**DANANG SOFTWARE CORPORATION**



**SOFTECH-APTECH**

****

***Theme:***

**THE COFFEE**

**Professor : Mr. NGUYEN DUC THAO**

**Members : LE THI NHAT TAN**

**TRAN PHUC TRIEU**

**NGUYEN PHI LIP**

**Group : 3**

**Batch : 151-152**

Danang, 11th Dec 2019



This is to certify that

**Mr./Ms.**

has successfully designed and

developed

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Submitted by:

Date of Issue: \_\_\_\_\_\_\_\_

Authorized Signature: \_\_\_\_\_\_\_\_\_\_\_





**Acknowledgements**

We would like to thank our professor Mr.Nguyen Duc Thao who persuaded and continuously guided us during the whole course of our project, for his assistance and insightful comments, and who willingly shared his expertise with us.

**Abstract**

In the current development trend in the world of science and technology which have changed drastically. Part of which is the application of Information Technology in the daily life of people. We human beings are aiming at establishing a smart planet. Today, with the strong development of IT combined with the development of the Internet has connected the world together into a unified whole. It has become an effective tool for many sectors: transportation, military, medicine ... and especially in the management in general and in particular the cafe manager.

Previously when the computer has not been widely used in the management of the work is done by hand so it takes time, human resources and finance. Today, with the strong development of information technology has helped the management is done more easily, reducing the cost, time ...

Through the process of surveying a few cafes, they have built up management topics cafe with an expectation that the management is done more easily, conveniently and reduce the mistakes.

Thanks to the attention and guidance of Mr.Nguyen Duc Thao we have gradually research and apply the knowledge they have learned to understand, analyze and build management program relative response some ra.Tuy requirements, due to limited knowledge of the program is still not free from shortcomings. So we look forward to receiving the comments of all the teachers and friends to be able to gradually build the program more complete and more efficient.

I sincerely thank Mr. Nguyen Duc Thao helped us.

**Table of content**

Section 1. Vision Document

1. Introduction

1.1 Purpose and Motivation

2. Project Overview

2.1 Background

3. Requirement Specifications

3.1 Main Requirements

3.2 Critical Use Case Requirements

3.3 Environment

Section 2. Project Plan

## 1. Project organization

1.1. Software Process Model

1.2. Roles and responsibilities

1.3. Tools and Techniques

## 2. Project Management Plan

2.1. Task

2.2. Task Sheet: Assignments and Timetable

3. Coding convention

Section 3. Architecture Design

1. Introduction

2. Architecture

3. Presentation Tier

## 4. Entity Relationship Diagram

5. Table

Section 4. Test Plan

1. Introduction

2. Features to be tested

3. Test cases

4. Approach

4.1 Unit testing

4.2 Load testing

4.3 System Testing

4.4 Performance Testing

4.5 Manual Testing

5. Pass/fail criteria

Section 5. Assessment Evaluation

1. Introduction

2. Manual Testing

3. Test Cases

3.1 Employee

3.2 Admin

Section 6. User Manual

**Section.1 VISION DOCUMENT**

**1. Introduction**

***1.1 Purpose and Motivation***

The objective of this project is to create a coffee shop management software. It allows users to perform tasks such as product purchase, human resources management, cash flow management … The competitive advantages of this software are that functions are performed at a high level of speediness, accuracy and most importantly it helps boost sales of various products.

Administrators will be offered supplementary functions different from ones used by other users.

*The origin of all endeavors to create this project*:

* We apply the technology of Java, SQL Server, and etc.
* We base on the demands of people from different existing coffee shops.

**2. Project Overview**

***2.1 Background***

*This project has the following functionalities:*

1. **A homepage offering service functions:**

This is the page when users can choose functions such as additional services ordering, extra tables, table choosing, bills settling, etc.

**2) Human Resources Management**

This is the page where functions are performed such as checking details of bills, counting total revenues per purchase day or the sum of one bill. This function is only accessible to the Admin.

**3) Inventory Management**

This function is only accessible to the Admin. This function allows the Admin to add extra services to the inventory as well as modify some information concerning prices or service classification.

**5) Administration**

Administrators will be provided with special functions as follows:

* Inventory Management
* Cash-flow Management

**3. Requirement Specifications**

***3.1 Main Requirements***

The Main Requirements include Java to develop the application, SQL Server 2008(Enterprise Edition) to design the database and Netbean is the main tool to help develop the software

***3.2 Critical Use Case Requirements***

Unregistered users, registered users and administrators are the main sources to make use of the software.

******

***Use Case Diagram***

**Use Cases details :**

**Administrator Use Cases**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE SPECIFICATION** | | | |
| **Use-case No.** | UC01 | **Use-case Version** | 1.0 |
| **Use-case Name** | Login | | |
| **Actors: Admin**  **Purpose:**  Admin login by use admin Id and password. Then a select query is used to check data from the database and display the notification: Valid or invalid account.Admin can use the function if account is valid.  **Input:**  Admin log in by use admin Id and password  **Output:**  The system will display the notification: Valid or invalid account. | | | |

**Administrator Use Cases**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE SPECIFICATION** | | | |
| **Use-case No.** | UC02 | **Use-case Version** | 1.0 |
| **Use-case Name** | Update and Fix Employee | | |
| **Actors: Admin**  **Purpose:**  After login successful, admin can update and fix employee which matches their subject. Then a select query is used to update and fix data from the database and display from the entered information.  **Input:**  Admin update and fix employee  **Output:**  The system will update information for database. | | | |

***3.3 Environment***

- Application will be developed in Netbean

- SQL Server 2012 (enterprise version) will be used as database server.

- Java will be used as the programming language.

**Section.2 Project Plan**

## Project organization

### 1.1 Software Process Model

With the schedule of weekly reports for every stage, the software will be developed by using a waterfall model which is very simple and require minimal resource for implementation. With a waterfall model, developing process will include six main phases:

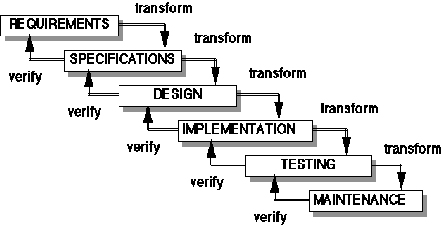


Figure 1 - Waterfall model

* **Requirement analysis and software definition**: The first phase is also the most important phase of the process which may cause great effect in other phases. This phase includes gathering as much as possible requirements from customer, or other information source like the World Wide Web, and producing a more detail and accurate software definition.
* **System design**: This phase is fundamental for the implementation phase. Based on the customer’s requirements to create logical modules, and definite their interrelations. Using algorithms and diagrams to describe an implementation of those modules.
* **Implementation**: This phase consists of actually constructing the product as per the design specification(s) developed in the previous step. Typically, this step is performed by a development team consisting of programmers, interface designers and other specialists, using tools such as compilers, debuggers, interpreters and media editors.
* **Testing:**
  + **System implementation and Unit testing**: Developing software modules follow detail designs, and doing unit testing of each module.
  + **Integration and System testing**: Testing output, performance in modules integrating process, and retests all functions of the whole system.
  + **User acceptance testing**: A proper execution of all the preceding stages ensures an application as per the provided requirements and most importantly, it ensures a satisfied client.
* **System deployment and maintenance**: After testing completed, the software was handled over a client, developing team will respond for maintenance to the system.

### 1.2 Roles and responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Lê Thị Nhật Tân | Team Leader, DEV, Tester | * Specify user requirement. * Control the development process * Managing process * Assign task for the team member * Designing database * Coding * Testing |
| **2** | Trần Phúc Triều | Team Member, DEV, Tester, Design | * Specify user requirement. * Control the development process * Testing * Coding |
| **3** | Nguyễn Phi Lip | Team Member, DEV, Tester, Design | * Specify user requirement. * Control the development process * Testing * Coding |

Table 3: Roles and Responsibility Details

### 1.3 Tools and Techniques

- Front-end technologies:. Java Swing

- Back-end: Java

- Database Management System: SQL Server 2012

## Project Management Plan

### Task

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ***No*** | ***Task*** | ***Description*** | ***Output*** | ***Deliverables*** | ***Resource Needed*** | ***Dependencies***  ***And constraints*** | ***Risk*** |
| 1 | Create Software Requirements Specification | Create software requirements specification (SRS) | SRS document | SRS document before  23/10/2019 | 3 people |  | Lack of experience when getting requirement |
| 2 | Design Database | Create conceptual, logical and physical database design | Database Design and Database script | Design Database before 27/10/2019 | 1 people | Depends on the completion of SRS | SRS may not be detailed enough to cover the business rules, causing the database design to be inappropriate or to be changed based on customers’ requirements |
| 3 | Create Software Design Description (SDD) | Design the system in an OOP manner | Architecture design, detailed design, diagrams and design specification | SDD 0 before 03/11/2019 | 3 people | Depends on the completion of SRS | Risks may include choosing an inappropriate architecture and design patterns, causing the system hard to maintain, or wasting coding efforts |
| 4 | Layout Design | Create the main GUI for layout, and create global styles that will be applied to the GUI | Swing components | Complete before 10/11/2019 | 2 people |  | Lack of professional designers in team. |
| 5 | Create Coding | Map the architecture design into source code, create the project solution files and common classes, and implement common functions | Java and solution file contain coding | Complete before 14/11/2019 | 1 people | Dependent on Architecture design and SRS |  |
| 6 | Coding | Implement the system based on the requirements | Source code of the project, unit test reports | Executable programs and source code before 22/11/2019 | 3 people | Depends on the completion of SRS, SDD, Database design, Coding framework | Unit test may not be performed thoroughly, causing spending many efforts in system test phase |
| 7 | Test Plan | Make a plan for testing | Test plan document | Test plan before 24/11/2019 | 2 people |  |  |
| 8 | System Test | Perform system test for the system | System test report | Software Test Documentation before 27/11/2019 | 2 people | Coding is finished | Lack of professional testers in team. Developers are also responsible for system testing, this may lead to compromise |
| 9 | User manual | Create User manual document | Report | Report before 01/12/2019 | 2 people |  |  |
| 10 | Input Initial Data | Collect and input initial data for the system before release | Initial data | Complete before 05/12/2019 | 2 people | Coding and system test is finished |  |
| 11 | Deployment | Deploy the system | Running application | Full Project | 3 people | Coding and system test is finished, initial data are input | The owner host lacks of support in some components such as database engine. |

Table 4: Task

### Task Sheet: Assignments and Timetable

Refer to Tasksheet.doc document.

## Coding Convention

-Java: Using to develop application.

Summary:

* Naming Convention.
* Indentation.
* Declaration.
* Code Examples

Follow “Code Conventions for the Java Programming Language, by Oracle

<http://www.oracle.com/technetwork/java/codeconvtoc-136057.html>

**Section.3 Architecture Design**

**1. Introduction**

The purpose of this document is to provide an architectural design for the Online Book Store. The design will show the presentation tier, the middle tier consisting of classes, sequence diagrams, and the data tier consisting of the database design diagram.

**2. Architecture**

Three-tier (layer) is a client-server architecture in which the user interface, business process (business rules) and data storage and data access are developed and maintained as independent modules or most often on separate platforms.

The Architecture of Online Book Store is based on three-tier architecture. The three logical tiers are

 Presentation tier -Java, Images.

 Middle tier – Java classes.

 Data tier- Database

Fig.4 below shows the model of 3-tier architecture.

The main reason for considering three-tier architecture for the Online Book store is as follows:

***Flexibility:***

 Management of data is independent from the physical storage support,

 Maintenance of the business logic is easier,

 Migration to new graphical environments is faster.

 If there is a minor change in the business logic, we don’t have to install the entire

system in individual user’s PCs.

**Reusability:**

 Reusability of business logic is greater for the presentation layer. As this component is developed and tested, we can use it in any other project and would be helpful for future use.

**Security:**

 More secured architecture since the client cannot access the database directly.

****

*Fig.4 3 tier Architecture*

**3. Presentation Tier**

The presentation tier for the Online Book Store is ASP.NET Web Forms with Master Pages. The presentation tier takes care of the user interaction and the visual part of the website. The various Web forms used by clients are listed below.

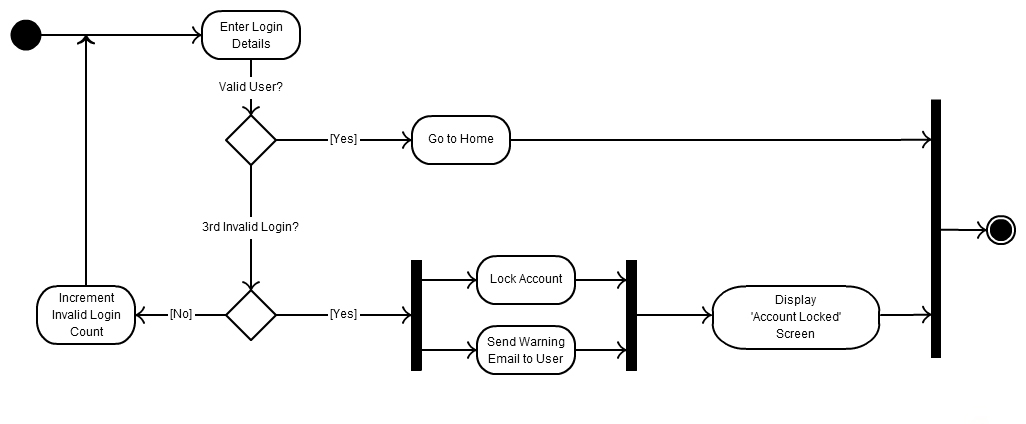
• Emloyee

|  |  |
| --- | --- |
| **JAVA Application Forms** | **Definition** |
| Default | This form is used to display the form menu |
| ViewListEmployee | This form is used to display: List Employee |
| ViewSubject | This form is used to display: subject |

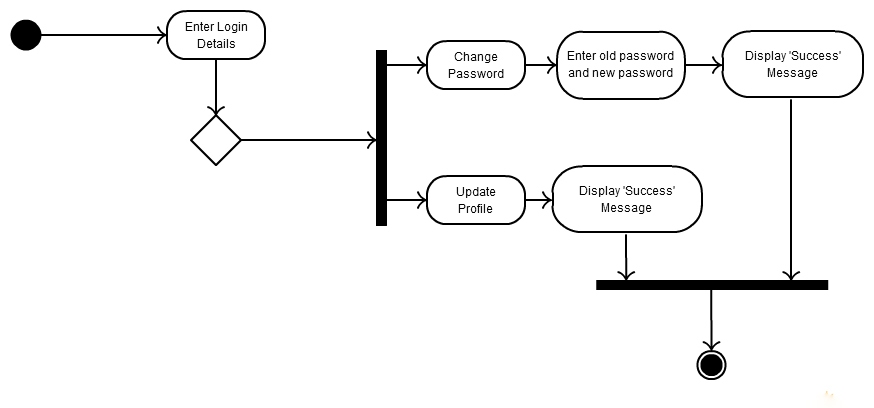
• Admin

|  |  |
| --- | --- |
| **JAVA Application Forms** | **Definition** |
| Default | This form is used to display the form menu |
| Login | This form is used for a admin to login |
| AdminManager | This form is used to manage sell |
| AdminEmployee | This form is used to manage employee |
| AdminManagerStore | This form is used to manage store |
| AdminManagerSell | This form is used to manage sell |

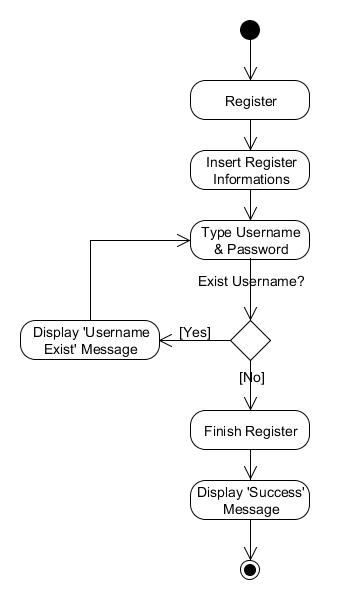
* *The diagram below captures the page flow for user in the Manager Coffe.*



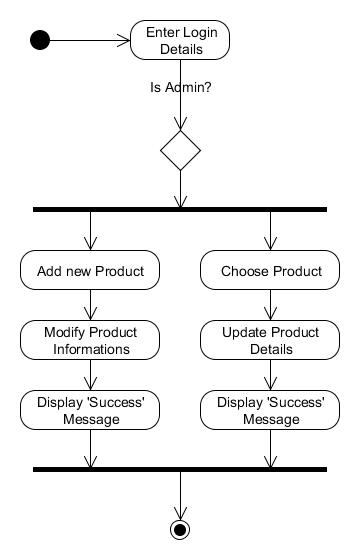
*Activity Diagram : Login*



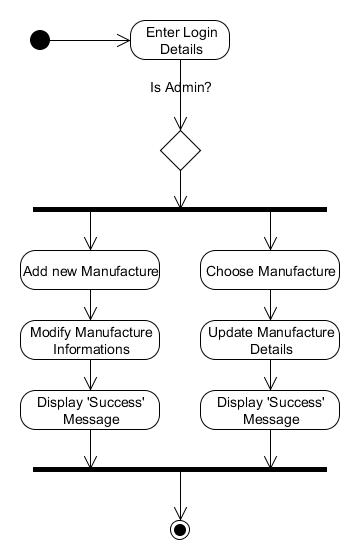
*Activity Diagram : Update Profile*



*Activity Diagram : User Register*



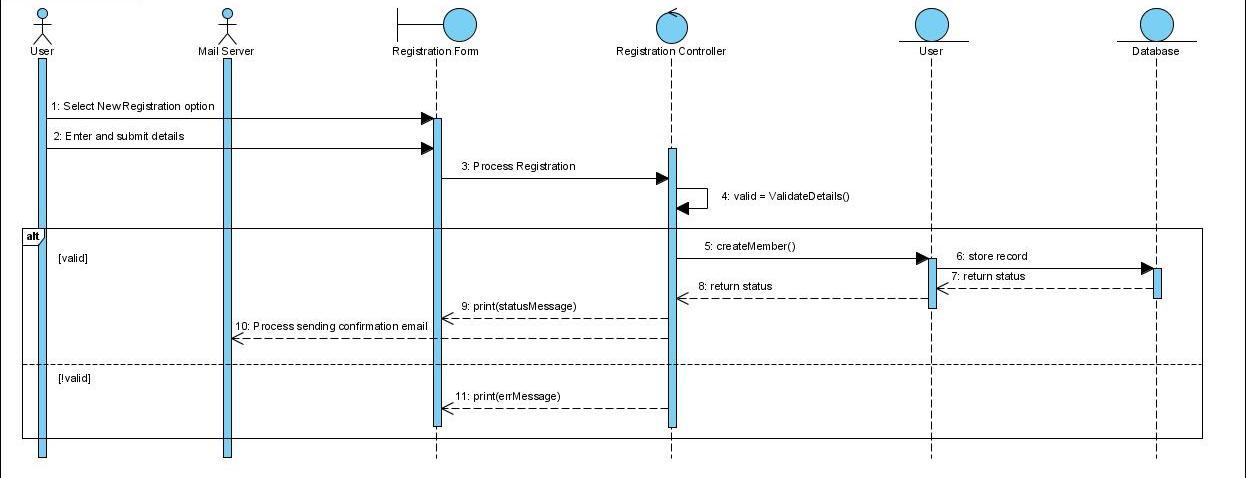
*Activity Diagram : Product Management*



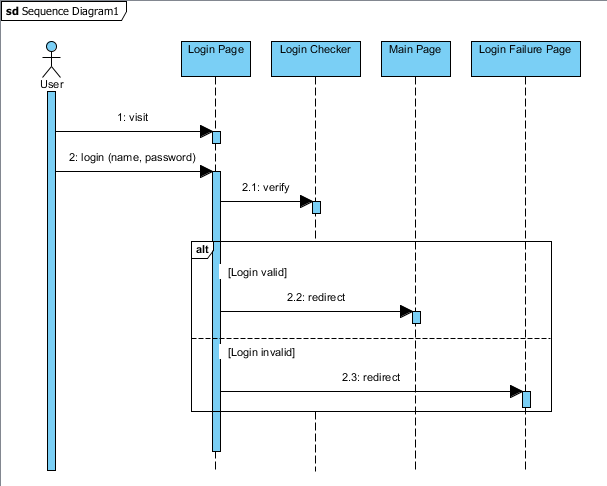
*Activity Diagram : Supplier Management*

* *The diagram below captures the Activity Diagram for user in the Online Book Store System.*

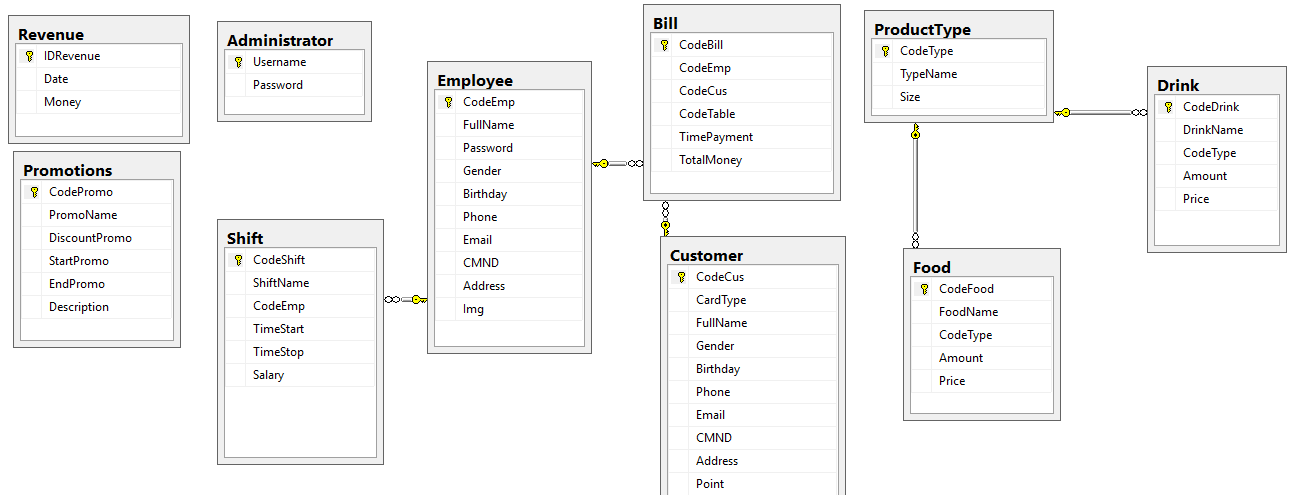
*a. Account Registration diagram*

****

*b. User login diagram*

****

## Entity Relationship Diagram

****

**5. Tables**

*The system database table design is shown below.*

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu** | **Mô tả** |
| **db\_Administrator** | | |
| Username | varchar(50) | Tên đăng nhập |
| Password | varchar(20) | Mật khẩu |
| **db\_** **Employee** | | |
| CodeEmp | varchar(10) | Mã nhân viên |
| FullName | nvarchar(50) | Họ tên nhân viên |
| Password | varchar(20) | Mật khẩu |
| Gender | nvarchar(10) | Giới tính |
| Birthday | varchar(20) | Ngày sinh |
| Phone | varchar(20) | Số điện thoại |
| Email | varchar(50) | Địa chỉ email |
| CMND | nchar(9) | Chứng minh nhân dân |
| Address | nvarchar(Max) | Địa chỉ nhà |
| Img | varchar(50) | Ảnh chân dung |
| **db\_** **Customer** | | |
| CodeCus | varchar(10) | Mã khách hàng |
| CardType | varchar(10) | Loại thẻ (Vip/Member) |
| FullName | nvarchar(50) | Họ tên khách hàng |
| Gender | nvarchar(10) | Giới tính |
| Birthday | varchar(20) | Ngày sinh |
| Phone | varchar(20) | Số điện thoại |
| Email | varchar(50) | Địa chỉ email |
| CMND | nchar(9) | Chúng minh nhân dân |
| Address | nvarchar(Max) | Địa chỉ nhà |
| Point | int | Điểm tích lũy |
| **db\_Shifts** | | |
| CodeShift | varchar(10) | Mã Ca làm việc |
| ShiftName | nvarchar(30) | Tên ca |
| CodeEmp | varchar(10) | Mã nhân viên |
| Timestart | datetime | Thời gian bắt đầu |
| Timestop | datetime | Thời gian kết thúc |
| Salary | float | Lương |
| **db\_ProductType** | | |
| CodeType | varchar(10) | Mã loại sản phẩm |
| TypeName | nvarchar(30) | Tên loại sản phẩm |
| Size | nvarchar(10) | Kích cỡ |
| **db\_Drink** | | |
| CodeDrink | varchar(20) | Mã đồ uống |
| DrinkName | nvarchar(100) | Tên đồ uống |
| CodeType | varchar(10) | Mã loại sản phẩm |
| Amount | int | Số lượng |
| Price | float | Giá tiền |
| **db\_Food** | | |
| CodeFood | varchar(20) | Mã thức ăn |
| FoodName | nvarchar(100) | Tên thức ăn |
| CodeType | varchar(10) | Mã loại sản phẩm |
| Amount | int | Số lượng |
| Price | float | Giá tiền |
| **db\_Bill** | | |
| CodeBill | int | Mã hóa đơn |
| CodeEmp | varchar (10) | Mã nhân viên |
| CodeCus | varchar (10) | Mã khách hàng |
| CodeTable | varchar (10) | Mã bàn |
| TimePayment | date | Tổng tiền |
| **db\_Promotions** | | |
| CodePromo | int | Mã khuyến mãi |
| PromoName | varchar(20) | Tên khuyến mãi |
| DiscountPromo | int | Giảm giá |
| StartPromo | varchar(20) | Thời gian bắt đầu |
| EndPromo | varchar(20) | Thời gian kết thúc |
| Description | nvarchar(max) | Mô tả |
| **db\_** **BillDetail** | | |
| CodeBill | int | Mã hóa đơn |
| CodeDrink | varchar(20) | Mã đồ uống |
| CodeFood | varchar(20) | Mã đồ ăn |
| Amount | int | Số lượng |
| Price | float | Giá tiền |
| PromoName | varchar(20) | Tên khuyến mãi |
| DiscountPromo | int | Giảm giá |
| **db\_** **Revenue** | | |
| IDRevenue | int | Mã thống kê |
| Date | varchar(20) | Ngày thống kê |
| Money | float | Số tiền |

**Section.4 Test Plan**

**1. Introduction**

The goal of this document is to develop a test plan for the Management Coffee design system. This document defines all the procedures and activities required to prepare for testing of the functionalities of the system which are specified in Vision document. The objectives of the test plan are to define the activities to perform testing, define the test deliverables documents and to identify the various risks and contingencies involved in testing.

**2. Features to be tested**

*The following list describes the features to be tested:*

**Admin:**

+ Login

+ Manage Service

+ Manage Sell

+ Manage Store

+ Manage Shift

**Employee:**

+ Manage Sell

+ Manage Store

**3. Test cases**

***3.1*** *Test case for Employee:*

|  |  |  |  |
| --- | --- | --- | --- |
| ***ID*** | ***Test case*** | ***User Input*** | ***Pass Criteria*** |
| S\_VLC\_1 | Employee view | Enter invalid id or Employee name | A message displays:  “No Employee matches with the entered id or Employee name” |
| S\_VLC\_2 | Employee view | Enter valid code or Employee name | Display respective study result of the entered Employee |

|  |  |  |  |
| --- | --- | --- | --- |
| ***ID*** | ***Test case*** | ***User Input*** | ***Pass Criteria*** |
| S\_VS\_1 | Employee view subject information | Enter invalid code or subject name | A message displays “ No subject matches with the entered id or subject name” |
| S\_VS2 | Employee view subject information | Enter valid code or subject name | Display respective information of the entered subject |

*3.2 Test case for Admin:*

|  |  |  |  |
| --- | --- | --- | --- |
| ***ID*** | ***Test case*** | ***User Input*** | ***Pass Criteria*** |
| AD\_STD\_1 | Manage Employee | Admin adds new Employee | Employee is added |
| AD\_STD\_2 | Manage Employee | Admin updates a Employee | Employee is updated |

|  |  |  |  |
| --- | --- | --- | --- |
| ***ID*** | ***Test case*** | ***User Input*** | ***Pass Criteria*** |
| AD\_TCH\_1 | Manage Customer | Admin adds new Customer | Customer is added |
| AD\_TCH\_2 | Manage Customer | Admin updates a Customer | Teacher is Customer |

**4. Approach**

*This section describes the overall approach of the testing which ensures that the each feature and the combination of the features are adequately tested. The major tasks that are used are*

***4.1 Unit testing***

Unit testing is a method of testing that verifies the individual units of source code are working properly. The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. The testing tool of NetBean will be used for unit testing.

***4.2 Load testing***

Load testing is the process of creating demand on a system or device and measuring its response. It generally refers to the practice of modeling the expected usage of a software program by simulating multiple users accessing the program concurrently. As such, this testing is most relevant for multi-user systems; often one built using a client/server model, such as application servers.

***4.3 System Testing***

Once the entire system has been built then it has to be tested against the Software Requirement Specification and System Specification to check if it delivers the features required. System testing can involve a number of specialist types of test to see if all the functional and non-functional requirements have been met.

***4.4 Performance Testing***

The system should meet the performance requirements as mentioned in the Vision document. The performance will be evaluated based on the response time of the GUI and the database commands.

***4.5 Manual Testing***

Manual Testing will be done to ensure the correctness of various parts of the code using test cases generated by the tester.

**5. Pass/fail criteria**

The system should satisfy all the functional requirements, in the Vision document. Each feature to be tested will be evaluated against its requirement as stated in the Vision Document. The pass or fail of a test depends on whether the system meets with all the particular post conditions. Test cases executed on the Online Book Store will pass if they meet the specific requirements as mentioned in the Vision Document.

**Section.5 Assessment Evaluation**

**1. Introduction**

The goal of this document is to present all the test results using the test cases defined in the Test Plan Document. I have performed Manual and Performance testing for my manage project.

**2. Manual Testing**

Manual testing is done to test the correctness of all the functionalities by manually entering the data. Test case functionalities for manual testing include:

**3. Test Cases**

***3.1*** *Test case for Employee:*

|  |  |  |  |
| --- | --- | --- | --- |
| ***ID*** | ***Test case*** | ***User Input*** | ***Pass Criteria*** |
| S\_VLC\_1 | Employee view | Enter invalid id or Employee name | A message displays:  “No Employee matches with the entered id or Employee name” |
| S\_VLC\_2 | Employee view | Enter valid code or Employee name | Display respective study result of the entered Employee |

|  |  |  |  |
| --- | --- | --- | --- |
| ***ID*** | ***Test case*** | ***User Input*** | ***Pass Criteria*** |
| S\_VS\_1 | Employee view subject information | Enter invalid code or subject name | A message displays “ No subject matches with the entered id or subject name” |
| S\_VS2 | Employee view subject information | Enter valid code or subject name | Display respective information of the entered subject |

*3.2 Test case for Admin:*

|  |  |  |  |
| --- | --- | --- | --- |
| ***ID*** | ***Test case*** | ***User Input*** | ***Pass Criteria*** |
| AD\_STD\_1 | Manage Employee | Admin adds new Employee | Employee is added |
| AD\_STD\_2 | Manage Employee | Admin updates a Employee | Employee is updated |

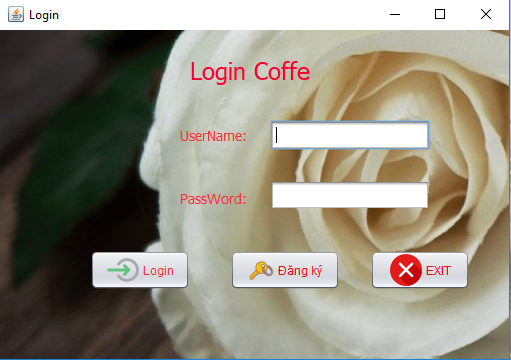
|  |  |  |  |
| --- | --- | --- | --- |
| ***ID*** | ***Test case*** | ***User Input*** | ***Pass Criteria*** |
| AD\_TCH\_1 | Manage Customer | Admin adds new Customer | Customer is added |
| AD\_TCH\_2 | Manage Customer | Admin updates a Customer | Teacher is Customer |

**Section.6 User Manual**

**1. User Manual**

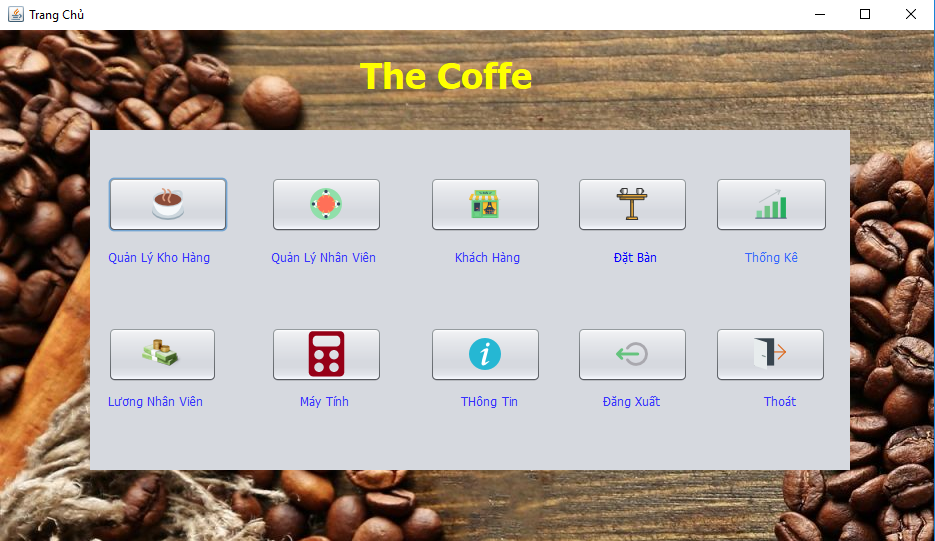
When we run the SMM application, the main form is displayed as below.

Admin, employee will have different authority.

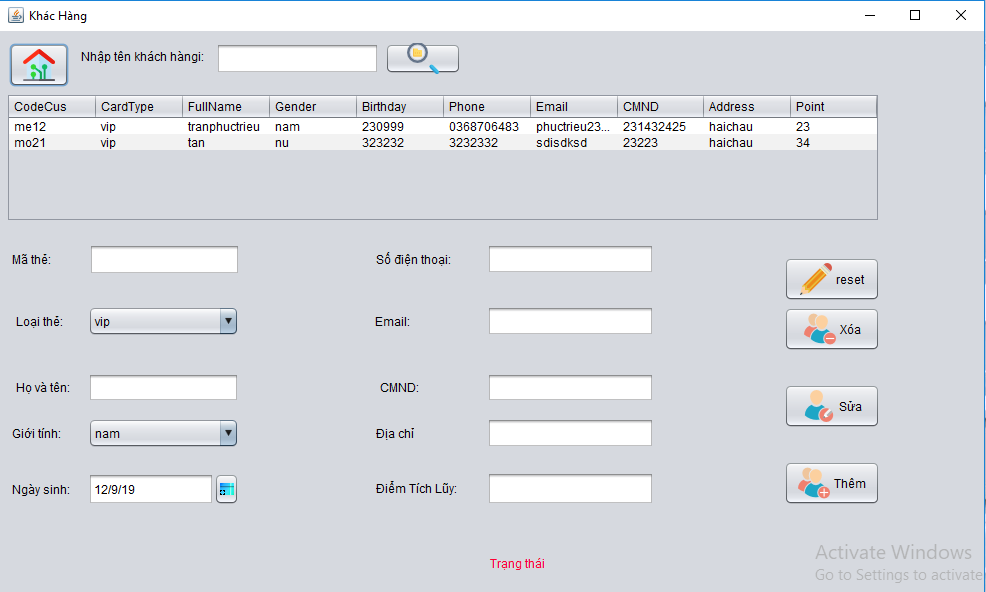


**The main interface** **of the coffee shop management software**. You can click any of these main functions as follows:

* Account log-out
* Password change
* Utilites
* Exit
* Employee manager
* Times management
* Sales manager
* Calculator

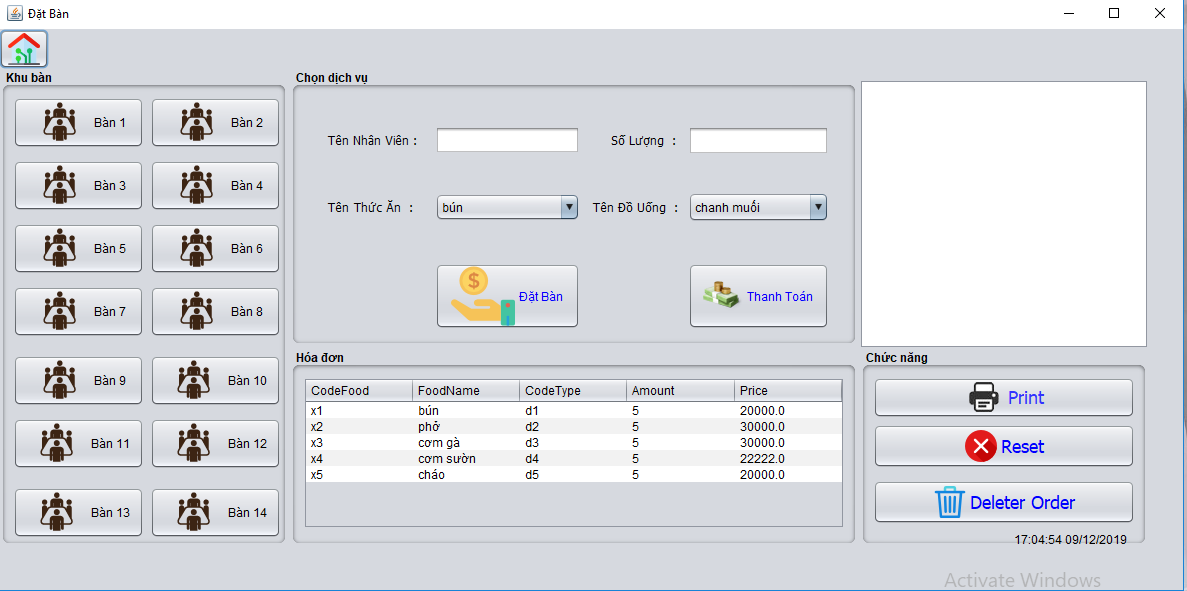


**Registration Form Interface**. Here, you can update all personal information to register for a new account to use our coffee shop management software. All you need to do is to fill in all necessary personal information and click the “Register” button.



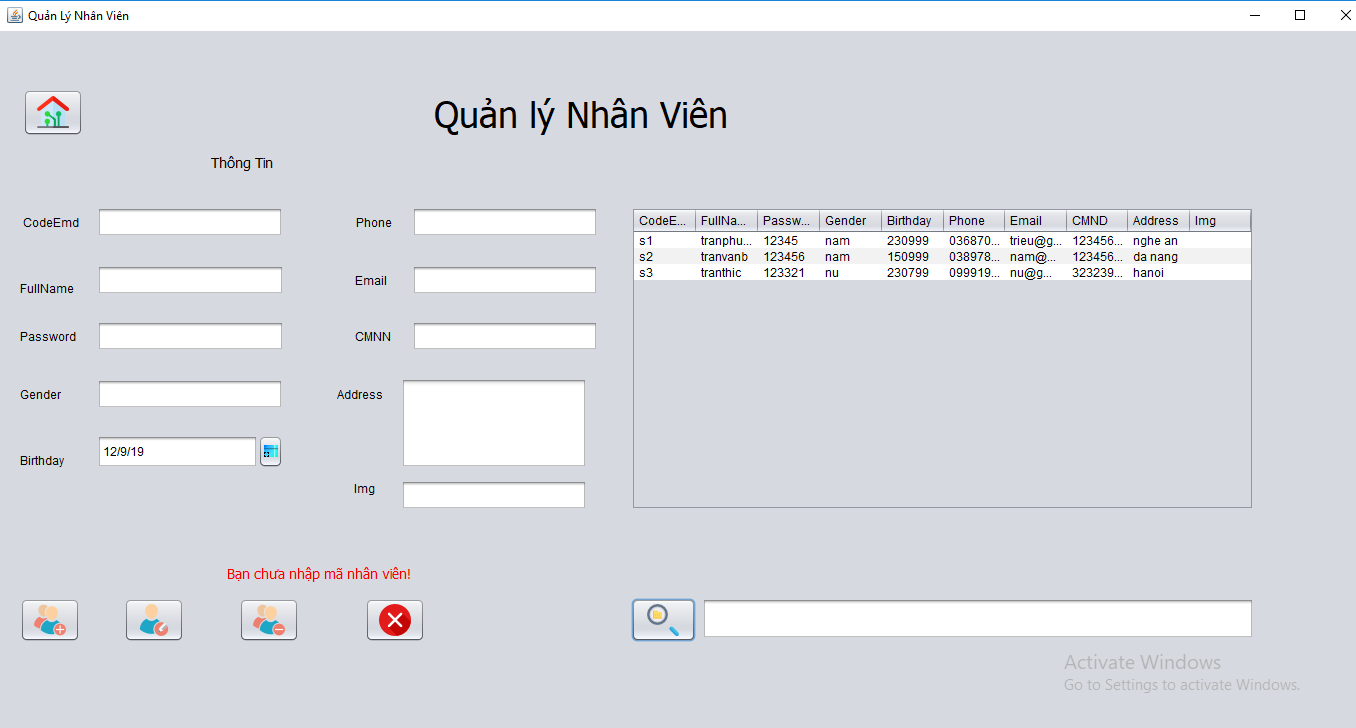
**Sales Order Interface**:

* Staff choose guests’ tables on the software. Click all services guests demand, and then click Nhân viên chọn bàn của khách ngồi trên phần mềm quản lý. Chọn các ‘Add’. All added services will appear on the bill. When guests ask for bill settling, staff will click “Settle” and then the bill will appear on the screen. Click “Save” and the bill will be saved to the database system. Click “Print” for printing the bill.
* Besides, the software also supports staff to add tables if all tables on the software have been booked.



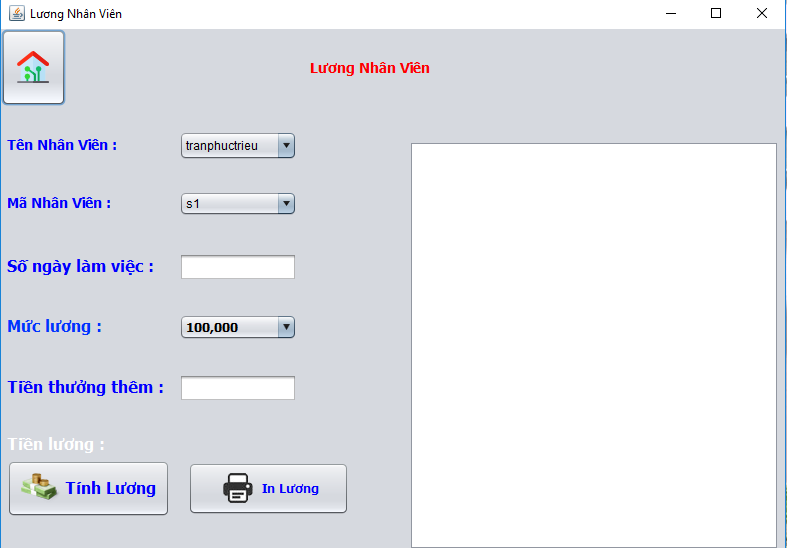
**Human Resources Management Interface:**

* The software allows administrators to add staff, search for staff information.



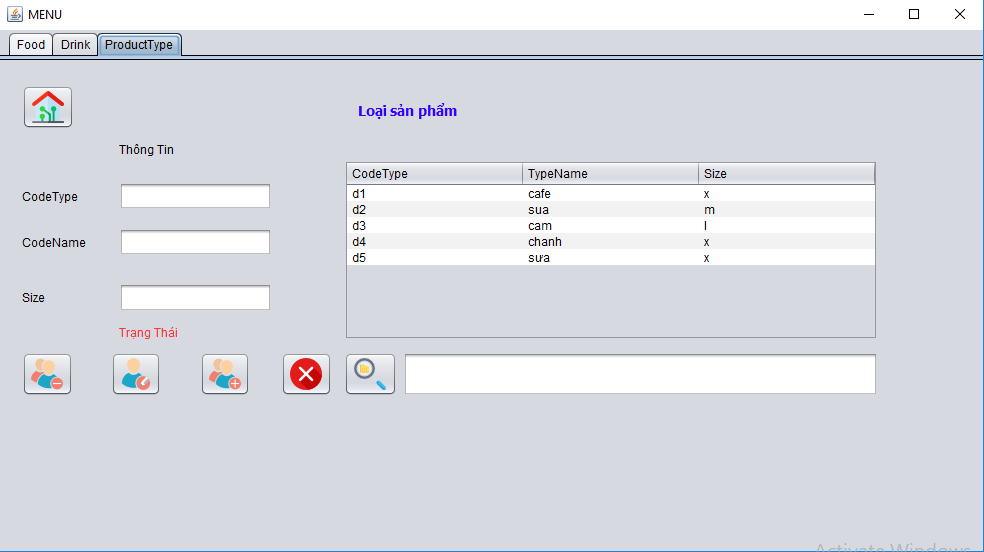
**Staff Roster Management Interface:**

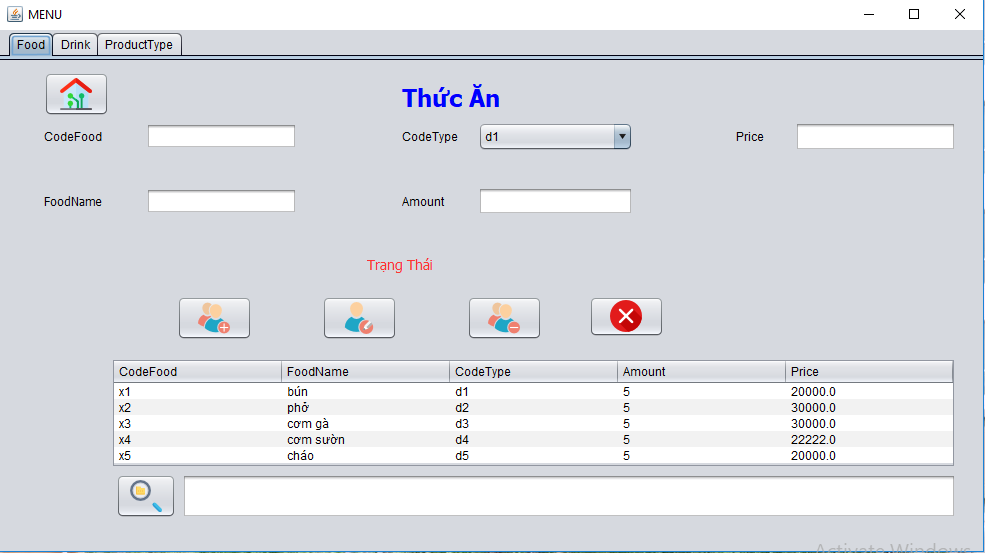
* To keep check on the shift duties, users can fill in information such as: Staff code, full name, duty. The software will retrieve any information related to the person’s shift duties.

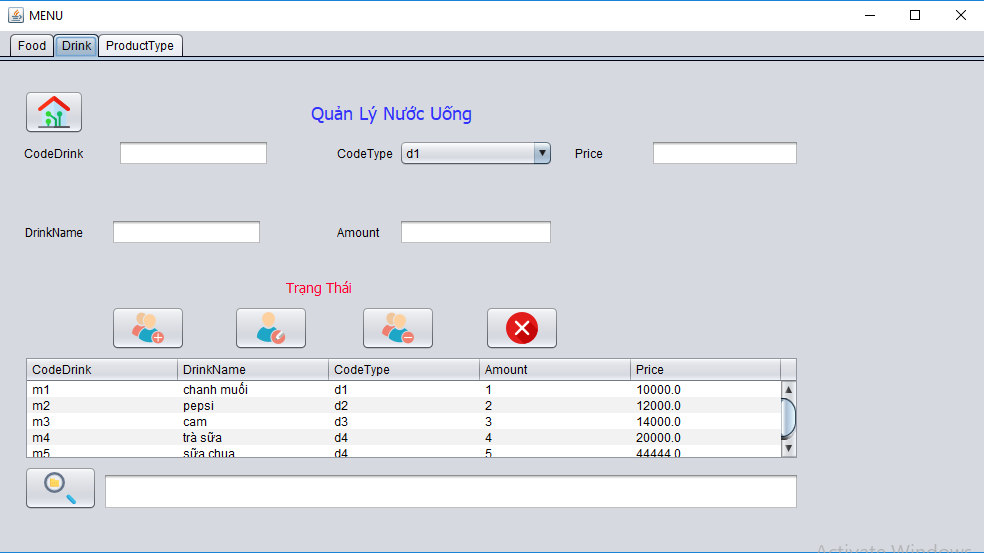


**Inventory Management Interface:**

* To keep check on the inventory, users can fill in information such as: Service code, name. The software will retrieve any information related to the products available in the inventory.







**Management Interface Calculate:**

****

1. **Conclusion**

We sincerely thank the teachers of the Center for International programmers Softech - Aptech has equipped them with basic knowledge, necessary in the past school year so I can finish this essay. In particular, we would like to sincerely thank Mr.Nguyen Duc Thao, who heartedly guided us throughout the project implementation over time.