

**Tech shop project**

**Software Requirement Specification**

– Hanoi, August 2022 –

Record of changeS

|  |  |  |  |
| --- | --- | --- | --- |
| Date | A\* M, D | In charge | Change Description |
| 14/Jan | A | TuLQ | Thêm introduction |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

\*A - Added M - Modified D - Deleted

**Table of Contents**

[I. Overview 4](#_Toc110459974)

[1. Introduction 4](#_Toc110459975)

[2. System Functions 5](#_Toc110459976)

[3. Entity Relationship Diagram 6](#_Toc110459977)

[II. Functional Requirements 7](#_Toc110459978)

[1. <<Feature Name 1>> 7](#_Toc110459979)

[a. <<Function Name 1>> 7](#_Toc110459980)

[b. <<Function Name 2>> 7](#_Toc110459981)

[2. <<Feature Name 2>> 7](#_Toc110459982)

# I. Overview

## 1. Introduction

Online Shopping System is a kind of online shop where a seller can sell his/her products or services and customers can buy. Selling online has been easier, faster, or more convenient than with traditional selling.. We have developed this mini project Online Shopping System on Java, JSP, and SQL. In case something bad happens, including system failure or negative feedback,… the administrator will be able to maintain and modify the site. This system supports the below types of user…

>>

## 2. System Functions

## 2.1.1 Visitor

A visitor is someone who visit the website to learn more information about tech devices sold on this website. Use cases:

1: View all products

2: Search for a product

3: View product’s detail

4: View all product by category

5: Read blog

6: View all feedback

## 2.1.2 Customer

A customer is someone who orders a product from the website. Use cases:

1: Login

2: View all products by category

3: Search for a product

4: View product’s detail

5: View and delete cart

6: Edit cart

7: Checkout

8: View and delete order

10: View order history

11: Add feedback

12: Update feedback

13: Read blog

14: Add blog

15: Add and delete address

16: Update address

17: View user profile

18: Update user profile

## 2.1.3 Seller

A seller is someone who has the responsibilities to manage the website’s products and the business important data

1: Create a product

2: Edit a product

3: Delete a product

4: View sale dashboard

5: View user profile

6: Update user profile

## 2.1.4 Marketing

1: View MKT dashboard

2: View blogs list

3: Edit blog

4: Delete blog

## 2.1.5 Administrator

An administrator is the person who has the responsibility to manage users of website. Use cases:

1: Change users’ information

2: Delete an user

2.2. Screen flow

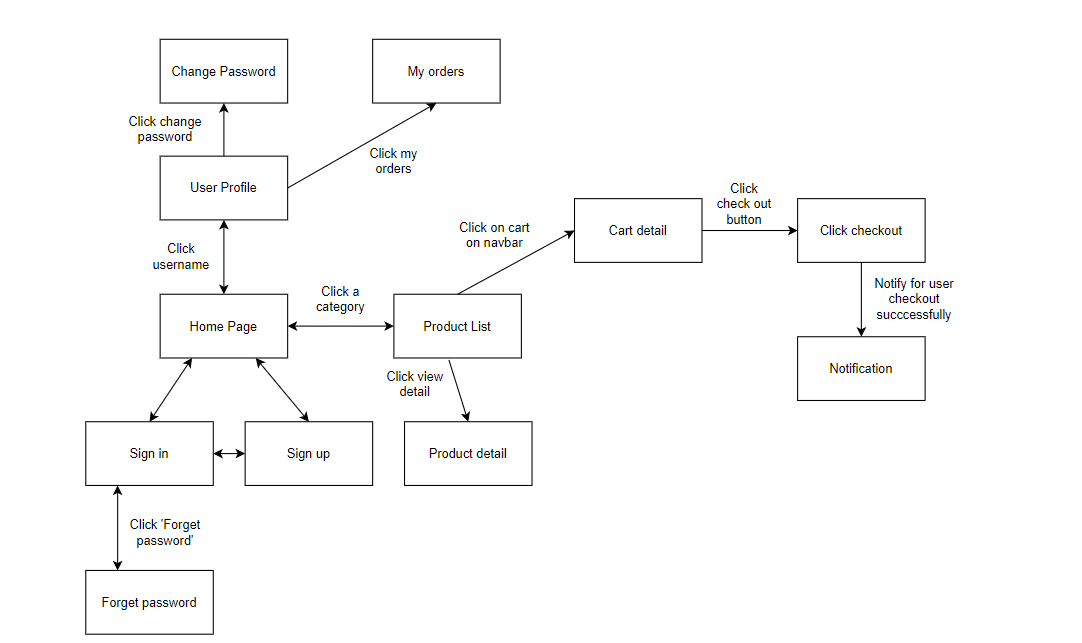
Customer

Figure 1-1: Screen Flow for User

Admin

Figure 1-2: Screen Flow for Admin

Seller

## 

Figure 1-3: Screen Flow for Seller

Marketing

#### 2.3: Screen Details

#### c. User Authorization

| **Screen** | **Guest** | **User** | **Seller** | **Admin** | **Marketing** |
| --- | --- | --- | --- | --- | --- |
| Home Page | x | x |  |  |  |
| Product Details | x | x |  |  |  |
| Cart Details |  | x |  |  |  |
| User Login | x |  |  |  |  |
| User Register | x |  |  |  |  |
| User Authorization | x |  |  |  |  |
| Users List [Admin] |  |  |  | x |  |
| Cart Completion |  | x | x | x |  |
| Reset Password |  | x | x | x |  |
| Change Password |  | x | x | x |  |
| User Profile |  | x | x | x |  |
| My Orders |  | x |  |  |  |
| Order Information |  | x |  |  |  |
| Products List [Admin] |  |  |  | x |  |
| Sale Dashboard [Sale] |  |  | x |  |  |
| Orders List [Sale] |  |  | x |  |  |
| Order Details [Sale] |  |  | x |  |  |
| Feedback | x | x |  |  |  |
| Blogs List | x | x |  |  |  |
| Blog Details | x | x |  |  |  |
| Notification |  | x | x | x |  |
| Error | x | x | x | x |  |
| Admin dashboard |  |  |  | x |  |

In which:

* Guest: Customer who has not yet logged in to the site
* User: Customer who has an account an has logged in
* Seller: People who sell products in the shop
* Marketing: People who manage
* Admin: People who manage the shop

#### d. Non-Screen Functions

*[Provide the descriptions for the non-screen system functions, i.e batch/cron job, service, API, etc.]*

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **System Function** | **Description** |
| 1 | <<Feature Name>> | <<Function Name1>> | <<Function Name1 Description>> |
| 2 | … |  |  |

## 3. Entity Relationship Diagram

*[Provide the entity relationship diagram and the entity descriptions in the table format as below]*



**Entities Description**

|  |  |  |
| --- | --- | --- |
| **#** | **Entity** | **Description** |
| 1 | User |  |
| 2 | Meal |  |
| 3 | Meal Subscription |  |
| 4 | … |  |

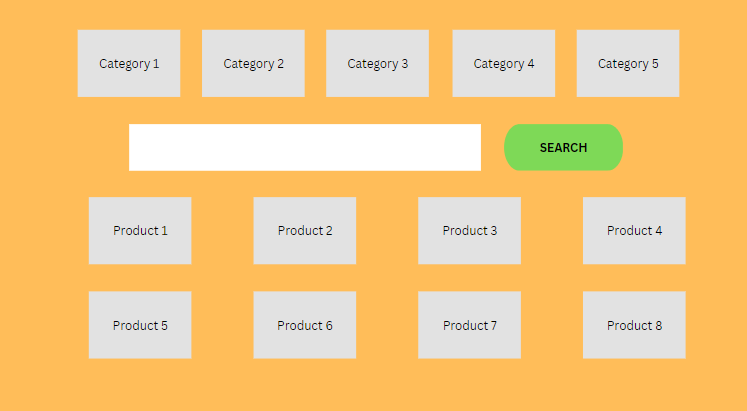
# II. Functional Requirements

## 1. Public

a, Function name: View and search product

- It allow everyone can view and search product in website

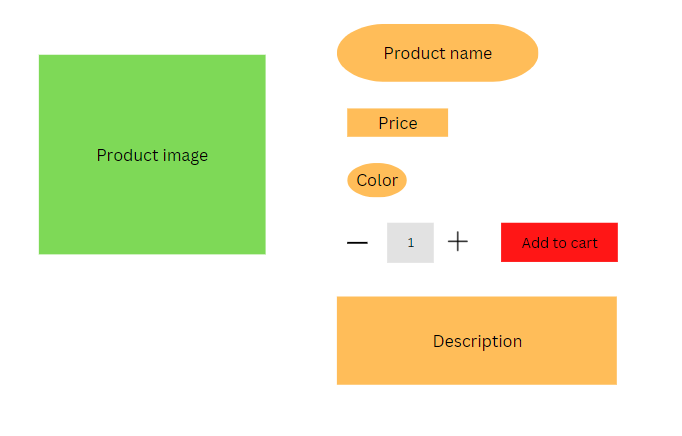
* Function trigger: User accesses the website
* Function description:
  + Role: Guest, User
  + Purpose: Users Search for the Product they want



b, Function name: View Product Details

- It allow everyone can all information about product (product name, color, brand)

* Function trigger: User click on view detail on a Product
* Function description:
  + Role: Guest, User
  + Purpose: User can view all the detail information of a Product



c, Function name: User login

- It allow everyone can sign by username and password

* Function trigger: User input username and password to login an account
* Function description:
  + Role: All role
  + Purpose: User login an account to view,buy product,view blog ,…

Graphical user interface, application

Description automatically generated

d, Function name: User Register

- It allow everyone can sign up

* Function trigger: If user does not have an account , User can input username and password , confirmation password , email , phone , gender to register an account
* Function description:
  + Role: All role
  + Purpose: User register new account.

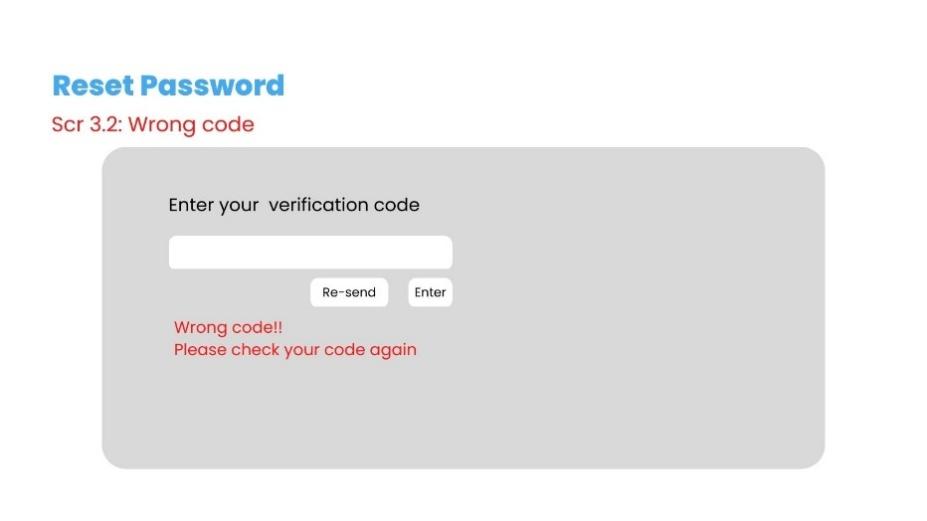
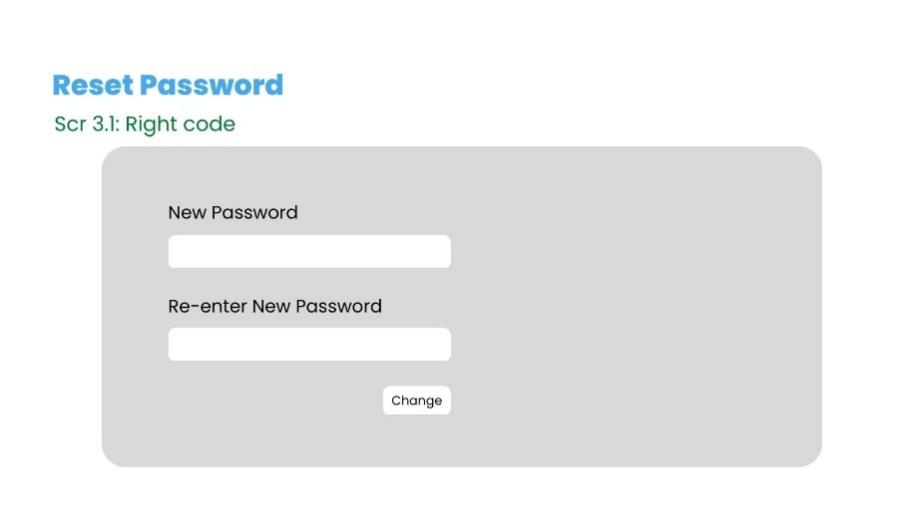
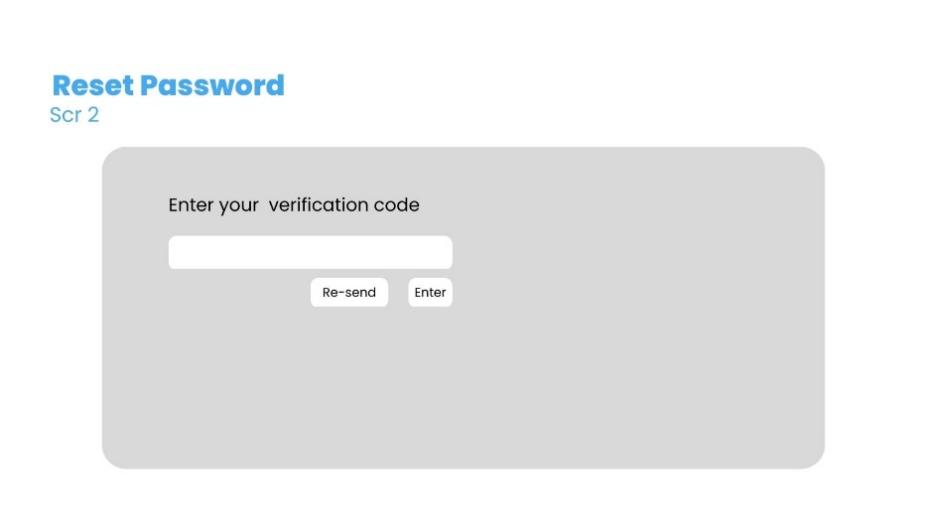
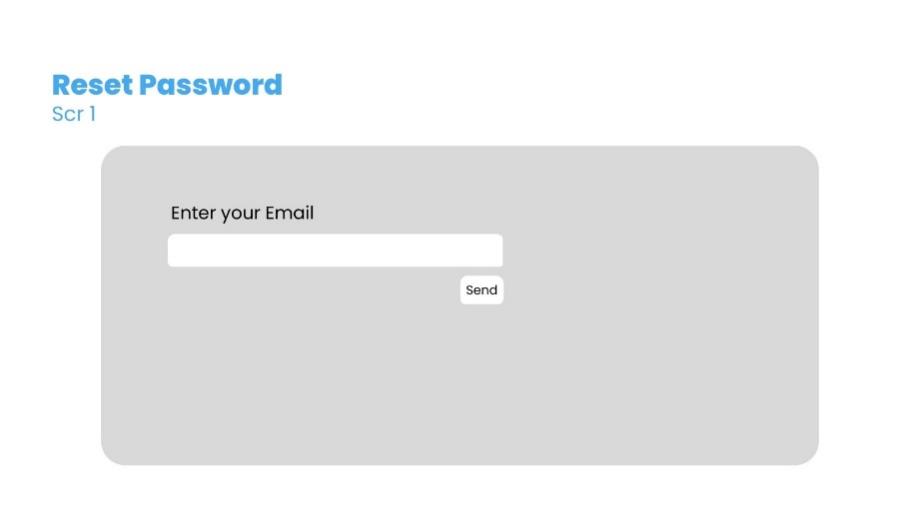
Graphical user interface, application

Description automatically generated

## e, Function name: Reset Password

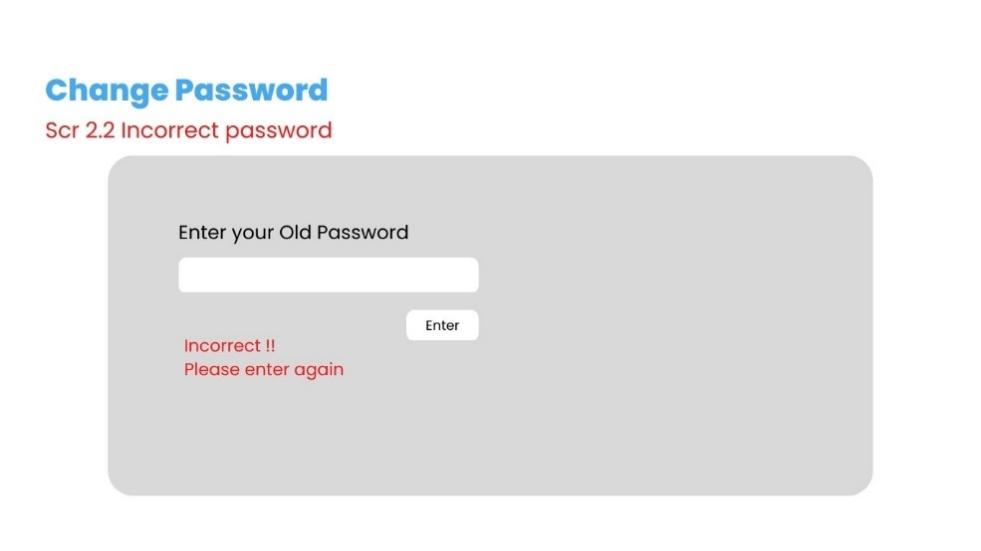
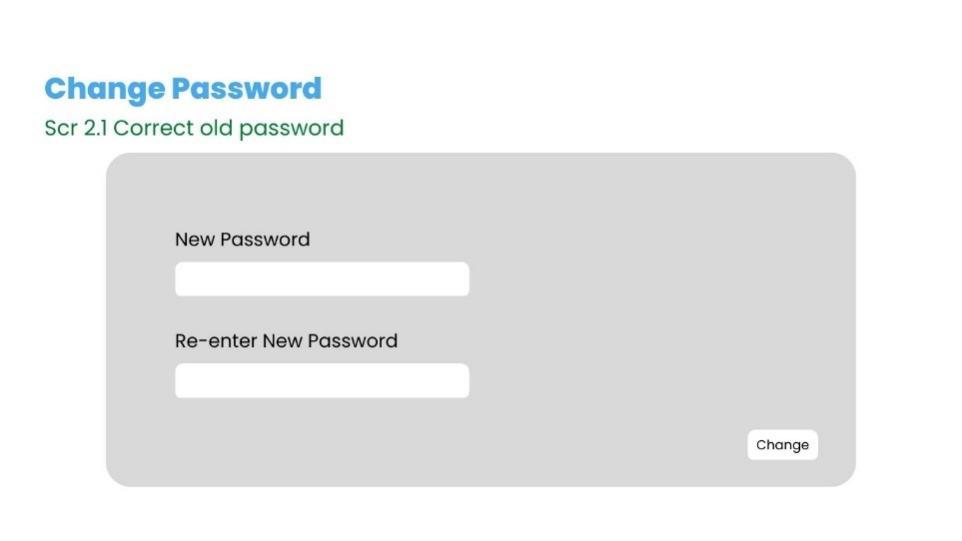
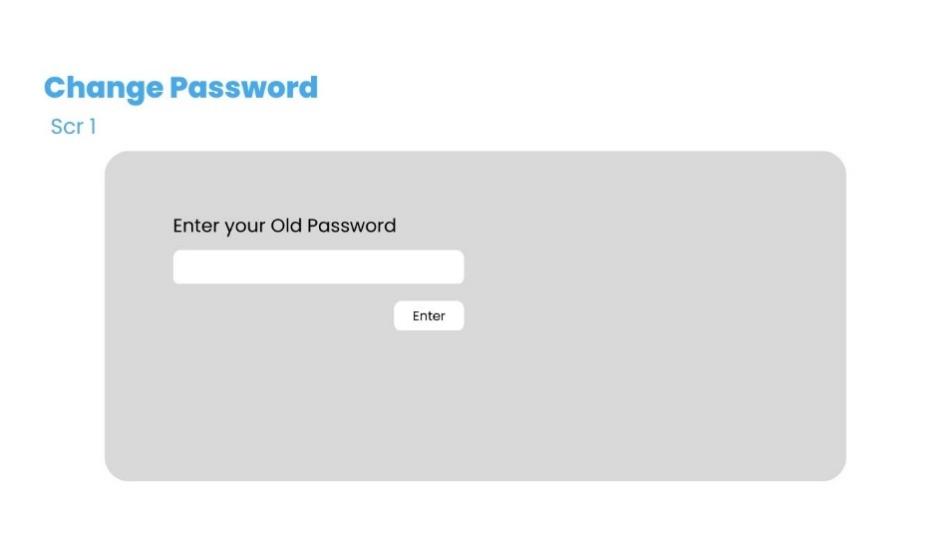
- It allow everyone can reset password though email

* Function trigger: Users access the website and click on “Reset Password” button
* Function description:
  + Role: Customer, Seller, Admin, Marketing
  + Purpose: User can retrieve password when forgotten by confirming information via email



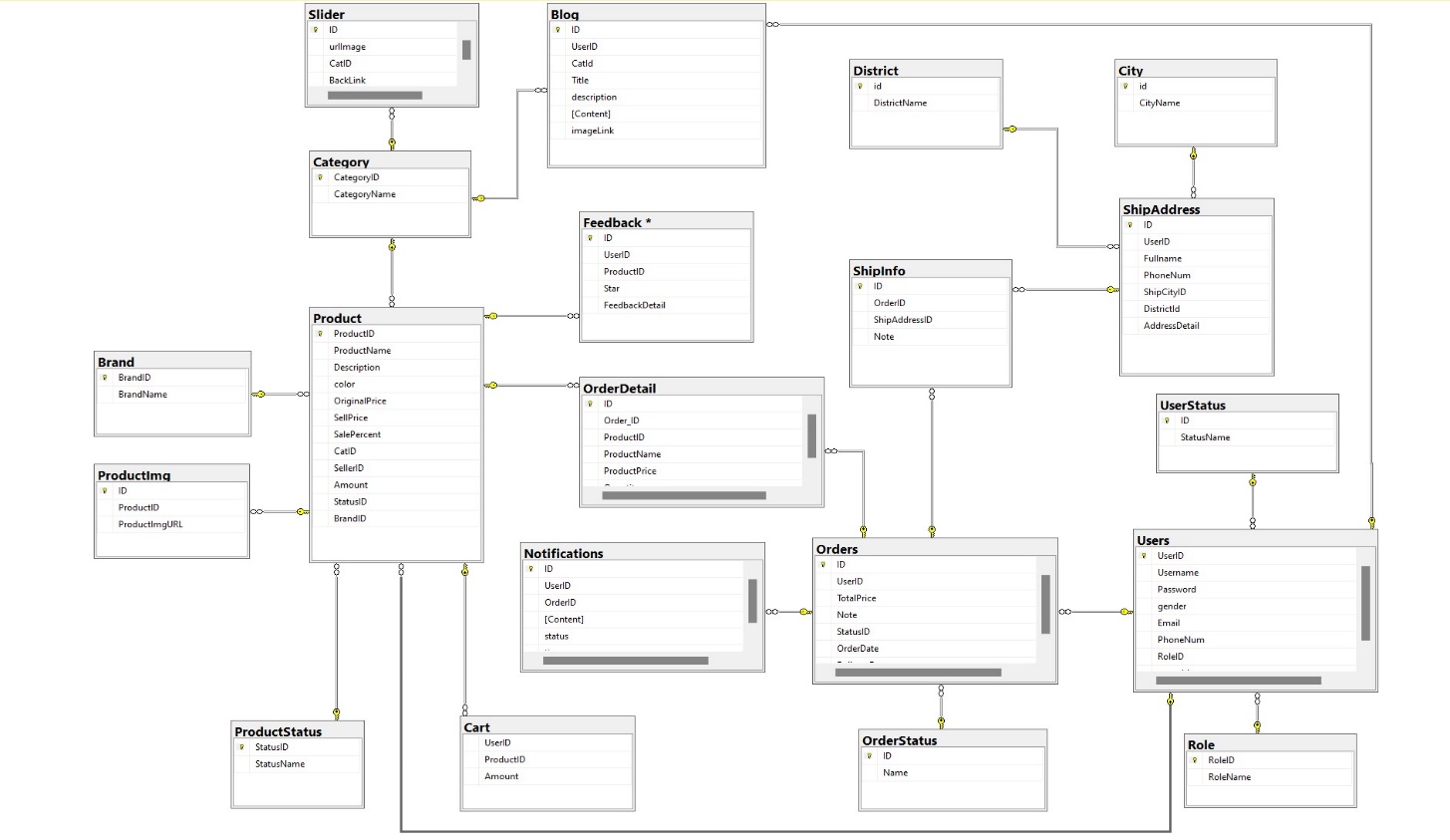
f, Function name: Change Password

* Function trigger: Users access the website and access the profile page and click on “Change Password” button
* Function description:
  + Role: Customer, Seller, Admin
  + Purpose: User can change their password



# III. Database Design

1. **Diagram Schema**



1. **Table Description**

|  |  |  |
| --- | --- | --- |
| **No** | **Table** | **Description** |
| *01* | *UserStatus* | *Account status*  *- Primary keys: ID*  *- Foreign keys:* |
| *02* | *Role* | *Role of a user (Admin, Marketing, Sale, Customer)*  *- Primary keys: RoleID*  *- Foreign keys:* |
| *03* | *Users* | *Users*  *- Primary keys: UserID*  *- Foreign keys: RoleID, statusID* |
| *04* | *Ship City* | *City Address*  *- Primary keys: id*  *- Foreign keys:* |
| *05* | *District* | *District Address*  *-Primary keys: id*  *-Foreign keys:* |
| *06* | *ShipAddress* | *Shipping Address*  *- Primary keys:ID*  *- Foreign keys: UserID, ShipCityID, DistrictId* |
| *07* | *Category* | *Category Data*  *- Primary keys: CategoryID*  *- Foreign keys:* |
| *08* | *ProductStatus* | *Status of a product*  *- Primary keys: StatusID*  *- Foreign keys:* |
| *09* | *Brand* | *Brand data*  *- Primary keys: BrandID*  *- Foreign keys:* |
| *10* | *Product* | *Product Data*  *- Primary keys: ProductID*  *- Foreign keys: CatID, SellerID, StatusID, BrandID* |
| *11* | *ProductImg* | *Image of products*  *- Primary keys: ID*  *- Foreign keys: ProductID* |
| *12* | *Cart* | *Cart info for each user*  *- Primary keys: ID*  *- Foreign keys: UserID, ProductID* |
| *13* | *OrderStatus* | *Status of order*  *- Primary keys: ID*  *- Foreign keys:* |
| *14* | *Orders* | *Order data*  *- Primary keys: ID*  *- Foreign keys: UserID, StatusID* |
| *15* | *OrderDetail* | *Detail of product in order*  *- Primary keys:ID*  *- Foreign keys: OrderID, ProductID* |
| *16* | *ShipInfo* | *Order’s shipping info*  *- Primary keys: ID*  *- Foreign keys: OrderID, ShipAddressID* |
| *17* | *Notifications* | *Notification data*  *- Primary keys: ID*  *- Foreign keys: OrderID, UserID* |
| *18* | *Feedback* | *Feedback of a product order*  *- Primary keys: ID*  *- Foreign keys: UserID, ProductID* |
| *19* | *Blog* | *Blog content + image*  *- Primary keys: ID*  *- Foreign keys: CatID, UserID* |
| *20* | *Slider* | *Image category*  *- Primary keys: ID*  *- Foreign keys: CatID* |