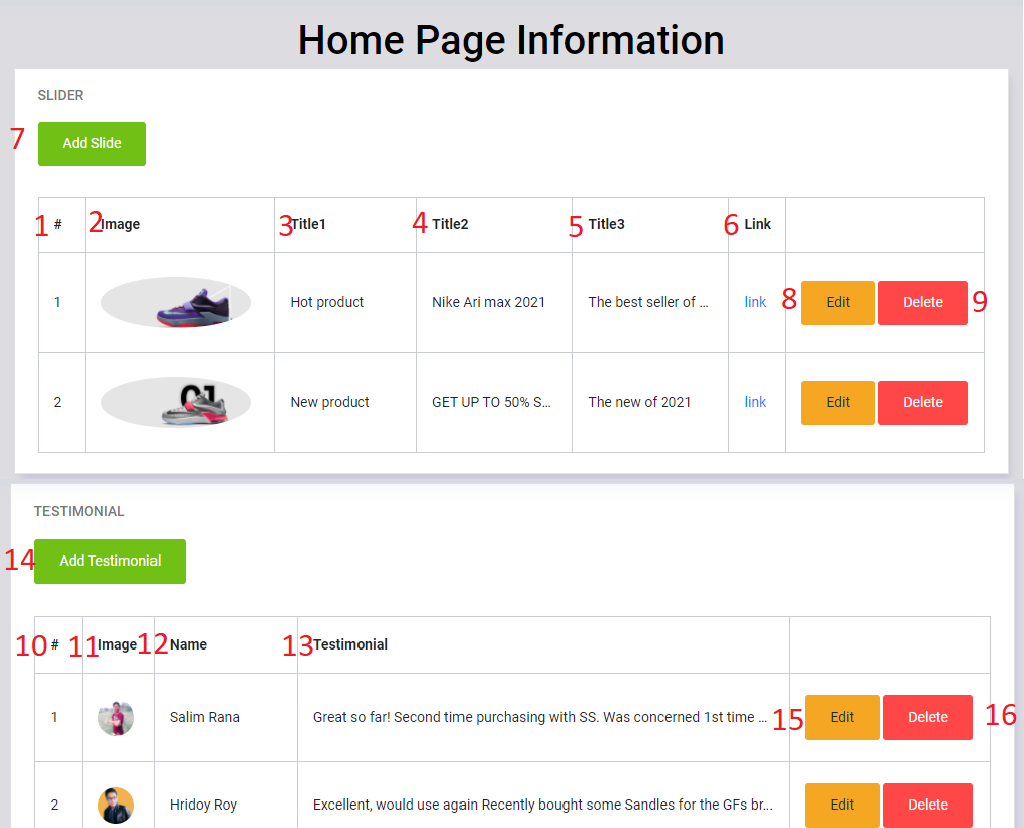
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Admin | Null | Not null, force matches in the database. |  | Admin’s account name. |
| 2 | Password | Null | Not null, force matches in the database. |  | Admin’s password. |
| 3 | Log in | Button |  | Click | Link to admin’s page if login success. |

#### **Admin manager**



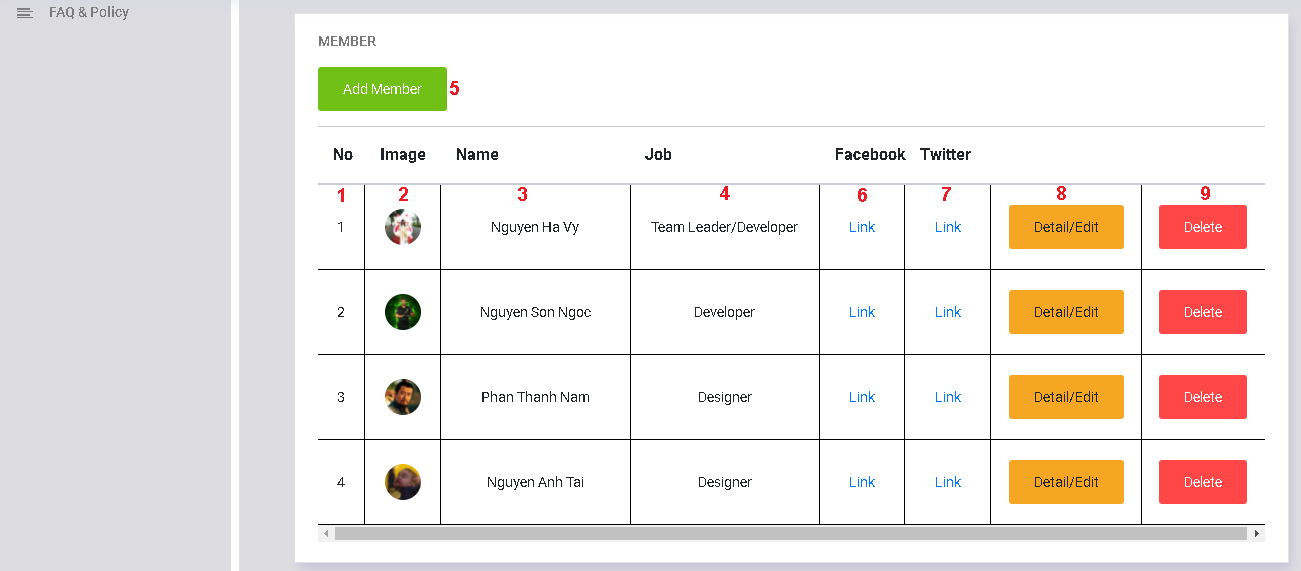
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Name** | **Type** | **Validation** | **Event** | **Description** |
| 1 | Slider No. | Not null, must be number. |  | Slider’s no. | Slider No. |
| 2 | Admin | Null | Not null, force matches in the database. |  | Admin’s account name. |
| 3 | Delete | Button |  | Click | Delete an admin. |
| 4 | Add Admin | Button |  | Click | Add an new admin. |
| 5 | Search admin | Input |  | Enter | Show an admin following name. |
| 6 | Change Password | Button |  | Click | Change password of current admin’s account. |
| 7 | Log Out | Button |  | Click | Quit current admin’s account. |

#### **Home manager**



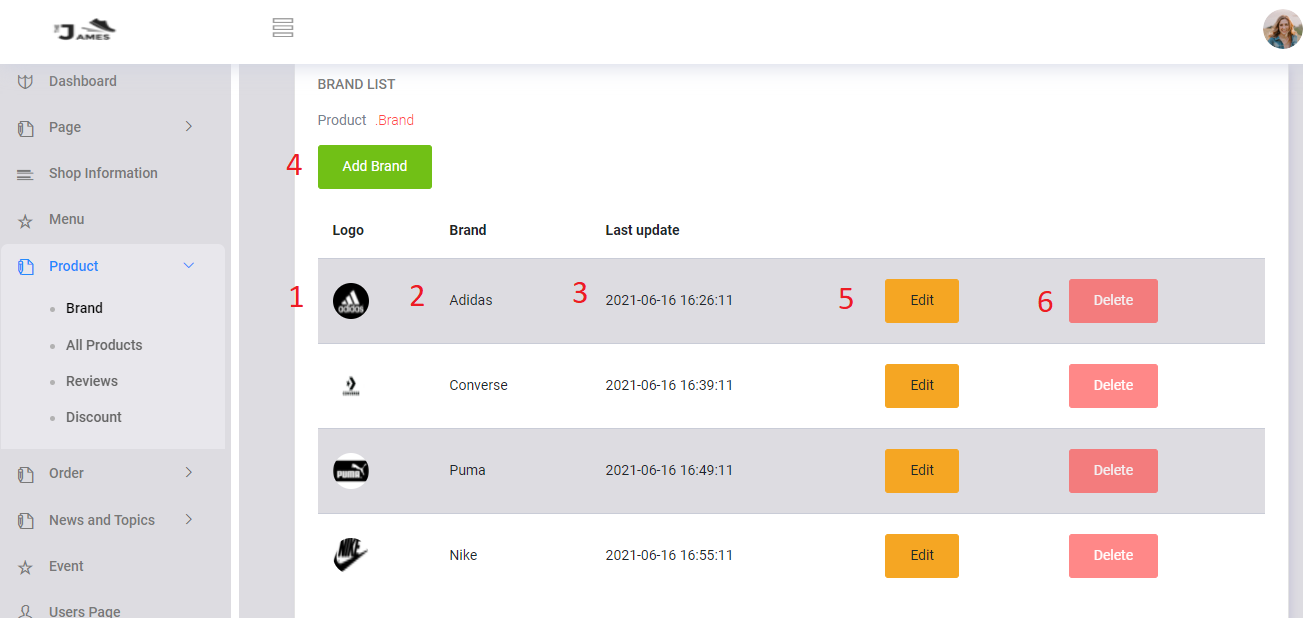
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Slider No. | Not null, must be number. |  | Slider’s no. |
| 2 | Image | Not null, force matches in the database. |  | Slider’s image |
| 3 | Title1 | Not null, force matches in the database. |  | Slider’s tilte. |
| 4 | Title2 | Not null, force matches in the database. |  | Slider’s tilte. |
| 5 | Title3 | Not null, force matches in the database. |  | Slider’s tilte. |
| 6 | Link | Not null, force matches in the database. |  | Link to product page. |
| 7 | Add Slide |  | Click | Link to add new slider. |
| 8 | edit |  | Click | Link to update slider’s form. |
| 9 | delete |  | Click | Link to delete slider’s form. |
| 10 | No. | Not null, must be number. |  | Feedback’s no. |
| 11 | Image | Not null, force matches in the database. |  | Feedback’s image |
| 12 | Name | Not null, force matches in the database. |  | Name of customer who did feedback. |
| 13 | Testimonial | Not null, force matches in the database. |  | Testimonial. |
| 14 | Add Testimonial |  | Click | Link to add new Feedback. |
| 15 | edit |  | Click | Link to update Feedback’s form. |
| 16 | delete |  | Click | Link to delete Feedback’s form. |

#### **About Us manager**



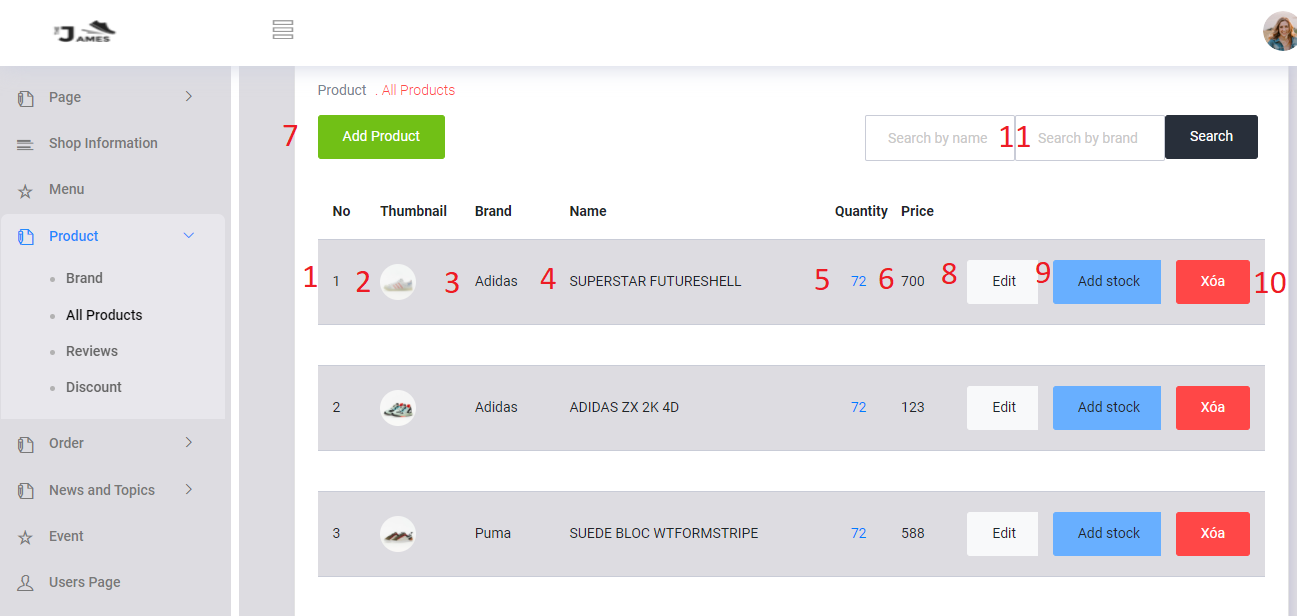
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Id member | Not null, must be number. |  | Member’s id. |
| 2 | Image | Not null, force matches in the database. |  | Member’s avatar. |
| 3 | Name | Not null |  | Member’s name. |
| 4 | Job | Not null |  | Member’s job. |
| 5 | Add member |  | Click | Link to add new member. |
| 6,7 | Link |  | Click | Member’s Facebook & Twitter link. |
| 8 | Detail/Edit |  | Click | Link to update member information. |
| 9 | delete |  | Click | Link to delete member. |

#### **Brands (category) manager**



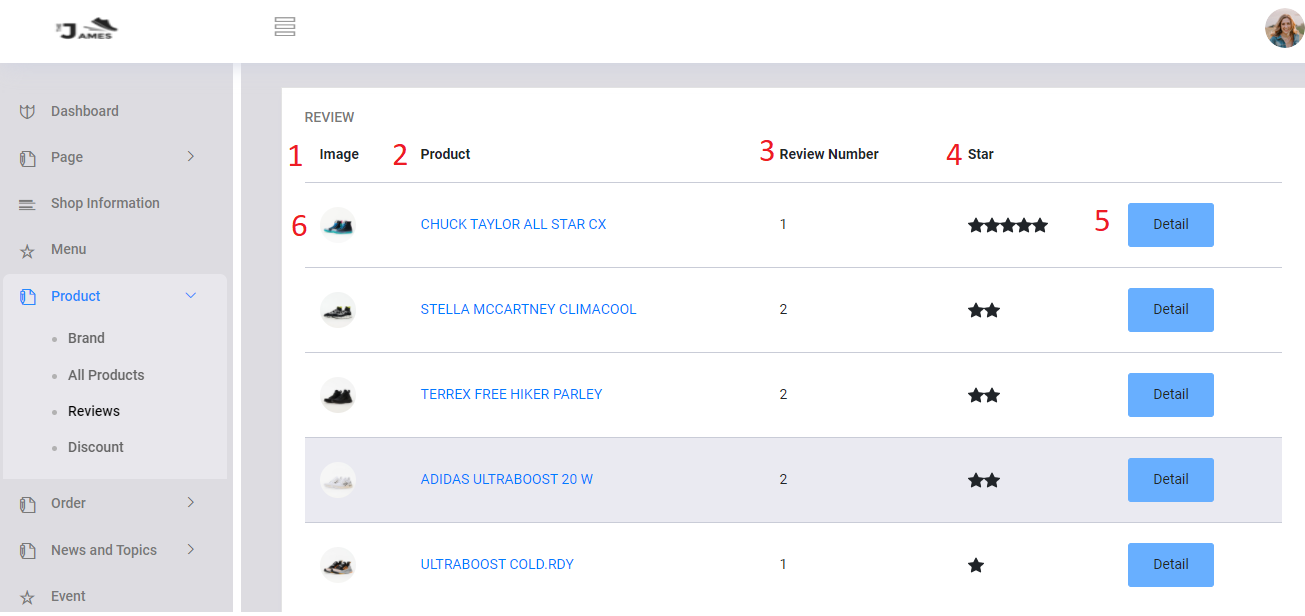
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Logo | Not null, force matches in the database. |  | Brand’s image. |
| 2 | Brand | Not null, force matches in the database. |  | Brand’s name. |
| 3 | Last update |  |  | Brand’s time update. |
| 4 | Add brand |  | Click | Link to add new category about us. |
| 5 | Edit |  | Click | Link to update Brand’s form. |
| 6 | delete |  | Click | Link to delete Brand’s form. |

Product manager



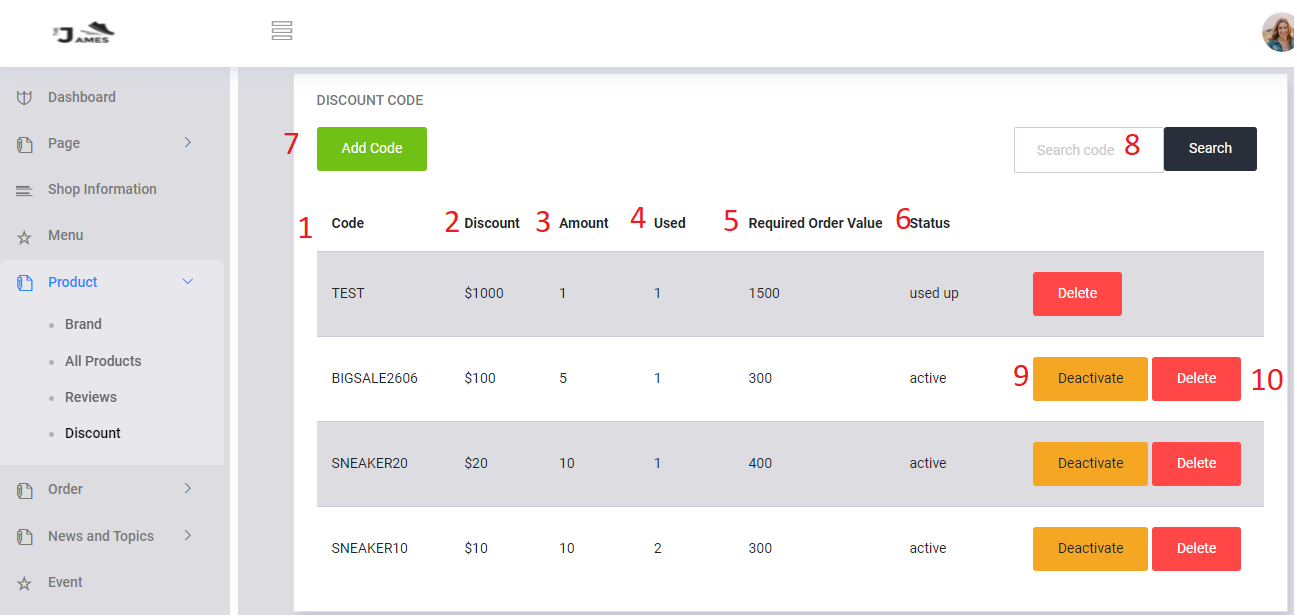
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | No | Not null, must be number. |  | Product’s No. |
| 2 | thumbnail | Not null, force matches in the database. |  | Product’s image. |
| 3 | Brand | Not null, force matches in the database. |  | Product’s category. |
| 4 | name | Not null, force matches in the database. |  | Product’s name. |
| 5 | Quantity | Not null, force matches in the database. |  | Product’s quantity. |
| 6 | price | Not null, must be number. |  | Product’s price. |
| 7 | Add product |  | Click | Link to add category product form. |
| 8 | edit |  | Click | Link to update product form. |
| 9 | Add stock |  | Click | Link to add stock product form. |
| 10 | Delete |  | Click | Link to delete product form. |

#### **Review**



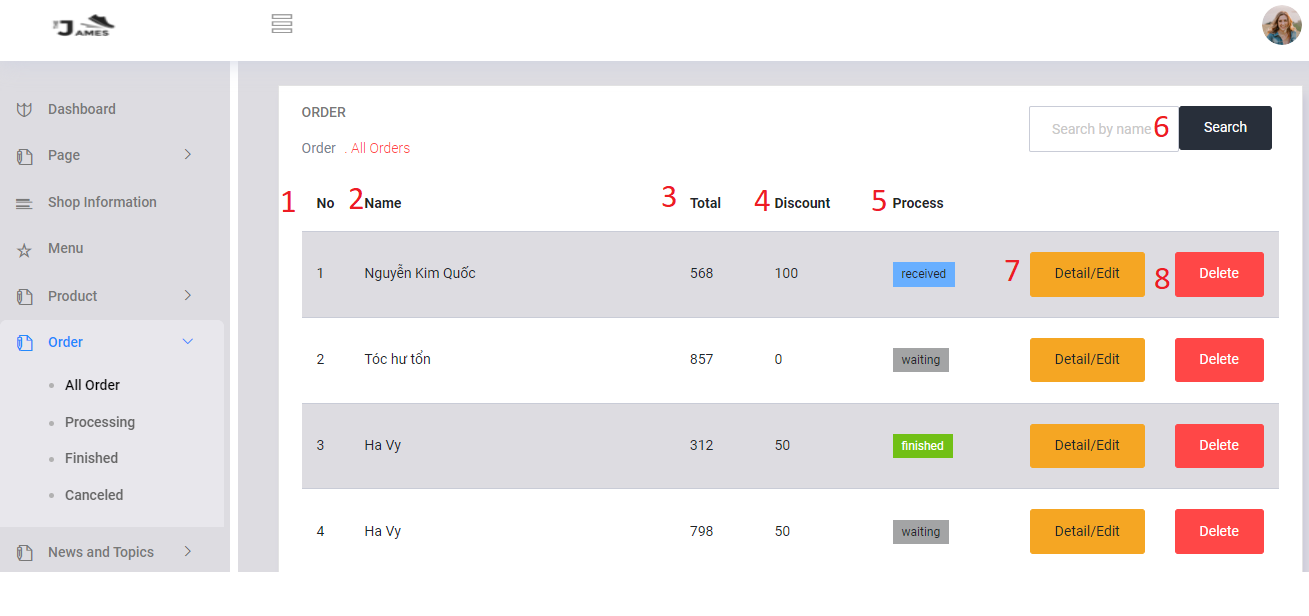
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | image | text |  | Product’s image |
| 2 | Product | text | Go to see product | Product’s name. |
| 3 | Review number | number |  | Number of review |
| 4 | Star | number |  | Product’s star |
| 5 | Detail | text |  | Go to review-detail page |

#### **Discount**



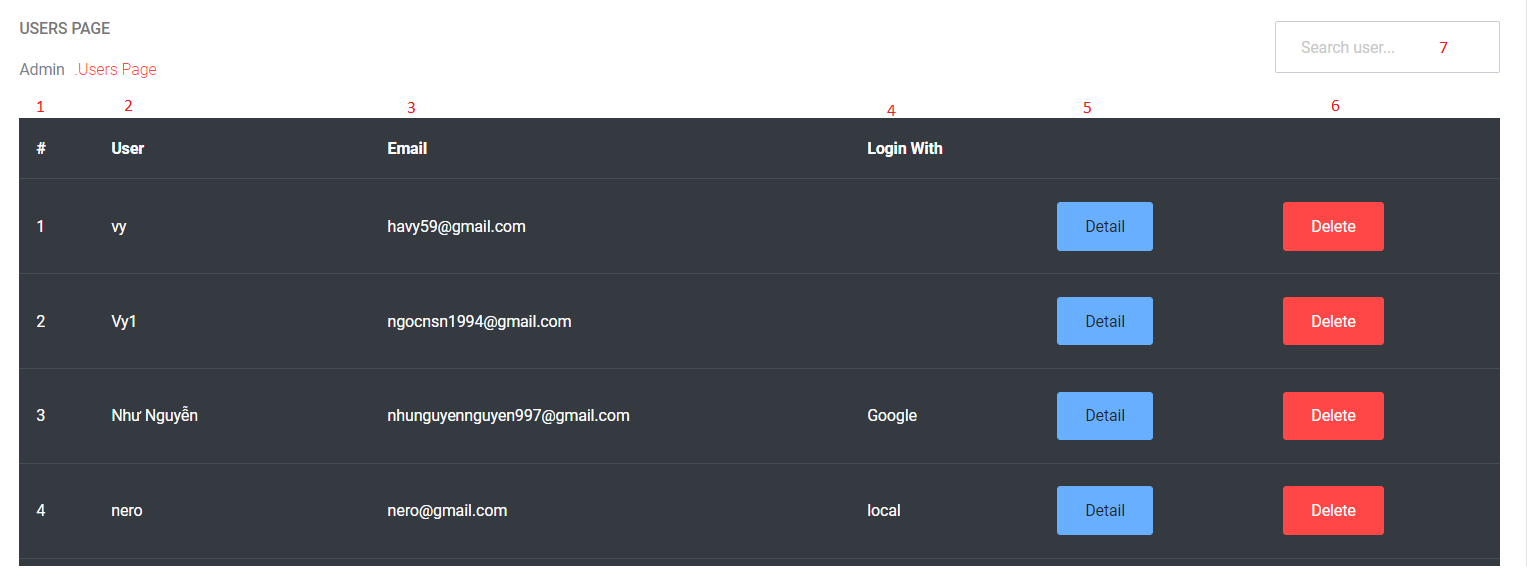
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Code | Text |  | Code Discount |
| 2 | Discount | Not null, must be number. |  | Discount’s value |
| 3 | Amount | Not null, must be number. |  | Amount of code |
| 4 | Used | Not null, not number. |  | Number of used discount |
| 5 | Required Order Value | Not null, must be number. |  | Minimum Value of Order to get discount |
| 6 | Status | text |  | Status of code |
| 7 | Add code | button | click | Go to Add-code page |
| 8 | Search |  | input | Search by code |
| 9 | Deactivate /Active | button | click | Active or deactivate code |
| 10 | Delete |  | Click | Delete code. |

#### Order



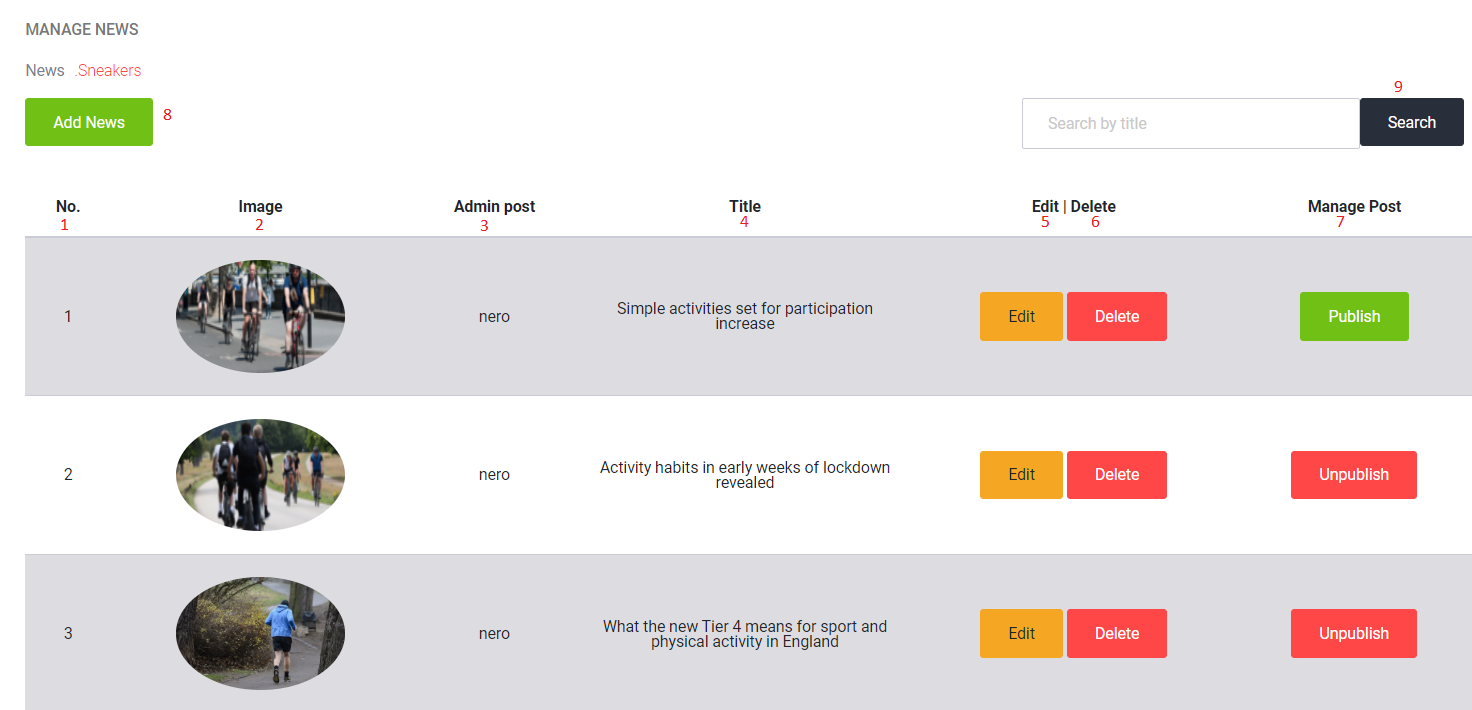
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | No | Not null, must be number. |  | Order’s No. |
| 2 | Name | Not null, not number. |  | Customer’s name. |
| 3 | Total | Not null, must be number. |  | Order’s price. |
| 4 | Discount | Not null, must be number. |  | Order’s discount. |
| 5 | Process | Text |  | Order’s status |
| 6 | Search | text | input | Search order by name |
| 7 | Detail/Edit | button | click | Go to detail order page |
| 8 | Delete |  | Click | Delete order. |

#### **Customer manager**



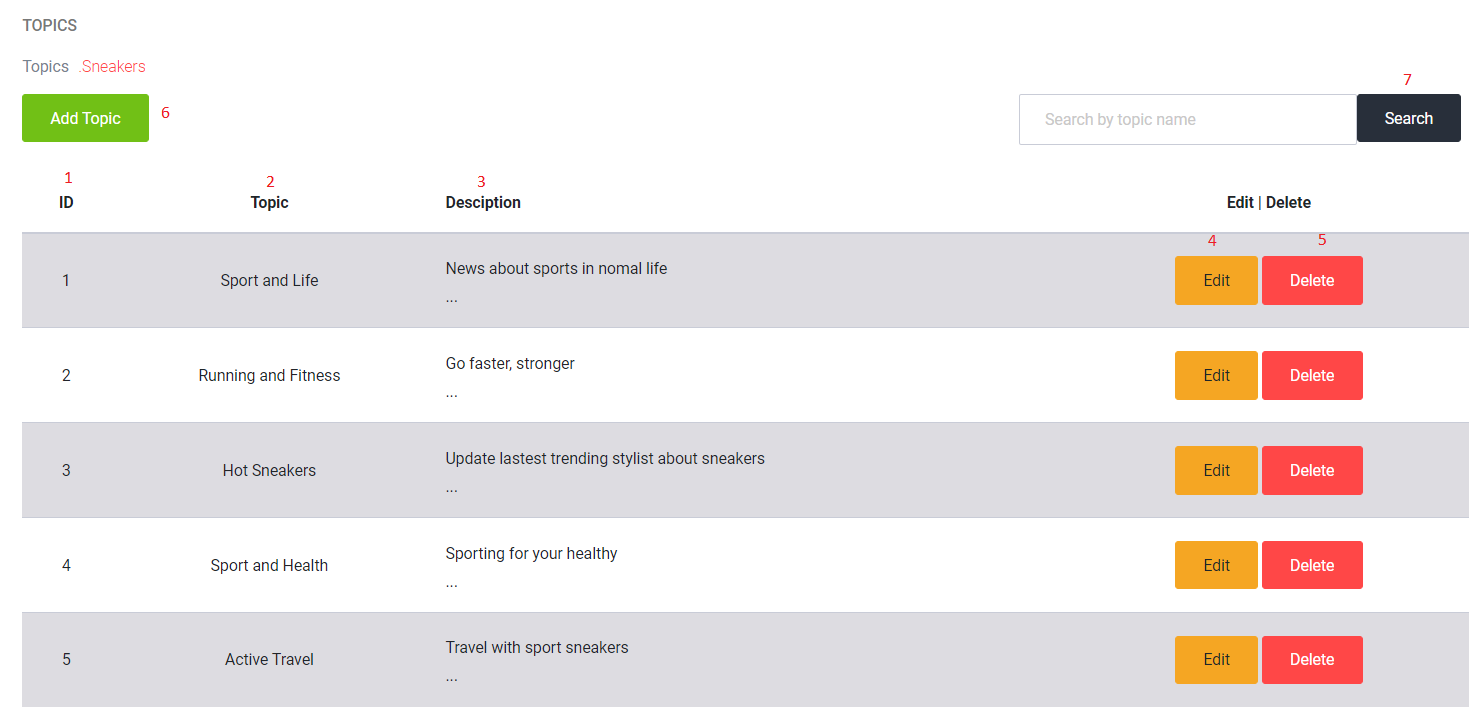
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | No | Not null, must be number. |  | Customer’s No. |
| 2 | username | Not null, force matches in the database. |  | Customer’s account name. |
| 3 | Email | Not null, force matches in the database. |  | Customer’s email. |
| 4 | Login with |  |  | Show method login of customer |
| 5 | Detail |  | Click | Link to detail customer form. |
| 6 | Delete |  | Click | Link to delete customer form. |
| 7 | Search |  | Enter | Show Customer’s account by user. |

#### **News management**



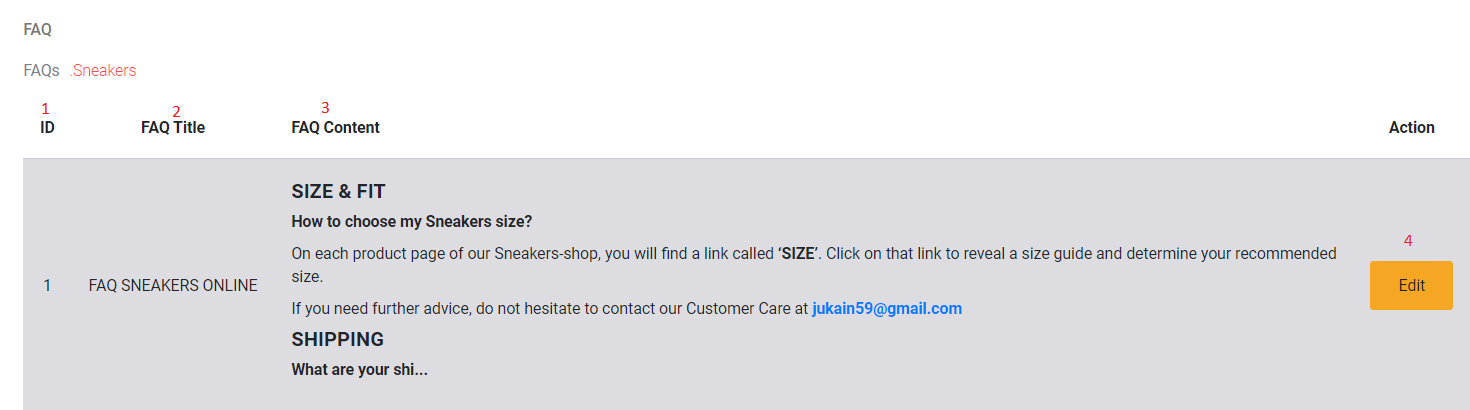
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Slider No. | Not null, must be number. |  | Slider’s no. |
| 2 | image | Not null, force matches in the database. |  | News’s image. |
| 3 | Admin post | Not null, force matches in the database. |  | Admin’s name who posted a News. |
| 4 | Title | Not null, force matches in the database. |  | News’s title. |
| 5 | Edit |  | Click | Link to edit news form. |
| 6 | Delete |  | Click | Link to delete news form. |
| 7 | Publish/Unpublish |  | Click | Manage showing or hide News in News Page. |
| 8 | Add News |  | Click | Link to create news form. |
| 9 | Search |  | Click | Search News by title name. |

#### **Topics management**



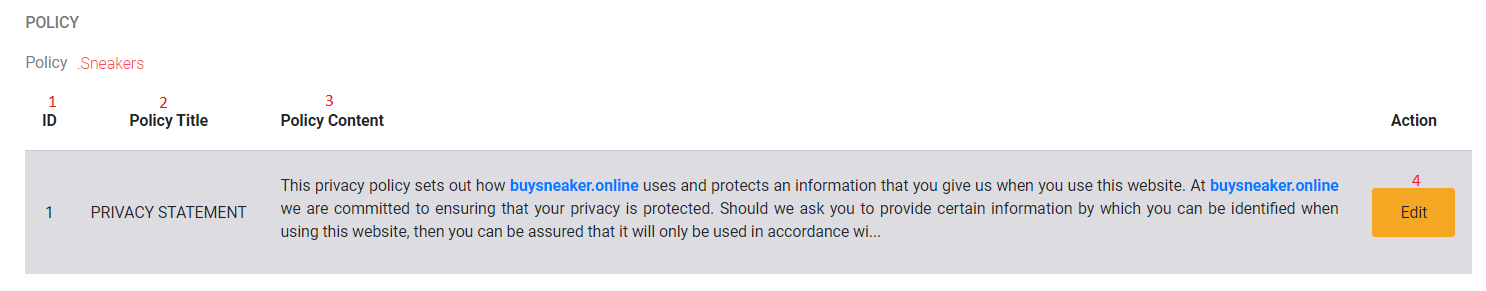
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Slider No. | Not null, must be number. |  | Slider’s no. |
| 2 | Topic Name | Not null, force matches in the database. |  | Topic’s name. |
| 3 | Description | Not null, force matches in the database. |  | Explain meaning of topic name. |
| 4 | Edit |  | Click | Link to edit Topic form. |
| 5 | Delete |  | Click | Link to delete Topic form. |
| 6 | Add Topic |  | Click | Link to create Topic form. |
| 7 | Search |  | Click | Search Topics by topic name. |

#### **FAQs**



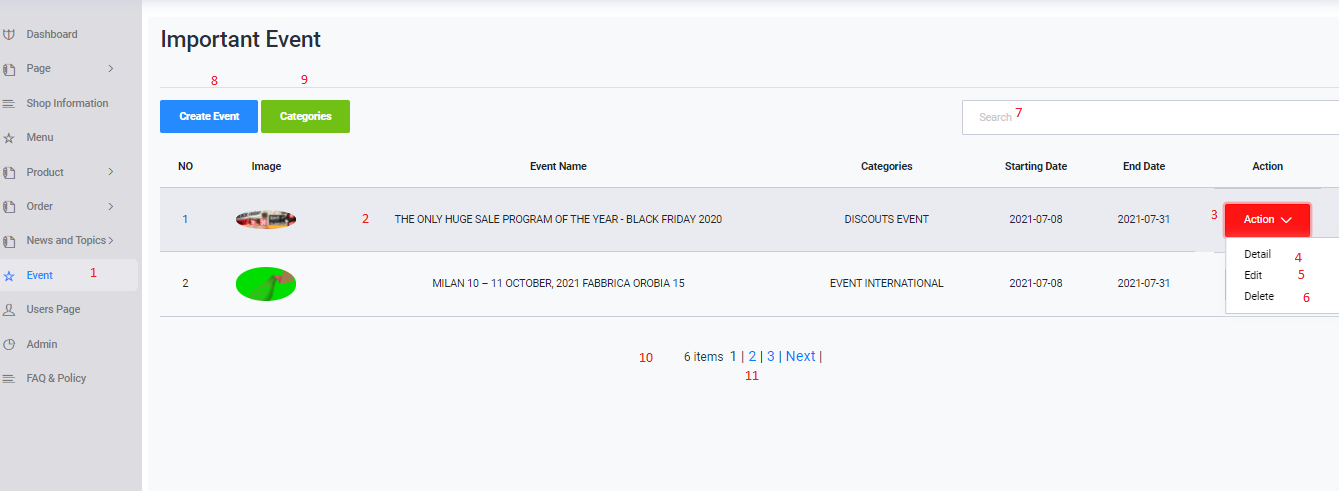
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | ID. | Not null, must be number. |  | FAQ’s id. |
| 2 | FAQ Title | Not null, force matches in the database. |  | FAQ’s title. |
| 3 | FAQ Content | Not null, force matches in the database. |  | Content of FAQ |
| 4 | Edit |  | Click | Link to edit FAQ form. |

#### **Privacy Policy**



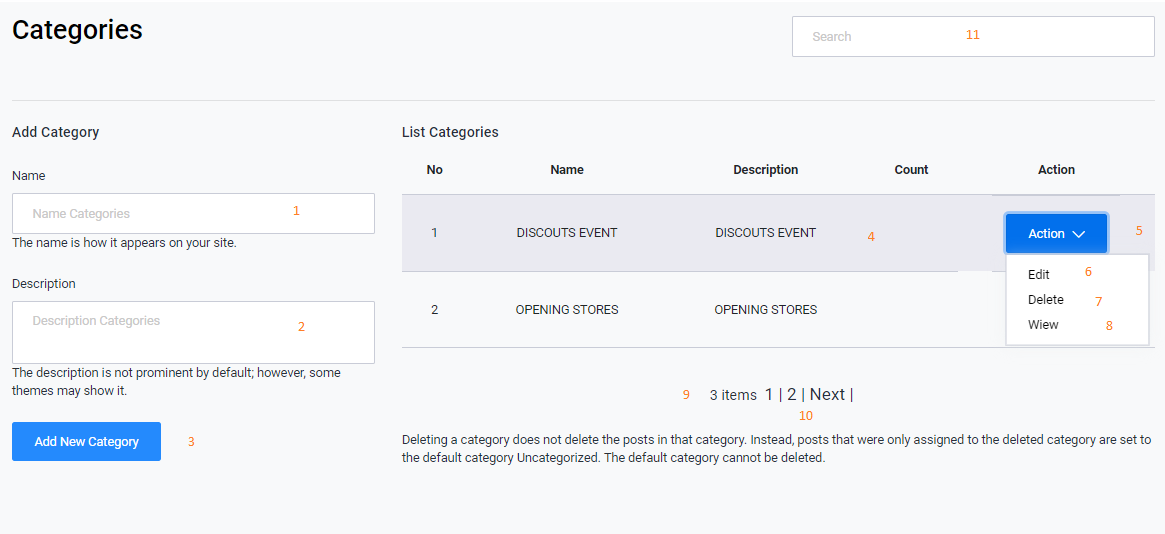
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | ID. | Not null, must be number. |  | Policy’s id. |
| 2 | Policy Title | Not null, force matches in the database. |  | Policy’s title. |
| 3 | Policy Content | Not null, force matches in the database. |  | Content of Policy |
| 4 | Edit |  | Click | Link to edit Policy form. |

#### **Event**



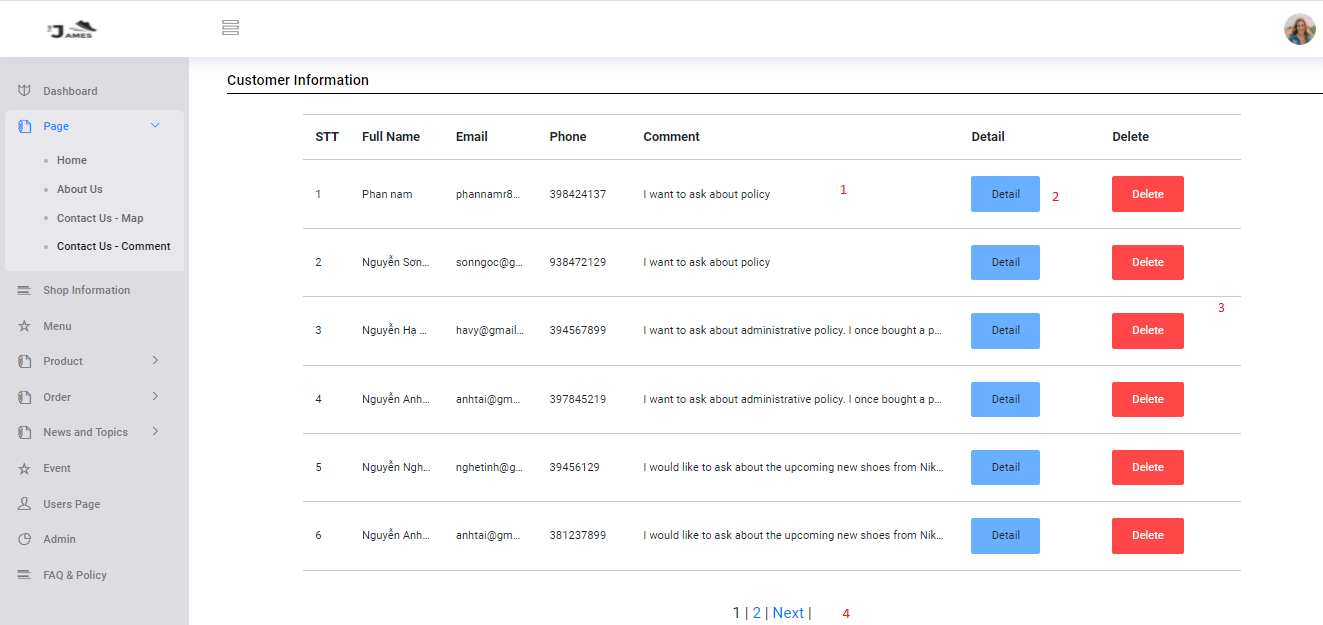
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Event |  | Click | Back page admin event |
| 2 | Post Event |  | Not Event | Wiew post Event |
| 3 | Action |  | Click | Wiew Detail/Edit/Delete |
| 4 | Detail |  | Click | Access to the other Page Detail Post Event |
| 5 | Edit |  | Click | Access to the other Page Edit Post Event |
| 6 | Delete |  | Click | Delete Post Event Selected |
| 7 | Search |  | Fil Value and Enter | Search Post Event |
| 8 | Create Event |  | Click | Acces to the other Page Create Event |
| 9 | Categories |  | Click | Acces to the other Page Categories |
| 10 | Number Wiew |  | Not Evet | Wiew all post Event |
| 11 | Pagination |  | Click | Acces to the Order Next Page |

#### **Event** [**Categories**](https://buysneaker.online/admin/event_Categories.php)

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Name |  | Input Value | Name Categories |
| 2 | Description |  | Input Value | Description Categories |
| 3 | Add New Categoy |  | Click | Create New Category |
| 4 | List Categories |  | Wiew | List Categories |
| 5 | Action |  | Click | Edit/Delete/Wiew |
| 6 | Edit |  | Click | Acess to the Other Page Edit Categoy |
| 7 | Delete |  | Click | Delete Category |
| 8 | Wiew |  | Click | Acess to the Other Fontend Page Event |
| 9 | Number Wiew |  | Not Event | All Categories |
| 10 | Pagination |  | Click | Acess to Next Page |
| 11 | Search |  | Input Value and Enter | Search to Categories |

#### **Contact Us Comment**

******

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name** | **Validation** | **Event** | **Description** |
| 1 | Rows Customer information |  | Not Event | View basic info |
| 2 | Detail |  | Click | Access to page detail info |
| 3 | Delete |  | Click | Delete to Row Selected |
| 4 | Pagination |  | Click | Access to next page |

## task sheet review 3

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No.: Sneakers Online** | | **Project Name** | **Activity Plan Prepared By** | **Date of Preparation of Activity Plan:** | | | |
| **Sr./No.** | **Task** | **Actual Start Date** | **Actual Days** | **Team member name** | **Status** |
| 1 | Admin page | Sneakers Online | Vy | 11-Jun-21 | 28 | Ngọc | Completed |
| 2 | Home page | 11-Jun-21 | 28 | Vy | Completed |
| 3 | Product page | 12-Jun-21 | 29 | Vy | Completed |
| 4 | Wishlist page | 12-Jun-21 | 29 | Tài | Completed |
| 5 | Login page | 12-Jun-21 | 30 | Vy | Completed |
| 6 | Register page | 11-Jun-21 | 30 | Vy | Completed |
| 7 | Cart page | 12-Jun-21 | 29 | Vy | Completed |
| 8 | Checkout page | 12-Jun-21 | 28 | Vy | Completed |
| 9 | News page and Topics | 11-Jun-21 | 29 | Ngọc | Completed |
| 10 | About us page | 13-Jun-21 | 3 | Tài | Completed |
| 11 | Contact us page | 13-Jun-21 | 4 | Nam | Completed |
| 12 | FAQ page | 15-Jun-21 | 1 | Ngọc | Completed |
| 13 | Privacy Policy page | 15-Jun-21 | 1 | Ngọc | Completed |
| 14 | Purchase history | 14-Jun-21 | 30 | Vy | Completed |
| 15 | Event page | 13-Jun-21 | 2 | Nam | Completed |

|  |  |
| --- | --- |
| **Date: 18 Jun 2020** | |
| Signature of Instructor:  **Mr. Hoang Duc Quang** | Signature of Team Leader:  **Nguyen Ha Vy** |