

**Manage Warehouse**

* **Faculty: Ho Hoan Kiem**
* **Batch:**
* **Team Member:**
  + - * **Đinh Quốc Hậu**
      * **Mai Tân Chinh**

**Ho Chi Minh City, Sep 10 2012**

Table of Contents

1. Review 1

* **Problem Definition..…………………………………………………………………………………01**
* **Customer Request Specification……………………………………………………………… 02**

1. Review 2

* ***Use Case Diagram* 04**
* ***Class Diagram* 15**
* ***Sequence Diagram* 16**
* ***ERD* 18**
* ***Entity Description* 19**
* ***Table Relationship Diagram* 26**

1. Review 3

* Interface Design………………………………………………………………………………..... 27
* Site Map……………………………………………………………………………………………… 51
* Report ………………………………………………………………………………………………..53
* Tasksheet……………………………………………………………………………………………. 56

REVIEW 1

Problem Definition

1. **The Object Of The Project**

This project aims to create a convenient store management system with small and medium scale. This will only be accessible to store owners with a unique id and password. This system provides the following facilities:

* Import goods into the warehouse.
* Exporting goods from warehouse.
* Print retail invoices to customers.
* Manage the amount of goods in stock.
* Goods information management.

Customer Request   
 Specification

1. **Functional Requirements**

Following is a list of functionalities of the system. More functionalities that you find appropriate can be added to this list.

* Shopkeeper must have a valid User Id and password to login to the system.
* If a wrong password is given thrice in succession, that account will be locked and the shopkeeper will not be able to use it. When an invalid password is entered a warning is given to the user that his/her account is going to get locked.
* After the valid user logs in he/she is shown the function screen.
* On selecting the function he/she is taken to a page which shows this function.
* If he/she select the function of importing goods into the warehouse, on the screen there will be lines for him/her to enter the necessary information, such as: product code, name, the number of items in 1 package, the number of packages of goods, prices of 1 package. After entering all information, the system will automatically calculate the total amount of the shipment and click Add to save.
* If he/she chooses the function of managing goods information in warehouse, he/she can edit product information, such as name and retail price of goods and click uUpdate to save it.
* Or when selecting the retail payment function, the retail screen will be displayed with a box to enter product information: code, product number. Click Add to add the product to the payment list and click Payment to issue an invoice. The system will automatically calculate the total amount to be paid save the order code.
* Shopkeeper can review all the goods sold that day and the total amount collected.

# Hardware/ Software Requirements

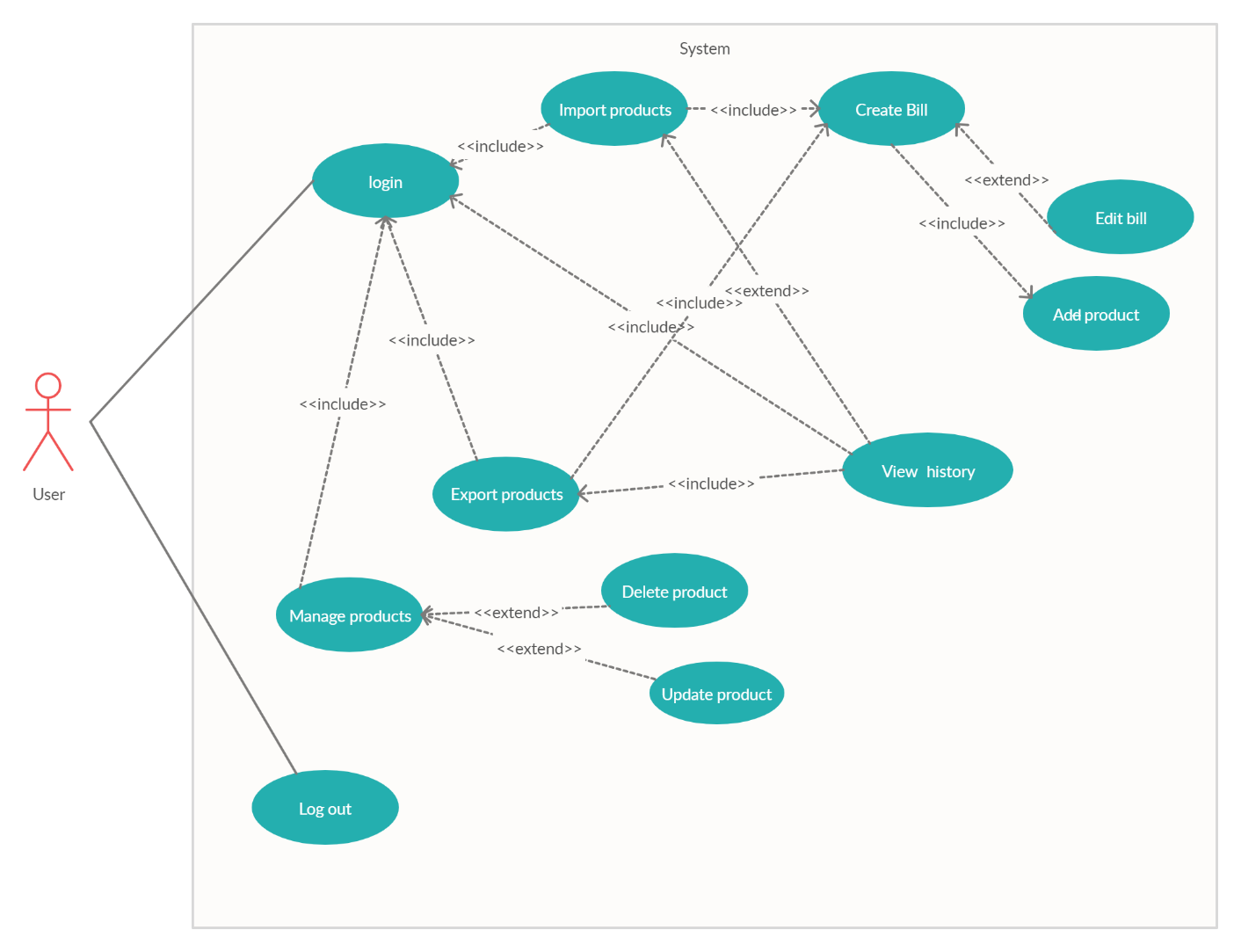
By adding new technologies such as ASP.NET 3.5, 3 layers architecture, the software becomes more interactive with users, security and easily updated.

1. **System Requirement**
   * **Hardware**
     + CPU: Pentium IV 2.0 GHz or higher
     + RAM: 512 MB or higher
   * **Software**
     + OS: Windows 7/XP Professional/Server 2003…
     + Browser: Internet Explorer 8.0 or higher

REVIEW 2

**USE CASE DIAGRAM**

**Customer :**

****

**USE CASE DESCRIPTION**

1. **Actor:**

User: Owners who buy software and are granted an account to operate the store's trading activities.

1. **Usecase**

|  |  |
| --- | --- |
| **Usecase** | Login |
| **Short Description** | Log in application to use any function of member. |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Use inputs the login information :   * Login ID. * Password.   \_User clicks on the Login button.  \_System check the Login ID and Password.  \_If Login ID and Password are invalid, system will notify error “Login Failed”. |
| **Alternative flow(s)** | \_If Login ID and Password are invalid, the Login ID field and Password field are blank.  \_ If users have problems logging in or forget their account and cannot access the system, they can contact our hotline for assistance. |

|  |  |
| --- | --- |
| **Use case** | Import product |
| **Short Description** | Import products into the warehouse. |
| **Pre-Conditions** | User must be login into the system |
| **Post-Conditions** |  |
| **Main flow** | \_ Users enter complete item information, package number and price of each package into the input fields.  \_ Users can search to see if the desired item is in stock or not with the Search feature right on the screen. If the item already exists in the warehouse, users only need to enter the number of packages and the price of each package (if you want to change) to import more products into the warehouse.  \_ If the user wants to import an entirely new item into the warehouse, press Create to go through the new item creation window. Users then fill in the item's information, press Save and go back to the Import screen to enter the quantity and price of each package.  \_ After the user completes the import, the user clicks Finish import to see a list of products that the item is imported.  \_ The user can delete the wrongly entered product and finally press Submit to save the import history and return to the function screen. |
| **Alternative flow(s)** | \_ If the customer clicks Import when the required input fields are not fully entered or in the wrong format, a warning will be displayed. |

|  |  |
| --- | --- |
| **Usecase** | Export product |
| **Short Description** | Get products out of the warehouse |
| **Pre-Conditions** | User must be login into the system |
| **Post-Conditions** |  |
| **Main flow** | \_ Users find items by entering the name of the item in the search box and clicking Search to find items in the warehouse.  \_ The user will enter the quantity, if the customer buys in a package the user will enter the required number of packages in the 'Number package' box, and if the customer wants to purchase each product individually, the user will enter the number of products in the 'Number sub package' box. Users can only enter one of them.  \_ After selecting the item to export, click Export to get the product out of stock.  \_ After finishing exporting all the required products, users click Finish to view the list of all products that have been exported in the current session.  \_ Users can review all products that have been exported, in this window users can undo any products in the list. Finally, the user press Submit to create an invoice and save the export history. |
| **Alternative flow(s)** |  |

|  |  |
| --- | --- |
| **Usecase** | Create bill |
| **Short Description** | Create bill payment |
| **Pre-Conditions** | User must be login into the system |
| **Post-Conditions** |  |
| **Main flow** | \_ The screen shows all the items that are imported or exported.  \_ User press the Submit button to print bill and save sales history. |
| **Alternative flow(s)** | \_ If the number of products to be sold is greater than the number of products that are left in stock, they will be notified on the screen. |

|  |  |
| --- | --- |
| **Usecase** | Manage product |
| **Short Description** | View product in warehouse |
| **Pre-Conditions** | User must be login into the system |
| **Post-Conditions** |  |
| **Main flow** | \_ Users can see all items in stock, the remaining quantity and besides, users can delete any product in stock and update the retail price of each product or each package. |
| **Alternative flow(s)** |  |

|  |  |
| --- | --- |
| **Usecase** | Update product |
| **Short Description** | Update retail prices |
| **Pre-Conditions** | User must be login into the system |
| **Post-Conditions** |  |
| **Main flow** | \_When viewing the remainder of stock, customers can update the retail price of each item if the market value changes. |
| **Alternative flow(s)** |  |

|  |  |
| --- | --- |
| **Usecase** | Delete product |
| **Short Description** | Remove products from the warehouse |
| **Pre-Conditions** | User must be login into the system |
| **Post-Conditions** |  |
| **Main flow** | \_In the product management screen, users can press the Remove button to delete any product from the warehouse. |
| **Alternative flow(s)** |  |

|  |  |
| --- | --- |
| **Usecase** | View history |
| **Short Description** | Total revenue |
| **Pre-Conditions** | User must be login into the system |
| **Post-Conditions** |  |
| **Main flow** | \_Calculate the total revenue earned from all products sold during the day and display it on the screen.  \_ Users can search for items that have been exported through the date. The user selects a date and press Search to search. |
| **Alternative flow(s)** |  |

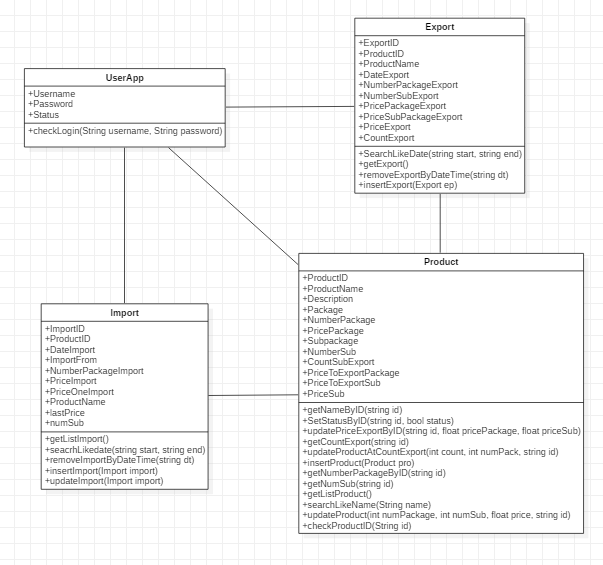
|  |  |
| --- | --- |
| **Usecase** | Add product |
| **Short Description** | Add product to bill |
| **Pre-Conditions** | User must be login into the system |
| **Post-Conditions** |  |
| **Main flow** | \_ After having enough information, the goods need to be traded. Users press Import or Export to add products to the bill. |
| **Alternative flow(s)** |  |

|  |  |
| --- | --- |
| **Usecase** | Edit bill |
| **Short Description** | Modify bill |
| **Pre-Conditions** | User must be login into the system |
| **Post-Conditions** |  |
| **Main flow** | \_ Users can edit the list of products that have been added to the bill, such as deleting products from the invoice. |
| **Alternative flow(s)** |  |

|  |  |
| --- | --- |
| **Usecase** | View payment history |
| **Short Description** | Review the products that have been paid |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | After you have finished adding products to the invoice, users can click the Checkout button to print the invoice and save payment history. |
| **Alternative flow(s)** |  |

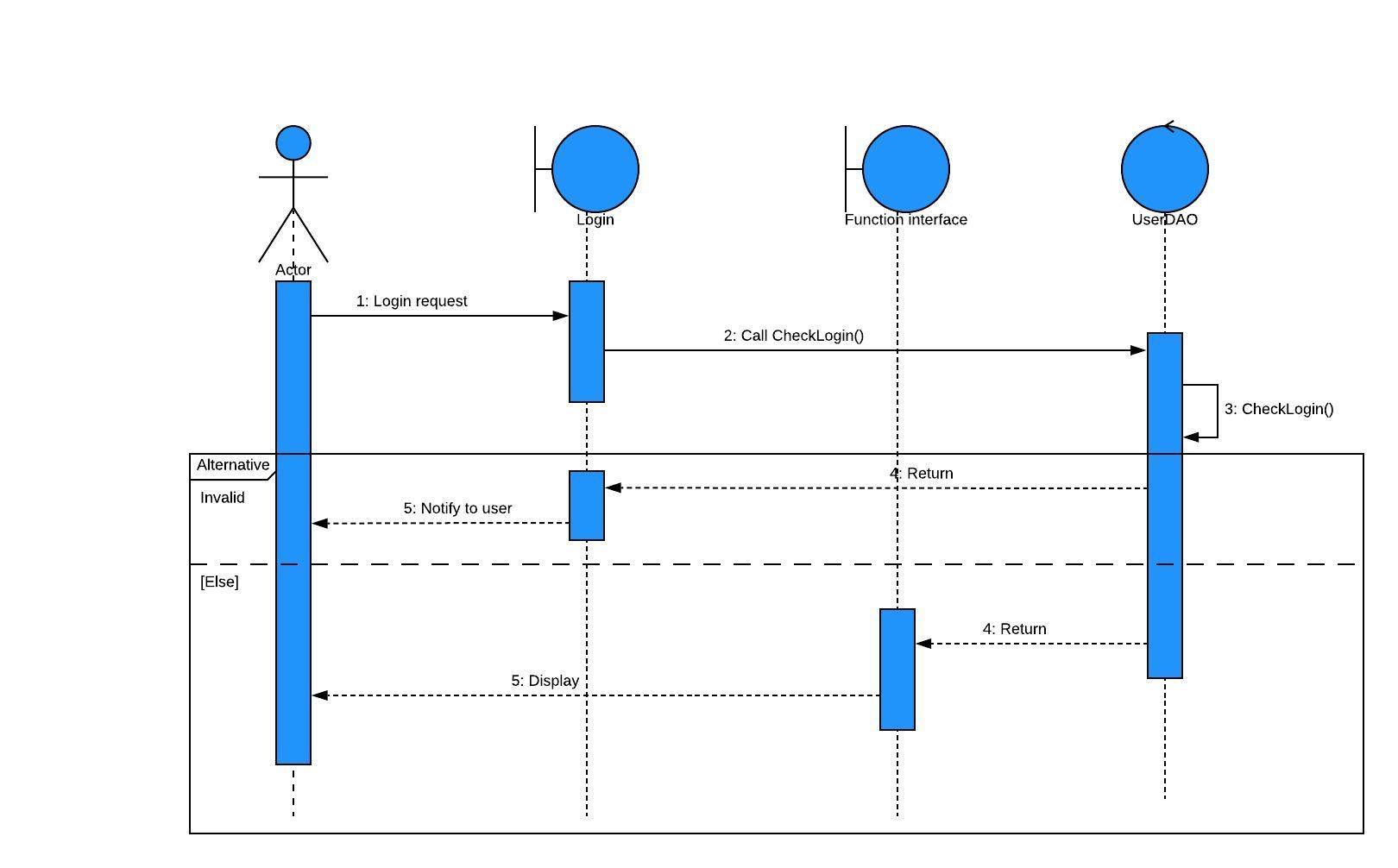
|  |  |
| --- | --- |
| **Usecase** | Log out |
| **Short Description** | Log out of the system |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | Back to login screen |
| **Alternative flow(s)** |  |

**CLASS DIAGRAM**

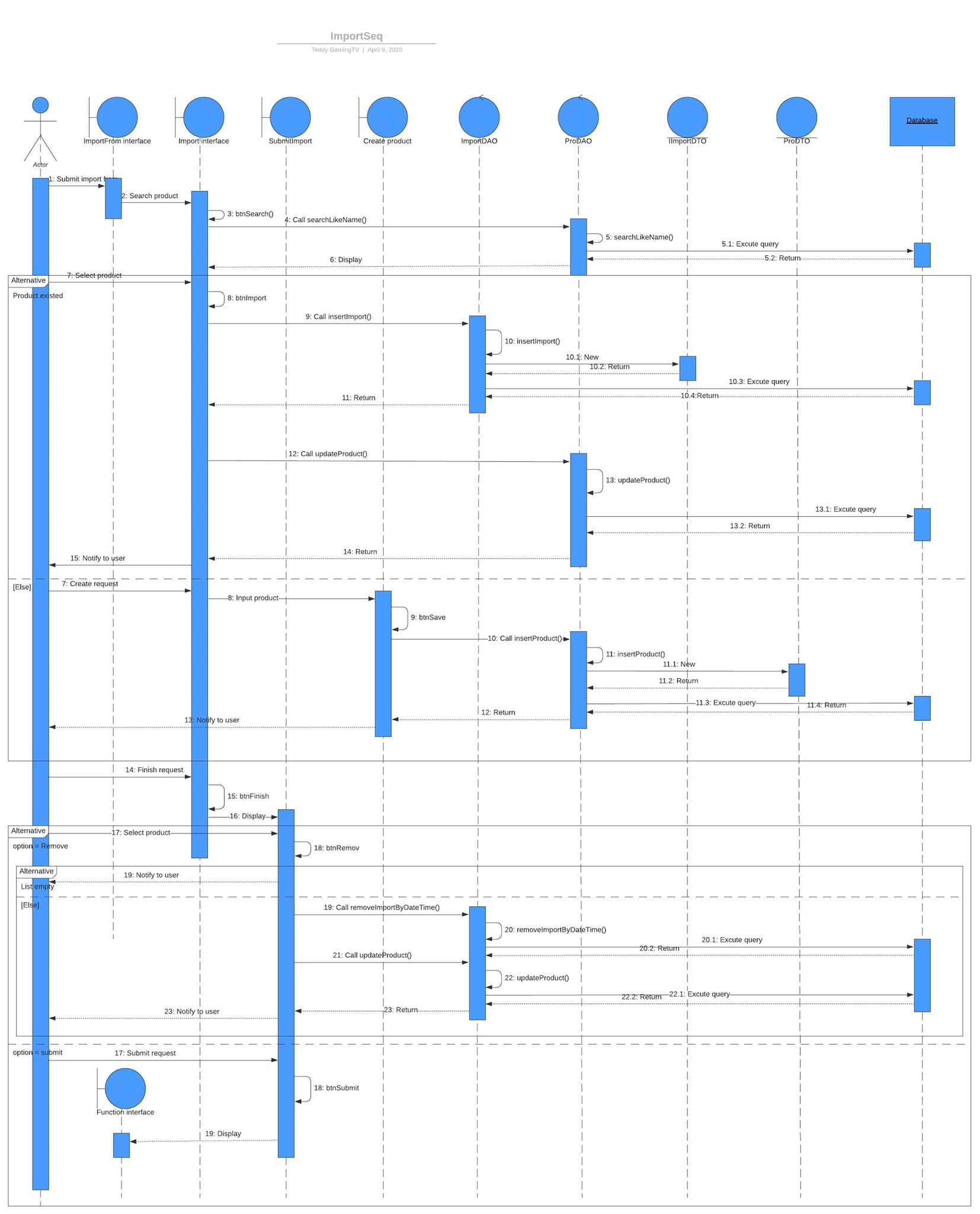


**SEQUENCE DIAGRAM**

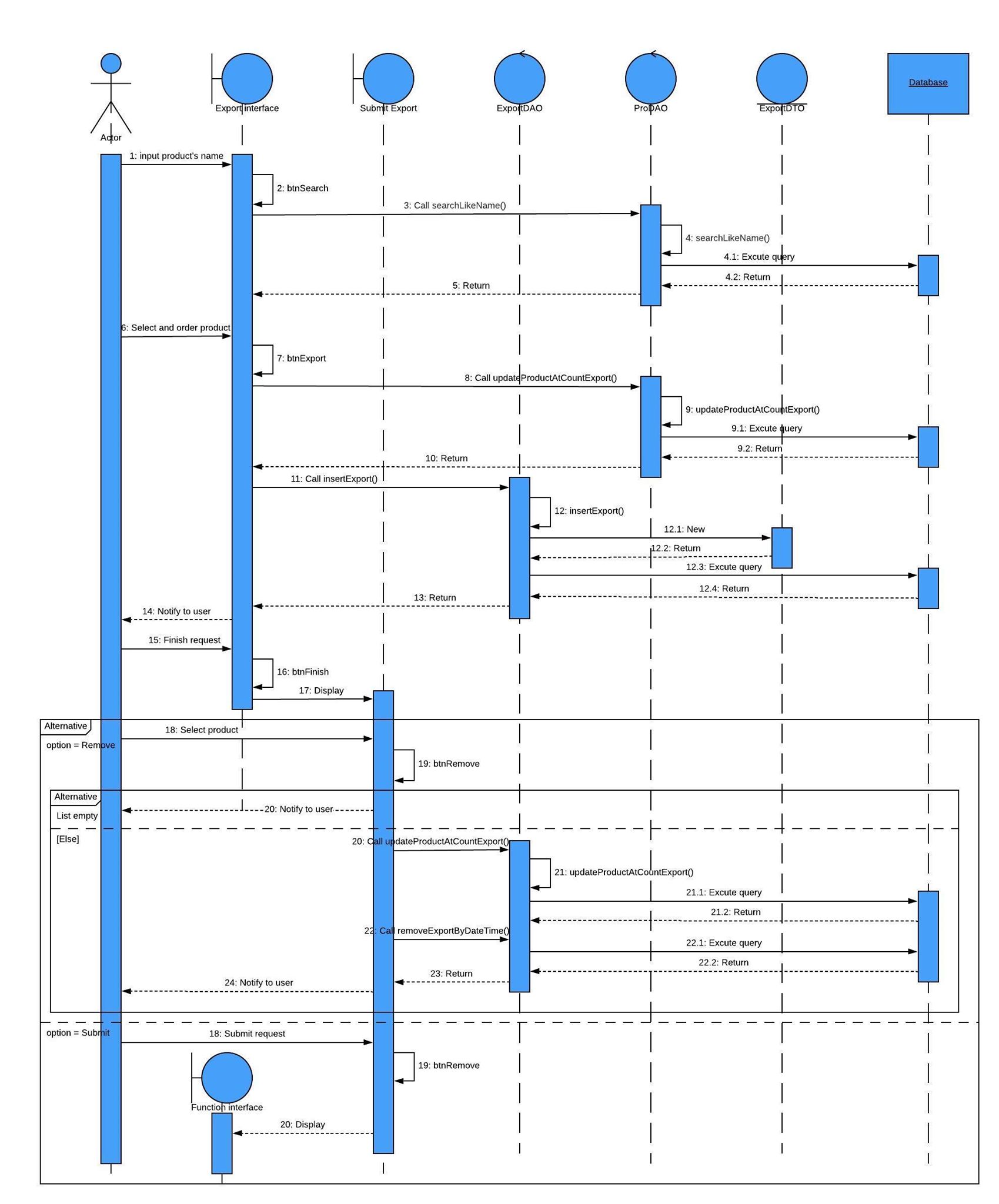
**1, login**



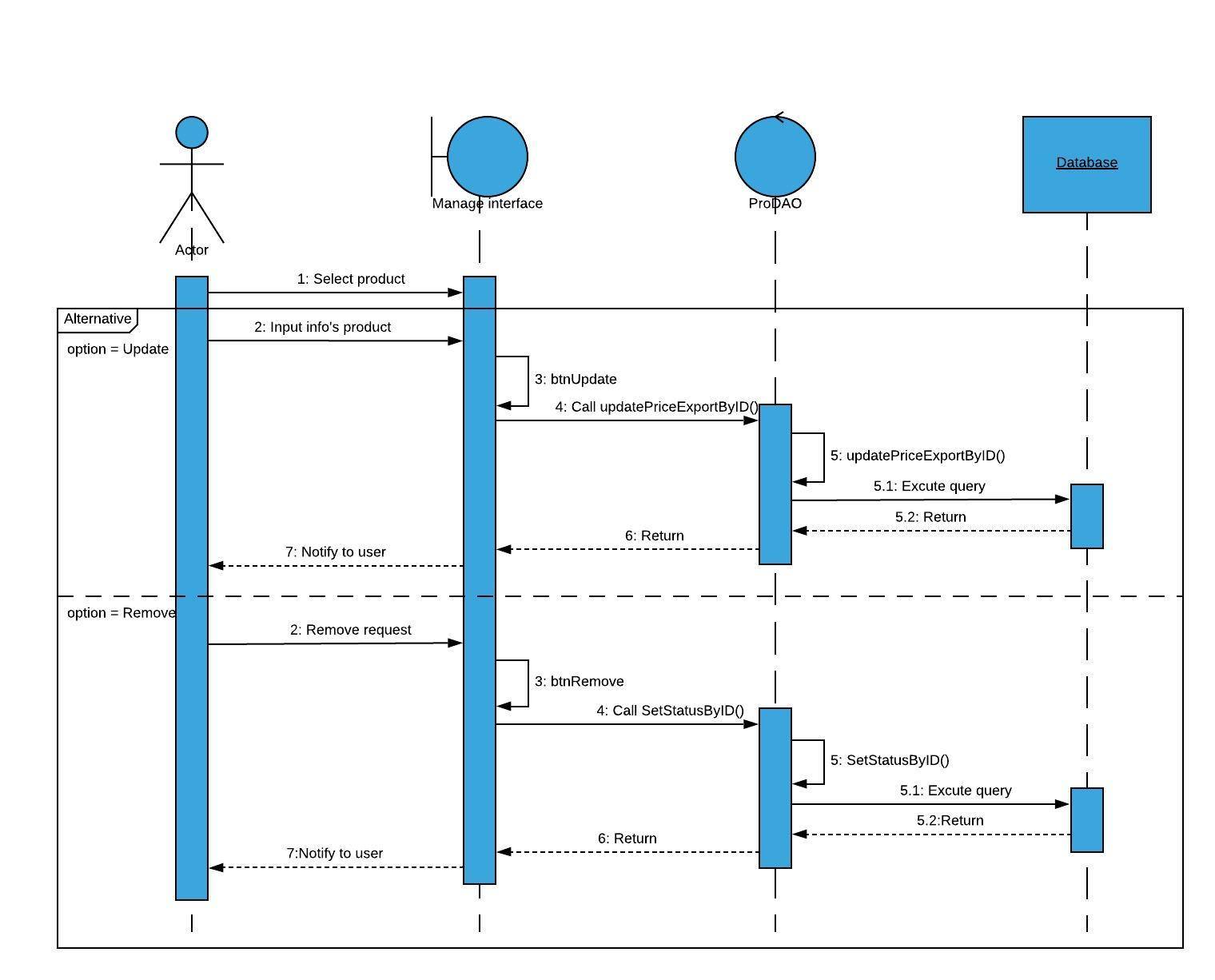
**2, Import**

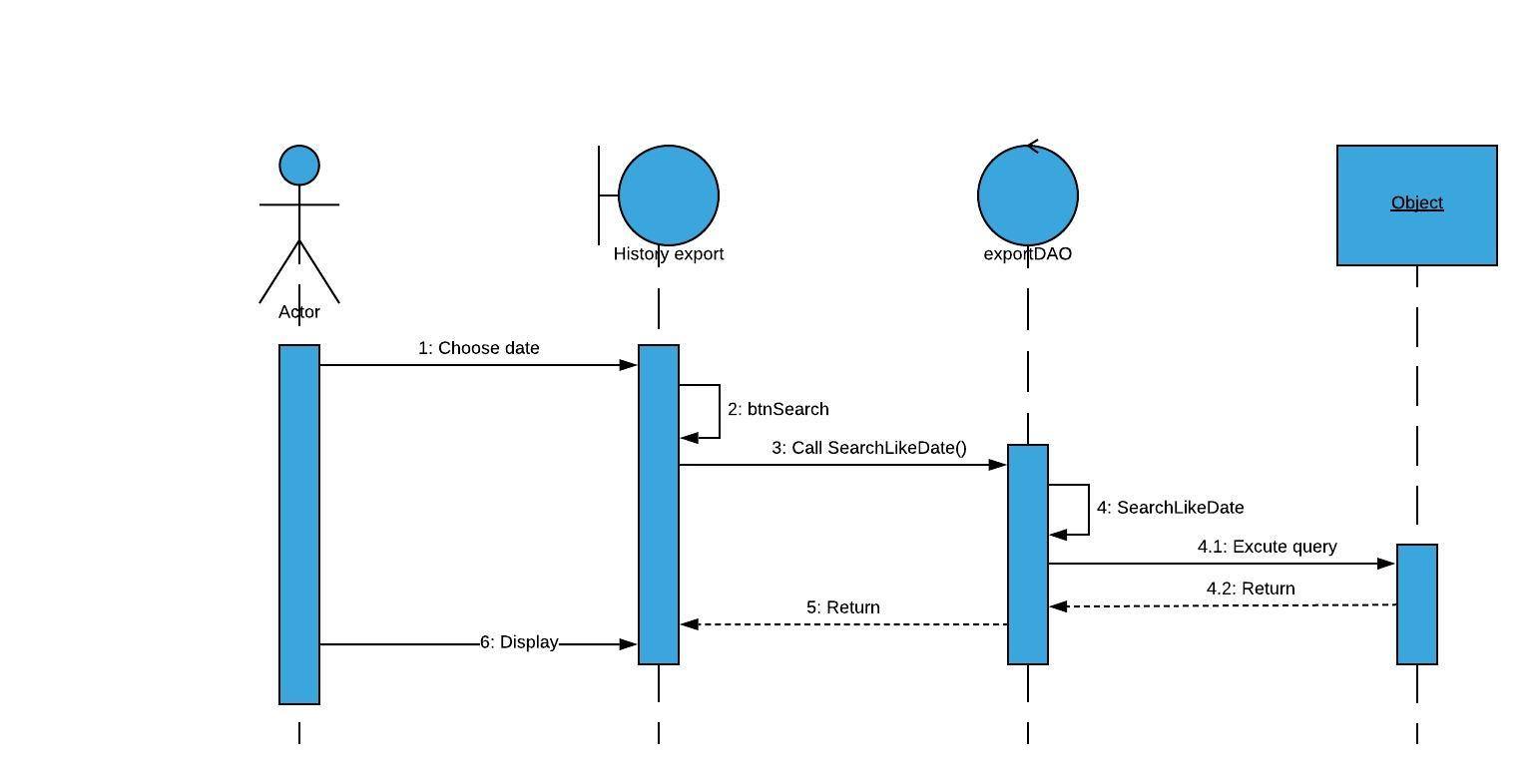


**3, Export**

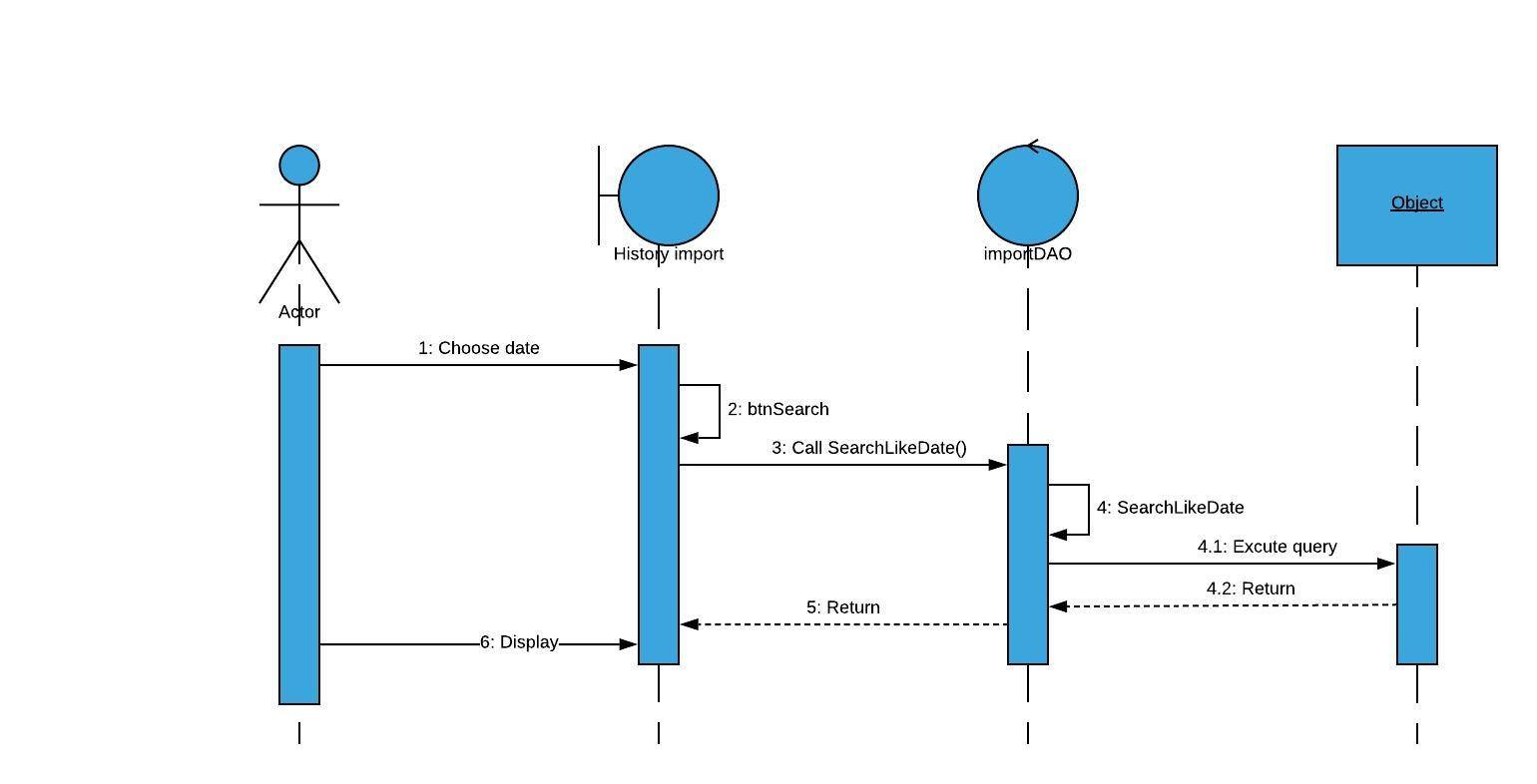


**4, Manage product**

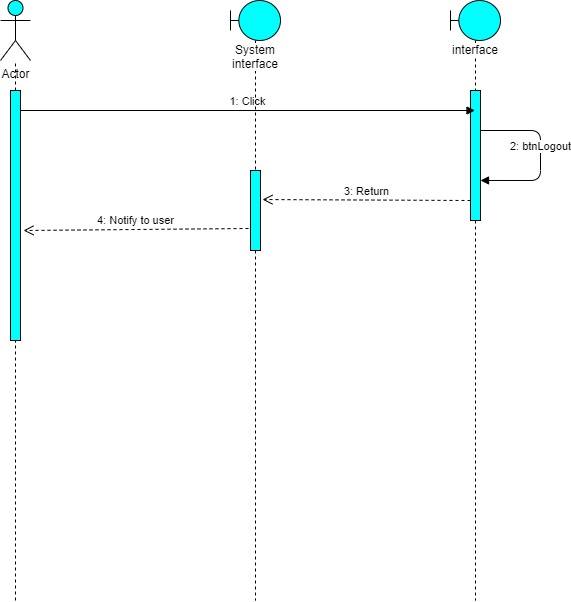
**5, View history export**

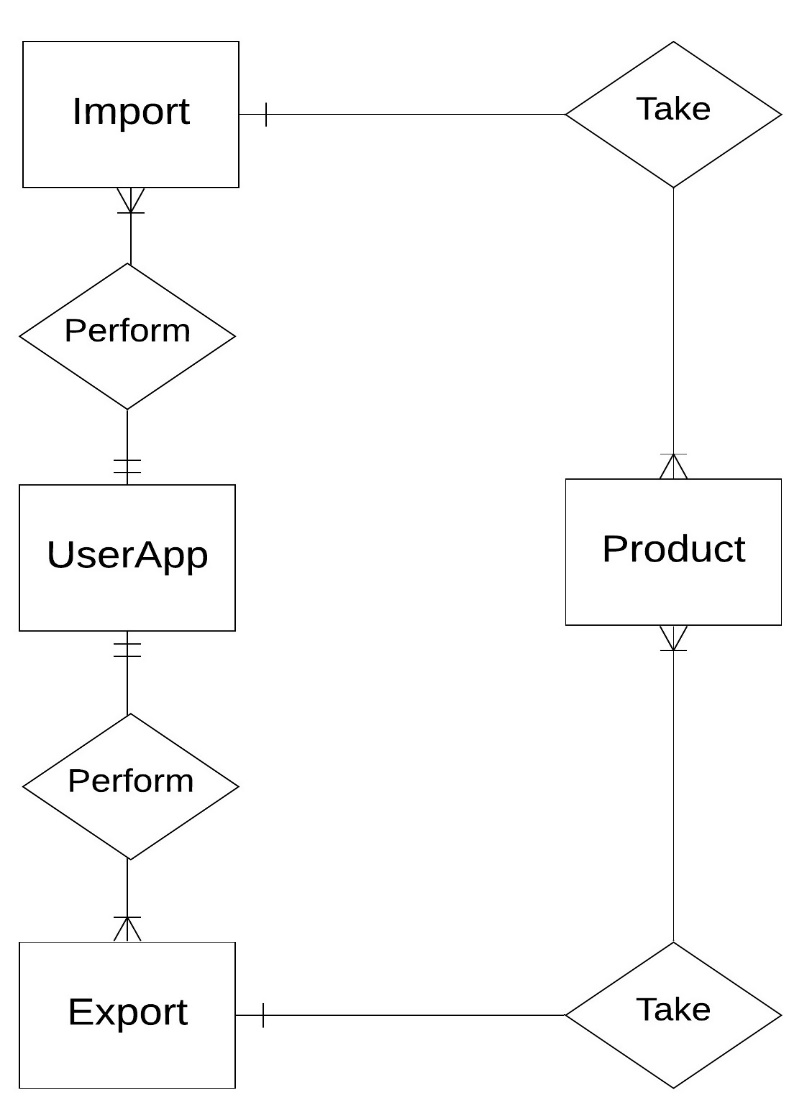


**6, View history import**



**6, log out**



**ENTITY RELATIONSHIP DIAGRAM (ERD)** **Note:  
  
  
ENTITY DESCRIPTION**

One to  
 many

Entity

Entity Relationship

only one to  
 one or many

**1.UserApp:**

**Account  
Customers**

**2.Product:**

**Product**

**3. Export Product :**

**Export**

**5. Import:**

**Import**

**TABLE DESIGN**

1. **UserApp:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | **🗝**Username | Nchar | 50 | Not Null | User ID |
| 2 | Passwords | Nchar | 50 | Null | User’s password |
| 3 | Status | Bit |  | Null | User’s user name |

1. **Product:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | **🗝**ProductID | Nchar | 10 | Not Null | Product’s ID |
| 2 | ProductName | Nvarchar | 150 | Null | Product’s name |
| 3 | Description | Nvarchar | 150 | Null | Product’s Description |
| 4 | PackageID | Nchar | 50 | Null | Package’s ID |
| 5 | NumbersPackage | Int |  | Null | Quantity of package |
| 6 | PriceOnePackage | Float |  | Null | Import price |
| 7 | PriceToExportPackage | Float |  | Null | Package’s Price |
| 8 | SubPackage | Nvarchar | 50 | Null | Package’ |
| 9 | NumberSubPackage | Int |  | Null | Quantity of sub in a package |
| 10 | CountSubExport | Int |  | Null | Quantity of sub in a package is sold |
| 11 | PriceToExportSub | Float |  | Null | Retail price |
| 12 | Status | Bit |  | Null | Delete incomplete products from the database |

1. **Export:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | ExportID | Nchar | 50 | Null |  |
| 2 | ProductID | Varchar | 30 | Null | Product’s ID |
| 3 | DateExport | Datetime |  | Null | Export date |

1. **Import:**

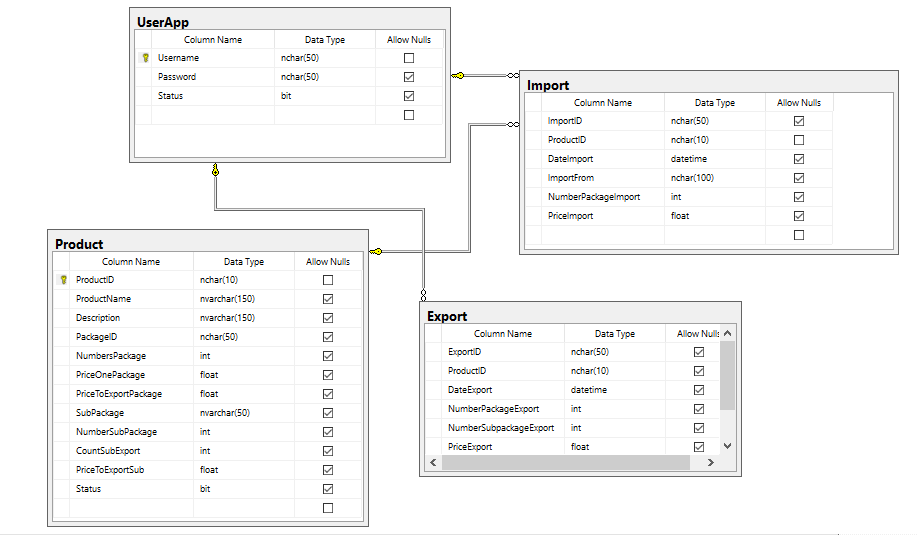
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | ImportID | Nchar | 50 | Null |  |
| 2 | ProductID | Nchar | 10 | Null | Product’s ID |
| 3 | DateImport | Datetime |  | Null | Import date |
| 4 | ImportFrom | Nchar | 100 | Null | Supplier |
| 5 | NumberPackageImport | Int |  | Null | Quantity of package |

**Note**

**🗝** Primary key

✓ Unique constraint

**TABLE RELATIONSHIP DIAGRAM**



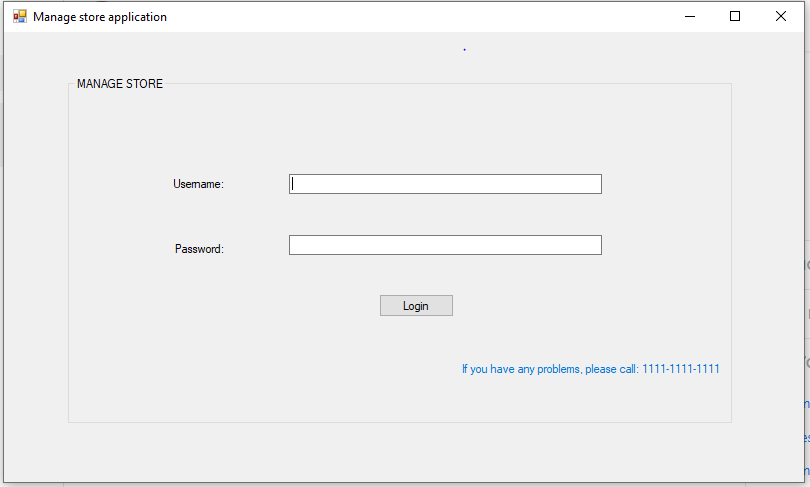
|  |  |  |
| --- | --- | --- |
| **Date: Sep 20, 2012** | **Prepared by Project Group 4**  **Team Leader:** | **Approved by**  **Mr. Ho Hoan Kiem** |
| **Signature** |  |  |

REVIEW 3

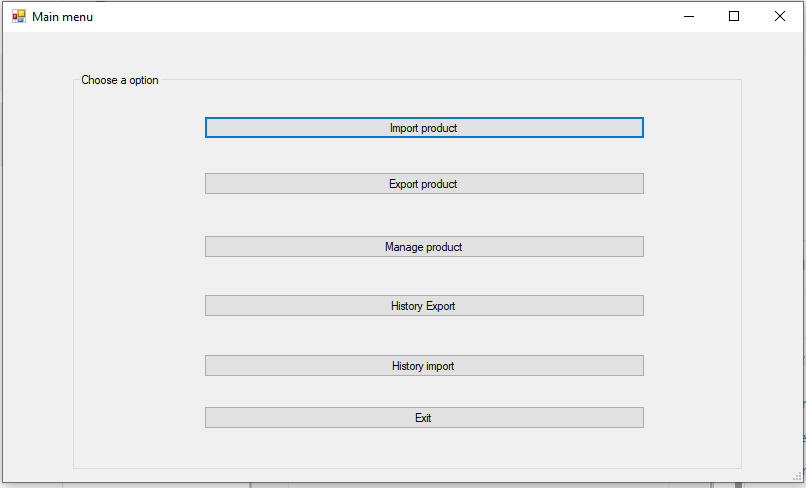
**INTERFACE DESIGN**

1. **Login screen**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| txtUsername | TextBox |  | String, not null | Input Customer’s user name |  |
| txtPassword | TextBox |  | String, not null, | Input Customer’s password |  |
| btnLogin | Button |  | Click | Choose Login |  |

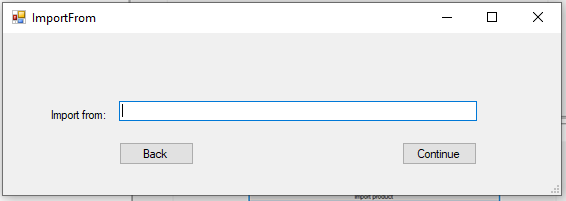


1. **Function Screen:**



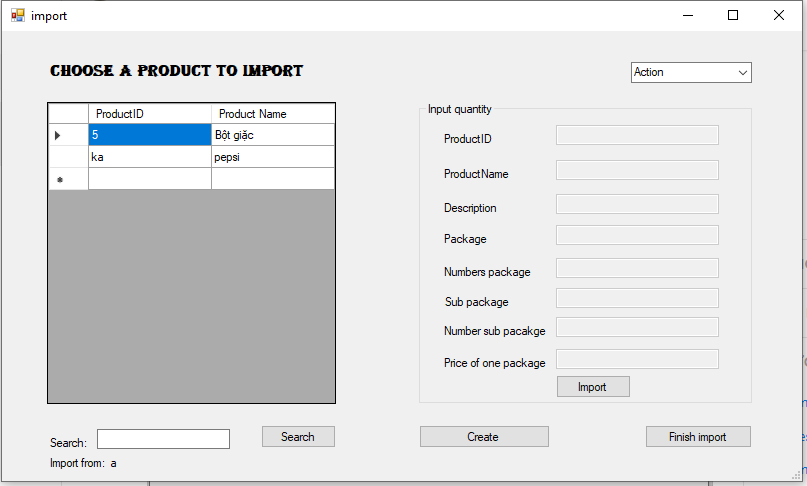
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Description** | **Status** |
| btnImport | Button | Click | Import function |  |
| btnExport | Button | Click | Export function |  |
| btnManage | Button | Click | Manage product function |  |
| btnHistoryExport | Button | Click | View history export function |  |
| btnHistoryImport | Button | Click | View history inport function |  |
| btnExist | Button | Click | Exit program |  |

**3. Import from screen:**



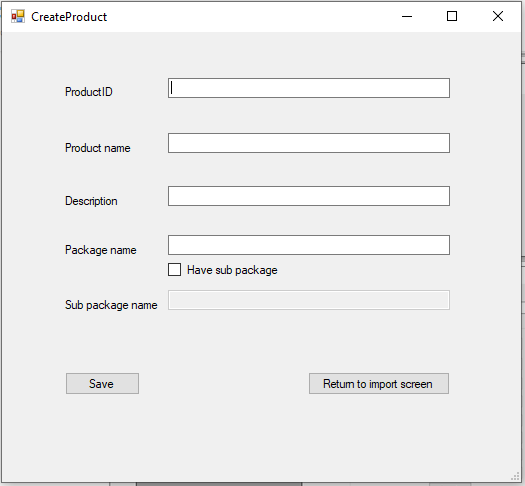
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| txtImportFrom | TextBox | Validating | String, not null | Source of goods |  |
| btnBack | Button | Click |  | Back to the previous screen |  |
| btnContinue | Button | Click |  | Continue to import |  |

**4.Import screen:**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| dgvListProduct | DataGridView | Click |  | List of products |  |
| cbAction | ComboBox | SelectedIndexChanged |  |  |  |
| txtProductID | TextBox |  | String, not null | Input product’s ID |  |
| txtProductName | TextBox |  | String, not null | Input product’s name |  |
| txtDescription | TextBox |  | String, not null | Input product’s description |  |
| txtPackage | TextBox |  | String, not null | Input product’s package |  |
| txtNumberPacakge | TextBox | Validating | int, not null, greater than 0 | Quantity of package |  |
| txtSubPackage | TextBox |  | String | Name of sub in package |  |
| txtNumberSub | TextBox | Validating | int, not null, greater than 0 | Quantity of sub in a package |  |
| txtPrice | TextBox | Validating | float, not null,greater than 0 | Price of a package |  |
| btnImport | Button | Click |  | Add product into stock |  |
| btnCreate | Button | Click |  | Create products that are not in stock |  |
| btnFinish | Button | Click |  | Finish import |  |
| btnSearch | Button | Click |  | Find product by ID |  |
| txtSearch | TextBox |  |  | Product’s ID need to search |  |

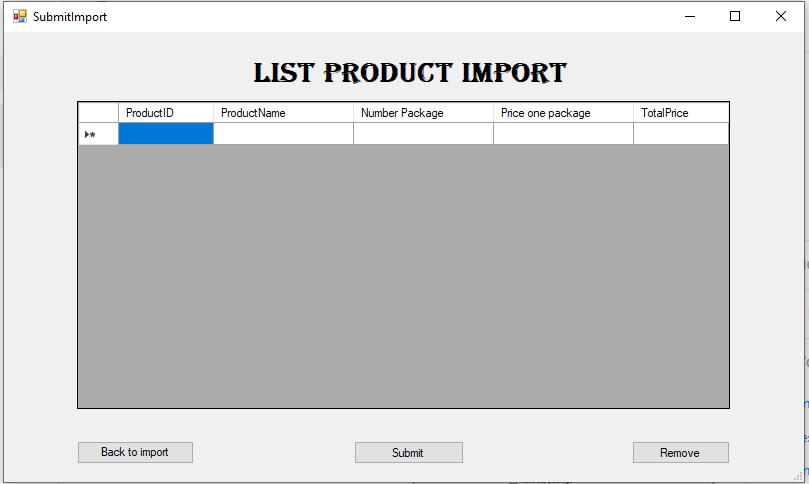
1. **Create product screen:**



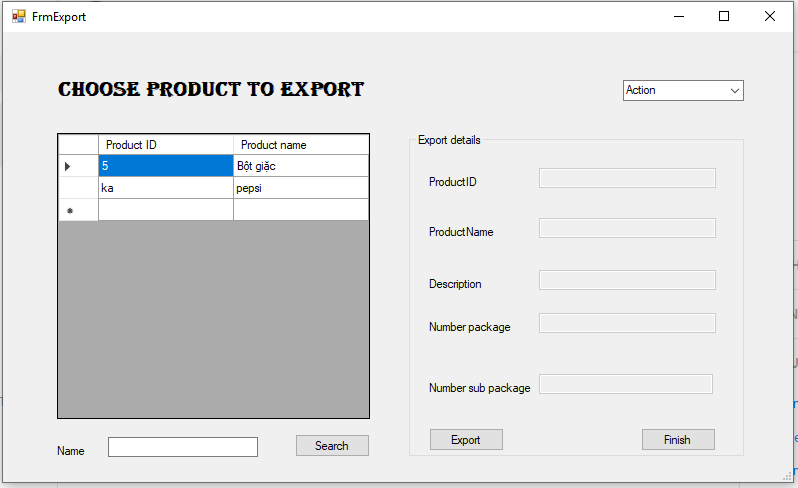
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| txtProductID | TextBox | Validating | String, not null | Input product’s ID |  |
| txtProductName | TextBox | Validating | String, not null, | Input product’s name |  |
| txtDescription | TextBox | Validating | String,not null | Input product’s description |  |
| txtPackage | TextBox | Validating | String,not null | Input product’s package |  |
| txtSubpacakge | TextBox | Validating | String,not null | Name of sub in package |  |
| chCheckSub | CheckBox | Click |  | have sub or not |  |
| btnSave | Button | Click |  | Save |  |
| btnReturn | Button | Click |  | Back to import screen |  |

1. **List Prodcut Inport screen:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| dgvSubmit | DataGridView | Click |  | Display all product are imported and get productID |  |
| btnBack | Button | Click |  | Back to import |  |
| btnSubmit | Button | Click |  | Finish import |  |
| btnRemove | Button | Click |  | Remove product |  |

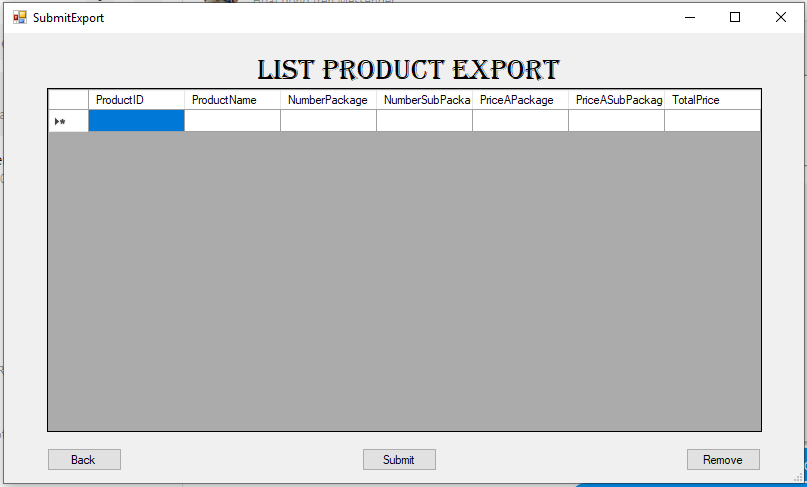


1. **Export Product screen:**



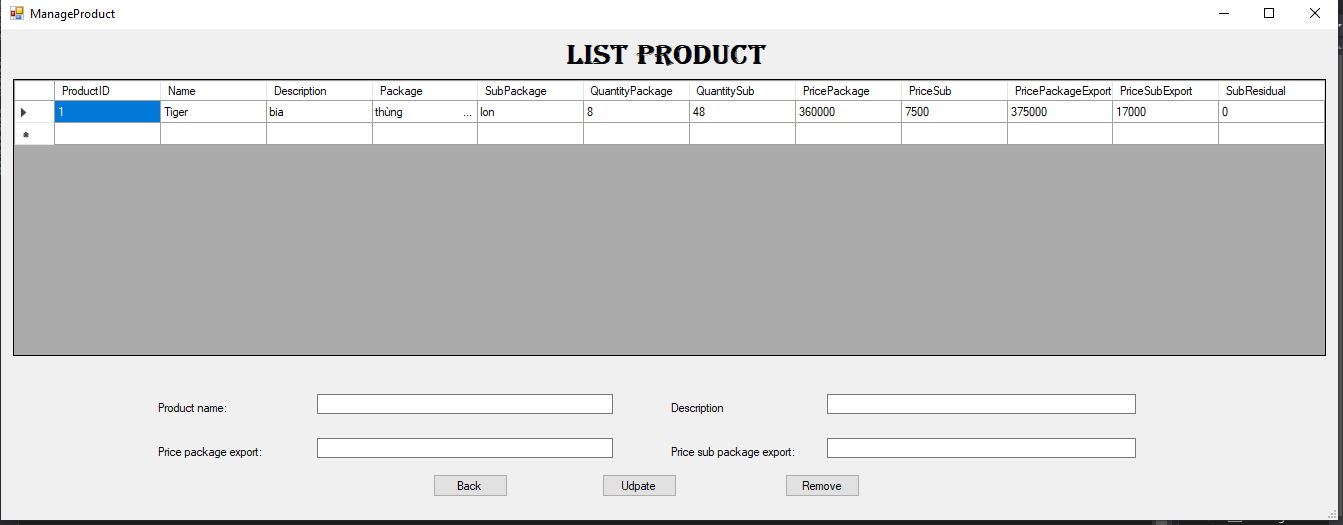
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| dgvExport | DataGridView | Click |  | Display product are exported |  |
| txtSearch | TextBox |  |  | Product’s ID need to search |  |
| btnSearch | Button | Click |  | Search product by ID |  |
| cbAction | ComboBox | SelectedIndexChanged |  | Back to function screen or log out |  |
| txtProductID | TextBox |  |  | Input product’s ID |  |
| txtProductName | TextBox |  |  | Input product’s name |  |
| txtDesCription | TextBox |  |  | Input product’s description |  |
| txtNumberPackage | TextBox | validating | Int, not null, greater than 0 | Quantity of package |  |
| txtNumberSub | TextBox | validating | Int, not null, greater than 0 | Quantity of sub package |  |
| btnExport | Button | Click |  | Export |  |
| btnFinish | Button | Click |  | Finish export |  |

1. **List Product Export screen:**



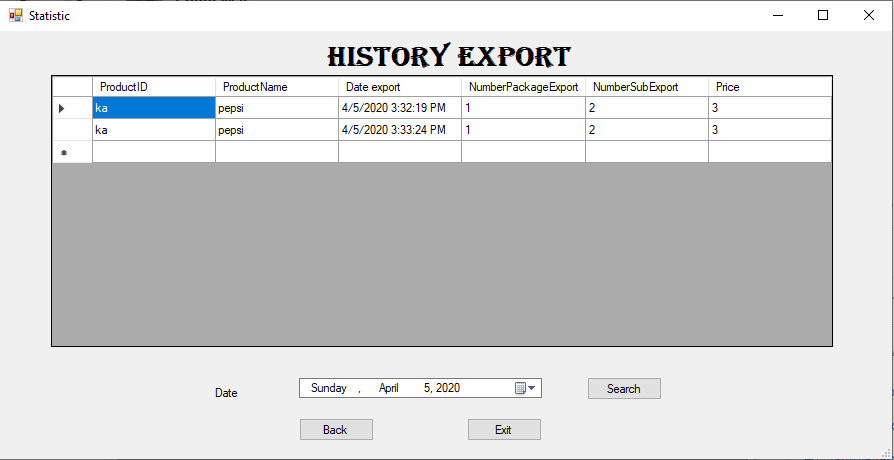
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| dgvSubmitExport | DataGridView | Click |  | Display products are exported and get productID |  |
| btnBack | Button | Click |  | Back to previous screen |  |
| btnSubmit | Button | Click |  | Finish export |  |
| btnRemove | Button | Click |  | Remove product |  |

**9.List product screen:**



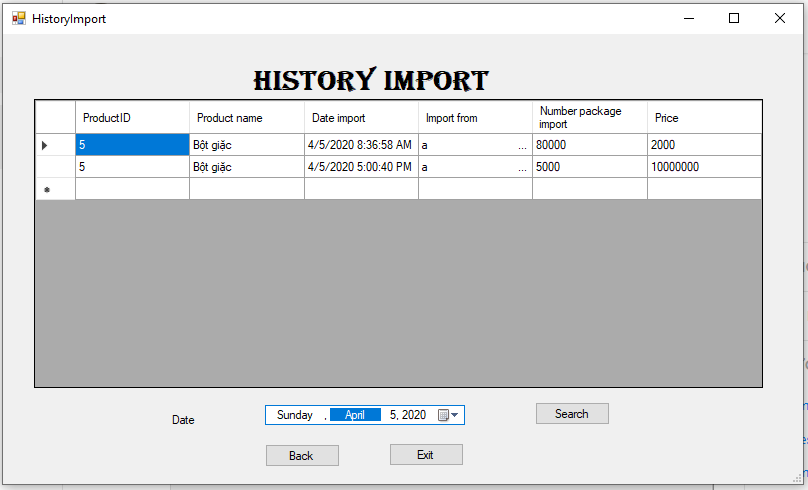
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| dgvView | DataGridView | click |  | Display all product  And get information of product |  |
| lbInfor | Label |  |  | Display ID of product |  |
| lbName | Label |  |  | Display title name |  |
| TxtProductName | Text |  | Not null | To input information about name of product |  |
| lbDescription | Label |  |  | Display title description |  |
| txtDescription | Text |  |  | To input description of product |  |
| lbPrice package export | label |  |  | Display title price package export: |  |
| txtPricePack | TextBox |  | Not null, greater than 0 | Input price package export |  |
| lbPriceSubExport | label |  |  | Display title price sub package export: |  |
| txtPriceSub | TextBox |  | Not null, greater than 0 | Input price sub package export |  |
| btnBack | Button | Click |  | Back to function screen |  |
| btnUpdate | Button | Click |  | Update price export of product.. |  |
| btnRemove | Button | Click |  | Remove product |  |

1. **History export screen:**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| dgvHistory | Datagirdview |  |  | Load history export |  |
| lbDate | Label |  |  | Display title to search |  |
| dtSearch | DateTimePicker |  | Date Time | Get date to search |  |
| btnSearch | Button | Click |  | Search export like date |  |
| btnBack | Button | Click |  | Back to function scrren |  |
| btnExit | Button | Click |  | Quit |  |

**11.History import screen:**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| dgvHistory | DataGirdView |  |  | Get history import |  |
| lbDate | Label |  |  | Display title search |  |
| dtSeacrh | DateTimePicker |  |  | Get search value |  |
| btnSearch | Button | click |  | Search history import |  |
| btnBack | Button | Click |  | Back to function screen |  |
| btnExit | Button | Click |  | Exist |  |

**TASK SHEET**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Group FFG** | | **Project Title**  **Laptop Shop Online** | | **Date of Preparation** | | **Activity Plan Prepared by**  **Team** |
| **No** | **Task Title** | | **Planned Start Date** | **Planned Completed Date** | **Team Mate Names** | **Status** |
| 1 | Login | | 04/04/2020 | 05/04/2020 |  |  |
| 2 | Import product | |  |
| 3 | Create product | |  |
| 4 | View history import | |  |
| 5 | Export product | |  |
| 6 | View history import | |  |  |
| 7 | Update product | |  |
| 8 |  | |  |
| 9 |  | |  |
| 10 |  | |  |
| 11 |  | |  |  |
| 12 |  | |  |
| 13 |  | |  |
| 14 |  | |  |
| 15 |  | |  |
| 16 |  | |  |  |
|  |
| 17 |  | |  |
| 18 |  | |  |
| 19 |  | |  |
| 20 |  | |  |  |
| 21 |  | |  |
| 22 |  | |  |
| 23 |  | |  |
| 25 |  | |  |
| 26 |  | |  |
| 27 |  | |  |

|  |  |  |
| --- | --- | --- |
| **Date: Oct 10, 2012** | **Prepared by Project Group 4**  **Team Leader:** | **Approved by**  **Mr. Ho Hoan Kiem** |
| **Signature** |  |  |