**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THỰC PHẨM TP. HỒ CHÍ MINH**



**TIỂU LUẬN : HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**ĐỀ TÀI: Quản lí cửa hàng bán đồng hồ**

Giảng viên hướng dẫn: **Đinh Thị Mận**

Nhóm sinh viên thực hiện: Nhóm 5

**THÀNH PHỐ HỒ CHÍ MINH, THÁNG 9 NĂM 2023**

# Lời cảm ơn

Để hoàn thành được bài tiểu luận này, em xin chân thành cảm ơn Ban Giám hiệu, các khoa, phòng và quý thầy, cô của trường Đại Học Công Thương Thành Phố Hồ Chí Minh những người đã tận tình giúp đỡ và tạo điều kiện cho em trong quá trình học tập. Đặc biệt, em xin gửi lời cảm ơn sâu sắc đến cô Đinh Thị Mận - người đã trực tiếp giảng dạy và hướng dẫn em thực hiện bài tiểu luận này bằng tất cả lòng nhiệt tình và sự quan tâm sâu sắc.

Trong quá trình thực hiện bài tiểu luận này, do hiểu biết còn nhiều hạn chế nên bài làm khó tránh khỏi những thiếu sót. Em rất mong nhận được những lời góp ý của quý thầy cô để bài tiểu luận ngày càng hoàn thiện hơn.

Em xin chân thành cảm ơn!

**MỤC LỤC**

[Lời cảm ơn 2](#_Toc18916)

[CHƯƠNG 1. GIỚI THIỆU 1](#_Toc4517)

[1.1. Lý do chọn đề tài 1](#_Toc19926)

[1.2. Mô tả bài toán 1](#_Toc24267)

[1.3. Mô tả chức năng hệ thống 1](#_Toc16194)

[1.4. Mô hình quan hệ ERD 2](#_Toc30727)

[1.5. Mô tả các bảng 2](#_Toc17065)

[CHƯƠNG 2. THIẾT KẾ VÀ CÀI ĐẶT CƠ SỞ DỮ LIỆU 4](#_Toc14751)

[2.1. Cài đặt các bảng 4](#_Toc28444)

[2.2. Câu lệnh 6](#_Toc25272)

[2.3. Hàm 8](#_Toc25114)

[2.4. Thủ tục 11](#_Toc20990)

[2.5. Trigger 15](#_Toc550)

[2.6. Cusor 16](#_Toc21604)

[CHƯƠNG 3. CÀI ĐẶT ỨNG DỤNG MINH HỌA 19](#_Toc20213)

[3.1. Chức năng danh mục 19](#_Toc32535)

[3.2. Chức năng thống kê 22](#_Toc29730)

[3.3. Chức năng tìm kiếm 22](#_Toc5595)

[3.4. Chức năng nghiệp vụ 22](#_Toc14997)

[CHƯƠNG 4. TỔNG KẾT 24](#_Toc18786)

[4.1. Ưu điểm 24](#_Toc5626)

[4.2. Khuyết điểm 24](#_Toc13671)

[4.3. Hướng phát triển 24](#_Toc14066)

**PHỤ LỤC TRA CỨU**

**ẢNH**

[Ảnh 1.1 .Daiaram. 2](#_Toc972)

[Ảnh 3.1 .Giao diện sản phẩm. 20](#_Toc17186)

[Ảnh 3.2 .Giao diện hóa đơn. 20](#_Toc29462)

[Ảnh 3.3 .Giao diện chi tiết hóa đơn. 21](#_Toc1650)

[Ảnh 3.4 .Giao diện nhân viên. 22](#_Toc26894)

[Ảnh 3.5 .Giao diện thống kê. 22](#_Toc32359)

[Ảnh 3.6 .Giao diện login. 23](#_Toc17900)

[Ảnh 3.7 .Thêm sản phẩm. 23](#_Toc23205)

**BẢNG**

[Bảng 1.1 .Bảng Product 2](#_Toc4486)

[Bảng 1.2 .Bảng Category 2](#_Toc15858)

[Bảng 1.3 .Bảng Vendor 3](#_Toc14548)

[Bảng 1.4 .Bảng Account 3](#_Toc25833)

[Bảng 1.5 .Bảng UserDetails 3](#_Toc26463)

[Bảng 1.6 .Bảng Costumer 3](#_Toc16237)

[Bảng 1.7 .Bảng Gender 3](#_Toc15324)

[Bảng 1.8 .Bảng IsAdmin 4](#_Toc17807)

[Bảng 1.9 .Bảng Order 4](#_Toc28553)

[Bảng 1.10 .Bảng OrderDetail 4](#_Toc15)

# 

# GIỚI THIỆU

## Lý do chọn đề tài

## Mô tả bài toán

Quản lí cửa hàng bán đồng hồ là một phần mềm yêu cầu các tác vụ về :

Bán hàng , quản lý sản phẩm , quản lí nhân viên ,...Để các tác vụ này diễn ra thuận lợi , chúng ta cần một cơ sở dữ liệu hoàn chỉnh theo quy trình : bán hàng , báo cáo , doanh thu.

Thêm sản phẩm mới vào cửa hàng :

## Mô tả chức năng hệ thống

Quản lí cửa hàng bán đồng hồ gồm các chức năng :

**Quản lí sản phẩm :**

- Thêm sản phẩm mới vào cho cửa hàng :

- Cập nhật sản phẩm : thêm xóa sửa , thông tin , giá cả sản phẩm khi thông tin bị sai , hoặc giá cả sản phẩm phầm khi sản phẩm tồn quá lâu.

- Thêm : khi nhu cầu mua tăng và cửa hàng không còn hàng.

- Xóa :Khi cửa hàng không còn kinh doanh mặt hàng đó nữa.

**Quản lí hóa đơn**

- Tạo hóa đơn nhập

- Tạo chi tiết hóa đơn cho hóa đơn nhập

**Quản lí nhân viên**

-Thêm : Khi có nhân viên mới vào làm , tạo tài khoản để nhân viên có thể đăng nhập vào phần mềm của cửa hàng.

- Xóa : Khi nhân viên đó nghỉ việc .

- Cập nhật : Giúp nhân viên cập nhật lại thông tin của mình , hoặc admin cấp lại mật khẩu khi nhân viên đó quên mật khẩu.

**Báo cáo :**

- Thống kê : Tổng tiền của hàng bán được theo hóa đơn , được thống kê theo , ngày , tháng , hoặc năm.

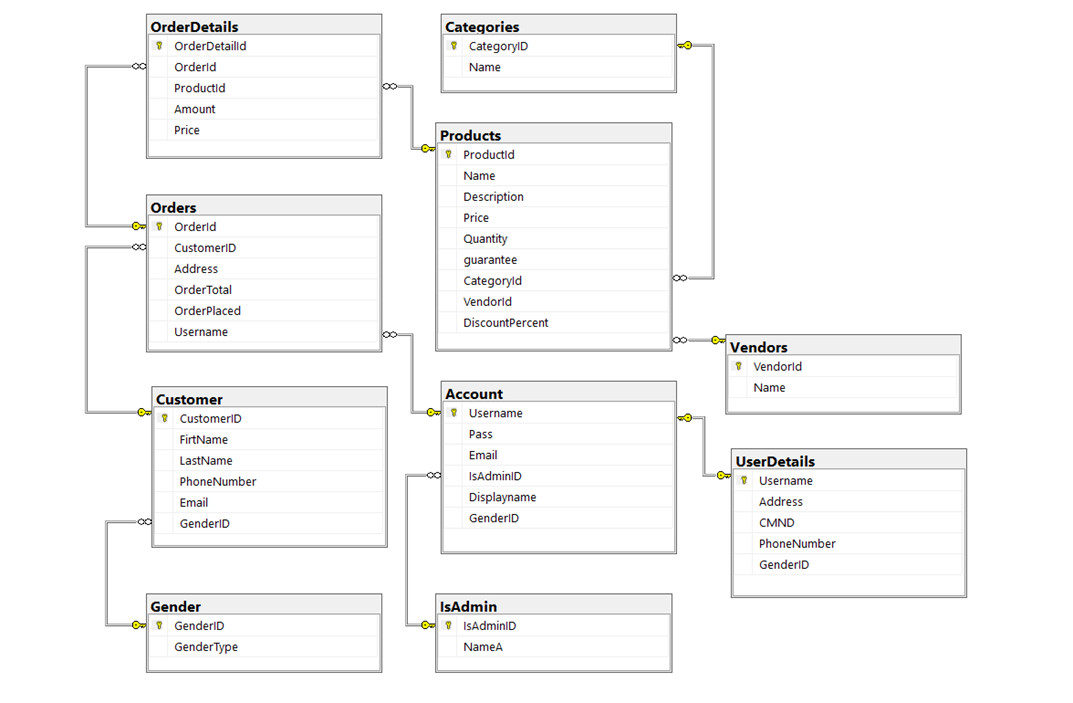
**Phân quyền :**

Quản lí cửa hàng gồm có 2 nhóm quyền : Sale và Admin

Sale : được phép thêm , xóa sửa sản phẩm , taọ nhập hóa đơn , thống kê doanh thu .

Admin : được phép toàn quyền hệ thống , thêm xóa sửa thông tin nhân viên ra khỏi cửa hàng.

## Mô hình quan hệ ERD



Ảnh 1.1.Daiaram.

## Mô tả các bảng

Bảng liên quan tới sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| ProductId | Mã sản phẩm | Char | 50 |
| Name | Tên sản phẩm | Nvarchar | 100 |
| Description | Mô tả | Nvarchar | Max |
| Price | Giá | Price |  |
| Quantity | Số lượng | Int |  |
| Guarantee | Bảo hành | Int |  |
| CategoryId | Mã nhà cung cấp | Varchar | 50 |
| VendorId | Mã loại sản phẩm | Varchar | 50 |
| DiscountPercent | Giảm giá | Float |  |

Bảng 1.1.Bảng Product

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| CategoryID | Mã loại | Varchar | 50 |
| Name | Tên loại | Nvarchar | 100 |

Bảng 1.2.Bảng Category

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| VendorID | Mã nhà cung cấp | Varchar | 50 |
| Name | Tên nhà cung cấp | Nvarchar | 50 |

Bảng 1.3.Bảng Vendor

Bản liên quan đến người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| Username | Tên đăng nhập | Varchar | 50 |
| Pass | Mật khẩu | Int | 100 |
| Email | Email | Varchar | 100 |
| IsAdmin | Quyền | Varchar | 8 |
| Displayname | Tên hiển thị | Nvarchar | 50 |
| GenderID | Mã giới tính | Char | 2 |

Bảng 1.4.Bảng Account

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| Username | Tên đăng nhập | Varchar | 50 |
| Address | Địa chỉ | Int | 100 |
| CMND | CMND | Varchar | 12 |
| Phone | Số điện thoại | Char | 11 |

Bảng 1.5.Bảng UserDetails

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| CustomerID | Tên đăng nhập | Varchar | 50 |
| FirtName | Địa chỉ | Int | 100 |
| LastName | CMND | Varchar | 12 |
| PhoneNumber | Số điện thoại | Char | 11 |
| Email | Email | Varchar | 50 |
| GenderID | Mã giới tính | GenderID |  |

Bảng 1.6.Bảng Costumer

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| GenderID | Mã giới tính | Int |  |
| GenderType | Giới tính | Nvarchar | 5 |

Bảng 1.7.Bảng Gender

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| IsAdmin | Mã quyền | Varchar | 8 |
| NameA | Tên nhóm quyền | NVarchar | 100 |

Bảng 1.8.Bảng IsAdmin

Bảng liên quan đến hóa đơn

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| OrderID | Mã hóa đơn | Varchar | 50 |
| CustomerID | Mã khách hàng | Varchar | 20 |
| Address | Địa chỉ lập hóa đơn | Nvarchar | 100 |
| OrderTotal | Tổng hóa đơn | Float |  |
| OrderPlace | Thời gian lập | Date |  |
| Usename | Tên người lập | Varchar | 50 |

Bảng 1.9.Bảng Order

|  |  |  |  |
| --- | --- | --- | --- |
| Tên trường | Mô tả | Kiểu dữ liệu | Độ dài |
| OrderDetaild | Mã chi tiết hóa đơn | Varchar | 50 |
| OrderId | Mã hóa đơn | Varchar | 50 |
| ProductId | Mã sản phẩm | Varchar | 50 |
| Amount | Số lượng | Float |  |
| Price | Giá của sản phẩm | Date |  |

Bảng 1.10.Bảng OrderDetail

# THIẾT KẾ VÀ CÀI ĐẶT CƠ SỞ DỮ LIỆU

## Cài đặt các bảng

A)Bảng Account

CREATE TABLE Account

(

Username VARCHAR(50) NOT NULL,

Pass VARCHAR(50),

Email varchar(100),

IsAdminID varchar(8),

Displayname NVarChar(50),

Constraint pk\_tk primary key (Username)

)

B)Bảng UserDetails

CREATE TABLE UserDetails

(

Username VARCHAR(50) NOT NULL PRIMARY KEY,

Address NVARCHAR(100),

CMND VARCHAR(12),

PhoneNumber VARCHAR(11),

)

C)Bảng IsAdmin

create table IsAdmin

(

IsAdminID varchar(8) not null,

NameA NVarChar(100),

Constraint pk\_Quyen primary key (IsAdminID)

)

D)Bảng Categories

CREATE TABLE Categories

(

CategoryID varchar(50) not null,

Name nvarchar(100),

Constraint pk\_cat primary key (CategoryID)

)

E)Bảng Vendors

CREATE TABLE Vendors

(

VendorId varchar(50) not null,

Name nvarchar(50),

Constraint pk\_Vendors primary key (VendorId)

)

F)Bảng Orders

create table Orders

(

OrderId varchar(50) not null,

CustomerID nvarchar(20),

Address nvarchar(100),

OrderTotal float,

OrderPlaced nvarchar(50),

Username VARCHAR(50) ,

Constraint pk\_Orders primary key (OrderId)

)

alter table Orders

alter column OrderPlaced date

G)Bảng Customer

create table Customer

(

CustomerID nvarchar(20)primary key not null,

FirtName nvarchar(20),

LastName nvarchar(50),

PhoneNumber varchar(50),

Email varchar(50),

GenderID INT,

)

H)Bảng Gender

create table Gender

(

GenderID INT primary key not null ,

GenderType NVARCHAR(5),

)

K)Bảng OrderDetails

Create table OrderDetails

(

OrderDetailId varchar(50) not null,

OrderId varchar(50),

ProductId varchar(50),

Amount int ,

Price float ,

Constraint pk\_OrderDetails primary key (OrderDetailId)

)

I)Bảng Products

Create table Products

(

ProductId varchar(50) not null,

Name nvarchar(100),

Description nvarchar(100),

Price float,

Quantity int ,

guarantee varchar(50),

CategoryId varchar(50),

VendorId varchar(50),

DiscountPercent float,

Constraint pk\_Products primary key (ProductId)

)

## Câu lệnh

**Insert**

|  |  |
| --- | --- |
| **1.Thêm quyền vào bảng IsAdmin** | |
| **Code** | Insert into IsAdmin VALUES  ('ad' , 'Admin'), |
| **Kết quả** |  |

|  |  |
| --- | --- |
| **2.Thêm nhân viên vào bảng Account** | |
| **Code** | Insert into Account VALUES  ('anhminh' ,'nguyenhai1231', 'anhmin123@gmail.com' , 'sl' , N'Anh Minh' , 1), |
| **Kết quả** |  |

|  |  |
| --- | --- |
| **3.Thêm thông tin bảng Gender** | |
| **Code** | INSERT INTO Gender VALUES  (1,N'NAM'),  (2,N'NỮ') |
| **Kết quả** |  |

|  |  |
| --- | --- |
| **4.Thêm nhân viên cho bảng Customer** | |
| **Code** | Insert into Customer VALUES  ('KH01',N'Anh',N'Nguyễn Hải','0903926730',N'haianh34@gmail.com',1), |
| **Kết quả** |  |

|  |  |
| --- | --- |
| **5.Thêm quyền vào bảng Categories** | |
| **Code** | INSERT INTO Categories  VALUES ('CAT01',N'Đồng Hồ Cơ'), |
| **Kết quả** |  |

|  |  |
| --- | --- |
| **6.Thêm thông tin cho bảng Vendor** | |
| **Code** | INSERT INTO Vendors  VALUES ('VEN01',N'Thế Giới Di Động'), |
| **Kết quả** |  |

|  |  |
| --- | --- |
| **7.Thêm thông tin cho bảng Products** | |
| **Code** | INSERT INTO Products  VALUES ('PRO01',N'Đồng hồ thông minh Microwear GT4 Max',N'Sở hữu thiết kế thời thượng, sang trọng. Cùng với các thuật toán thông minh giúp việc hoạt động hiệu quả, cung cấp sức mạnh nâng cao khả năng tính toán…',950000,10,N'2 Năm','CAT03','VEN04',20), |
| **Kết quả** |  |

## Hàm

|  |
| --- |
| 1)LẤY HỌ TÊN, SĐT, EMAIL CỦA CUSTOMER KHI TRUYỀN VÀO CUSTOMERID |
| CREATE FUNCTION f\_inTT\_Customer(@CUSID VARCHAR(50))  RETURNS @TT\_CUSTOMER table (HOTEN NVARCHAR(50), SDT VARCHAR(10), EMAIL VARCHAR(50))  AS  BEGIN  if exists(select \* from Customer where CustomerID = @CUSID)  insert into @TT\_CUSTOMER  select CONCAT(LastName, ' ', FirtName), PhoneNumber, Email  from Customer  where CustomerID = @CUSID  RETURN  END |

**Kết quả test**

|  |  |
| --- | --- |
| Code | select \* from dbo.f\_inTT\_Customer('KH021') |
| Kết quả |  |

|  |
| --- |
| 2)CHO BIẾT SỐ LƯỢNG ĐƠN HÀNG CUSTOMER ĐÃ ĐẶT KHI TRUYỀN VÀO CUSTOMERID |
| CREATE FUNCTION f\_inSL\_DH\_Customer\_Order(@CUSID VARCHAR(50))  RETURNS @TT\_CUSTOMER table (HOTEN\_KH NVARCHAR(50), SL\_DONHANG INT)  AS  BEGIN  if exists(select \* from Orders where CustomerID = @CUSID)  insert into @TT\_CUSTOMER  select CONCAT(LastName, ' ', FirtName), Count(OrderDetailId)  from Orders O, Customer C, OrderDetails OD  where O.CustomerID = C.CustomerID and C.CustomerID = @CUSID and O.OrderId = OD.OrderId  group by CONCAT(LastName, ' ', FirtName)  RETURN  END  go |

**Kết quả test**

|  |  |
| --- | --- |
| Code | select \* from dbo.f\_inSL\_DH\_Customer\_Order('KH01') |
| Kết quả |  |

|  |
| --- |
| 3)In ra danh sách hóa đơn bán khi truyền vào mã nhân viên |
| create function f\_inDS\_HoaDonDaLap(@username varchar(30))  returns @tblHD table(MaHD varchar(30))  as  begin  insert into @tblHD  select OrderId from Orders where Username = @username  return  end |

**Kết quả test**

|  |  |
| --- | --- |
| **Code** | select \* from dbo.f\_inDS\_HoaDonDaLap('nhut1235') |
| **Kết quả** |  |

|  |
| --- |
| 4)In ra hóa đơn bán được nhiều tiền nhất của nhân viên |
| create function f\_InHoTenNVBanDuocNhieuDonNhat(@username varchar(30))  returns @ttnv table(HOTEN nvarchar(30), TONGTIEN FLOAT)  as  begin  insert into @ttnv  select Displayname, Sum(OrderTotal) from Orders O, Account A  where O.Username = A.Username and O.Username = @username  group by Displayname  return  end |

**Kết quả test**

|  |  |
| --- | --- |
| Code | select \* from dbo.f\_InHoTenNVBanDuocNhieuDonNhat('nhut1235') |
| Kết quả |  |

|  |
| --- |
| 5)Tổng đồng hồ hiện có |
| CREATE FUNCTION f\_Tong\_SoLuongDongHo()  RETURNS INT  BEGIN  DECLARE @TONG INT  SET @TONG=(SELECT SUM(Quantity) FROM Products)  RETURN @TONG  END |

**Kết quả test**

|  |  |
| --- | --- |
| Code | DECLARE @TONG\_SoLuongDongHo INT  SET @TONG\_SoLuongDongHo=dbo.f\_Tong\_SoLuongDongHo()  PRINT N'Tổng số lượng đồng hồ shop hiện có là : '+Convert(varchar,@TONG\_SoLuongDongHo)+N' cái. ' |
| Kết quả |  |

|  |
| --- |
| 6)Mặt hàng bán chạy nhất |
| CREATE FUNCTION f\_mathang\_banchaynhat()  RETURNS NVARCHAR(50)  AS  BEGIN  DECLARE @TENMH NVARCHAR(50)  SELECT TOP 1 @TENMH = pro.Name  FROM OrderDetails ord  INNER JOIN Products pro ON pro.ProductId = ord.ProductId  GROUP BY pro.Name  ORDER BY SUM(ord.Amount) DESC -- Lấy sản phẩm có tổng Amount lớn nhất  RETURN @TENMH  END |

**Kết quả test**

|  |  |
| --- | --- |
| Code | SELECT dbo.f\_mathang\_banchaynhat() AS TenMatHangBanChayNhat; |
| Kết quả |  |

## Thủ tục

|  |
| --- |
| 1)Mặt hàng bán chạy nhất |
| CREATE PROC p\_mathang\_banchaynhat  AS  DECLARE @TENMH NVARCHAR(50)  SET @TENMH=(SELECT pro.Name from OrderDetails ord, Products pro where pro.ProductId=ord.ProductId AND ORD.Amount=(SELECT top 1 SUM(Amount) from OrderDetails ord, Products pro where pro.ProductId=ord.ProductId  GROUP BY pro.ProductId,ord.ProductId))  PRINT @TENMH |

**Kết quả test**

|  |  |
| --- | --- |
| Code | DECLARE @TenMH NVARCHAR(50)  EXEC p\_mathang\_banchaynhat @TenMH OUTPUT  SELECT @TenMH AS TenMatHangBanChayNhat |
| Kết quả |  |

|  |
| --- |
| 2)In ra danh sách hóa đơn khi truyền vào mã nhân viên |
| create proc p\_inDS\_HoaDonDaLap @username varchar(30)  as  select OrderId from Orders where Username = @username |

**Kết quả test**

|  |  |
| --- | --- |
| Code | EXEC p\_inDS\_HoaDonDaLap 'nhut1235' |
| Kết quả |  |

|  |
| --- |
| 3)In ra họ tên nhân viên có tổng tiền bán được nhiều nhất dựa trên bảng order |
| create PROC p\_nhanvien\_banduocnhieunhat\_OrderTotal  as  DECLARE @TENNV NVARCHAR(50)  SET @TENNV=(SELECT ac.Displayname FROM Orders ords,Account ac where ac.Username=ords.Username and ords.OrderTotal=(SELECT top 1 sum(OrderTotal) FROM Orders ords,Account ac where ac.Username=ords.Username  GROUP BY ords.CustomerID,ac.Username))  print @TENNV |

**Kết quả test**

|  |  |
| --- | --- |
| Code | exec p\_nhanvien\_banduocnhieunhat\_OrderTotal |
| Kết quả |  |

|  |
| --- |
| 4)Trả về số lượng sản phẩm khi truyền vào mã sản phẩm |
| CREATE PROCEDURE GetProductQuantity  @ProductId VARCHAR(50),  @Quantity INT OUTPUT  AS  BEGIN  SELECT @Quantity = Quantity  FROM Products  WHERE ProductId = @ProductId  END |

**Kết quả test**

|  |  |
| --- | --- |
| Code | DECLARE @OutputQuantity INT  EXEC GetProductQuantity @ProductId = 'PRO01', @Quantity = @OutputQuantity OUTPUT  SELECT @OutputQuantity AS ProductQuantity |
| Kết quả |  |

|  |
| --- |
| 5)Lấy danh sách sản phẩm khi truyền vào nhà cung cấp |
| CREATE PROCEDURE GetProductsByVendor  @VendorId VARCHAR(50)  AS  BEGIN  SELECT \*  FROM Products  WHERE VendorId = @VendorId  END |

**Kết quả test**

|  |  |
| --- | --- |
| Code | DECLARE @VendorIdParam VARCHAR(50) = 'VEN01'  EXEC GetProductsByVendor @VendorIdParam |
| Kết quả |  |

|  |
| --- |
| 6)Thêm sản phẩm |
| CREATE PROCEDURE InsertProduct  @ProductId VARCHAR(50),  @Name NVARCHAR(100),  @Description NVARCHAR(100),  @Price FLOAT,  @Quantity INT,  @Guarantee VARCHAR(50),  @CategoryId VARCHAR(50),  @VendorId VARCHAR(50),  @DiscountPercent FLOAT  AS  BEGIN  INSERT INTO Products (ProductId, Name, Description, Price, Quantity, Guarantee, CategoryId, VendorId, DiscountPercent)  VALUES (@ProductId, @Name, @Description, @Price, @Quantity, @Guarantee, @CategoryId, @VendorId, @DiscountPercent)  END |

**Kết quả test**

|  |  |
| --- | --- |
| Code | EXEC InsertProduct  @ProductId = 'PRO123',  @Name = 'Casio World Time AE-1200WHD',  @Description = N'Đồng hồ nam Casio AE1200WHD có mặt đồng hồ vuông to với phong cách thể thao',  @Price = 1373000,  @Quantity = 10,  @Guarantee = 12,  @CategoryId = 'CAT14',  @VendorId = 'VEN01',  @DiscountPercent = 0 |
| Kết quả |  |

|  |
| --- |
| 7)Xóa sản phẩm |
| CREATE PROCEDURE DeleteProduct  @ProductId VARCHAR(50)  AS  BEGIN  DELETE FROM Products  WHERE ProductId = @ProductId  END |

**Kết quả test**

|  |  |
| --- | --- |
| Code | EXEC DeleteProduct @ProductId = 'PRO123' |
| Kết quả |  |

|  |
| --- |
| 8)Lấy danh sách sản phẩm còn |
| CREATE PROCEDURE p\_ds\_SanPham\_ConHang  AS  BEGIN  SELECT \*  FROM Products  WHERE Quantity > 0;  END  GO  EXEC p\_ds\_SanPham\_ConHang |

**Kết quả test**

|  |
| --- |
| **Kết quả** |
|  |

|  |
| --- |
| 9)Lấy danh sách sản phẩm còn |
| CREATE PROC p\_tangsl\_len10\_ID  @ProductID varchar(50)  AS  UPDATE Products SET Quantity = Quantity + 10 where ProductId=@ProductID  EXEC p\_tangsl\_len10\_ID 'PRO07' |

**Kết quả test**

|  |  |
| --- | --- |
| **Trước** | **Sau** |
|  |  |

## Trigger

|  |
| --- |
| 1)Trigger Cập nhật tổng tiền của hóa đơn theo chi tiết hóa đơn. |
| CREATE TRIGGER updateOrderTotal  ON orderDetails  AFTER INSERT, UPDATE, DELETE  AS  BEGIN  UPDATE Orders  SET OrderTotal=(SELECT SUM(ord.amount \* ord.price) FROM orderDetails ord,inserted i WHERE ord.OrderId=i.OrderId )  From Orders ord  where ( select i.OrderId from inserted i)=ord.OrderId  UPDATE Orders  SET OrderTotal=OrderTotal-(SELECT SUM(ord.amount \* ord.price) FROM orderDetails ord,deleted del WHERE ord.OrderId=del.OrderId )  From Orders ord  where ( select del.OrderId from deleted del)=ord.OrderId  END |

Kết quả test

|  |  |
| --- | --- |
|  |  |

|  |
| --- |
| 2)Trigger giảm số lượng hàng trong kho khi thêm số lượng vào OrderDetails |
| CREATE TRIGGER trg\_CapNhapProduct on OrderDetails  for INSERT, UPDATE, DELETE  AS  begin  UPDATE Products  set quantity = quantity - (SELECT Amount FROM INSERTED)  FROM Products P, INSERTED i  WHERE P.productid = i.productid    UPDATE Products  set quantity = quantity + (SELECT Amount FROM DELETED)  FROM Products P, DELETED i  WHERE P.productid = i.productid  END |

|  |
| --- |
| 3)Trigger cập nhật mô tả “Hết sản phẩm” khi số lượng về 0 |
| CREATE TRIGGER trg\_UpdateProductDescription  ON Products  AFTER UPDATE  AS  BEGIN  IF UPDATE(Quantity)  BEGIN  UPDATE Products  SET Description = N'Hết sản phẩm'  FROM inserted i  WHERE Products.ProductId = i.ProductId  AND i.Quantity = 0  END  END |

|  |
| --- |
| 4)Khi tạo account , tạo thông tin cho userDetails |
| CREATE TRIGGER set\_undefined\_address  ON Account  AFTER INSERT  AS  BEGIN  SET NOCOUNT ON;  -- Insert vào UserDetails cho mỗi dòng được chèn vào bảng Account  INSERT INTO UserDetails (Username, Address)  SELECT ins.Username, 'Không xác định'  FROM inserted ins;  END; |

## Cusor

|  |
| --- |
| 1)Nhập vào ProductId In ra các thông tin sau CustumerId,LastName,FirstName |
| DECLARE @CUSTOMERID\_LASTNAME\_FIRSTNAME NVARCHAR(MAX)  DECLARE @CUSTOMERID VARCHAR(50)  DECLARE @FIRSTNAME NVARCHAR(50)  DECLARE @LASTNAME NVARCHAR(50)  DECLARE cursor\_CUSTOMERID\_LASTNAME\_FIRSTNAME CURSOR FOR  SELECT cus.CustomerID,cus.FirtName,cus.LastName from Customer cus,Orders ords,OrderDetails ordd,Products pro  WHERE cus.CustomerID=ords.CustomerID AND ords.OrderId=ordd.OrderId AND ordd.ProductId=pro.ProductId AND pro.ProductId='PRO06'  OPEN cursor\_CUSTOMERID\_LASTNAME\_FIRSTNAME  FETCH NEXT FROM cursor\_CUSTOMERID\_LASTNAME\_FIRSTNAME INTO @CUSTOMERID, @FIRSTNAME,@LASTNAME  WHILE @@FETCH\_STATUS = 0  BEGIN  SET @CUSTOMERID\_LASTNAME\_FIRSTNAME = N'Mã Khách Hàng : ' + @CUSTOMERID + ' - Tên Khách Hàng : '+ @LASTNAME+' '+ @FIRSTNAME  PRINT @CUSTOMERID\_LASTNAME\_FIRSTNAME  FETCH NEXT FROM cursor\_CUSTOMERID\_LASTNAME\_FIRSTNAME INTO @CUSTOMERID, @FIRSTNAME,@LASTNAME  END  CLOSE cursor\_CUSTOMERID\_LASTNAME\_FIRSTNAME  DEALLOCATE cursor\_CUSTOMERID\_LASTNAME\_FIRSTNAME |

|  |
| --- |
| **Kết quả test** |
|  |

|  |
| --- |
| 2)In thông tin sản phẩm từ bảng Products |
| DECLARE @ProductId VARCHAR(50)  DECLARE @ProductName NVARCHAR(100)  DECLARE productCursor CURSOR FOR  SELECT ProductId, Name  FROM Products  OPEN productCursor  FETCH NEXT FROM productCursor INTO @ProductId, @ProductName  WHILE @@FETCH\_STATUS = 0  BEGIN  -- Do something with @ProductId and @ProductName here  PRINT 'Product ID: ' + @ProductId + ', Product Name: ' + @ProductName  FETCH NEXT FROM productCursor INTO @ProductId, @ProductName  END  CLOSE productCursor  DEALLOCATE productCursor |

|  |
| --- |
| **Kết quả test** |
|  |

|  |
| --- |
| 3)In ra username, tên nhân viên, quyền, địa chỉ của tất cả nhân viên |
| --In ra username, tên nhân viên, quyền, địa chỉ của tất cả nhân viên  DECLARE @username VARCHAR(50),  @displayname NVARCHAR(50),  @roleName NVARCHAR(50),  @Address NVARCHAR(50)  DECLARE account\_cursor CURSOR FOR  SELECT A.username, A.displayname, I.NameA, U.Address  FROM Account A  LEFT JOIN isAdmin I ON A.isAdminId = I.isAdminId  LEFT JOIN UserDetails U ON A.username = U.username;  OPEN account\_cursor;  FETCH NEXT FROM account\_cursor INTO @username, @displayname, @roleName, @Address;  WHILE @@FETCH\_STATUS = 0  BEGIN  -- In ra thông tin Account tại đây  PRINT 'Username: ' + @username + ', Display Name: ' + @displayname + ',Role: ' + @roleName + ',Address: ' + @Address;  -- Nếu không có displayname, hiển thị Chưa có  IF @displayname IS NULL  PRINT 'Username: ' + @username + ', Display Name: Chưa có' + ', Role: ' + @roleName + N', Address: Không xác định';  -- Nếu không có địa chỉ, hiển thị Không xác định  IF @Address IS NULL  PRINT 'Username: ' + @username + ', Display Name: ' + @displayname + ', Role: ' + @roleName + N', Address: Không xác định';  FETCH NEXT FROM account\_cursor INTO @username, @displayname, @roleName, @Address;  END  CLOSE account\_cursor;  DEALLOCATE account\_cursor; |

|  |
| --- |
| **Kết quả test** |
|  |

|  |
| --- |
| 4)Cusor kết hợp funtion ,xuất mã khách hàng , tên khách hàng khi truyền mã khách hàng |
| --In ra username, tên nhân viên, quyền, địa chỉ của tất cả nhân viên  DECLARE @username VARCHAR(50),  @displayname NVARCHAR(50),  @roleName NVARCHAR(50),  @Address NVARCHAR(50)  DECLARE account\_cursor CURSOR FOR  SELECT A.username, A.displayname, I.NameA, U.Address  FROM Account A  LEFT JOIN isAdmin I ON A.isAdminId = I.isAdminId  LEFT JOIN UserDetails U ON A.username = U.username;  OPEN account\_cursor;  FETCH NEXT FROM account\_cursor INTO @username, @displayname, @roleName, @Address;  WHILE @@FETCH\_STATUS = 0  BEGIN  -- In ra thông tin Account tại đây  PRINT 'Username: ' + @username + ', Display Name: ' + @displayname + ',Role: ' + @roleName + ',Address: ' + @Address;  -- Nếu không có displayname, hiển thị Chưa có  IF @displayname IS NULL  PRINT 'Username: ' + @username + ', Display Name: Chưa có' + ', Role: ' + @roleName + N', Address: Không xác định';  -- Nếu không có địa chỉ, hiển thị Không xác định  IF @Address IS NULL  PRINT 'Username: ' + @username + ', Display Name: ' + @displayname + ', Role: ' + @roleName + N', Address: Không xác định';  FETCH NEXT FROM account\_cursor INTO @username, @displayname, @roleName, @Address;  END  CLOSE account\_cursor;  DEALLOCATE account\_cursor; |

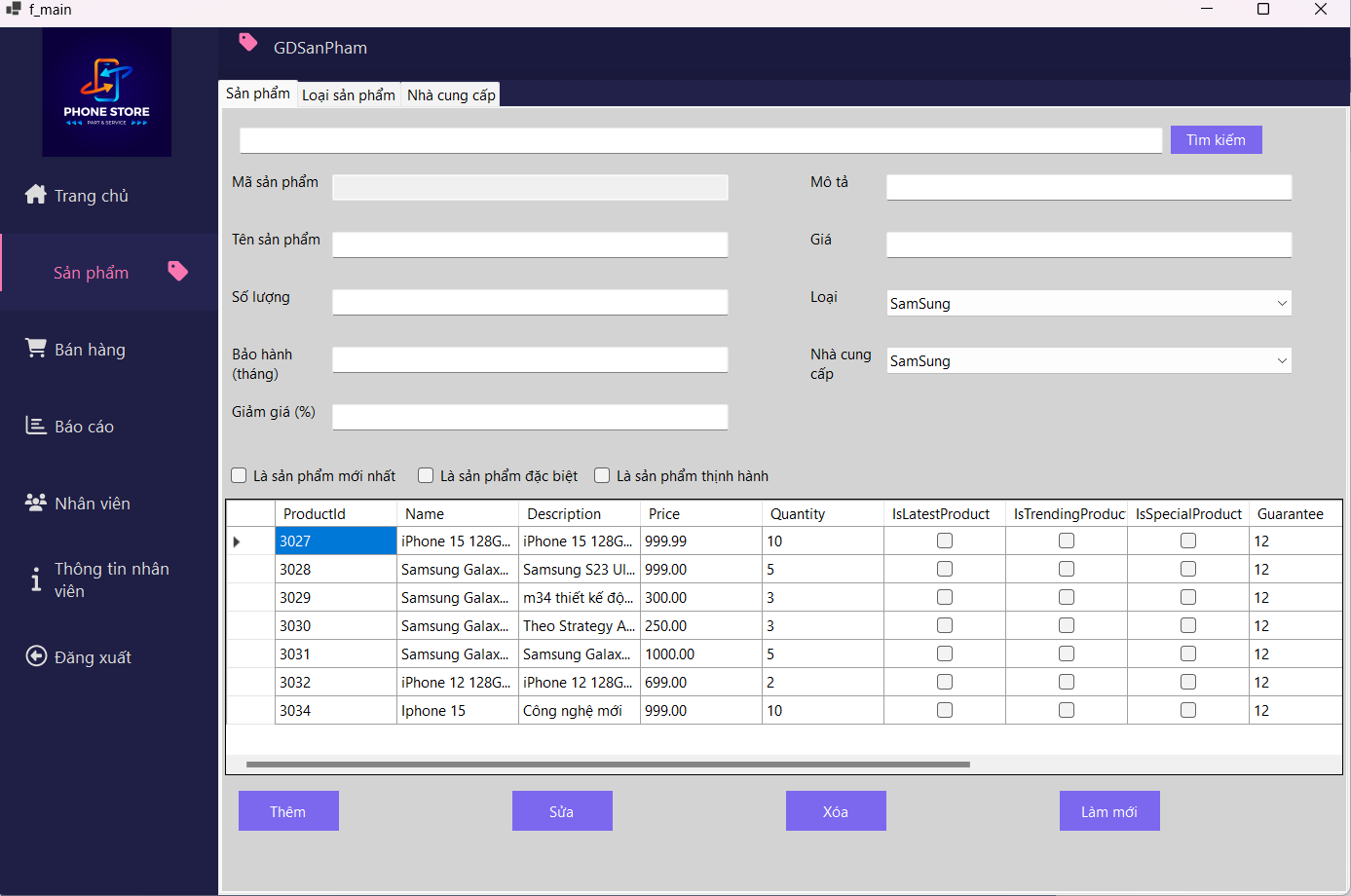
|  |
| --- |
| **Kết quả test** |
|  |

# CÀI ĐẶT ỨNG DỤNG MINH HỌA

## Chức năng danh mục

**Danh mục sản phẩm**

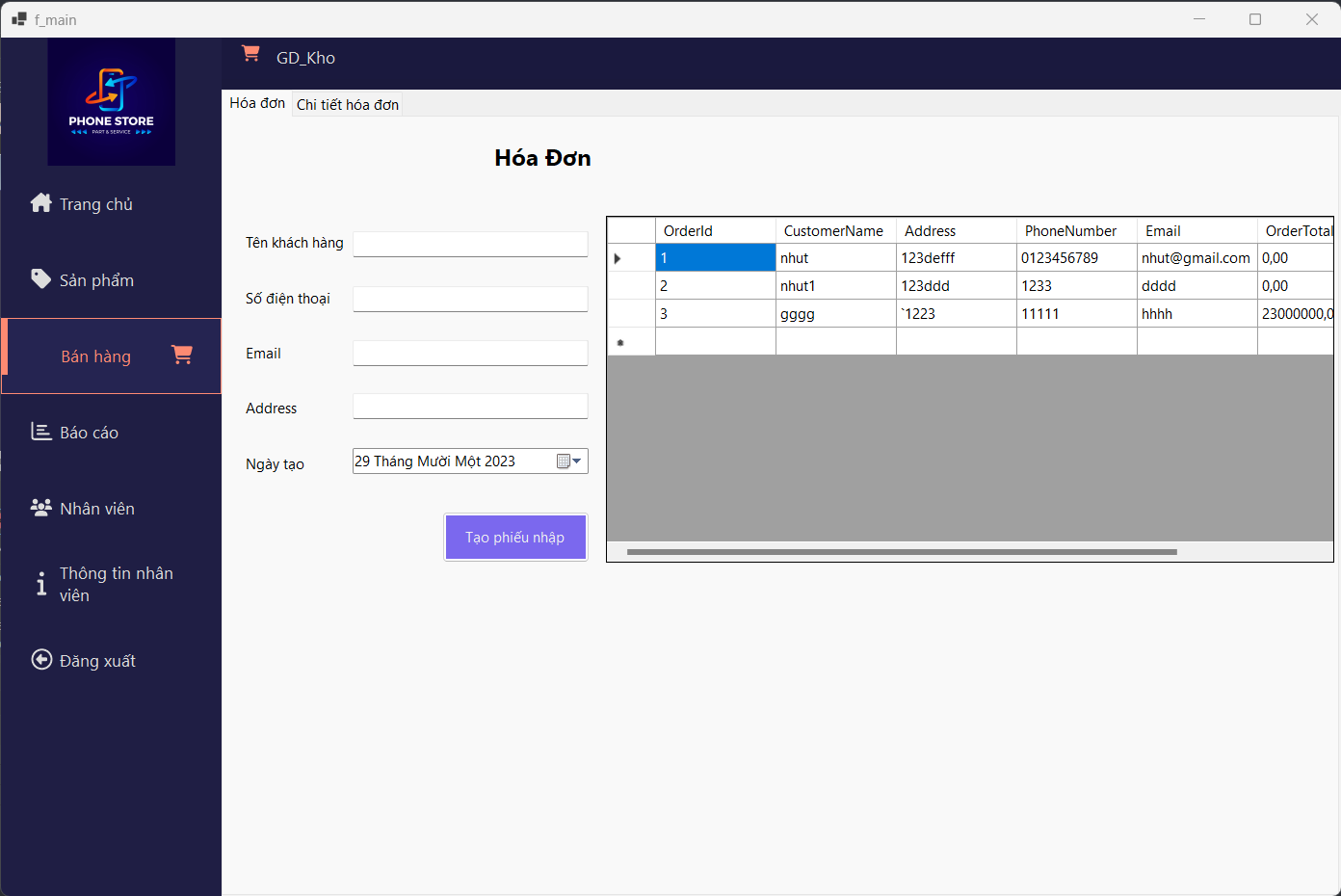
Hiển thị đầy đủ các thông tin về mã sản phẩm , số lượng , giá ,...của mỗi sản phẩm



Ảnh 3.1.Giao diện sản phẩm.

**Danh mục hóa đơn**

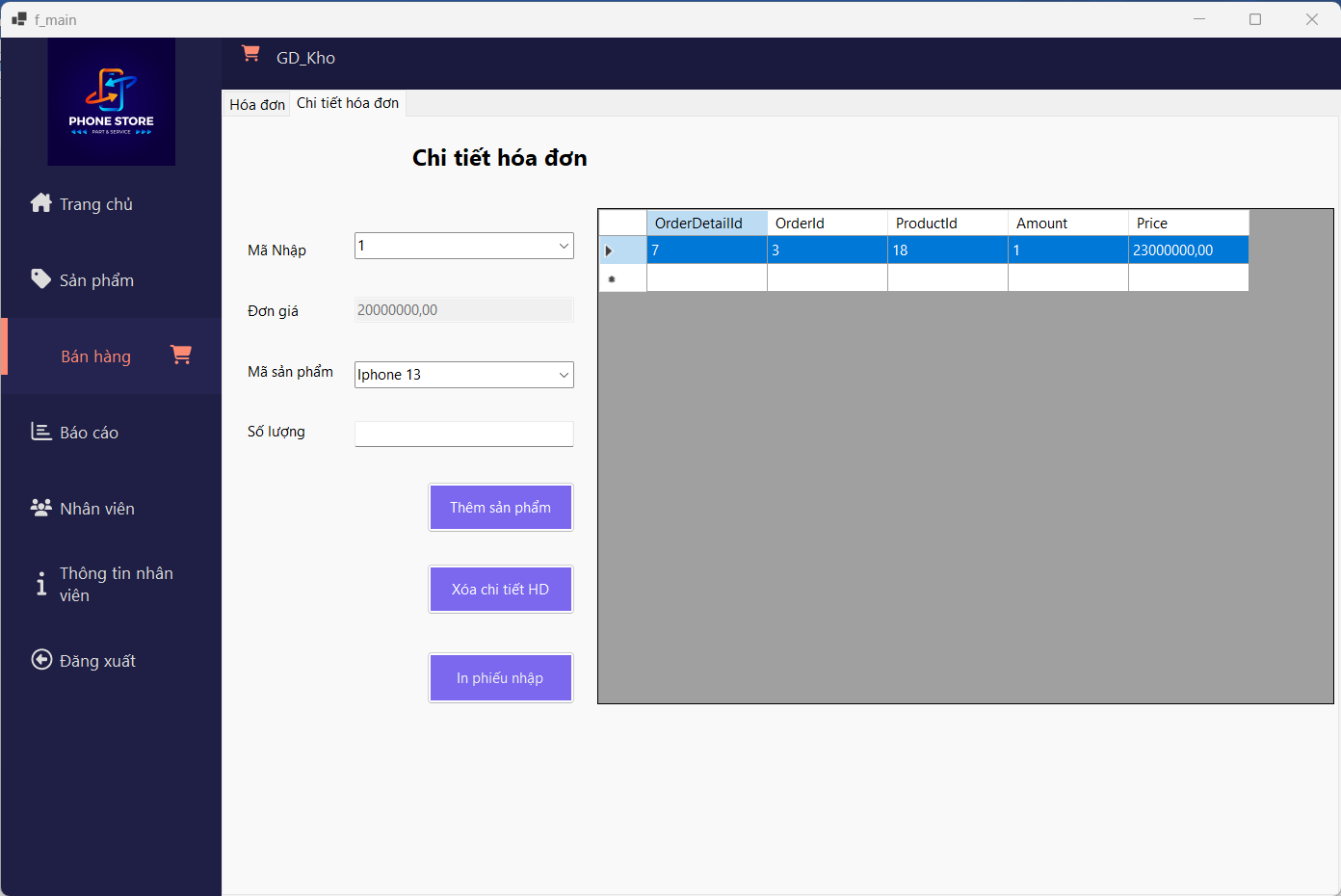
- Hiển thị các thông tin cần thiết trong hóa đơn , mã hóa đơn , địa chỉ , tên nhân viên lập, tên khách hàng , ngày lập và địa chỉ lập



Ảnh 3.2.Giao diện hóa đơn.

**Danh mục chi tiết hóa đơn**

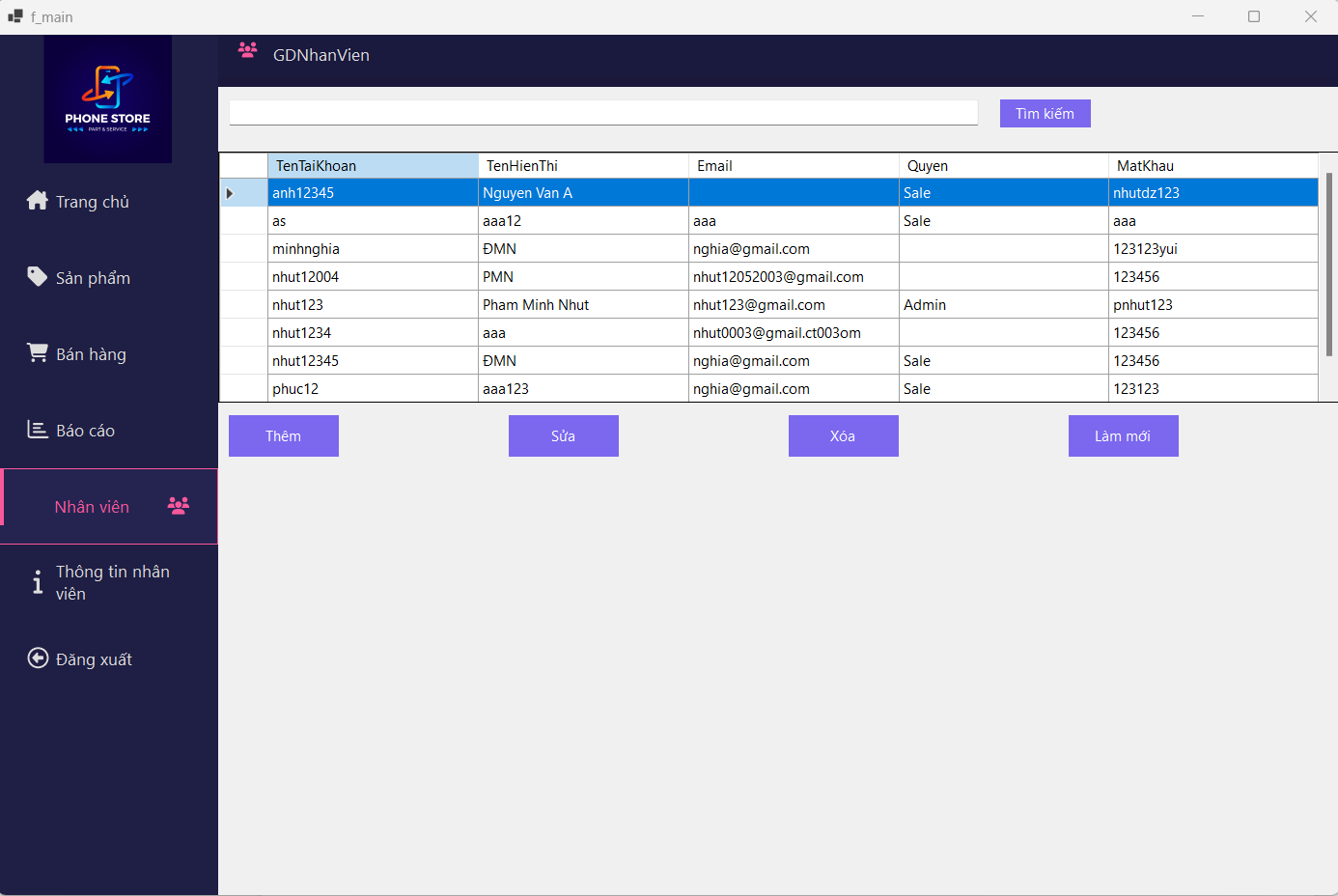
- Hiển thị thông tin mã chi tiết hóa đơn , mã hóa đơn tương ứng với hóa đơn đó , mã sản phẩm bán , số lượng , giá tiền của sản phẩm.



Ảnh 3.3.Giao diện chi tiết hóa đơn.

**Danh mục nhân viên**

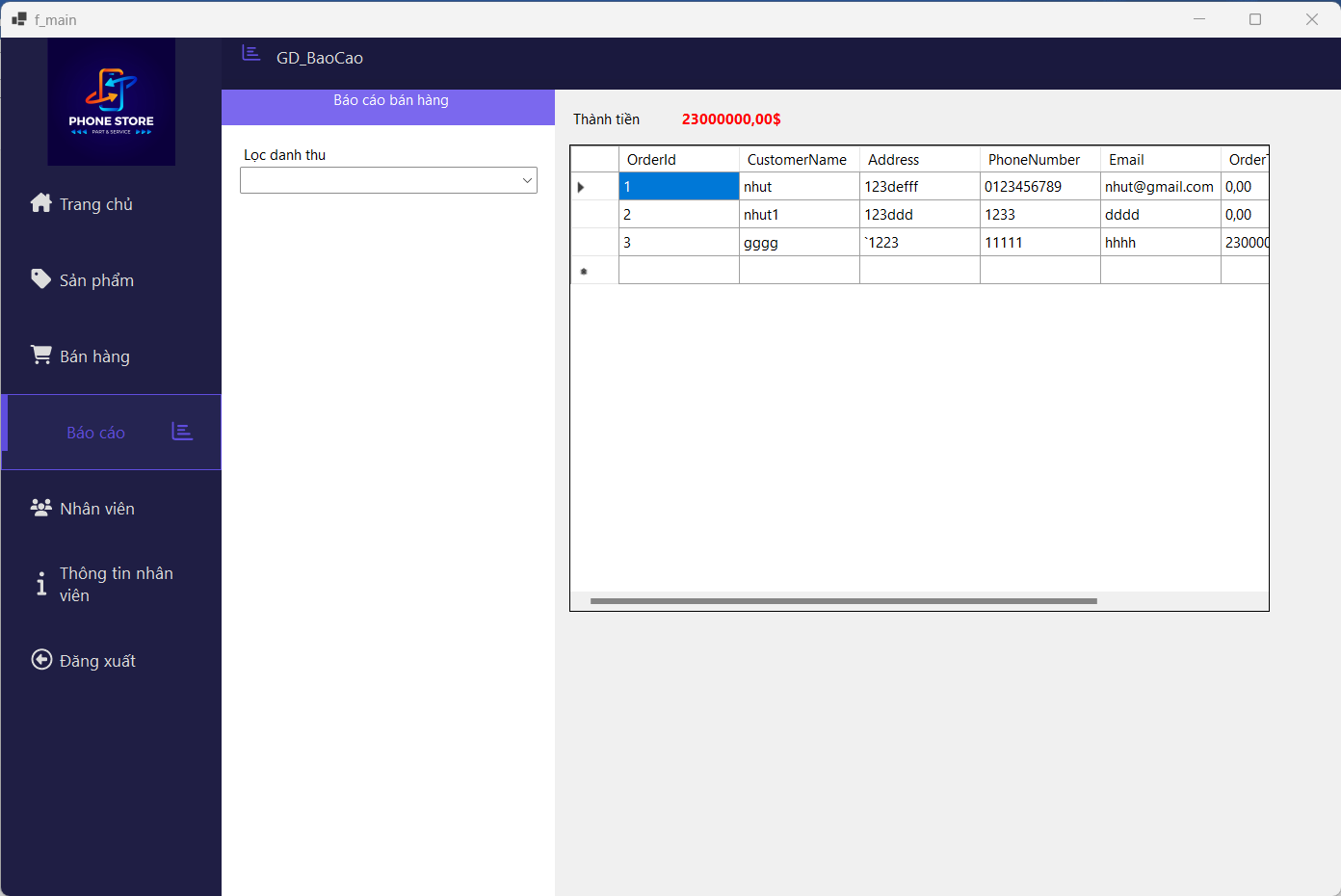
- Hiển thị các thông tin của nhân viên trong cửa hàng , tên hiển thị , số điện thoại , email , nhóm quyền của nhân viên , và mật khẩu , cho việc cấp lại tài khoản



Ảnh 3.4.Giao diện nhân viên.

## Chức năng thống kê

Ở giao diện này người dùng có thể lựa chọn thống kế tiền bán được của cửa hàng theo ngày , tháng , nằm bằng việc lựa chọn thông tin tương ứng trên combobox , sau khi lựa chọn , danh sách hóa đơn sẽ hiện thị ra và tổng tiền bán được theo thời gian lựa chọn.



Ảnh 3.5.Giao diện thống kê.

## Chức năng tìm kiếm

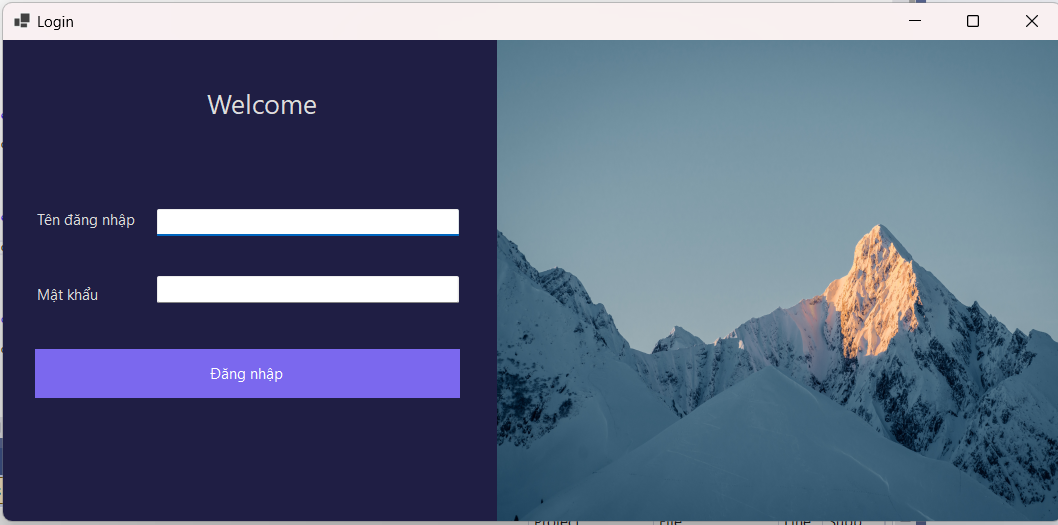
Ở chúc năng này , phần mềm hỗ trợ người dùng tìm kiếm sản phẩm theo tên .

- Tìm kiếm nhân viên , khách hàng . Hỗ trợ các tác vụ luên quan đến tìm kiếm.

## Chức năng nghiệp vụ

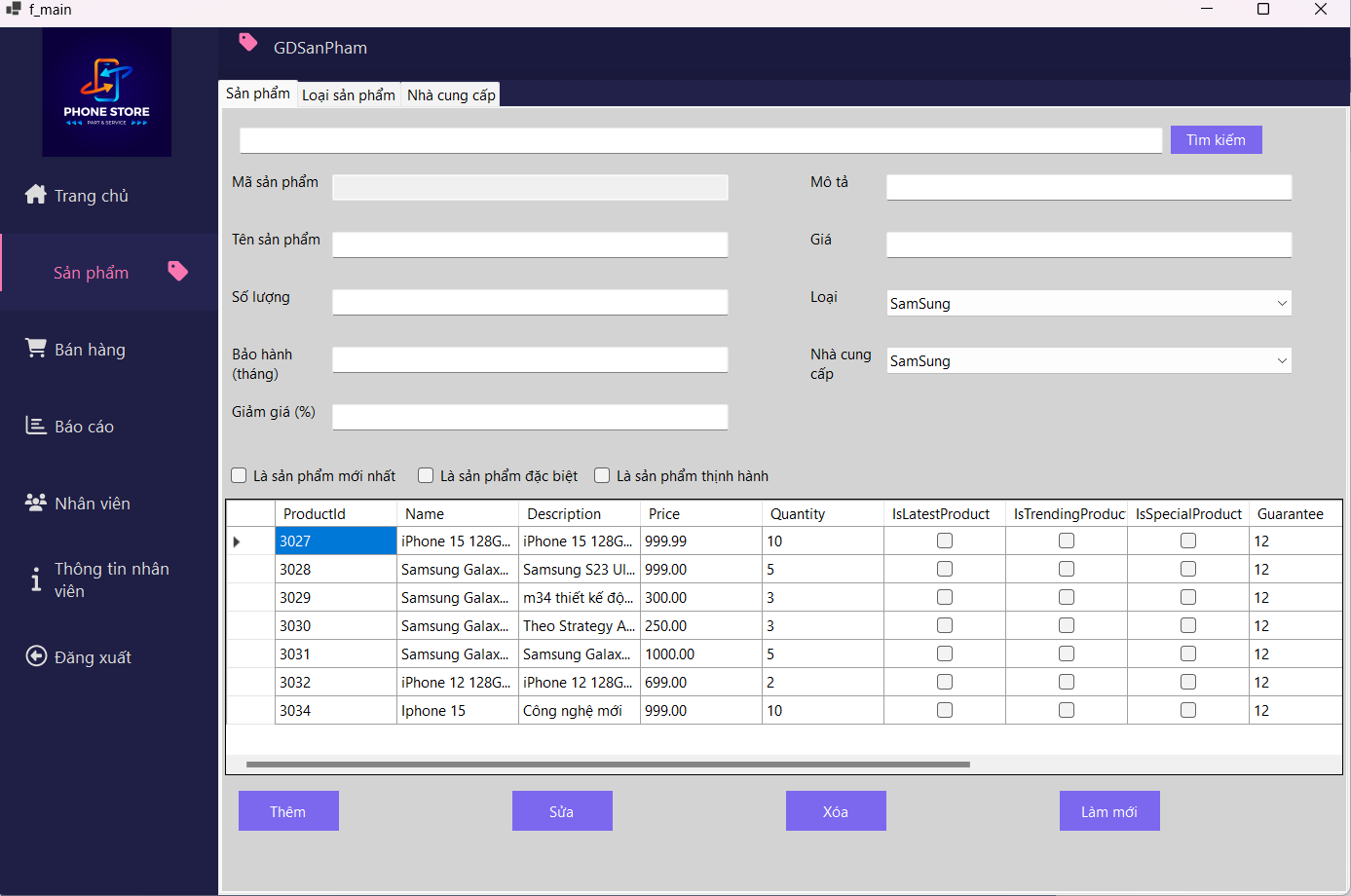
**Login**

Nhân viên nhập mật khẩu do chủ cửa hàng cung cấp.



Ảnh 3.6.Giao diện login.

**Thêm , xóa , sửa sản phẩm**



Ảnh 3.7.Thêm sản phẩm.

**Quản lí nhân viên**

Ở tác vụ này chỉ có người có quyền admin mới được phép truy cập .

Người dùng có thể thực hiện , thêm xóa sửa , cấp lại mật khẩu cho nhân viên.

Đổi thông tin nhân viên

Đổi mật khẩu

# TỔNG KẾT

## Ưu điểm

- Phần mềm trực quan , dễ thao tác , quản lí.

- Dữ liệu được tự động cập nhật khi thêm xóa , sửa.

## Khuyết điểm

- Hệ thống cơ sở dữ liệu có vài chỗ còn sai sót .

- Tính bảo mật chưa cao : chưa mã hóa mật khẩu khi đẩy từ trên phần mềm xuống.

## Hướng phát triển

- Tiếp tục nghiên cứu , phát triển để hoàn thiện phần mềm , cơ sở dữ liệu hơn trong tương lai .

- Xây dựng hệ thống có tính bảo mật hơn.

- Đề ra các giải pháp về sao lưu phục hồi khi mất mát dữ liệu.