****

**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG**

**THÀNH PHỐ HỒ CHÍ MINH**

**KHOA HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM**



**TIỂU LUẬN**

**PHƯƠNG PHÁP PHÁT TRIỂN PHẦN MỀM**

**HƯỚNG ĐỐI TƯỢNG**

**QUẢN LÝ BÁN RƯỢU**

Sinh viên thực hiện: **Vương Gia Huy**

Mã sinh viên : **0950080032**

Lớp **: 09\_ĐH\_CNPM2**

Khoá  **: 2020 - 2024**

Giảng viên hướng dẫn : **ThS. Ngô Tân Khai**

***TP. Hồ Chí Minh, tháng 10 năm 2023***

**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG**

**THÀNH PHỐ HỒ CHÍ MINH**

**KHOA HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM**

A logo with a person holding a book

Description automatically generated

**TIỂU LUẬN**

**PHƯƠNG PHÁP PHÁT TRIỂN PHẦN MỀM**

**HƯỚNG ĐỐI TƯỢNG**

**QUẢN LÝ BÁN RƯỢU**

Sinh viên thực hiện: **Vương Gia Huy**

Mã sinh viên : **0950080032**

Lớp **: 09\_ĐH\_CNPM2**

Khoá  **: 2020 - 2024**

Giảng viên hướng dẫn : **ThS. Ngô Tân Khai**

***TP. Hồ Chí Minh, tháng 10 năm 202*3**

**MỤC LỤC**

[**MỤC LỤC** 3](#_Toc149056579)

[**MỞ ĐẦU** 1](#_Toc149056580)

[Chương 1: TỔNG QUAN 2](#_Toc149056581)

[1.1 Mục tiêu chính của đề tài 2](#_Toc149056582)

[1.2 Phương pháp nghiên cứu 2](#_Toc149056583)

[Chương 2: KHẢO SÁT HIỆN TRẠNG 3](#_Toc149056584)

[2.1 Xác định yêu cầu 3](#_Toc149056585)

[2.2 Phương pháp thu thập yêu cầu 3](#_Toc149056586)

[2.3 Phân tích yêu cầu 3](#_Toc149056587)

[Chương 3: PHÂN TÍCH VÀ THIẾT KẾ 4](#_Toc149056588)

[3.1 Phân tích hệ thống 4](#_Toc149056589)

[3.1.1 Sơ đồ Use case 4](#_Toc149056590)

[3.1.2 Sơ đồ Activity 10](#_Toc149056591)

[3.1.3 Sơ đồ luân chuyển [2] 12](#_Toc149056592)

[3.2 Thiết kế hệ thống 13](#_Toc149056593)

[3.2.1 Thiết kế dữ liệu 13](#_Toc149056594)

[3.2.2 Thiết kế giao diện 15](#_Toc149056595)

[Chương 4: CÀI ĐẶT VÀ THỬ NGHIỆM 16](#_Toc149056596)

[4.1 Dữ liệu thử nghiệm 16](#_Toc149056597)

[4.2 Cài đặt phần mềm 18](#_Toc149056598)

[4.2.1 Môi trường cài đặt 18](#_Toc149056599)

[4.2.2 Ngôn ngữ cài đặt 18](#_Toc149056600)

[4.2.3 Nội dung chính cài đặt 18](#_Toc149056601)

[4.3 Thử nghiệm phần mềm 23](#_Toc149056602)

[Chương 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 24](#_Toc149056603)

[5.1 Kết quả đạt được của đề tài 24](#_Toc149056604)

[5.2 Hạn chế của đề tài 24](#_Toc149056605)

[5.3 Hướng phát triển 24](#_Toc149056606)

[**TÀI LIỆU THAM KHẢO** 25](#_Toc149056607)

**MỞ ĐẦU**

Môn học phương pháp phát triển phần mềm hướng đối tượng là môn học giúp ta có thể xây dựng, phát triển một phần mềm theo hướng đối tượng bằng các ngôn ngữ như Java, Python, … trong đó bao gồm 1 quy trình phát triển phần mềm đầy đủ.

Hiện nay, quản lý quán rượu là 1 lĩnh vực kinh doanh lớn và có tiềm năng phát triển. Với sự phát triển công nghệ trong thời đại cách mạng công nghiệp 4.0 này thì thị trường này ngày càng trở nên sôi động hơn.

Đó là lí do em chọn đề tài này để xây dựng 1 phần mềm theo hướng đối tượng giúp cho việc quản lý trở nên thuận tiện và dễ dàng hơn.

# Chương 1: TỔNG QUAN

## Mục tiêu chính của đề tài

Việc xây dựng phần mềm quản lý rượu là 1 chủ đề có tính ứng dụng cao, giúp cho ta hiểu rõ hơn về cách thức quản lý, những vấn đề cần lưu ý, những khó khăn và thách thức. Dựa vào đó, phần mềm này sẽ giúp cho việc quản lý trở nên dễ dàng, thuận tiện hơn trong thời đại công nghệ số này.

## Phương pháp nghiên cứu

Đề tài này chủ yếu thực hiện phương pháp nghiên cứu tài liệu, tập trung vào việc thu thập và phân tích các tài liệu như sách, báo, tài liệu học thuật, các bài báo khoa học. Cụ thể ở đây là nghiên cứu trên mạng và các tài liệu được cung cấp từ thầy ThS. Ngô Tân Khai.

Nội dung chính của đề tài gồm năm phần:

+ Chương 1: Tổng quan

Chương này sẽ có các nội dung như là mục tiêu của việc xây dựng phần mềm quản lý bán rượu và phương pháp nghiên cứu để thực hiện đề tài này.

+ Chương 2: Khảo sát hiện trạng

Chương này bao gồm cách xác định yêu cầu cần có, phân tích các yêu cầu của đề tài từ đó đưa ra các chức năng cho phần mềm.

+ Chương 3: Phân tích và thiết kế

Chương này sẽ dựa trên yêu cầu đã được nêu ở chương 2 từ đó phân tích yêu cầu, dựa vào những yêu cầu bằng lời từ đó thiết kế ra các mô hình hệ thống cần phải có.

+ Chương 4: Cài đặt và thử nghiệm

Chương này bao gồm các dữ liệu mẫu dùng để thử nghiệm trên hệ thống. Bên cạnh đó là cấu hình của máy tính mà phần mềm có thể chạy được. Những đoạn chương trình quan trọng và quá trình kiểm thử.

+ Chương 5: Kết luận và hướng phát triển

Chương này bao gồm các mục tiêu đã đạt được, những khó khăn chưa được giải quyết và hướng phát triển trong tương lai.

# KHẢO SÁT HIỆN TRẠNG

## Xác định yêu cầu

Việc quản lý bán rượu hiện nay đòi hỏi 1 phần mềm được quản lý bởi nhiều đơn vị khác nhau. Cụ thể, là quản lý sản phẩm, quản lý khách hàng và quản lý nhân viên. Bên cạnh đó, phần mềm cũng được thiết kế 1 cách dễ sử dụng, thuận tiện.

## Phương pháp thu thập yêu cầu

Phương pháp thu thập yêu cầu là quá trình thu thập, xác định, và hiểu rõ các yêu cầu của một hệ thống hoặc ứng dụng cụ thể từ các bên liên quan như khách hàng, người dùng cuối, hoặc các bên liên quan khác. Quá trình này rất quan trọng trong quá trình phát triển phần mềm vì nó giúp xác định rõ những gì hệ thống hoặc ứng dụng cần phải thực hiện để đáp ứng yêu cầu của các bên liên quan.

Trong đề tài này, em thực hiện phương pháp nghiên cứu tài liệu để hiểu rõ hơn yêu cầu trong việc xây dựng phần mềm quản lý bán rượu.

## Phân tích yêu cầu

Dựa vào yêu cầu được trình bày ở 2.1, phần mềm phải có các chức năng như đăng nhập để phân quyền cho các tác nhân (cụ thể là khách hàng, nhân viên, người chủ) khác nhau, mỗi tác nhân sẽ có thể tương tác với hệ thống bằng các chức năng khác nhau mà tác nhân còn lại không thực hiện được.

# PHÂN TÍCH VÀ THIẾT KẾ

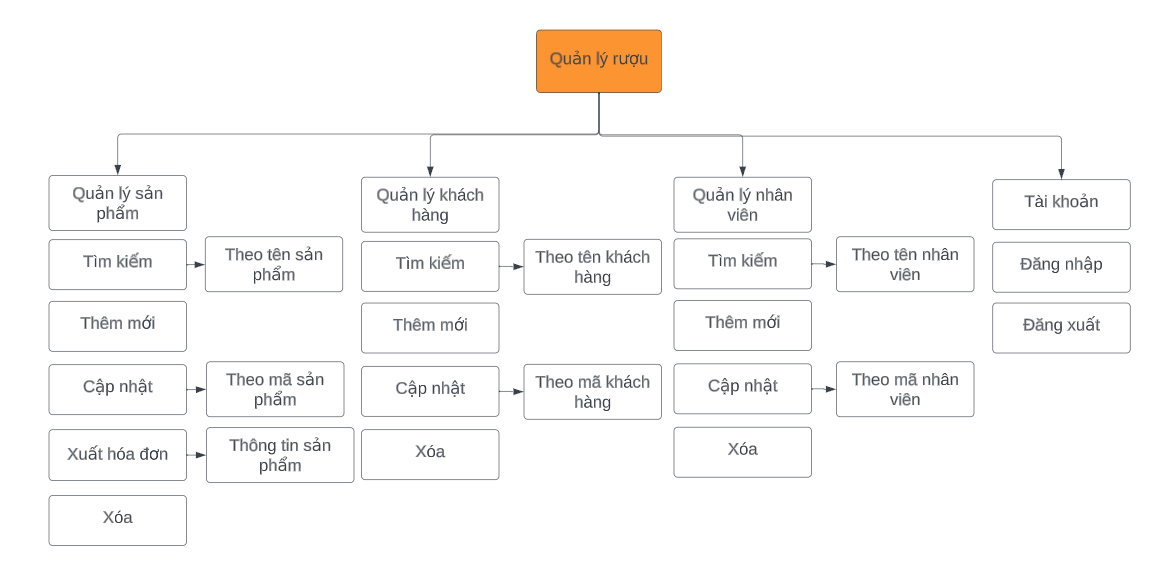
## Phân tích hệ thống

+ Phân tích yêu cầu nghiệp vụ [1]

* Quản lý sản phẩm: Hệ thống cần có tính năng cho phép quản lý danh mục sản phẩm, bao gồm thông tin về các sản phẩm cụ thể ở đây là các loại rượu. Hệ thống cũng cần cho phép thêm, sửa đổi và xóa sản phẩm.
* Quản lý nhân viên: Hệ thống cũng cần có tính năng quản lý danh sách nhân viên làm việc tại cửa hàng. Gồm các thông tin cá nhân của nhân viên.
* Quản lý khách hàng: Hệ thống cần có tính năng cho phép quản lý danh sách khách hàng đã mua hàng, bao gồm thông tin của khách hàng. Hệ thống cũng có thể sửa đổi thông tin khách hàng trong trường hợp nhập sai.
* Thanh toán và xuất hóa đơn: Khách hàng có thể vào xem và thực hiện giao dịch trên hệ thống và sau đó có thể xuất hóa đơn.

### 3.1.1 Sơ đồ Use case

+ Biểu đồ phân cấp chức năng:



+ Danh sách các tác nhân:

|  |  |
| --- | --- |
| Tác nhân | Mô tả |
| Người quản lý sản phẩm | Là người quản lý rượu, có trách nhiệm trong các hoạt động như thêm, xóa, sửa sản phẩm. Cụ thể ở đây là nhân viên cửa hàng quản lý. |
| Người quản lý nhân viên | Là người quản lý các nhân viên làm việc tại cửa hàng. Thực hiện bởi chủ cửa hàng (người quản trị). |
| Người quản lý khách hàng | Chịu trách nhiệm trong việc quản lý thông tin khách hàng. Thực hiện bởi nhân viên cửa hàng |
| Khách hàng | Thực hiện xem danh sách sản phẩm và thanh toán. |

+ Danh sách các nghiệp vụ:

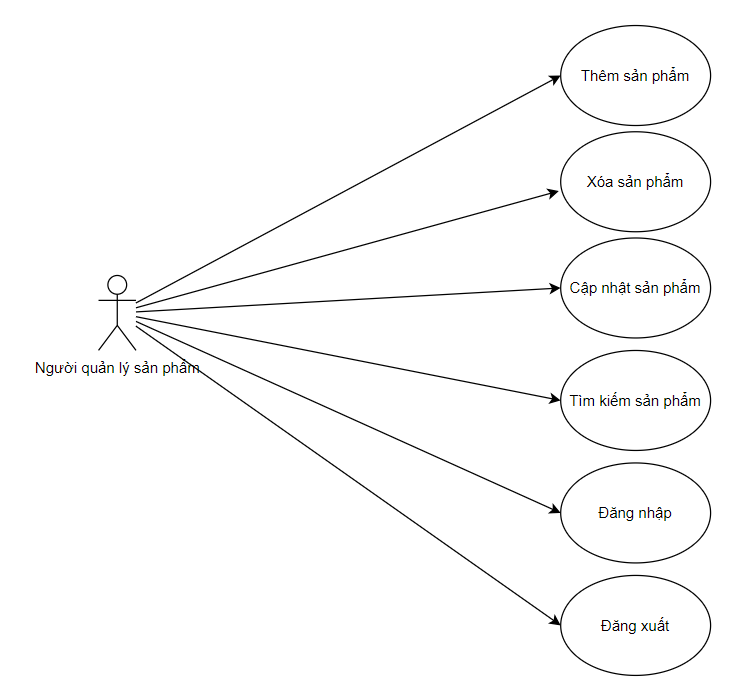
|  |  |
| --- | --- |
| Tác nhân | Nghiệp vụ |
| Người quản lý sản phẩm | Thực hiện đăng nhập vào hệ thống. Sau đó mới có thể thêm, xóa, sửa thông tin sản phẩm và đăng xuất. |
| Người quản lý nhân viên | Thực hiện đăng nhập vào hệ thống. Sau đó mới có thể thêm, xóa, sửa thông tin nhân viên và đăng xuất. |
| Người quản lý khách hàng | Thực hiện đăng nhập vào hệ thống. Sau đó mới có thể xem thông tin, chỉnh sửa và xuất thông tin khách hàng và đăng xuất. |
| Khách hàng | Thực hiện xem danh sách sản phẩm. Sau đó thanh toán. |

+ Thiết kế biểu đồ Use Case:

#### 3.1.1.1 Người quản lý sản phẩm

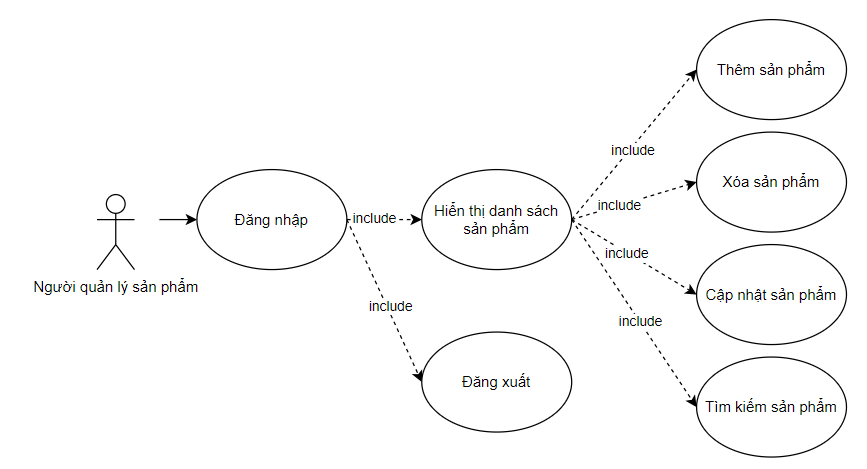
Trong biểu đồ này, tác nhân là "người quản lý sản phẩm", đại diện cho người sử dụng hệ thống. Các use case (chức năng) được hiển thị ở bên phải của biểu đồ và liên quan đến việc quản lý sản phẩm. Việc sử dụng hệ thống để thực hiện các use case này được thể hiện bằng các mũi tên hướng từ "người quản lý" đến các use case tương ứng.

##### 3.1.1.1.1 Use case tổng quát:



##### 3.1.1.1.2 Use case chi tiết

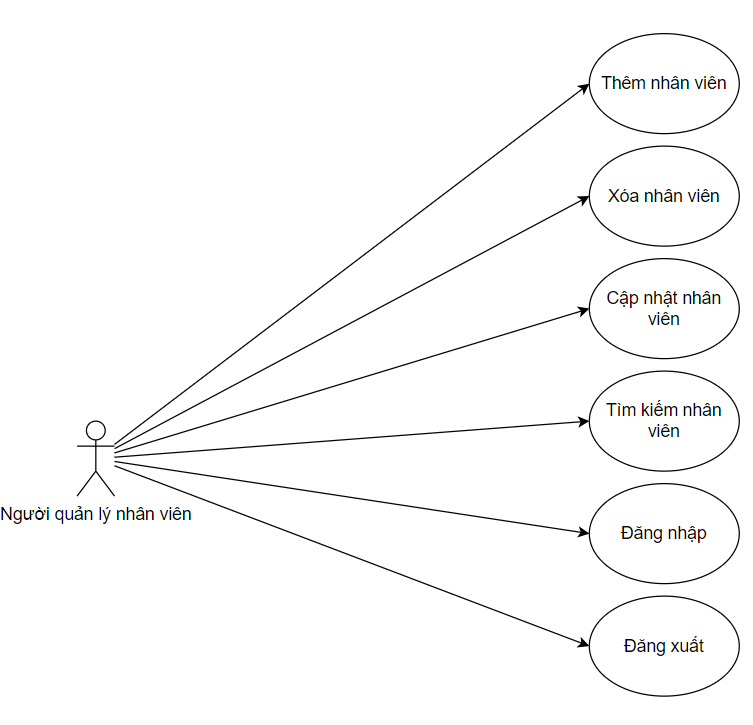
Trong biểu đồ này, từ khóa “include” thể hiện một use case chỉ được thực hiện khi đã thực hiện use case trước đó. Ví dụ, người quản lý phải thực hiện đăng nhập trước trước khi có thể đăng xuất hoặc xem danh sách sản phẩm.



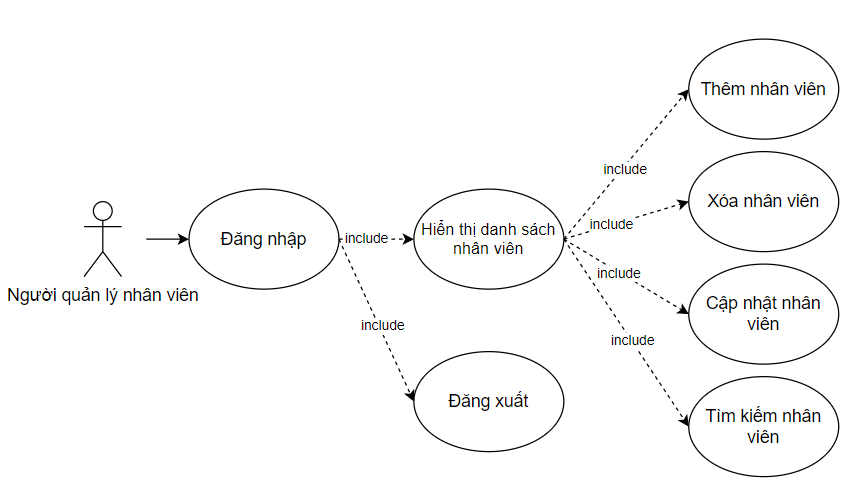
#### 3.1.1.2 Người quản lý nhân viên

Tương tự, người quản lý nhân viên cũng có các nghiệp vụ để quản lý nhân viên. Cụ thể ở đây là chủ cửa hàng.

##### 3.1.1.2.1 Use case tổng quát:



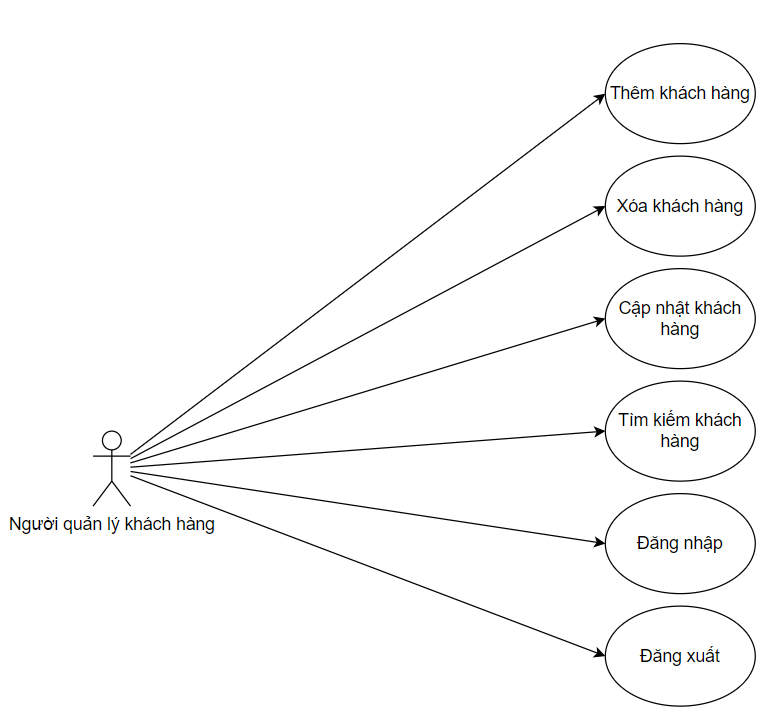
##### 3.1.1.2.2 Use case chi tiết:



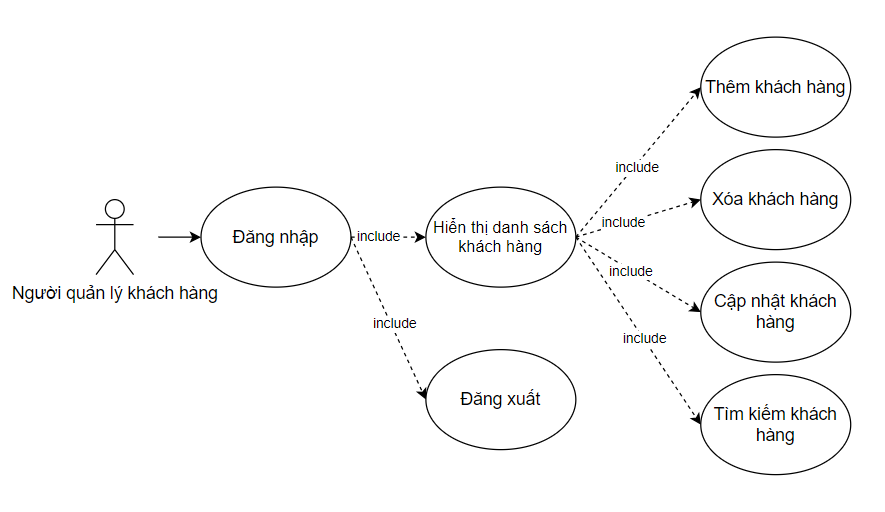
#### 3.1.1.3 Người quản lý khách hàng

Các nghiệp vụ của người quản lý khách hàng.

##### 3.1.1.3.1 Use case tổng quát:



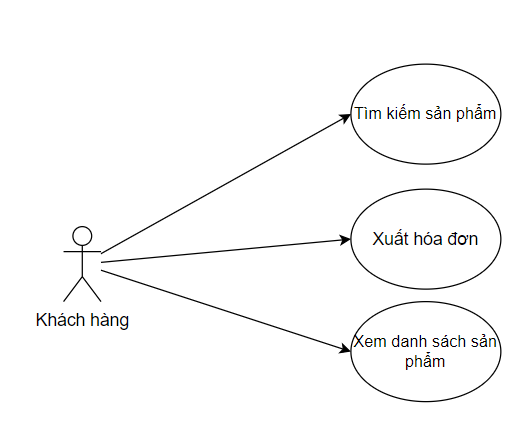
##### 3.1.1.3.2 Use case chi tiết:



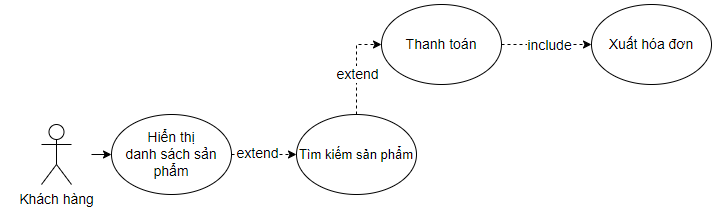
#### 3.1.1.4 Khách hàng

Trong biểu đồ này, tác nhân là “khách hàng” thực hiện các use case (chức năng) trong hệ thống. Khác với các biểu đồ use case trước, khách hàng không có use case đăng nhập và đăng xuất.

##### 3.1.1.4.1 Use case tổng quát:

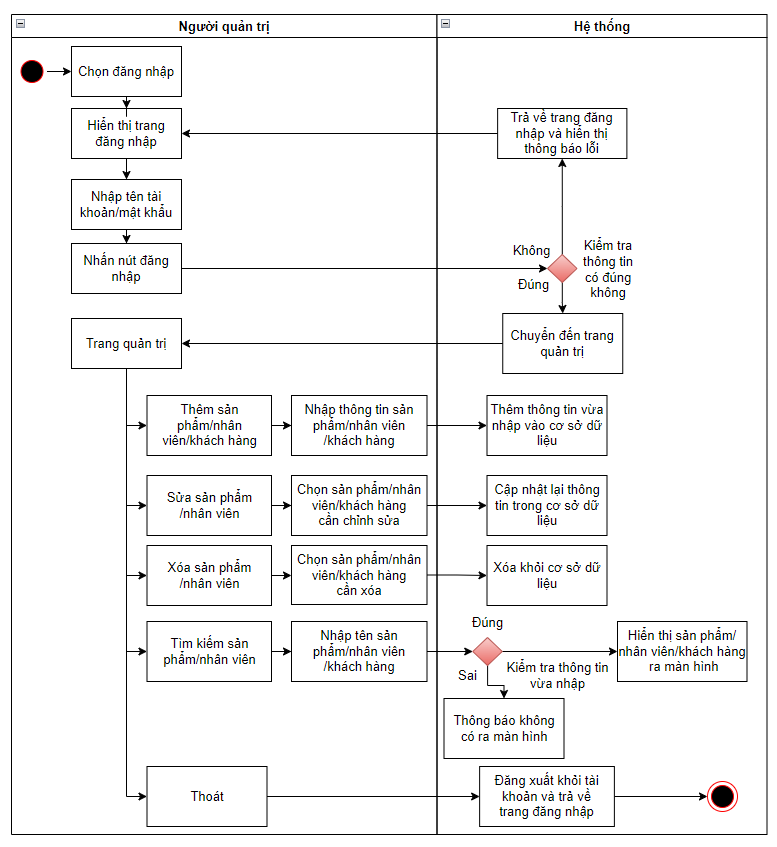


##### Use case chi tiết:

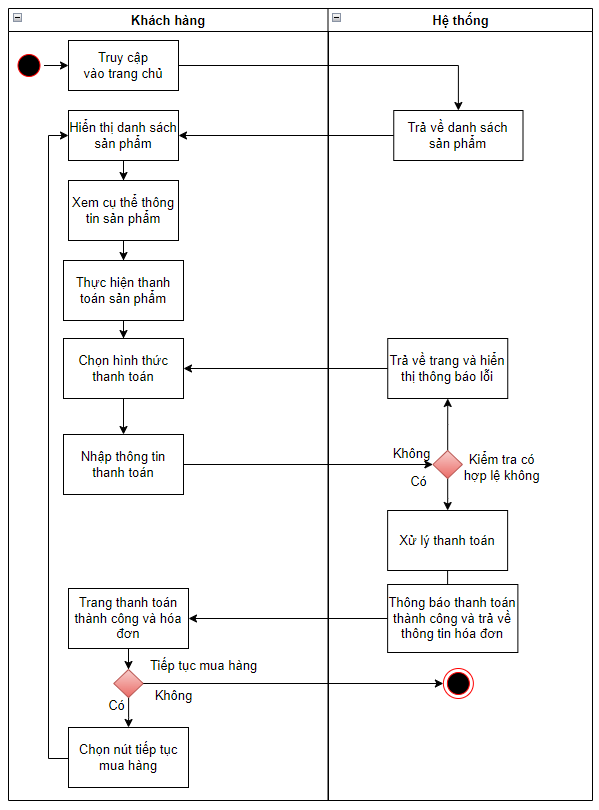


### 3.1.2 Sơ đồ Activity

#### 3.1.1.1 Sơ đồ Activity của người quản trị (người quản lý nhân viên, sản phẩm và khách hàng)

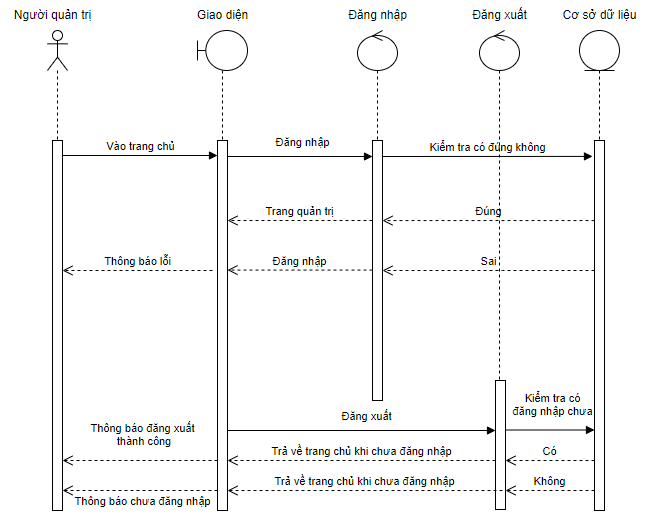


#### Sơ đồ Activity cho khách hàng

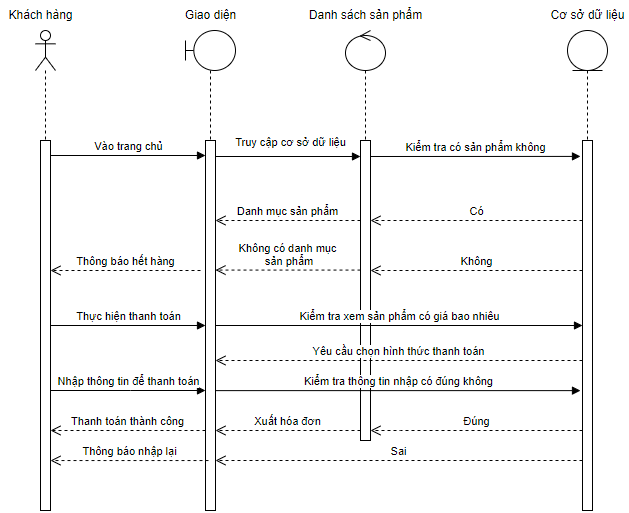


### 3.1.3 Sơ đồ luân chuyển [2]

#### 3.1.2.1 Sơ đồ luân chuyển cho người quản trị



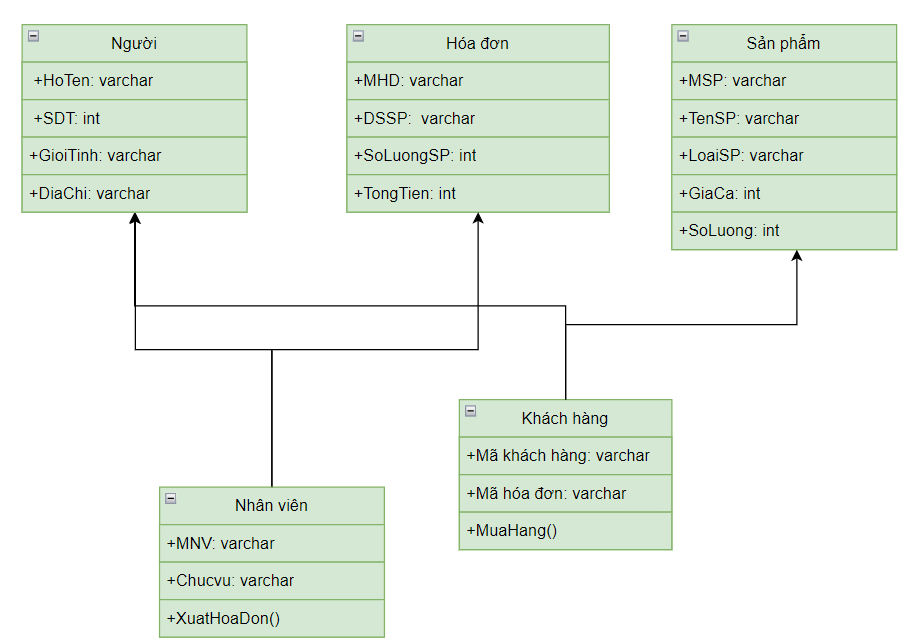
#### 3.1.2.2 Sơ đồ luân chuyển cho khách hàng



## Thiết kế hệ thống

### Thiết kế dữ liệu

Thiết kế biểu đồ lớp (Class Diagram)



+ Cấu trúc các bảng:

* Bảng Khách hàng:

|  |  |  |  |
| --- | --- | --- | --- |
| Tên dữ liệu | Kiểu dữ liệu | Độ dài | Mô tả |
| MKH | varchar | 5 | Mã khách hàng |
| HoTen | varchar | 30 | Họ tên khách hàng |
| GioiTinh | varchar | 3 | Giới tính khách hàng |
| SDT | int |  | Số điện thoại khách hàng |
| DiaChi | varchar | 50 | Địa chỉ khách hàng |
| MHD | varchar | 5 | Mã hóa đơn |

* Bảng Sản phẩm:

|  |  |  |  |
| --- | --- | --- | --- |
| Tên dữ liệu | Kiểu dữ liệu | Độ dài | Mô tả |
| MSP | varchar | 5 | Mã sản phẩm |
| TenSP | varchar | 20 | Tên sản phẩm |
| LoaiSP | varchar | 10 | Loại sản phẩm |
| GiaCa | int |  | Giá sản phẩm |
| SoLuong | int |  | Số lượng sản phẩm |
| HinhAnh | varchar | 3000 | Hình ảnh sản phẩm |

* Bảng Hóa đơn:

|  |  |  |  |
| --- | --- | --- | --- |
| Tên dữ liệu | Kiểu dữ liệu | Độ dài | Mô tả |
| MHD | varchar | 5 | Mã hóa đơn |
| DSSP | varchar | 100 | Danh sách sản phẩm |
| SoLuongSP | int |  | Số lượng sản phẩm trong hóa đơn |
| TongTien | int |  | Tổng tiền hóa đơn |

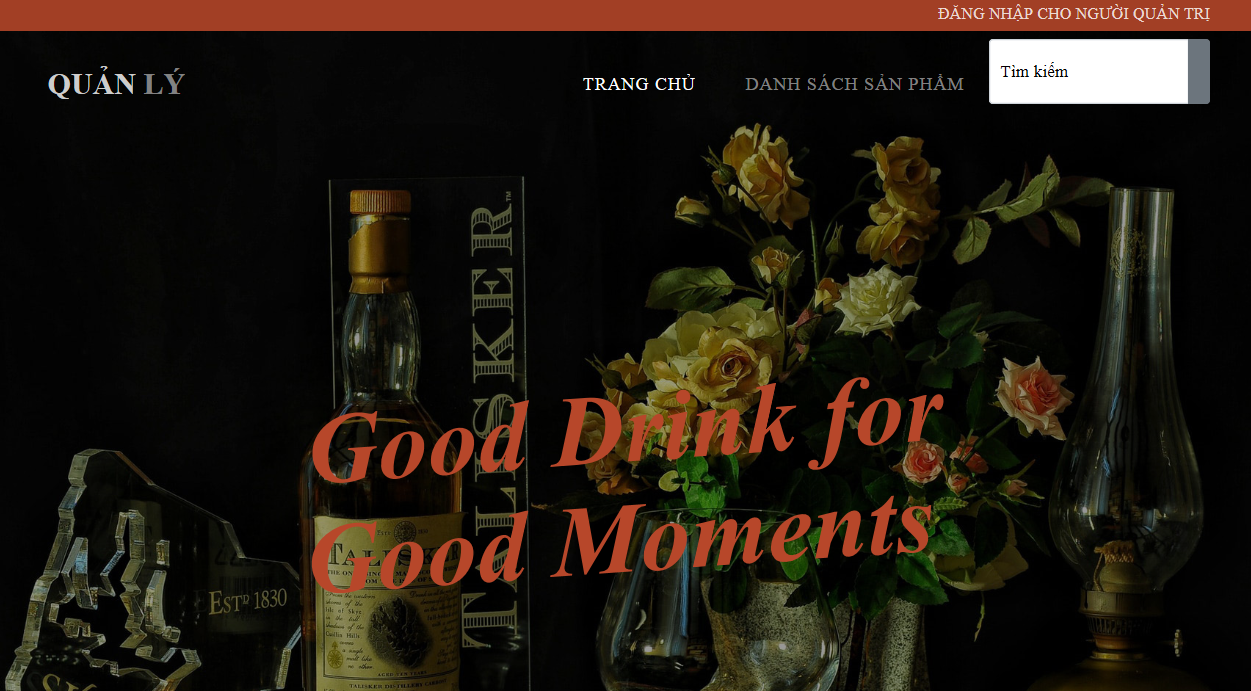
* Bảng Nhân viên:

|  |  |  |  |
| --- | --- | --- | --- |
| Tên dữ liệu | Kiểu dữ liệu | Độ dài | Mô tả |
| MNV | varchar | 5 | Mã nhân viên |
| HoTen | varchar | 30 | Họ tên nhân viên |
| GioiTinh | varchar | 3 | Giới tính nhân viên |
| SDT | int |  | Số điện thoại nhân viên |
| DiaChi | varchar | 50 | Địa chỉ nhân viên |
| ChucVu | varchar | 20 | Chức vụ nhân viên |

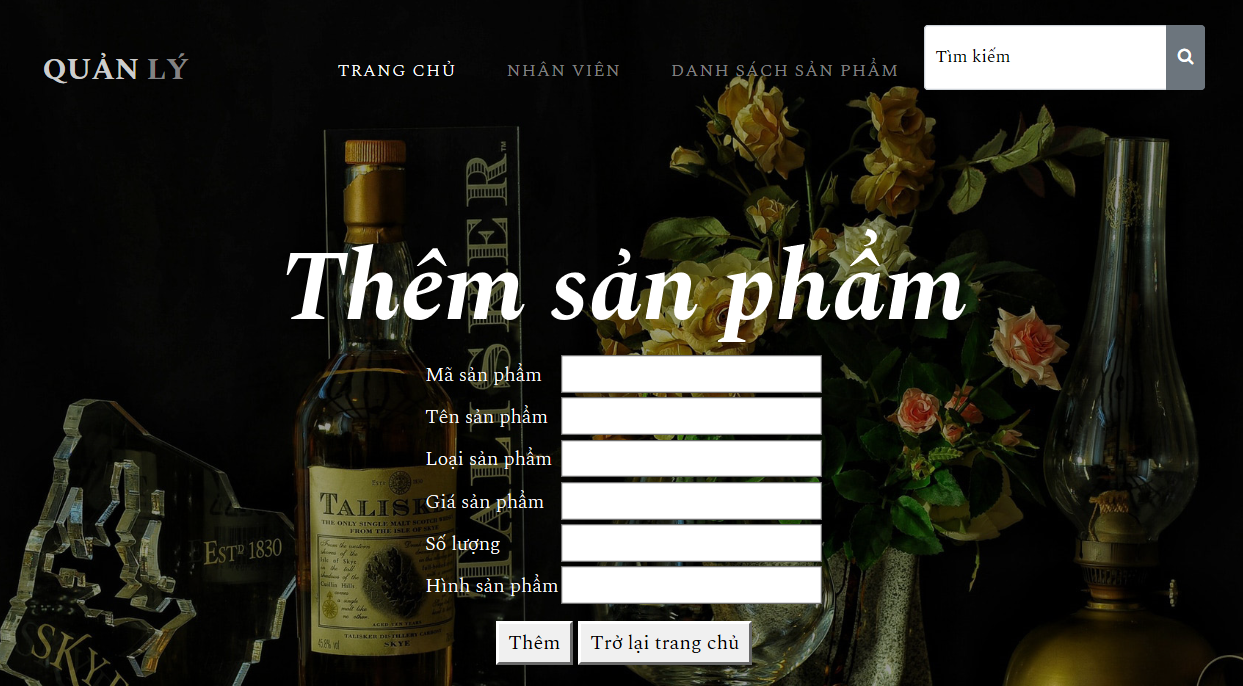
### Thiết kế giao diện

#### Màn hình chính

Màn hình chính bao gồm tổng thể của trang web. Gồm có trang chủ, đăng nhập, quản lý nhân viên, danh mục sản phẩm.

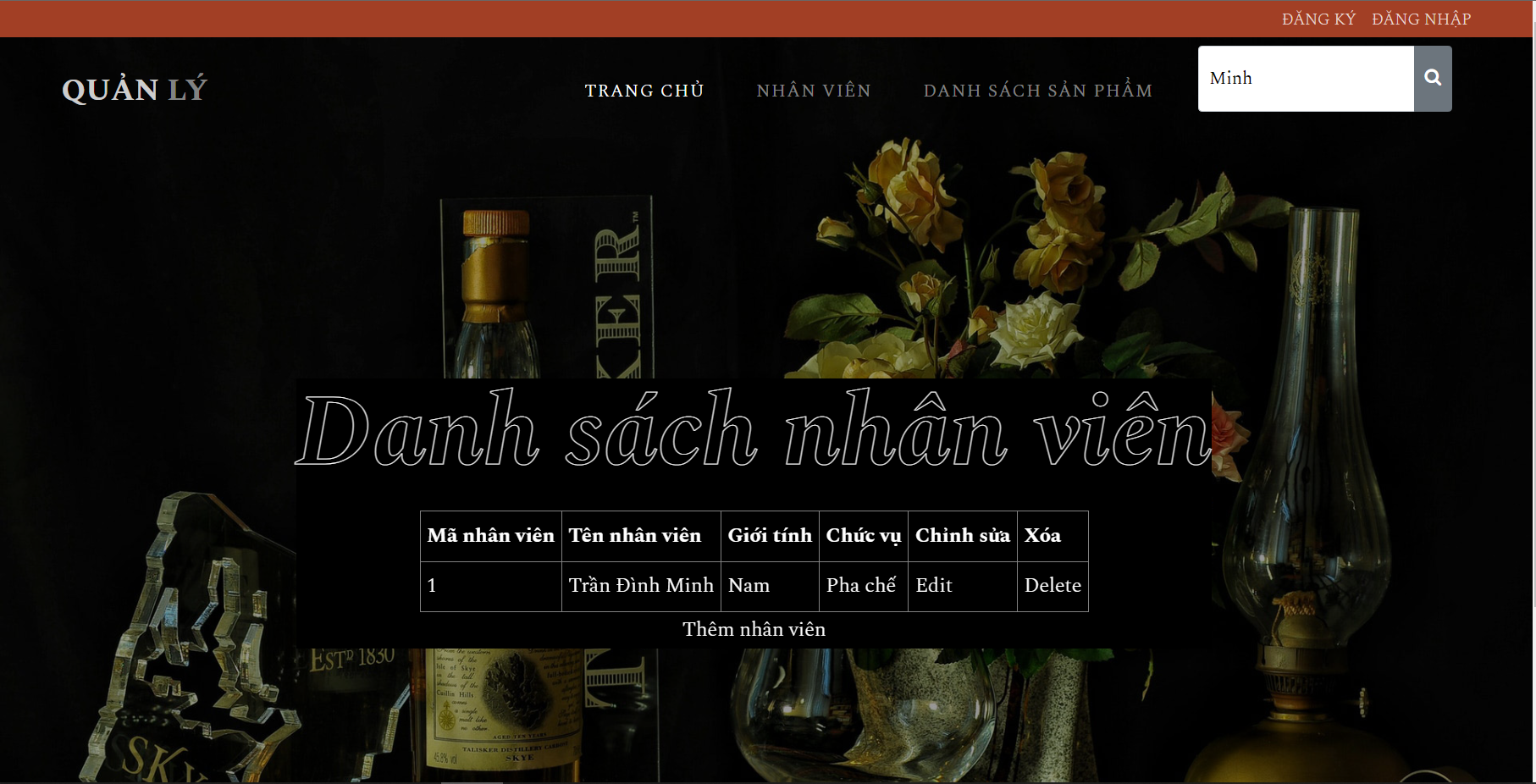


#### Màn hình nhập liệu



Sau khi đã nhập đầy đủ thông tin thì bấm nút “Thêm” để thực hiện thêm thông tin, bấm nút “Trở lại trang chủ” để trở về màn hình chính.

#### Màn hình tìm kiếm



# CÀI ĐẶT VÀ THỬ NGHIỆM

## Dữ liệu thử nghiệm

Bảng Nhân viên:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Mã nhân viên | Họ tên nhân viên | SĐT | Địa chỉ | Giới tính | Chức vụ |
| NV1 | Nguyễn Huy | 0931238931 | 550 Nam Hòa, Phước Long A, quận 9 | Nam | Pha chế |
| NV2 | Trần Bình Minh | 0942180421 | 550 Tầm Vu, phường 26, quận Bình Thạnh | Nam | Pha chế |
| NV3 | Nguyễn Tiến Đạt | 0941924211 | 650 3 tháng 2, phường 6, quận 10 | Nam | Bồi bàn |
| NV4 | Nguyễn Bảo Khánh | 0917384111 | 20 Phạm Ngũ Lão, phường 3, quận Gò Vấp | Nam | Bảo vệ |
| NV5 | Trần Văn Hoàng | 0940120805 | 2 Nguyễn Trãi, phường 11, quận 5 | Nam | Pha chế |

Bảng Sản phẩm:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Mã sản phẩm | Tên sản phẩm | Loại sản phẩm | Giá cả | Số lượng | Hình ảnh |
| SP1 | Bourbon | Whisky | 25000 | 5 |  |
| SP2 | Captain Morgan | Rum | 50000 | 4 |  |
| SP3 | Smirnoff | Vodka | 35000 | 6 |  |
| SP4 | Ginjo | Sake | 45000 | 8 |  |
| SP5 | Veuve Clicquot | Champagne | 45000 | 2 |  |

Bảng Hóa đơn:

|  |  |  |  |
| --- | --- | --- | --- |
| Mã hóa đơn | Danh sách sản phẩm | Số lượng sản phẩm | Tổng tiền |
| HD1 | Bourbon, Ginjo | 2 | 70000 |
| HD2 | Veuve Clicquot | 1 | 45000 |
| HD3 | Smirnoff | 2 | 70000 |
| HD4 | Captain Morgan | 1 | 50000 |
| HD5 | Ginjo | 1 | 45000 |

Bảng Khách hàng:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mã khách hàng | Họ tên | Số điện thoại | Địa chỉ | Mã hóa đơn |
| KH1 | Vương Gia Huy | 0901457288 | 207/52/94 Nguyễn Văn Đậu P11, quận Bình Thạnh, TP. HCM | HD1 |
| KH2 | Trần Đình Minh | 0941493892 | 04/3C Song Hành, xã Trung Chánh, quận 12, TP. HCM | HD02 |
| KH3 | Phạm Phú Đạt | 0955888239 | 727 Xóm Bắp, xã Trung Chánh, quận 12, TP. HCM | HD03 |
| KH4 | Nguyễn Phương Bảo Khánh | 0941023823 | B25, khu Tái định cư, phường Chánh Nghĩa, TP. Thủ Dầu Một, Tỉnh Bình Dương | HD04 |
| KH5 | Trần Văn Hoàng | 094242213 | 23/1A Nơ Trang Long, P7, quận Bình Thạnh, TP. HCM | HD05 |

## Cài đặt phần mềm

### Môi trường cài đặt

Phần cứng:

* CPU: Intel core i5 1157G7 2.5Ghz
* RAM: 8GB
* Ổ cứng: 512GB

Phần mềm:

Hệ điều hành Windows 10, 64 bit

### Ngôn ngữ cài đặt

Sử dụng eclipse jee 2018, XAMPP 8.1.17/PHP 8.1.17.

### Nội dung chính cài đặt

+ Hàm thao tác trên cơ sở dữ liệu MySQL của bảng sản phẩm [4]:

**public** **static** List<Product> queryProduct(Connection conn) **throws** SQLException {

String sql = "Select a.id, a.name, a.type, a.price, a.soluong, a.image from Product a ";

PreparedStatement pstm = conn.prepareStatement(sql);

ResultSet rs = pstm.executeQuery();

List<Product> list = **new** ArrayList<Product>();

**while** (rs.next()) {

String id = rs.getString("Id");

String name = rs.getString("Name");

String type = rs.getString("Type");

**int** price = rs.getInt("Price");

**int** soluong = rs.getInt("soluong");

String image = rs.getString("Image");

Product product = **new** Product();

product.setId(id);

product.setName(name);

product.setType(type);

product.setPrice(price);

product.setSoLuong(soluong);

product.setImage(image);

list.add(product);

}

**return** list;

}

**public** **static** List<Product> Search(Connection conn,String txtSearch) **throws** SQLException {

List<Product> list = **new** ArrayList<Product>();

String sql = "Select \* from product where Name like ? ";

PreparedStatement pstm = conn.prepareStatement(sql);

pstm.setString(1, "%"+txtSearch+"%");

ResultSet rs = pstm.executeQuery();

**while** (rs.next()) {

String id = rs.getString("id");

String name = rs.getString("name");

String type = rs.getString("type");

**int** price = rs.getInt("price");

**int** soluong = rs.getInt("soluong");

String image = rs.getString("image");

Product product = **new** Product();

product.setId(id);

product.setName(name);

product.setType(type);

product.setPrice(price);

product.setSoLuong(soluong);

product.setImage(image);

list.add(product);

}

**return** list;

}

**public** **static** **void** updateProduct(Connection conn, Product product) **throws** SQLException {

String sql = "Update Product set Name =?, Type =?, Price=?, soluong=?, Image=? where Id=? ";

PreparedStatement pstm = conn.prepareStatement(sql);

pstm.setString(1, product.getName());

pstm.setString(2, product.getType());

pstm.setFloat(3, product.getPrice());

pstm.setInt(4, product.getSoLuong());

pstm.setString(5, product.getImage());

pstm.setString(6, product.getId());

pstm.executeUpdate();

}

**public** **static** **void** insertProduct(Connection conn, Product product) **throws** SQLException {

String sql = "Insert into Product(id, name, type, price, soluong, image) values (?,?,?,?,?,?)";

PreparedStatement pstm = conn.prepareStatement(sql);

pstm.setString(1, product.getId());

pstm.setString(2, product.getName());

pstm.setString(3, product.getType());

pstm.setFloat(4, product.getPrice());

pstm.setInt(5, product.getSoLuong());

pstm.setString(6, product.getImage());

pstm.executeUpdate();

}

**public** **static** **void** deleteProduct(Connection conn, String id) **throws** SQLException {

String sql = "Delete From Product where id= ?";

PreparedStatement pstm = conn.prepareStatement(sql);

pstm.setString(1, id);

pstm.executeUpdate();

}

+ Giao diện nhập liệu của Nhân viên:

<div class=*"hero-wrap"* style="background-image: *url('images/bg\_2.jpg')*;" data-stellar-background-ratio=*"0.5"*>

<div class=*"overlay"*></div>

<div class=*"container"*>

<div class=*"row no-gutters slider-text align-items-center justify-content-center"*>

<center>

<h1 style="color:*white*">Thêm nhân viên</h1>

<form method=*"POST"* action=*"newNhanVienServlet"* enctype=*"multipart/form-data"*>

<table>

<tr>

<td style="color:*white*">Mã nhân viên: </td>

<td><input type=*"text"* name=*"Id"*/></td>

</tr>

<tr>

<td style="color:*white*">Tên nhân viên: </td>

<td><input type=*"text"* name=*"Username"*/></td>

</tr>

<tr>

<td style="color:*white*">Giới tính: </td>

<td><input type=*"text"* name=*"Gender"*></td>

</tr>

<tr>

<td style="color:*white*">Chức vụ: </td>

<td><input type=*"text"* name=*"Chucvu"*/></td>

</tr>

<tr>

<td colspan=*"2"*>

<a href=*"NhanVienListServlet"*>

<center>

<button style="margin-top: *10px*;">Thêm</button>

</a>

</a>

</td>

</tr>

</table>

</center>

</div>

<a href=*"index.jsp"*>Trở lại trang chủ</a>

</div>

</div>

+ Kết nối đến cơ sở dữ liệu MySQL [4]:

**package** Servlets;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.SQLException;

**public** **class** MySQLConntUtils {

**public** **static** Connection getMySQLConnection() **throws** ClassNotFoundException,

SQLException{

String hostname="localhost";

String dbName="quanlybanhang";

String username="root";

String password="";

**return** *getMySQLConnection*(hostname,dbName,username, password);

}

**private** **static** Connection getMySQLConnection(String hostname, String dbName, String

username, String password)

**throws** ClassNotFoundException, SQLException {

Class.*forName*("com.mysql.jdbc.Driver");

String connectionURL="jdbc:mysql://"+hostname+":3306/"+dbName;

Connection conn=DriverManager.*getConnection*(connectionURL, username, password);

**return** conn;

}

}

+Thực hiện kết nối đến cơ sở dữ liệu sau đó chạy các hàm thao tác trên danh sách sản phẩm [3]:

**package** Servlets;

**import** java.io.IOException;

**import** java.sql.Connection;

**import** java.sql.SQLException;

**import** java.util.ArrayList;

**import** java.util.List;

**import** javax.servlet.RequestDispatcher;

**import** javax.servlet.ServletException;

**import** javax.servlet.annotation.WebServlet;

**import** javax.servlet.http.HttpServlet;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

@WebServlet("/ProductListServlet")

**public** **class** ProductListServlet **extends** HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

/\*\*

\* **@see** HttpServlet#HttpServlet()

\*/

**public** ProductListServlet() {

**super**();

// **TODO** Auto-generated constructor stub

}

/\*\*

\* **@see** HttpServlet#doGet(HttpServletRequest request, HttpServletResponse

response)

\*/

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

// **TODO** Auto-generated method stub

request.setCharacterEncoding("utf-8");

response.setCharacterEncoding("utf-8");

response.getWriter().append("Served at: ").append(request.getContextPath());

//Connection conn = MyUtils.getStoredConnection(request);

String errorString = **null**;

List<Product> list = **new** ArrayList<Product>();

**try** {

Connection conn = MySQLConntUtils.*getMySQLConnection*();

list = DBUtils.*queryProduct*(conn);

} **catch** (SQLException e) {

e.printStackTrace();

} **catch** (ClassNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

request.setAttribute("errorString",errorString);

request.setAttribute("productList", list);

// Forward sang /WEB-INF/views/productListView.jsp

request.getRequestDispatcher("/Productlist.jsp").forward(request, response);

}

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse

response) **throws** ServletException, IOException {

// **TODO** Auto-generated method stub

request.setCharacterEncoding("utf-8");

response.setCharacterEncoding("utf-8");

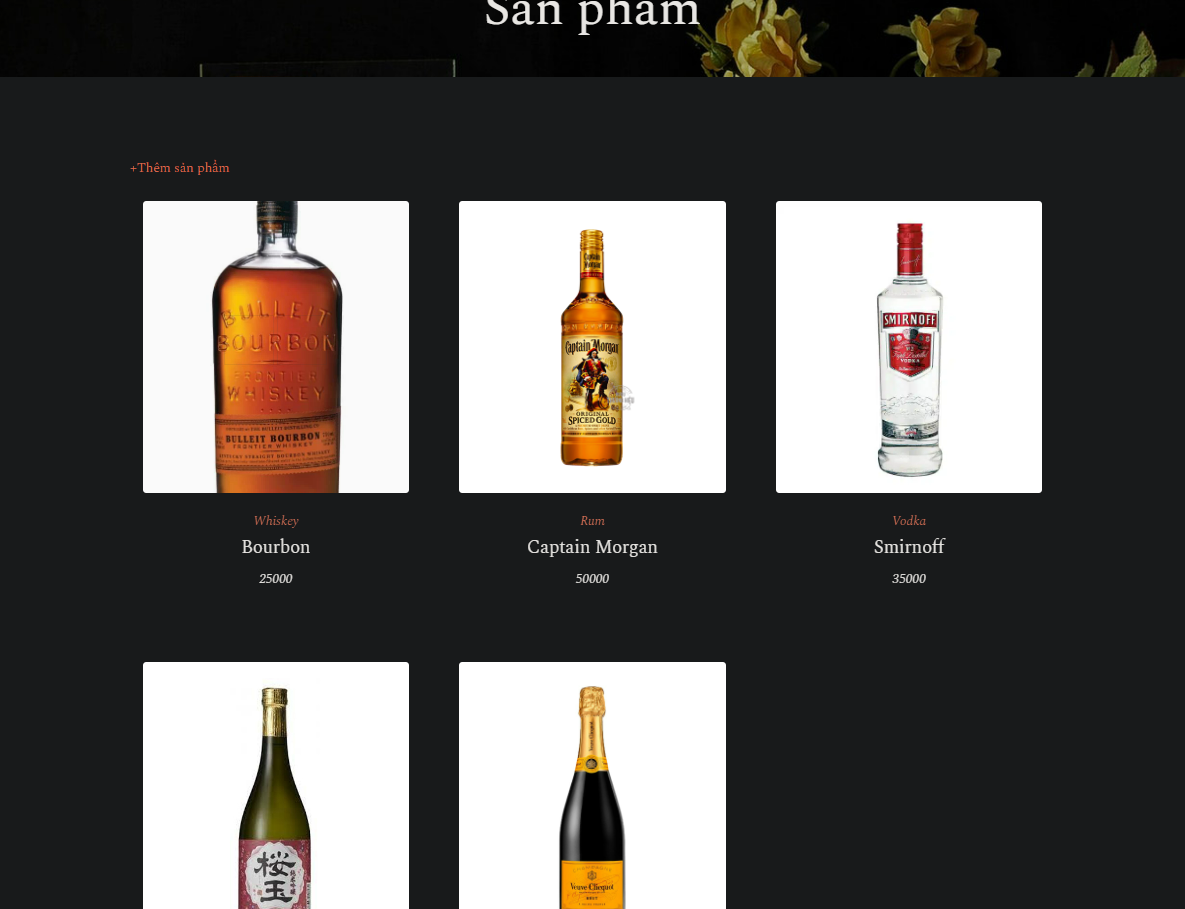
doGet(request, response);

}

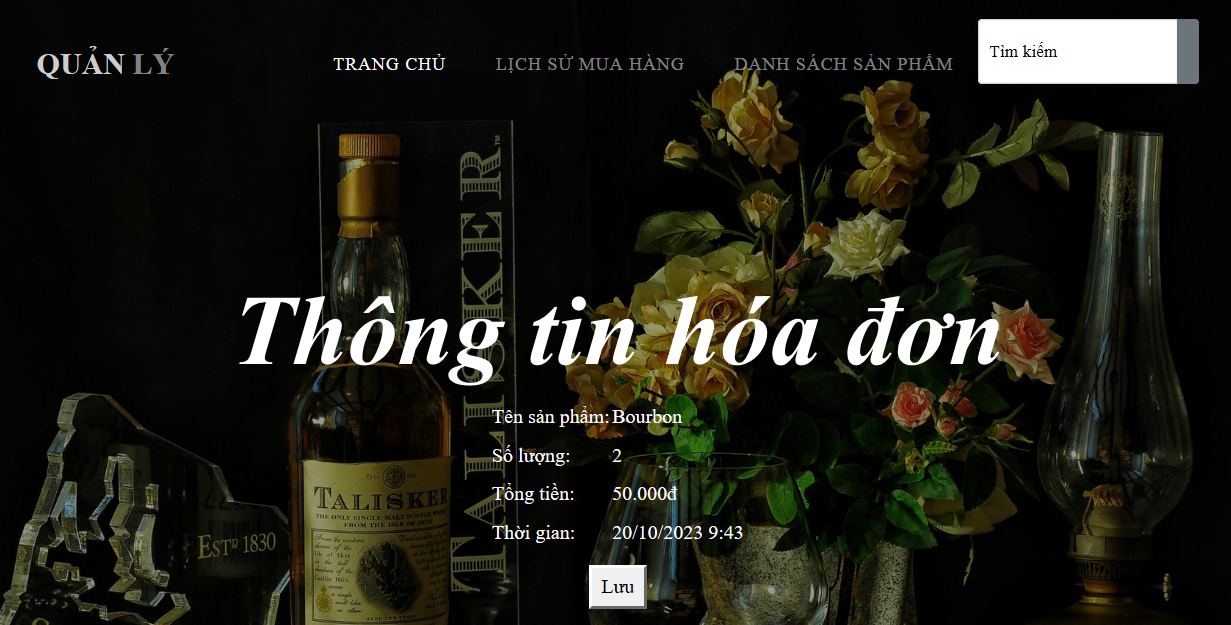
}

## Thử nghiệm phần mềm

Thực hiện thêm danh sách sản phẩm với dữ liệu mẫu (phần 4.1) vào cơ sở dữ liệu, sau đó xuất ra màn hình.



Thực hiện chạy thử phần mềm với dữ liệu mẫu và dưới đây là màn hình hóa đơn sau khi thanh toán.



# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết quả đạt được của đề tài

Kết quả đạt được:

Phần mềm có thể thực hiện đăng nhập để quản lý cho người quản lý (thêm, xóa sửa), thực hiện thanh toán và xuất hóa đơn cho khách hàng (không cần đăng nhập).

## Hạn chế của đề tài

Do phương thức cập nhật hình ảnh sản phẩm còn hạn chế nên phải có kết nối Internet thì mới có thể xem được hình ảnh sản phẩm.

Trang web vẫn chưa có các tính năng như AI, chatbot để tư vấn người mua.

## Hướng phát triển

Trang có thể phát triển lên thành trang web bán rượu như 1 trang thương mại điện tử trong tương lai bao gồm cả AI nếu có thêm thời gian, kiến thức cùng với kinh phí.

Các đề xuất cụ thể như sau:

* Thiết kế giao diện thân thiện, tiện ích cho người dùng hơn.
* Tăng cường tính bảo mật cho tài khoản.
* Tối ưu hóa trang web về tốc độ lẫn số lượng truy cập trong 1 thời điểm nhất định.
* Cập nhật hàng hóa, các chức năng mới thường xuyên (thanh toán, tư vấn, bảo hành, hướng dẫn sử dụng trang web, vv).
* Triển khai các chiến dịch marketing để thu hút khách hàng.
* Áp dụng trí tuệ nhân tạo vào trang web.

**TÀI LIỆU THAM KHẢO**

1. Đinh Khắc Quyền, Lê Xuân Thọ, “Phân tích và thiết kế hệ thống thông tin”, Đại học Cần Thơ.
2. Lê Thị Thanh Trúc, “Phân tích thiết kế hệ thống hướng đối tượng”, Đại học Tây Đô, 2015.
3. Nguyễn Phương Lan, "Lập Trình ứng dụng Web với Java Servlet và JSP", 2004.
4. Trần Nguyên Phong, “Giáo Trình SQL”, Đại học Huế, 2004.