**Học viện Công nghệ thông tin và Thiết kế VTC**



PROJECT REPORT

Project name: CT STORE

Class: PF1118

Group: Group 3

Team members: Bui Quoc Cuong

Nguyen Tien Thanh

[CT Store](#_heading=h.gjdgxs) **1**

[Index](#_heading=h.30j0zll) **2**

[**I. Project Introduction**](#_heading=h.mf27dndfkrzy) **5**

[II. Analyze System Requirements](#_heading=h.3znysh7) **6**

[1. Use Case](#_heading=h.do2y7zl3b30a) 6

[1.1 Use Case](#_heading=h.dvae2oi9wxu4) 7

[1.2 Login description](#_heading=h.ynj7wd7fl57y) 8

[1.3 Search laptops description](#_heading=h.e0vbzm8oas41) 9

[1.5 View laptop details description](#_heading=h.rh395g40t1fg)  11

[1.6 Create order description](#_heading=h.7ta2p53jlqkg)  12,13

[1.7 Payment description](#_heading=h.7ta2p53jlqkg) 14.15

[2. Activity Diagram](#_heading=h.elrgafm9yds5) 16

[2.1 Login](#_heading=h.lntcuah3cf40) 16

[2.4 Search](#_heading=h.2pqd0zgmp8y1) mobile phone by id 19

[2.4 Create order](#_heading=h.2pqd0zgmp8y1) 21

[2.5 Payment](#_heading=h.gk73kvi933rx) 22

[III. Design Details](#_heading=h.2et92p0) **23**

[1. UI Design](#_heading=h.xzgg4ze4rz64) 23

[1.1 Login](#_heading=h.bmjlblwpxhba) 23

[1.2 Show](#_heading=h.qnd9c1omgub7) list mobile phone 24

[1.4 View](#_heading=h.wh6cbq47ip8y) detail mobile phone information 26

[1.6 Create](#_heading=h.uv5ewtbi95rl) account seller 28

[1.8 [Show](#_heading=h.qnd9c1omgub7) list mobile phone / 1.9 Create order](#_heading=h.icow6bof4cit)  30

[1.10 Payment](#_heading=h.5z3geh1pruu5) 31

[2. Code Design (Class Diagram)](#_heading=h.zfoucxek2qcv) 32

[3. Sequence Diagram](#_heading=h.hi7qd1aaxwml) 33

[3.1 Login](#_heading=h.q8e6qj2m3pt9) 33

[3.2 Search laptops](#_heading=h.5chv08pknqs3) by id 34

[3.4 Create order](#_heading=h.dyngqhsjirml) 35

[3.5 Payment](#_heading=h.nlxlz64309dt) 36,37

[4. Database Design](#_heading=h.km3xonbv4v4v) 38

[4.1 Entity Relationship Diagram](#_heading=h.4xnhlaj95v2e) 38

[4.2 Database Design Details](#_heading=h.2rcov8m5l3h7) 39,40,41

[IV.Test](#_heading=h.hr4laroibsit) **42**

[1. Login test](#_heading=h.e14ltbvf970m) 42

[1.1 Login test 1](#_heading=h.6guvu9eg10xw) 42

[1.2 Login test 2](#_heading=h.mndtiu316p2l) 43

[2. Search Mobile phone test](#_heading=h.1zkbohd537yd) 44

[2.1 Search by id test 1](#_heading=h.rboasyvlba2o) 44

[2.2 Search by id test 2](#_heading=h.l8plla9kzhnh) 45

[2.3 Search by operating system test 3](#_heading=h.91z67q5a5in3) 45

[2.4 Search by operating system test 4](#_heading=h.l3l9bh8x3l2) 46

[3. Change status account test](#_heading=h.ar1zbnhe3xgn) 47

[3.1 Change status account test 1](#_heading=h.k3fx3uyvtkyq) 47

[3.2 Change status account test 2](#_heading=h.5zquwntk4w0k) 48

[4. Create order test](#_heading=h.cyy0d54epngp) 49

[4.1 Create order test 1](#_heading=h.qgd61wo5ix7t) 49

[4.2 Create order test 2](#_heading=h.yr33f4o8vetm) 50

[V. TaskAssign (to each team member)](#_heading=h.1t3h5sf) **51**

1. Project introduction

The project provides a support system for the management of mobile phone stores, including basic features: + order creation, payment, order cancellation.

1. Proposed System

* Creating a program to Seller create order and payment.
* Creating a program for Managers to manage mobile phones, manage invoices, manage merchant accounts.

1. The scope of the project to be applied

* Make for actor are Seller and Manager.

1. System Name

* CT STORE

1. Deployment Environment

* Windows.

1. Development Tools

* Visual Studio Code
* Draw.io Diagram
* MySQL Workbench 8.0 CE

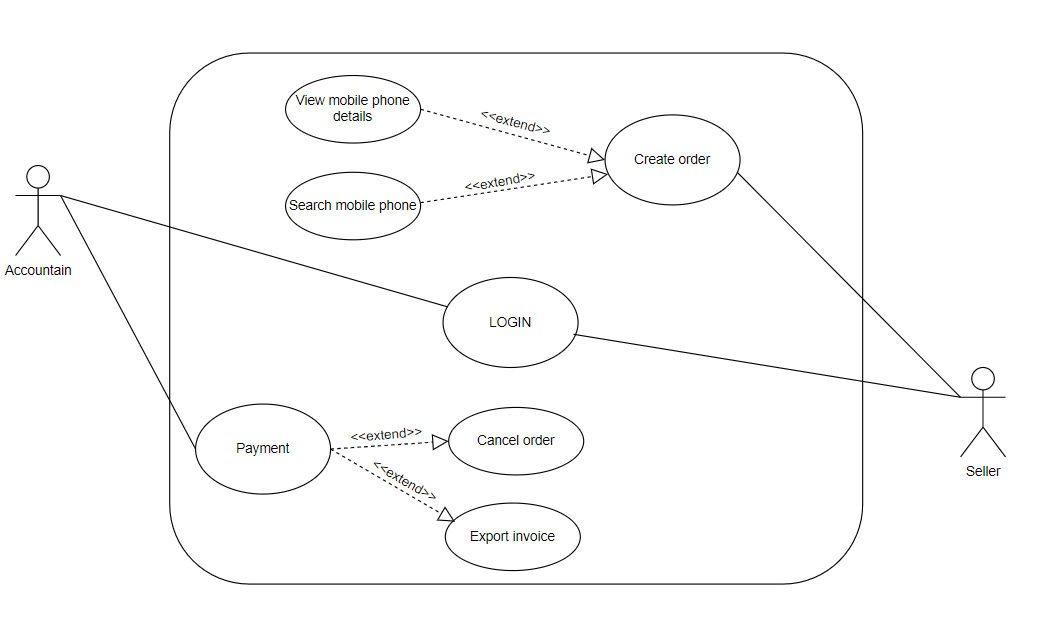
# Analyze System Requirements

## The system makes it easier for mobile phone shops to manage orders. It can help sellers get more information about mobile phones sold in the store and create

## orders according to buyer requirements. As for the manager, the system makes it easier for them to manage the store.

1. Use case

* 1. Use case



1.2 Login description

|  |  |
| --- | --- |
| Name | Login |
| ID | 1 |
| Description | Used to access the program and define the feature accessible depending on the type of account. |
| Actor | Seller and Manager |
| Organizational  Benefits | This feature was created to define the features and data that seller and manager can access in the program. |
| Trigger | Login will be automatically activated as soon as we start the program. |
| Preconditions | Must enter correct username and password, database must be connected, the manager account must be available, merchant account can be created upon login |
| Postconditions | Success: Seller or Manager have logged into the system, the main menu of the program must be displayed.  Fail: Seller or Manager haven’t logged into the system |
| Main Course | Input username and password  Verify login  Show main menu |
| Alternate Courses | AC 2.1: User enters wrong username, password  Display message error   1. Show the login screen again 2. Re-enter request   AC2.2: Username and password exists  Show successful login message  Show main menu |
| Exceptions | EX 1.1: Input Username/Password  Display: “Invalid username or password, please re-enter!” and ask to re-enter until correct  Request to re-enter username and password.  EX 2.1: Disconnect to database  Step 1: Display “Error!!! Can’t connect to database”  Request to recheck internet connection |

* 1. Search mobile phone by id description

|  |  |
| --- | --- |
| Name | Search mobile phone by id |
| ID | 2 |
| Description | Used to search for mobile phone |
| Actor | Seller |
| Organizational  Benefits | You can search mobile phone in the system |
| Trigger | Login to the program by the seller’s account and you can use this feature. |
| Preconditions | Must have at least one mobile phone previously created. |
| Postconditions | Success: Show list of valid mobile phones.  Fail: No valid mobile phones |
| Main Course | 1. Input mobilePhone\_id you want to find 2. Show list of mobile phone found |
| Alternate Courses | AC2.1: Data doesn't exist in the database.  Display “Mobile phone not found!” |
| Exceptions | EX1: Disconnect to database   1. Request to recheck internet connection. 2. Request program restart. |

* 1. View mobile phone details description

|  |  |
| --- | --- |
| Name | View mobile phone details |
| ID | 4 |
| Description | Used to view mobile phone details |
| Actor | Manager and Seller |
| Organizational  Benefits | You can view all mobile phone information |
| Trigger | Login to the program by the seller's account or manager’s account and you can use this feature. |
| Preconditions | Must have at least one mobile phone previously created. |
| Postconditions | Success: Show all mobile phone information.  Fail: No mobile phone found. |
| Main Course | 1. Input mobile phone id you want to find 2. If mobile phone id valid, show all mobile phone information |
| Alternate Courses | AC2.1: No data exists in the database.   1. Display msg: “Mobile phone not found!” 2. Requires pressing any key to go back |
| Exceptions | EX1: Disconnect to database   1. Request to recheck internet connection. 2. Request program restart. |

* 1. Create order description

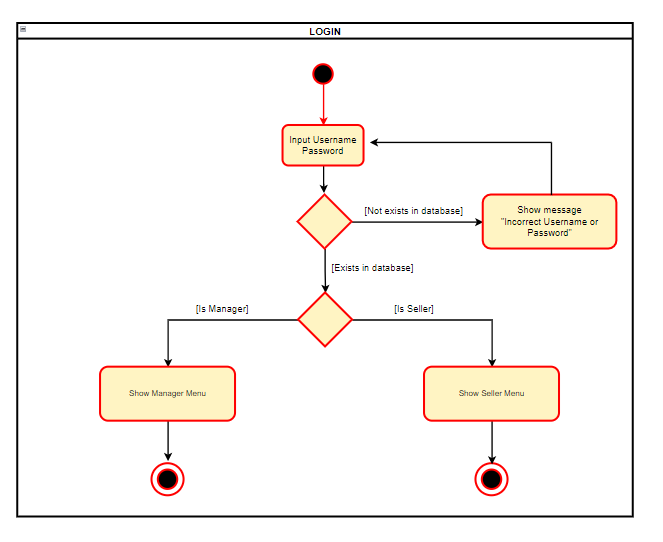
|  |  |
| --- | --- |
| Name | Create order |
| ID | 5 |
| Description | Used to create order |
| Actor | Seller |
| Organizational  Benefits | You can create one or more orders according to customer's request |
| Trigger | Login to the program by the seller's account and you can use this feature. |
| Preconditions | Must have at least one laptop previously created. |
| Postconditions | Success: Create order complete.  Fail: Create order not complete. |
| Main Course | 1. Input mobile phone id and quantity you want to add to order 2. Valid data 3. Input customer information 4. Create order complete. |
| Alternate Courses | AC 2.1: Customer information already exists.   1. You just need to enter the customer's phone number 2. The rest of the information will be entered automatically   AC 2.2: Invalid customer’s phone number   1. Display msg error 2. Request re-entry   AC 3.1: Invalid mobile phone id or quantity.   1. Display msg error 2. Request re-entry |
| Exceptions | EX1: Disconnect to database   1. Request to recheck internet connection. 2. Request program restart. |

* 1. Payment Description

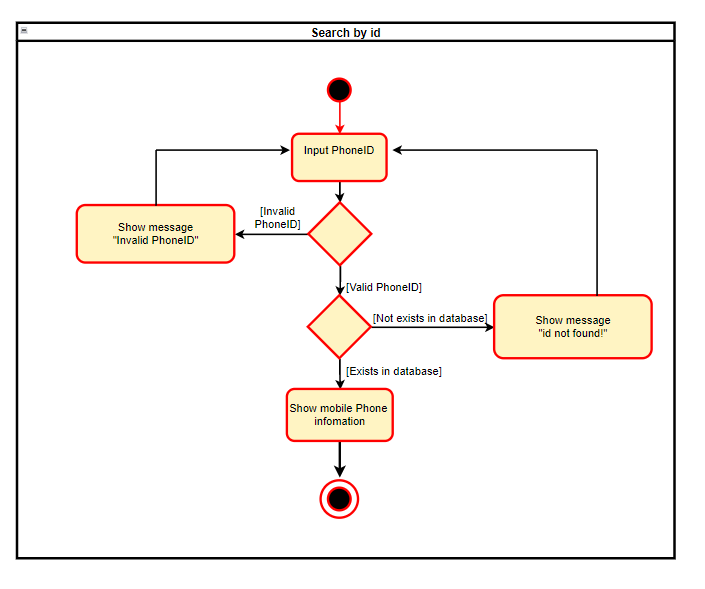
|  |  |
| --- | --- |
| Name | Payment |
| ID | 6 |
| Description | Used to confirm payment or cancel payment |
| Actor | Seller |
| Organizational  Benefits | Allow seller to confirm paid orders or cancel orders |
| Trigger | Login to the program by the accountant's seller and you can use this feature. |
| Preconditions | Must have at least one order in unpaid status. |
| Postconditions | Success: Confirm payment/Cancel order complete  Fail: Confirm payment or cancel order not complete. |
| Main Course | 1. Input order id you want 2. Show order information 3. Choose to payment, cancel order or exit 4. Request complete |
| Alternate Courses | AC 3.1: Confirm payment   1. Enter the amount given by the customer 2. Export invoice   AC 3.2: Cancel order   1. Press ‘X’ to cancel order 2. Display msg “Cancel order complete!’ 3. Requires pressing any movie to go back to list order   AC 3.3: Exit   1. Press ESCAPE to exit 2. Requires pressing any movie to go back to list order   AC 4.1: If payment or cancel order not complete   1. Display msg error 2. Requires pressing any key to go back to list order |
| Exceptions | EX1: Disconnect to database   1. Request to recheck internet connection. 2. Request program restart. |

2. Activity diagram

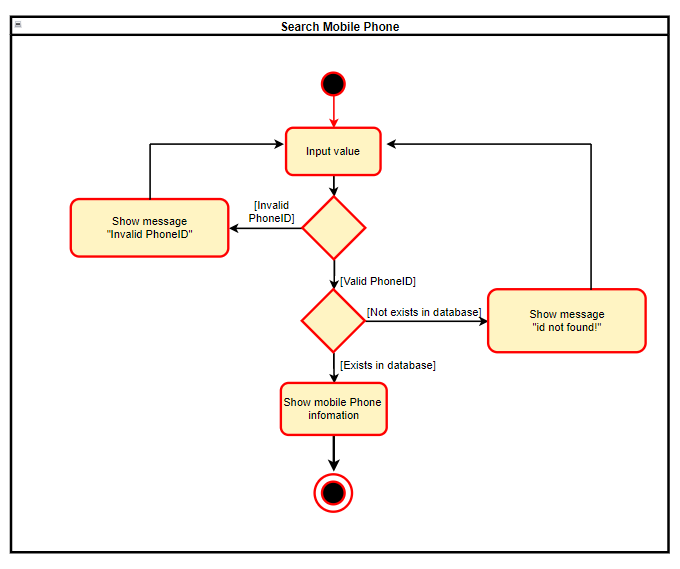
2.1 Login



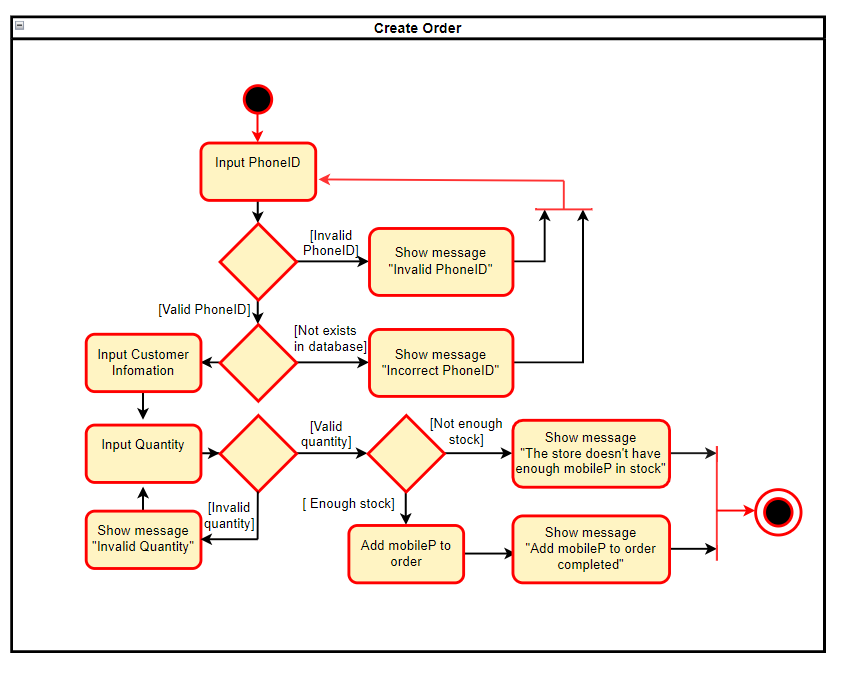
2.4 Search mobile phone by id



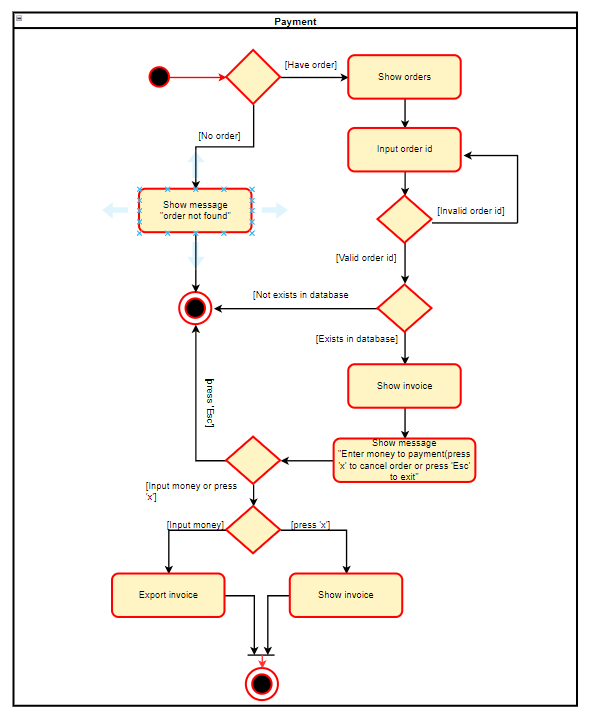
2.5 Search mobile phone by operating system and price



2.6 Create order



2.7 Payment



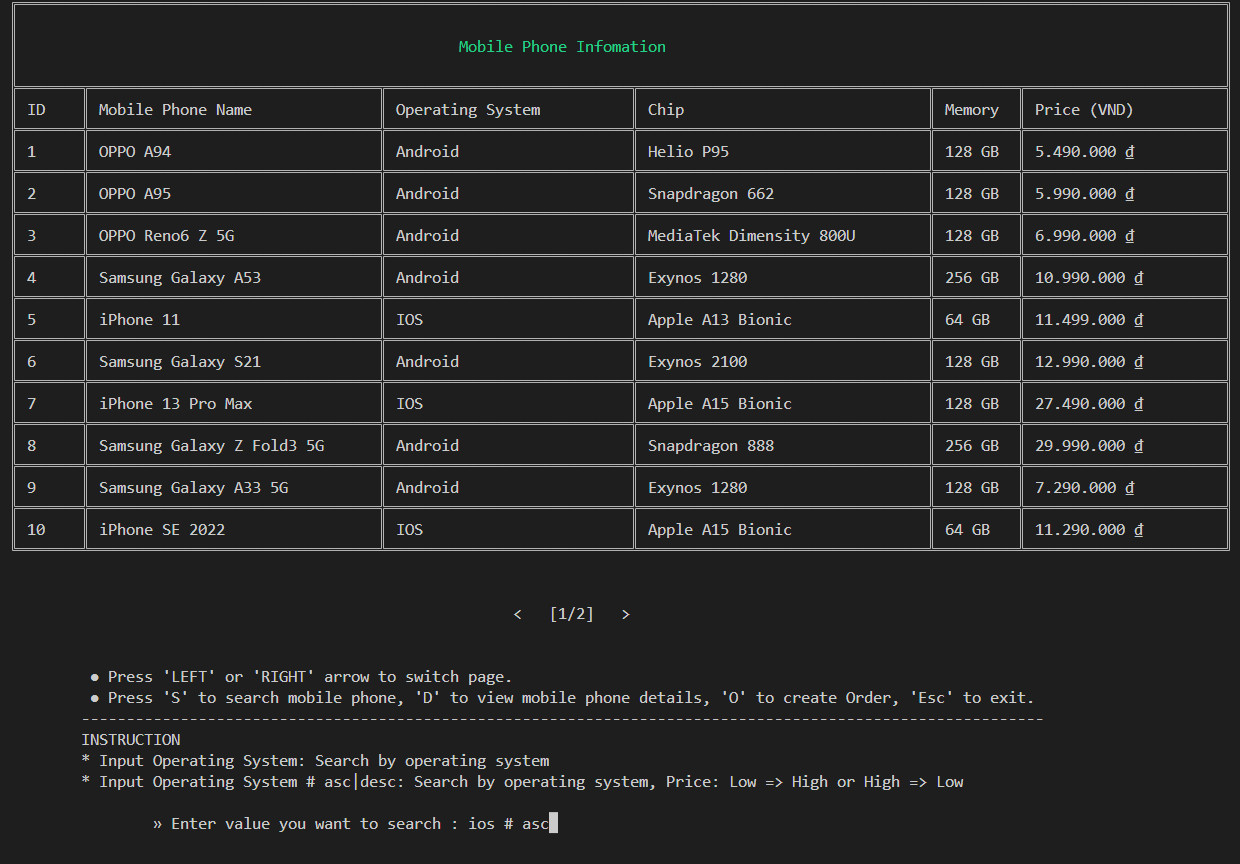
III. Design Details

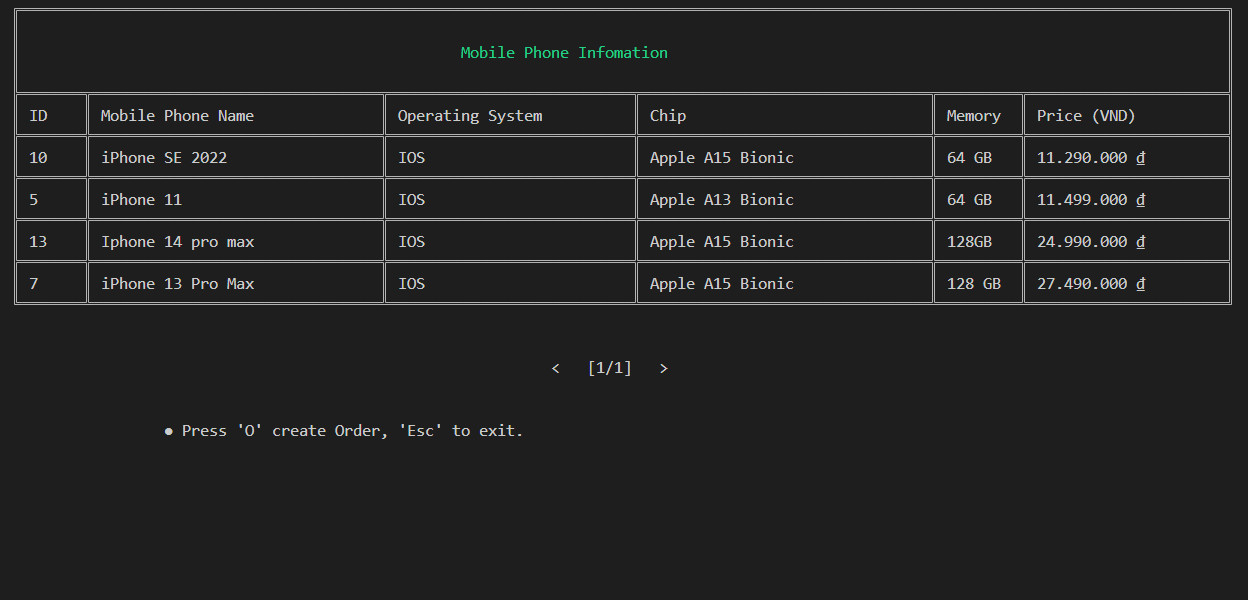
1. UI design

1.1 login

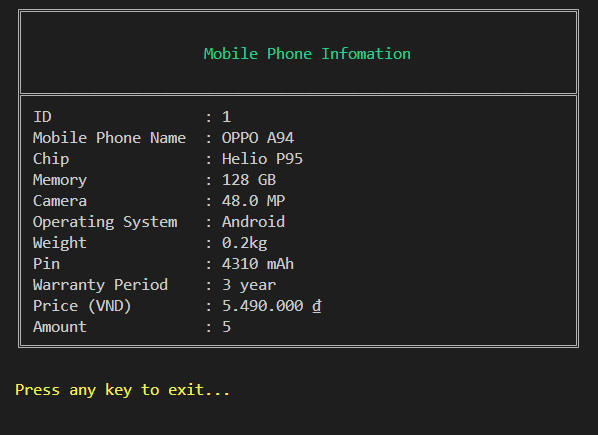


1.2 Search mobile phone by operating System and price

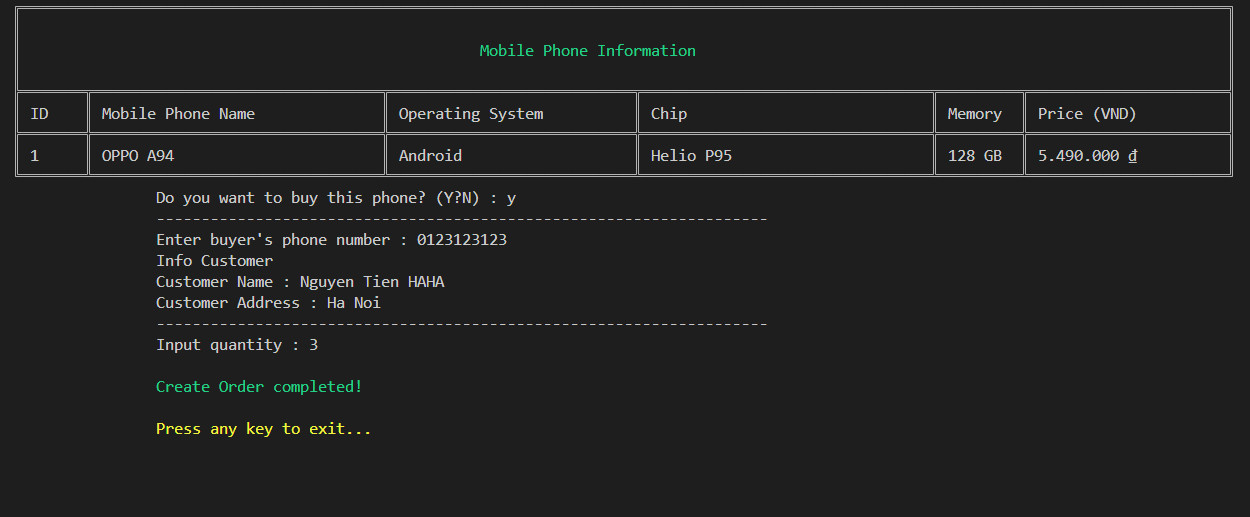




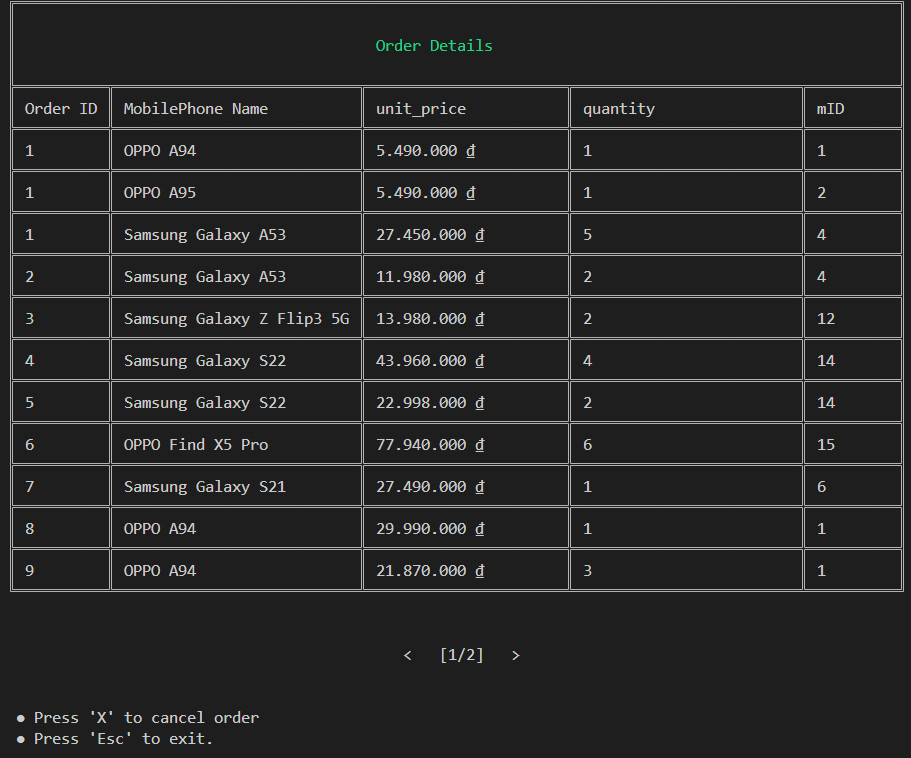
1.3 View mobile phone details



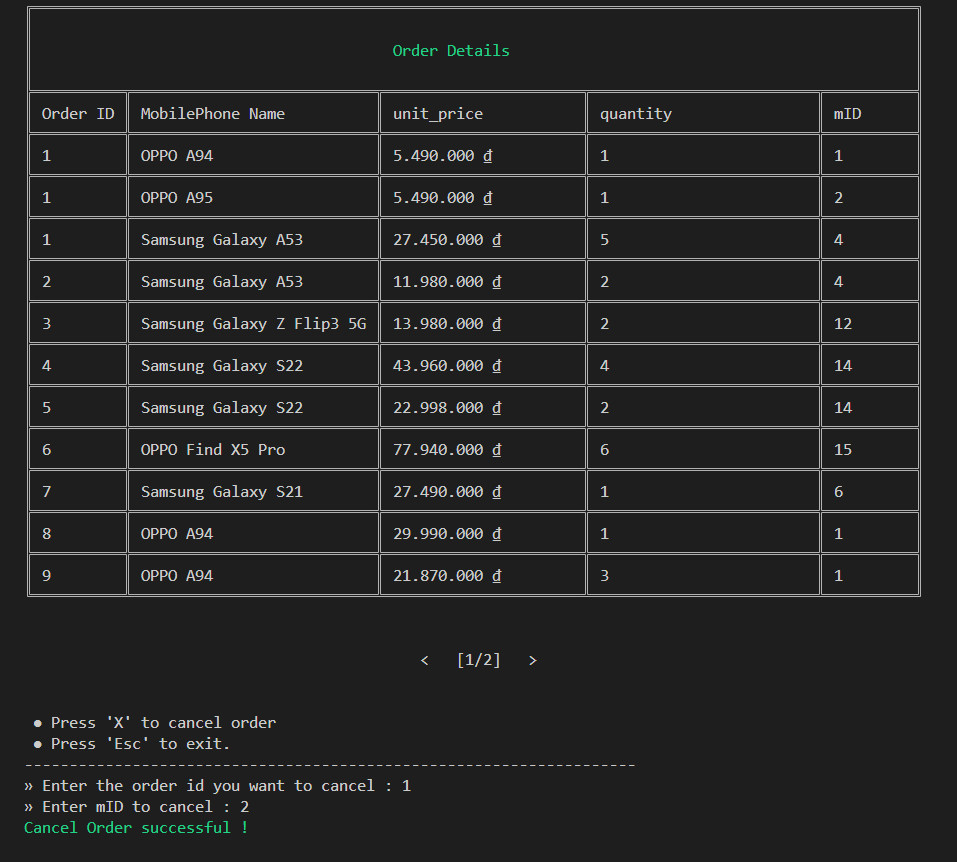
1.4 Create order



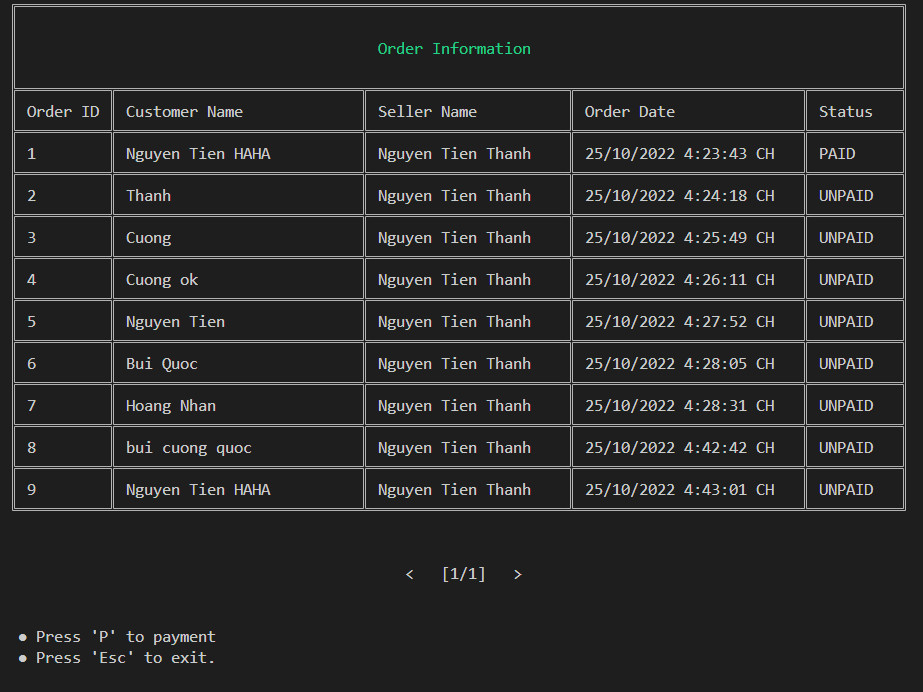
1.5 Show list order details

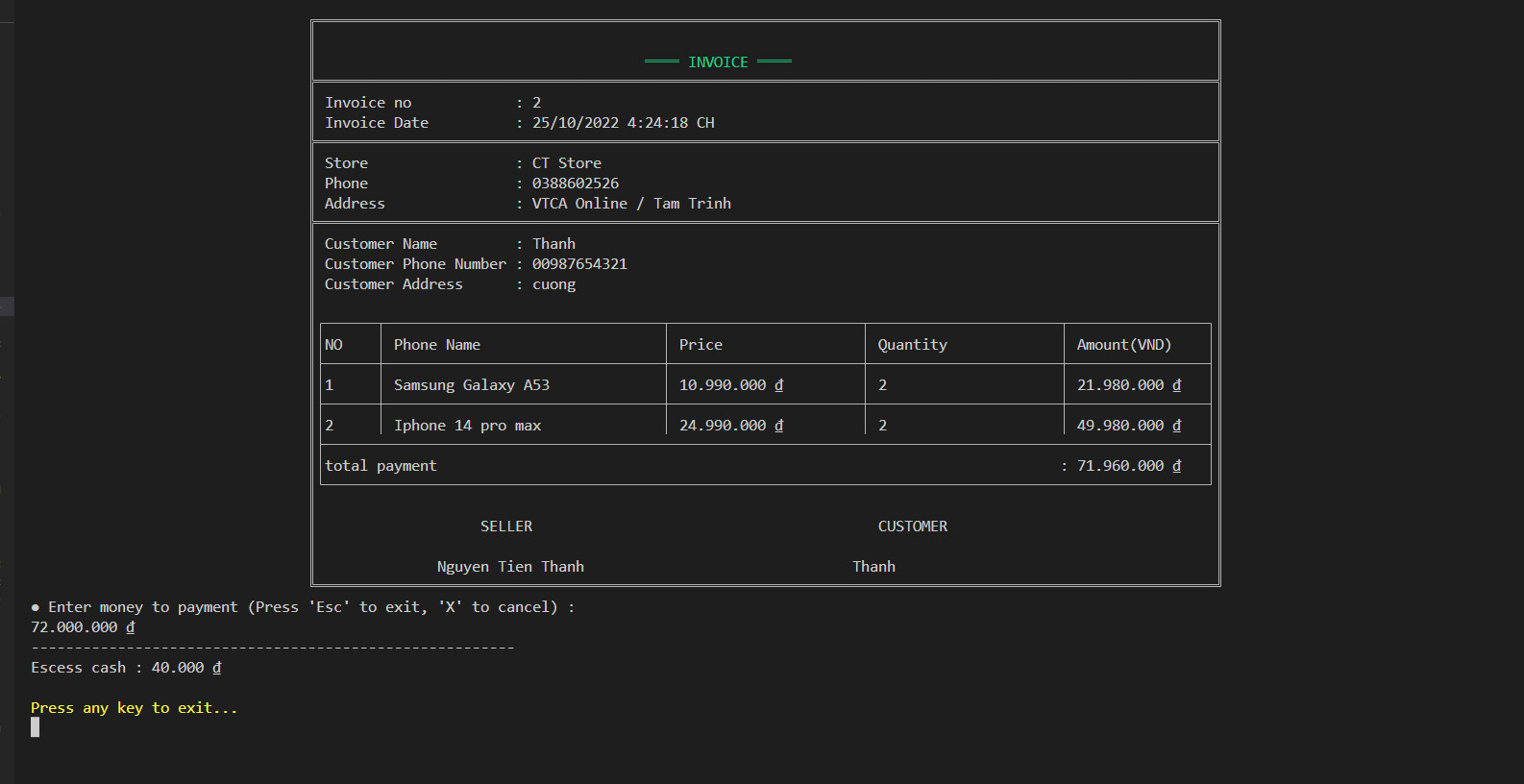


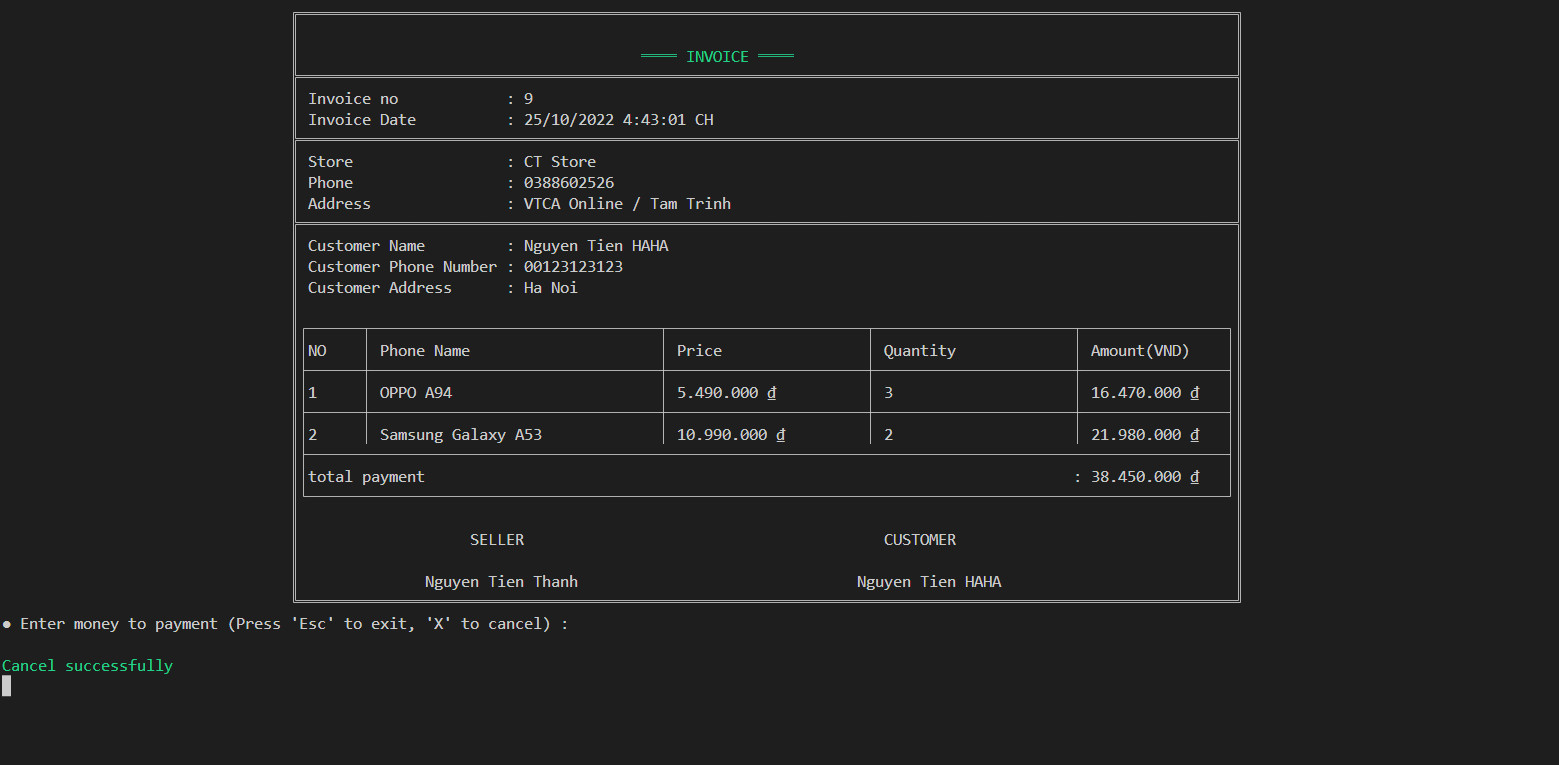
!.6 Cancel order



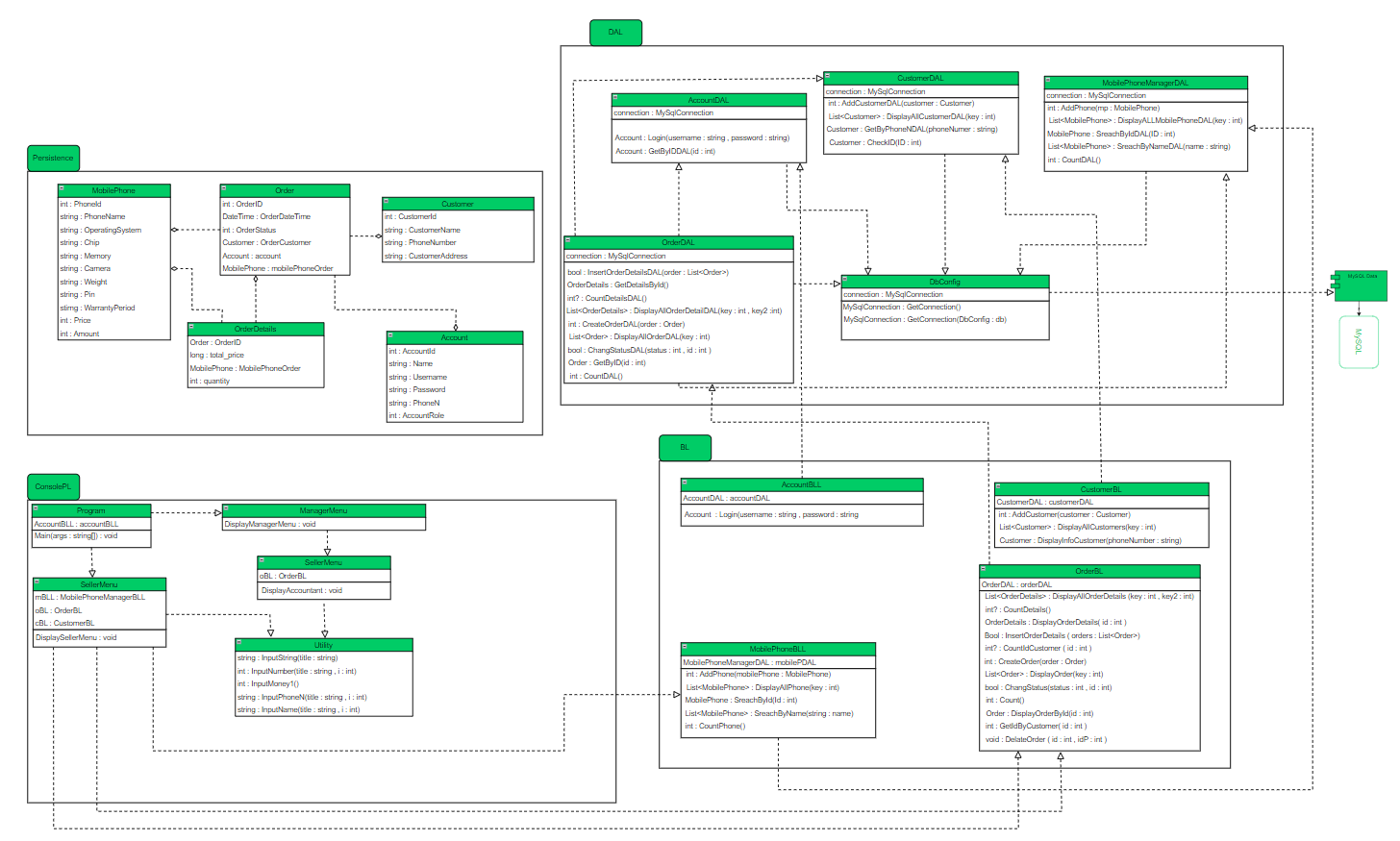
1.7 Payment





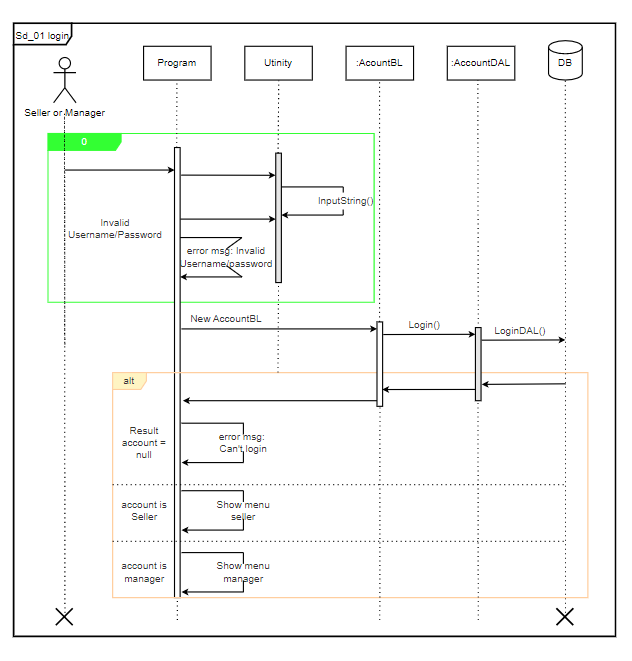


2. Code design ( class diagram)

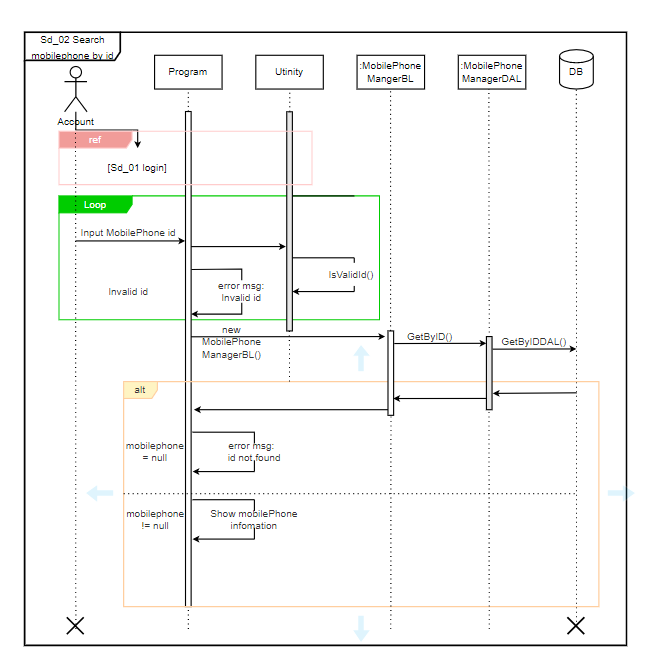


3. Sequence diagram

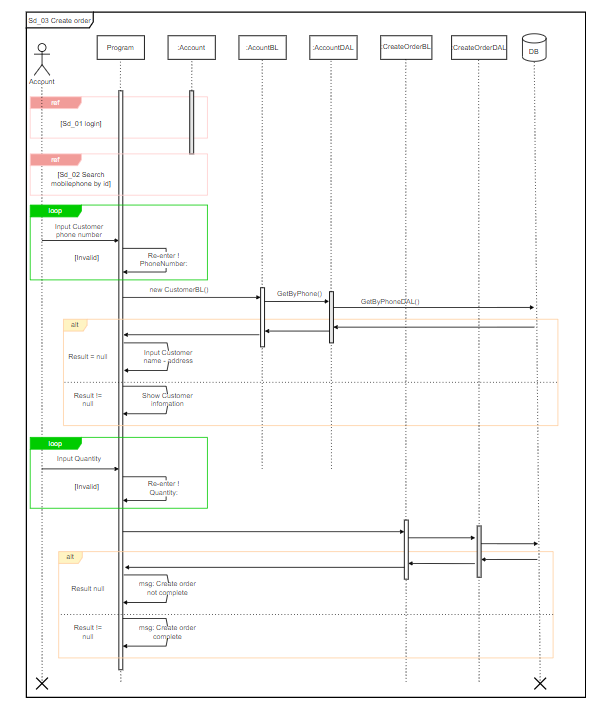
3.1 Login



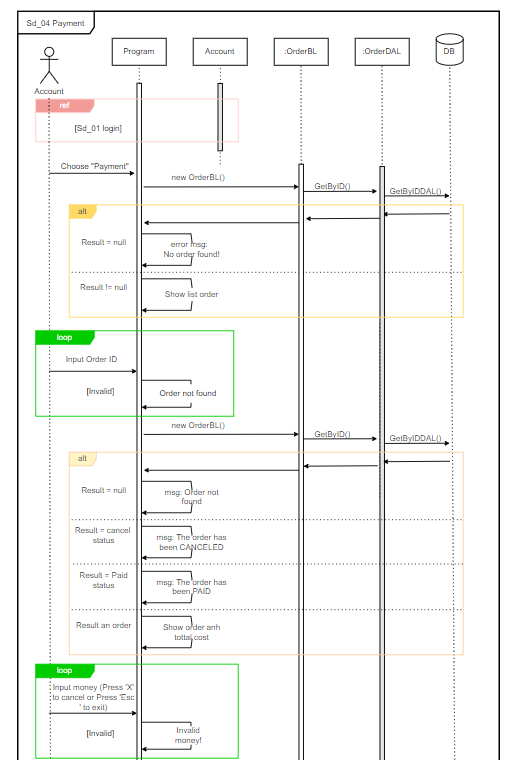
3.2 Search mobile phone by id

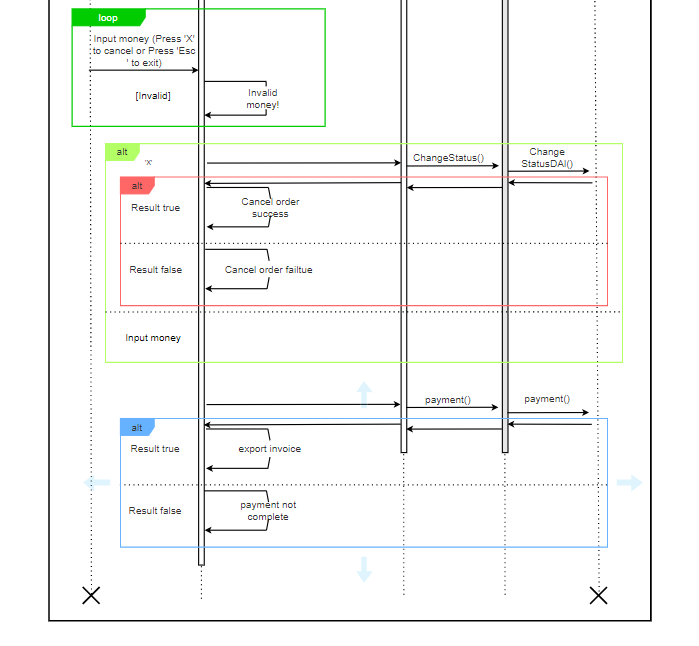


3.3 Create Order



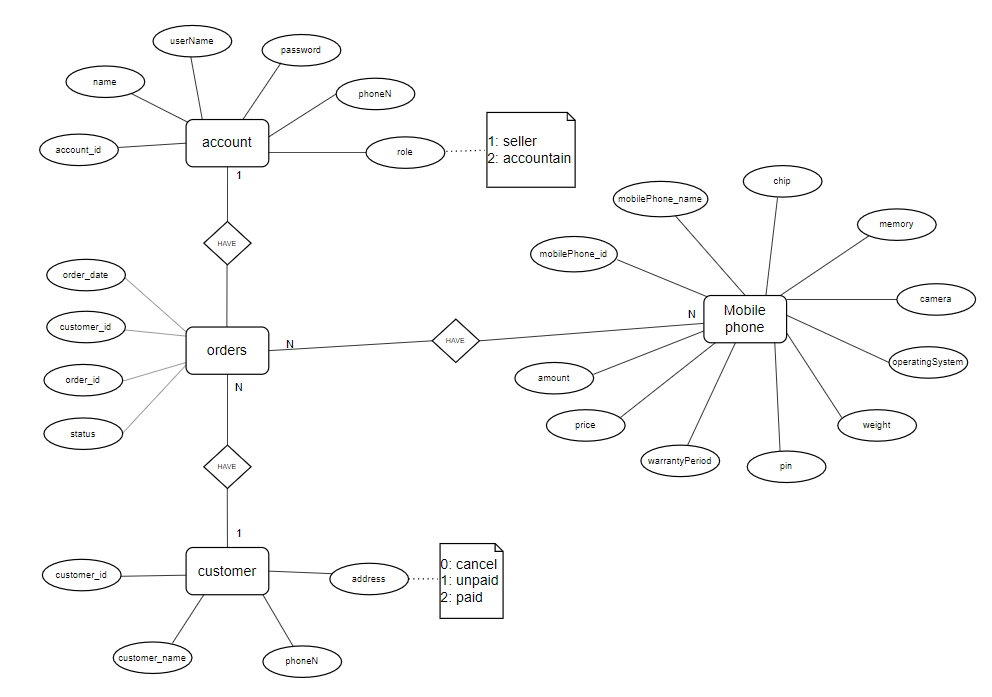
3.4 Payment



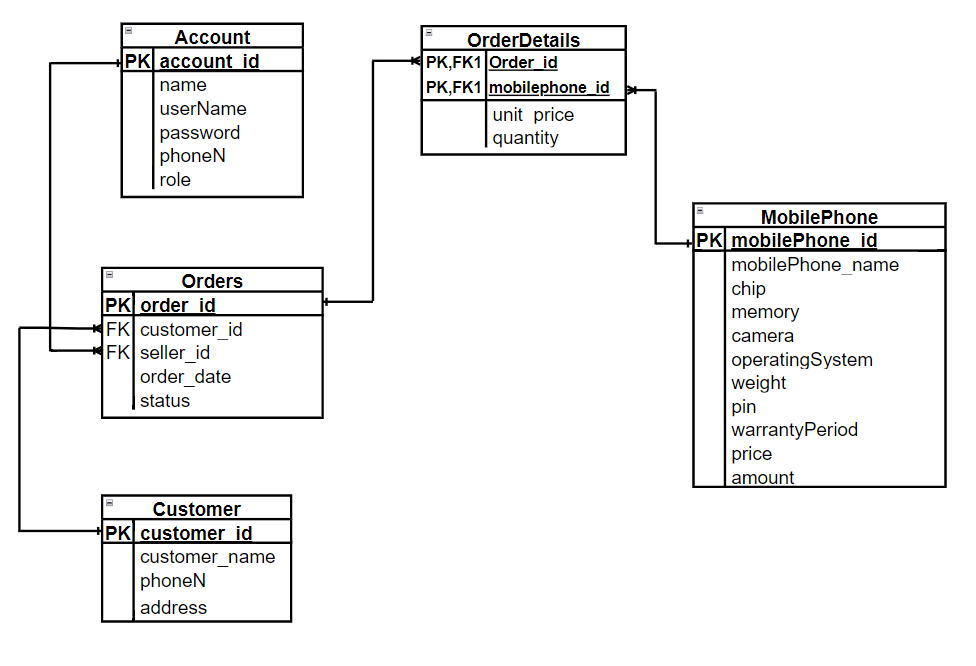


4. Database design

4.1 entity relationship diagram



4.2 database design datails



|  |  |  |  |
| --- | --- | --- | --- |
| Account | | | |
| Column Name | Data Type | Constraints | Description |
| account\_id | int | PRIMARY KEY, AUTO INCREMENT |  |
| name | varchar(100) | NOT NULL | Seller’s name |
| userName | varchar(100) | NOT NULL |  |
| password | varchar(100) | NOT NULL |  |
| phoneN | varchar(10) | NOT NULL | Seller’s phone number |
| role | int(10) | NOT NULL | 1: Manager  2: Seller |

|  |  |  |  |
| --- | --- | --- | --- |
| Customer | | | |
| Column Name | Data Type | Constraints | Description |
| customer\_id | int | PRIMARY KEY, AUTO INCREMENT |  |
| customer\_name | varchar(50) | NOT NULL | Cutomer’s name |
| customer\_phoneN | varchar(12) | NOT NULL, UNIQUE | Cutomer’s phone number |
| address | varchar(50) | NOT NULL | Cutomer’s address |

|  |  |  |  |
| --- | --- | --- | --- |
| MobilePhone | | | |
| Column Name | Data Type | Constraints | Description |
| mobilePhone\_id | int | PRIMARY KEY, AUTO INCREMENT |  |
| mobilePhone\_name | varchar(50) | NOT NULL | Mobile phone’s name |
| chip | varchar(50) | NOT NULL | Mobile phone’s chip |
| memory | varchar(50) | NOT NULL | Mobile phone’s memory |
| camera | varchar(10) | NOT NULL | Mobile phone’s camera |
| operatingSystem | varchar(20) | NOT NULL | Mobile phone’s operating system |
| weight | varchar(20) | NOT NULL | Mobile phone’s weight  Unit: kilogram |
| pin | varchar(10) | NOT NULL | Mobile phone’s pin |
| warrantyPeriod | varchar(10) | NOT NULL | Mobile phone’s warranty period |
| price | decimal(20,2) | NOT NULL | Mobile phone’s price |
| amout | int(10) | NOT NULL | Mobile phone’s amount |

|  |  |  |  |
| --- | --- | --- | --- |
| Orders | | | |
| Column Name | Data Type | Constraints | Description |
| order\_id | int | PRIMARY KEY, AUTO INCREMENT |  |
| mobilePhone\_id | int | NOT NULL |  |
| customer\_id | int | NOT NULL |  |
| account\_id | int | NOT NULL |  |
| price | decimal(20,2) | NOT NULL | Mobile phone’s price |
| quantity | int | NOT NULL | Mobile phone’s quantity |
| order\_date | datatime | NOT NULL, DEFAULT NOW | Order creation date |
| status | int | NOT NULL | 0: Cancal  1: Unpaid  2: Paid |

|  |  |  |  |
| --- | --- | --- | --- |
| order\_details | | | |
| Column Name | Data Type | Constraints | Description |
| order\_id | int | PRIMARY KEY, FOREIGN KEY |  |
| mobilephone\_id | int | PRIMARY KEY, FOREIGN KEY |  |
| unit\_price | Decimal | NOT NULL | Mobilephone’s unit price |
| quantity | int | NOT NULL | Mobilephone’s Quantity |

1. Test code

1. login test

### Login test 1

|  |  |
| --- | --- |
| Test Case Number | 001 |
| Test Case Name | LoginTest1() |
| Test Case Description | This test case checks username and password correct or not |
| Preconditions | Username and password must exist in the database |
| Test Case Input | “cuong”, “thanh”  “thanh”, “cuong” |
| Test Case Expected Output | A Account |
| Test Case Steps | New Account  New AccountDAL  Call Login method  Compare output with expected output |
| Default Value Preverving |  |

### Login test 2

|  |  |
| --- | --- |
| Test Case Number | 002 |
| Test Case Name | LoginTest2() |
| Test Case Description | This test case checks username and password correct or not |
| Preconditions |  |
| Test Case Input | "cuongd.asd", "12345678"  "cuong", "ddf12345678" |
| Test Case Expected Output | Null |
| Test Case Steps | New Account  New AccountDAL  Call Login method  Compare output with expected output |
| Default Value Preverving |  |

## 2. Search Mobile Phone test

### 2.1 Search by id test 1

|  |  |
| --- | --- |
| Test Case Number | 001 |
| Test Case Name | SearchByIDTest1() |
| Test Case Description | Search by product id in the system |
| Preconditions | Logged in |
| Test Case Input | 1, 5, 10  2, 3, 8 |
| Test Case Expected Output | List mobile phone |
| Test Case Steps | New GetByIDDAL  Call Search method  Compare output with expected output |
| Default Value Preverving |  |

### 

### 

### 2.2 Search by id test 2

|  |  |
| --- | --- |
| Test Case Number | 002 |
| Test Case Name | SearchByIDTest2() |
| Test Case Description | Search by product id in the system |
| Preconditions | Logged in |
| Test Case Input | “a123”, “a.123”  “12./asd” |
| Test Case Expected Output | null |
| Test Case Steps | New GetByIDDAL  Call Search method  Compare output with expected output |
| Default Value Preverving |  |

### 2.3 Search by operating system test 3

|  |  |
| --- | --- |
| Test Case Number | 003 |
| Test Case Name | SearchByOperatingSTest3() |
| Test Case Description | Search by product operating system in the system |
| Preconditions | Logged in |
| Test Case Input | “ios”, “android” |
| Test Case Expected Output | List mobile phone |
| Test Case Steps | New SearchByOperatingSDAL  Call Search method  Compare output with expected output |
| Default Value Preverving |  |

### 

### 2.4 Search by operating system test 4

|  |  |
| --- | --- |
| Test Case Number | 004 |
| Test Case Name | SearchByOperatingSTest4() |
| Test Case Description | Search by product operating system in the system |
| Preconditions | Logged in |
| Test Case Input | “ios123..”, “android//123a” |
| Test Case Expected Output | null |
| Test Case Steps | New SearchByOperatingSDAL  Call Search method  Compare output with expected output |
| Default Value Preverving |  |

3. Change status account

### 3.1 Change status account test 1

|  |  |
| --- | --- |
| Test Case Number | 001 |
| Test Case Name | ChangeStatusAccountTest1() |
| Test Case Description | Change status account |
| Preconditions | Logged in |
| Test Case Input | 2, 3, 4 |
| Test Case Expected Output | A ID |
| Test Case Steps | New ChangeStatusAccountDAL  Call ChangeStatusAccount method  Compare output with expected output |
| Default Value Preverving |  |

### 3.2 Change status account test 2

|  |  |
| --- | --- |
| Test Case Number | 002 |
| Test Case Name | ChangeStatusAccountTest2() |
| Test Case Description | Change status account |
| Preconditions | Logged in |
| Test Case Input | 17, 28, 98 |
| Test Case Expected Output | null |
| Test Case Steps | New ChangeStatusAccountDAL  Call ChangeStatusAccount method  Compare output with expected output |
| Default Value Preverving |  |

4. Create order test

4.1 Create order test 1

|  |  |
| --- | --- |
| Test Case Number | 001 |
| Test Case Name | CreateOrderTest1() |
| Test Case Description | Take order by insert order id |
| Preconditions | Logged in |
| Test Case Input | 1  2  3 |
| Test Case Expected Output | A ID |
| Test Case Steps | New CreateOrderDAL  Call CreateOrder method  Compare output with expected output |
| Default Value Preverving |  |

### 4.2 Create order test 2

|  |  |
| --- | --- |
| Test Case Number | 002 |
| Test Case Name | CreateOrderTest2() |
| Test Case Description | Take order by insert order id |
| Preconditions | Logged in |
| Test Case Input | 0, 999, 1010  100213, 200003 |
| Test Case Expected Output | null |
| Test Case Steps | New CreateOrderDAL  Call CreateOrder method  Compare output with expected output |
| Default Value Preverving |  |

4. Search mobile phone

4.1 Search mobile phone test 1

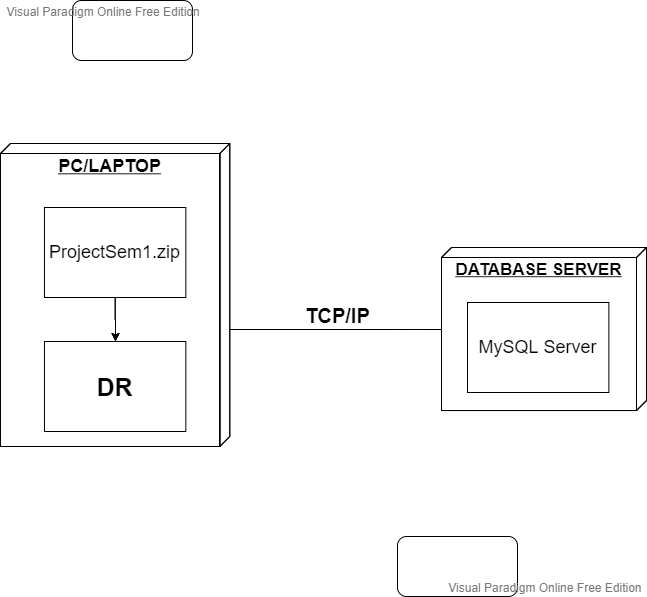
|  |  |
| --- | --- |
| Test Case Number | 001 |
| Test Case Name | SearchByOperatingSTest4() |
| Test Case Description | Search by product operating system and price in the system |
| Preconditions | Logged in |
| Test Case Input | Ios # asc  android # asc |
| Test Case Expected Output | List mobile phone |
| Test Case Steps | New SearchByOperatingSDAL  Call Search method  Compare output with expected output |
| Default Value Preverving |  |

### 4.2 Search mobile phone test2

|  |  |
| --- | --- |
| Test Case Number | 002 |
| Test Case Name | SearchByOperatingSTest4() |
| Test Case Description | Search by product operating system and price in the system |
| Preconditions | Logged in |
| Test Case Input | Asd//# asc  Android ### desc |
| Test Case Expected Output | null |
| Test Case Steps | New SearchByOperatingSDAL  Call Search method  Compare output with expected output |
| Default Value Preverving |  |

# VI. Installation Instructions

1. Deployment Diagram



Installation steps

Database Install

Server install

Application install