



RMIT UNIVERSITY

Term	Semester 3, 2022
Course code	COSC2081
Course name	Programming 1
Title of Assignment	Assignment 3: Group Assignment
Due date	16/01/2023
Group Name	Siuuuu
Names and student IDs	Nguyen Thien Co - s3938338 Hoang Quoc Bao - s3926050 Hoang Vinh Khue - s3927474 Nguyen Dang Ha - s3924594
Location	Hanoi Campus
Lecturer	Tom Huynh Phong Ngo Tuan
Number of Page	21 (including Cover Page, Table of Contents and Images)

Store Order Management System

Table of Contents

1. Introduction	2
1.1. Project Overview	2
1.2. Team Members	2
2. Project Detailed Description	2
3. Implementation Details	3
3.1. Class diagram	3
3.2. Features and Illustration	3
A. All users functions	3
B. Customer-only functions	5
C. Administration functions	13
3.3. Updates In The Future	17
4. Project Planning Report	17
5. Conclusion	18

1. Introduction

1.1. Project Overview

This is the report documentation for our Store Order Management System, a text-based program that allows customer users to make online purchases while admin users can manage and monitor all of the store's inventory as well as customer orders.

The document's main objective is to give readers all the essential information regarding the application's user interface, its features, the relationships between classes with a class diagram, and a demonstration of how to use it.

1.2. Team Members

- **Hoang Quoc Bao:** *Team leader*, Side Developer (Admin Flow)
- **Nguyen Thien Co:** Main Developer (Customer Flow & Review Project)
- **Hoang Vinh Khue:** Side Developer (Product Function)
- **Nguyen Dang Ha:** Project Tester

2. Project Detailed Description

The Store Order Management System application's primary goal is to handle client product orders. It must be capable of managing store items, order details, registered users, including customers and their lists of purchases. The admin users of the system will be given an administrator account to complete all of these duties. With the help of the application, customer users are now able to purchase and view their favorite products online, which significantly cuts down on the time that they have to spend doing traditional shopping. Additionally, the system will be able to turn customers into members of the shop based on their prior purchases. Numerous advantages come with store membership, including discounts on future purchases. On the other hand, staff members can use the program's functional features to operate the store's inventory and order system, which significantly lessens their workload in terms of paperwork, printing, and distribution.

A simplistic text-based console interface will be implemented for the application's User Interface (UI). This would greatly improve the application's usability so that even non-technical people may use it without experiencing any problems.

- **GitHub repo URL:** <https://github.com/BaoHoangRmit/COSC2081-A3-Siuuuu>

3. Implementation Details

3.1. Class diagram

Given that the entire application was created in Java using all of Java's OOP (Object-Oriented Programming) ideas, we will present a class diagram to show how each class is organized in terms of its attributes, methods, and relationships. The diagram will give readers a clearer grasp of how the application works and how users interact with its interface.

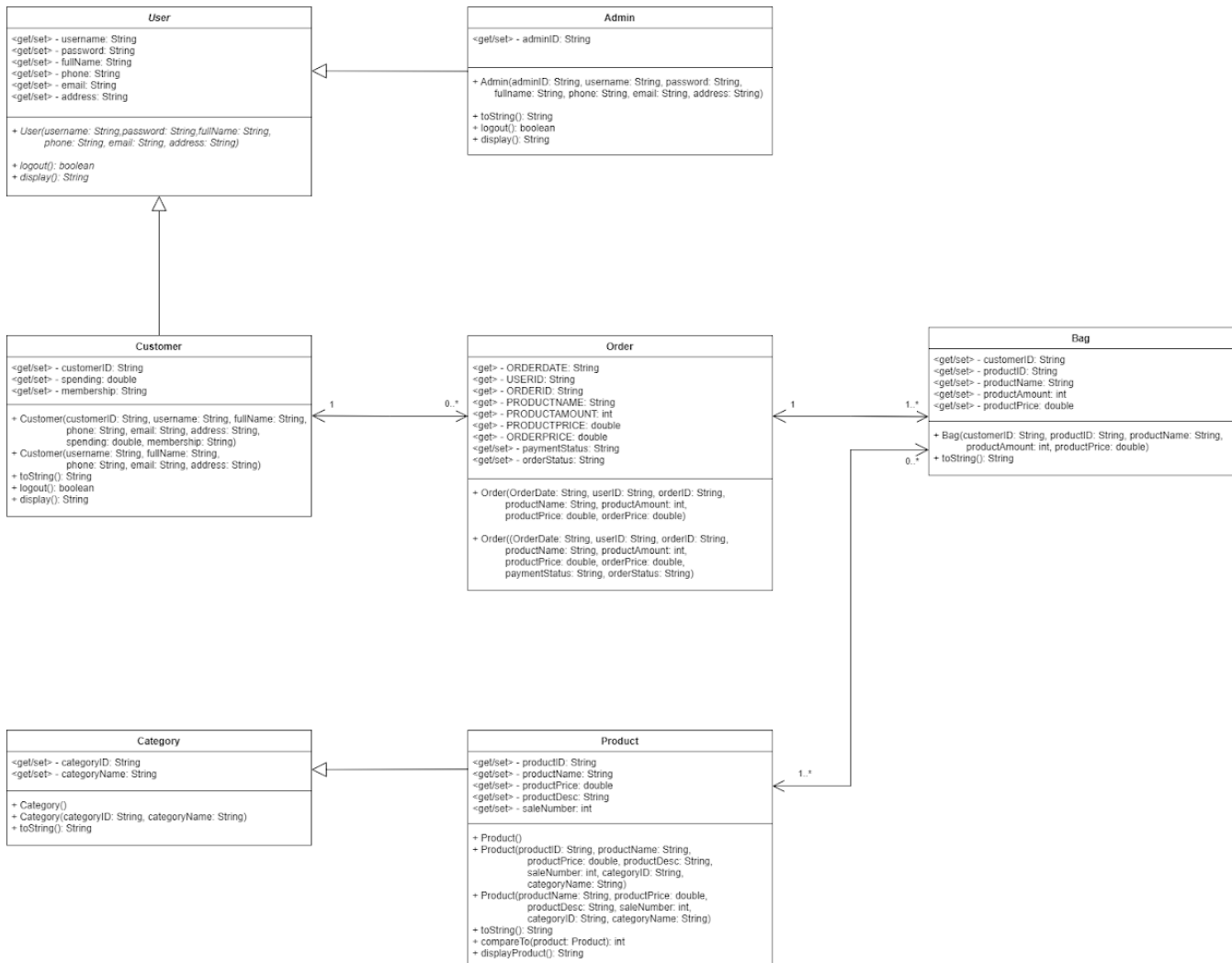


Diagram 1: UML Class Diagram

Diagram 1 explains how the Main Objects interact with each other in our system. With 7 classes with relations, the system is tightly coordinated and transparent. By renewing the design and concept of Bag, Product, and Order interaction, we save some time and resources, allowing a newly created system.

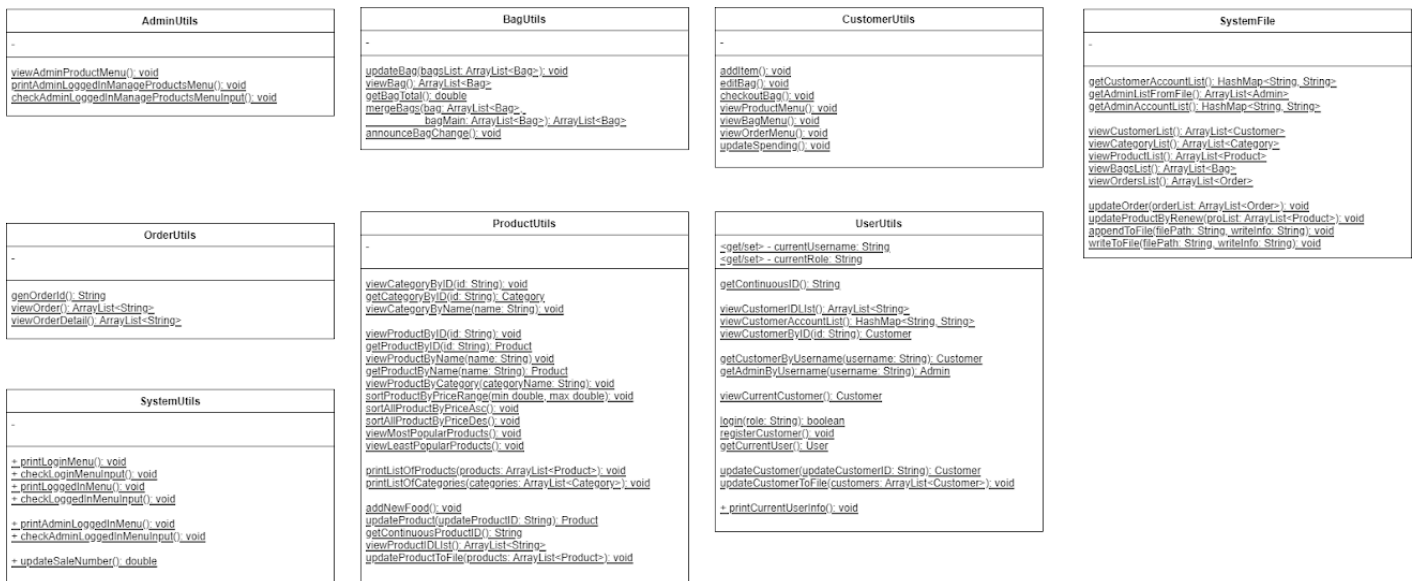


Diagram 2: UML Extra Class Diagram

The Utils classes in Diagram 2 will consist of 8 elements, which will help us align and order managing functions of our system. The Main.java class will start the system, and will call the first function of SystemFile.java, which will respectively navigate to other classes. SystemFile.java and SystemUtils.java will be the classes to read information from the database (txt file) and render display. UserUtils will consist of methods relating to the User management process. Respectively, all Utils classes will contain functions relating to its' classname, aiding the managing and maintenance process.

3.2. Features and Illustration

This section of the report will detailly illustrate the instructions for using the application and all of its features.

A. All users functions

The console interface will greet the user with a Welcome Screen when the application is launched. Below the Welcome Screen is the Menu Screen, where users can select their first four actions:

```
----- WELCOME SCREEN -----  
COSC2081 GROUP ASSIGNMENT  
STORE ORDER MANAGEMENT SYSTEM  
Instructor: Mr. Tom Huynh & Dr. Phong Ngo  
Group: Siuuuu  
s3926050, Hoang Quoc Bao  
s3938338, Nguyen Thien Co  
s3927474, Hoang Vinh Khue  
s3924594, Nguyen Dang Ha  
----- WELCOME SCREEN -----  
  
----- MENU SCREEN -----  
1: Login as Customer  
2: Signup as Customer  
3: Login as Admin  
4: Exit Application  
Enter your number option: |
```

Figure 1. The Welcome Screen and Menu Screen.

- Login as customer or admin

If the user chooses action number 1 or 3, the user will be directed to their login screen, where they can enter their account username and password.

```
Enter your number option: 1  
Enter username: customer11  
Enter password: password11|
```

```
Enter your number option: 3  
Enter username: admin1  
Enter password: adminPassword1
```

Figure 2 & Figure 3. The input field for login function.

After entering their account username and password, they will be logged in and be directed to the Customer Screen or Admin Screen. They can select the log out action on the appropriate screen whenever they want to log out of their account. The user will be redirected to the Menu Screen if they enter the wrong account username or password during the login process.

- Sign up as customer

By selecting action number 2, users can register for an account as a customer. The program will leave an area for the user to fill up with their new account's username, password and their personal information including their full name, phone number, email address and home address.

```
Enter your number option: 2

Enter your new username: khue2511
Enter your new password: khuepassword
Enter your full name: Hoang Vinh Khue
Enter your phone number: 096-228-3938
Enter your email address: khue251103@gmail.com
Enter your address: Phao Dai Lang, HN
```

Figure 4. The information form when you sign up.

However, since this role is not accessible to all users, they will not be able to register as admin users. The store's employees will be given their personal admin accounts.

- Exit application

To close the application, users can type "4".

```
Enter your number option: 4

Process finished with exit code 0
```

Figure 5. Exit application.

There will, however, always be a method for them to quit the application at any point while they are using it.

```
7: Quit application (your account will be logged out)
```

Figure 6. Another function to exit the application.

B. Customer-only functions

```
----- Bag Update -----
Bag is updated due to the change of product price or existence!
2 item(s) (changes) found in your Bag
1. P114 - Headphone1
3. P00117 - Banana
Please re-check your Bag!
-----
```

Figure 7. Bag Update

After logging in, customer users can begin using the application's customer features that are displayed on the Customer Screen.

```
----- CUSTOMER SCREEN -----  
1: View Personal Information  
2: Update Personal Information  
3: View Product  
4: View Current Bag  
5: View Order(s)  
6: Logout  
7: Quit application (your account will be logged out)  
  
Enter your number option:
```

Figure 8. Customer Screen.

- Bag Announcement/ Bag Update

After logging in, if there are any product changes by the administrator that have effects on the users' bags, the current user will be informed and their bag will be updated.

- View Personal Information

Choosing option number one will trigger this function, where users can view their personal information on their account.

```
Enter your number option: 1  
ID= C001, Username= customer11, Password = password11:  
fullName= Customer 11,  
Phone= 111-111-111,  
Email= 11@gmail.com,  
Address= customer address 11,  
Spending= 0.0000,  
Membership= None  
Returning to Menu...
```

Figure 9. Personal information Screen.

- Update Personal Information

The user can modify their account's personal information by using the second customer feature. Once the user has finished filling out the field, the updated information will show up immediately on the screen.


```

Enter your number option: 2

Enter your desired username: khue2511
Enter your desired password: khue2003
Enter your full name: Khue Hoang
Enter your phone number: 0962283938
Enter your email address: khue2511@gmail.com
Enter your address: Hanoi Vietnam
Updated Info:
ID= C001, Username= khue2511, Password = khue2003:
fullName= Khue Hoang,
Phone= 0962283938,
Email= khue2511@gmail.com,
Address= Hanoi Vietnam,
Spending= 0.0000,
Membership= None
Returning to Menu...

```

Figure 10. Change information Screen.

- View Products

The third choice is the option to view products. Users through this function will have access to a variety of side functions, such as one that lets them browse every item in the store, and others that let them filter products to locate items that suit their interests. Before placing an order, users can use the second side function to select and add their items to their Bag.

```

Enter your number option: 3

----- Product List -----

Product List Options:
1. View All Product(s)
2. Add Item to Bag
3. View Product(s) By Category
4. View Product(s) By ID
5. View Product(s) By Name
6. Sort Product(s) By Price Ascending
7. Sort Product(s) By Price Descending
8. View Most Popular Product
9. View Least Popular Product
10. Return to Menu

Enter a number: |

```

Figure 11. Side functions of View Products function.

The functionality of the side functions will be described below this paragraph.

- View All Product(s)

This function will display every product that is being sold on the store.

```

Enter a number: 1
Product(ID. Name):
P111. NAcbookNew:
  Description: This is macbook
  Category: Laptop
  Price: 9.9900
  Total Sale: 20
P112. iPhone2:
  Description: This is expensive
  Category: Smartphone
  Price: 19.9900
  Total Sale: 30
P113. Mouse1:
  Description: This mouse is made for gaming
  Category: Mouse
  Price: 20.9900
  Total Sale: 15
P114. Headphone1:
  Description: This is a great headphone
  Category: Headphone
  Price: 5.9900
  Total Sale: 30

```

Figure 12. All products Screen.

- Add Item to Bag

Given the information about the products from other functions, users can add their wanted product to their bag by choosing this option.

The user will be prompted for the product ID or name before the system displays the results on the screen and gives them the option of adding the item to their bag.

```

Enter a number: 2

----- Add Items(s) -----
Enter ProductID/ ProductName: P112
Item Found: iphone7, Price: 29.98

Do you want to add this item [y/n]: y

Enter amount: 1
Do you want to add another item [y/n]: y
Processing...
Enter ProductID/ ProductName: Dell
Item Found: Dell, Price: 15.0

Do you want to add this item [y/n]: y

Enter amount: 2
Do you want to add another item [y/n]: n

```

Figure 13. Add item to Bag Screen.

Later on the document, we will show the function where users can view their current bag.

- View Product(s) By Category

This function allows users to look for the products that have their chosen category. Users will need to specify that category name in order to see every item within that category.

```
Enter a number: 3
Enter Category Name: Mouse
Products with the category: Mouse
ID | Name | Price | Category
P113 | Mouse1 | 20.99 | Mouse
```

Figure 14. Product information provided with the category.

- View Product(s) By ID

Users after filling in the specific product ID will be able to view the item's information, including ID, Name, Price and Category.

```
Enter a number: 4
Enter ProductID: P114
Products with the ID: P114
ID | Name | Price | Category
P114 | Headphone1 | 5.99 | Headphone
```

Figure 15. Product information provided with the ID.

- View Product(s) By Name

Similar to the previous function, users will be able to view a specific product's information after they search for the product name.

```
Enter a number: 5
Enter Product Name: Mouse1
Products with the name: Mouse1
ID | Name | Price | Category
P113 | Mouse1 | 20.99 | Mouse
```

Figure 16. Product information provided with the name.

- Sort Product(s) By Price Ascending

All items from the store will be displayed using this function, with prices sorted from lowest to highest.

```
Enter a number: 6
All products sorted by price ascendant order:
ID | Name | Price | Category
P114 | Headphone1 | 5.99 | Headphone
P111 | NAckbookNew | 9.99 | Laptop
P112 | iPhone2 | 19.99 | Smartphone
P113 | Mouse1 | 20.99 | Mouse
```

Figure 17. Sorted Products from low to high price.

- Sort Product(s) By Price Descending

Unlike the 6th function, this function will present the items from most expensive to least expensive.

```
Enter a number: 7
All products sorted by price descendant order:
ID | Name | Price | Category
P113 | Mouse1 | 20.99 | Mouse
P112 | iPhone2 | 19.99 | Smartphone
P111 | NAckbookNew | 9.99 | Laptop
P114 | Headphone1 | 5.99 | Headphone
```

Figure 18. Sorted Products from high to low price.

- View Most Popular Product

The information about the products that share the largest sales number will be shown to the users through this function.

```
Enter a number: 8
Most popular products with 30 items sold

ID | Name | Price | Category
P112 | iPhone2 | 19.99 | Smartphone
P114 | Headphone1 | 5.99 | Headphone
```

Figure 19. Most popular products information.

- View Least Popular Product

Additionally, we have another function for users to view the items that have the fewest sales.

```

Enter a number: 9
Least popular products with 15 items sold

ID | Name | Price | Category
P113 | Mouse1 | 20.99 | Mouse

```

Figure 20. Least popular products information.

- View Current Bag

After getting items from the “Add Item to Bag” function, users can view their current bag by choosing this function from the Customer Screen.

```

Enter your number option: 4

----- Bag Content -----
1. iphone7: Amount: 1, Price: 29.9800
2. Dell: Amount: 2, Price: 30.0000

Total: 3 items.
Total Price: 59.9800
Membership Discount: 0%
Final Price: 59.9800

Bag Options
1. Edit Bag
2. Check Out Bag
3. Return To Menu
Please Select A Number:

```

Figure 21. Current Bag content Screen.

The content of the bag including total items and their information, price, Membership discount and final price after the discount will be displayed.

User then can choose 3 different options:

- + **Option 1** will let them edit their current bag, allowing them to remove any items from the bag.

```

----- Edit Bag -----
1. iphone7: Amount: 1, Price: 29.9800
2. Dell: Amount: 2, Price: 30.0000

Total: 3 items.
Total Price: 59.9800
Membership Discount: 0%
Final Price: 59.9800

Enter a number to edit the corresponding section or:
3. Remove All Items
4. Cancel

```

Figure 22. Edit Bag Screen.

- + **Option 2** will let the customer checkout their bag and make the order

```
----- Check Out Bag -----
1. iphone7: Amount: 1, Price: 29.9800
2. Dell: Amount: 2, Price: 30.0000

Total: 3 items.
Total Price: 59.9800
Membership Discount: 0%
Final Price: 59.9800
Do you want to order these items [y/n]: y

----- Orders Updated -----
Returning to Menu...
```

Figure 23. Bag checkout Screen.

- + **Option 3** will return the user to the Customer Screen.

- View Order(s)

After the user has checked out their bag, they can then check the order information by choosing the 5th option on the screen.

```
----- Orders -----
1 Bill(s) Found!

1.B2:
Date: 17/01/2023 16:54:23
iphone7, Amount: 1, Price: 29.9800
Dell, Amount: 2, Price: 30.0000

Order Price: 59.9800
Payment Status: Unpaid
Order Status: Delivering

Order Options
1. Pay Order(s)
2. Return To Menu
Please Select A Number:
```

Figure 24. View Order Screen.

If the user chooses “1” in the Order Options, the system will ask them for confirmation about the payment of the order. Typing “y” in the input field will complete the payment and also , while typing “n” will do the opposite.

```
Order Options
1. Pay Order(s)
2. Return To Menu
Please Select A Number: 1
Do you want to order these items [y/n]: y

----- Orders Updated -----
All bills have been paid!
Returning to Menu...
```

Figure 25. Options to update order.

- Log out

This option will allow users to log out their account and return them to the menu page.

C. Administration functions

Admin users can use the application's admin features that are present on the Admin Screen after logging in.

```
----- ADMIN SCREEN -----  
Store Revenue: 62.97 VND  
1: View personal information  
2: Manage Products  
3: View Categories  
4: Manage Orders  
5: View Customers List  
6: Logout
```

Figure 26. Admin Screen.

- View personal information

Similar to the customer's one, this function will display the personal information of the admin user.

```
----- Personal Information -----  
ID= A001, Username= admin1, Password = adminPassword1:  
fullName= Admin 1,  
Phone= 111-111-111,  
Email= 1@admin.com,  
Address= admin address 1
```

Figure 27. Admin personal information Screen.

- Manage Products

This function will help admin users to manage their store products by offering four side functions.

```
----- Edit Product -----  
1: View Products  
2: Add new product  
3: Update product  
4: Delete product  
5: Back to Menu
```

Figure 28. Options to manage products

- View Products

This function is very similar to the customer's View Products function. The only difference is that admin users cannot add item to bag,

- Add new product

Using this feature, admin users can update the database of the shop with new products. Users must provide the product's name, price, description, and category to successfully add it. The Product ID will be generated automatically by the system.

```
----- Add New Product -----
Enter the product's name: Xiaomi2023
Enter the product's price: 100.000
Enter product's description: This is a great budget phone
ID | Name
1 | Laptop
2 | Smartphone
3 | Watch
4 | Mouse
5 | Keyboard
6 | Headphone

Enter the product's category id: 2

----- Product added successfully -----
```

Figure 29. Add product Screen.

- Update product

This function allows the admin to choose an existing product and edit its information. The system will provide an input field for the user so that they can fill in their new information for the product

```
Enter your number option: 3
Please enter the product ID you want to update: P111
Enter your new product name: Macbook
Enter the product's price: 20.0000
Enter product's description: This is a macbook
ID | Name
1 | Laptop
2 | Smartphone
3 | Watch
4 | Mouse
5 | Keyboard
6 | Headphone

Enter the product's category id: 1
Updated Info:
P111,Macbook,20.00,This is a macbook,0,1,Laptop
Returning to Menu...
```


Figure 30. Update product information Screen.

If the admin user updates an item that is currently in a customer's bag, it will return a notification to the customer when they log into the system.

```
----- Bag Update -----  
Bag is updated due to the change of product price or existence!  
1 item(s) (changes) found in your Bag  
1. 1112 - iPhone2  
Please re-check your Bag!  
-----  
  
----- CUSTOMER SCREEN -----
```

Figure 31. Bag changes notification.

- Delete product

As suggested by the function's name, admin users may use this feature to remove an item from the store's inventory. The system will delete the item after they have entered the product's name or ID.

```
Enter your number option: 4  
Please enter the product ID/Name to remove: Dell  
Return to Menu...  
  
----- Item Removed -----
```

Figure 32. Item deleted Screen.

- View Categories

This function will display every category of products in the store.

```
----- Category List -----  
ID | Name  
1 | Fashion  
2 | Electronics  
3 | Furniture  
4 | Beauty  
5 | Accessories  
6 | Household Items  
7 | Toys  
8 | Sports  
9 | Meat  
10 | Fruit  
11 | Vegetable
```

Figure 33. Category list display.

- Manage Orders

The system will display all of the orders that are made by the customers of the store. The order information will include the Customer ID, the order items, the price, the order and payment status. They can then choose 1 to deliver all of the orders in the list, or choose 2 to return to the menu.

```
----- Order List -----
1. B1(OrderID):
Customer: C002

Order Content:
- Mouse1, Amount: 3, Price: 62.9700
Order Price: 62.9700
Payment Status: Paid
Order Status: Delivered

2. B2(OrderID):
Customer: C003

Order Content:
- MacbookNew, Amount: 2, Price: 36.0000
- Banana, Amount: 10, Price: 90000.0000
Order Price: 90036.0000
Payment Status: Unpaid
Order Status: Delivering

Order Options:
1. Deliver All Order(s)
2. Return to Menu
Enter a number:
```

Figure 34. Order list and options display.

- View Customers List

With option number five, admin users can view every detail information of every customer of the store.

```
----- Customer List -----
1 ID= C004, Username= customer4, Password = password4:
fullName= Customer 4,
Phone= 444-444-444,
Email= 4@customer.com,
Address= customer address 4,
Spending= 21.0000,
Membership= Platinum

2 ID= C002, Username= customer2, Password = password2new:
fullName= Customer2,
Phone= 222-222-222,
Email= 2@customer.comnew,
Address= address 2,
Spending= 0.0000,
Membership= None

3 ID= C003, Username= customer3, Password = password3:
fullName= Customer 3,
Phone= 333-333-333,
Email= 3@customer.com,
Address= customer address 3,
Spending= 0.0000,
Membership= None
```

Figure 35. Customer List

- Log-out

This function will simply log out the user's account.

3.3. Updates In The Future

In our initial scope of the project, there are still a number of other functions. The functions list include: Pay order separately (for customer), Manage Categories (Delete) and Manage Customer (Delete, Update). Besides, there are several functions that our group wants to update such as "Pay order separately (for customer)" or "Deliver order separately (for administrator)". These features will be taken into consideration for future addition to the application but are not currently implemented in this project. The lack of time we are dealing with, which will be covered later in the project, is the cause of this predicament.

4. Project Planning Report

To approach this project, we did not jump right into the coding part immediately. Initially, we had to get to know the members we got on our team, made an initial meeting to analyze each member's strengths and weaknesses, then assigned suitable roles for everyone. After that, we had to take a look at the project requirements. It is obvious that we had to figure out what tasks our program should be able to complete, and then draw out the UML diagram, in this case it's a class diagram, to see what classes, functions or methods should be implemented and built in the program. Once the diagram is finished, we start coding. At first, we implemented the necessary classes to form the initial outline for the whole program. Then the functions and methods coding was conducted. This was when the tough part emerged. Many errors occurred while developing, making debugging and testing the hardest job in this project. It was intended for one week to test and debug the program. However, the problem was more severe than we expected. To preserve the outcome quality, we decided to make changes in the scope. More effort to test and debug the implemented functions is needed. Finally, with that decision, the final product, however, was not as detailed as we expected, but it was functionable and usable.

Team members	Role and Task Given	Contribution
--------------	---------------------	--------------

Hoang Quoc Bao	Draw the class diagram for the projects, take part in the log in, log out function for all users, take part in the Admin flow. Write the Administrator functions part of the report	26%
Hoang Vinh Khue	Take part in coding the function for managing products, categories in the ProductUtils class. Write the Introduction, Project Description and Customer functions part in the report.	26%
Nguyen Thien Co	Take part in the function for the Customer flow, including coding of the class SystemFile and many Utility files. Write the All users functions and Class diagram section in the report	30%
Nguyen Dang Ha	Take part in the coding of the Admin class. Write the Project Planning and Conclusion section in the report.	18%

Table 1. Team members and contribution

5. Conclusion

In terms of the coding part of this project, our team generally performed fairly well. But a delayed finish has taught us a valuable lesson regarding the importance of time management when working on a project. Three members of our team - Bao, Khue, and Co - who had previously interacted on other projects, are the members that made up our team when the project first began. Ha, on the other hand, is a new member of the group, and because of his shy nature, it was extremely difficult for him to get along with the others. Bao, the project manager, demonstrated his leadership skills by making an effort to find a solution. If it was not because of his effort of hosting meeting sessions between members, the communication among the group could have been really tough. We are delighted to report that everyone on the team has behaved professionally toward one another, which has improved communication. Thus, while it required some work to bring everyone together for a meeting, it was not a huge problem. Undoubtedly, it was for a programming project, therefore the primary issue arose as soon as we began to code. When the group started to develop the essential classes that had been created in the UML diagram, things were initially going very

nicely. However, by the time we added functions and methods, the issue had already emerged. Bugs in the program kept happening, particularly as we approached the halfway point of the schedule. We may have some knowledge of the fundamentals of coding, but when it came to testing and debugging programs, we struggled to resolve issues. The hardest decision we had to make was to submit the project after the deadline (17/01/2023) in order to give ourselves more time to address any problems we discovered. We eventually succeeded in finishing our program inside the intended time frame. To finish the application, some adjustments had to be made, and some sub-content had to be reduced for the project to stay within its intended scope. With that mentioned, although the final outcome was acceptable for us, a late submission was definitely not what we expected at the beginning. This course in general and the final project in particular has taught us a valuable lesson: technical abilities alone will not help you complete a project successfully; you must also need to have a comprehensive planning phase, excellent teamwork, and, most crucially, effective time management. The team members' poor time management skills have cost us not just a 10% reduction on the mark, but also a product that could have been done much better.