

# **Knowhere Restaurant POS**

Design Document

|  |  |  |  |
| --- | --- | --- | --- |
| **Faculty:** | NGÔ PHƯỚC NGUYÊN | |  |
| **Batch No:** | F2-13-1610-S1 | |  |
| **Group No:** | 04 | |  |
| **Students:** |  | **Full Name** | **Roll No.** |
|  | 1. | NGUYỄN HOÀNG NAM | Student993905 |
|  | 2. | TỪ KHẮC DUY | Student996819 |
|  | 3. | TRẦN QUỐC TUẤN | Student993899 |

**REVIEW 1**

1. Problem Definition:
   1. **Introduction:**

Knowhere Restaurant POS is an online sales management application for Knowhere Company to help companies manage chain restaurants more easily and efficiently.

* 1. **Existing Scenario:**

In order to help Knowhere Company's restaurant management to a new height, apply modern technology to simplify the work of the employees and minimize the possible mistakes in the work. This increases the productivity of the restaurant and helps to manage the restaurant chain more easily and efficiently.

1. Customer Requirements Specifications:

**2.1. Sales Staff:**

* Sale Staff can help customers book seats and order food.
* Sale Staff can manage the bills of each table and help customers pay bills
* Sale Staff can manage the store’s menu.

**2.2. Waiter:**

* Waiter can help customers book seats and order food.

**2.3. Kitchen:**

* Kitchen can view the order menu of each table, order confirmation and print the receipt.

**2.4. Administrator:**

* Administrator can manage employees and customer information
* Administrator can manage store
* Administrator can manage the products of the store
* Administrator can view and manage order information
* Administrator can view customer and product reports

1. Function Requirement Specification
   1. **Sales Staff:**

* Sales Staff can update his/her profile
* Sales Staff can view all customer
* Sales Staff can add new custemer
* Sales Staff can edit customer
* Sales Staff can delete customer
* Sales Staff can help customers book seats
* Sales Staff can help customers order food
* Sales Staff can help customers change dishes
* Sales Staff can view information of all oder
* Sales Staff can issue invoices to customers
* Sales Staff can add new product
* Sales Staff can view the store's product information
* Sales Staff can edit the store's product information
* Sales Staff can delete the store's product information
* Sales Staff can view and print store menus
* Sales Staff can set a product for sale or not for sale.
* Sales Staff can add new table and new zone
* Sales Staff can edit table and zone
* Sales Staff can delete table and zone
  1. **Waiter:**
* Waiter can update his/her profile
* Waiter can help customers book seats
* Waiter can help customers order food
* Waiter can help customers change dishes
  1. **Kitchen:**
* Kitchen can update his/her profile
* Kitchen can view the order of each table
* Kitchen can confirm the order and print the receipt
  1. **Administrator:**
* Administrator can add new account.
* Administrator can delete account.
* Administrator can view all customer
* Administrator can add new custemer
* Administrator can edit customer
* Administrator can delete customer
* Administrator can view all order.
* Administrator can add new table and new zone
* Administrator can edit table and zone
* Administrator can delete table and zone
* Administrator can add new product
* Administrator can view the store's product information
* Administrator can edit the store's product information
* Administrator can delete the store's product information
* Administrator can set a product for sale or not for sale.
* Administrator can view and print store menus
* Administrator can view reports (customers, product number) (monthly stats in the current or other years) ... etc

1. System Requirement:
2. **Hardware Requirements:**
   * AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support, Intel Pentium IV with EM64T support
   * Processor Speed minimum 1.4hz
   * 1G of RAM or higher. (4 GB **Recommended**)
   * Hard disk requirement: minimum 50GB
   * Windows 7 ultimate, or window server 2008 R2 or higher
3. **Software Requirements:**
   * SQL Server 2012 x64 bit
   * Netbean 7.4 or higher
   * JavaEE 7 SDK
   * Java Development Kit (JDK) 1.7 or higher (Recommended jdk 1.8)
   * Glassfish Server 4.0
   * Type 4-2.0 JDBC driver for SQL Server 2012 RC4
   * Internet Explorer 10 or higher

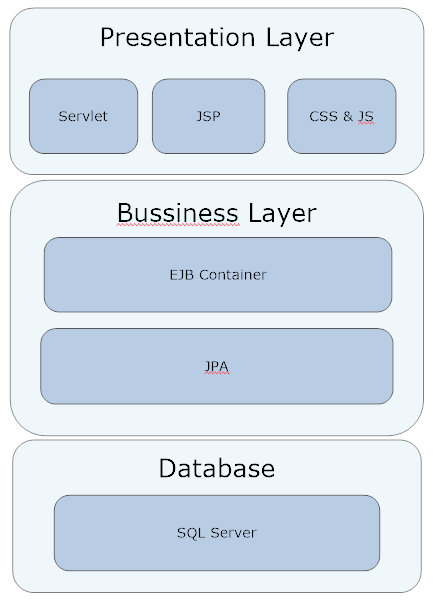
## Task Sheet 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project Ref. No.** | | **Project Name:**  **Knowhere Restaurant POS** | **Date of Preparation of Activity Plan** | | |
| No. | Task | Description | Start  Date | Complete  Date | Team member |
| 01 | **Review 1** | Problem statement | **28 / 06 / 2018** | **07 / 07 / 2018** |  |
| 02 | Function requirement specification |

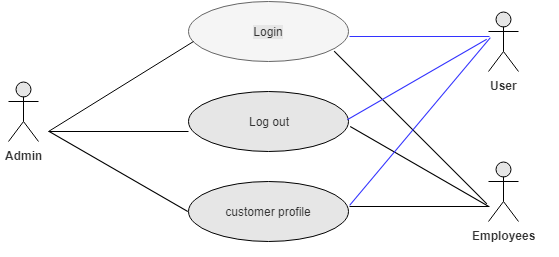
**Signature of Team leader** **Signature of Instructor**

REVIEW 2

1. Architecture and the design of the project:



1. User Case and Sequence Diagram:
   1. Customer:



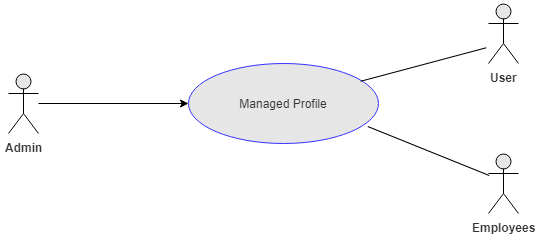
* 1. Login:

|  |  |  |
| --- | --- | --- |
| **Author** | Trần Quốc Tuấn | |
| **Use Case Name** | Login | |
| **Actors** | Customer | |
| **Description** | Customer login as User, or Employees, or Admin | |
| **Requirements** | Actor provides username and password | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: User logged in to website.  Fail: try again login page | |
| **Basic flow** | Actor Actions:  1. Actor go to Login page of website.  3. Actor inputs username and Password, then click the Sign In button. | System Responses:  2. System redirects to Login form and displays Home page on the left with the following control:  -“username” text field  - ‘Password’ text field  - ‘Sign in’ button  4. System checks the information  [Exception 1]  5. System redirects to Actor’s Home page. |
| **Alternative flow** | Actor Actions:  [Alternative 2]  1. Actor click ‘OK’ button, inputs Username and Password, then click the Login button.  [Alternative 1]  1. Actor click ‘Back to Home’ button. | System Responses:  2. System checks the information  [Exception 1]  3. System redirects to Actor’s Home page.  2. System returns to Home page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor input invalid username and password. | System Responses:  - System redirect to Login page with the following controls:  - ‘Username text field  - ‘Password’ text field  - ‘Sign In’ button.  - ‘Back to Login button.  System show a message box: “Invalid username and password”.[Alternative 2] |

* 1. Logout

|  |  |  |
| --- | --- | --- |
| **Author** | Trần Quốc Tuấn | |
| **Use Case Name** | Logout | |
| **Actors** | User, Employees and Admin | |
| **Description** | User, Employees and Admin wants to log out of the System. | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has logged in. | |
| **Post-conditions** | Success: Actor Logged out. | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Log out’ button on the Actor’s Login page. | System Responses:  2. System redirect to Logout page, and then redirect to Login page. |
| **Alternative flow** |  |  |
| **Exception** |  |  |

* 1. Update profile Customer



|  |  |  |
| --- | --- | --- |
| **Author** | Trần Quốc Tuấn | |
| **Use Case Name** | Update Profile Customer | |
| **Actors** | User, Employees and Admin | |
| **Description** | Customer wants to Update his/her profile’s information. | |
| **Requirements** | Actor provides profile information | |
| **Pre-conditions** | Employees update profile for customer. | |
| **Post-conditions** | Success: Profile’s information of this user has been updated.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘People chose Customer button in Actor’s Home page menu.  3. Actor updates information and click ‘Save’ button. [Alternative 1] | System Responses:  2. System redirects to Update Profile Customer page and displays edit profile customer form with the following control :  - ‘Customer\_name’ text field (read only)  - ‘Phone’ text field  - ‘Email’ text field  -‘Discount’ text field  - ‘Save’ button  - ‘Back to Home’ button  4. System checks the information  [Exception 1]  5. System update the account into database and then redirects to Actor Home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back ’ button. | System Responses:  2. System returns to Customer. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Customer\_name’ text field blank.  2. Actor lefts ‘Phone’ text field blank.  3. Actor lefts ‘ Email’ text field blank.  4. Actor lefts ‘Discount’ text field blank. | System Responses:  System displays message to Actor :  - ‘Name cannot be blank’’.  - ‘Phone cannot be blank.  - ‘Email cannot be blank’.  - ‘Discount’ cannot be blank. |

* 1. Add new Employees

|  |  |  |
| --- | --- | --- |
| **Author** | Trần Quốc Tuấn | |
| **Use Case Name** | Add New Employees | |
| **Actors** | Administrator | |
| **Description** | Add New Employees member. | |
| **Requirements** | Admin provides Employees’s information | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Employees creates a account in this website  Fail: return Add User page | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Settings chose User’ link in Home page  5. Actor inputs register’s information, select ‘Seller’ radio button and click ‘Add User’ button. [Alternative 1] | System Responses:  2. System redirects to Register page and displays Registration form with the following control :  - ‘employees Name’ text field  - ‘Username’ text field  - ‘Password’ text field  - ‘Confirm Password’ text field  - ‘Role’ checkbox group  - ‘Email’ text field  - ‘Phone’ text field  -‘avatar’ chose files  - ‘Register’ button  - ‘Back to Home’ button  6. System checks the information  [Exception 1]  7. System inserts the account into database, show message: ‘Your account has been created!’ and then redirects to Login page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back ‘ button. | System Responses:  2. System returns to ‘User’. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Employees name’ text field blank or input an invalid employees name.  2.Actor lefts ’Username’ text field blank.  3. Username has already exists.  4. Actor lefts ‘Password’ text field blank.  5. Actor lefts ‘Confirm Password’ text field blank.  5. Password and confirm password don’t match.  6. Actor lefts ‘Email’ text field blank.  7. Email has already exists.  8. Actor lefts ‘Phone’ text field blank. | System Responses:  System displays message to Actor :  - ‘Employees name is invalid’.  - ‘Username cannot be blank’’.  - ‘This Username has already exist!’  - ‘Password cannot be blank’’.  - ‘Confirm Password can’t be.  - ‘Confirm password not match password.’  - ‘Email cannot be blank’.  - ‘This Email has already exist!’.  - ‘Phone is invalid’. |

* 1. Add new Product

|  |  |  |
| --- | --- | --- |
| **Author** | Tu Khac Duy | |
| **Use Case Name** | Insert new Product | |
| **Actors** | Administrator, Sales Staff | |
| **Description** | Sales Staff or Administrator add new product. | |
| **Requirements** | Admin or Sales Staff provide Product Infomation | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Create new Product for website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Add New Product’ link in Product Page  5. Actor inputs product’s information. | System Responses:  2. System redirects to Register page and displays Registration form with the following control:  - ‘Product Name’ text field  - ‘Categories’ combobox  -‘Price’ text field  -‘Discount’ text field  - ‘Choose Image’ input file  -‘Product description’ textarea  - ‘Color display’ radio button  - ‘Submit’ button  6. System checks the information  [Exception 1]  7. System inserts the account into database, show message: ‘This product has been created!’ and then redirects to Create Page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back’ button. | System Responses:  2. System returns to Product page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘ Product Name’ text field blank or input an invalid Product Name.  2. Product has already exists. | System Responses:  System displays message to Actor :  - ‘Product Name is invalid’.  - ‘This Product has already exist!’. |

* 1. Update Product

|  |  |  |
| --- | --- | --- |
| **Author** | Tu Khac Duy | |
| **Use Case Name** | Update Product | |
| **Actors** | Administrator, Sales Staff | |
| **Description** | Sales Staff or Administrator update product. | |
| **Requirements** | Admin or Sales Staff provide Product Infomation | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Update Product for website  Fail: Refill information | |
| **Basic flow** | Actor actions:  1. From ‘View List of Product’ page, actor clicks on button ‘Edit Product’ hyperlink which wants to modify.  3. Actor:   * Select Directory Category in ‘Directory Category’ combobox. [Alternative 1] * Inputs Directory Name into ‘Directory Name’ textfield. * Inputs Description into ‘Description’ textfield. * Inputs Directory Type into ‘Directory Type’ textfield. * Clicks on to ‘Browse’ button to select path of Directory Image, this path is displayed in ‘Directory Image Path’ textfield. * Clicks on to ‘Save’ button to finish adding. [Alternative 2] | System response:  2. System redirects to ‘Modify Directory’ page and display form with the following controls:  - ‘Product Name’ text field  - ‘Categories’ combobox  -‘Price’ text field  -‘Discount’ text field  - ‘Choose Image’ input file  -‘Product description’ textarea  - ‘Color display’ radio button  - ‘Submit’ button  4. System checks input information.  [Exception 1]  5. System updates information into the database.[Alternative 3] |
| **Alternative flow** | [Alternative 1]  1. Actor clicks on to ‘Other Directory Category’ button.  3. Actor inputs Directory Category Name into ‘Directory Category Name’ text field.  [Alternative 2]  1. Actor clicks on to ‘Back’ button.  [Alternative 3]  1. Actor has inputted Directory Category Name into ‘Directory Category Name’ text field. | 2. The ‘Directory Category Name’ text field appears.  2. System redirects to ‘View List of Directories’ page.  2. System updates Directory and adds new Directory Category into the database. |
| **Exception** | Actor Actions:  [Exception 1] | System Responses:  System displays message to Actor :  - ‘Product Name is invalid’. |

* 1. Add new category

|  |  |  |
| --- | --- | --- |
| **Author** | Tu Khac Duy | |
| **Use Case Name** | Insert new Category | |
| **Actors** | Administrator, Sales Staff | |
| **Description** | Sales Staff or Administrator add new category. | |
| **Requirements** | Admin or Sales Staff provide Categories Infomation | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Create new Category for website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Add New Category’ link in Categories Page  5. Actor inputs categories’s information. | System Responses:  2. System redirects to Register page and displays Registration form with the following control:  - ‘Category Name’ text field  - ‘Submit’ button  6. System checks the information  [Exception 1] |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Back’ button. | System Responses:  2. System returns to Categories page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘ Category Name’ text field blank or input an invalid Product Name.  2. Category has already exists. | System Responses:  System displays message to Actor:  - ‘Category Name is invalid’.  - ‘This Category has already exist!’. |

* 1. Add new Zone

|  |  |  |
| --- | --- | --- |
| **Author** | Tu Khac Duy | |
| **Use Case Name** | Insert new Zone | |
| **Actors** | Administrator | |
| **Description** | Administrator add new Zone. | |
| **Requirements** | Admin provide Zones Infomation | |
| **Pre-conditions** | Actor has already logged into the system as Administrator | |
| **Post-conditions** | Success: Create new Zone for website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Add New Zone’ link in Table Manager Page  5. Actor inputs categories’s information. | System Responses:  2. System redirects to Register page and displays Registration form with the following control:  - ‘Zone Name’ text field  - ‘Submit’ button  6. System checks the information  [Exception 1] |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘Close’ button. | System Responses:  2. System returns to Table Manager page. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘ Zone Name’ text field blank or input an invalid Zone Name.  2. Zones has already exists. | System Responses:  System displays message to Actor:  - ‘Zone Name is invalid’.  - ‘This Zones has already exist!’. |

* 1. Delete Zone

|  |  |  |
| --- | --- | --- |
| **Author** | Tu Khac Duy | |
| **Use Case Name** | Delete Zone | |
| **Actors** | Administrator | |
| **Description** | Administrator delete Zone. | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has already logged into the system as Administrator | |
| **Post-conditions** | Success: Delete Zone for website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. From View List of Zone, actor clicks on to the “Delete” CommandButton which actor wants to delete  3. Actor click “Yes” button.  [Alternative 1] | System Responses:  2. A confirms message box appears: “Do you want to delete this Zone?”  “All the tables related to this zone will be deleted”  4. System update the Zone in the database and do not display that zone on the website.  System update table of that Zone in database and do not display that table on the website |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click ‘No’ button. | System Responses:  2. System returns to Table Manager page. |
| **Exception** |  |  |

* 1. Update status Table:

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Hoang Nam | |
| **Use Case Name** | Update status Table | |
| **Actors** | Administrator, Employees | |
| **Description** | Update status is deliver status | |
| **Requirements** |  | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Orders status is delivered | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Stores List’ link in Admin Home page | System Responses:   1. System redirects to Stores List page and displays Tables List :   - Search for Tables for to be set.  - Click Deliver. |
| **Alternative flow** |  |  |
| **Exception** |  |  |

* 1. Cancel Order

|  |  |  |
| --- | --- | --- |
| **Author** | Nguyen Hoang Nam | |
| **Use Case Name** | Cancel Orders | |
| **Actors** | Administrator, Employees | |
| **Description** | Employees or Administrator Cancel Order. | |
| **Requirements** | Admin or Employees Cancel Order | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Cancel Order Tables | |
| **Basic flow** | Actor Actions:  1. From ‘Tables List’ page, actor clicks on to the ‘Cancel’ hyperlink which wants to cancel. | System Responses:  2. System redirects to Purchase Order Detail page and displays Order Detail with the following control:  -Order Items.  3. System response and show message ‘Are You Sure Cancel!!!!!’. |
| **Alternative flow** |  |  |
| **Exception** |  | . |

1. Table Definition
   1. Employees

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| PK | Employee\_id | Varchar(10) | No |  |  | Primary key of Employees tables |
|  | Employees\_name | Nvarchar(50) | No |  |  | Name of employees |
|  | Username | Varchar(50) | No |  |  | User of employee |
|  | Password | Varchar(50) | No |  |  | Password of user |
|  | Phone | Varchar(50) | No |  |  | Phone of employee |
|  | Email | Varchar(50) | Yes |  |  | email of employee |
|  | Avatar\_link | Varchar(max) | No |  |  | Avatar Profile of employee |
| FK | Role\_id | Varchar(10) | No |  | Roles | Foreign key with role\_id |
|  | Deleted | Int | No | 0 |  | Delete of employee |

* 1. Customer

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| Pk | Cus\_id | Varchar(10) | No |  |  | Primary key of customer tables |
|  | Cus\_name | Nvarchar(50) | No |  |  | Name customer |
|  | Phone | Varchar(15) | Yes |  |  | Phone of customer |
|  | Email | Varchar(50) | Yes |  |  | Email of customer |
|  | discount | Float | No |  |  | discount for customer |
|  | deleted | int | No | 0 |  | Delete customer |

* 1. Zones

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Tye** | **Allow Null** | **Defautlt value** | **Table-field-Ref** | **Description** |
| Pk | Zone\_id | Varchar(10) | No |  |  | Primary key Zones table |
|  | Zone\_name | Nvarchar(500 | No |  |  | Zone name |
|  | Deleted | Int | No | 0 |  | Delete of Zones |
| Fk | Sto\_id | Varchar(10) | No |  | Stores | Foreign key to stores |

* 1. Stores

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| PK | Sto\_id | Varchar(10) | No |  |  | Primary key Stores table |
|  | Sto\_name | Nvarchar(50) | No |  |  | Store name |
|  | Address | Varchar(100) | No |  |  | Address stores |
|  | Phonesto | Varchar(15) | No |  |  | Phone store |
|  | Deleted | Int | No | 0 |  | Delete of Store |

* 1. Product

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| PK | Pro\_id | Varchar(10) | No |  |  | Primary key of product tables |
|  | Pro\_name | Nvarchar(50) | No |  |  | Product name |
| FK | Cat\_id | Varchar(10) | No |  | catogories | Foreign key with catogories |
|  | Price | money | No |  |  | price of product |
|  | Imagelink | nvarchar(max) | Yes |  |  | Avatar Profile of product |
|  | Discount | Float | No |  |  | Discount of product |
|  | Deleted | int | No | 0 |  | Delete product |
|  | Descirption | Nvarchar(200) | Yes |  |  | Description of product |
|  | color | Varchar(20) | Yes |  |  | Color of product |

* 1. Catogories

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| Pk | Catogorie\_id | Varchar(10) | No |  |  | Primary key of categories table |
|  | Catogories\_name | Nvarchar(50) | No |  |  | Categories name |
|  | CreateDay | Datetime | No | (GETDATE()) |  | The Input date of new categories |
|  | deleted | int | No | 0 |  | Delete of categories |

* 1. Roles

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| PK | Role\_id | Varchar(10) | No |  |  | Primary key of Roles table |
|  | Role\_name | Nvarchar(50) | No |  |  | Name of role |
|  | Deleted | Int | No | 0 |  | Delete of roles |

* 1. Orders

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| PK | Order\_id | Varchar(10) | No |  |  | Primary key of orders table |
|  | Ordertime | Datetime | No | (getdate()) |  | Order time |
|  | Order\_tax | Float | No |  |  | Order tax |
|  | Subtotal | Money | No |  |  | Subtotal |
|  | Paymethod | Nvarchar(50) | No |  |  | Total |
|  | Deleted | Int | No | 0 |  | Delete of orders |
|  | Discount | Float | No |  |  | Discount of order |
|  | Total | money | No |  |  | Total |
| FK | Cus\_id | Varchar(10) | No |  | Customer | Foreign key with customer |
| FK | Emp\_id | Varchar(10) | No |  | Employees | Foreign key with employees |
| FK | Tab\_id | Varchar(10) | No |  | Table | Foreign key with tables |

* 1. Orders detail

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| FK | Order\_id | Varchar(10) | No |  |  | Foreign key of order |
| FK | Pro\_id | Varchar(10) | No |  | Product | Foreign key of product |
|  | Quantity | Int | No |  |  | Quantity of product |
|  | Price | Money | No |  |  | Price of product |
|  | Discount | Float | No |  |  | Discount of product |
|  | Option | Nvarchar(150) | Yes |  |  | Option product |

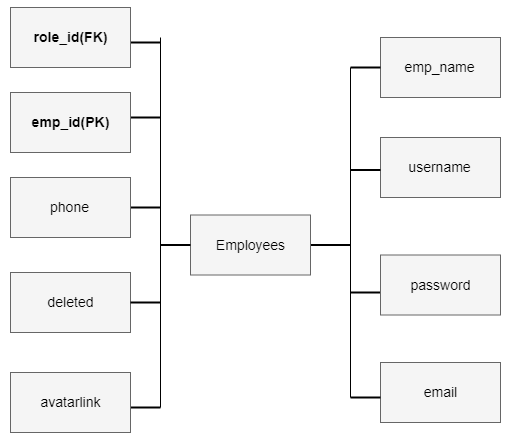
* 1. Table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| PK | Tab\_id | Varchar(10) | No |  |  | Primary key of tables table |
|  | Tab\_name | Nvarchar(50) | No |  |  | Name table |
|  | Status | Bit | Yes |  |  | Status table |
|  | Deleted | Int | No | 0 |  | Delete tables |
| FK | Zone\_id | Varchar(10) | No |  | Zones | Foreign key with Zones |

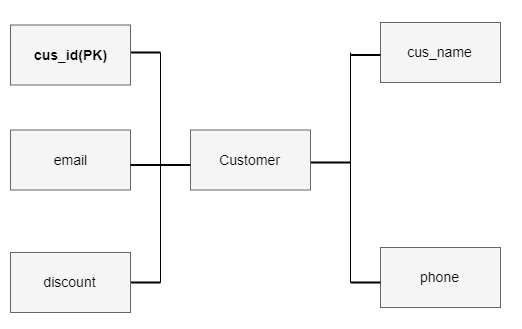
* 1. Hide Product

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Key | Field Name | Data Tye | Allow Null | Defautlt value | Table-field-Ref | Description |
| FK | Sto\_id | Varchar(10) | No |  | Stores | Foreign key of stores |
| FK | Pro\_id | Varchar(10) | No |  | Product | Foreign key of product |
|  | Hide | Int | No |  |  | Hide of Product |

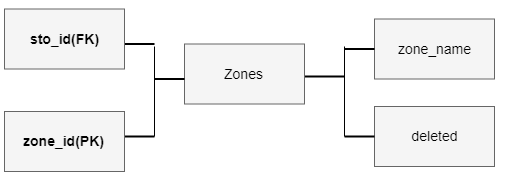
1. Entity Description:
   1. Employees



* 1. Customer



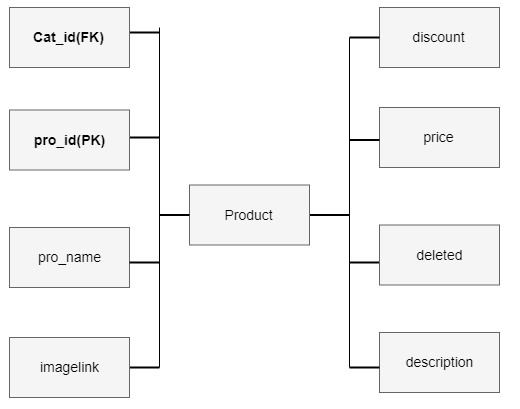
* 1. Zones



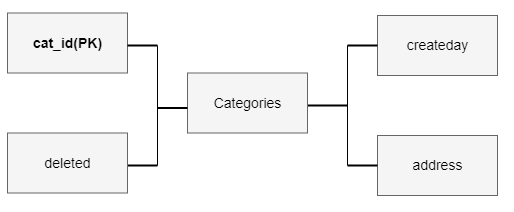
* 1. Stores



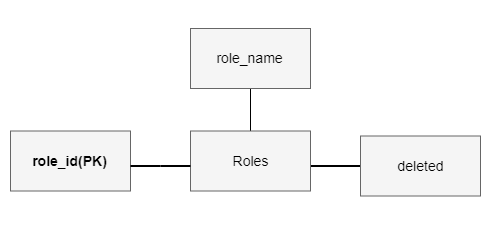
* 1. Product



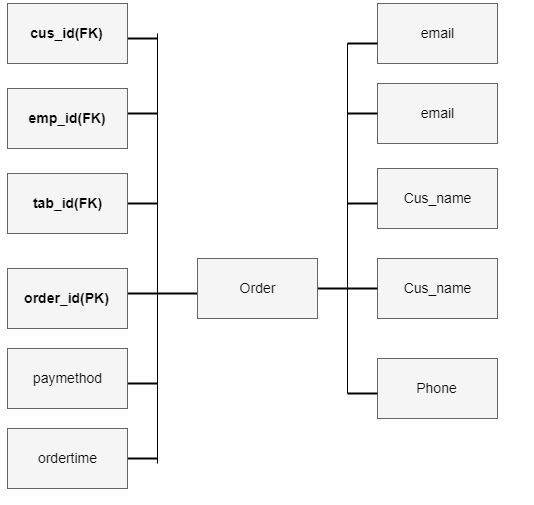
* 1. Catogories



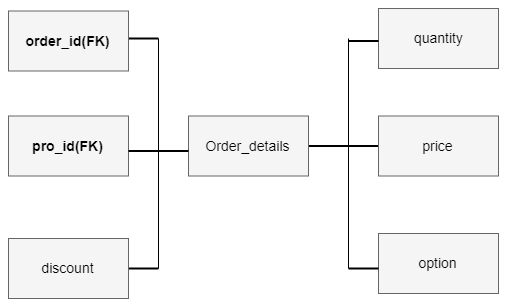
* 1. Roles



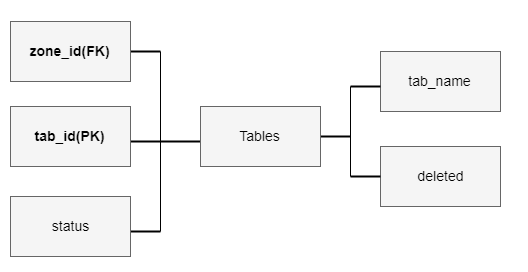
* 1. Orders



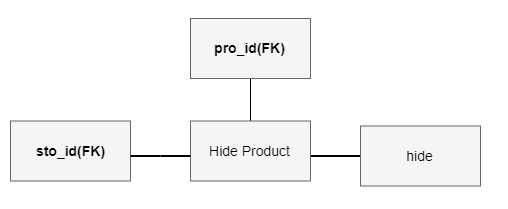
* 1. Orders detail



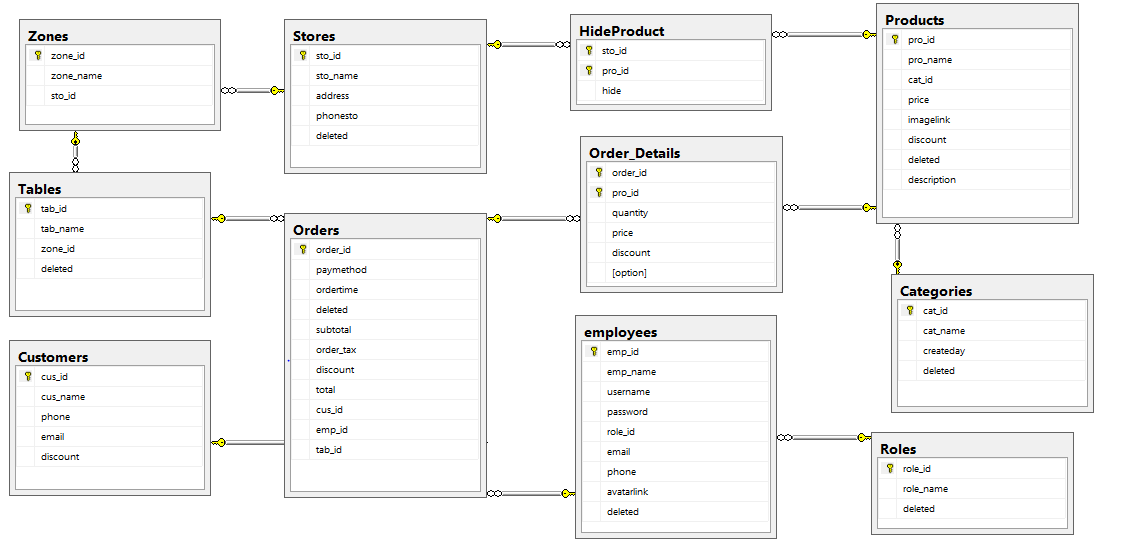
* 1. Tables



* 1. Hide Product



1. Tables Relationship Diagram:

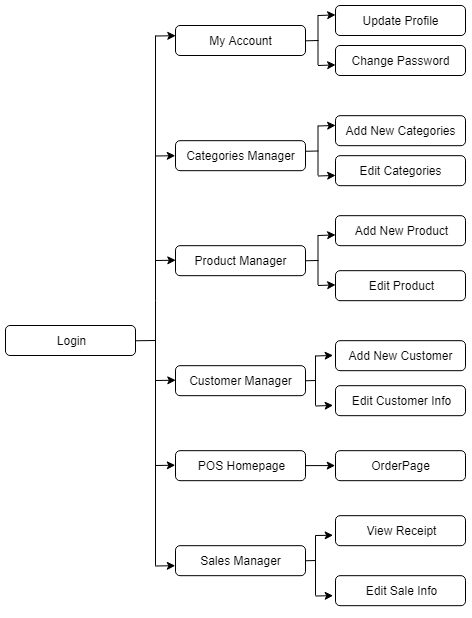
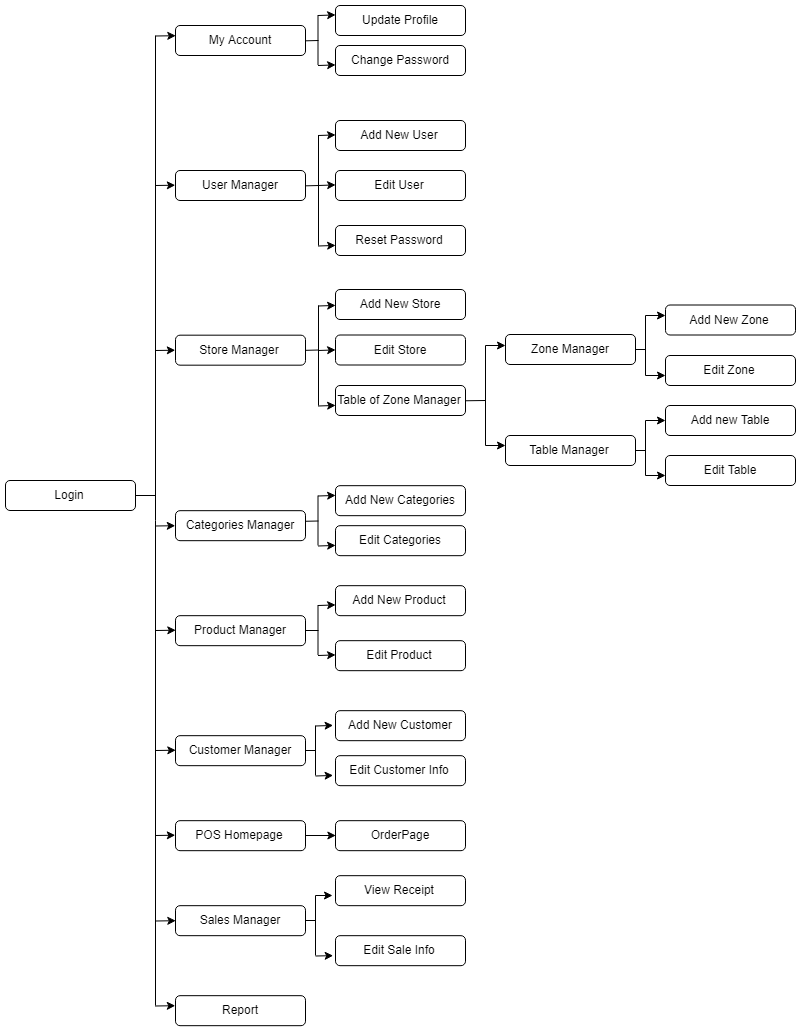


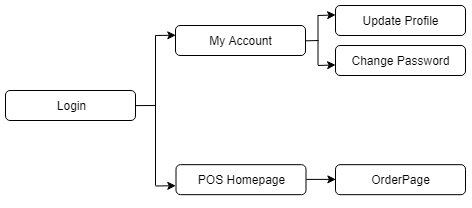
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project Ref. No.** | | **Project Name:**  **Unify Fashion Shop** | **Date of Preparation of Activity Plan** | | |
| No. | Task | Description | Start  Date | Complete  Date | Team member |
| 01 | **Review 2** | User Case and Sequence Diagram | **07 / 07/ 2018** | **27/ 07/ 2018** | All Team Member |
| 02 | Entity Definition |
| 03 | Tables Relationship Diagram |

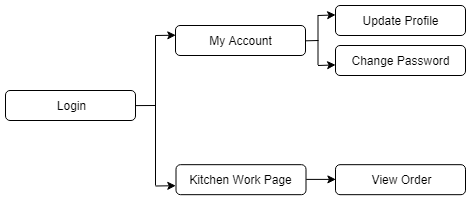
**Signature of Team leader** **Signature of Instructor**

**REVIEW 3**

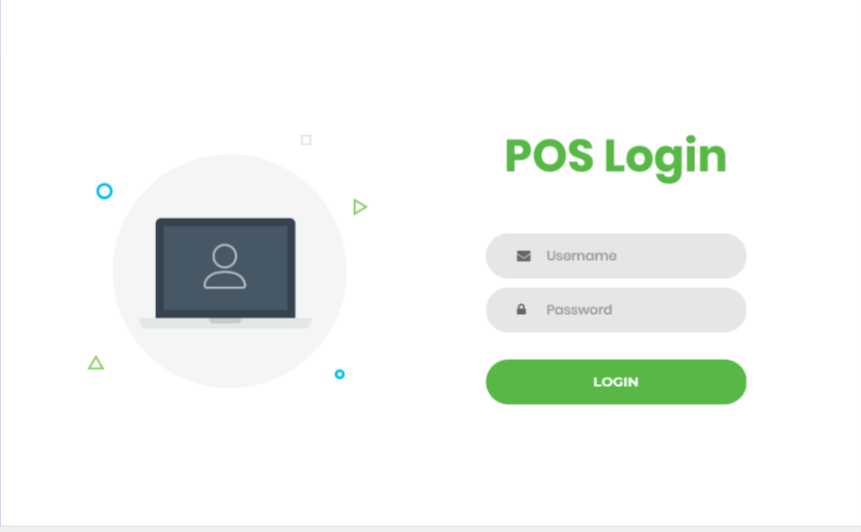
**SITE MAP**

1. **Sales Staff Site**
2. **Admin Site**
3. **Waiter Site**

****

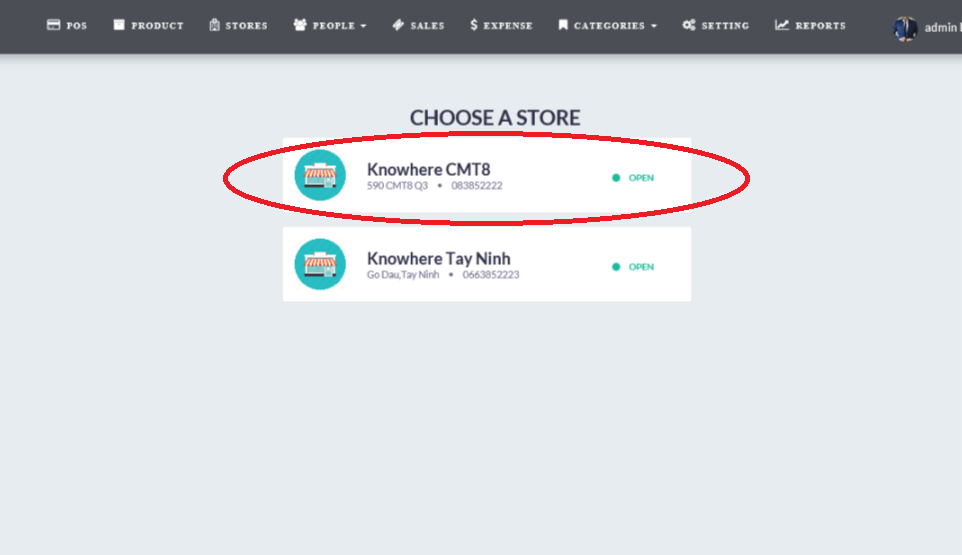
1. **Kitchen Site**

**GUI Design**

1. **Login page:**

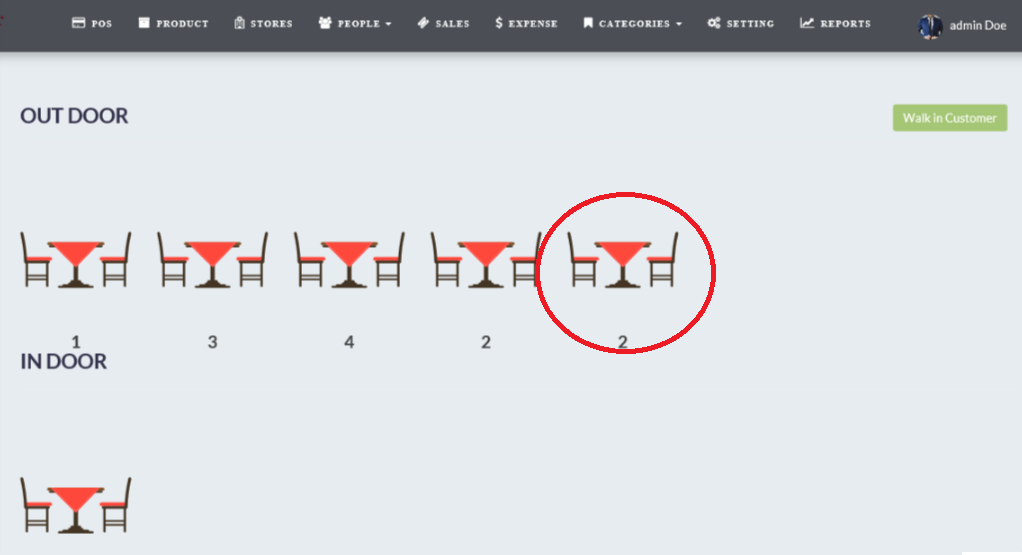
*Login Page*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | username | Text Field | Not blank |  | Username | Enable |
| 2 | password | Text Field | Not blank |  | Password | Enable |
| 3 | btnLogin | Button |  | Action | Check Username and password and if they are correct, redirect to POS homepage | Enable |

1. **POS Home page:** 

*Choose store page*

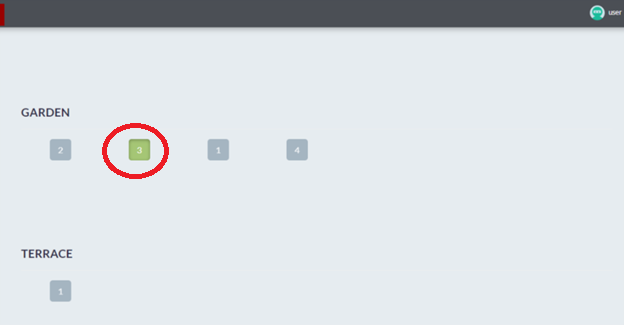
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | storeName | CommandLink |  | Action | Open POS Table Page | Enable |



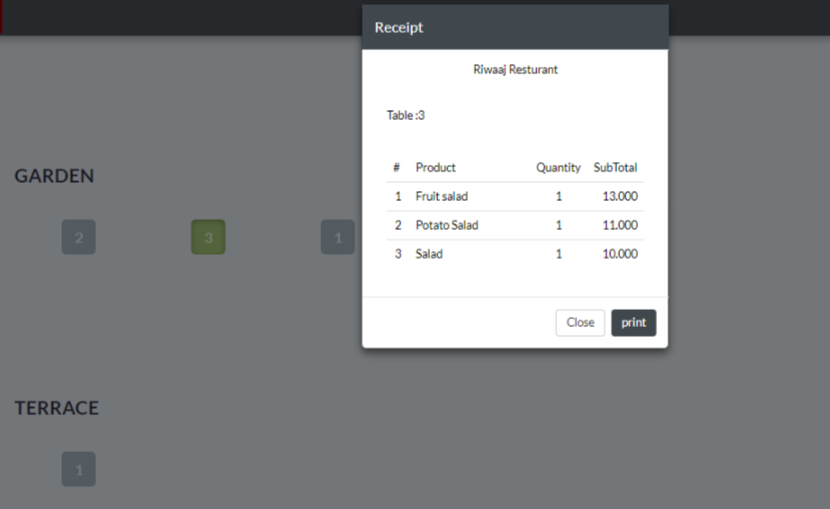
POS Table Page

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | Table Name | CommandLink |  | Action | Open Order Page | Enable |

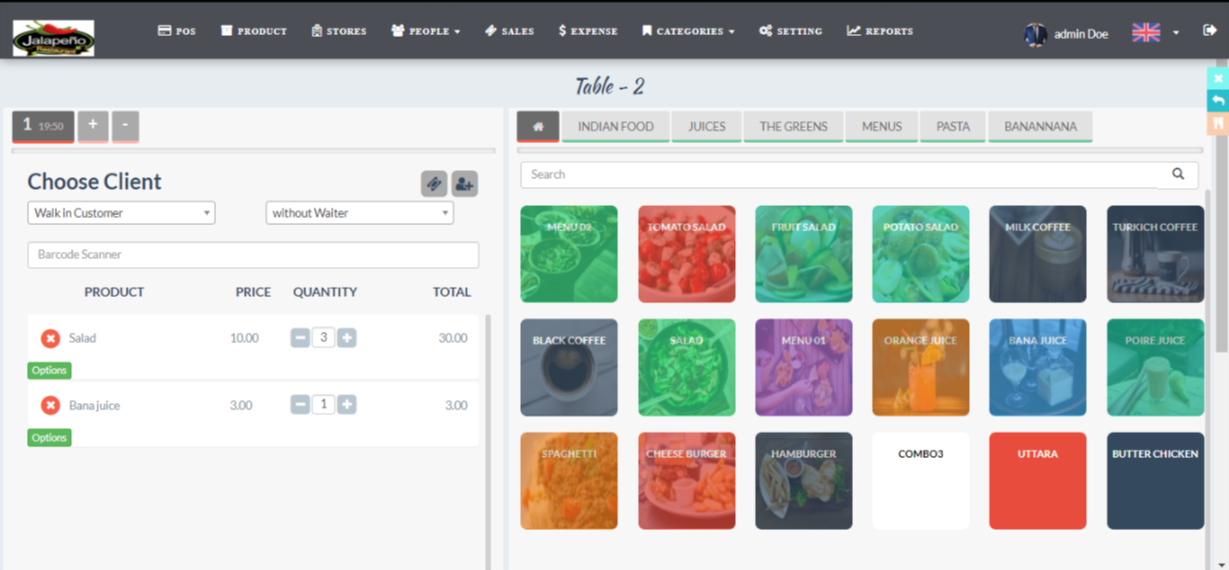
1. **Kitchen HomePage:**



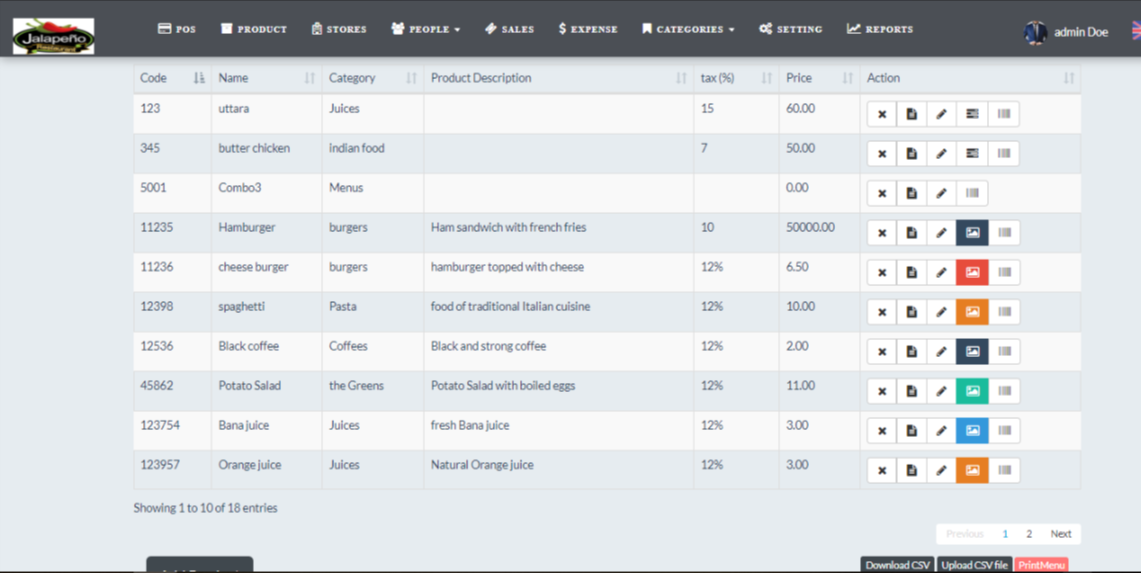
*Kitchen table page*



*Receipt Dialog*

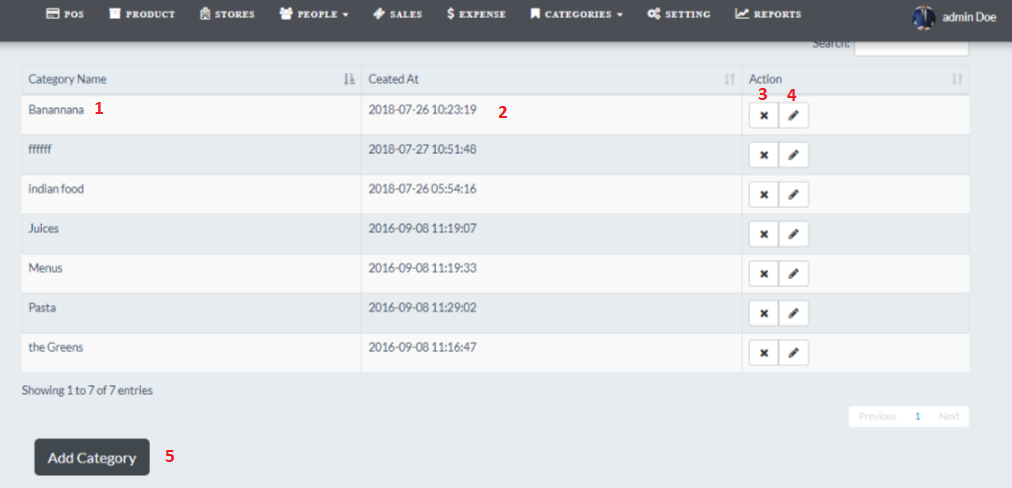
1. **Order page:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

1. **Product Manager Page**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

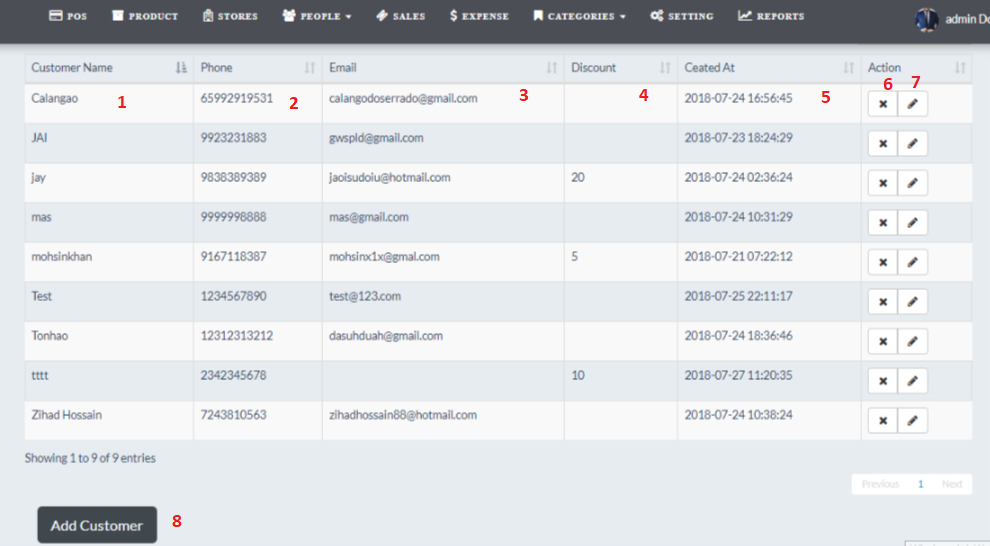
1. **Categories Manager Page**

****

*Category Page*

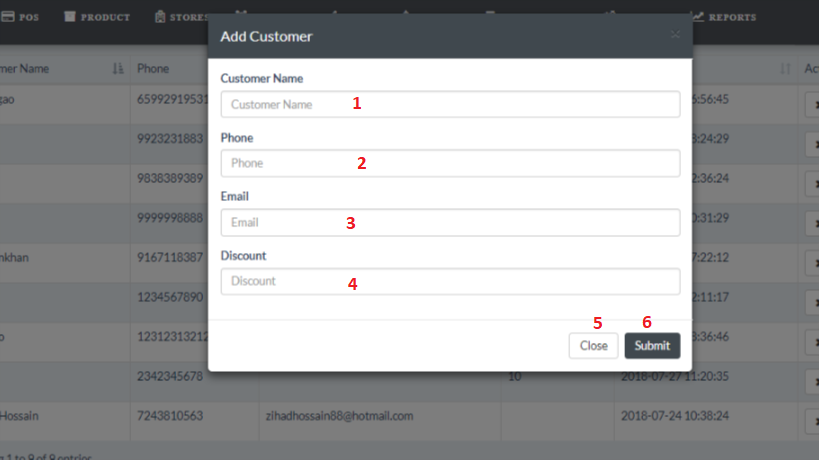
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | cat\_name | Output Text |  |  | Category Name | Disable |
| 2 | createday | Output Text |  |  | Create Day | Disable |
| 3 | Delete | Button |  | Action | Click delete, delete Category | Enable |
| 4 | Edit | Button |  | Action | Click edit, open edit Category dialog | Enable |
| 5 | Add Category | Button |  | Action | Click add, open add new Category dialog | Enable |

1. **Customer Manager Page**



*Customer page*

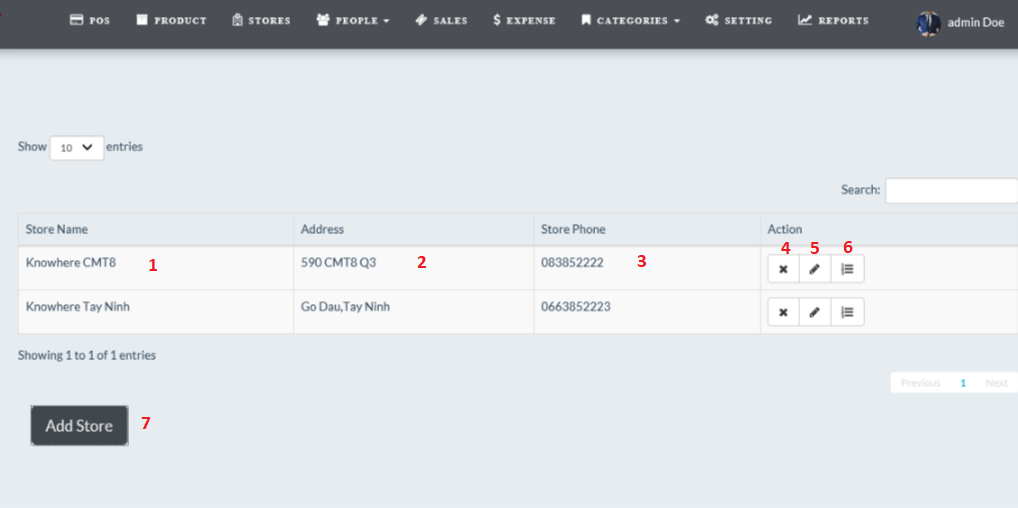
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | Cus\_name | outputText |  |  | Customer Name | Disable |
| 2 | Phone | outputText |  |  | Phone | Disable |
| 3 | Email | outputText |  |  | Email | Disable |
| 4 | Discount | outputText |  |  | Discount | Disable |
| 5 | Createday | outputText |  |  | CreateDay | Disable |
| 7 | Delete | Commandbutton |  | Action | Delete Customer | Enable |
| 8 | Edit | Commandbutton |  | Action | Open Edit Customer dialog | Enable |
| 9 | Add new Customer | Button |  | Action | Open Add Customer dialog | Enable |

****

*Customer Dialog*

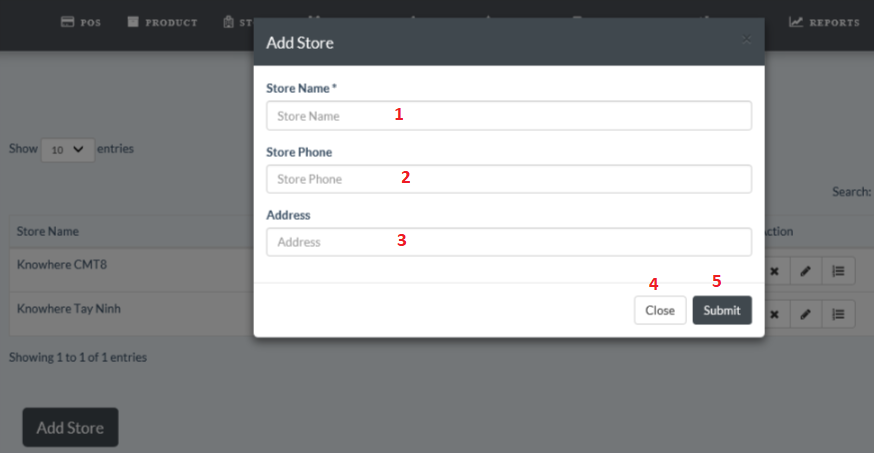
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | Cus\_name | inputText | Not blank, 5 – 50 character |  | Customer Name | Enable |
| 2 | Phone | inputText |  |  | Phone | Enable |
| 3 | Email | inputText |  |  | Email | Enable |
| 4 | Discount | inputText |  |  | Discount | Enable |
| 5 | Close | Button |  | Action | Return Customer page | Enable |
| 6 | Submit | Button |  | Action | Add new Customer in database | Enable |

1. **Store Manager Page**

****

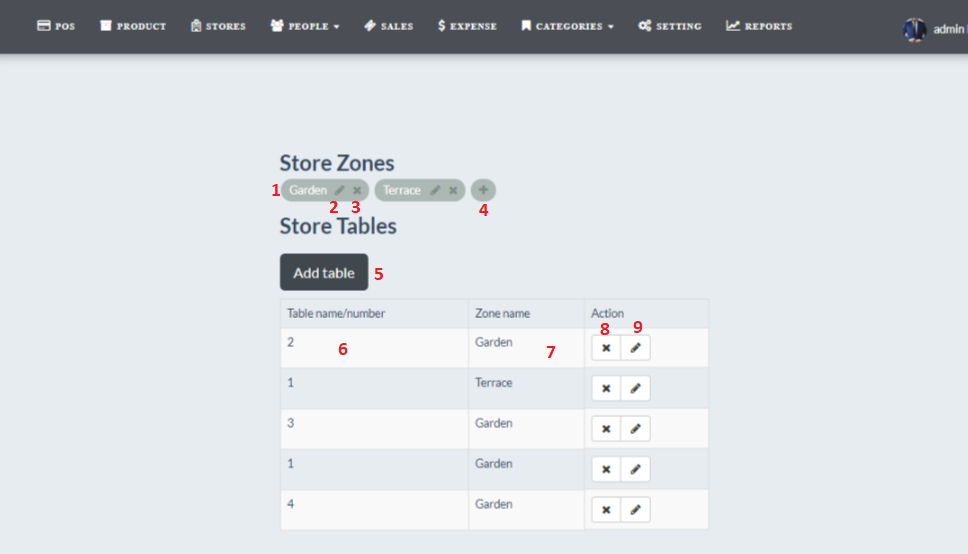
*Store Page*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | storeName | outputText |  |  | Store Name | Disable |
| 2 | Address | outputText |  |  | Address | Disable |
| 3 | storePhone | outputText |  |  | Store Phone | Disable |
| 4 | Delete Store | commandLink |  | Action | Delete Store | Enable |
| 5 | Edit Store | commandLink |  | Action | Open Edit Store dialog | Enable |
| 6 | Manager Table | commandLink |  | Action | Open TableManager Page | Enable |
| 7 | Add Store | Button |  | Action | Open Add Store dialog | Enable |

****

*Add Store Dialog*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | storeName | inputText | Not blank, 5 -50 character |  | Store Name | Enable |
| 2 | storePhone | inputText |  |  | Store Phone | Enable |
| 3 | Address | inputText |  |  | Address | Enable |
| 4 | Close | Button |  | Action | Close Add Store dialog | Enable |
| 5 | Submit | Button |  | Action | Insert new store in database | Enable |

****

*Table Manager Page*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | zoneName | OutputText |  |  | Zone Name | Disable |
| 2 | Edit Zone | commandLink |  | Action | Open edit zone dialog | Enable |
| 3 | Delete zone | commandLink |  | Action | Delete zone | Enable |
| 4 | Add zone | CommandLink |  | Action | Add new zone | Enable |
| 5 | Add table | Button |  | Action | Add new table | Enable |
| 6 | tableName | outputText |  |  | Table Name | Disable |
| 7 | zoneName | outputText |  |  | Zone Name | Disable |
| 8 | Delete Table | commandLink |  | Action | Delete Table | Enable |
| 9 | Edit Table | commandLink |  | Action | Edit Table | Enable |

1. **Report Page**

