|  |  |
| --- | --- |
| https://user-cdn.uef.edu.vn/newsimg/files/logo%20nhan%20dien%20TH/3.jpg | **HCM UNIVERSITY OF ECONOMICS AND FINANCE**  **FACULTY OF INFORMATION TECHNOLOGY** |

**FINAL PROJECT**

**SUBJECT: MOBILE DEVICE PROGRAMMING   
Teacher: Msc. Trần Thành Công**

**PROJECT NAME**

**MOBILE STORE APPLICATION**

*Team Members:*

1. Lê Duy Khánh - 185170616
2. Đặng Nguyễn Kim Long - 185050510
3. Đinh Văn Hoàn - 185050082
4. Nguyễn Nhật Khoa - 185050875
5. Văn Hoàng Phúc - 185050810
6. Nguyễn Đức Trung - 185050701

TP. Hồ Chí Minh – 2021

**HCM UNIVERSITY OF ECONOMICS AND FINANCE**

**FACULTY OF INFORMATION TECHNOLOGY**

**FINAL PROJECT**

**SUBJECT: MOBILE DEVICE PROGRAMMING**

**Teacher: Msc. Trần Thành Công**

**PROJECT NAME:**

**MOBILE STORE APPLICATION**

*Team Members:*

1. Lê Duy Khánh - 185170616
2. Đặng Nguyễn Kim Long - 185050510
3. Đinh Văn Hoàn - 185050082
4. Nguyễn Nhật Khoa - 185050875
5. Văn Hoàng Phúc - 185050810
6. Nguyễn Đức Trung - 185050701

TP. Hồ Chí Minh – 2021

**TABLE OF CONTENTS**

[LIST OF FIGURES 2](#_Toc31905)

[LIST OF TABLES 4](#_Toc27918)

[INTRODUCTION 5](#_Toc4296)

[TEAM MEMBERS 6](#_Toc10700)

[CHAPTER 1: OVERVIEW 7](#_Toc26776)

[1. TARGET – SCOPE OF STUDY 7](#_Toc2910)

[1.1. Target 7](#_Toc20351)

[1.2. Scope of Study 7](#_Toc24564)

[2. PROJECT DESCRIPTION 7](#_Toc12669)

[CHAPTER 2: SOLUTION STACK 9](#_Toc10199)

[1. Android Studio 9](#_Toc26844)

[2. Firebase 11](#_Toc19142)

[3. Chatbot 11](#_Toc17038)

[CHAPTER 3: USE CASE VIEW 13](#_Toc7503)

[1. USE CASE DIAGRAM 13](#_Toc13996)

[2. USECASE DESCRIPTION 13](#_Toc24041)

[2.1. UC-01 Payment 13](#_Toc32162)

[2.2. UC-02 Make a confirmation of delivery form 14](#_Toc20439)

[2.3. UC-03 Login 15](#_Toc12302)

[2.4. UC-04 Register 16](#_Toc2581)

[2.5. UC-05 Search product 17](#_Toc20071)

[2.6. UC-06 Add To Cart 18](#_Toc749)

[2.7. UC-07 Manage product 19](#_Toc14038)

[2.8. UC-08 Making the import product form 19](#_Toc25822)

[2.9. UC-09 Manage Staff 20](#_Toc751)

[2.10. UC-10 Counsel Online 21](#_Toc24563)

[CHAPTER 4: LOGICAL VIEW 23](#_Toc21552)

[1. CLASS DIAGRAM 23](#_Toc717)

[2. CLASS DESCRIPTION 23](#_Toc3435)

[2.1. Order 23](#_Toc16608)

[2.2. Product 24](#_Toc10374)

[2.3. ImportForm 25](#_Toc4918)

[2.4. Category 25](#_Toc4771)

[2.5. Person 25](#_Toc12612)

[2.6. Customer 26](#_Toc27925)

[2.7. Employee 26](#_Toc26420)

[2.8. Brand 26](#_Toc21016)

[CHAPTER 5: PROCESS VIEW 27](#_Toc8153)

[1. SEQUENCE DIAGRAM 27](#_Toc31933)

[1.1. Login 27](#_Toc32588)

[1.2. Register 27](#_Toc10462)

[1.3. Create Warehouse Receipt Form 28](#_Toc24981)

[1.4. Process Payment 28](#_Toc31637)

[2. COLLABORATION DIAGRAM 29](#_Toc29655)

[2.1. Process Payment 29](#_Toc16446)

[2.2. Process Order 29](#_Toc25525)

[3. STATE DAIGRAM 30](#_Toc9918)

[3.1. Cart 30](#_Toc16136)

[3.2. Order 30](#_Toc21742)

[4. ACTIVITY DAIGRAM 31](#_Toc11807)

[4.1. Login 31](#_Toc27729)

[4.2. Process Payment 31](#_Toc17694)

[CHAPTER 6: IMPLEMENTATION VIEW 32](#_Toc28285)

[1. COMPONENT DIAGRAM 32](#_Toc19770)

[2. DEMO 33](#_Toc32231)

[CHAPTER 7: CONCLUSION 43](#_Toc11486)

[1. Result 43](#_Toc9712)

[2. Limitations of the topic 43](#_Toc3587)

[3. The development direction of the topic 43](#_Toc27743)

[4. Suggest opinions 43](#_Toc7233)

[REFERENCES 45](#_Toc16302)

# LIST OF FIGURES

[Figure 1 : The project files in Android view. 10](#_Toc22921)

[Figure 2 : Traditional Database and FireBase Database 11](#_Toc25473)

[Figure 3 : Usecase Diagram 13](#_Toc17324)

[Figure 4 : Class Diagram 23](#_Toc29611)

[Figure 5 : Class Employee Description 26](#_Toc7758)

[Figure 6 : Login Sequence Diagram 27](#_Toc27337)

[Figure 7 : Register Sequence Diagram 28](#_Toc11877)

[Figure 8 : Create Warehouse Receipt Form Sequence Diagram 28](#_Toc8373)

[Figure 9 : Process Payment Sequence Diagram 28](#_Toc20134)

[Figure 10 : Process Payment Collaboraion Diagram 29](#_Toc10978)

[Figure 11 : Process Order Collaboraion Diagram 29](#_Toc20053)

[Figure 12 : Cart State Diagram 30](#_Toc20954)

[Figure 13 : Order State Diagram 30](#_Toc12891)

[Figure 14 : Login Activity Diagram 31](#_Toc31466)

[Figure 15 : Process Payment Activity Diagram 31](#_Toc3225)

[Figure 16 : Component Diagram 32](#_Toc23067)

[Figure 17 : Main screen of the application 33](#_Toc23932)

[Figure 18 : Search Product 34](#_Toc101)

[Figure 19 : Searched Items 35](#_Toc13265)

[Figure 20 : Product Details 36](#_Toc6688)

[Figure 21 : Navigation Bar 37](#_Toc32475)

[Figure 22 : Cart 38](#_Toc29115)

[Figure 23 : Payment Notification 39](#_Toc18083)

[Figure 24 : Category Filter 40](#_Toc2736)

[Figure 25 : Chatbot for Online Counseling 41](#_Toc16016)

[Figure 26 : QR Code Scanning 42](#_Toc26807)

# LIST OF TABLES

[Table 1 : Team Members 6](#_Toc11621)

[Table 2 : Payment Usecase Description 14](#_Toc5171)

[Table 3 : Make a confirmation of delivery form UseCase Description 15](#_Toc20236)

[Table 4 : Login UseCase Description 16](#_Toc15220)

[Table 5 : Register UseCase Description 17](#_Toc4264)

[Table 6 : Search Product UseCase Description 18](#_Toc14854)

[Table 7 : Add to Cart UseCase Description 18](#_Toc27051)

[Table 8 : Manage Product UseCase Description 19](#_Toc19735)

[Table 9 : Making the import product form UseCase Description 20](#_Toc23508)

[Table 10 : Manage Staff UseCase Description 21](#_Toc23326)

[Table 11 : Counsel Online UseCase Description 22](#_Toc1221)

[Table 12 : Class Order Description 24](#_Toc11222)

[Table 13 : Class Product Description 24](#_Toc24569)

[Table 14 : Class ImportForm Description 25](#_Toc25710)

[Table 15 : Class Category Description 25](#_Toc8181)

[Table 16 : Class Person Description 26](#_Toc9458)

[Table 17 : Class Customer Description 26](#_Toc23483)

[Table 18 : Class Brand Description 26](#_Toc26664)

# INTRODUCTION

In the current era, information technology is applied in almost all fields from production and manufacturing to services and entertainment, contributing to the development of the economy. In particular, everyone today owns at least one smartphone to connect to the internet. By using the internet, people are also gradually moving from shopping directly from stores to online shopping, where they can view product information quickly and with just a few clicks: touch on a smartphone, a click on a computer can make a product order, just wait a while for the product they ordered will be delivered to their home. It is because of the convenience and speed of those online stores that the demand for an app is heightened.

In particular, an app to order smartphone products is indispensable in the current trend. The App fully meets the needs of viewing phone product details, models, production information as well as configuration information to help customers see clearly and based on needs to make an order for themselves. new smartphone through online credit card payment or payment confirmation upon receipt of the product. At the same time, customers can also choose for themselves an online consultant provided by the store's staff, answer questions related to products, ensure full service and make customers satisfied. . Not only helps customers to shop for mobile phones easily, the website also makes it easier for store employees and managers to manage mobile phone products, manage employees, Manage the stages to bring products to customers as well as provide relevant information and latest announcements to customers.

Because of the above needs, it is necessary to build a convenient, easy-to-see, easy-to-use system that fully satisfies those needs.

# TEAM MEMBERS

|  |  |  |  |
| --- | --- | --- | --- |
| Ord | ID | Name | Email |
| 1 | 185170616 | Lê Duy Khánh | khanhld218@uef.edu.vn |
| 2 | 185050510 | Đặng Nguyễn Kim Long | longdnk18@uef.edu.vn |
| 3 | 185050082 | Đinh Văn Hoàn | hoandv18@uef.edu.vn |
| 4 | 185050810 | Văn Hoàng Phúc | phucvh18@uef.edu.vn |
| 5 | 185050875 | Nguyễn Nhật Khoa | khoann18@uef.edu.vn |
| 6 | 185050701 | Nguyễn Đức Trung | trungnd18@uef.edu.vn |

Table 1: Team Members

# OVERVIEW

## TARGET – SCOPE OF STUDY

### Target

Based on the knowledge learned from the Information Systems Design and Analysis and Mobile Programming subjects, our team implements the construction and design system for the mobile store app. The design system contains full and detailed information to build up a complete app, including: problem description, required function, map design and flow processing in the outgoing system. with special constraints. Through it, our team members can easily build products.

In addition, the group's goal through this project can be determined to think about the system and understand the processing processes of Android Studio project structure to build a complete Mobile Application. We also know how to use tools to design and represent processing flows through maps, thereby expanding analysis and writing system specifications.

### Scope of Study

* The project is implemented on the basis of the subjects’ area Analysis and design of information systems and Mobile Programming.
* The scope of the project is limited to the analysis and design of the system and build up the complete Mobile Store app.

## PROJECT DESCRIPTION

**Customers** will use app to view and order. Order information: number of turns, order date, customer information, items, full name, phone number and recipient's address.

When ordering will require login, registration information to log in includes: account, and password, and declare additional information such as full name, customer name, phone number, address and email. They can also chat with the chatbot for problem supporting.

**Salesman** will approve the order and print the order to deliver to the delivery staff. The salesperson will confirm the order completion after the delivery staff submits the order with the recipient's confirmation. Additional information in the order includes the time, date of receipt and delivery personnel.

Check the warehouse and inform the manager to add more goods.

**The manager will perform the following tasks:**

* Manage product groups, information includes: group code, group name (such as iOS, Android), manufacturer information including company code, company name (such as Apple, Samsung, Nokia, ...) and products include product code, product name, capacity, memory, screen size, selling price, quantity in stock.
* Employee management, employee information including employee code, last name, employee name, phone number, address, job (sales, delivery, ...)
* Make a phone entry form, the information of the entry ticket includes the number of the entry form, the date of entry, the total amount and the input details including the phone information, the quantity entered, and the unit price.

# SOLUTION STACK

## Android Studio

According to <https://developer.android.com/>: Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA . On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

* A flexible Gradle-based build system.
* A fast and feature-rich emulator.
* A unified environment where you can develop for all Android devices.
* Apply Changes to push code and resource changes to your running app without restarting your app.
* Code templates and GitHub integration to help you build common app features and import sample code.
* Extensive testing tools and frameworks.
* Lint tools to catch performance, usability, version compatibility, and other problems.
* C++ and NDK support.
* Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine.

Each project in Android Studio contains one or more modules with source code files and resource files. Types of modules include:

* Android app modules.
* Library modules.
* Google App Engine modules.

By default, Android Studio displays your project files in the Android project view. This view is organized by modules to provide quick access to your project's key source files. All the build files are visible at the top level under Gradle Scripts and each app module contains the following folders:

* ***manifests:*** Contains the AndroidManifest.xml file.
* ***java:*** Contains the Java source code files, including JUnit test code.
* ***res:*** Contains all non-code resources, such as XML layouts, UI strings, and bitmap images.



Figure 1: The project files in Android view.

## Firebase

Firebase is a toolset to “build, improve, and grow your app”, and the tools it gives you cover a large portion of the services that developers would normally have to build themselves, but don’t really want to build, because they’d rather be focusing on the app experience itself. This includes things like analytics, authentication, databases, configuration, file storage, push messaging, and the list goes on. The services are hosted in the cloud, and scale with little to no effort on the part of the developer.

This is different than traditional app development, which typically involves writing both frontend and backend software. The frontend code just invokes API endpoints exposed by the backend, and the backend code actually does the work. However, with Firebase products, the traditional backend is bypassed, putting the work into the client. Administrative access to each of these products is provided by the Firebase console.

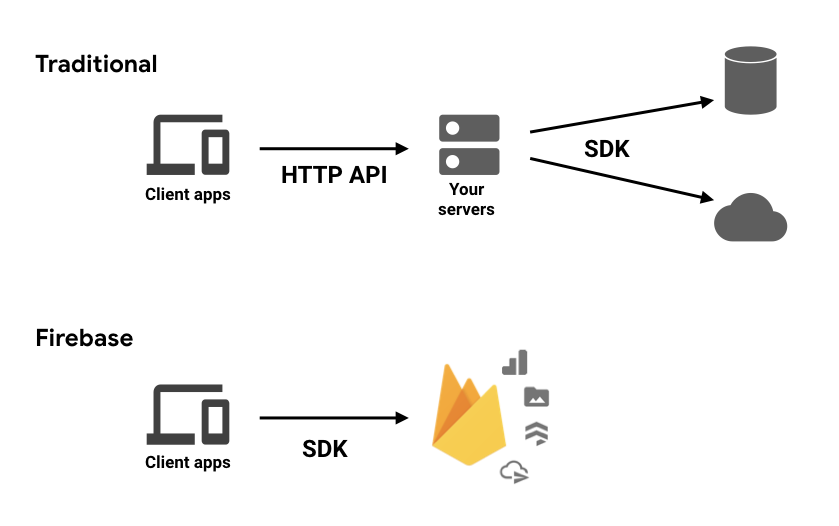


Figure 2: Traditional Database and FireBase Database

## Chatbot

At the most basic level, a chatbot is a computer program that simulates and processes human conversation (either written or spoken), allowing humans to interact with digital devices as if they were communicating with a real person. Chatbots can be as simple as rudimentary programs that answer a simple query with a single-line response, or as sophisticated as digital assistants that learn and evolve to deliver increasing levels of personalization as they gather and process information.

***There are two main types of chatbots:***

* Task-oriented (declarative) chatbots are single-purpose programs that focus on performing one function. Using rules, NLP, and very little ML, they generate automated but conversational responses to user inquiries. Interactions with these chatbots are highly specific and structured and are most applicable to support and service functions—think robust, interactive FAQs. Task-oriented chatbots can handle common questions, such as queries about hours of business or simple transactions that don’t involve a variety of variables. Though they do use NLP so end users can experience them in a conversational way, their capabilities are fairly basic. These are currently the most commonly used chatbots.
* Data-driven and predictive (conversational) chatbots are often referred to as virtual assistants or digital assistants, and they are much more sophisticated, interactive, and personalized than task-oriented chatbots. These chatbots are contextually aware and leverage natural-language understanding (NLU), NLP, and ML to learn as they go. They apply predictive intelligence and analytics to enable personalization based on user profiles and past user behavior. Digital assistants can learn a user’s preferences over time, provide recommendations, and even anticipate needs. In addition to monitoring data and intent, they can initiate conversations. Apple’s Siri and Amazon’s Alexa are examples of consumer-oriented, data-driven, predictive chatbots.

# USE CASE VIEW

## USE CASE DIAGRAM

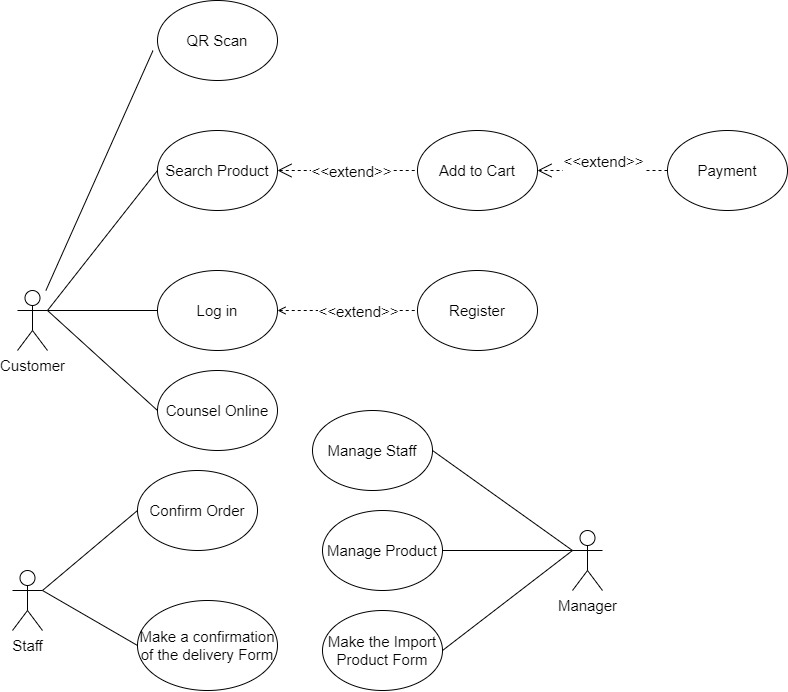


Figure 3: Usecase Diagram

## USECASE DESCRIPTION

### UC-01 Payment

|  |  |
| --- | --- |
| Use Case ID | UC-01 |
| Use Case | Payment |
| Description | Customer is redirected to the Payment page. There are 2 way for payment: payment on receipt and payment by credit card. The customer is required to enter information to a form for payment. |
| Actor | Customer |
| Pre-conditions | - The customer must log in for payment.  - The order must contain at least one product or more.  - The provided information must be valid |
| Flow | Basic   1. The customer chooses the payment feature in the Cart. 2. If the customer is not logged in, they will be redirected to Login page, 3. Choose a way for payment: payment on receipt or payment by credit card. 4. Fill in the information to a form 5. Confirm 6. The System shows the payment status to the Customer and redirects to the order management page 7. (Optional) Add a bonus to Customer |
| Alternative  - The customer can go to the shop to pay and receive their order |
| Exception  - Confirm Credit Card failed  - The balance in credit card is not enough for payment  - The System cannot process the payment feature due to an error (example: in maintenance, the product is out of stock, …)  - The customer cannot receive any bonus |
| Post-conditions | - The customer payment successfully |

Table 2: Payment Usecase Description

### UC-02 Make a confirmation of delivery form

|  |  |
| --- | --- |
| Use Case ID | UC-02 |
| Use Case | Make a confirmation of delivery form |
| Description | The salesman will be make a confirmation of order completion after the shipper submits a confirmation of recipient that they got the ordered product |
| Actor | Staff |
| Pre-conditions | - The order must be in the delivered status |
| Flow | ***Basic:***   1. The staff choose the confirming order feature 2. Choose orders that is in Delivered Status. 3. Confirm making the confirmation of delivery form 4. The system will be store the information of form, including time, receive date, and Shipper 5. Show the notification |
| ***Alternative:***  - If the feature is not performed, the staff can fill in the information by hand and send it to the system |
| ***Exception:***  - Confirming Making form doesn’t work  - The system doesn’t store the information although it showed the success notification. |
| Post-conditions | - Features work  - Making the confirmation of delivery form successfully |

Table 3: Make a confirmation of delivery form UseCase Description

### UC-03 Login

|  |  |
| --- | --- |
| Use Case ID | UC-03 |
| Use Case | Login |
| Description | The customer logs in to the System by an account that is registered to do some features that require confirming the customer account. |
| Actor | Customer |
| Pre-conditions | - The customer registered an account |
| Flow | ***Basic:***   1. Choose Login feature 2. Enter account and password 3. Confirm Login 4. Show the notification that the customer log in successfully or not |
| ***Alternative***  - Not existing |
| ***Exception:***  - Login failed  - Connecting to the System failed |
| Post-conditions | - Login successfully |

Table 4: Login UseCase Description

### UC-04 Register

|  |  |
| --- | --- |
| Use Case ID | UC-04 |
| Use Case | Register |
| Description | The customer request to register a new account to confirm the information in order to do some feature that need to the customer is authenticated |
| Actor | Customer |
| Pre-conditions | none |
| Flow | ***Basic:***   1. The customer chooses the register feature to create a new account. 2. The customer enters some necessary information the System require 3. Confirming register a new account 4. Show the notification that registers successfully or failed |
| ***Alternative:***  - not Existing |
| ***Exception:***  - Register failed  - The information the customer entered is not in the right format  - The customer didn’t enter some required information  - The customer registered a new account that existing in the system |
| Post-conditions | - Show the notification that Register successfully or failed  - Store the customer’s information to the system to finish registering |

Table 5: Register UseCase Description

### UC-05 Search product

|  |  |
| --- | --- |
| Use Case ID | UC-05 |
| Use Case | Search product |
| Description | The customer wants to search for some product with some information the customer provided to the system and then, show the suited product list to the customer |
| Actor | Customer |
| Pre-conditions | none |
| Flow | ***Basic:***   1. The customer enters a keyword/string or selects some available condition the system provided. 2. The system gets the conditions/ keywords/string and shows the suited product list |
| ***Alternative:***  - Not existing |
| ***Exception:***  - Connecting to the system is failed |
| Post-conditions | - Show to the customer the suited product list |

Table 6: Search Product UseCase Description

### UC-06 Add To Cart

|  |  |
| --- | --- |
| Use Case ID | UC-06 |
| Use Case | Add product to cart |
| Description | Allow customers to save products they want to buy when viewing on the website. When the customer presses the add to cart button, the item will be added to the cart. Customers can browse the shopping cart to see the selected products, the total price and can add and remove the number of products in the basket. |
| Actor | Customer |
| Pre-conditions | none |
| Flow | ***Basic:***   1. The customer selects the product wants to add to the cart. 2. Customer enter the number of products they want to buy and the promotion code if any. 3. The system adds, quantities, and promotional information to the existing customer’s cart. 4. Notice successfully added to cart to Customer |
| ***Alternative:***  - Not existing. |
| ***Exception:***  - Products displayed in stock while the system has processed is out of stock.  - The system does note save the product to the cart. |
| Post-conditions | - Add to the Customer’s cart a new product that the customer. |

Table 7: Add to Cart UseCase Description

### UC-07 Manage product

|  |  |
| --- | --- |
| Use Case ID | UC-07 |
| Use Case | Manage product |
| Description | Managers handle product data (add, edit, delete product information), add new products as well as modify and update product information. |
| Actor | Manager |
| Pre-conditions | Log in to your account and authenticate as Manager. |
| Flow | ***Basic:***   1. The manager enters the product management category. 2. View the list of products in the system, update and add new products. 3. Confirm save the modifications. 4. The system updates the data. 5. Update success/failed notification. 6. Update the system to display the product list again. |
| ***Alternative:***  - Not existing. |
| ***Exception:***  - Update new product information that coincides with other existing products in the system.  - System connection failed.  - User authentication as manager failed. |
| Post-conditions | - Update product information in the system. |

Table 8: Manage Product UseCase Description

### UC-08 Making the import product form

|  |  |
| --- | --- |
| Use Case ID | UC-08 |
| Use Case | Making the import product form |
| Description | The manager confirms the goods import and enters the information about the import into the goods receipt, stores it in the system and updates the product just entered into the system. |
| Actor | Manager |
| Pre-conditions | Log in to your account and authenticate as Manager. |
| Flow | ***Basic:***   1. The manager chooses the feature to fill in the order form. 2. The manager fills in the input form with the information provided by the system. 3. The system confirms the successful entry form. 4. The system saves the import slip. 5. Product management inventory update system. 6. Notification of successful vote click. |
| ***Alternative:***  - Not existing. |
| ***Exception:***  - Enter the information in the wrong format.  - System connection failed.  - User authentication as manager failed. |
| Post-conditions | - Update new products imported into the system. |

Table 9: Making the import product form UseCase Description

### UC-09 Manage Staff

|  |  |
| --- | --- |
| Use Case ID | UC-09 |
| Use Case | Manage Staff |
| Description | Employee management, employee information including employee code, last name, employee name, phone number, address, job (sales, delivery, ...). |
| Actor | Manager |
| Pre-conditions | The person performing this function must be the Manager. |
| Flow | ***Basic:***   1. Sign in with your manager account. 2. Select employee management. 3. Select update employee information. 4. Press the save button. 5. The system sends notifications to all employees. |
| ***Alternative:***  - Not existing. |
| ***Exception:***  - Unable to access the employee management section even though the manager account is used.  - Unable to update information.  - The system does not send notifications to employees. |
| Post-conditions | - The functions work normally. |

Table 10: Manage Staff UseCase Description

### UC-10 Counsel Online

|  |  |
| --- | --- |
| Use Case ID | UC-10 |
| Use Case | Counsel Online |
| Description | Customers can message through the chat window of the website. The system will connect the chatbot to respond to customer inquiries. |
| Actor | Customer - ChatBot |
| Pre-conditions | - Not existing. |
| Flow | ***Basic:***   1. Customers click on the chat icon on the website. 2. The chat window pops up. 3. Customers enter the information they want to ask. 4. Chat bot to answer questions. |
| ***Alternative:***  - Call the store’s hot line directly for advice. |
| ***Exception:***  - ChatBot is not working.  - Did not receive a message from either party even though the message was sent successfully. |
| Post-conditions | - Customers have questions answered. |

Table 11: Counsel Online UseCase Description

# LOGICAL VIEW

## CLASS DIAGRAM

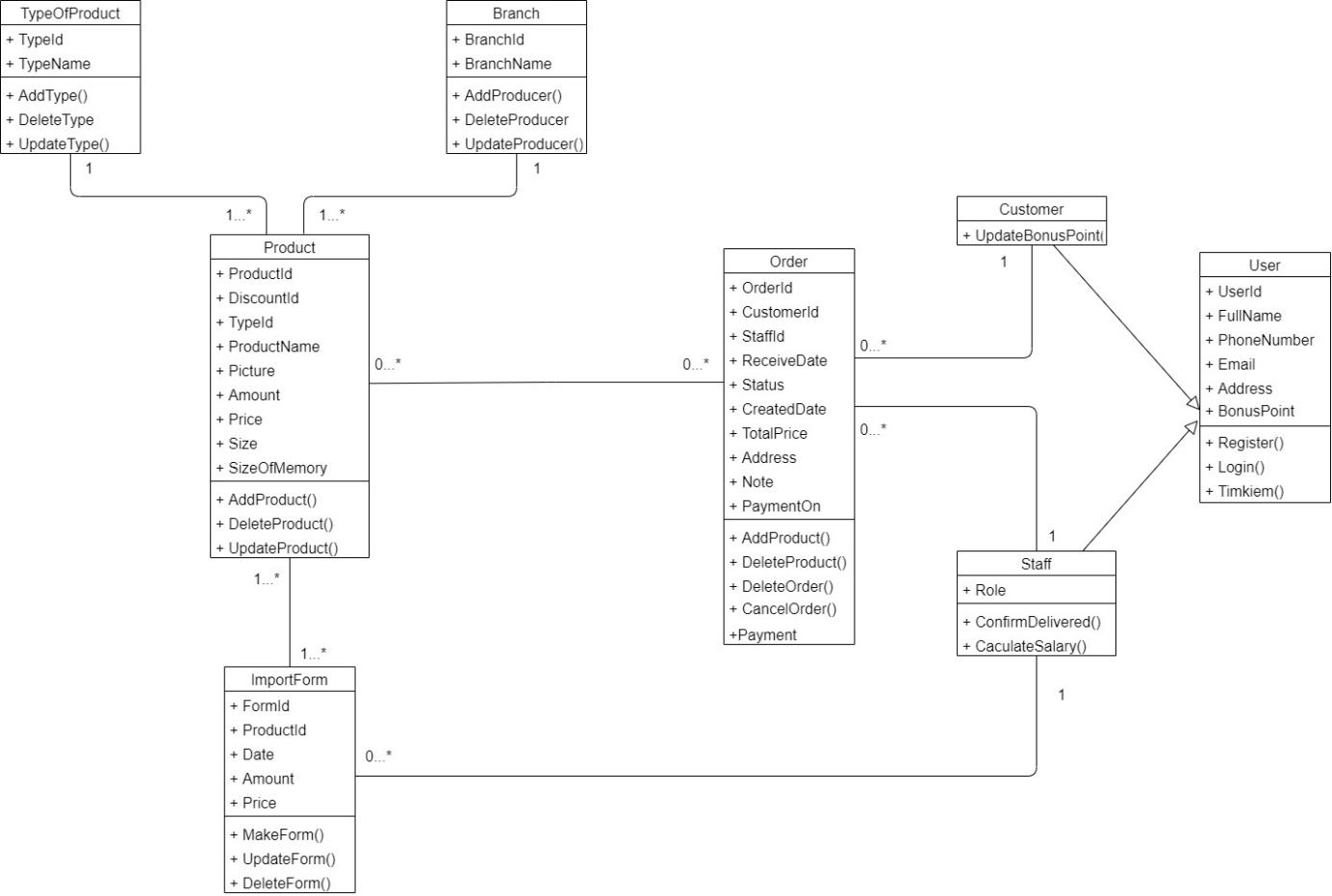


Figure 4: Class Diagram

## CLASS DESCRIPTION

### Order

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ord | Field Name | Description | Type | Key |
| 1 | OrderId | Order’s Id | Int | Primary |
| 2 | CustomerId | Customer’s Id | Int | Foreign |
| 3 | StaffId | Staff’s Id | Int | Foreign |
| 4 | ReceiveDate | The received date | Date |  |
| 5 | Status | Order’s Status | Nvarchar(32) |  |
| 6 | CreatedDate | The created Date | Date |  |
| 7 | TotalPrice | The total price | Float |  |
| 8 | CustomerAddress | The Address that customer receives the Product | Nvarchar(255) |  |
| 9 | Note | The customer’s Note | Nvarchar(255) |  |
| 10 | PaymentMethod | The method of Payment (on receipt or Credit card) | Nvarchar(32) |  |
| 11 | AddProduct() | Add a new product | Method |  |
| 12 | DeleteProduct() | Delete an existing product in the Order | Method |  |
| 13 | DeleteOrder() | Delete an existing order | Method |  |
| 14 | CancelOrder() | Cancel an order | Method |  |
| 15 | Payment() | Process Payment | Method |  |

Table 12: Class Order Description

### Product

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ord | Field Name | Description | Type | Key |
| 1 | ProductId | The product’s Id | Int | Primary |
| 2 | DiscountId | The discount’s Id | Int | Foreign |
| 3 | TypeId | The type of Product Id | Int | Foreign |
| 4 | ProductName | The product’s Name | Nvarchar(255) |  |
| 5 | Picture | The product’s picture | Nvarchar(5000) |  |
| 6 | Amount | The amount of Product | Int |  |
| 7 | Price | The price of one product | Float |  |
| 8 | Size | The product’s size | Int |  |
| 9 | SizeOfMemory | The size of Memory | Int |  |
| 10 | AddProduct() | Add a new product | Method |  |
| 11 | DeleteProduct() | Delete an existing product | Method |  |
| 12 | UpdateProduct | Update an existing product | Method |  |

Table 13: Class Product Description

### ImportForm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ord | Field Name | Description | Type | Key |
| 1 | FormId | The Form’s Id | Int | Primary |
| 2 | ProductId | The Product’s Id | Int | Foreign |
| 3 | DateIn | The import product Date | Datetime | Foreign |
| 4 | Amount | The Amount of Product | Int |  |
| 5 | Price | The price of one product | Float |  |
| 6 | MakeForm() | Make a new form | Method |  |
| 7 | UpdateForm() | Update an existing form | Method |  |
| 8 | DeleteForm() | Delete an existing form | Method |  |

Table 14: Class ImportForm Description

### Category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ord | Field Name | Description | Type | Key |
| 1 | CategoryId | Category ID | Int | Primary |
| 2 | CategoryName | Name of Category | Int | Foreign |
| 3 | Add() | Add new Category | Method |  |
| 4 | Delete() | Delete Category | Method |  |
| 5 | Update() | Update existing Category | Method |  |

Table 15: Class Category Description

### Person

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ord | Field Name | Description | Type | Key |
| 1 | Id | Id | Int | Primary |
| 2 | Name | Name | Nvarchar(50) |  |
| 3 | Phone | Phone | Nvarchar(13) |  |
| 4 | Email | Email | Nvarchar(50) |  |
| 5 | Address | Address | Nvarchar(500) |  |
| 6 | Bonuspoints | Bonus Poitnts | Int |  |
| 7 | Register() | Register new account | Method |  |
| 8 | Login() | Login | Method |  |
| 9 | Search() | Search for product | Method |  |

Table 16: Class Person Description

### Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ord | Field Name | Description | Type | Key |
| 1 | UpdateBonusPoints | Update the bonus point when ordering | Method |  |

Table 17: Class Customer Description

### Employee

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ord | Field Name | Description | Type | Key |
| 1 | position | Position | Nvarchar(50) |  |
| 2 | ConfirmOrder() | Confirm the Order | Method |  |
| 4 | CalculateSalary() | Calcalate the Salary | Method |  |

Figure 5: Class Employee Description

### Brand

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ord | Field Name | Description | Type | Key |
| 1 | BrandId | Brand ID | Int | Primary |
| 2 | BrandName | Name of the Brand | Nvarchar(50) |  |
| 3 | Add() | Add new Branch | Method |  |
| 4 | Delete() | Delete existing Brand | Method |  |
| 5 | Update() | Update existing Brand | Method |  |

Table 18: Class Brand Description

# PROCESS VIEW

## SEQUENCE DIAGRAM

### Login

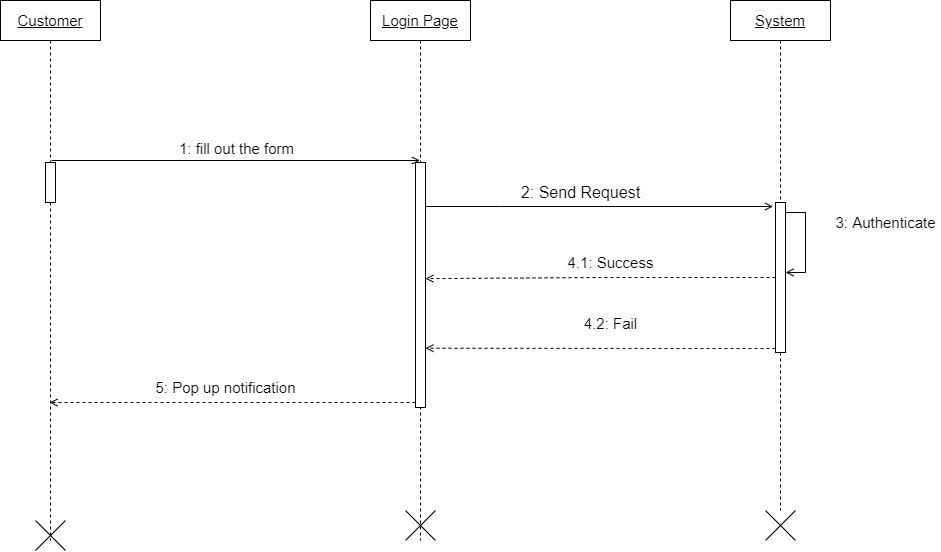


Figure 6: Login Sequence Diagram

### Register

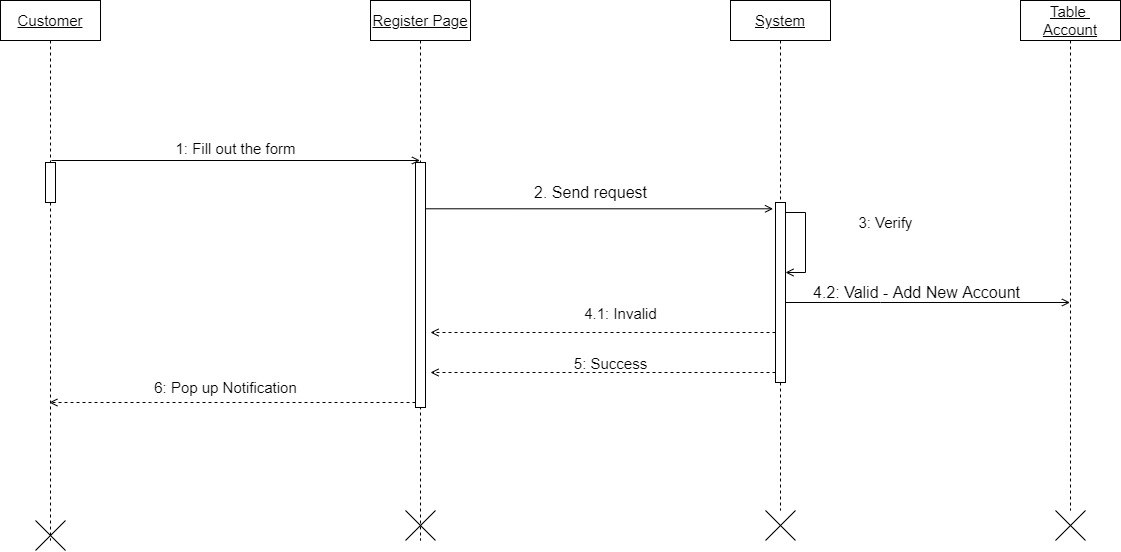


Figure 7: Register Sequence Diagram

### Create Warehouse Receipt Form

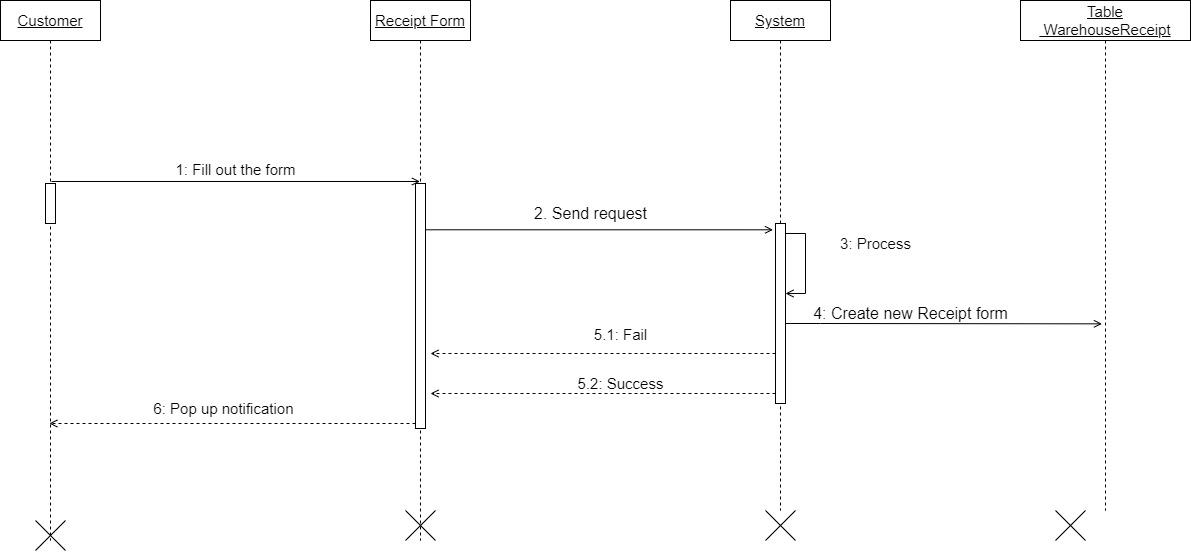


Figure 8: Create Warehouse Receipt Form Sequence Diagram

### Process Payment

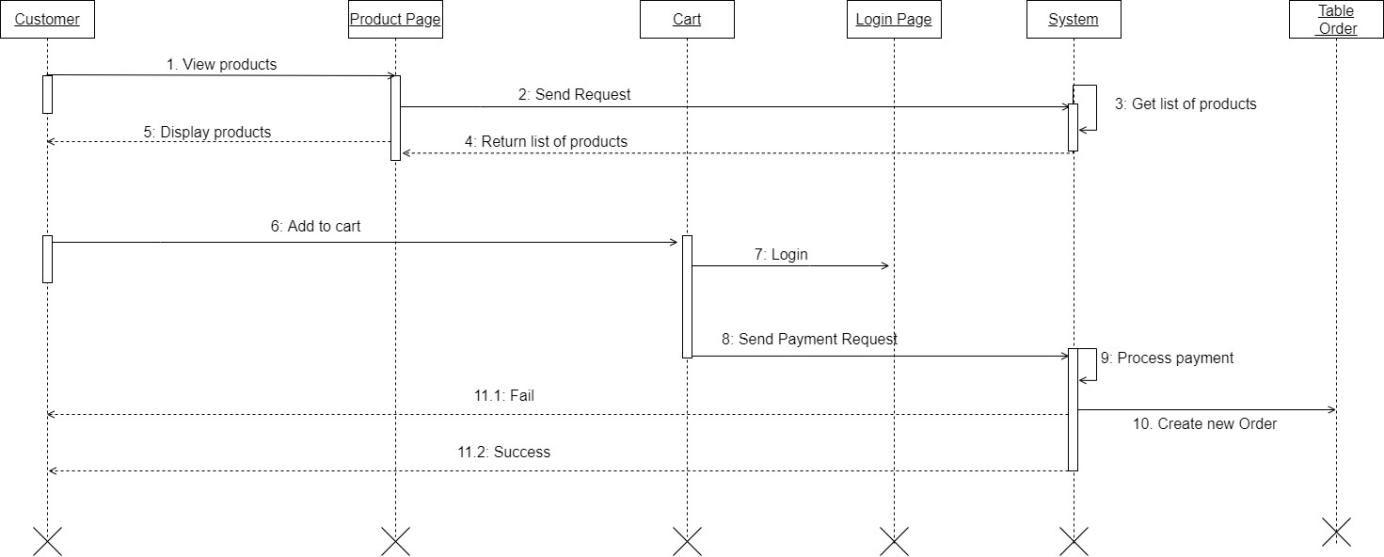


Figure 9: Process Payment Sequence Diagram

## COLLABORATION DIAGRAM

### Process Payment

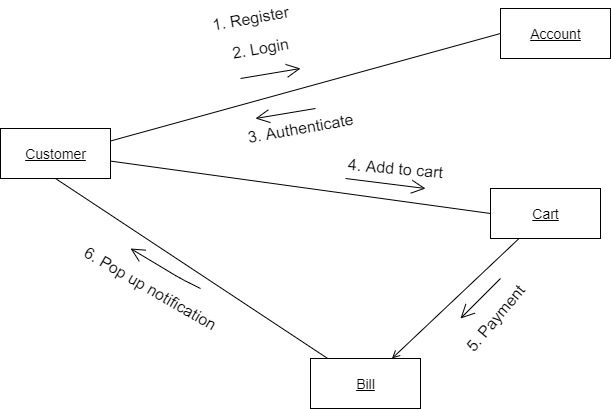


Figure 10: Process Payment Collaboraion Diagram

### Process Order

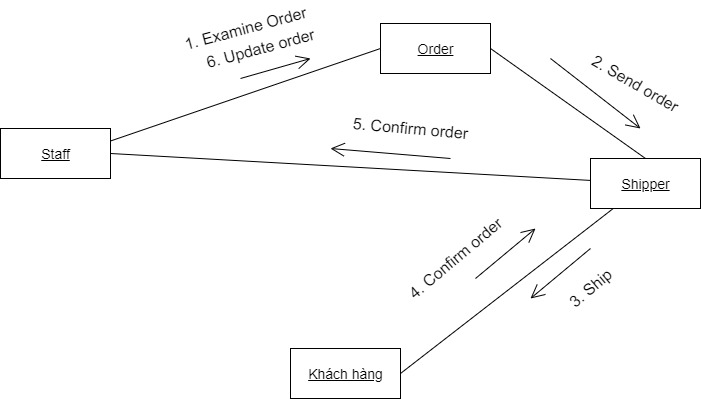


Figure 11: Process Order Collaboraion Diagram

## STATE DAIGRAM

### Cart

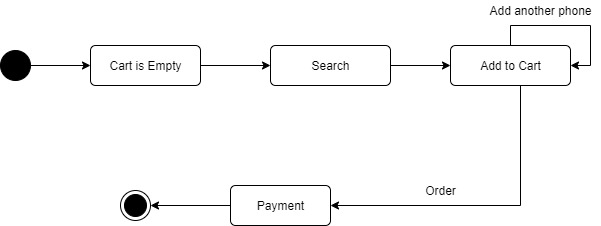


Figure 12: Cart State Diagram

### Order

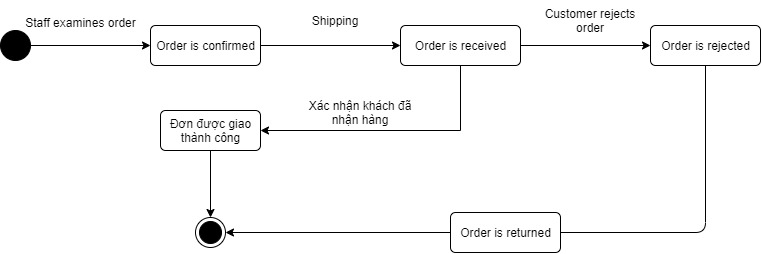


Figure 13: Order State Diagram

## ACTIVITY DAIGRAM

### Login

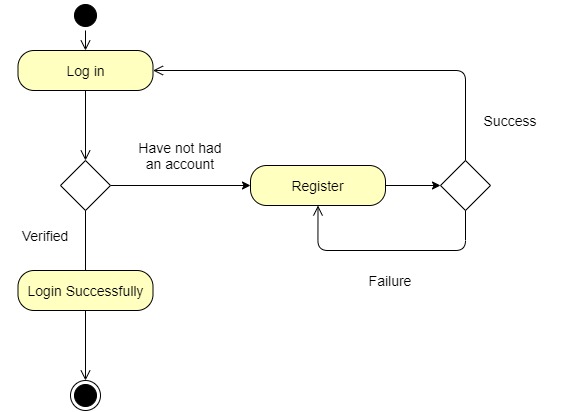


Figure 14: Login Activity Diagram

### Process Payment

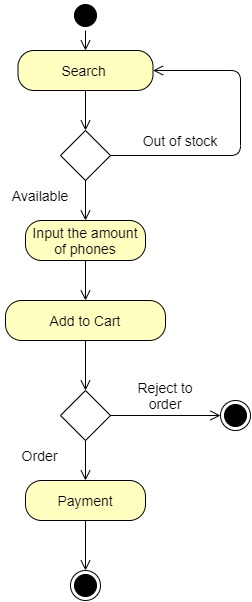


Figure 15: Process Payment Activity Diagram

# IMPLEMENTATION VIEW

## COMPONENT DIAGRAM

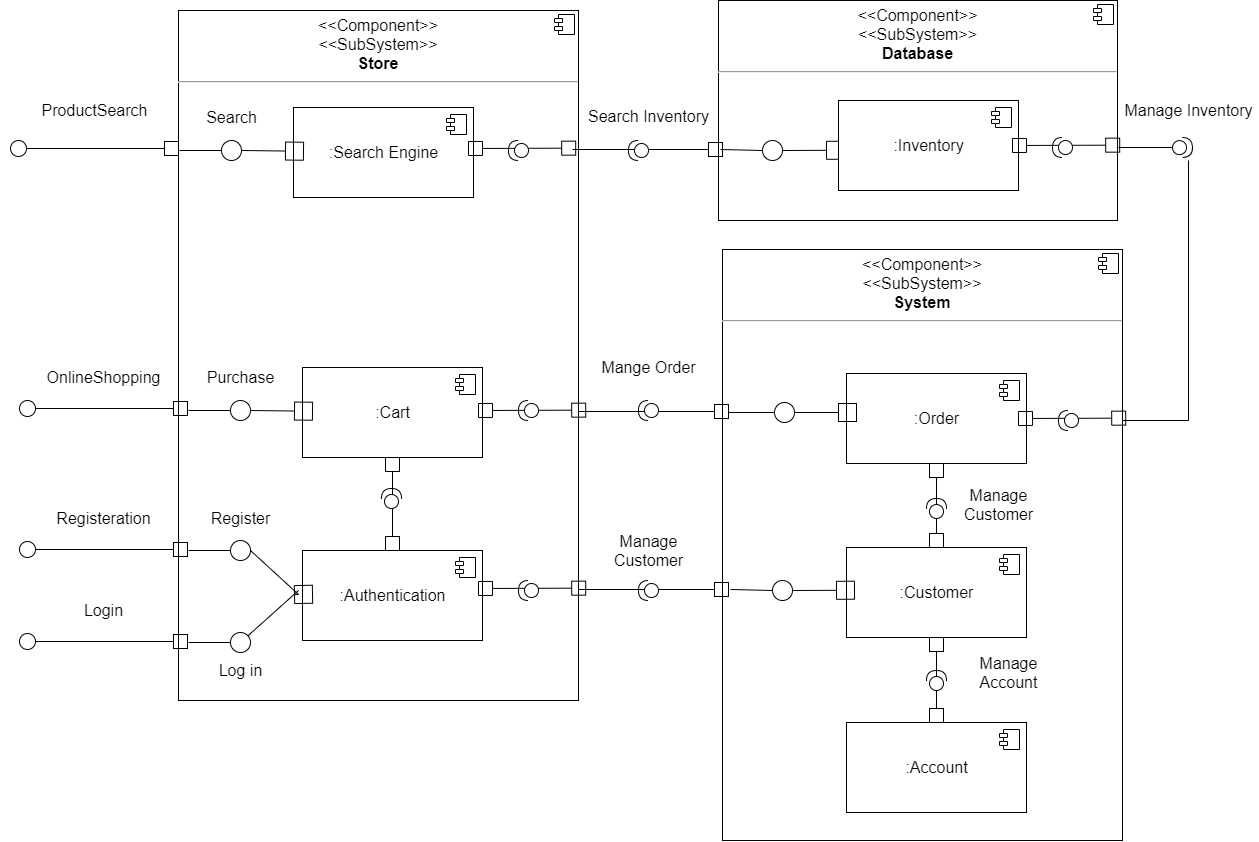


Figure 16: Component Diagram

## DEMO

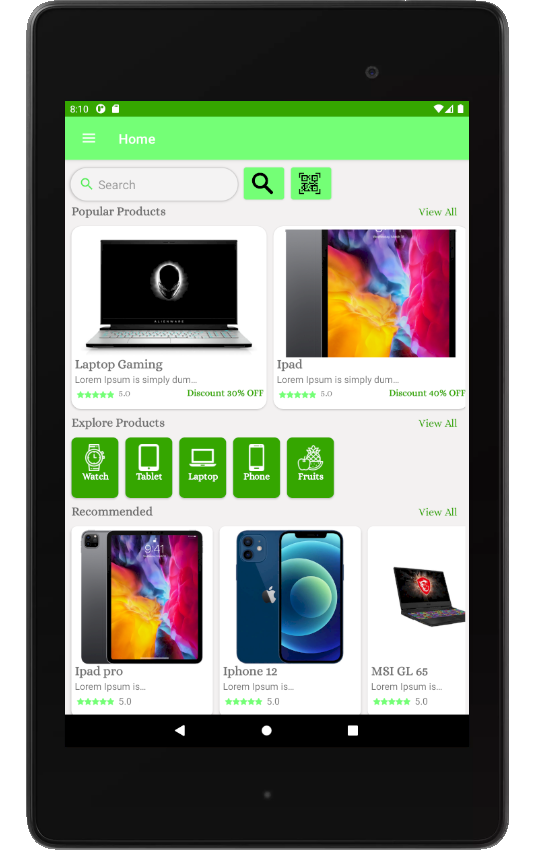


Figure 17: Main screen of the application

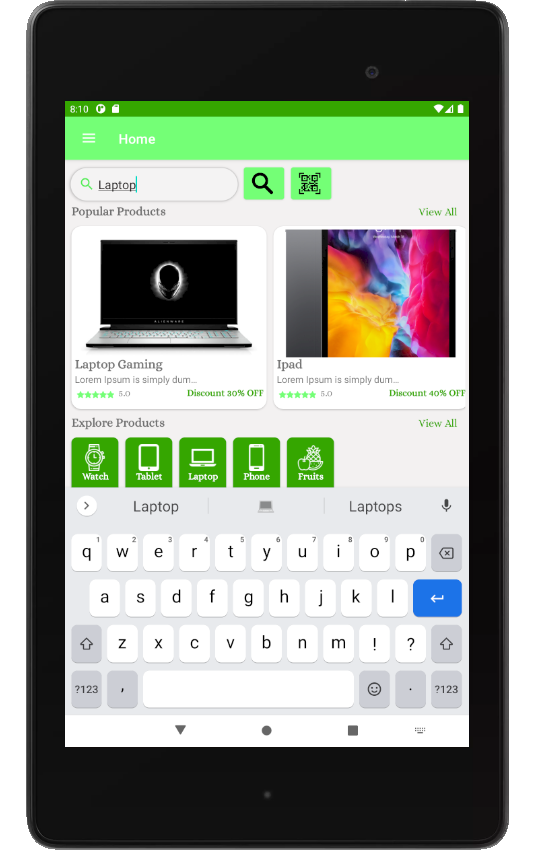


Figure 18: Search Product

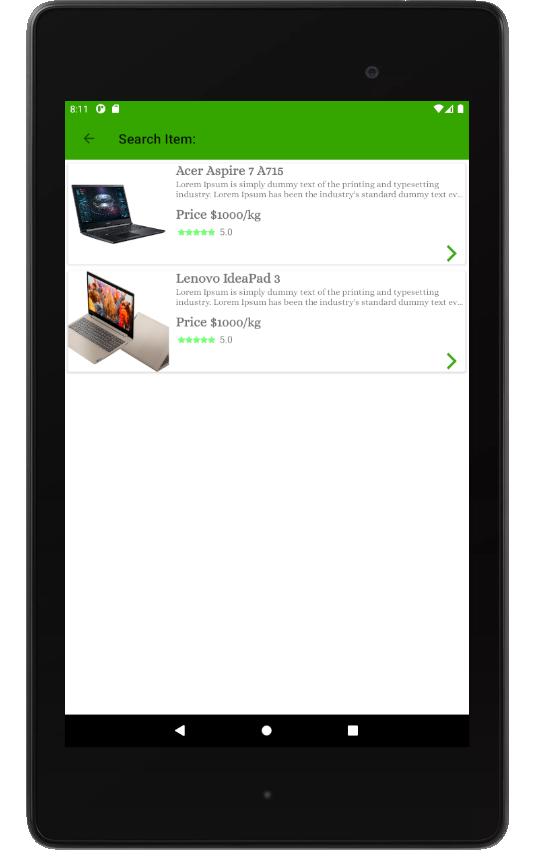


Figure 19: Searched Items

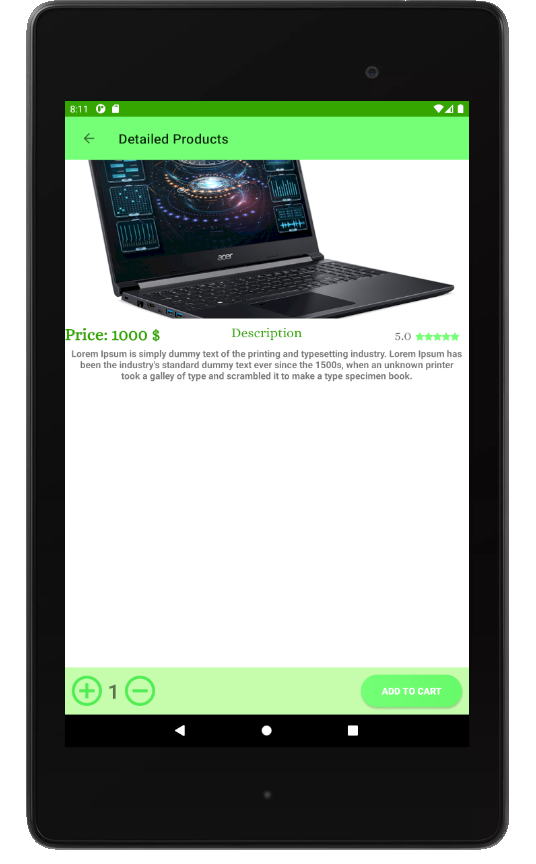


Figure 20: Product Details

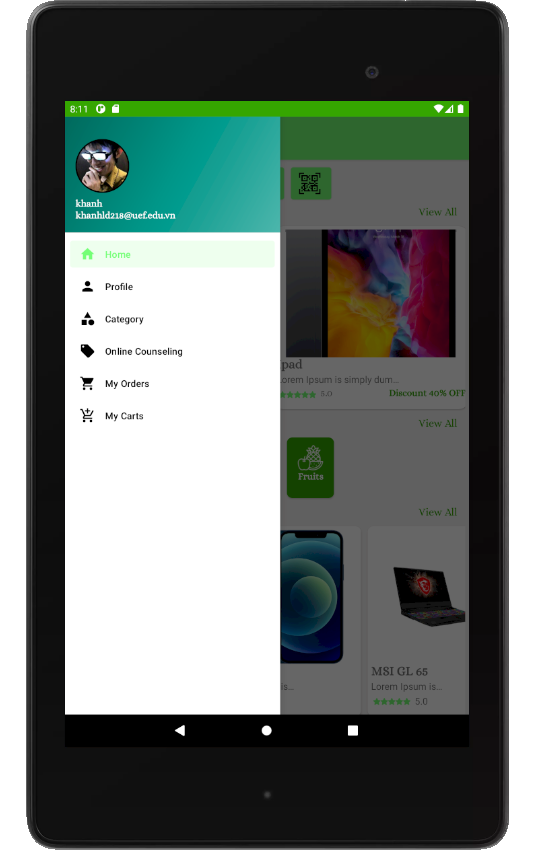


Figure 21: Navigation Bar

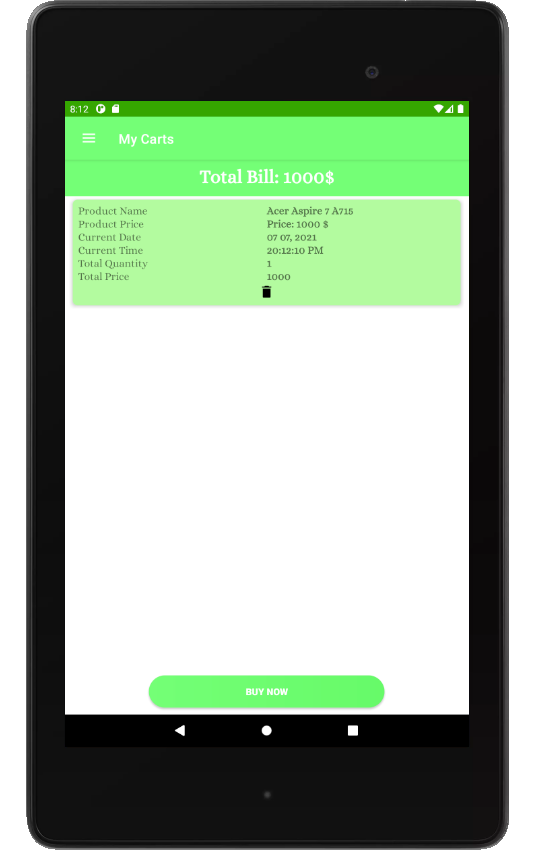


Figure 22: Cart

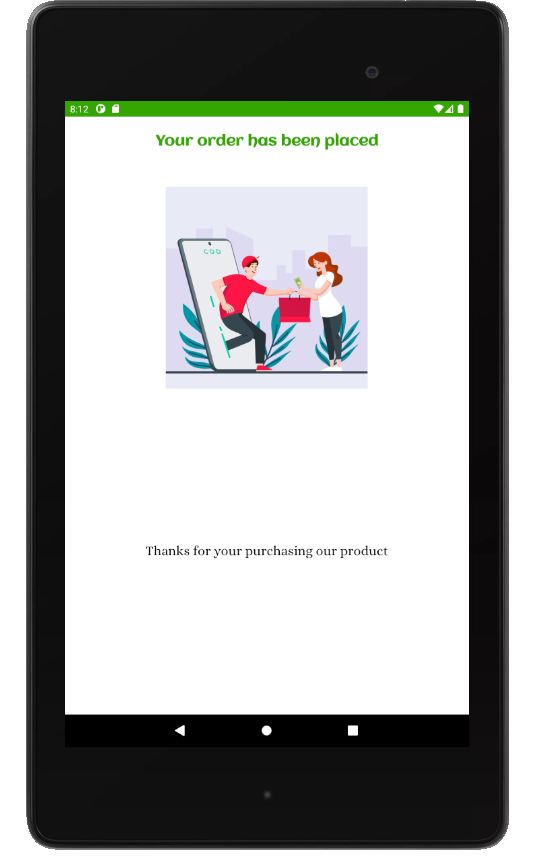


Figure 23: Payment Notification

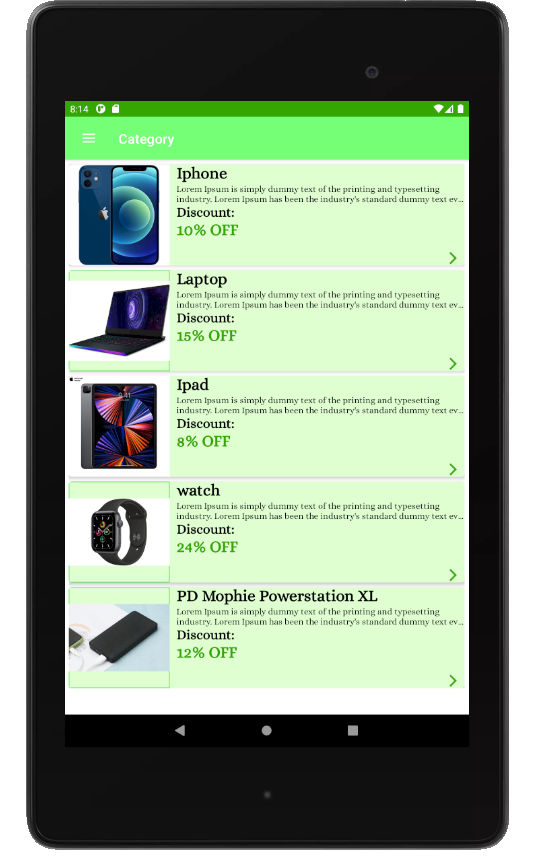


Figure 24: Category Filter

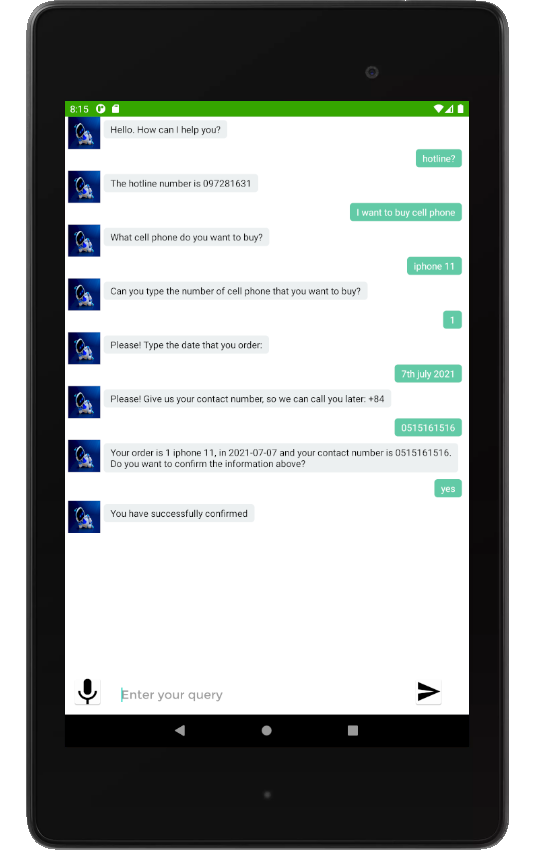
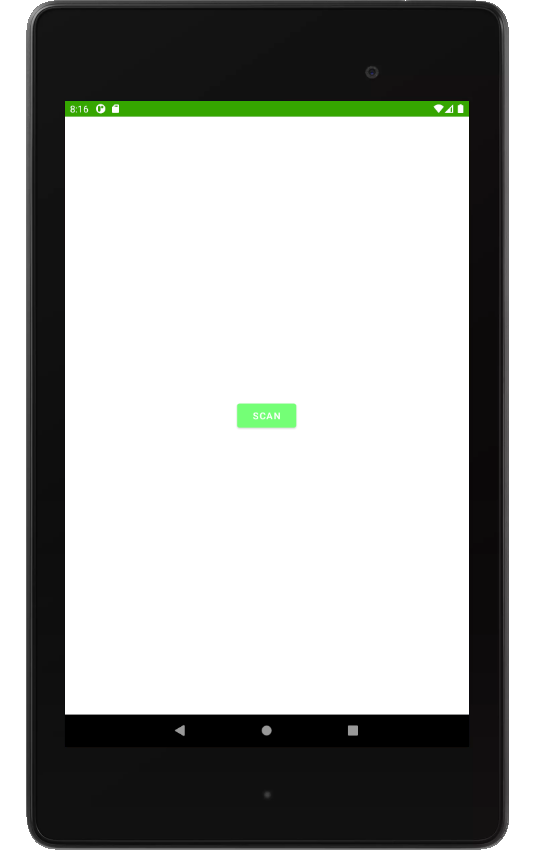


Figure 25: Chatbot for Online Counseling



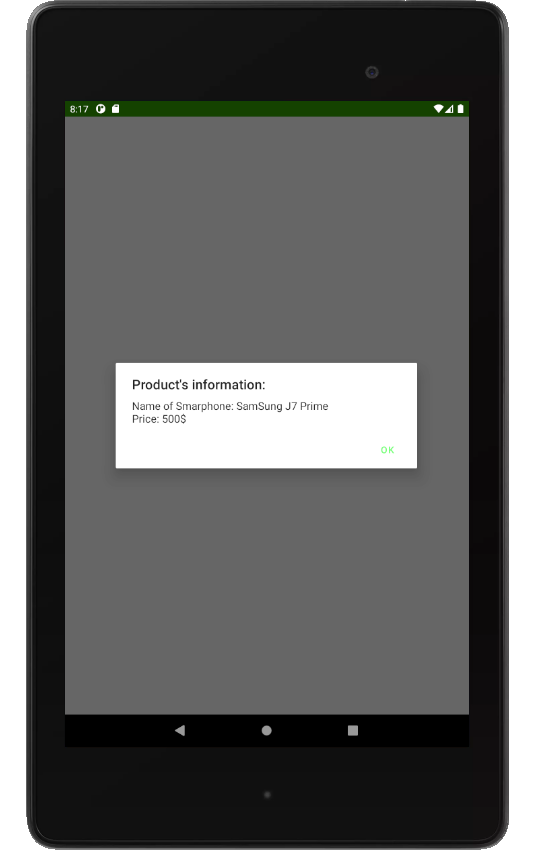


Figure 26: QR Code Scanning

# CONCLUSION

## Result

After the implementation of the project, the program has been completed and achieved the following results:

* Understand the process of selling mobile phones online
* Successfully building an app to sell mobile phones to meet the needs of consumers.
* Learn and understand quite well about tools to build apps like: Android studio. - The interface of the program is friendly, easy to use,
* App has helped users save time and effort to get a satisfactory product. Easily manage user details, information about products and orders of product buyers, from which it will be convenient for payment and delivery.
* Perform software testing at programming level.

## Limitations of the topic

* It is not possible to update all the products currently on the market.
* App has not been tested on the Internet widely.

## The development direction of the topic

With their own efforts, the project team tried to complete the requirements of the topic. Due to limited time and capacity, the group's app only goes deep into the function of selling products. The group of topics oriented to develop the app into a professional sales app. Provide a full range of products currently on the market at reasonable and affordable prices. Accompanying sales are the most prestigious and quality services to serve customers.

## Suggest opinions

During the implementation of this project, the project team cannot avoid shortcomings,

the project team is looking forward to receiving comments from teachers and friends

so that the group can better improve this online sales app.

# REFERENCES

1. <https://developer.android.com/studio/intro>.

2. <https://www.oracle.com/chatbots/what-is-a-chatbot/>.

3. <https://www.youtube.com/watch?v=G4rXpe9Wod8>.

4. <https://medium.com/firebase-developers/what-is-firebase-the-complete-story-abridged-bcc730c5f2c0>.