

**TRƯỜNG ĐẠI HỌC BÀ RỊA – VŨNG TÀU**  
**VIỆN CÔNG NGHỆ THÔNG TIN**  
**ĐIỆN – ĐIỆN TỬ**



**BARIA VUNGTAU**  
**UNIVERSITY**  
CAP SAINT JACQUES

**BÁO CÁO LẬP TRÌNH .NET**

**Đề tài:** Phần mềm Quán Coffee T&T

**Giảng viên hướng dẫn:** *ThS. Lê Thị Vĩnh Thanh*

**Sinh viên thực hiện:**

Nguyễn Huỳnh Hải Phương -17032376

Nguyễn Thị Minh Thắm-17032486

Lê Thành Trung-17031538

## ĐÁNH GIÁ CỦA GIÁO VIÊN

[illegible]

**Giáo Viên Xác Nhận**

## Lời Nói Đầu

Xã hội càng ngày càng hiện đại thì nhu cầu ăn uống của mỗi con người tăng cao. Nhận biết được tình hình trên nhiều quán nước giải khát, café mọc lên như nấm.

Tuy nhiên, theo sự khảo sát thì một số quán nước về vấn đề ghi chép dữ liệu thì được lưu trực tiếp trên giấy và có khi không cần ghi chép vì là quán gia đình mở.

Tác hại của việc làm trên là khi muốn kiểm tra xem tháng này lời hay lỗ, khách hàng A có thường tới hay không, loại nước uống nào được khách ưa dùng, .....lập ra báo cáo doanh thu thì gặp nhiều khó khăn và trở ngại. Nguyên nhân là do dữ liệu không được đồng nhất(thống nhất) và ghi chép cẩu thả, không rõ ràng.

Thấy được những khó khăn trên Nhóm chúng em đã xây dựng một phần mềm Quản lý “Quán Nước T&T”. Nó có ưu điểm là quản lý dữ liệu gọn gàng, tìm kiếm, thống kê doanh thu theo yêu cầu của chủ cửa hàng, giao diện hiện đại dễ sử dụng. Khiến việc quản lý trở nên dễ dàng và dữ liệu mua bán của quán sẽ được thể hiện một cách rành mạch hơn.

Với sự nỗ lực của các thành viên trong nhóm, chúng em đã hoàn thành đề tài trên. Tuy nhiên không tránh được sự thiếu sót. Mong thầy(cô) và các bạn góp ý để đề tài được hoàn thiện hơn.

Chúng em xin gửi lời cảm ơn chân thành tới giảng viên Lê Thị Vĩnh Thanh là người hướng dẫn tận tình và chỉ bảo để nhóm hoàn thành đề tài trên.

*Nhóm chúng em xin chân thành cảm ơn.*

## Mục Lục

I.	Khảo Sát .....	6
1.	Khảo Sát Thực Tế.....	6
2.	Khảo Sát Nghiệp Vụ.....	6
II.	Mô Tả Yêu Cầu Bài Toán .....	6
1.	Yêu Cầu Về Phần Cứng Và Phần Mềm .....	6
2.	Yêu Cầu Về Chức Năng Và Công Cụ Trên Phần Mềm .....	6
III.	Phân Tích Thiết Kế Hệ Thống.....	7
1.	Xác Định Tác Nhân.....	7
3.1	Khách Hàng .....	7
2.	Use Case Tổng Quát .....	8
3.	Mô Tả Use Case .....	8
3.1	Mô tả Use Case “Đăng nhập” cho hai nhân viên.....	8
3.2	Mô tả Use Case “Đổi mật khẩu” .....	9
3.3	Mô tả Use Case “Lập hóa đơn” .....	11
3.4	Mô tả Use Case “Cập nhật dữ liệu” .....	12
IV.	Cơ Sở Dữ Liệu.....	14
1.	Các Thực Thể Và Các Thuộc Tính.....	14
2.	Sơ Đồ ER.....	15
3.	Sơ Đồ Quan Hệ.....	15
4.	Xây Dựng và Cập Nhật Các Ràng Buộc Toàn Vẹn.....	15
V.	Thiết Kế Form .....	17
	Form Đăng Nhập .....	17
	Form Chính.....	18
	Form thông tin phần mềm .....	20
	Form Thêm món .....	20
	Form Đổi trả món .....	22
	Form giao diện quản trị viên .....	23
	Form Thông tin tài khoản .....	30
	Form Báo Cáo(Thống kê).....	30
VI.	Xử lý Form .....	31
1.	Xây dựng Class hỗ trợ (KetNoi).....	31
2.	Form Đăng Nhập (frmLogIn) .....	32
3.	Form Chính(frmMain).....	33
4.	Form thêm món(frmAddFood) .....	34
5.	Form Trả món(frmReturnFood) .....	36
6.	Form Admin(frmAdmin) .....	36

7.	Form thông tin tài khoản .....	38
8.	Form Báo Cáo.....	39
VII.	Hướng Dẫn Sử Dụng Phần Mềm.....	40
1.	Thêm Món Cho Một Bàn .....	40
2.	Trả Món Cho Một Bàn .....	40
3.	Thanh Toán Một Bàn.....	41
4.	Xem thông tin tài khoản và đổi mật khẩu.....	41
5.	Thêm, Sửa, Xóa Thức Ăn-Loại Thức Ăn-Bàn-Tài Khoản-Hóa Đơn- Chi Tiết Hóa Đơn.....	41
6.	Thống kê.....	41
VIII.	Tổng Kết & Hướng Phát Triển.....	42
1.	Tổng kết:.....	42
2.	Hướng phát triển.....	42
IX.	Tài Liệu Kham Khảo .....	42

## **I. Khảo Sát**

### **1. Khảo Sát Thực Tế**

Với cách quản lý nhân viên, quản lý hóa đơn thanh toán, lương nhân viên, hàng hóa bằng công tác thủ công không mang tính tự động, bộc lộ một số hạn chế như sau:

- +Tốn nhiều công sức mà sự hiệu quả mang lại không cao
- +Tra cứu, thống kê và hiệu chỉnh thông tin rất vất vả, đôi khi còn không chính xác
- +Việc lưu trữ thông tin cần nhiều giấy tờ cồng kềnh, dễ thất lạc và tốn kém
- +Cập nhật thông tin hằng ngày tốn nhiều thời gian.
- +Khó khăn trong quá trình thực hiện báo cáo thống kê, nhất là khi có sự cố đột xuất

### **2. Khảo Sát Nghiệp Vụ**

## **II. Mô Tả Yêu Cầu Bài Toán**

### **1. Yêu Cầu Về Phần Cứng Và Phần Mềm**

- Hỗ trợ laptop hoặc máy tính chạy Windows 7 trở lên.
- Cài Framework từ 4.0 trở lên.
- Máy tính hoặc laptop phải cài đặt SQL Server 2014 trở lên
- Chip sử dụng: tối thiểu là Core I3 (khuyến nghị sử dụng core I5)
- Kết nối được với máy in.
- Bộ nhớ trống: Tối thiểu 8Gb
- Ngôn ngữ: Tiếng Việt.
- Giao diện dễ sử dụng.

### **2. Yêu Cầu Về Chức Năng Và Công Cụ Trên Phần Mềm**

Xây dựng hệ thống để đáp ứng những yêu cầu cơ bản sau :

- Giảm khối lượng ghi chép nhằm lưu trữ thông tin.
- Cập nhật dữ liệu nhanh chóng hiệu quả,nhanh chóng và kịp thời.
- + Thêm, Xóa, Sửa món trong thực đơn
- Thống kê được số lượng hàng hóa nhập xuất, thu chi, và tình hình doanh thu của cửa hàng.
- Quản lý nhân viên
- In hóa đơn cần thiết như : Phiếu yêu cầu nhập hàng , Phiếu thu, Phiếu chi.
- Báo cáo,Tính tiền,Lưu trữ thông tin lâu dài, đảm bảo truy vấn nhanh khi cần thiết

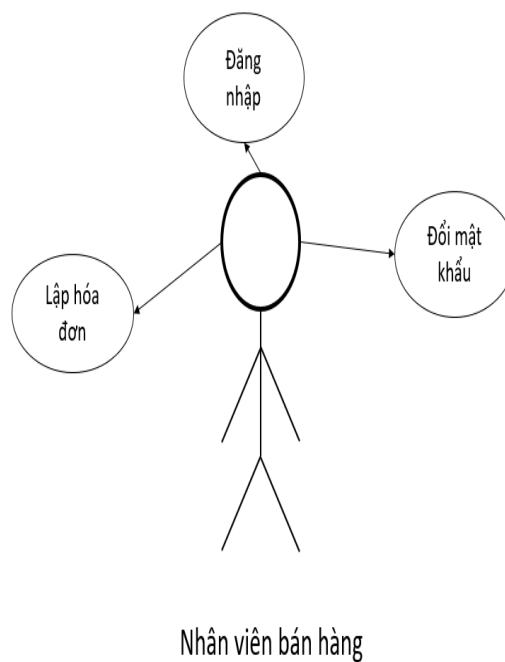
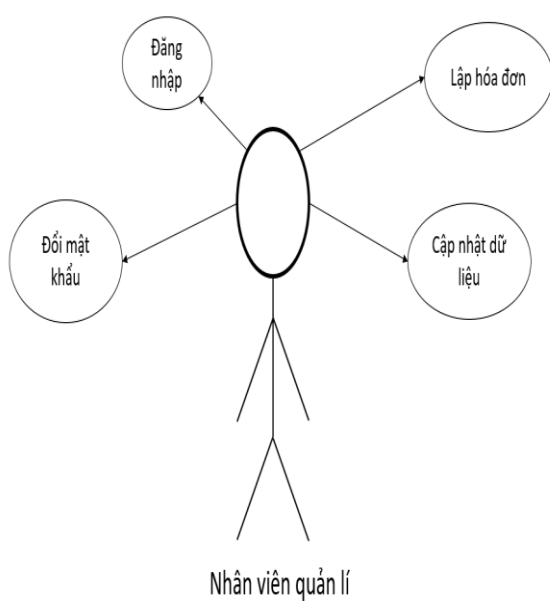
### III. Phân Tích Thiết Kế Hệ Thống

#### 1. Xác Định Tác Nhân

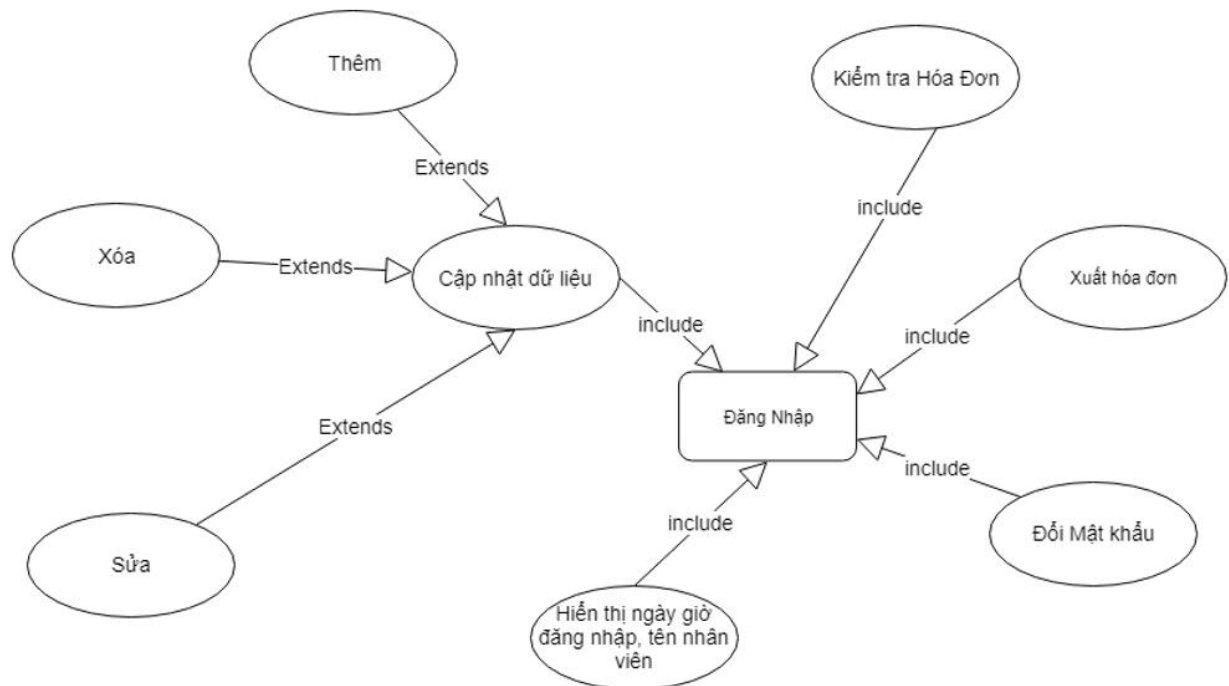
##### 3.1 Khách Hàng

Không có tác động đến phần mềm.

##### 3.2 Nhân Viên



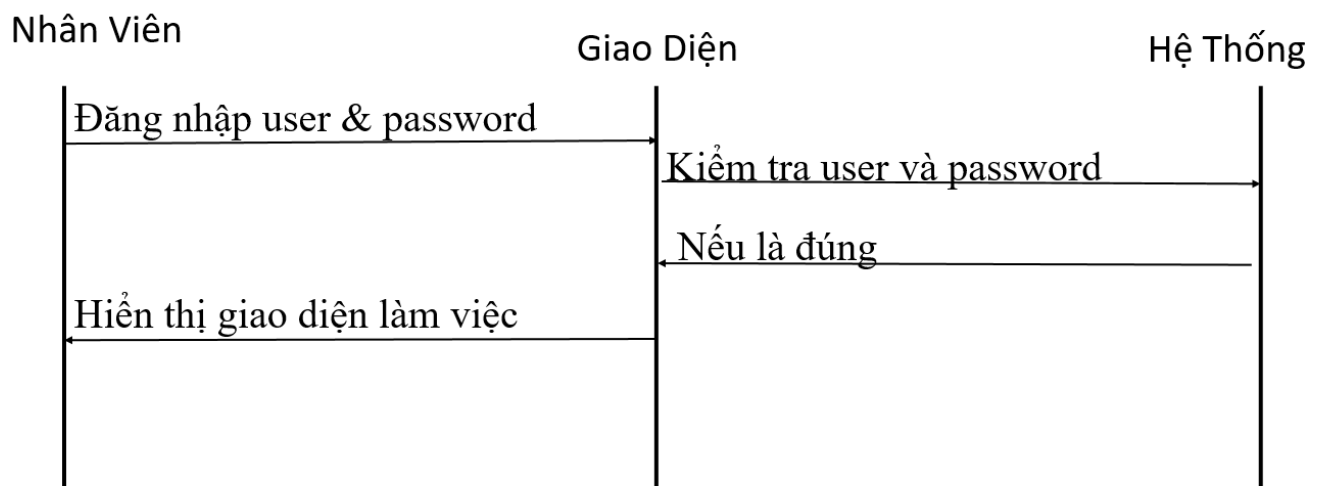
## 2. Use Case Tổng Quát



## 3. Mô Tả Use Case

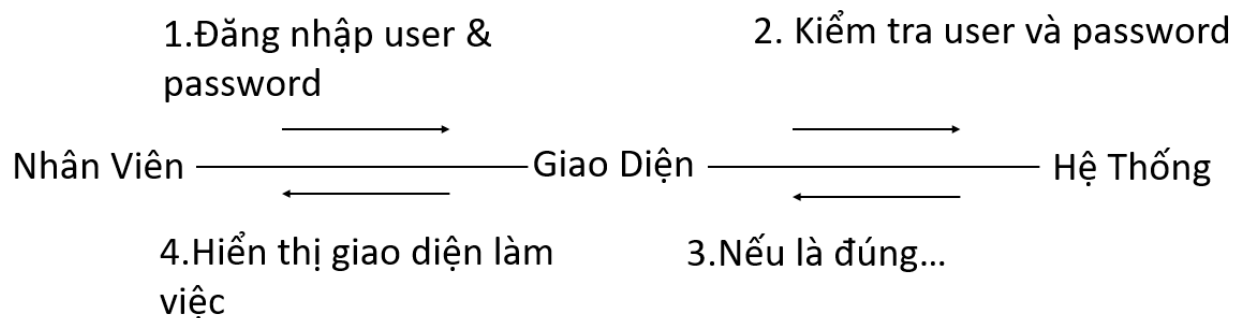
### 3.1 Mô tả Use Case “Đăng nhập” cho hai nhân viên

#### Sơ Đồ Tuần Tự

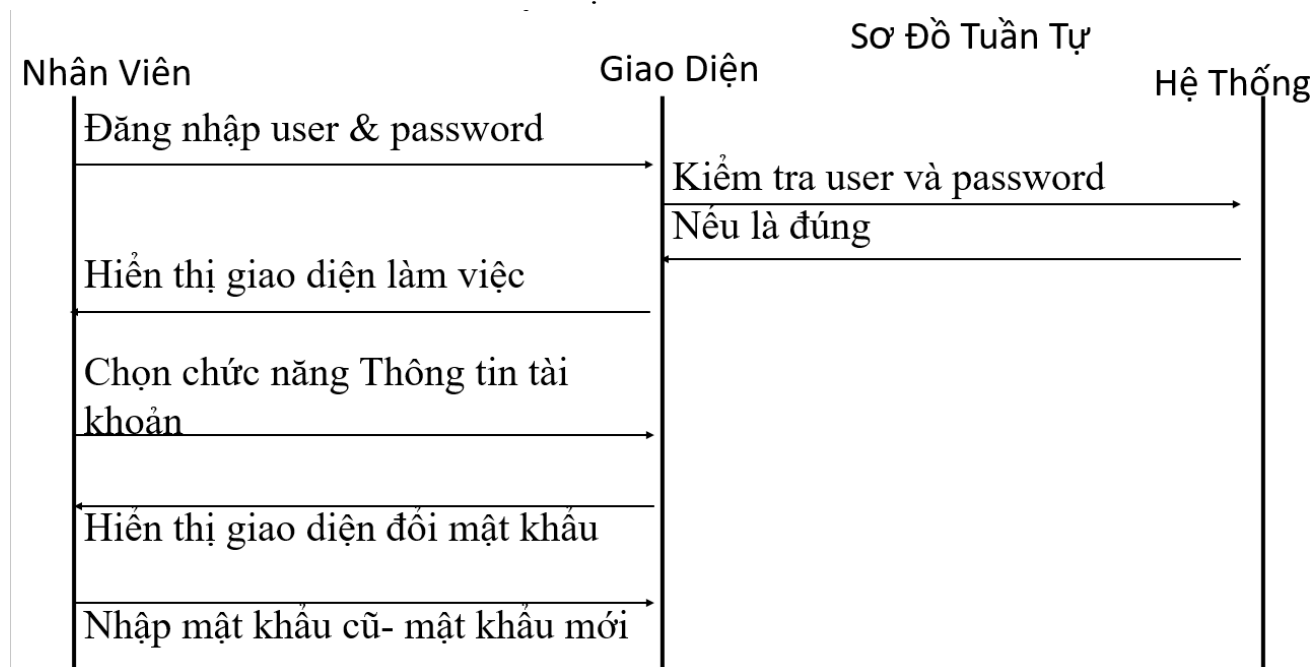




\*Sơ Đồ Hợp Tác:

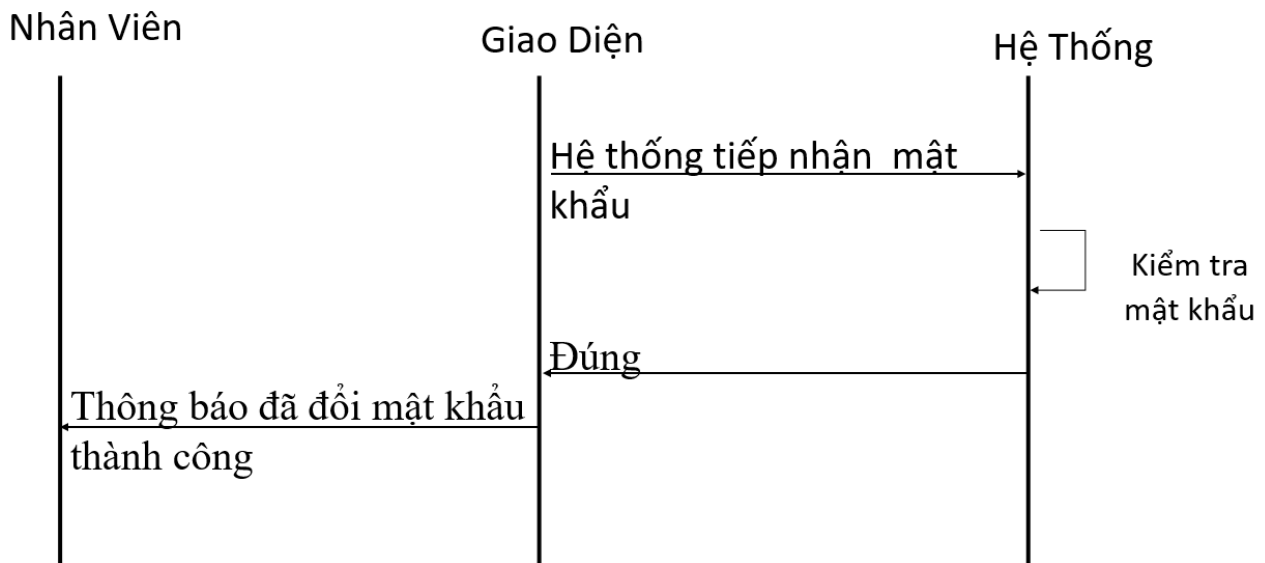


### 3.2 Mô tả Use Case “Đổi mật khẩu”



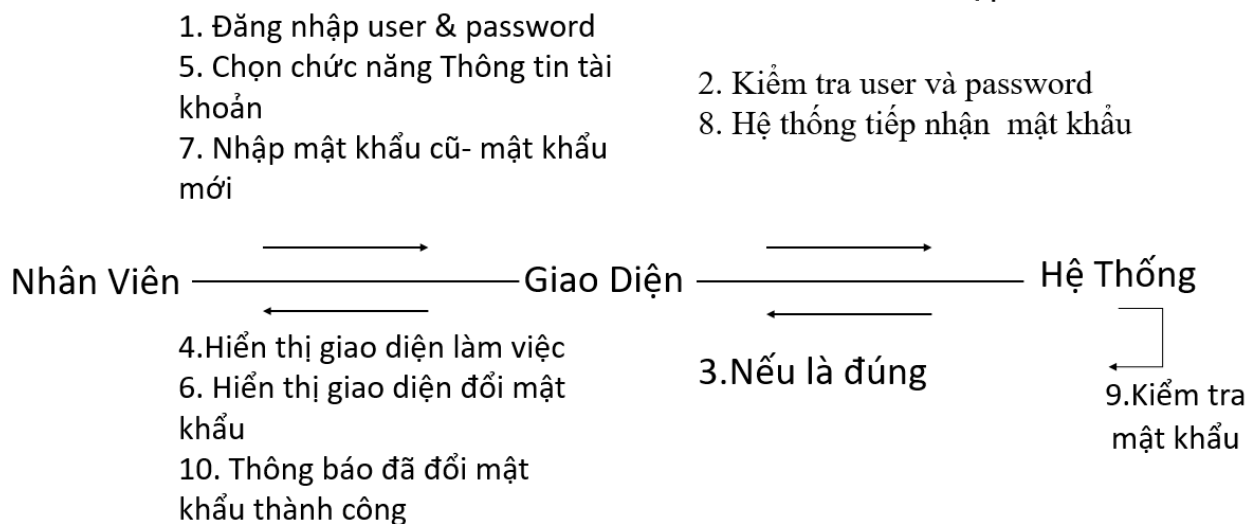
## Mô Tả Quá Trình Đổi mật khẩu

### Sơ Đồ Tuần Tự



## Mô Tả Quá Trình Đổi mật khẩu

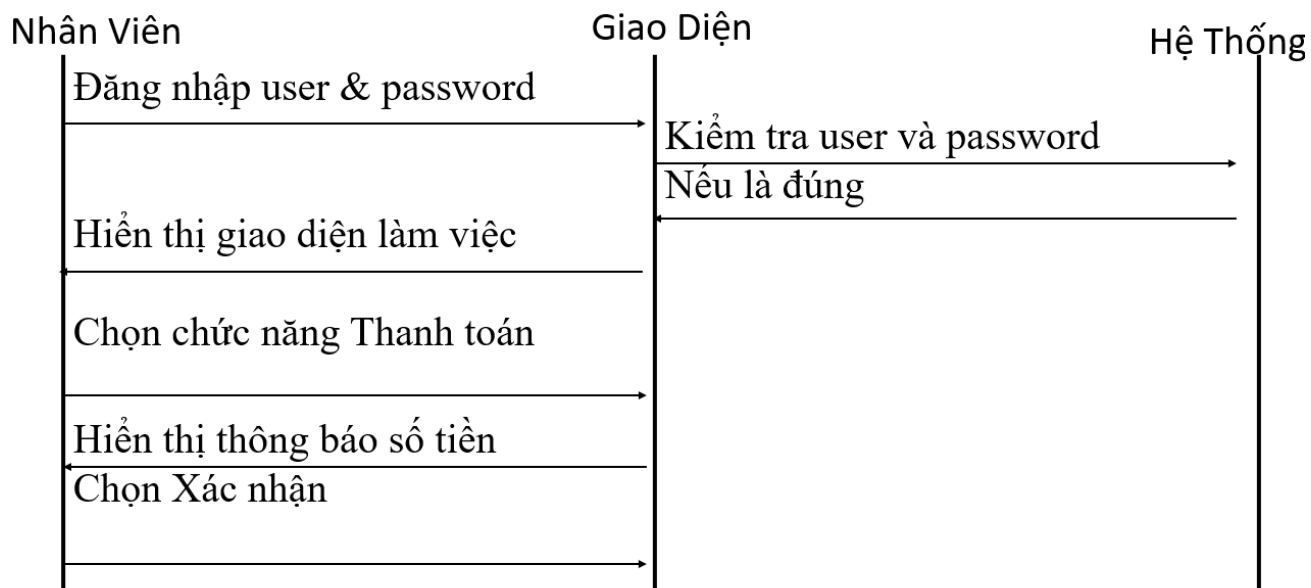
### Sơ Đồ Hợp Tác



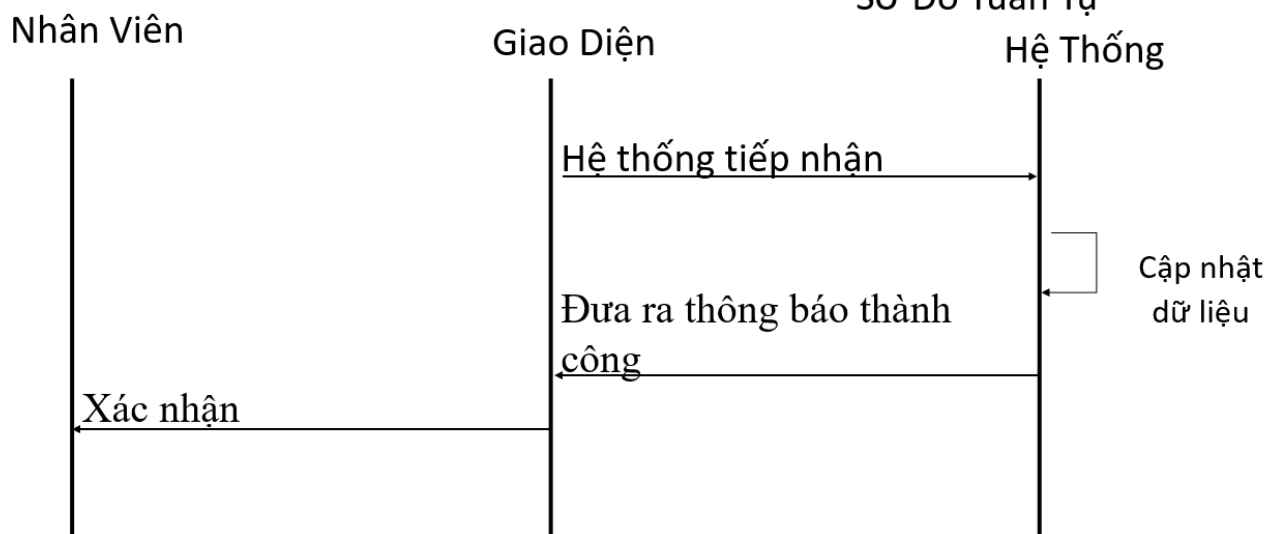
## 3.3 Mô tả Use Case “Lập hóa đơn”

**Mô Tả Quá Trình Lập hóa đơn**

Sơ Đồ Tuần Tự

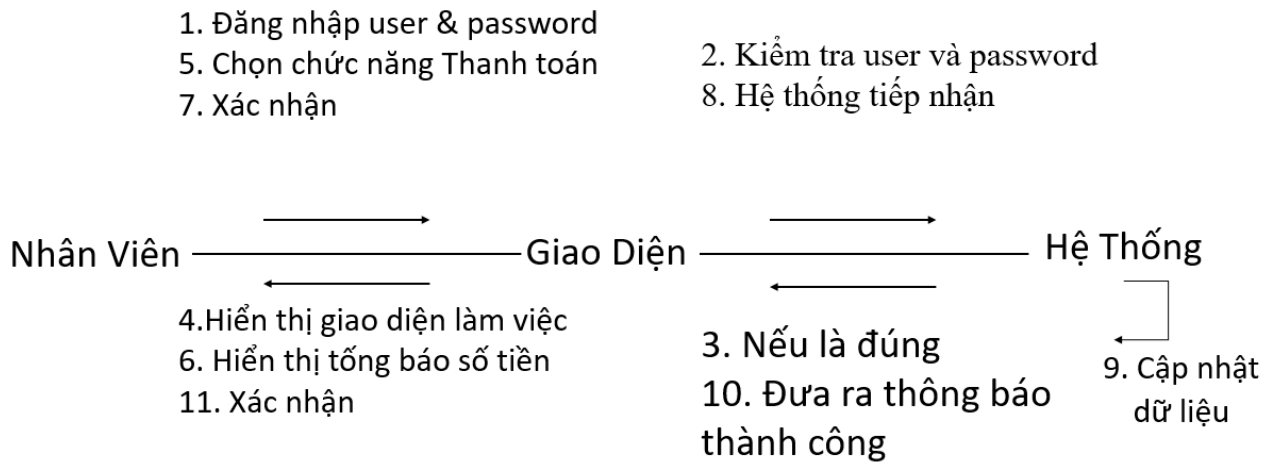
**Mô Tả Quá Trình Lập hóa đơn**

Sơ Đồ Tuần Tự



## Mô Tả Quá Trình Lập hóa đơn

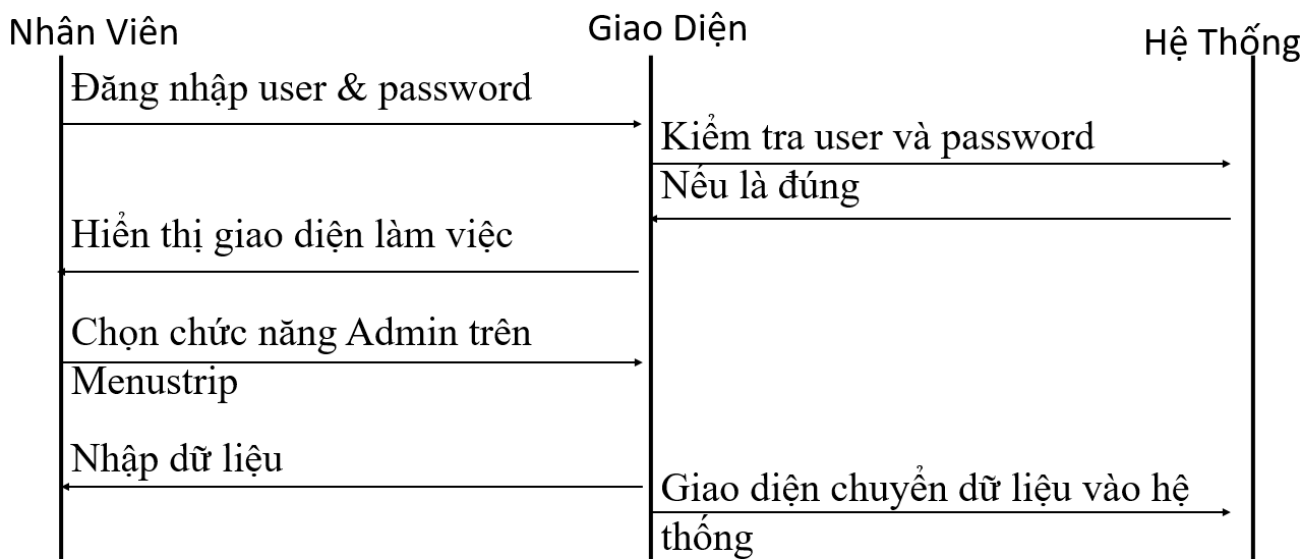
Sơ Đồ Hợp Tác:



### 3.4 Mô tả Use Case “Cập nhật dữ liệu”

## Mô Tả Quá Trình Cập nhật dữ liệu

Sơ Đồ Tuần Tự



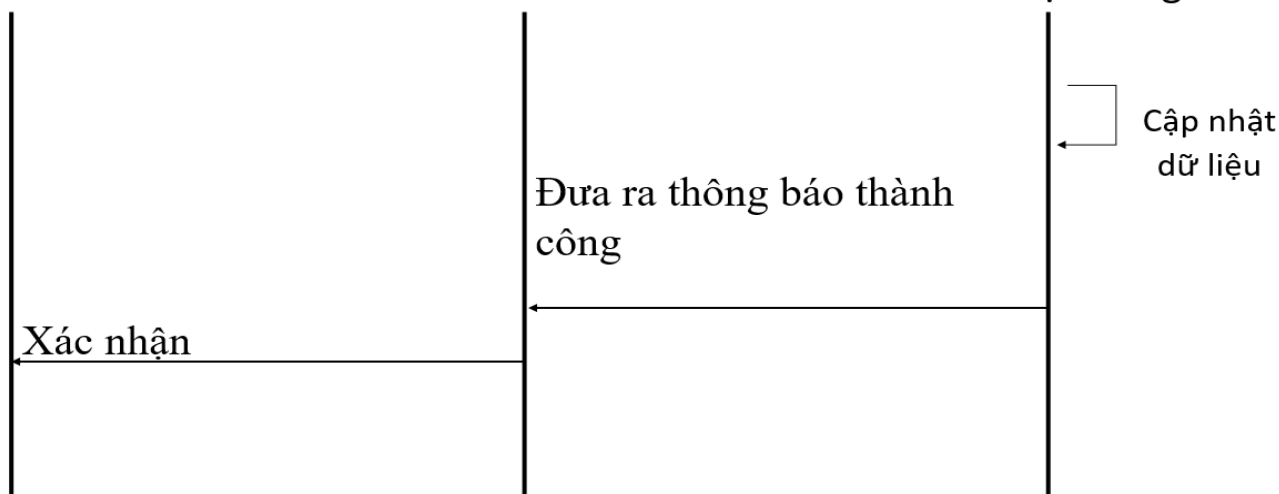
## Mô Tả Quá Trình Cập nhật dữ liệu

Nhân Viên

Giao Diện

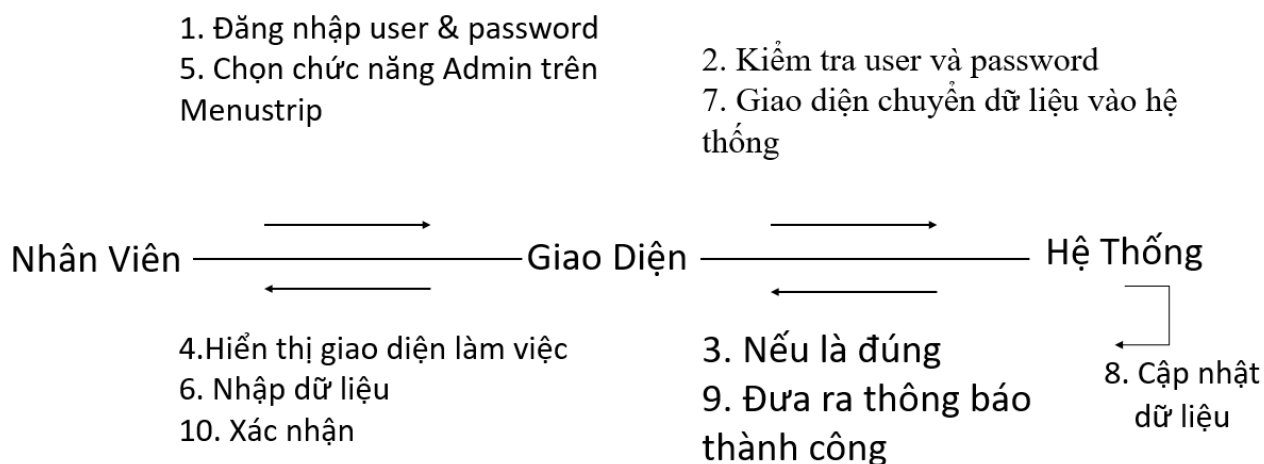
Sơ Đồ Tuần Tự

Hệ Thống



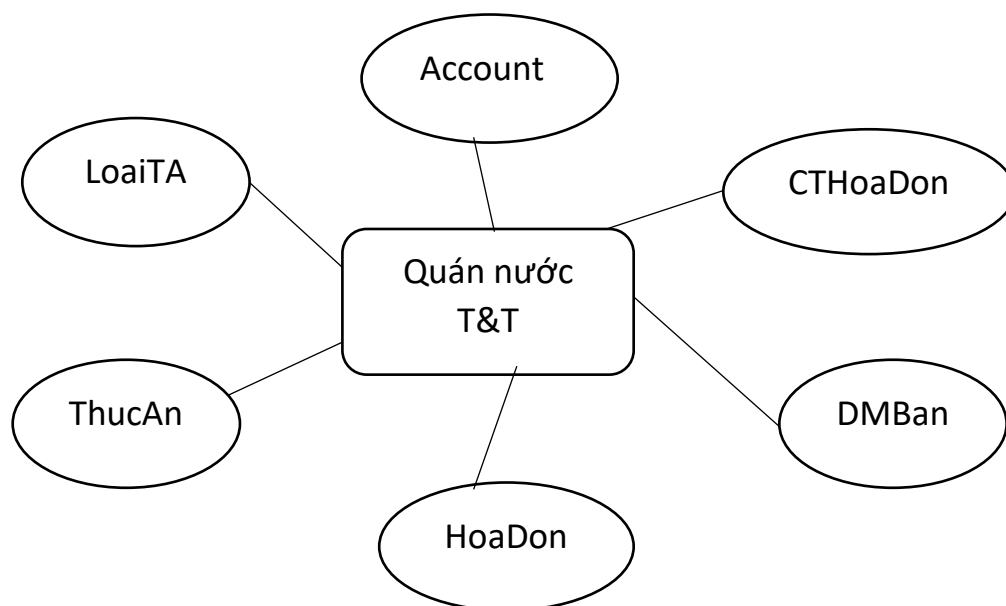
## Mô Tả Quá Trình Cập nhật dữ liệu

Sơ Đồ Hợp Tác



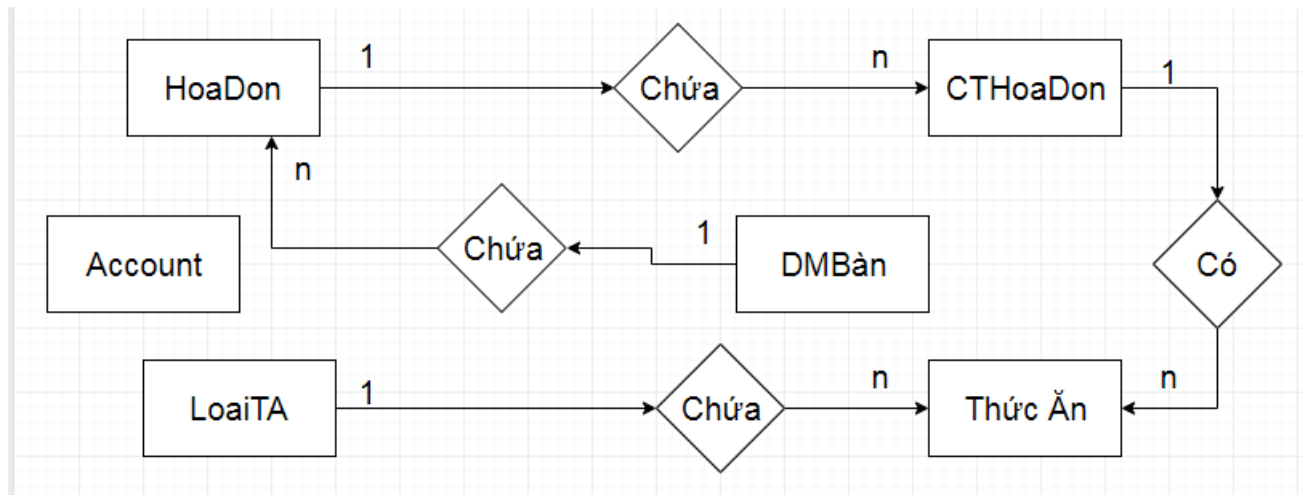
## IV. Cơ Sở Dữ Liệu

### 1. Các Thực Thể Và Các Thuộc Tính

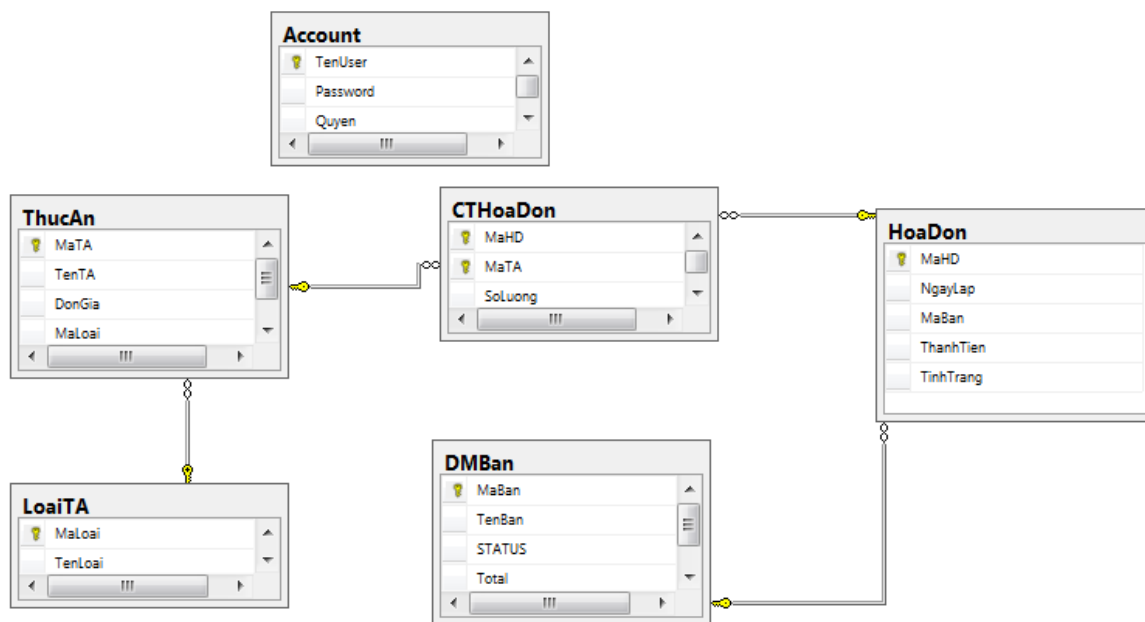


Thực thể	STT	Thuộc tính	Kiểu dữ liệu	Diễn Giải
Account	1	TenUser	varchar(30)	Tên tài khoản
	2	Password	varchar(30)	Mật khẩu
	3	Quyen	varchar(15)	Quyền truy cập
CTHoaDon	1	MaHD	int	Mã hóa đơn
	2	MaTA	int	Mã thức ăn
	3	SoLuong	int	Số lượng
DMBan	1	MaBan	int	Mã bàn
	2	TenBan	nvarchar(30)	Tên bàn
	3	STATUS	nvarchar(15)	Trạng thái
	4	Total	int	Tổng
HoaDon	1	MaHD	int	Mã hóa đơn
	2	NgayLap	date	Ngày lập
	3	MaBan	int	Mã bàn
	4	ThanhTien	int	Thành tiền
	5	TinhTrang	int	Tình trạng
LoaiTA	1	MaLoai	int	Mã loại thức ăn
	2	TenLoai	nvarchar(50)	Tên loại thức ăn
ThucAn	1	MaTA	int	Mã thức ăn
	2	TenTA	nvarchar(50)	Tên thức ăn
	3	DonGia	int	Đơn giá
	4	MaLoai	int	Mã loại thức ăn

## 2. Sơ Đồ ER



## 3. Sơ Đồ Quan Hệ



## 4. Xây Dựng và Cập Nhật Các Ràng Buộc Toàn Vẹn

### 4.1 Ràng buộc của những thuộc tính khóa

- RB1: Mỗi tài khoản khi được tạo có một User name riêng biệt, không trùng với bất kỳ tài khoản nào.
- RB2: Mỗi hóa đơn khi được tạo phải có mã hóa đơn riêng biệt, không trùng lặp.
- RB3: Mỗi bàn được tạo ra phải có mã bàn riêng biệt, không trùng lặp.
- RB4: Mỗi thức ăn được tạo ra phải có mã thức ăn riêng biệt, không trùng lặp.
- RB5: Mỗi loại thức ăn được tạo ra phải có mã loại thức ăn riêng biệt, không trùng lặp.
- RB6: Mỗi chi tiết hóa đơn được tạo ra phải có chi tiết riêng biệt, không trùng lặp.

- Bảng tầm ảnh hưởng

Ràng Buộc	Bối cảnh	Thêm	Sửa	Xóa
RB1	T <sub>Accout</sub>	+	+	-
RB2	T <sub>HoaDon</sub>	+	+	+
RB3	T <sub>DMBan</sub>	+	+	+
RB4	T <sub>ThucAn</sub>	+	+	-
RB5	T <sub>LoaiTA</sub>	+	+	+
RB6	T <sub>CTHoaDon</sub>	+	+	-

#### 4.2 Ràng buộc giữa các bảng thực thể

RB7: Khi nhập thông tin thức ăn, mã loại phải nằm trong bảng Loại Thức Ăn.

RB8: Khi nhập thông tin chi tiết hóa đơn, mã thức ăn phải nằm trong bảng Thức Ăn.

RB9: Khi nhập thông tin hóa đơn, mã bàn phải nằm trong bảng DM Bàn.

RB10: Khi nhập thông tin chi tiết hóa đơn, mã hóa đơn phải nằm trong bảng Hóa Đơn.

Ràng buộc	Bối cảnh	Thêm	Sửa	Xóa
RB7	T <sub>ThucAn</sub>	+	+	-
	T <sub>LoaiTA</sub>	+	-(*)	+
RB8	T <sub>CTHoaDon</sub>	+	+	-
	T <sub>ThucAn</sub>	+	-(*)	-
RB9	T <sub>HoaDon</sub>	+	+	-
	T <sub>DMBan</sub>	-	-(*)	-
RB10	T <sub>CTHoaDon</sub>	+	+	-
	T <sub>HoaDon</sub>	-	-(*)	+

#### 4.3. Cập nhật ràng buộc

\*RB1:

ALTER TABLE Account ADD PRIMARY KEY (TenUser)

\*RB2:

ALTER TABLE HoaDon ADD PRIMARY KEY (MaHD)

\*RB3:

ALTER TABLE DMBan ADD PRIMARY KEY (MaBan)

\*RB4:

ALTER TABLE ThucAn ADD PRIMARY KEY (MaTA)

\*RB5:

ALTER TABLE LoaiTA ADD PRIMARY KEY (MaLoai)

\*RB6:

ALTER TABLE CTHoaDon ADD PRIMARY KEY (MaHD, MaTA)

\*RB7:

ALTER TABLE dbo.ThucAn ADD CONSTRAINT FK\_ThucAn\_LoaiTA  
FOREIGN KEY(MaLoai) REFERENCES dbo.LoaiTA(MaLoai)

\*RB8

ALTER TABLE dbo.CTHoaDon ADD CONSTRAINT FK\_CTHoaDon\_ThucAn



```
FOREIGN KEY(MaTA) REFERENCES dbo.ThucAn(MaTA)
```

```
*RB9
```

```
ALTER TABLE dbo.HoaDon ADD CONSTRAINT FK_HoaDon_DMBan
```

```
FOREIGN KEY(MaBan) REFERENCES dbo.DMBan(MaBan)
```

```
*RB10
```

```
ALTER TABLE dbo.CTHoaDon ADD CONSTRAINT FK_CTHoaDon_HoaDon
```

```
FOREIGN KEY(MaHD) REFERENCES dbo.HoaDon(MaHD)
```

## V. Thiết Kế Form

### Form Đăng Nhập

-Name: frmLogIn

- Chức năng : Form đăng nhập cho phép người dùng đăng nhập vào hệ thống dưới hai quyền Admin and Staff.

#### Lable

STT	Name	Thiết lập các thuộc tính
1	Lable1	Text = “ Đăng nhập hệ thống ”
2	Lable2	Text = “ Tên User ”
3	Lable3	Text = “ Mật khẩu ”

#### TextBox

STT	Name	Thiết lập các thuộc tính
3	txtTenDangNhap	Text = “ admin ”
4	txtMatKhau	UseSystempasswordchar= True

#### Button

STT	Name	Thiết lập các thuộc tính
6	btnLogIn	Text = “ Đăng nhập ”
7	btnExit	Text = “ Thoát ”

### CheckBox

STT	Name	Thiết lập các thuộc tính
8	ChkShow_Hide_password	Text = “ Hiện mật khẩu ”

### Form Chính

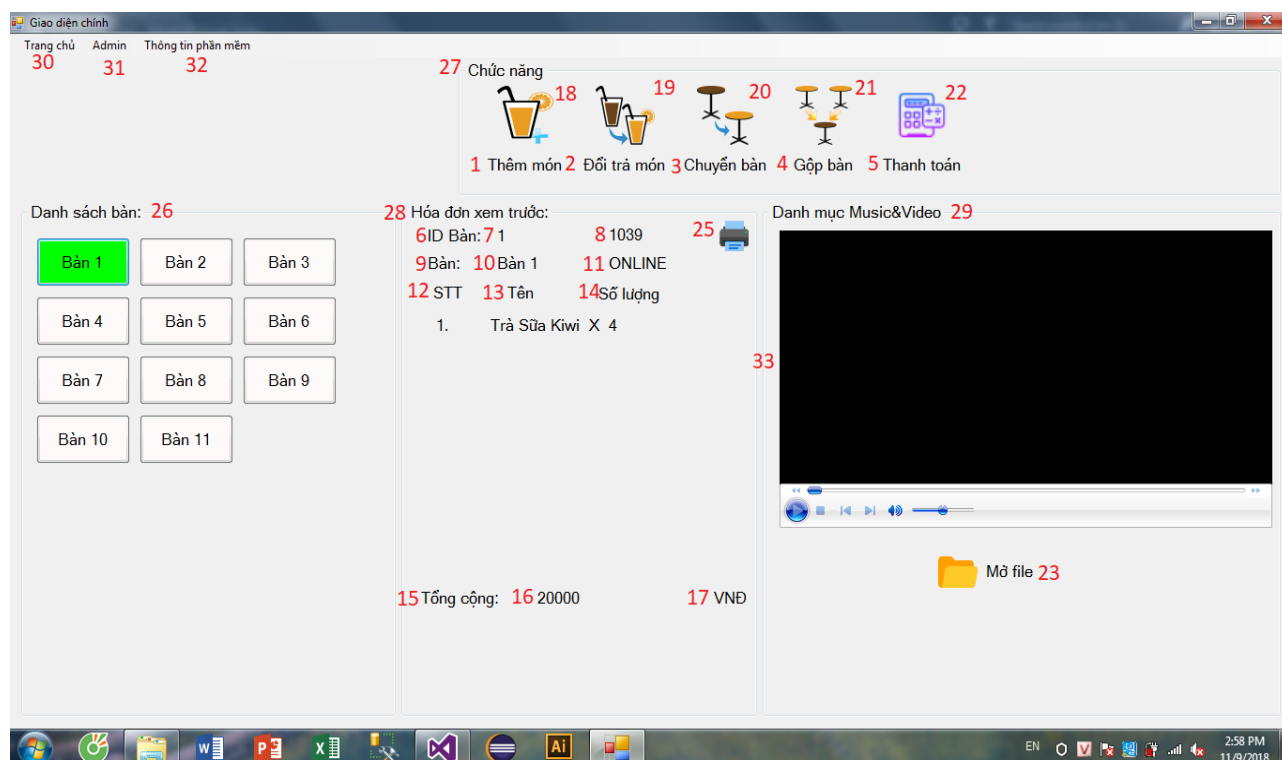
-Name: frmMain

Chức năng : Form chính cho phép người dùng sử dụng giao diện chính bao gồm :

- Danh sách bàn
- Chức năng : + Thêm món
  - + Đổi trả món
  - + Chuyển bàn
  - + Gộp bàn
  - + Thanh toán
- Hóa đơn xem trước , thanh toán
- Danh mục Music&Video

Ngoài ra người dùng với quyền Admin có thể thêm và sửa các loại sản phẩm và cuối cùng là xem về thông tin của

phần mềm



**Lable**

STT	Name	Thiết lập các thuộc tính
1	Lable 1	Text =" Thêm món"
2	Lable 2	Text = "Đổi trả món"
3	Lable 3	Text = "Chuyển bàn"
4	Lable 4	Text = "Gộp bàn "
5	Lable 5	Text = "Thanh toán"
6	Lable 6	Text = " ID Bàn "
7	Lable 7	
8	Lable 8	
9	Lable 9	Text = "Bàn "
10	Lable 10	Text = (Tên bàn)
11	Lable 11	Text = (Trạng Thái)
12	Lable 12	Text = "STT "
13	Lable 13	Text = " Tên "
14	Lable 14	Text = "Số lượng "
15	Lable 15	Text = "Tổng cộng "
16	Lable 16	Text = (Số Tiền)
17	Lable 17	Text = "VND "

**Button**

STT	Name	Thiết lập các thuộc tính
18	btnAddFood	
19	btnReturnFood	
20	btnChangeTable	
21	btnLumpedTable	
22	btnPay	
23	btnOpenFile	Text = "Mở file "
25	btnPrint	

**Groupbox**

STT	Name	Thiết lập các thuộc tính
26	groupBox1	Text = "Danh sách bàn "
27	groupBox2	Text = "Chức năng "
28	groupBox3	Text = "Hóa đơn xem trước "
29	groupBox4	Text = "Danh mục Music&&Video"

**ToolStripMenuItem**

STT	Name	Thiết lập các thuộc tính
30	homeToolStripMenuItem	Text = "Trang chủ "
31	MnuIAdmin	Text = "Admin "
32	MnuISoftwareInfo	Text = "Thông tin phần mềm"

**AxWindowsMediaPlayer**

STT	Name	Thiết lập các thuộc tính
33	axWindowsMediaPlayer1	

## Form thông tin phần mềm

-Name: frmSoftwareInfo



### Lable

STT	Name	Thiết lập các thuộc tính
1	lblTopic	Text = “Thông Tin Phần Mềm ”
2	lable1	Text = “Version 1.0.0”
3	Lable2	Text = “Người thực hiện ”
4	Lable3	Text = “Nguyễn Huỳnh Hải Phương”
5	lblAdmin1	Text = “Nguyễn Thị Minh Thảo”
6	lblAdmin2	Text = “Lê Thành Trung ”

## Form Thêm món

-Name: frmAddFood

Chức năng: Cho phép người dùng thêm món và tùy chỉnh số lượng trên bàn

**Thêm Món**

1 Thêm Món

2 Mã bàn: 4 10

3 Tên bàn: Bàn 4 11

4 Trạng thái: Trống 12

5 DM Loại: Cà Phê 13

6 Tên món: Cà Phê Đen 14

7 Số lượng: 1 15

8 Đơn giá: 2000 16

9 Thành tiền: 17

Add 18

### Lable

STT	Name	Thiết lập các thuộc tính
1	Lable 1	Text="Thêm món"
2	Lable 9	Text="Mã bàn "
3	Lable 2	Text="Tên bàn"
4	Lable 4	Text="Trạng thái"
5	Lable 6	Text="DM loại"
6	Lable 8	Text="Tên món "
7	Lable 7	Text="Số lượng "
8	Lable 3	Text="Đơn giá "
9	Lable 5	Text="Thành tiền "
10	lblIDTable	
11	lblNameTable	
12	lblStatus	

### ComboBox

STT	Name	Thiết lập các thuộc tính
13	cboCategoryFood	
14	cboNameFood	

**NumericUpDown**

STT	Name	Thiết lập các thuộc tính
15	numQuantity	

**TextBox**

STT	Name	Thiết lập các thuộc tính
16	txtPrice	
17	txtTotal	

**Button**

STT	Name	Thiết lập các thuộc tính
18	btnAddFood	Text="Add"

**Form Đổi trả món**

-Name: frmReturnFood

- Chức năng : Cho phép người dùng đổi trả lại món ăn đã được chọn lựa trước trên bàn.

Lưu ý: Món ăn chỉ được đổi trả trước khi món ăn được chế biến .

**1 Đổi Trả Món**

2 Mã hóa đơn: 29 9

3 ID bàn: 1 10

4 Tên bàn: Bàn 1 11

5 ID món: 1 12

6 Tên món: Cà Phê Đen 13

7 Số lượng hiện tại: 3 14

8 Số lượng trả lại: 0 15

Xác nhận đổi 16

### Lable

STT	Name	Thiết lập các thuộc tính
1	label1	Text=" Đổi Trả Món "
2	label3	Text=" Mã hóa đơn "
3	label2	Text=" ID bàn"
4	label4	Text=" Tên bàn"
5	Lable6	Text=" ID món"
6	label7	Text=" Tên món"
7	label8	Text=" Số lượng hiện tại"
8	label9	Text=" Số lượng trả lại "
9	lblIDBill	
10	lblIDTable	
11	lblNameTable	
12	lblIDFood	

### ComboBox

STT	Name	Thiết lập các thuộc tính
13	cboNameFood	

### NumericUpDown

STT	Name	Thiết lập các thuộc tính
14	numQuantity	
15	numReturnQuantity	

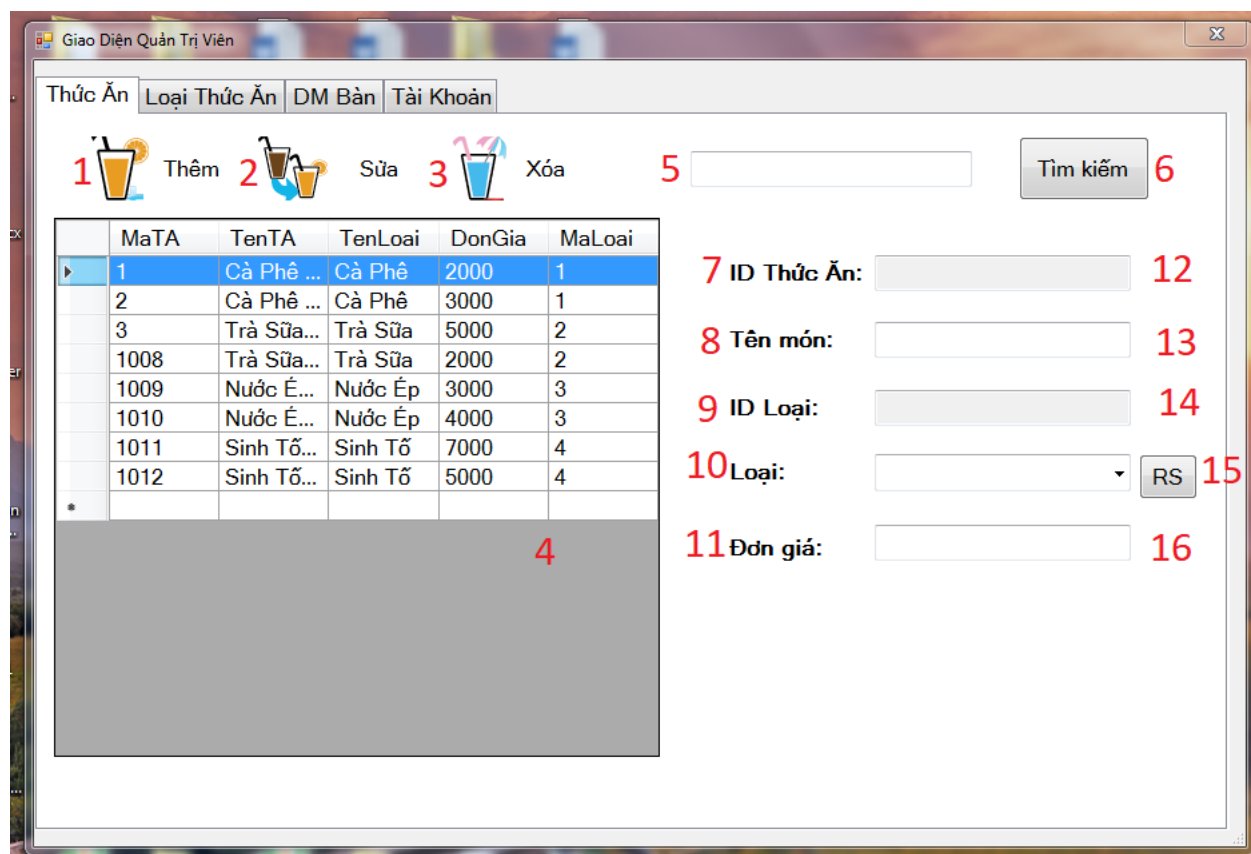
### Button

STT	Name	Thiết lập các thuộc tính
16	btnOkey	Text="Xác nhận đổi "

## Form giao diện quản trị viên

### 1.Tab Thức ăn

Cho phép người dùng xem, tìm kiếm thức ăn trong Menu qua kí tự chữ ,có thể thêm ,xóa, sửa món ăn trong menu qua ID thức ăn , Tên món, ID loại ,Loại ,đơn giá



### Button

STT	Name	Thiết lập các thuộc tính
1	btnAddFood	Text="Thêm"
2	btnChangeFood	Text="Sửa"
3	btnDeleteFood	Text="Xóa"
6	btnSearch	Text="Tìm kiếm "
15.1	btnResetCategory	Text="RS"

### DataGridView

STT	Name	Thiết lập các thuộc tính
4	dtgvFood	

### Textbox

STT	Name	Thiết lập các thuộc tính
5	txtContentSearch	
12	txtIDFood	
13	txtNameFood	
14	txtCategoryFood	
16	txtPrice	

### Lable

STT	Name	Thiết lập các thuộc tính
7	Lable1	Text=" ID Thức Ăn"
8	Lable2	Text="Tên món"
9	Lable13	Text=" ID Loại"
10	Lable3	Text=" Loại"
11	Lable4	Text=" Đơn giá "

### Combobox

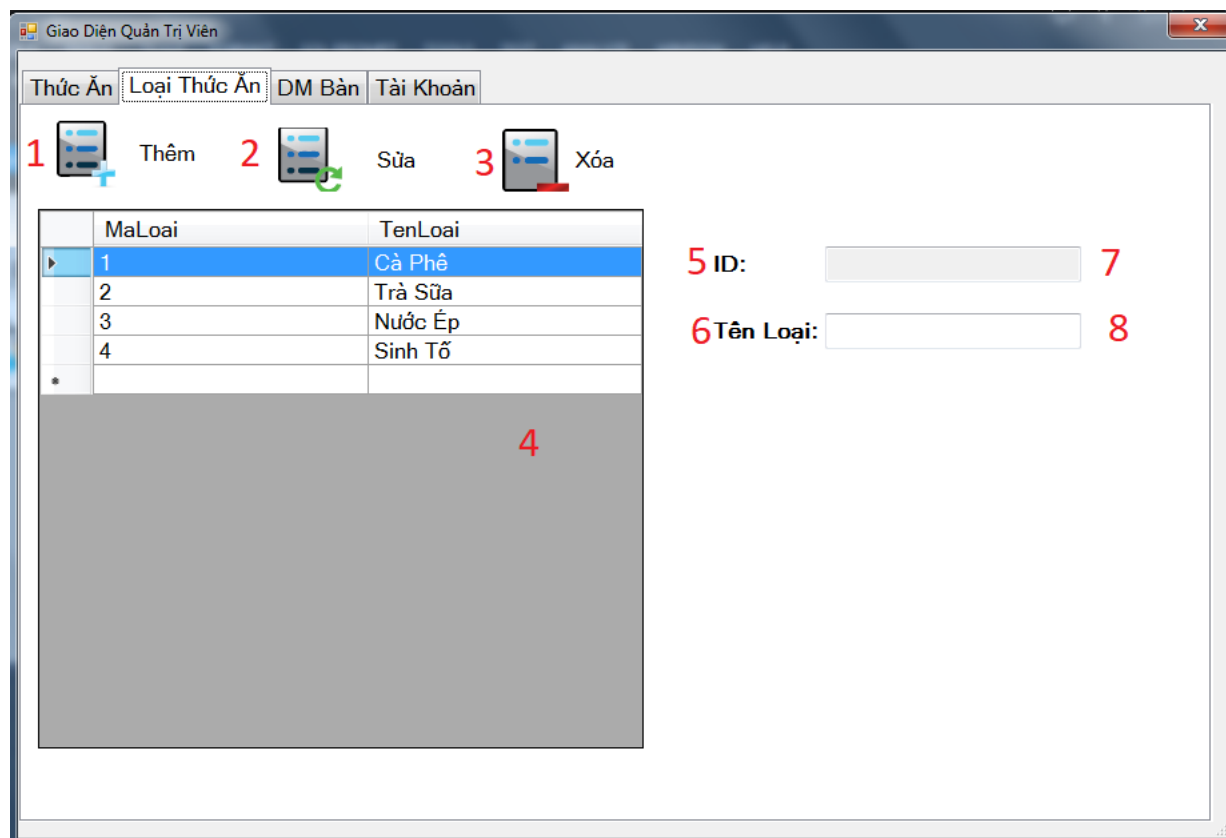
STT	Name	Thiết lập các thuộc tính
-----	------	--------------------------



15.2	cboCategoryFood	
------	-----------------	--

## 2.Tab Loại thức ăn

Cho phép người dùng xem , thêm ,xóa , sửa loại thức ăn qua ID và Tên loại



### A. Button

STT	Name	Thiết lập các thuộc tính
1	btnAddCategory	Text="Thêm"
2	btnChangeCategory	Text="Sửa"
3	btnDeleteCategory	Text="Xóa"

### B. DataGridView

STT	Name	Thiết lập các thuộc tính
4	dtgvCategory	

### C. Lable

STT	Name	Thiết lập các thuộc tính
5	label8	Text="ID"
6	label7	Text="Tên Loại"

### D. Textbox

STT	Name	Thiết lập các thuộc tính
7	txtIDCategory	
8	txtNameCategory	

## 3.Tab DM Bàn

Cho phép người dùng xem mã bàn , tên bàn , trạng thái và tổng tiền .Cho phép thêm, sửa, xóa bàn

### A. Button

STT	Name	Thiết lập các thuộc tính
1	btnAddTable	Text="Thêm"
2	btnChangeTable	Text="Sửa"
3	btnDeleteTable	Text="Xóa"

### B. DataGridView

STT	Name	Thiết lập các thuộc tính
4	dtgvTable	

### C. Lable

STT	Name	Thiết lập các thuộc tính
5	Label6	Text="ID"
6	Label5	Text="Tên Bàn"
7	Label11	Text="Trạng thái"
8	Label12	Text="Tổng tiền "

### D. Textbox

STT	Name	Thiết lập các thuộc tính
9	txtIDTable	
10	txtNameTable	
12	txtTotalTable	

**E. ComboBox**

STT	Name	Thiết lập các thuộc tính
11	cboStatusTable	

**4.Tab Tài khoản**

Cho phép người dùng xem Tên user và quyền .Cho phép thêm, sửa, xóa qua tên tk và quyền

**A. Button**

STT	Name	Thiết lập các thuộc tính
1	btnAddAccount	Text="Thêm"
2	btnChangeAccount	Text="Sửa"
3	btnDeleteAccount	Text="Xóa"

**B. DataGridView**

STT	Name	Thiết lập các thuộc tính
4	dtgvAccount	

**C. Lable**

STT	Name	Thiết lập các thuộc tính
5	Label10	Text="Tên TK"
6	Label9	Text="Quyền "

**D. Textbox**

STT	Name	Thiết lập các thuộc tính
7	txtUserName	
8	txtAccountType	

## 5.Tab chi tiết hóa đơn

Cho phép người dùng xem mã hóa đơn ,mã thức ăn , số lượng .

### A. Data GridView

STT	Name	Thiết lập các thuộc tính
1	dtgvbillinfo	

### B. Lable

STT	Name	Thiết lập các thuộc tính
2	Lable 1	Text="Mã hóa đơn"
3	Lable 2	Text="Mã thức ăn "
4	Lable 3	Text="Số lượng"

### C. TextBox

STT	Name	Thiết lập các thuộc tính
5	txtIDBillInfo	
6	txtIDFood	

### D. NumericUpDown

STT	Name	Thiết lập các thuộc tính
7	numBillInfo	

## 6.Tab hóa đơn

Cho phép người dùng xem Mã hóa đơn , ngày lập , mã bàn ,thành tiền và Tình trạng

### A. Data GridView

STT	Name	Thiết lập các thuộc tính
1	dtgvbill	

### B. Lable

STT	Name	Thiết lập các thuộc tính
2	Lable 1	Text="Mã hóa đơn"
3	Lable 2	Text="Ngày lập "
4	Lable 3	Text="Mã bàn"
5	Lable 4	Text="Thành tiền"
6	Lable 5	Text="Tình trạng"

### C. TextBox

STT	Name	Thiết lập các thuộc tính
7	txtIDBill	
9	txtIDTable	
10		

### D. NumericUpDown

STT	Name	Thiết lập các thuộc tính
11	numBillStatus	

### E. ComboBox

STT	Name	Thiết lập các thuộc tính
8	cmbBillday	

**Form Thông tin tài khoản**

-Name: frmInfoUser

-Chức năng: Xem thông tin user đang đăng nhập và đổi mật khẩu.

**A> Lable**

STT	Name	Thiết lập các thuộc tính
1	Lable 1	Text="Thông Tin Tài Khoản"
2	Lable 2	Text="Tên User:"
3	Lable 3	Text="Mật khẩu:"
4	Lable 4	Text="Quyền:"
5	Lable 5	Text="Mật khẩu cũ:"
6	Lable 6	Text="Mật khẩu mới:"
7	Lable 7	Text="Xác nhận mật khẩu:"

**B> TextBox**

STT	Name	Thiết lập các thuộc tính
8	txtUserName	
9	txtPassWord	UseSystemPasswordChar=true
10	txtRight	ReadOnly=true
11	txtPassWordOld	UseSystemPasswordChar=true
12	txtPassWordNew	UseSystemPasswordChar=true
13	txtConfigPassWord	UseSystemPasswordChar=true

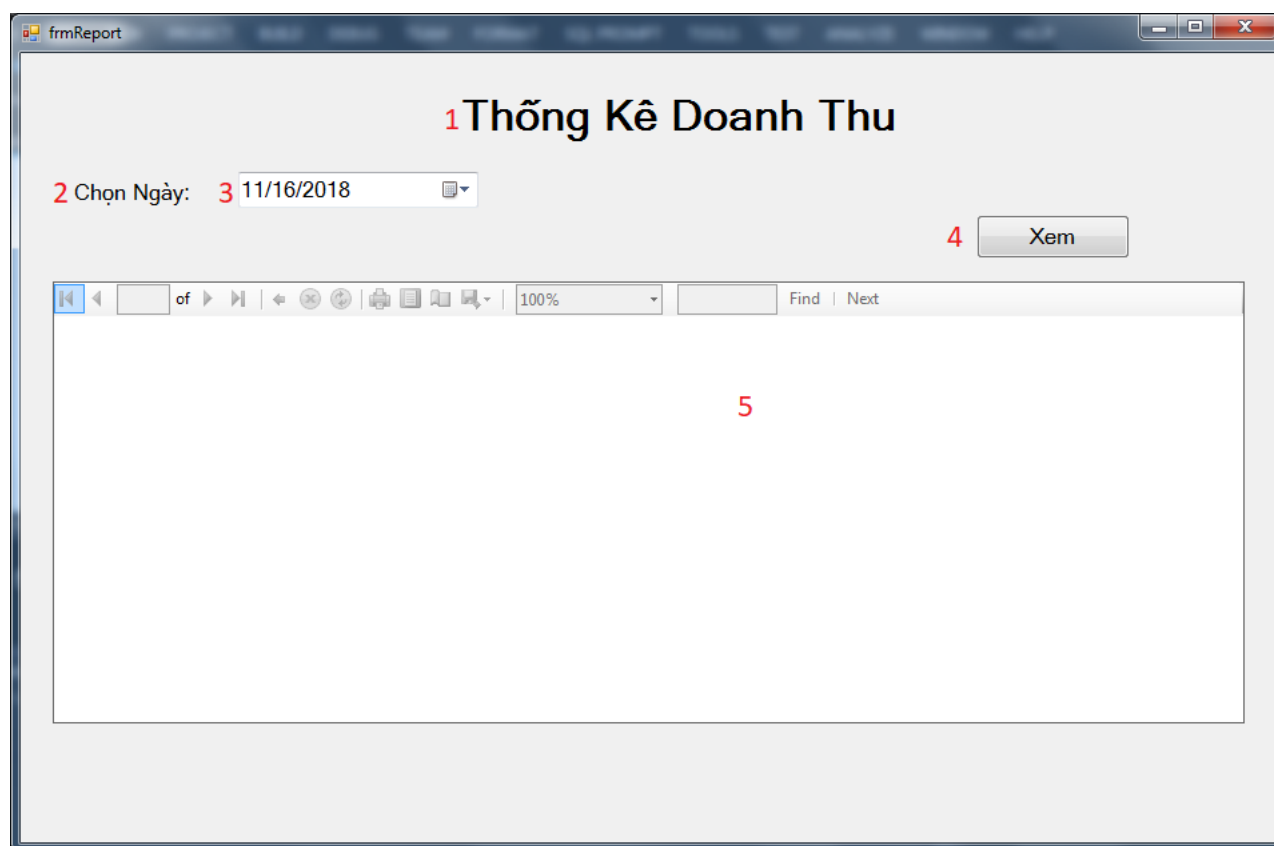
**C> Buttons**

STT	Name	Thiết lập các thuộc tính
14	btnChangePassword	Text="Đổi mật khẩu"
15	btnExit	Text="Thoát"
16	btnShowPassWord	Text=Null

**Form Báo Cáo(Thống kê)**

Name: frmReport

Chức năng: Cho phép người dùng có thể tạo Báo cáo



### Lable

STT	Name	Thiết lập các thuộc tính
1	Lable 1	Text="Thống kê doanh thu"
2	Lable 2	Text="Chọn ngày"

### DateTimePicker

STT	Name	Thiết lập các thuộc tính
3	dtphchooseDay	

### Button

STT	Name	Thiết lập các thuộc tính
4	Btnview	Text="Xem "

### ReportViewer

STT	Name	Thiết lập các thuộc tính
5	rpvReport	

## VI. Xử lý Form

### 1. Xây dựng Class hỗ trợ (KetNoi)

-Là cầu nối giữa Sql hay cơ sở dữ liệu với các form

-Phương thức kết nối SQL với các form-xây dựng một phương thức khởi tạo mặc định không tham số truyền vào:

```
public KetNoi()
```

```

{
    try
    {
        connection.ConnectionString = @"Data Source=trung-pc\trung;Initial Catalog=Coffee;Integrated Security=True";
        connection.Open();
    }
    catch { }
}

```

→ Ở đây khi làm việc trên máy tính khác chỉ cần thay đổi chuỗi kết nối **ConnectionString**

-Sử dụng các StoredProcedure có bên SQL khiến quá trình truy xuất dữ liệu nhanh chóng hơn. Một dạng truy xuất dựa vào tên của StoredProcedure

```

private DataTable Execute(string NameStoredProcedure)
{
    DataTable data = new DataTable();
    SqlCommand command = new SqlCommand();
    command.Connection = connection;
    command.CommandType = CommandType.StoredProcedure;
    command.CommandText = NameStoredProcedure;
    SqlDataAdapter adapter = new SqlDataAdapter(command);
    adapter.Fill(data);
    connection.Close();
    return data;
}

```

→ Từ đây khi muốn truy xuất dữ liệu đến chỉ cần thay đổi tên của store procedure

-Load dữ liệu có dữ liệu truyền vào:

```

public DataTable LoadFood(string nameC)
{
    DataTable data = new DataTable();
    SqlCommand command = new SqlCommand();
    command.Connection = connection;
    command.CommandType = CommandType.StoredProcedure;
    command.CommandText = "SP_Load_Food";
    command.Parameters.AddWithValue("@NAMECATEGORY", SqlDbType.NVarChar).Value =
nameC;
    SqlDataAdapter adapter = new SqlDataAdapter(command);
    adapter.Fill(data);
    connection.Close();
    return data;
}

```

→ Các phương thức khác tương tự chỉ thay đổi các tham số truyền vào và tên store procedure.

## 2. Form Đăng Nhập (frmLogin)

-Xử lý nút đăng nhập như sau:

\* Xây dựng một phương thức hỗ trợ khác như sau:

```

private bool CheckLogin(string username, string password)
{
    KetNoi ketnoi = new KetNoi();
}

```



```

DataTable table = ketnoi.LoadAccount();
for (int i = 0; i < table.Rows.Count; i++)
{
    if (table.Rows[i][0].ToString() == username && table.Rows[i][1].ToString() == password )
    {
        string name = table.Rows[i][0].ToString();
        quyen = table.Rows[i][2].ToString();
        MessageBox.Show("Xin chào " + name + " :", "Đăng nhập thành công",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        return true;
    }
}
return false;
}

```

→ Từ class KetNoi kiểm tra tài khoản và mật khẩu trong cơ sở dữ liệu và thông báo ra màn hình

\*Even Button đăng nhập(btnLogIn)

→ Từ phương thức trên truyền vào dữ liệu từ bàn phím. Trả về kết quả ra màn hình:

```

if (CheckLogin(txtUserName.Text, txtPassword.Text) == true)
{
    frmMain main = new frmMain(quyen,txtUserName.Text);
    this.Hide();
    main.ShowDialog();
    this.Show();
}
else
{
    MessageBox.Show("Sai tài khoản hoặc mật khẩu", "Lỗi...", MessageBoxButtons.OK,
    MessageBoxIcon.Error);
    txtUserName.Focus();
    txtUserName.SelectAll();
}

```

### 3. Form Chính(frmMain)

-Dựa vào tài khoản mà giao diện form hiện thị nếu là admin( nhân viên quản lý) sẽ hiện đầy đủ còn staff(nhân viên bán hàng) chỉ hiện thị các chức năng bán hàng

--Load thông tin Bàn lên Form:

```

private void LoadDataTable()
{
    try
    {
        flowLayoutPanel1.Controls.Clear();
        KetNoi ketnoi = new KetNoi();
        DataTable table = ketnoi.LoadTableFood();
        for (int i = 0; i < table.Rows.Count; i++)
        {
            Button btnTable = new Button()
            {
                Name = "btnTable" + i,

```

```

        Text = table.Rows[i][1].ToString(), //ten ban
        Tag = table.Rows[i][0].ToString(), //ID Ban
        Width = 100,
        Height = 50,
    };
    //Set trạng thái ban
    if (table.Rows[i][2].ToString() == "Trống")
    {
        btnTable.BackColor = ColorTranslator.FromHtml("Snow");
    }
    else if (table.Rows[i][2].ToString() == "ONLINE")
    {
        btnTable.BackColor = ColorTranslator.FromHtml("Lime");
    }
    btnTable.MouseClick += btnTable_MouseClick;
    btnTable.MouseHover += btnTable_MouseHover;
    flowLayoutPanel1.Controls.Add(btnTable);
    }
}
catch
{
    MessageBox.Show("Không thể tải bàn!", "Lỗi...", MessageBoxButtons.OK, MessageBoxIcon.Error);
}
}

```

--Load Bill lên form:

```

public void LoadDataBill()
{
    try
    {
        //Don rac
        pnlBillInfo.Controls.Clear();
        KetNoi ketnoi = new KetNoi();
        DataTable table = ketnoi.LoadBillInfo(int.Parse(lblIDTable.Text));
        //Load thông tin các món trong bill
        int y = 10;
        for (int i = 0; i < table.Rows.Count; i++)
        {
            //Thêm vào bill
            Label lbl = new Label()
            {
                Name = "lblFB" + i,
                Text = (i + 1) + ". " + table.Rows[i][2].ToString() + " X " + table.Rows[i][3].ToString(),
                Width = pnlBillInfo.Width - 20,
                Height = 20,
                Location = new Point(5, y)
            };
            y += 25;
            pnlBillInfo.Controls.Add(lbl);
        }
    }
    catch {}
}

```

-Tạo phương thức khi nhấn vào các table

#### 4. Form thêm món(frmAddFood)

-Load Food và Category(Loại thức ăn) lên form

```

private void LoadFood()
{
    cboNameFood.Items.Clear();
}

```

```

KetNoi ketnoi = new KetNoi();
DataTable table = ketnoi.LoadAllFood();
for (int i = 0; i < table.Rows.Count; i++)
{
    cboNameFood.Items.Add(table.Rows[i][0].ToString());
}
cboNameFood.SelectedIndex = 0;
}
private void LoadCategoryFood()
{
    cboCategoryFood.Items.Clear();
    KetNoi ketnoi = new KetNoi();
    DataTable table = ketnoi.LoadAllCategoryFood();
    for (int i = 0; i < table.Rows.Count; i++)
    {
        cboCategoryFood.Items.Add(table.Rows[i][0].ToString());
    }

    cboCategoryFood.SelectedIndex = 0;
}

```

### -Các phương thức hỗ trợ khác

```

public void openTable()
{
    KetNoi ketnoi = new KetNoi();
    ketnoi.ResetTable(int.Parse(lblIDTable.Text));
}

//Them mon moi
public void addFood()
{
    KetNoi ketnoi = new KetNoi();
    ketnoi.ThemMon(int.Parse(lblIDTable.Text), lblStatus.Text, int.Parse(lblIDFood.Text),
    Int16.Parse(numQuantity.Value.ToString()), Int16.Parse(numQuantity.Value.ToString()) * int.Parse(txtPrice.Text));
    //dong thoi tang total
    SetTotal();
}
//Tang so luong mon len
public void addCountFood()
{
    KetNoi ketnoi = new KetNoi();
    ketnoi.TangSLMon(int.Parse(lblIDTable.Text), int.Parse(lblIDFood.Text),
    Int16.Parse(numQuantity.Value.ToString()));
    //dong thoi tang total
    SetTotal();
}
public void IsCountFood()
{
    try
    {
        KetNoi ketnoi = new KetNoi();
        DataTable table = ketnoi.checkFoodTable(int.Parse(lblIDTable.Text), cboNameFood.Text);
        if (Int16.Parse(table.Rows[0][2].ToString()) > 0)
        {
            isFood = true;
        }
    }
    catch
    {
        isFood = false;
    }
}

```

```
//Set Tong tien
private void SetTotal()
{
    int total = int.Parse(txtPrice.Text) * (int)numQuantity.Value;
    KetNoi ketnoi = new KetNoi();
    ketnoi.setTotal(int.Parse(lblIDTable.Text), total);
}
}
```

## 5. Form Trả món(frmReturnFood)

-Load dữ liệu lên form

```
private void LoadFood()
{
    cboNameFood.Items.Clear();
    try
    {
        // Dieu kien dung
        KetNoi ketnoi = new KetNoi();
        datatable = ketnoi.LoadBillInfo(int.Parse(lblIDTable.Text));
        for (int i = 0; i < datatable.Rows.Count; i++)
        {
            cboNameFood.Items.Add(datatable.Rows[i][2].ToString());
        }
        cboNameFood.SelectedIndex = 0;
    }
    catch
    {
        MessageBox.Show("Loi");
        condition = false;
    }
}
}
```

-Sự kiện thay đổi item trong chọn lựa món để đổi:

```
numQuantity.Maximum = 100;
KetNoi ketnoi = new KetNoi();
lblIDFood.Text = datatable.Rows[cboNameFood.SelectedIndex][6].ToString();
numQuantity.Value = int.Parse(datatable.Rows[cboNameFood.SelectedIndex][3].ToString());
numQuantity.Maximum = numQuantity.Value;
numReturnQuantity.Maximum = numQuantity.Value;
```

## 6. Form Admin(frmAdmin)

-Load dữ liệu lên form

```
void LoadDataFood()
{
    KetNoi ketnoi = new KetNoi();
    dtgvFood.DataSource = ketnoi.LoadFood_Admin();
}
void LoadDataCategoryFood()
{
    KetNoi ketnoi = new KetNoi();
    dtgvCategory.DataSource = ketnoi.LoadCategory();
}
void LoadDataTableFood()
```

```

{
    KetNoi ketnoi = new KetNoi();
    dtgvTable.DataSource = ketnoi.LoadTable_Admin();
}
void LoadDataAccounts()
{
    KetNoi ketnoi = new KetNoi();
    dtgvAccount.DataSource = ketnoi.LoadAccount_Admin();
}

void LoadBillInfo()
{
    KetNoi ketnoi = new KetNoi();
    dtgvBillInfo.DataSource = ketnoi.LoadBillInfo_Admin();
}
void LoadBill()
{
    KetNoi ketnoi = new KetNoi();
    dtgvBill.DataSource = ketnoi.LoadBill_Admin();
}

```

-Sự kiện click vào các item của Datagridview

```

void GetDataAccount()
{
    ClearTabAccount();
    try
    {
        txtUserName.DataBindings.Add("Text", dtgvAccount.DataSource, "TenUser");
        txtAccountType.DataBindings.Add("Text", dtgvAccount.DataSource, "Quyen");
    }
    catch
    {
        MessageBox.Show("Lỗi Nhận DL");
    }
}

```

→ Các tab còn lại tương tự

-Chức năng thêm

```

try
{
    KetNoi ketnoi = new KetNoi();

    ketnoi.AddFood(txtNameFood.Text, int.Parse(txtPrice.Text),
int.Parse(txtCategoryFood.Text));
    MessageBox.Show("Thêm thành công", "Thông Báo", MessageBoxButtons.OK,
MessageBoxIcon.Information);
}

```

```

        LoadDataFood();
    }
    catch
    {
        MessageBox.Show("Thêm không thành công!", "Lỗi", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
    }

```

→ Tương tự các chức năng xóa và sửa còn lại của các tab

-Chức năng tìm món ăn :

```

        try
        {
            KetNoi ketnoi = new KetNoi();
            dtgvFood.DataSource = ketnoi.SearchFood(txtContentSearch.Text);
        }
        catch
        {
            MessageBox.Show("Lỗi tìm kiếm");
        }

```

## 7. Form thông tin tài khoản

--Load thông tin user:

```

void LoadInfoUser()
{
    KetNoi ketnoi = new KetNoi();
    DataTable table = ketnoi.LoadInfoAccount(txtUserName.Text);
    txtPassWord.Text = table.Rows[0][1].ToString();
    txtRight.Text = table.Rows[0][2].ToString();
}

```

--Sự kiện thay đổi mật khẩu:

```

private void btnChangePassword_Click(object sender, EventArgs e)
{
    if (txtPassWordOld.Text=="")
    {
        MessageBox.Show("Vui lòng điền mật khẩu cũ", "Cảnh Báo", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
        txtPassWordOld.Focus();
    }
    else if (txtPassWordNew.Text=="")
    {
        MessageBox.Show("Vui lòng điền mật khẩu mới", "Cảnh Báo", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
        txtPassWordNew.Focus();
    }
    else if (txtConfigPassWord.Text=="")
    {
        MessageBox.Show("Vui lòng điền xác nhận mật khẩu mới", "Cảnh Báo",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

```

```

        txtConfigPassWord.Focus();
    }
    else if (txtConfigPassWord.Text != txtPassWordNew.Text)
    {
        MessageBox.Show("Mật khẩu mới và xác nhận không trùng khớp", "Cảnh Báo",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        txtPassWordNew.Focus();
        txtPassWordNew.SelectAll();
    }
    else if (txtPassWordOld.Text != txtPassWord.Text)
    {
        MessageBox.Show("Mật khẩu cũ không đúng. Vui lòng kiểm tra lại", "Cảnh Báo",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
        txtPassWordOld.Focus();
        txtPassWordOld.SelectAll();
    }
    else
    {
        try
        {
            KetNoi ketnoi = new KetNoi();
            ketnoi.UpDatePassWord(txtUserName.Text, txtPassWordNew.Text);
            MessageBox.Show("Mật khẩu mới đã được cập nhật", "Cảnh Báo",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
            txtPassWordNew.Clear();
            txtPassWordOld.Clear();
            txtConfigPassWord.Clear();
            LoadInfoUser();
        }
        catch
        {
            MessageBox.Show("Lỗi. Không cập nhật được mật khẩu", "Cảnh Báo",
            MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
    }
}
}

```

## 8. Form Báo Cáo

Xử lý nút Xem:

```

private void btnViewByDay_Click(object sender, EventArgs e)
{
    KetNoi ketnoi = new KetNoi();

    DataTable table = ketnoi.Report_Bill(dtpChooseDay.Value);
}

```

```
rpvReport.ProcessingMode = ProcessingMode.Local;  
rpvReport.LocalReport.ReportPath = "HoaDon.rdlc";  
ReportDataSource rds = new ReportDataSource();  
rds.Name = "dsHoaDon";  
rds.Value = table;  
rpvReport.LocalReport.DataSources.Clear();  
rpvReport.LocalReport.DataSources.Add(rds);  
rpvReport.RefreshReport();  
}
```

## VII. Hướng Dẫn Sử Dụng Phần Mềm

Tại giao diện đăng nhập thực hiện nhập tài khoản mặc định hoặc được cung cấp bởi người viết phần mềm.

Tên tài khoản (mặc định): admin    mật khẩu: 1

Khi điền đầy đủ tài khoản và mật khẩu tiếp đến nhấn nút Đăng nhập<Button>

Nếu đăng nhập thành công sẽ xuất hiện giao diện chính của chương trình, tùy vào loại(quyền) của tài khoản mà user có thể truy cập (chỉnh sửa, cập nhật) nội dung trong giao diện.

Nếu User đang chạy là thuộc quyền nhân viên bán hàng → sẽ được sử dụng những chức năng sau:

### 1. Thêm Món Cho Một Bàn

B1: Chọn vào bàn có trạng thái trống( hay màu nền là màu trắng) hệ thống sẽ thông báo hợp thoại yêu cầu user mở bàn mới. [Nếu Bàn trống sẽ xuất hiện hộp thoại trên]. Ngược lại khi chọn bàn có màu xanh( hay trạng thái online) thì sẽ trực tiếp chuyển sang bước 2

B2: Hệ thống sẽ tiếp tục chuyển qua giao diện thêm món. Tại đây khi chọn món và nhập số lượng sau đó nhấn Nút thêm món <Buttton>. Nếu đã đầy đủ món user có thể nhấn thoát .

### 2. Trả Món Cho Một Bàn

B1: Chọn bàn cần trả. Tiếp đó chọn món cần trả và nhập(hoặc nhập) số lượng vào .

B2: Nhấp chọn xác nhận trả món<Buttton>

B3: Hệ thống sẽ đưa ra cảnh báo xác nhận có thực sự trả không. Tại đây nhấn <yes> để đồng ý trả món. Sau khi trả thành công user tiếp tục trả lại món. Khi muốn dừng việc trả thì thoát khỏi giao diện Trả món



### 3. Thanh Toán Một Bàn.

B1: Chọn bàn có trạng thái online hoặc có nền màu xanh lá. Tiếp đó nhấn nút thanh toán trên giao diện<Button>

B2: Tại đây giao diện sẽ xuất hiện hộp thoại mới. Cảnh báo user có thực sự muốn thanh toán không?

B3: Nhấn Yes<đồng ý> để tiếp tục thanh toán

### 4. Xem thông tin tài khoản và đổi mật khẩu

\*Xem thông tin

B1: Tại giao diện chính làm việc

B2: Chọn chức năng Thông tin tài khoản trên thanh Menustrip

\*Đổi mật khẩu

B1 + B2: Như trên

B3: Nhập mật khẩu cũ, mật khẩu mới, và xác nhận mật khẩu.

B4: Nhấn nút Đổi mật khẩu<Button>

Nếu User đang chạy là thuộc quyền nhân viên quản lý → sẽ được sử dụng những chức năng trên và một số chức năng riêng:

### 5. Thêm, Sửa, Xóa Thức Ăn-Loại Thức Ăn-Bàn-Tài Khoản-Hóa Đơn-Chi Tiết Hóa Đơn

\*Thêm

B1: Nhập đầy đủ thông tin.

B2: Nhấp chọn nút Thêm

\*Sửa

B1: Tương tự như chức năng Thêm.

B2: Nhấp chọn nút Sửa

\*Xóa

B1: Tương tự như trên

B2: Nhấp chọn nút Xóa

### 6. Thống kê

Lưu ý: Khi đăng nhập với quyền admin: người dùng có thể sử dụng chức năng thống kê

B1: Chọn ngày

B2: Nhấn nút Xem<Button>

## VIII. Tổng Kết & Hướng Phát Triển

### 1. Tổng kết:

- + Phần mềm giúp công việc lưu trữ và quản lý thông tin hóa đơn tiện lợi, rành mạch.
- + Giao diện hiện đại dễ sử dụng với người dùng.
- + Tính tương thích và bảo mật cao.
- + Tiết kiệm thời gian và công sức tăng độ chính xác.

### 2. Hướng phát triển

- + Tạo một list lưu các mã code □ từ đó khiến việc thêm món trở nhanh chóng hơn □ giúp tiết kiệm thời gian nhập liệu và dữ liệu truyền vào chính xác hơn
- + Form Admin có thể đi sâu vào phân xử lý dữ liệu cũng như thêm các ràng buộc để dữ liệu nhập từ bàn phím chính xác.
- + Thêm tính năng tạo báo cáo (xuất file Excel)
- + Cải thiện giao diện thao tác dễ sử dụng
- + Thay các nút buttons bằng các phím tắt khiến người sử dụng thao tác nhanh chóng
- + Thêm các chức năng thực tế như sau: gộp bàn, chuyển bàn, phát nhạc trực tuyến....vv...

## IX. Tài Liệu Kham Khảo

Nguồn:

<https://www.howkteam.vn>

<https://tailieu.vn>

<https://sharecode.vn>

<http://luanvan.co>

Hình ảnh:

Phần mềm có sử dụng các hình ảnh từ các nguồn Internet:

<https://icons8.com/>

<https://www.flaticon.com>

<http://thuviendohoa.vn>