Contents

[1. Design Overview 3](#_Toc325963057)

[2. System Architectural Design 4](#_Toc325963058)

[3. Component Diagram 5](#_Toc325963059)

[4. Detailed Description of Components 6](#_Toc325963060)

[4.1: Component diagram 6](#_Toc325963061)

[4.2. CRC Cards (Class-Responsibility-Collaborators) and Description 7](#_Toc325963062)

[4.2.1: Customer 7](#_Toc325963063)

[4. 2.2: Customer Controller 7](#_Toc325963064)

[4. 2.3: Order 8](#_Toc325963065)

[4. 2.4: Order Controller 8](#_Toc325963066)

[4. 2.5: User 9](#_Toc325963067)

[4. 2.6: User Controller 9](#_Toc325963068)

[4. 2.7: Gas 10](#_Toc325963069)

[4. 2.8: Gas Controller 11](#_Toc325963070)

[4. 2.9: Supplier 11](#_Toc325963071)

[4. 2.10: Supplier Controller 12](#_Toc325963072)

[5. Sequence Diagram 13](#_Toc325963073)

[5.1 Manipulate data 13](#_Toc325963074)

[5.2: Select data 13](#_Toc325963075)

[6. User Interface Design 14](#_Toc325963076)

[6.1: All-customer form: 14](#_Toc325963077)

[6.2: Edit form. 15](#_Toc325963078)

[6.3: Report Import/ Export: 15](#_Toc325963079)

[6.4: Sell Statistic: 16](#_Toc325963080)

[6.5: Call diary: 16](#_Toc325963081)

[6.6 Sale Statistic and Employee 17](#_Toc325963082)

[6.7 Setup modem 18](#_Toc325963083)

[6.8 : Supplier list: 18](#_Toc325963084)

[6.9: Employee list: 19](#_Toc325963085)

[6.10: Import ourself: 19](#_Toc325963086)

[6.11: Import from suppliers: 20](#_Toc325963087)

[6.12: Check repository: 20](#_Toc325963088)

[6.12: Notification: appear when someone is calling 21](#_Toc325963089)

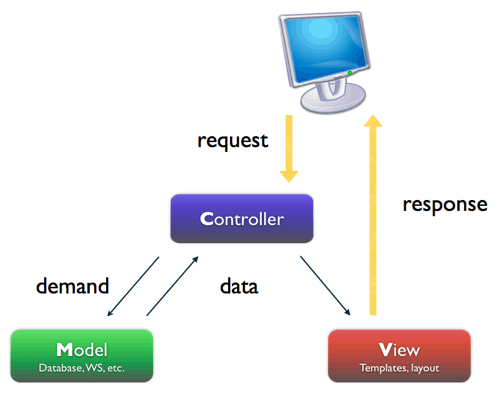
[7. Database Design or Data Structures 21](#_Toc325963090)

# 

# 1. Design Overview

Our Gas Seller System uses multi-tier design. Our multi-tier design contains three part:

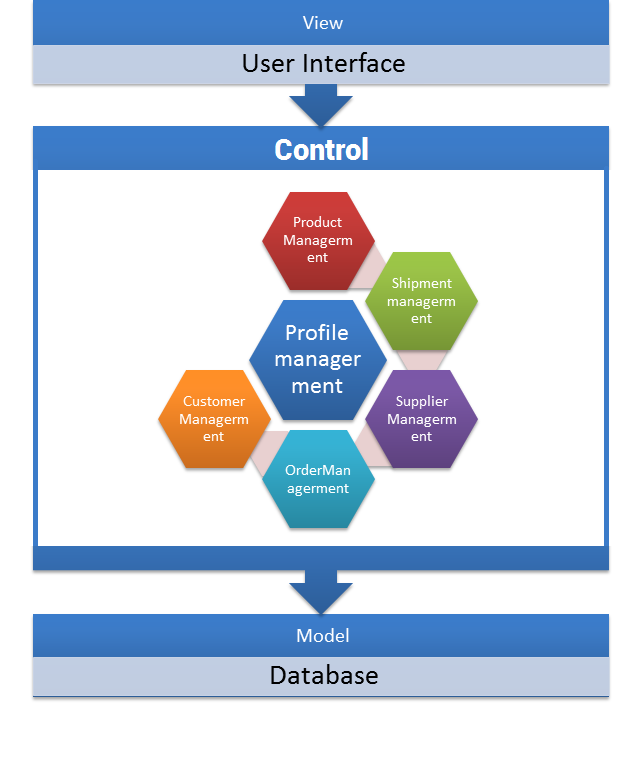
* **View**: Receives Input and display output for users. This part include Form, UI component, sound and everythings that interact with users. Based on role of user (admin, moderation, deliverer), the View display suitable UI. For example: Admin can add and remove orders but deliverer only can see the order.
* **Control**: translates user input into operations on the model. Execute function that modify data from Model part. Check for securities access, database injection.
* **Model**: Manager Data, check and give notification for users (for promotion event, user buying…) and respond error for admin.



# 2. System Architectural Design

4.2.1 Choice of System Architecture

- Multities architect.



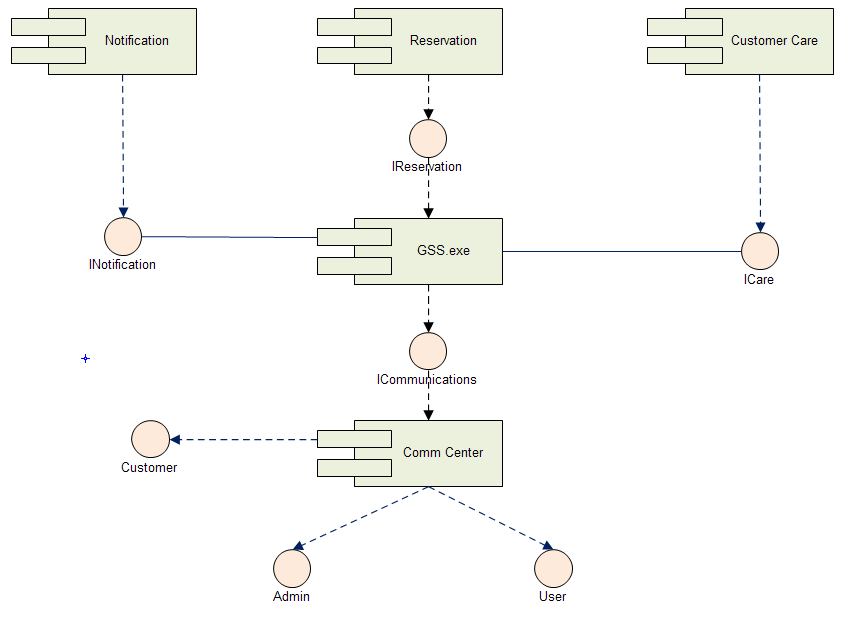
We consider some important attribute for our system:

**Perfomance**: It’s most important attribute of our system. With every database query, it should respond within 5 seconds. Progam need to work corectly and never crash itself. Longest delay must be beetwen CallerID and computor, not communication beetwen softwares components.

**Usability**: We build a easy-to-use software for user with tree view like windows explorer. Sepherate list of users function and edit user details function. Use friendly color and component. With notifications function, users can reduce their effort for searching in database. Just sit in front of computor and drink coffe. Computor can do your work automaticaly.

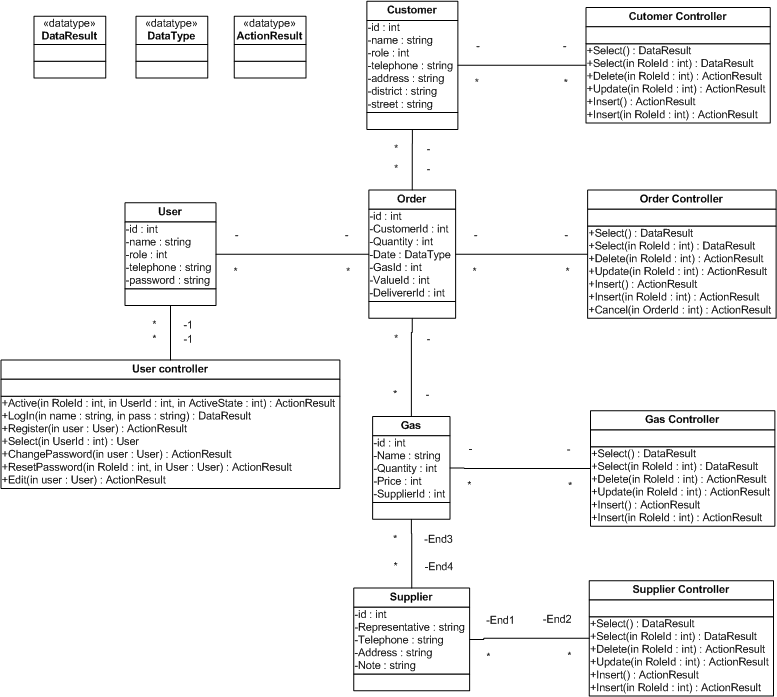
**Security**: We created a multi-roles software for some users. You can be admin and add more user, delete any records, make report and do anything you want. All password is stored by hash algorythm.

# 3. Component Diagram

****

# 4. Detailed Description of Components

## 4.1: Component diagram



## 4.2. CRC Cards (Class-Responsibility-Collaborators) and Description

### 4.2.1: Customer

|  |  |
| --- | --- |
| Cutomer | |
| Responsibilities | **Collaborator** |
| * Keep information of customer. | * Order * CustomerControler |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Size | Constrain | Description |
| 1 | Id | Integer | 128 bits | Unique | Unique Id for Customer |
| 2 | Name | String | 100 characters | Only alphabet characters | Name of customer |
| 3 | Telephone | String | No | No | Phone number of customer |
| 4 | Address | String | 100 character, Except district and street | No | Address of customer |
| 5 | District | String | 50 character | No | Address of customer |
| 6 | Street | String | 50 character | No | Address of customer |

### 4. 2.2: Customer Controller

|  |  |
| --- | --- |
| Cutomer Controler | |
| Responsibilities | **Collaborator** |
| * Modify information of customer. * Auto notify user when customer is approved to a promotion event. * Auto create,select customer when someone is calling. | * Customer |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Parameter | Return type | Exception | Description |
| 1 | Select |  | DataResult | NotFoundException | Search name of Customer |
| 2 | Select | RoleId | DataResult | NotFoundException  RoleNotEnoughException | Search Customer |
| 3 | Delete | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Delete Customer |
| 4 | Update | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Update Customer |
| 5 | Insert |  | Action Result |  | Insert Customer |
| 6 | Insert | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Insert Customer |
|  |  |  |  |  |  |

### 4. 2.3: Order

|  |  |
| --- | --- |
| Order | |
| Responsibilities | **Collaborator** |
| * Keep information of Order. | * OrderControler * Customer * Gas * User |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Size | Constrain | Description |
| 1 | Id | Integer | 128 bits | Unique | Unique Id for Order |
| 2 | CustomerId | Integer | 128 bits | Unique | Unique Id for Customer |
| 3 | Quantity | Integer | 128 bits | Just Number | Number of gas |
| 4 | Date | DateTime |  | After year 2011 | Date of order |
| 5 | GasId | Integer | 128 bits | Unique | Unique Id for Gas |
| 6 | ValueId | Integer | 128 bits | Unique | Unique Id for Value |
| 7 | DeliverId | Integer | 128 bits | Unique | Unique Id for Deliver |

### 4. 2.4: Order Controller

|  |  |
| --- | --- |
| Order Controller | |
| Responsibilities | **Collaborator** |
| * Modify information of Order. * Every user can make Order but only Admon can delete Order record. | * Order |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Parameter | Return type | Exception | Description |
| 1 | Select |  | DataResult | NotFoundException | Search name of Customer order |
| 2 | Select | RoleId | DataResult | NotFoundException  RoleNotEnoughException | Search Order |
| 3 | Delete | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Delete Order |
| 4 | Update | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Update Order |
| 5 | Insert |  | Action Result |  | Insert Order |
| 6 | Insert | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Insert Order |
|  |  |  |  |  |  |

### 4. 2.5: User

|  |  |
| --- | --- |
| User | |
| Responsibilities | **Collaborator** |
| * Keep information of user User | * UserControler * Order |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Size | Constrain | Description |
| 1 | Id | Integer | 128 bits | Unique | Unique Id for User |
| 2 | Name | String | 100 | Just characters | Name of user |
| 3 | Role | Integer | 128 bits | Unique | Unique Id for Role (invidual, restaurent, family) |
| 4 | Telephone | String | No | No | Phone number of customer |
| 5 | Password | String | 32 character | No | Hash string of password |

### 4. 2.6: User Controller

|  |  |
| --- | --- |
| User Controller | |
| Responsibilities | **Collaborator** |
| * Manage information of user User * Password is stored as hash string. * Only Admin can add and modify user information. | * User |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Parameter | Return type | Exception | Description |
| 1 | Active | RoleId  UserId  ActiveState | ActionResult | NotFoundException | Active or Deactive user |
| 2 | Login | Name  Pass | DataResult | NotFoundException | Login. Check hash password in database. |
| 3 | Register | User | ActionResult | NameExistException | Register new User. Only admin can do that |
| 4 | Select | UserId  Role | User | NotFoundException | Select Users. Only admin can do that. |
| 5 | ChangePassword | User | ActionResult | NotFoundException | Change Password. Only user can change password themselves. |
| 6 | Edit | User | ActionResult | NotFoundException | Edit detail, only user can do that. |
| 7 | ResetPassword | UserId  Role | ActionResult | NotFoundException | Reset password to ‘12345678’. Only admin can do that. |

### 4. 2.7: Gas

|  |  |
| --- | --- |
| Gas | |
| Responsibilities | **Collaborator** |
| * Keep information of Gas. | * GasControler * Supplier * Order |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Size | Constrain | Description |
| 1 | Id | Integer | 128 bits | Unique | Unique Id for Gas |
| 2 | Name | String | 50 | Just characters | Name of gas |
| 3 | Quantity | Integer | 128 bits | No | Quantity of Gas |
| 4 | Price | Integer | 128 bits | No | Prices per gas. |
| 5 | SupplierId | Integer | 128 bits | Unique | Unique Id for Supplier |
|  |  |  |  |  |  |

### 4. 2.8: Gas Controller

|  |  |
| --- | --- |
| Gas Controller | |
| Responsibilities | **Collaborator** |
| * Manage information of gas. * Auto notify when user run out of gas by tracking gas using of users. | * Gas |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Parameter | Return type | Exception | Description |
| 1 | Select |  | DataResult | NotFoundException | Search name of Gas |
| 2 | Select | RoleId | DataResult | NotFoundException  RoleNotEnoughException | Search Gas |
| 3 | Delete | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Delete Gas |
| 4 | Update | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Update Gas |
| 5 | Insert |  | Action Result |  | Insert Gas |
| 6 | Insert | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Insert Gas |
|  |  |  |  |  |  |

### 4. 2.9: Supplier

|  |  |
| --- | --- |
| Supplier | |
| Responsibilities | **Collaborator** |
| * Keep information of Supplier. | * Supplier controller * Gas |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Size | Constrain | Description |
| 1 | Id | Integer | 128 bits | Unique | Unique Id for Supplier |
| 2 | Representative | String | 100 | Just characters | Name of representative |
| 3 | Telephone | String | 30 | No | Name of representative |
| 4 | Address | String | No | No | Address of Supplier |
| 5 | Note | String | No | No | Note |
|  |  |  |  |  |  |

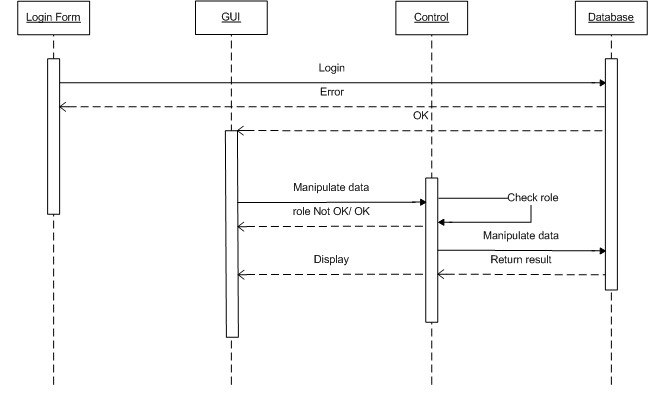
### 4. 2.10: Supplier Controller

|  |  |
| --- | --- |
| Supplier Controller | |
| Responsibilities | **Collaborator** |
| * Manage information of Supplier. | * Supplier |

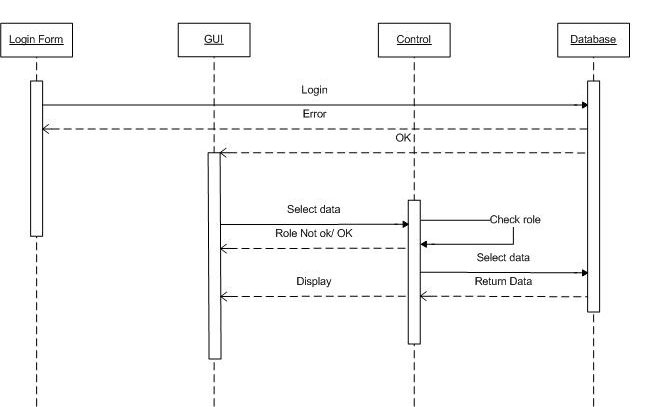
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Parameter | Return type | Exception | Description |
| 1 | Select |  | DataResult | NotFoundException | Search name of Supplier |
| 2 | Select | RoleId | DataResult | NotFoundException  RoleNotEnoughException | Search Supplier |
| 3 | Delete | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Delete Supplier |
| 4 | Update | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Update Supplier |
| 5 | Insert |  | Action Result |  | Insert Supplier |
| 6 | Insert | RoleId | Action Result | NotFoundException  RoleNotEnoughException | Insert Supplier |
|  |  |  |  |  |  |

# 5. Sequence Diagram

## 5.1 Manipulate data

****

## 5.2: Select data

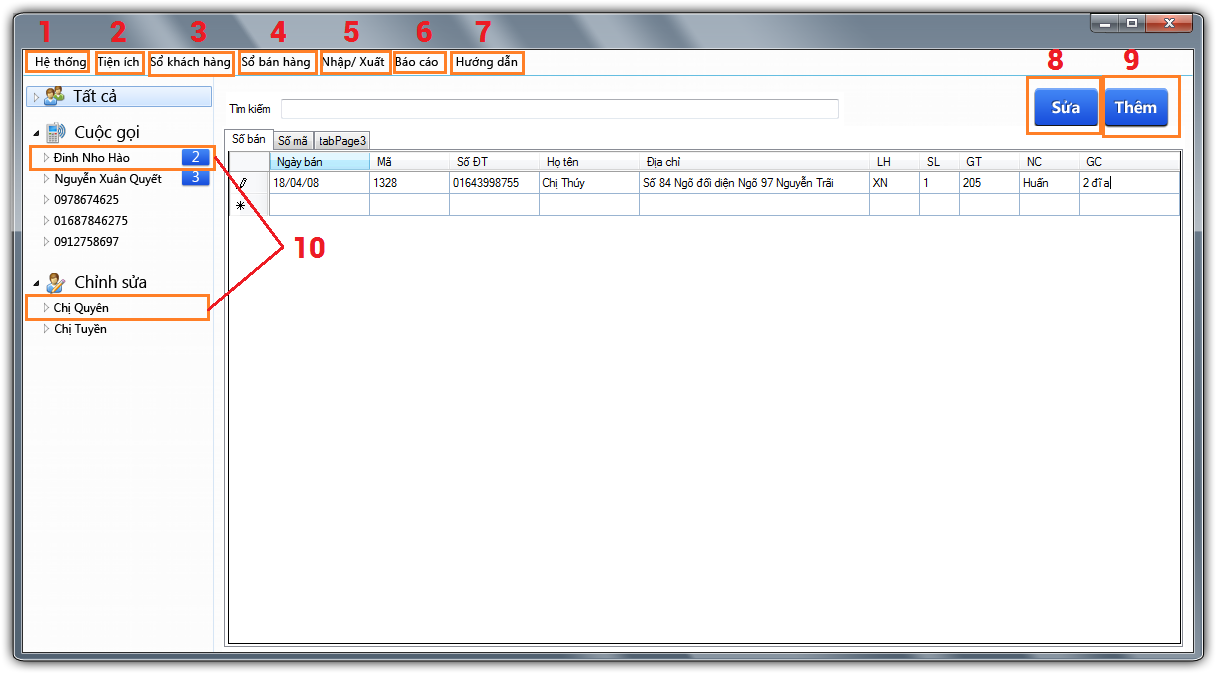
****

# 6. User Interface Design

**Overview: K**eep **I**t **S**hort & **S**imple is very important attribute of our design. Button have big font size and friendly color. Notification system appears whenever system found that we need it. We just have 2 forms. Other functions are designed as simple popup.

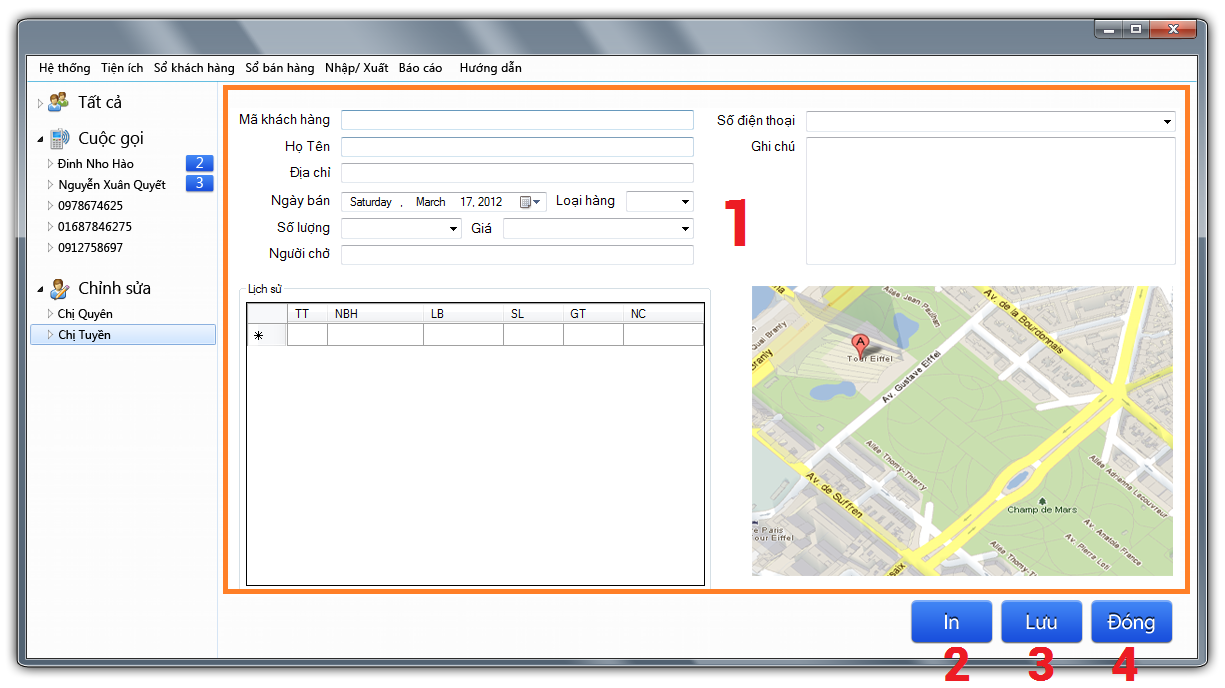
User will be able to access system after the validation of their roles.

## 6.1: All-customer form:



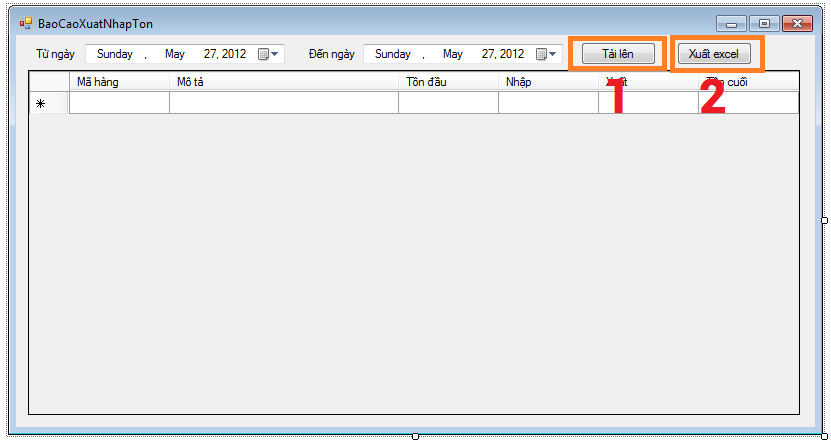
|  |  |  |
| --- | --- | --- |
| Number | Name | Description |
| 1 | System | System setting and control (Time, modem configuration, bill configuration, backup, employee list, suppliers list, good list) |
| 2 | Utilities | Insert, Update, Delete database. Fake call, datachange diary. |
| 3 | Customer manager | Remove customer, display customer, export customer list |
| 4 | Seller Manager | Remove selected sell record, export seller statistic, check sell from date…to date. |
| 5 | Import/Export | Check, import/ export from supplier, import/ export from itself. |
| 6 | Report | Create report call, sell, employee, customer… |
| 7 | Help | Manual |
| 8 & 10 | Edit | Edit customer |
| 9 | Add | Add customer |

## 6.2: Edit form.



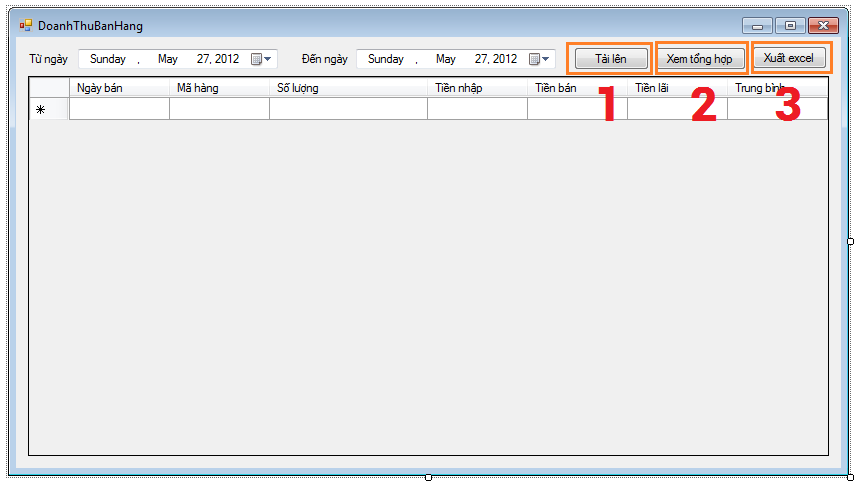
|  |  |  |
| --- | --- | --- |
| Number | Name | Description |
| 1 | Edit fields | Details of selected customer |
| 2 | Print | Print bill. |
| 3 | Save | Save details to database |
| 4 | Close | Close editing panel and open All customer panel |
|  |  |  |

## 6.3: Report Import/ Export:



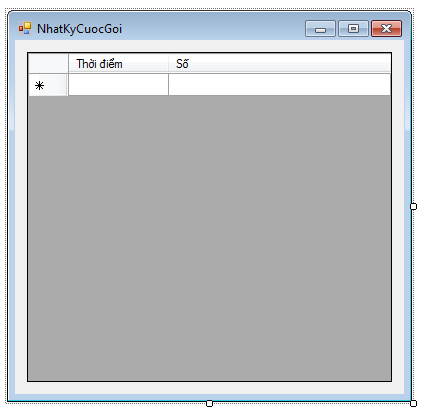
|  |  |  |
| --- | --- | --- |
| Number | Name | Description |
| 1 | Import | Import from excel |
| 2 | Export | Export to excel |
|  |  |  |

## 6.4: Sell Statistic:

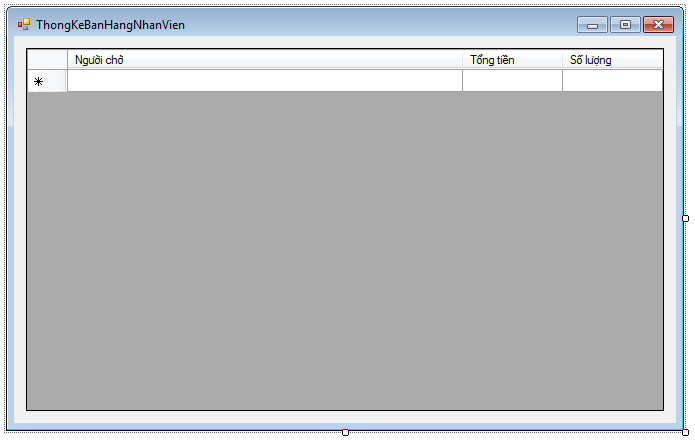


|  |  |  |
| --- | --- | --- |
| Number | Name | Description |
| 1 | Import | Import from excel |
| 2 | See All | See all sale statistic |
| 3 | Export | Export to excel |

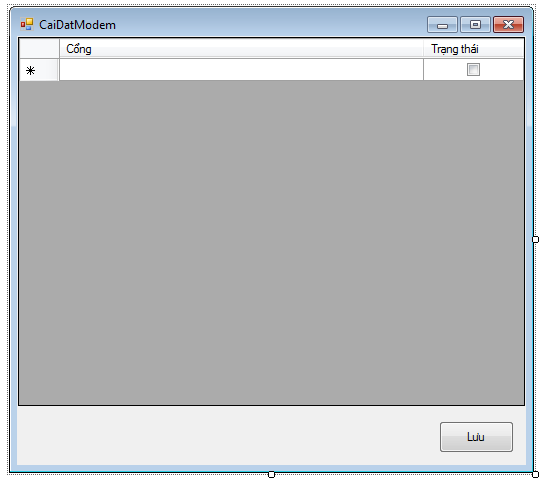
## 6.5: Call diary:



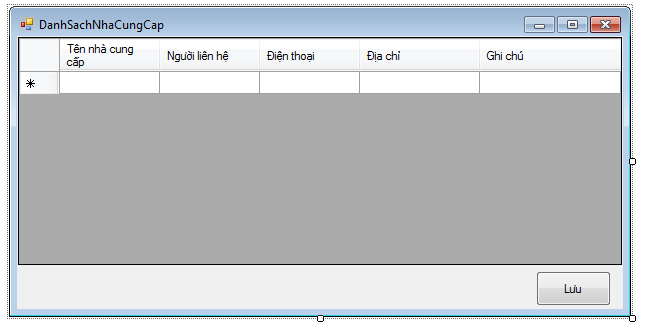
## 6.6 Sale Statistic and Employee



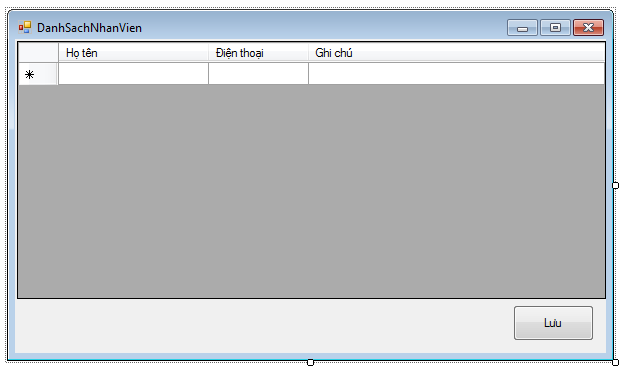
## 6.7 Setup modem



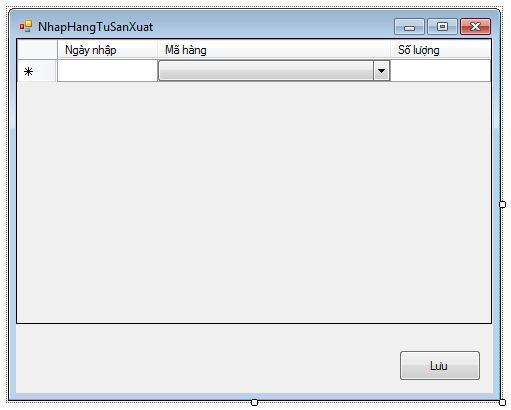
## 6.8 : Supplier list:



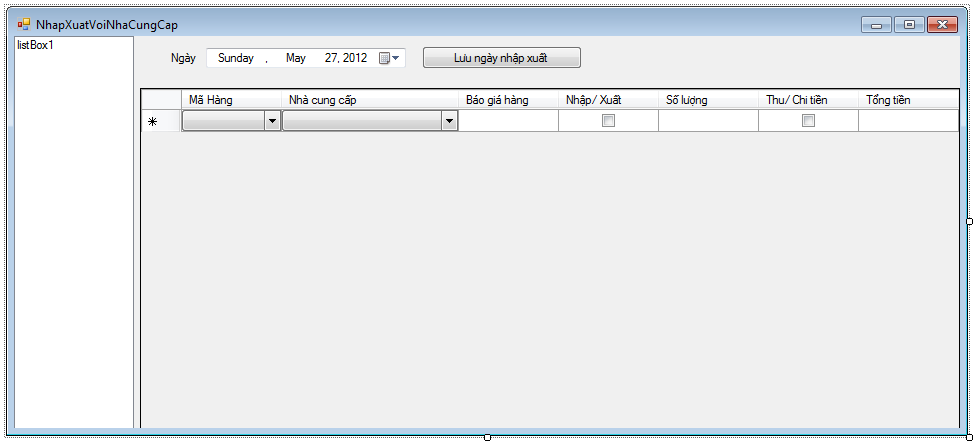
## 6.9: Employee list:



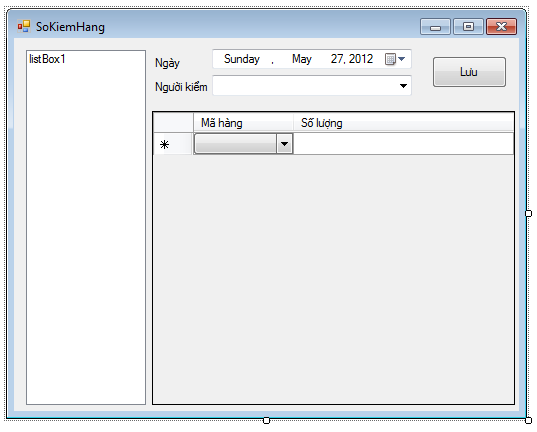
## 6.10: Import ourself:



## 6.11: Import from suppliers:



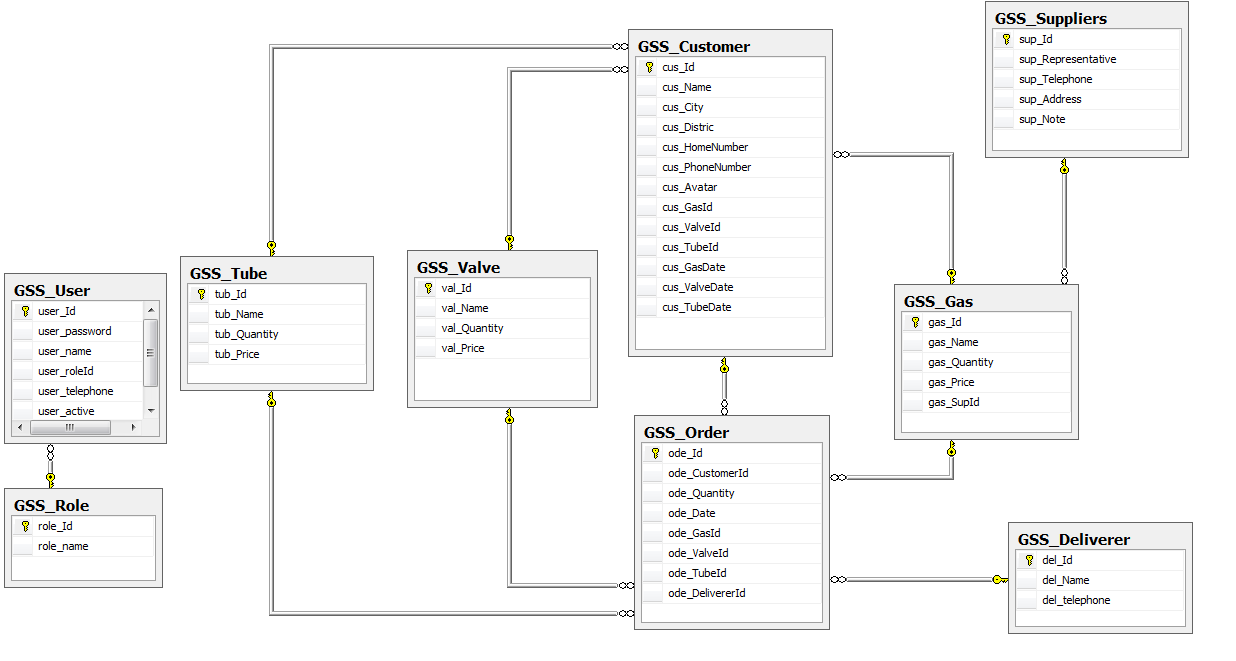
## 6.12: Check repository:



## 6.12: Notification: appear when someone is calling



# 7. Database Design or Data Structures

**­­­**