

**ONLINE SHOP**

**Software Design Document**

– Hanoi, May 2022 –

**Table of Contents**

Table of Contents

[Project Report 3](#_Toc106050031)

[1. Status Report 3](#_Toc106050032)

[2. Team Involvements 3](#_Toc106050033)

[I. Overview 4](#_Toc106050034)

[1. Code Packages/Namespaces 4](#_Toc106050035)

[2. Database Table 5](#_Toc106050036)

[II. Code Designs 6](#_Toc106050037)

[1. <Feature/Function Name> 6](#_Toc106050038)

[a. Class Diagram 7](#_Toc106050039)

[b. Class Specifications 7](#_Toc106050040)

[c. Sequence Diagram(s) 10](#_Toc106050041)

[2. Feature “Login” 11](#_Toc106050042)

[a. Class Diagram 11](#_Toc106050043)

[b. Sequence Diagram(s) 12](#_Toc106050044)

[3. Feature “SignUp” 13](#_Toc106050045)

[a. Class Diagram 13](#_Toc106050046)

[b. Sequence Diagram(s) 14](#_Toc106050047)

[4. Feature “Logout” 14](#_Toc106050048)

[b. Sequence Diagram(s) 16](#_Toc106050049)

[5. Feature “AddToCart” 16](#_Toc106050050)

[b. Sequence Diagram(s) 17](#_Toc106050051)

[6. Feature “View Product detail” 18](#_Toc106050052)

[a. Class Diagram 18](#_Toc106050053)

[b. Sequence Diagram(s) 18](#_Toc106050054)

[7. Feature “Home Page” 19](#_Toc106050055)

[a. Class Diagram 19](#_Toc106050056)

[b. Sequence Diagram(s) 19](#_Toc106050057)

[8. Feature “Slider” 20](#_Toc106050058)

[a. Class Diagram 20](#_Toc106050059)

[b. Sequence Diagram(s) 20](#_Toc106050060)

[9. Feature “Product List” 21](#_Toc106050061)

[a. Class Diagram 21](#_Toc106050062)

[b. Sequence Diagram(s) 21](#_Toc106050063)

[10. Feature “Cart Detail” 22](#_Toc106050064)

[a. Class Diagram 22](#_Toc106050065)

[b. Sequence Diagram(s) 22](#_Toc106050066)

[11. Feature “Cart Completion”(CheckOut) 23](#_Toc106050067)

[a. Class Diagram 23](#_Toc106050068)

[b. Sequence Diagram(s) 23](#_Toc106050069)

[12. Feature “Change Password” 24](#_Toc106050070)

[a. Class Diagram 24](#_Toc106050071)

[b. Sequence Diagram(s) 25](#_Toc106050072)

[13. Feature “User Profile (Admin Profile)” 25](#_Toc106050073)

[a. Class Diagram 25](#_Toc106050074)

[b. Sequence Diagram(s) 26](#_Toc106050075)

[III. Database Tables 26](#_Toc106050076)

[1. Order 26](#_Toc106050077)

[2. Order detail 27](#_Toc106050078)

[3. Blog 27](#_Toc106050079)

[4. Blog\_categories 28](#_Toc106050080)

[5. User account 28](#_Toc106050081)

[6. Authority 28](#_Toc106050082)

[7. Feedback 28](#_Toc106050083)

[8. Cart 29](#_Toc106050084)

[9. Cart\_item 29](#_Toc106050085)

[10. Post 30](#_Toc106050086)

[11. Post\_categories 30](#_Toc106050087)

[12. Product 30](#_Toc106050088)

[13. Product category 31](#_Toc106050089)

[14. Slide 31](#_Toc106050090)

[15. Status 32](#_Toc106050091)

[16. Product category 32](#_Toc106050092)

[17. Image 32](#_Toc106050093)

[18. Tool and Infrastructures 32](#_Toc106050094)

# Project Report

## 1. Status Report

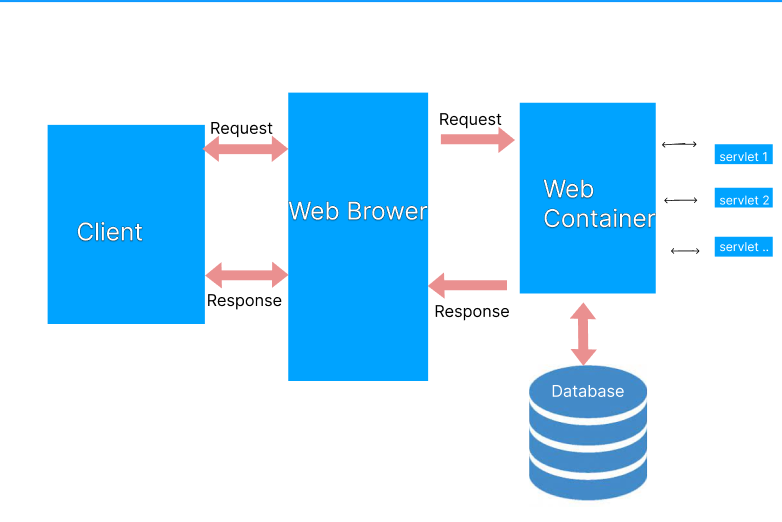
|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Work Item** | **Status** | **Notes (Work Item in Details)** |
| 1 | Overall Description | Completed |  |
| 2 | System Architecture Design | Completed |  |
| 3 | System Detailed Design | Completed |  |
| 4 | Object Specifications | Completed |  |
| 5 | Data & Database Design | Completed |  |

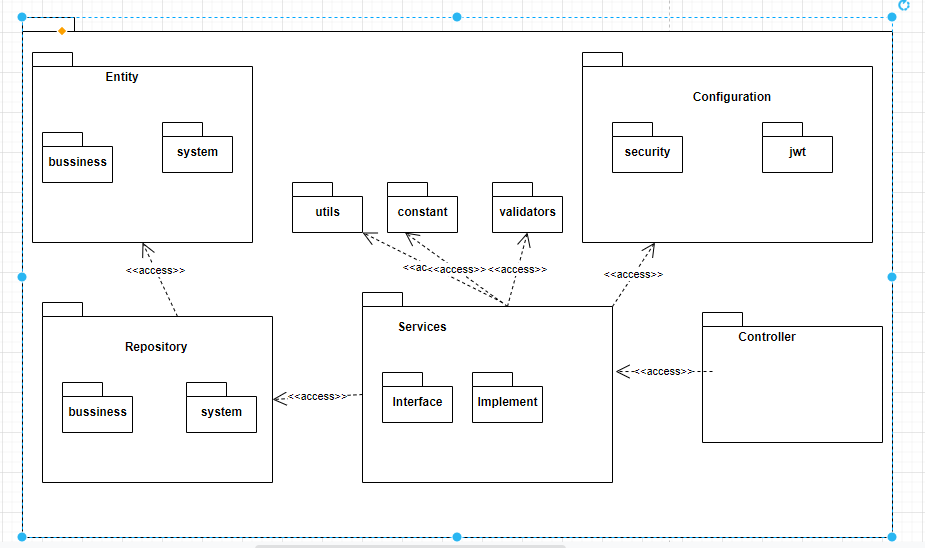
## 2. Team Involvements

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Task** | **Member** | **Notes (Task Details, etc.)** |
| 1 | Overall Description | DucLM |  |
| 2 | System Architecture Design | DucLM+HungNM |  |
| 3 | System Detailed Design | DucLM+ThachDP |  |
| 4 | Object Specifications | HoangNV+VoNVA |  |
| 5 | Data & Database Design | HoangNV+HungNM |  |

# I. Overview

## 1. Overall Architeture

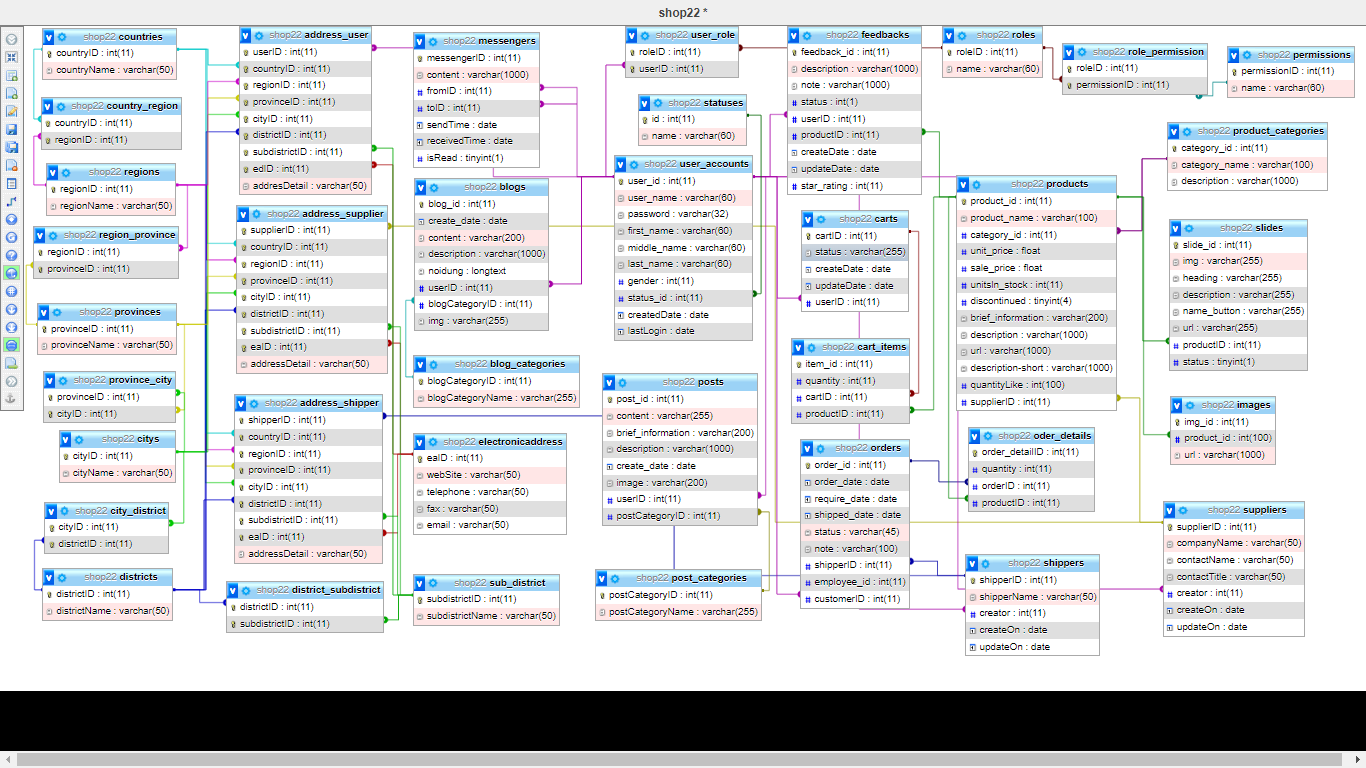


Package Diagram

***Package descriptions & package class naming conventions***

|  |  |  |
| --- | --- | --- |
| **No** | **Package** | **Description** |
| 1 | entity | The entities package contains entity classes that are the fundamental building blocks  of systems that an entity is a collection of fields and associated database operations.  They correspond to database tables and can have attributes, operations,  dependencies, inherited relations, and aggregations. |
| 2 | configurations | The Configuration package contains the necessary configuration of the easyStay  system. It helps to set up an environment for functionalities with extendable  reliability. |
| 3 | validators | The validator package contains classes that are responsible for validators information |
| 4 | controller | Controllers are responsible for controlling the flow of the application execution. When  a user makes a request to easyStay application, a controller is responsible for  returning the response to that request |
| 5 | service | Service classes can query multiple Repository classes and combine their data to form  new, more complex business objects. Further, they introduce a layer of abstraction  between the web application and the Repository so that they can change more  independently. |
| 6 | repository | The repositories package contains classes that handle getting data into and out of the  data store, with the important caveat that each Repository only works against a single  Entity class. |
| 7 | utils | Contains classes that provide support processing the system development. |
| 8 | constants | Contains all defined variables of easyStay system |

## 2. Database Table



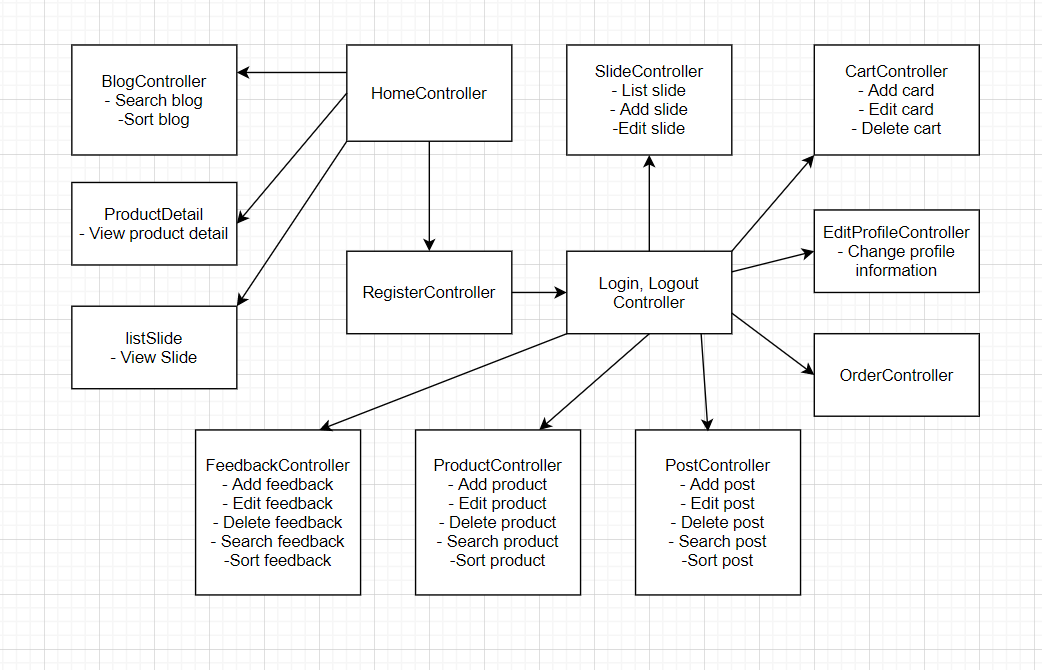
***Table descriptions & package class naming conventions are as below***

|  |  |  |
| --- | --- | --- |
| **No** | **Table** | **Description** |
| 01 | authority | - Primary keys: id |
| 02 | blog | - Primary keys: blog\_id  - Foreign keys: category\_blog\_id  - Foreign keys: User\_Account\_user\_id |
| 03 | blog\_categories | - Primary keys: order\_id |
| 04 | cart | - Primary keys: user\_id  - Foreign keys: User\_Account\_user\_id |
| 05 | cart\_item | - Primary keys: user\_id  - Foreign keys: Cart\_user\_id  - Foreign keys: Products\_product\_id |
| 06 | feedback | - Primary keys: feedback\_id  - Foreign keys: User\_Account\_user\_id  - Foreign keys: products\_product\_id |
| 07 | image | - Primary keys: img\_id  - Foreign keys: product\_id |
| 08 | order\_detail | - Primary keys: order\_id  - Foreign keys: Order\_order\_id, Products\_product\_id |
| 09 | order | - Primary keys: order\_id  - Foreign keys: User\_Account\_user\_id |
| 10 | post | - Primary keys: post\_id  - Foreign keys: User\_Account\_user\_id |
| 11 | post\_categories | - Primary keys: post\_id  - Foreign keys: post\_id |
| 12 | product | - Primary keys: product\_id  - Foreign keys: category\_id |
| 13 | product\_categories | - Primary keys: category\_id |
| 14 | slide | - Primary keys: slide\_id |
| 15 | status | - Primary keys: id |
| 16 | user\_account | - Primary keys: user\_id  - Foreign keys: status\_id , authority\_id |
| 17 | Colors | - Primary keys: colorID |
| 18 | district | - Primary keys: id  - Foreign keys: \_province\_id |
| 19 | electronicaddress | - Primary keys: eaID |
| 20 | favorite\_product | - Primary keys: productID,userID  - Foreign keys: productID,userID |
| 21 | group | - Primary keys: groupID |
| 22 | messages | - Primary keys: messageID  - Foreign keys: messageID,creatorID |
| 23 | message\_recipient | - Primary keys: mrID  - Foreign keys: messageID,recipientGroupID,recipientID |
| 24 | permissions | - Primary keys: permissionID |
| 25 | product\_color | - Primary keys: colorID,productID  - Foreign keys: colorID,productID |
| 26 | project | - Primary keys: id  - Foreign keys: \_province\_id,district\_id |
| 27 | province | - Primary keys: id |
| 28 | roles | - Primary keys: roleID |
| 29 | role\_permission | - Primary keys: roleID,permissionID  - Foreign keys: roleID,permissionID |
| 30 | shipper | - Primary keys: shipperID |
| 31 | street | - Primary keys: id  - Foreign keys: : \_province\_id,district\_id |
| 32 | supplierID | - Primary keys: supplierID |
| 33 | Supplier\_address | - Primary keys: saID  - Foreign keys: provinceID,districtID,wardID,projectID,eaID |
| 34 | user\_address | - Primary keys: uaID  - Foreign keys: provinceID,districtID,wardID,projectID,eaID |
| 35 | User\_group | - Primary keys: ugID  - Foreign keys: userID,groupID |
| 36 | User\_role | - Primary keys: userID,roleID  - Foreign keys: userID,roleID |
| 37 | ward | - Primary keys: id  - Foreign keys: \_province\_id,district\_id |

# II. Code Designs

## 1. <Feature/Function Name>

### a. Class Diagram

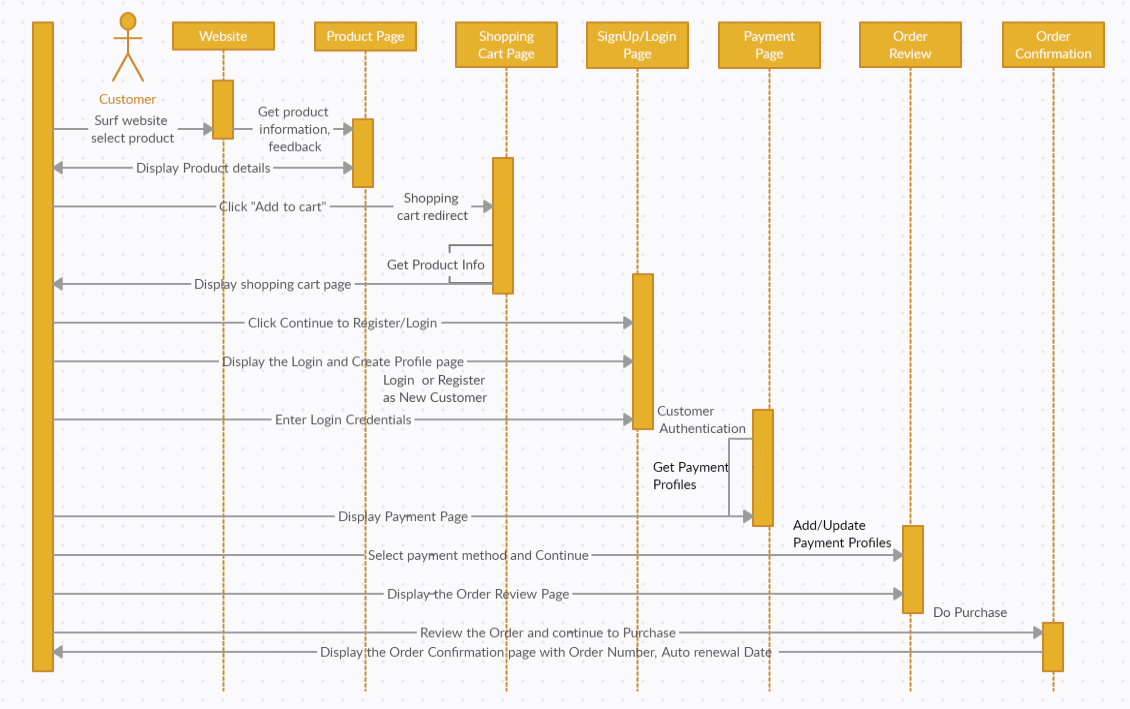


### b. Class Specifications

***Class Methods***

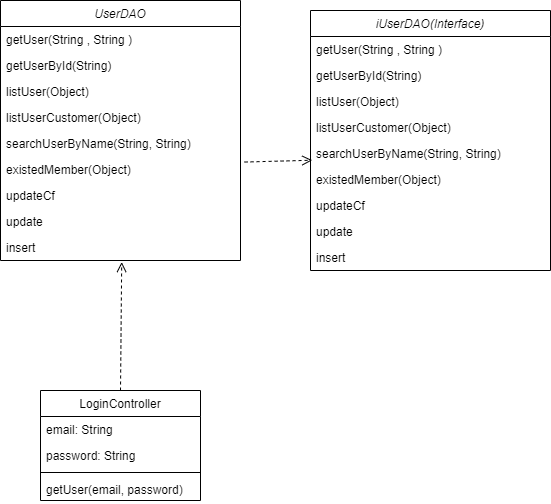
|  |  |
| --- | --- |
| Home controller | Shows slider , hot posts , products  - Shown slider information includes its image and title the user is redirected to the slider's backlink on his/her clicking  - Shown post information includes its thumbnail, title, brief-info. The user is redirected to the post's details on his/her clicking  - Shown product information includes its thumbnail, title, brief information. The user is redirected to the product's details on his/her clicking |
| Register controller | This is a pop-up screen which allows the user to register himself/herself to the system by input following information: full name, gender, email, mobile, and address; User then need to verify by clicking the link sent via email to him/her before being able to access the system |
| Login, log out controller | This is a pop-up screen which allows the user to enter email & password to login; on this page, there are also options for user to register new information or reset the password for the case he/she forget it |
| Blog controller | Show the posts pagination (sorted by updated date, paginated) + the slide with the post search box, post categories, the latest posts, and static contacts/links; Shown post information includes its thumbnail, title, brief-info; the user is redirected to the post's details on his/her clicking |
| Product detail | Show product details (including title, category, images, original price, sale price, and product details) + the sider with the product search box, product categories, the latest products, and static contacts/links |
| List slide | Show detailed slider information (image, title, backlink, status, notes) |
| Slide controller | Show the paginated list of sliders (including sliders' id, title, image, backlink, status)  - The user can filter the sliders by the status + search by title or backlink  - For each slider, the user can choose to hide, show, edit it |
| Cart controller | Show the cart details with list of the user's chosen products (including Id, title, price, quantity, total cost) and the total order price + the sider with the product search box, product categories, the latest products, and static contacts/links; The user can  - Change the quantity of each product: the total cost is automatically adjusted  - Choose to delete/remove product by clicking on the deleting icon  - Click the Choose More Product button to go to Products List page  - Click the Check Out button to go to the Cart Contact page  Please note: the user's chosen products from the last browsing session remain in his/her cart until he/she submits the cart. However, the product prices are updated with the latest prices in the system |
| Edit profile controller | This is a pop-up screen which allows the user to edit his/her profile information, including the registered ones & the avatar image. Please note that s/he is not allowed to change the email |
| Order controller | Show the order details, including the following information:  - Basic order information: order id, customer full name, email, mobile, order date, total cost, sale name, status  - The receiver information: full name, gender, email, mobile, address  - This list of ordered products, each product include following information: thumbnail, name, category, unit price, quantity, total cost |
| Post controller | Show detailed post information (thumbnail, category, title, brief information, description, flag to turn the featurning on/off, status), from that allow the user to input, view or edit them |
| Product controller | Show the paginated list of products (include products' id, thumbnail, title, category, list price, sale price, featured, and status information):  - The user can filter the product by category, status or search by title, brief info  - Sortable by title, category, list price, sale price, featured, or status  - From each product, the user can choose to hide, show, view, edit it  - The page also have the button/link that allows the user to add new product |
| Feedback controller | On receiving the bought products successfully, the receiver would receive a confirmation email with the guides to feedback on each of the received product; Beside, the user/receive could also give the feedback (on a specific product or general feedback) via the website; The user also call this screen from the system menu on the header;  The feedback information includes contact information (full name, gender, email, mobile), rated star, the attached images and the contact's free-format feedback  The page sider is also shown with the product search box, product categories, static contacts/links |

### c. Sequence Diagram(s)

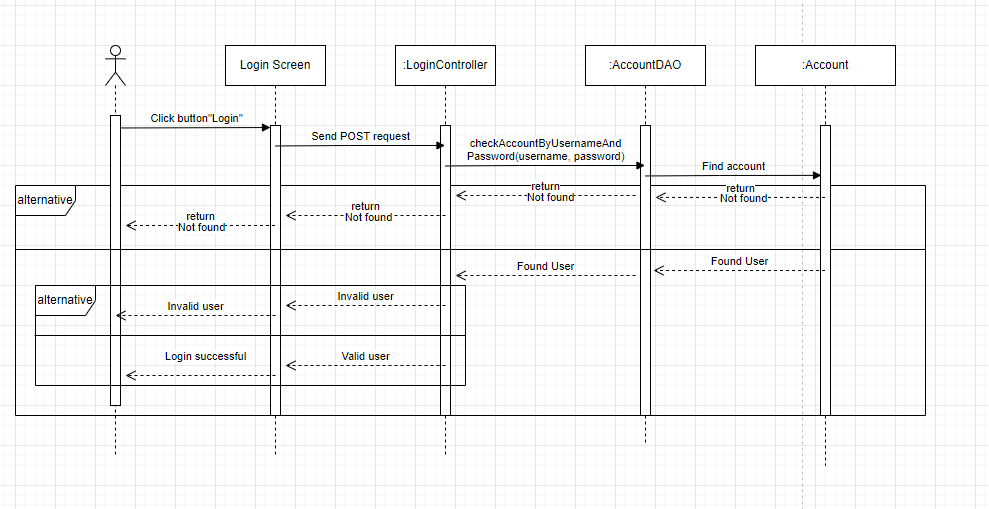


## 2. Feature “Login”

### a. Class Diagram

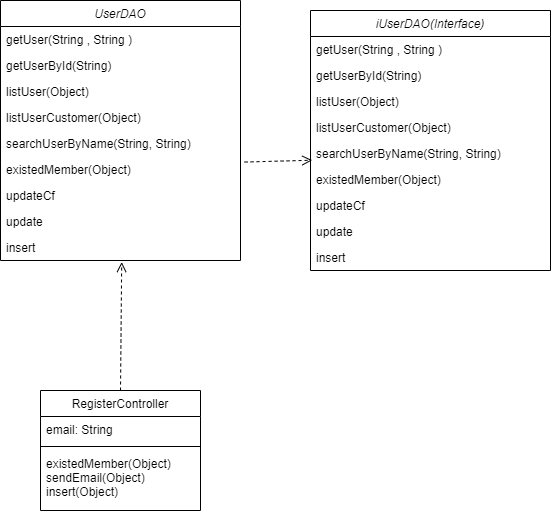


### b. Sequence Diagram(s)

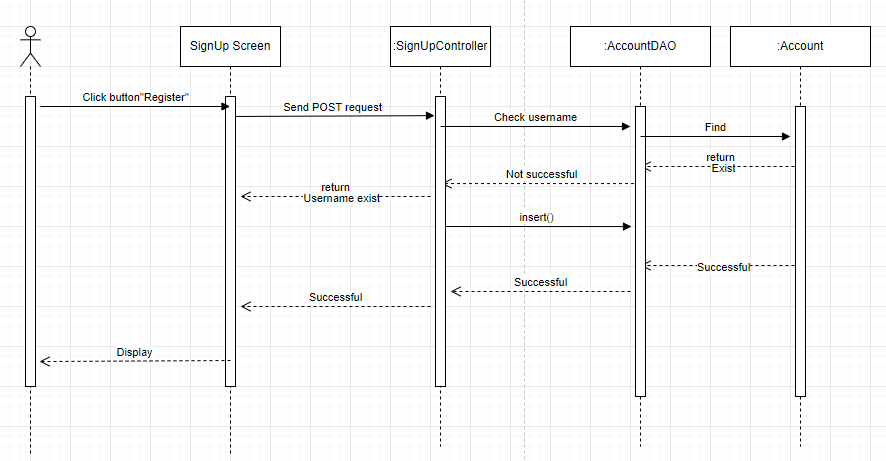


## 3. Feature “SignUp”

### a. Class Diagram

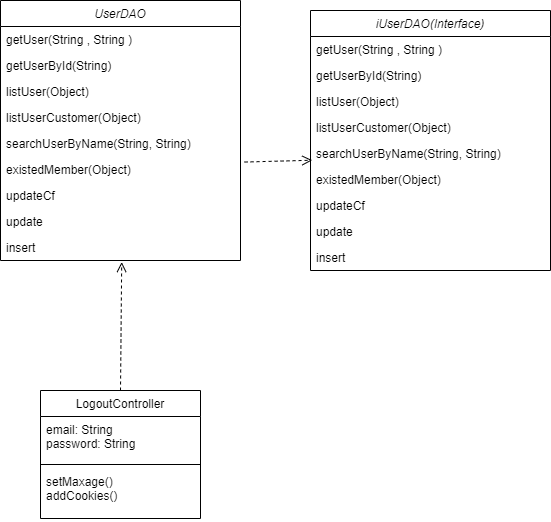


### b. Sequence Diagram(s)

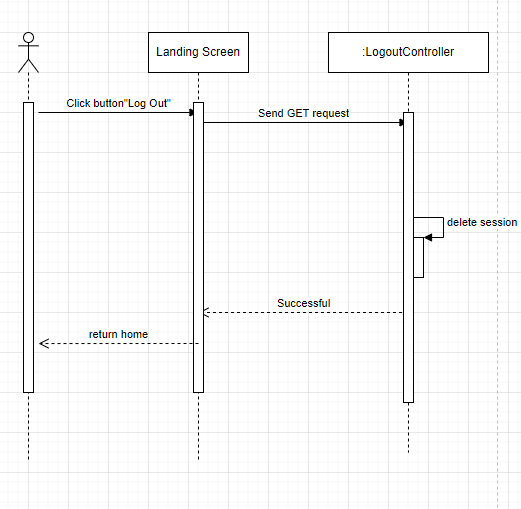


## 4. Feature “Logout”

**a. Class Diagram**

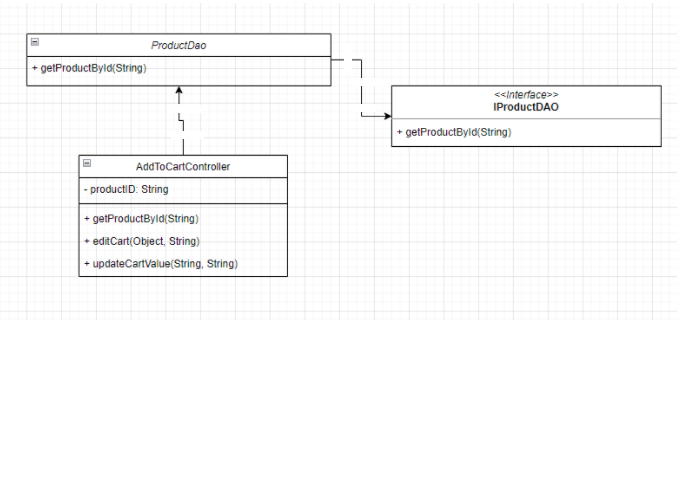


### b. Sequence Diagram(s)

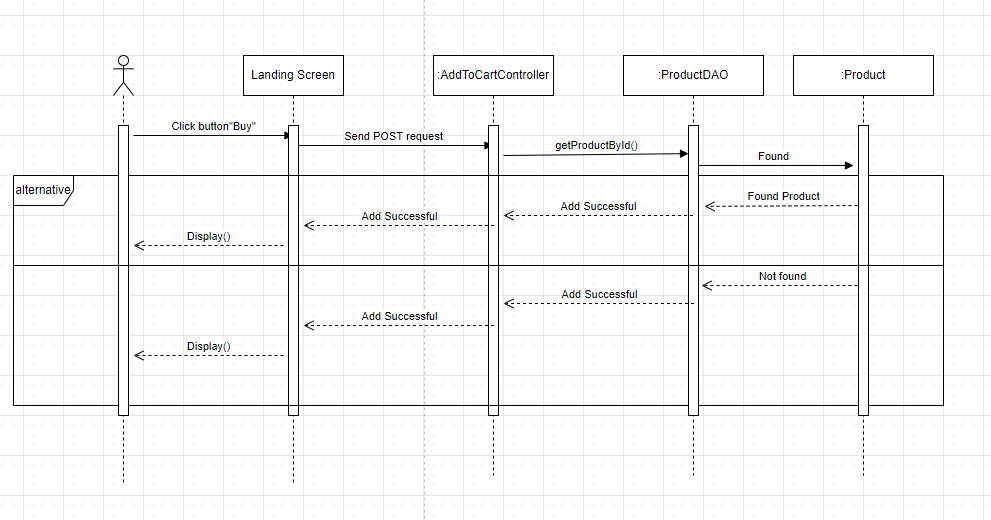


## 5. Feature “AddToCart”

**a. Class Diagram**

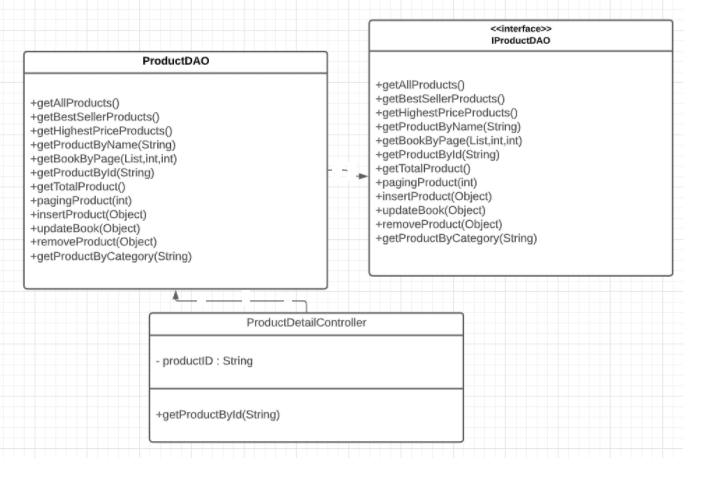


### b. Sequence Diagram(s)

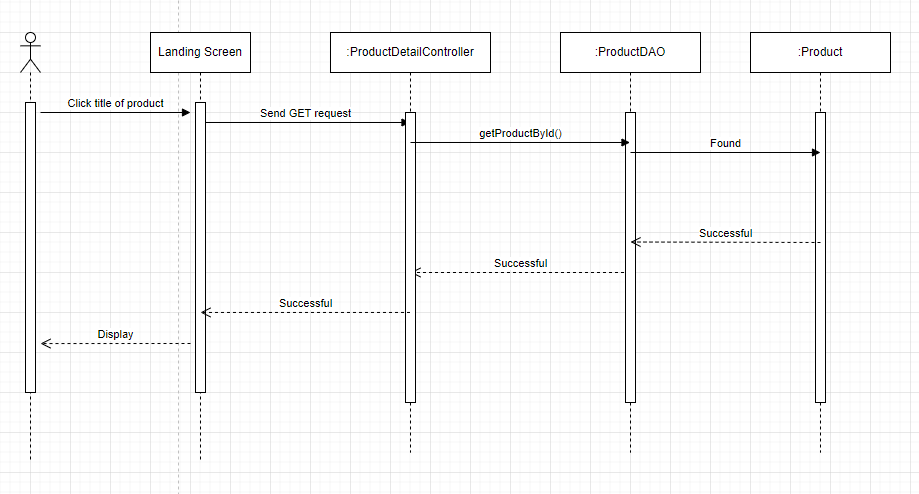


## 6. Feature “View Product detail”

### a. Class Diagram

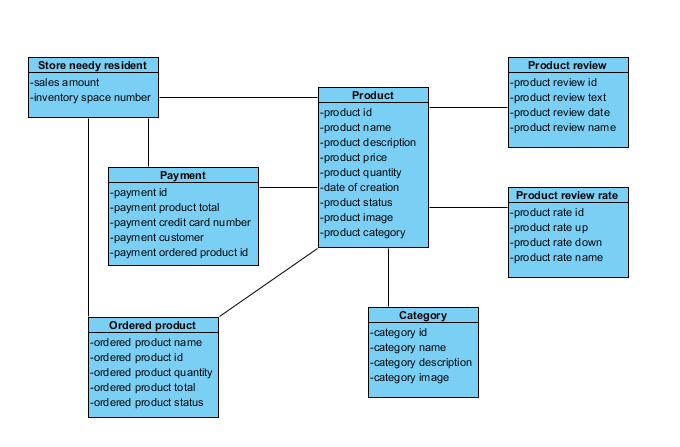


### b. Sequence Diagram(s)

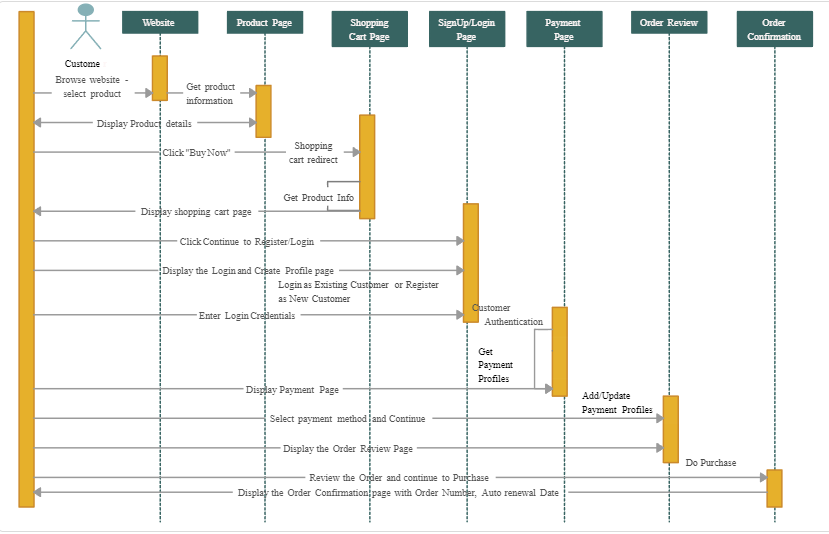


## 7. Feature “Home Page”

### a. Class Diagram

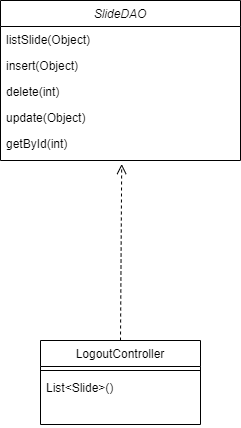


### b. Sequence Diagram(s)

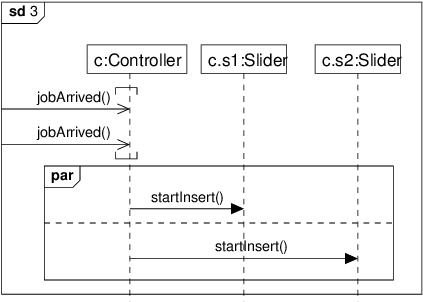


## 8. Feature “Slider”

### a. Class Diagram

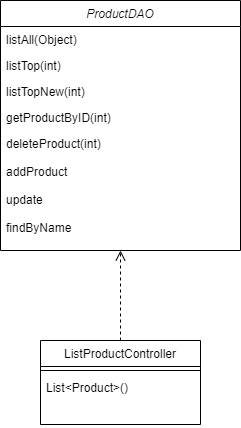


### b. Sequence Diagram(s)

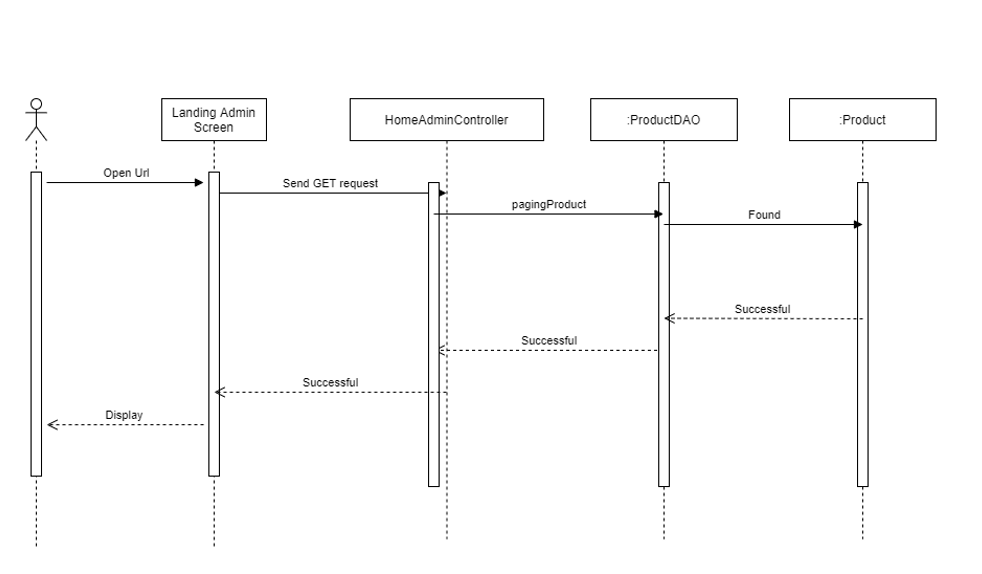


## 9. Feature “Product List”

### a. Class Diagram

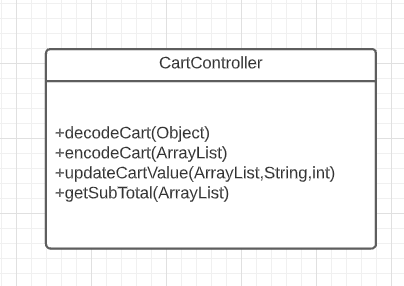


### b. Sequence Diagram(s)

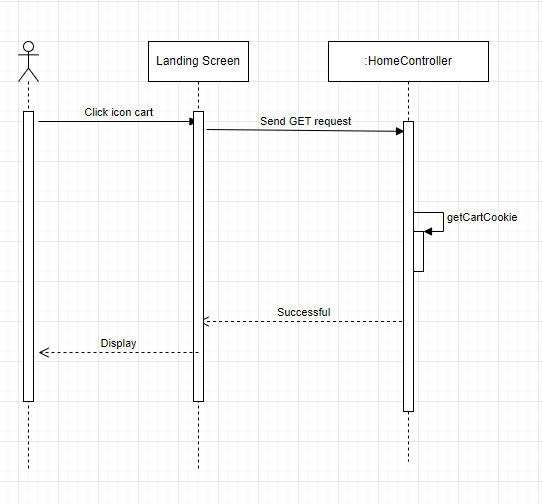


## 10. Feature “Cart Detail”

### a. Class Diagram

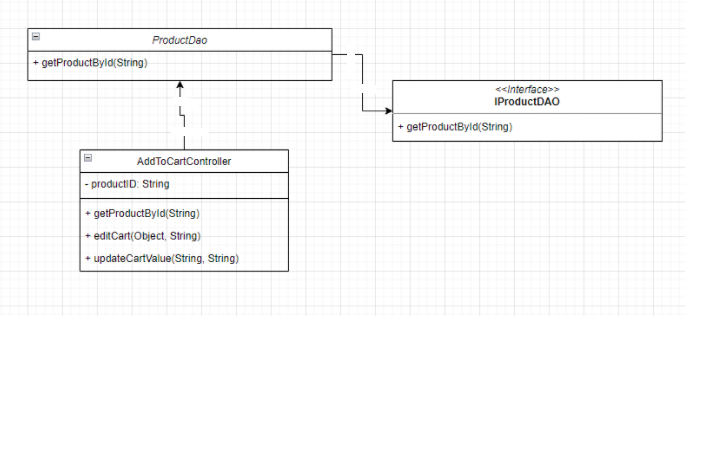


### b. Sequence Diagram(s)

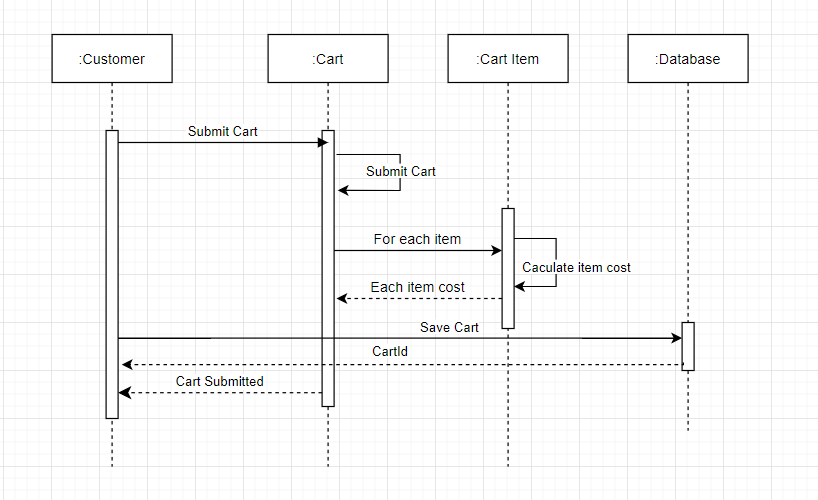


## 11. Feature “Cart Completion”(CheckOut)

### a. Class Diagram



### b. Sequence Diagram(s)



## 12. Feature “Change Password”

### a. Class Diagram

Diagram

Description automatically generated

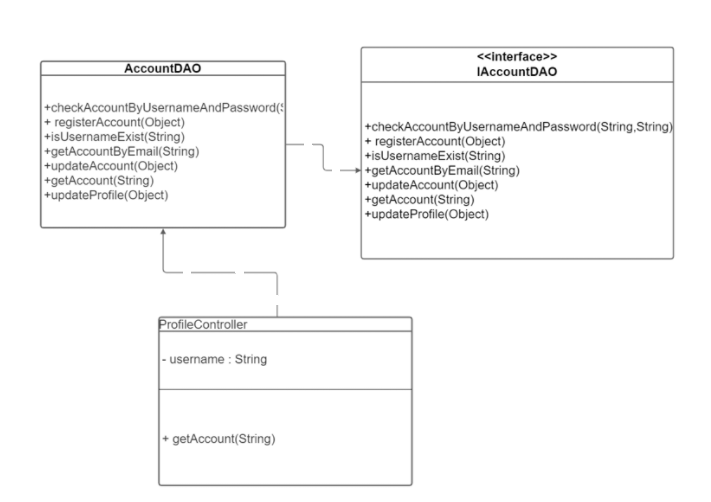
### b. Sequence Diagram(s)

Calendar

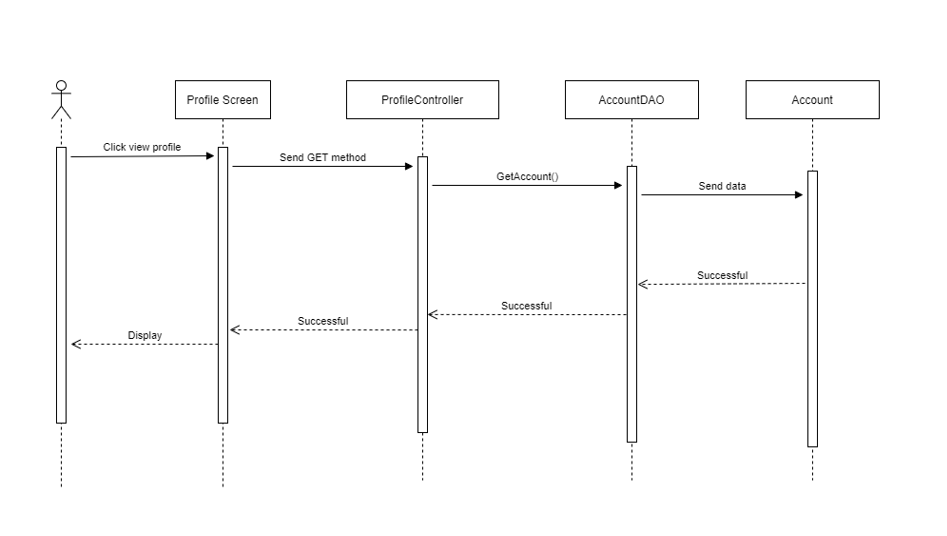
Description automatically generated

## 13. Feature “User Profile (Admin Profile)”

### a. Class Diagram



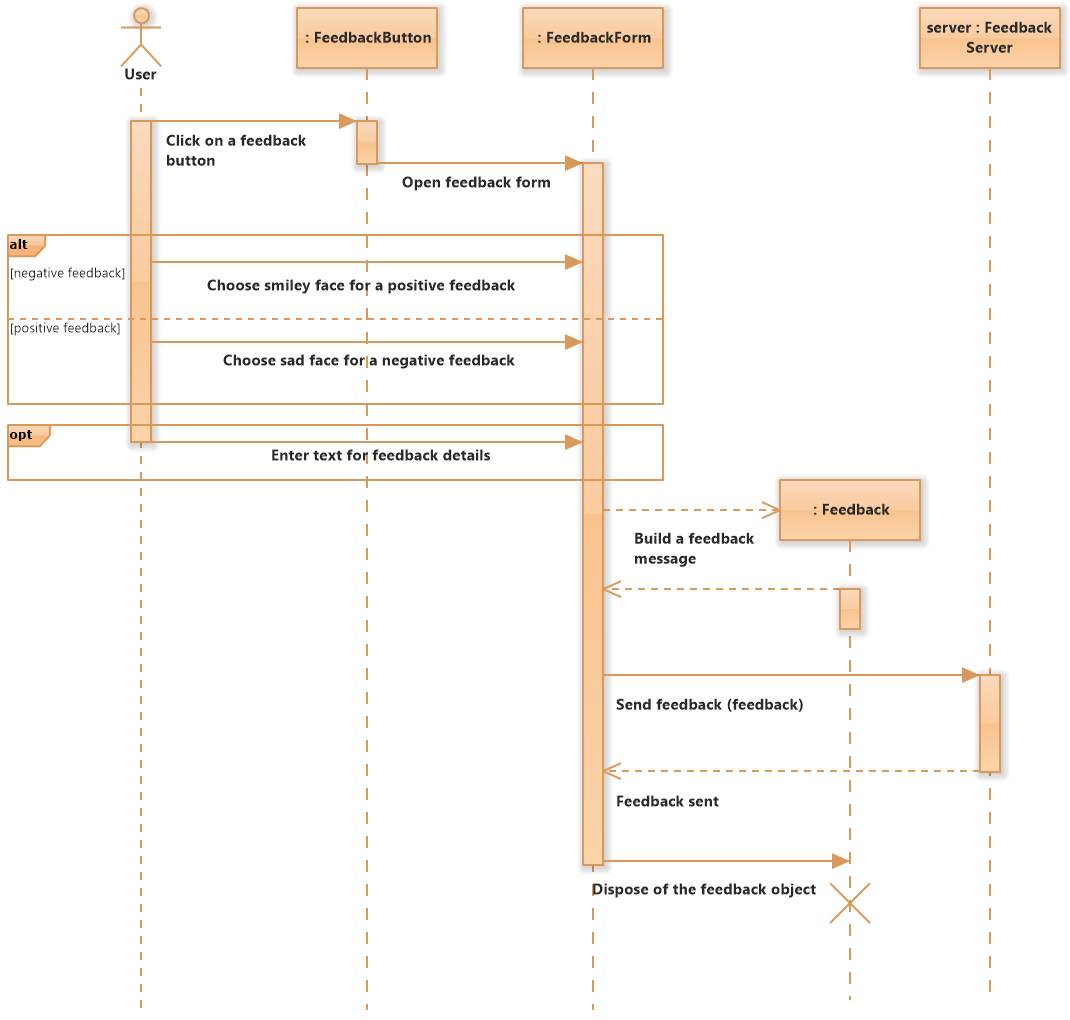
### b. Sequence Diagram(s)



## 14. Feature “Feedback”

### a. Class Diagram

### b. Sequence Diagram(s)



# III. Database Tables

## 1. Order

*This table contain order information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Order\_id | Int | 11 | x | x | PK | Order ID |
| 2 | Customer\_id | Int | 11 | x | x | FK | Customer ID |
| 3 | Name\_receiver | Varchar | 60 | x | x |  | Receive name |
| 4 | Phone | Int | 11 | x | x | FK | Phone number |
| 5 | Email | Varchar | 60 | x |  |  | Email address |
| 6 | Address | Varchar | 100 | x | x |  | Delivery address |
| 7 | Employee\_id | Int | 11 | x | x | FK | Employee ID |
| 8 | Order\_date | Date |  |  | x |  | Order date |
| 9 | Ship\_date | Date |  |  | x |  | Delivery date |
| 10 | Status | Varchar | 45 |  | x |  | Order status |
| 11 | Note | Varchar | 100 |  |  |  | Order note |
| 12 | User\_account | Int | 11 | x | x | FK | User account |

## 2. Order detail

*This table contain order detail information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Order\_id | Int | 11 | x | x | PK | Order ID |
| 2 | Product\_id | Int | 11 | x | x | PK | Product ID |
| 3 | Unit\_price | Float |  |  | x | FK | Unit price |
| 4 | Quantity | Int | 11 |  | x | FK | Order quantity |
| 5 | Order\_order\_id | Int | 11 |  |  | FK |  |
| 6 | Product\_product\_id | Int | 11 |  |  | FK |  |

## 3. Blog

*This table contain blog information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Blog\_id | Int | 11 | x | x | PK | Blog id |
| 2 | Create\_date | Date |  |  | x |  | Blog create date |
| 3 | Content | Varchar | 200 |  | x |  | Blog content |
| 4 | Description | Varchar | 1000 | x | x |  | Blog description |
| 5 | noidung | Longtext |  | x | x | FK | User account |
| 6 | User\_Account\_User\_id | Int | 11 |  | x | FK |  |
| 7 | category\_blog\_id | Int | 11 |  | x | FK |  |
| 8 | img | Varchar | 255 |  | x |  |  |

## 

## 

## 

## 4. Blog\_categories

*This table contain blog information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Blog\_id | Int | 11 | x | x | PK | Blog id |
| 2 | Blog\_name | Varchar | 255 |  |  |  | Blog create date |

## 

## 5. User account

*This table contain user account information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | User\_id | Int | 11 | x | x | PK | User ID |
| 2 | User\_name | Varchar | 60 | x | x |  | User name |
| 3 | Password | Varchar | 32 |  | x |  | Password |
| 4 | Full\_name | Varchar | 60 |  | x |  | User full name |
| 5 | Gender | Int | 11 |  |  |  | Gender |
| 6 | Phone | Int | 11 |  | x |  | Phone number |
| 7 | Email | Varchar | 60 |  |  |  | Email address |
| 8 | Status\_id | Int | 11 |  | x | FK | User authority |
| 9 | authority\_id | Int | 11 |  | x | FK | Decentralization |
| 10 | code | Varchar | 10 |  |  |  |  |

## 6. Authority

*This table contain user authority information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Id | Int | 11 | x | x | PK | Authority id |
| 2 | Name | Varchar | 60 |  | x |  | Authority name |

## 7. Feedback

*This table contain feedback information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Feedback\_id | Int | 11 | x | x | PK | Feedback id |
| 2 | Description | Varchar | 1000 |  | x |  | Description |
| 3 | Product\_id | Int |  | x | x | FK | Feedback product id |
| 4 | User\_account | Int |  | x | x | FK | User account |
| 5 | Note | Varchar | 1000 |  |  |  |  |
| 6 | Status | Bit |  |  |  |  |  |

## 8. Cart

*This table contain cart information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | User\_id | Int |  | x | x | PK | User ID |
| 2 | Status | Varchar | 255 |  | x |  | Cart status |
| 3 | Full\_name | Varchar | 60 |  | x |  | Full name |
| 4 | Phone | Int | 11 |  | x |  | Phone number |
| 5 | Email | Varchar | 60 |  |  |  | Email address |
| 6 | Address | Varchar | 100 |  | x |  | Address |
| 7 | User\_Account\_user\_id | Int | 11 |  | x | FK |  |

## 9. Cart\_items

*This table contain post information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Item\_id | Int | 11 | x | x | PK |  |
| 2 | Product\_id | Int | 11 |  | x | FK |  |
| 3 | Product\_name | Varchar | 100 |  | x |  |  |
| 4 | Url | Varchar | 1000 | x | x |  |  |
| 5 | Unit\_price | Int | 11 | x | x | FK |  |
| 6 | Quantity | Int | 11 |  | x | FK |  |
| 7 | Cart\_user\_id | Int | 11 |  | x | FK |  |
| 8 | Products\_product\_id | Int | 11 |  | x | FK |  |

## 

## 10. Post

*This table contain post information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Post\_id | Int | 11 | x | x | PK | Post ID |
| 2 | Content | Varchar | 255 |  | x |  | Post content |
| 3 | Brief\_information | Varchar | 200 |  | x |  | Create date |
| 4 | Description | Varchar | 200 | x | x |  | User ID |
| 5 | create\_date | Date |  | x | x |  | Product id |
| 6 | image | Varchar | 200 |  |  |  |  |
| 7 | User\_Account\_user\_id | Int | 11 |  | x | FK |  |

## 11. Post\_categories

*This table contain category information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Post\_id | Int | 11 | x | x | PK |  |
| 2 | Post\_name | Varchar | 255 |  | x |  |  |

## 12. Product

*This table contain product information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Product\_id | Int | 11 | x | x | PK | Product ID |
| 2 | Product\_name | Varchar | 100 |  | x |  | Product name |
| 3 | Category\_id | Int | 11 | x | x | FK | Category ID |
| 4 | Unit\_price | Float |  |  | x | FK | Product price |
| 5 | Sale\_price | Float |  |  | x | FK | Unit stock |
| 6 | Unitsln\_stock | Int | 11 |  |  | FK |  |
| 7 | Discontinued | Tinyint | 4 |  | x | FK |  |
| 8 | Brief\_information | Varchar | 200 | x | x |  |  |
| 9 | Description | Varchar | 1000 | x | x |  | Product description |
| 10 | Url | Varchar | 1000 |  |  |  | Product url |
| 11 | Description-short | Varchar | 1000 |  |  |  |  |

## 13. Product category

*This table contain category information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Category\_id | Int |  | x | x | PK | Category ID |
| 2 | Category\_name | Varchar | 100 |  | x |  | Category Name |
| 3 | Description | Varchar | 1000 |  |  |  | Description |

## 14. Slide

*This table contain category item information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Slide\_id | Int | 6 | x | x | PK |  |
| 2 | Img | Varchar | 255 | x | x |  |  |
| 3 | Heading | Varchar | 255 |  | x |  |  |
| 4 | Description | Varchar | 255 |  |  |  |  |
| 5 | Name\_button | Varchar | 255 |  | x |  |  |
| 6 | Url | Varchar | 255 |  |  |  |  |
| 7 | Status | Tinyint | 1 |  | x | FK |  |

## 15. Status

*This table contain category information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Id | Int | 11 | x | x | PK |  |
| 2 | Name | Varchar | 60 |  | x |  |  |

## 16. Product category

*This table contain category information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Feedback\_id | Int | 11 | x | x | PK |  |
| 2 | Description | Varchar | 1000 |  | x |  |  |
| 3 | User\_Account\_user\_id | Int | 11 |  | x | FK |  |
| 4 | Product\_product\_id | Int | 11 |  | x | FK |  |
| 5 | Star\_rating | Int | 11 |  | x | FK |  |

## 17. Image

*This table contain category information*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | Img\_id | Int | 100 | x | x | PK | Category ID |
| 2 | Product\_id | Varchar | 100 |  | x | FK | Category Name |
| 3 | Url | Varchar | 1000 |  |  |  | Description |

## 18. Tool and Infrastructures

|  |  |  |
| --- | --- | --- |
| **#** | **Category** | **Tools/Infrastructures** |
| 1 | Techology | Languague: Java + JSP+HTML+ Xamapp |
| 2 | IDEs/Editors | Netbeans IDE 8(8.2 or more) |
| 3 | Database | MySql of Xamapp |
| 4 | Diagram Programing | Draw.io |
| 5 | Documentation | Google docs/sheet, Word |
| 6 | Version Control And Manage Source Code | Gitlab, Google Drive |
| 7 | Deployment Server | None |
| 8 | Project Management | Google meet(Tasks)  Meeting minutes(schedule work) |

**18.1 Assumptions**

System’s name**: Online Shop** **(OS)**

This system is designed based on:

* Windows 10/11
* JDK 16
* Java
* Xamapp/MySQL
* Netbeans 8 (8.2 or more)

**18.2 Design Constraints**

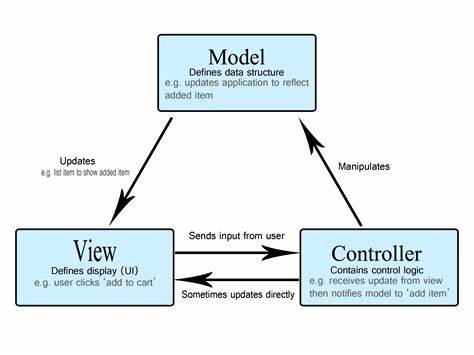
* It is a sales website with user and management sides. In the epidemic situation, a website for users to buy goods is really necessary.
* End-user’s Environment: Windows Version >=10.
* Database: MySQL and Xamapp.
* Support language: Eng and VN.
* Criteria: snappy, soft and responsive.
* Permission: Internet Access

**18.3 System Architecture Diagram**

****

**18.3.1 Component Explanation**

**18.3.1.1 MVC model**



MVC (short for Model - View - Controller) is a type of software architecture pattern that separates the logic in the application into three main components: Model, View and Controller with each component having a separate function.

* Models manage application behavior and data.
* View has the function to perform tasks related to display functions (text, images, graphics...) in the application.
* Controller has the function to perform tasks related to handling input provided to the application (maybe from users or other applications) and send requests to the Model or View for further processing.

**18.3.1.2 JSP, Servlet**



JSP stands for Java Server Pages is a technology for building web applications that support dynamic content and acts as a Java servlet technology. You can consider it a different option to a servlet, and it has a lot more capabilities than a servlet.

Servlet is a server-side Java program module that handles client requests and implements the servlet interface. Servlets can respond to any type of request, and they are commonly used to extend the applications hosted by web servers.

**18.3.1.3 Xamapp**



XAMPP is a popular**cross-platform web server** that allows programmers to write and test their code on a local webserver. It was created by Apache Friends, and the public can revise or modify its native source code. It includes MariaDB, Apache HTTP Server, and interpreters for PHP and Perl, among other computer languages.

**18.3.1.4 MySQL**



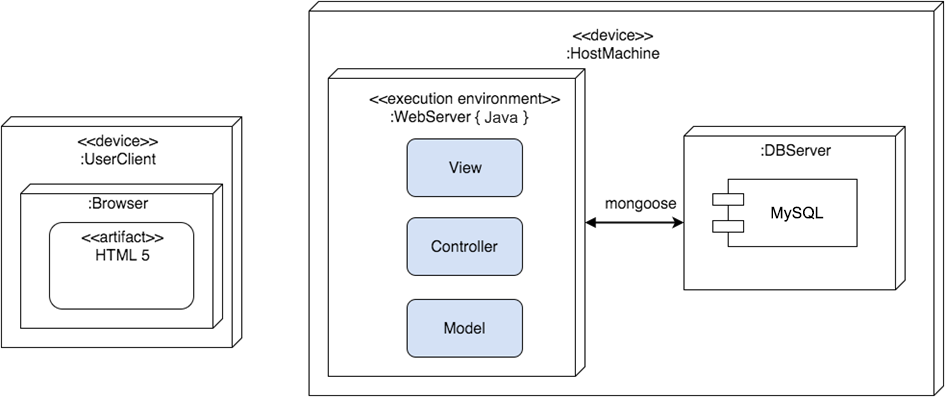
MySQL is a relational database management system (RDBMS) developed by Oracle that is based on structured query language (SQL). A database is a structured collection of data.

**18.3.1.5 NetBeans**



NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, it has extensions for other languages like PHP, C, C++, HTML5, and JavaScript.

**19. Deployment Diagram**

****