



Nhom 6 QLMP-1 - hệ quản trị cơ sở dữ liệu tài liệu tham khảo về bài tập nhóm của sinh viên đh

Cơ sở Dữ Liệu (Đại học Thủy lợi)



Scan to open on Studeersnel

TRƯỜNG ĐẠI HỌC THỦY LỢI

KHOA CÔNG NGHỆ THÔNG TIN



BÀI TẬP LỚN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

NHÓM 6

QUẢN LÍ QUẢN LÝ CỦA HÀNG MỸ PHẨM

Giảng viên: T.S Lê Thị Tú Kiên

Nhóm sinh viên:

Nguyễn Thị Ngọc Ánh	- 2251161950
Hà Văn Đạt	- 2251161964
Hà Thị Phương Linh	- 2251162053
Trần Quang Linh	- 2251162058
Nguyễn Thị Trang	- 2251162180

Hà Nội, Tháng 11 năm 2024

LỜI NÓI ĐẦU

Trong bối cảnh nền kinh tế phát triển không ngừng, ngành mỹ phẩm ngày càng trở thành một lĩnh vực hấp dẫn và đầy tiềm năng. Sự gia tăng nhu cầu về sản phẩm chăm sóc sắc đẹp không chỉ xuất phát từ yếu tố thẩm mỹ mà còn từ ý thức chăm sóc bản thân ngày càng cao của người tiêu dùng. Trước xu hướng này, việc quản lý hiệu quả một cửa hàng mỹ phẩm trở nên vô cùng quan trọng để đáp ứng nhu cầu thị trường và tạo dựng thương hiệu vững mạnh.

Bài tập lớn này nhằm xây dựng ***Mô hình quản lý cho một cửa hàng mỹ phẩm***, từ khâu lựa chọn sản phẩm, marketing, đến quản lý kho và chăm sóc khách hàng. Chúng tôi sẽ phân tích những yếu tố quan trọng giúp tối ưu hóa hoạt động kinh doanh, tăng cường sự hài lòng của khách hàng và tạo ra lợi nhuận bền vững.

Thông qua việc áp dụng lý thuyết và thực tiễn, chúng tôi hy vọng sẽ mang đến những giải pháp hữu ích và thiết thực, góp phần nâng cao hiệu quả quản lý cho cửa hàng mỹ phẩm trong tương lai.

MỤC LỤC

LỜI NÓI ĐẦU.....	1
CHƯƠNG I: MÔ TẢ VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU.....	3
1.1.1. Mô tả bài toán.....	3
1.1.2 Các đối tượng quản lý.....	4
 1.2. THIẾT KẾ BIỂU ĐỒ THỰC THỂ LIÊN KẾT.....	5
1.2.1. Xác định các thực thể.....	5
1.2.2. Mô hình thực thể liên kết:.....	6
 1.3. ÁNH XẠ CƠ SỞ DỮ LIỆU QUAN HỆ.....	7
 1.4. MÔ HÌNH QUAN HỆ ERD.....	13
CHƯƠNG II : XÂY DỰNG CÁC ĐỐI TƯỢNG DỮ LIỆU.....	14
 2.1. Xây dựng View.....	14
 2.2 Xây dựng Procedure.....	17
 2.3. Xây dựng Function.....	20
 2.4. Xây dựng Trigger.....	24
CHƯƠNG III: ỨNG DỤNG CÁC ĐỐI TƯỢNG VÀO PHẦN MỀM QUẢN LÝ.....	28
 3.1. Mô tả chức năng phần mềm quản lý.....	28
3.1.1. Chức năng đăng nhập.....	28
3.1.2 Chức năng của Admin.....	29
3.1.2.1: Chức năng quản lý nhân viên:.....	29
3.1.2.2: Chức năng quản lý nhà cung cấp:.....	30
3.1.2.3: Chức năng quản lý người giao hàng:.....	31
3.1.3: Chức năng của nhân viên:.....	32
3.1.3.1 Màn hình tổng quan trang nhân viên.....	32
3.1.3.2. Chức năng quản lý sản phẩm:.....	33
3.1.3.3. Chức năng quản lý loại sản phẩm (category).....	34
3.1.3.4. Chức năng xem thông tin đơn hàng của từng khách hàng.....	35
3.1.3.5. Chức năng xem thông tin chi tiết đơn hàng và người giao hàng.....	36
3.1.3.6. Chức năng tạo đơn hàng.....	36
 3.2 Mô tả ứng dụng một số đối tượng dữ liệu trong phần mềm.....	37
3.2.1 View:.....	37
3.2.2 Procedure:.....	37
3.2.3 Trigger:.....	38

CHƯƠNG I: MÔ TẢ VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU

1.1.1. Mô tả bài toán

Quản lý cửa hàng mỹ phẩm là một trong những yếu tố quan trọng quyết định sự thành công của một cửa hàng trong lĩnh vực bán lẻ mỹ phẩm . Bài toán này bao gồm việc tối ưu hoá không chỉ việc quản lý sản phẩm mỹ phẩm mà còn cả trải nghiệm mua sắm của khách hàng .

Quản lý cửa hàng mỹ phẩm đòi hỏi việc hiểu rõ về xu hướng thị trường, sở thích của khách hàng , và khả năng cung ứng sản phẩm. Đồng thời cần phải quản lý kho hàng một cách hiệu quả để đảm bảo sự đa dạng và đủ số lượng sản phẩm, đồng thời tối ưu hoá chi phí lưu trữ.

Việc quản lý nhân viên cửa hàng mỹ phẩm là một yếu tố quan trọng từ việc đào tạo nhân viên, quản lý hiệu suất làm việc , đến tạo môi trường làm việc tích cực.Với sự phát triển của thương mại điện tử , bài toán quản lý cửa hàng mỹ phẩm cũng bao gồm việc tích hợp kênh bán hàng trực tuyến và offline một cách hợp nhất để tối ưu hóa trải nghiệm mua sắm của khách hàng

Quản lý cửa hàng mỹ phẩm không chỉ là việc bán sản phẩm mà còn là việc xây dựng mối quan hệ lâu dài với khách hàng thông qua dịch vụ tốt và trải nghiệm mua sắm đáng nhớ . Để thành công , cần có chiến lược quản lý toàn diện , linh hoạt và sáng tạo để đáp ứng nhu cầu đa dạng của khách hàng trong ngành công nghiệp mỹ phẩm ngày nay.

Bài toán đưa ra yêu cầu xây dựng cơ sở dữ liệu quản lý mỹ phẩm giải quyết được các vấn đề về quản lý nhân viên, quản lý khách hàng, quản lý các sản phẩm, quản lý hàng tồn kho, quản lý các loại hàng, quản lý hóa đơn

Tìm hiểu quy tắc nghiệp vụ, các yêu cầu đặt ra của đề bài. Các ngôn ngữ lập trình và cơ sở dữ liệu. Bắt tay vào xây dựng hệ thống . Tiến hành kiểm tra và chạy thử. Thay đổi.

1.1.2 Các đối tượng quản lý

1. Khách hàng (Customers)

Mỗi khách hàng có một mã khách hàng duy nhất.

Khách hàng có thông tin cá nhân như: mã khách hàng, tên, khách hàng, họ khách hàng, địa chỉ, điện thoại, email, thành phố, mã bưu điện, và số fax.

2. Nhân viên (Employees)

Mỗi nhân viên có một mã nhân viên duy nhất.

Nhân viên có thông tin cá nhân như: mã nhân viên, họ nhân viên, tên nhân viên, ngày sinh, địa chỉ, điện thoại, email, thành phố, mã bưu điện, số fax, điện thoại nhà riêng, và ngày bắt đầu làm việc.

3. Đơn hàng (Orders)

Mỗi đơn hàng có một mã đơn hàng duy nhất.

Đơn hàng liên kết với mã khách hàng, mã nhân viên và mã vận chuyển .

Đơn hàng có thông tin: mã đơn hàng, mã khách hàng, mã nhân viên, ngày đặt hàng, ngày yêu cầu, ngày giao hàng, địa chỉ giao hàng, thành phố giao hàng, phương thức giao hàng và mã bưu điện giao hàng.

Mỗi đơn hàng chỉ được liên kết với một khách hàng, một nhân viên, và một người vận chuyển duy nhất.

4. Nhà cung cấp (Suppliers)

Mỗi nhà cung cấp có một mã nhà cung cấp duy nhất .

Nhà cung cấp có thông tin: mã nhà cung cấp, tên công ty, tên người liên hệ, địa chỉ, thành phố, mã bưu điện, điện thoại, trang chủ, và số fax.

5. Loại hàng (Categories)

Mỗi loại hàng có một mã loại hàng duy nhất.

Loại hàng có thông tin: mã thẻ loại, tên thẻ loại và mô tả.

6. Sản phẩm (Products)

Mỗi sản phẩm có một mã sản phẩm duy nhất .

Sản phẩm có thông tin: mã sản phẩm, mã thẻ loại, mã nhà cung cấp, tên sản phẩm và giá bán, số lượng trong kho, kích thước, màu sắc.

7. Chi tiết đơn hàng (OrderDetails)

Chi tiết đơn hàng có thông tin về: mã hóa đơn, mã sản phẩm, giá từng sản phẩm, số lượng, giảm giá

8. Người vận chuyển (Shippers)

Mỗi người vận chuyển có một mã vận chuyển duy nhất.

Người vận chuyển có thông tin gồm , mã giao hàng, tên công ty và số điện thoại.

1.2. THIẾT KẾ BIỂU ĐỒ THỰC THẾ LIÊN KẾT

1.2.1. Xác định các thực thể

Suppliers (Supplier_ID, CompanyName, ContactName, Address, City, PostalCode, Phone, HomePage, Fax),

Products (Product_ID, Category_ID, Supplier_ID ,ProductName, UnitPrice, Size, Color, Quantity),

Categories (Category_ID, CategoryName, Description),

Oders_Products (Order_ID, Product_ID, UnitPrice, Quantity, Discout),

Orders (Order_ID, Customer_ID, Employee_ID, OrderDate, RequiredDate, ShippedDate, ShipVia, ShipAddress, ShipCity, ShipPostalCode),

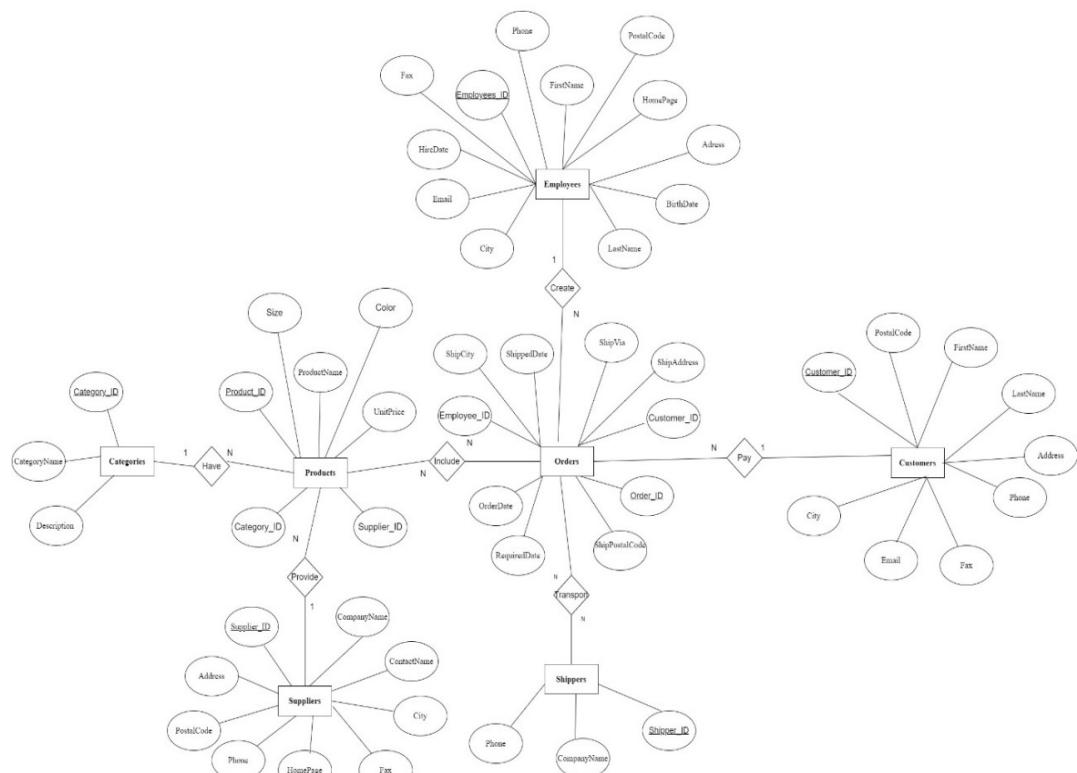
Employees (Employees_ID, FirstName, LastName, BirthDate, Adress, Phone, Email, City, PostalCode, Fax, HomePage, HireDate),

Customers (Customer_ID, FirstName, LastName, Address, Phone, Email, City, PostalCode, Fax),

Shippers (Shipper_ID, CompanyName, Phone),

Shippers_Orders (Shipper_ID, Order_ID, Date, Status)

1.2.2. Mô hình thực thể liên kết:



1.3. ÁNH XÃ CƠ SỞ DỮ LIỆU QUAN HỆ

**Bảng Customers*

Thuộc tính	Kiểu DL	Độ dài	Mô tả
🔑 Customer_ID	int		Mã khách hàng
FirstName	nvarchar	50	Tên khách hàng
LastName	nvarchar	50	Họ khách hàng
Address	nvarchar	255	Địa chỉ khách hàng
Phone	varchar	15	Sđt khách hàng
Email	Varchar	100	Email khách
City	nvarchar	50	Thành phố khách hàng sinh sống
Postacode	Varchar	10	Mã bưu điện của khách hàng
Fax	Varchar	20	Số máy fax của khách hàng

*Bảng Employees

Thuộc tính	Kiểu DL	Độ dài	Mô tả
🔑 Employee_ID	int		Mã nhân viên
FirstName	nvarchar	50	Tên nhân viên
LastName	nvarchar	50	Họ nhân viên
Address	nvarchar	255	Địa chỉ nhân viên
Phone	varchar	15	Số điện thoại nhân viên
Email	varchar	100	Email nhân viên
BirthDate	date		Ngày sinh của nhân viên
City	nvarchar	50	Thành phố nhân viên sinh sống
PastaCode	varchar	10	Mã bưu điện của nhân viên
Fax	varchar	20	Số máy fax của nhân viên
HireDate	date		Ngày bắt đầu làm việc của nhân viên

**Bảng Orders*

Thuộc tính	Kiểu DL	Độ dài	Mô tả
🔑 Order_ID	int		Mã hoá đơn
🔑 Customer_ID	Int		Mã khách hàng
🔑 Employee_ID	int		Mã nhân viên
OrderDate	Date		Ngày đặt hàng
RequiredDate	date		Ngày yêu cầu
ShippedDate	date		Ngày giao hàng
ShipVia	int		Phương thức vận chuyển
ShipAddress	nvarchar	255	Nơi giao hàng
ShipCity	nvarchar	50	Thành phố giao hàng
ShipPostalCode	Varchar	10	Mã bưu điện giao hàng

**Bảng Suppliers*

Thuộc tính	Kiểu DL	Độ dài	Mô tả
-------------------	----------------	---------------	--------------

🔑 Supplier_ID	int		Mã công ty
CompanyName	nvarchar	100	Tên công ty phân phối hàng
ContactName	nvarchar	50	Tên giao dịch hàng hoá
Address	nvarchar	255	Địa chỉ phân phối
City	nvarchar	50	Thành phố phân phối
PostalCode	nvarchar	10	Mã bưu điện của địa chỉ nhà phân phối
Phone	Varchar	15	Số điện thoại nhà phân phối
HomePage	nvarchar	100	Page nhà cung cấp
Fax	Varchar	20	Số mã fax của nhà cung cấp

*Bảng Categories

Thuộc tính	Kiểu DL	Độ dài	Mô Tả
------------	---------	--------	-------

🔑 Category_ID	int		Mã loại hàng
CategoryName	nvarchar	100	Tên loại hàng
Description	text	255	Mô tả thể loại

**Bảng Products*

Thuộc tính	Kiểu DL	Độ dài	Mô Tả
🔑 Product_ID	int		Mã sản phẩm
🔑 Category_ID	int		Mã thể loại
🔑 Supplier_ID	int		Mã nhà cung cấp
ProductName	nvarchar	100	Tên sản phẩm
UnitPrice	decimal		Giá từng sản phẩm
Quantity	int		Số lượng
Size	nvarchar	50	Kích thước
Color	nvarchar	50	Màu sắc

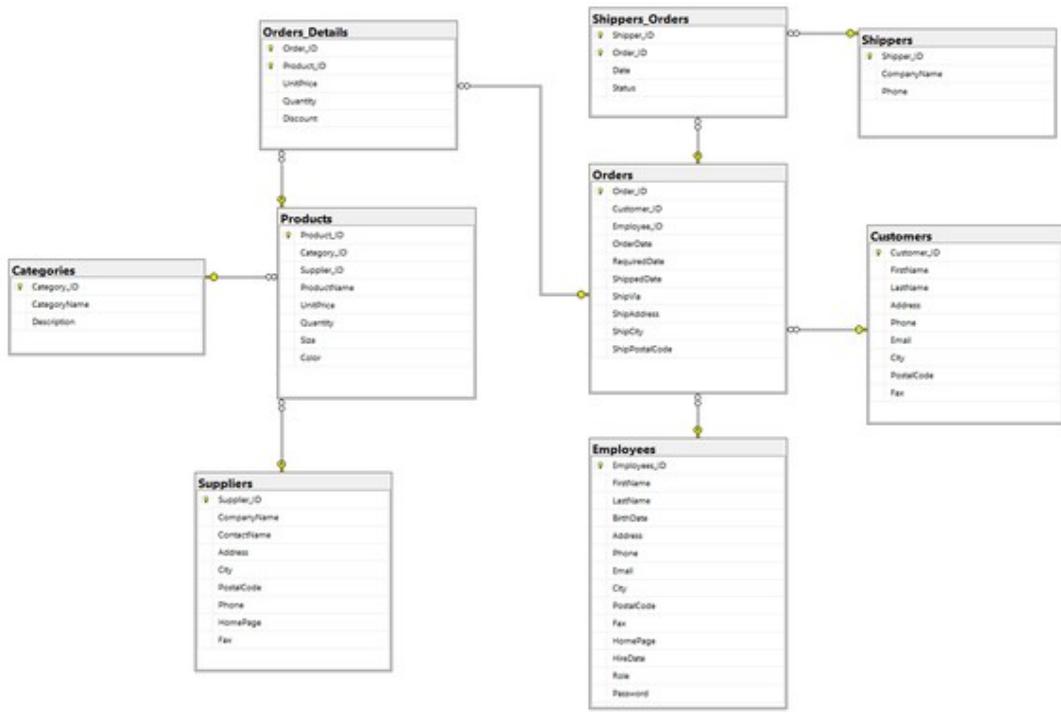
**Bảng Orders_Details*

Thuộc tính	Kiểu DL	Độ dài	Mô Tả
🔑 Order_ID	int		Mã hóa đơn
🔑 Product_ID	int		Mã sản phẩm
UnitPrice	int	-	Giá từng sản phẩm
Quantity	varchar	-	Số lượng
Discount			Giảm giá

**Bảng Shippers*

Thuộc tính	Kiểu DL	Độ dài	Mô Tả
🔑 Shipper_ID	int		Mã giao hàng
CompanyName	nvarchar	100	Tên công ty
Phone	varchar	15	Số điện thoại

1.4. MÔ HÌNH QUAN HỆ ERD



CHƯƠNG II : XÂY DỰNG CÁC ĐỐI TƯỢNG DỮ LIỆU

2.1. Xây dựng View

2.1.1. View của Ngọc Ánh

Mục đích: cung cấp thuận tiện để truy cập và hiển thị thông tin sản phẩm và danh mục sản phẩm được kết hợp từ 2 bảng product và category

```
CREATE VIEW Categories_View AS
SELECT
    P.Product_ID,
    P.ProductName,
    P.UnitPrice,
    P.Quantity,
    P.Color,
    P.Size,
    C.CategoryName,
    C.Description
FROM Products P
JOIN Categories C ON P.Category_ID = C.Category_ID;
```

2.1.2. View Hà Văn Đạt

Mục đích: Hiển thị thông tin bán hàng về danh sách các sản phẩm có giá lớn hơn 10 được bán từ 05/01/2023 đến 10/01/2023

```

drop view SalesReport
create view SalesReport
as
select p.ProductName as N'Tên sản phẩm',
       cate.CategoryName as N'Thể loại',
       p.UnitPrice as N'Giá bán',
       p.Quantity as N'Số lượng hàng trong kho',
       od.Quantity as N'Số lượng bán',
       (c.FirstName + ' ' + c.LastName) as N'Họ tên khách hàng',
       (o.ShipAddress ) as N'Địa chỉ giao hàng chi tiết',
       (od.Quantity * od.UnitPrice ) as N'Tổng tiền của đơn hàng trước giảm giá',
       (od.Discount) as N'Giảm giá',
       ((od.Quantity * od.UnitPrice ) * (od.Discount)) as N'Tổng tiền giảm',
       (od.Quantity * od.UnitPrice - ((od.Quantity * od.UnitPrice ) * (od.Discount))) as N'Tổng tiền của đơn hàng sau khi giảm giá',
       (e.FirstName + ' ' + e.LastName) as N'Nhân viên bán hàng',
       o.OrderDate as N'Ngày đặt hàng',
       s.CompanyName as N'Công ty vận chuyển',
       so.Status as N'Trạng thái đơn hàng'
from Products as p
join Orders_Details as od on p.Product_ID= od.Product_ID
join Orders as o on od.Order_ID = o.Order_ID
join Customers as c on o.Customer_ID = c.Customer_ID
join Employees as e on o.Employee_ID = e.Employees_ID
join Suppliers as su on p.Supplier_ID = su.Supplier_ID
join Shippers_Orders as so on o.ShipVia = so.Shipper_ID
join Shippers as s on so.Shipper_ID = s.Shipper_ID
join Categories as cate on p.Category_ID = cate.Category_ID
where p.UnitPrice > 10 and o.OrderDate between '2023-01-05' and '2023-01-10'
group by p.ProductName , cate.CategoryName, p.UnitPrice , p.Quantity,od.Quantity,(c.FirstName + ' ' + c.LastName),o.ShipAddress,s.CompanyName,
so.Status ,(e.FirstName + ' ' + e.LastName),o.OrderDate,(od.Discount),(od.Quantity * od.UnitPrice )

```

2.1.3. View của Phuong Linh

Mục đích: dễ dàng tra cứu các đơn hàng cùng với thông tin vận chuyển từ các bảng dữ liệu gốc mà không cần phải viết câu lệnh JOIN phức tạp mỗi lần truy vấn.

--STT 28_Ha Thi Phuong Linh.
--Ý tưởng : Liệt kê thông tin hóa đơn và người vận chuyển.
---View vw_ShipperWithOrder tổng hợp dữ liệu từ các bảng Orders, Shippers, và Shipper_Orders.
---Mục đích: Sử dụng để báo cáo việc giao hàng, bao gồm cách thức vận chuyển, thời gian giao hàng và trạng thái đơn hàng.
---đơn giản hóa và tối ưu hóa quá trình truy vấn dữ liệu về đơn hàng và công ty vận chuyển, giúp tiết kiệm thời gian,
---công sức và cải thiện hiệu suất khi làm việc với dữ liệu lớn.

```

CREATE VIEW vw_ShipperWithOrder
AS
SELECT DISTINCT
    o.Customer_ID,
    o.Employee_ID,
    o.Order_ID,
    o.OrderDate,
    o.ShipAddress,
    o.ShipCity,
    o.ShipPostalCode,
    s.CompanyName AS ShipperCompany,
    so.Shipper_ID,
    so.Date AS ShippingDate,
    so.Status AS ShippingStatus,
CASE
    WHEN o.ShipVia = 1 THEN 'Road'
    WHEN o.ShipVia = 2 THEN 'Airline'
    ELSE 'Train'
END AS 'Transport'

FROM Shippers_Orders so JOIN Shippers s on so.Shipper_ID = s.Shipper_ID
JOIN Orders o on so.Order_ID = o.Order_ID

SELECT * FROM vw_ShipperWithOrder

```

2.1.4. View của Quang Linh

Mục tiêu: cung cấp một bảng dữ liệu dễ truy vấn và báo cáo về doanh số bán hàng theo khách hàng, giúp quản lý hoặc bộ phận bán hàng có thể nhanh chóng xem được tổng số tiền đã chi trả bởi mỗi khách hàng.

```
CREATE VIEW vw_CustomerSales AS
SELECT
    c.FirstName + ' ' + c.LastName AS CustomerName,
    SUM(od.Quantity * od.UnitPrice * (1 - od.Discount)) AS TotalAmount
FROM Customers c
JOIN Orders o ON c.Customer_ID = o.Customer_ID
JOIN Orders_Details od ON o.Order_ID = od.Order_ID
GROUP BY c.FirstName, c.LastName;
```

2.1.5. View của Trang

Mục đích: Lọc ra các khách hàng có tên bắt đầu bằng chữ "A" một cách tự động và nhanh chóng, giúp tiết kiệm thời gian so với việc viết lại câu truy vấn nhiều lần.

```
--VIEW lọc những khách hàng đã từng mua hàng có FirstName bắt đầu bằng A

CREATE VIEW CustomersWithA AS
SELECT DISTINCT c.Customer_ID, c.FirstName, c.LastName, c.Address, c.Phone, c.Email, c.City, c.PostalCode, c.Fax
FROM Customers c
JOIN Orders o ON c.Customer_ID = o.Customer_ID
JOIN Orders_Details od ON o.Order_ID = od.Order_ID
JOIN Products p ON od.Product_ID = p.Product_ID
WHERE c.FirstName LIKE 'A%';

go
SELECT * FROM CustomersWithA;

SELECT * FROM Customers

--mỗi 1 thành phố có bnh khách hàng
SELECT City, count(Customer_ID) as countCustomer
from CustomersWithA
group by City
```

2.2 Xây dựng Procedure

2.2.1. Procedure của Ngọc Ánh

Mục đích: tìm kiếm thông tin sản phẩm trong

view Categories View dựa trên tên sản phẩm hoặc chuỗi sản phẩm được truyền vào qua @productname

```
CREATE PROCEDURE searchproduct
    @productname NVARCHAR(50)
AS
BEGIN
    SELECT *
    FROM Categories_View
    WHERE ProductName LIKE '%' + @productname + '%';
END
EXEC searchproduct @productname = 'cream';
```

2.2.2. Procedure của Hà Văn Đạt

Mục đích: Cập nhật số lượng cho sản phẩm trong kho

```

create proc UpdateQuantityStock
    @Product_ID nvarchar(50) = null ,
    @Quantity_add nvarchar(50) = null [as]
begin
    begin try
        if @Product_ID is null and @Quantity_add is null
        begin
            print(N'Hãy nhập mã sản phẩm muốn thêm số lượng và số lượng muốn thêm !!!')
            return
        end
        else if @Product_ID is null
        begin
            print(N'Vui lòng cung cấp mã sản phẩm ')
            return
        end
        else if @Quantity_add is null
        begin
            print(N'Vui lòng nhập số lượng muốn thêm')
            return
        end
        if try_cast(@Product_ID as int) is null
        begin
            print(N'Mã sản phẩm phải là một số nguyên.')
            return
        end
        if try_cast(@Quantity_add as int) is null
        begin
            print(N'Số lượng cần thêm phải là một số nguyên hợp lệ.')
            return
        end
        declare @Product_ID_int int = cast(@Product_ID as int)
        declare @Quantity_add_int int = cast(@Quantity_add as int)

        if not exists (select * from Products where Product_ID = @Product_ID)
        begin
            print(N'Sản phẩm không tồn tại !')
            return
        end
        if (@Quantity_add_int <= 0)
        begin
            Print(N'Số lượng cần thêm phải lớn hơn 0 !')
            return
        end
        else
        begin
            declare @NewQuantity INT;
            select @NewQuantity = Quantity + @Quantity_add_int
            from Products where Product_ID = @Product_ID_int

            if @NewQuantity > 500
            begin
                print(N'Số lượng mới vượt quá mức tối đa cho phép.')
                return;
            end
            end
        UPDATE Products
        set Quantity = @NewQuantity
        where Product_ID = @Product_ID_int
        print(N'Cập nhật số lượng sản phẩm thành công')
    end try
    begin catch
        print error_message()
    end catch
end

```

2.2.3. Procedure của Phương Linh

Mục đích: Lấy thông tin về đơn hàng và nhà vận chuyển dựa trên các tham số đầu vào. Cho phép người dùng tìm kiếm các đơn hàng theo từng nhà vận chuyển cụ thể hoặc theo trạng thái của đơn hàng

```
CREATE PROCEDURE sp_ShipperWithOrder
    @ShipperID INT = NULL,
    @Status NVARCHAR(50) = NULL
AS
BEGIN
    SELECT DISTINCT
        o.Customer_ID,
        o.Employee_ID,
        o.OrderDate,
        o.ShipAddress,
        o.ShipCity,
        o.ShipPostalCode,
        s.CompanyName AS ShipperCompany,
        so.Shipper_ID,
        so.Date AS ShippingDate,
        so.Status AS ShippingStatus,
        CASE
            WHEN o.ShipVia = 1 THEN 'Road'
            WHEN o.ShipVia = 2 THEN 'Airline'
            ELSE 'Train'
        END AS Transport
    FROM Shippers_Orders so
    JOIN Shippers s ON so.Shipper_ID = s.Shipper_ID
    JOIN Orders o ON so.Order_ID = o.Order_ID
    WHERE
        (@ShipperID IS NULL OR s.Shipper_ID = @ShipperID)
        AND (@Status IS NULL OR so.Status = @Status);
END;
DROP PROCEDURE sp_ShipperWithOrder
```

2.2.4. Procedure của Quang Linh

Mục tiêu: giúp người dùng dễ dàng xem xét và phân tích doanh thu theo khách hàng dựa trên một phạm vi ngày tháng nhất định, chẳng hạn như trong một tuần, tháng, hoặc năm, tùy thuộc vào giá trị của các tham số ngày truyền vào.

```
CREATE PROCEDURE sp_GetCustomerSalesByDate
    @StartDate DATE,
    @EndDate DATE
AS
BEGIN
    SELECT
        c.FirstName + ' ' + c.LastName AS CustomerName,
        SUM(od.Quantity * od.UnitPrice * (1 - od.Discount)) AS TotalAmount
    FROM Customers c
    JOIN Orders o ON c.Customer_ID = o.Customer_ID
    JOIN Orders_Details od ON o.Order_ID = od.Order_ID
    WHERE o.OrderDate BETWEEN @StartDate AND @EndDate
    GROUP BY c.FirstName, c.LastName;
END;
```

2.2.5. Procedure của Trang

Mục đích: Thực hiện việc cập nhật thông tin khách hàng có Customer_ID là 1 và sau đó kiểm tra xem thông tin của khách hàng này đã được cập nhật hay chưa.

```
--PROCEDURE sửa thông tin khách đã từng mua hàng
CREATE PROCEDURE Update_Customers
    @Customer_ID INT,
    @FirstName NVARCHAR(50),
    @LastName NVARCHAR(50),
    @Address NVARCHAR(100),
    @Phone NVARCHAR(20),
    @Email NVARCHAR(50),
    @City NVARCHAR(50),
    @PostalCode NVARCHAR(20),
    @Fax NVARCHAR(20)
AS
--Định nghĩa phần thân của thủ tục, bắt đầu với BEGIN và kết thúc với END.
BEGIN
    UPDATE Customers --Cập nhật thông tin khách hàng trong bảng Customers dựa trên Customer_ID.
    SET FirstName = @FirstName,
        LastName = @LastName,
        Address = @Address,
        Phone = @Phone,
        Email = @Email,
        City = @City,
        PostalCode = @PostalCode,
        Fax = @Fax
    WHERE Customer_ID = @Customer_ID;
    PRINT N'Thông tin của khách hàng đã được cập nhật thành công.';
END;

--thực thi PROCEDURE
EXEC Update_Customers
    @Customer_ID = 1,
    @FirstName = 'An',
    @LastName = 'Nguyen',
    @Address = '123 street ABC',
    @Phone = '0123456789',
    @Email = 'an.nguyen@example.com',
    @City = 'Hà Nội',
    @PostalCode = '100000',
    @Fax = '0123456789';

--ktra xem Customer_ID = 1 đã dc cập nhật lại hay chưa
SELECT * FROM Customers
where Customer_ID = 1
```

2.3. Xây dựng Function

2.3.1. Function của Ngọc Ánh

Mục đích: trả về một bảng kết quả chứa thông tin về các sản phẩm có giá bán lớn hơn hoặc bằng giá trị được chỉ định

```
CREATE FUNCTION ProductsByPrice
(
    @minPrice DECIMAL(10, 2)
)
RETURNS TABLE
AS
RETURN
(
    SELECT p.Product_ID, p.ProductName, p.UnitPrice, p.Quantity, c.CategoryName, s.CompanyName
    FROM Products p
    JOIN Categories c ON p.Category_ID = c.Category_ID
    JOIN Suppliers s ON p.Supplier_ID = s.Supplier_ID
    WHERE p.UnitPrice >= @minPrice
);
```

2.3.2. Function của Hà Văn Đạt

Mục đích: Trả về khách hàng không có đơn hàng nào

```
CREATE FUNCTION GetCustomersNo_Orders1
    (@Days NVARCHAR(50))
RETURNS @Customers TABLE
(
    Customer_ID INT,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    Address VARCHAR(200),
    Phone VARCHAR(20),
    Email VARCHAR(100)
)
AS
BEGIN
    DECLARE @DaysInt INT;
    SET @DaysInt = TRY_CONVERT(INT, @Days);

    IF @DaysInt IS NULL OR @DaysInt <= 0
    BEGIN
        RETURN;
    END

    INSERT INTO @Customers (Customer_ID, FirstName, LastName, Address, Phone, Email)
    SELECT c.Customer_ID, c.FirstName, c.LastName, c.Address, c.Phone, c.Email
    FROM Customers c
    LEFT JOIN Orders o ON c.Customer_ID = o.Customer_ID
    GROUP BY c.Customer_ID, c.FirstName, c.LastName, c.Address, c.Phone, c.Email
    HAVING COUNT(o.Order_ID) = 0
        OR MAX(o.OrderDate) < DATEADD(DAY, -@DaysInt, GETDATE());

    RETURN;
END
```

2.3.3. Function của Phuong Linh

Mục đích: Là cung cấp một cách dễ dàng và linh hoạt để truy vấn thông tin về các đơn hàng và nhà vận chuyển trong một cấu trúc bảng duy nhất.

```
CREATE FUNCTION fn_ShipperWithOrder ()  
RETURNS TABLE  
AS  
RETURN  
(  
    SELECT DISTINCT  
        o.Customer_ID,  
        o.Employee_ID,  
        o.Order_ID,  
        o.OrderDate,  
        o.ShipAddress,  
        o.ShipCity,  
        o.ShipPostalCode,  
        s.CompanyName AS ShipperCompany,  
        so.Shipper_ID,  
        so.Date AS ShippingDate,  
        so.Status AS ShippingStatus,  
        CASE  
            WHEN o.ShipVia = 1 THEN 'Road'  
            WHEN o.ShipVia = 2 THEN 'Airline'  
            ELSE 'Train'  
        END AS Transport  
    FROM Shippers_Orders so  
    JOIN Shippers s ON so.Shipper_ID = s.Shipper_ID  
    JOIN Orders o ON so.Order_ID = o.Order_ID  
);  
  
DROP FUNCTION fn_ShipperWithOrder;  
  
SELECT * FROM fn_ShipperWithOrder();
```

2.3.4. Function của Quang Linh

Mục tiêu: cung cấp một cách nhanh chóng để lấy tổng doanh thu cho từng khách hàng, giúp đơn giản hóa các truy vấn khi cần tính toán doanh thu cá nhân.

```

CREATE FUNCTION dbo.fn_GetCustomerSales (@CustomerID INT)
RETURNS DECIMAL(18, 2)
AS
BEGIN
    DECLARE @TotalAmount DECIMAL(18, 2);

    SELECT
        @TotalAmount = SUM(od.Quantity * od.UnitPrice * (1 - od.Discount))
    FROM Orders o
    JOIN Orders_Details od ON o.Order_ID = od.Order_ID
    WHERE o.Customer_ID = @CustomerID;

    RETURN ISNULL(@TotalAmount, 0);
END;

SELECT dbo.fn_GetCustomerSales(1) AS TotalAmount;

CREATE TABLE CustomerSales (
    Customer_ID INT PRIMARY KEY,
    TotalAmount DECIMAL(18, 2) DEFAULT 0
);

```

2.3.5. Function của Trang

Mục đích: Hàm này được tạo ra để tính tổng giá trị các đơn hàng mà một khách hàng cụ thể đã mua. Tổng giá trị này bao gồm cả các khoản giảm giá được áp dụng cho các sản phẩm trong đơn hàng.

```

--tính tổng số tiền mà một khách hàng đã chi tiêu, bao gồm cả các đơn đặt hàng và giảm giá.
CREATE FUNCTION TinhGiaTriKhachHangVIPA
(
    @CustomerID INT
)
RETURNS DECIMAL(18, 2)
AS
BEGIN
    DECLARE @TongTien DECIMAL(18, 2);

    SELECT @TongTien = SUM(od.UnitPrice * od.Quantity * (1 - od.Discount))
    FROM Orders o
    JOIN Orders_Details od ON o.Order_ID = od.Order_ID
    WHERE o.Customer_ID = @CustomerID;

    RETURN ISNULL(@TongTien, 0);
END;
| |
SELECT dbo.TinhGiaTriKhachHangVIPA(15);
SELECT * FROM Orders_Details ;
SELECT * FROM Customers ;

```

2.4. Xây dựng Trigger

2.4.1. Trigger của Ngọc Ánh

Mục đích: sau mỗi lần dữ liệu được chèn hoặc cập nhật trong bảng products nó sẽ tự động chuyển đổi giá trị trong cột color thành chữ in hoa (uppercase)

```
CREATE_TRIGGER UpdateColorToUppercase
ON Products
AFTER INSERT, UPDATE
AS
BEGIN
    SET NOCOUNT ON;
    UPDATE Products
    SET Color = UPPER(Color)
    WHERE Product_ID IN (SELECT Product_ID FROM Inserted);
END;
INSERT INTO Products (ProductName, Supplier_ID, Category_ID, UnitPrice, Quantity, Color, Size)
VALUES ('Merzy Siren Melting Color Lipbalm ', 1, 1, 10.99, 100, ' Red Bell', '3');
```

2.4.2. Trigger của Hà Văn Đạt

Mục đích : Cập nhật số lượng sản phẩm sau khi có đơn hàng được tạo

```
create trigger UpdateQuantityInProducts
on Orders_Details
after insert
as
begin
    IF EXISTS (
        SELECT 1
        FROM Products p
        JOIN inserted i ON p.Product_ID = i.Product_ID
        WHERE p.Quantity >= i.Quantity
    )
    begin
        Update Products
        set Quantity = Quantity - (select Quantity from inserted where Products.Product_ID = inserted.Product_ID)
        where Products.Product_ID in (select Product_ID from inserted)
        end
    ELSE
    BEGIN
        RAISERROR('Số lượng sản phẩm trong kho không đáp ứng được đơn hàng !', 16, 1);
    END
end
```

2.4.3. Trigger của Phương Linh

Mục đích: Lưu trữ thông tin của nhà vận chuyển, bao gồm tên công ty, ngày cập nhật, trạng thái vận chuyển, và số điện thoại, vào bảng ShipperLog. Cập nhật 1 nhà vận chuyển.

```
CREATE TABLE ShipperLog
(
    id int identity(1,1) primary key,
    companyName VARCHAR(255),
    date DATETIME,
    status VARCHAR(255),
    phone VARCHAR(255)
);

DROP TABLE ShipperLog

CREATE TRIGGER trg_ShippersOrder
ON Shippers
AFTER INSERT, UPDATE
AS
BEGIN

    INSERT INTO ShipperLog (companyName, date, status , phone)
    SELECT DISTINCT s.CompanyName, so.Date, so.Status , s.Phone
    FROM Shippers s
    INNER JOIN Shippers_Orders so ON s.Shipper_ID = so.Shipper_ID
    WHERE s.Shipper_ID IN (SELECT Shipper_ID FROM inserted);
END;

DROP TRIGGER trg_ShippersOrder
```

2.4.4. Trigger của Quang Linh

Mục tiêu: duy trì tổng doanh thu (TotalAmount) của từng khách hàng trong bảng CustomerSales và đảm bảo dữ liệu luôn nhất quán và được cập nhật ngay khi có thay đổi về số lượng, giá, hoặc chiết khấu trong các đơn hàng của khách hàng.

```
--  
CREATE TRIGGER trg_UpdateCustomerSales  
ON Orders_Details  
AFTER INSERT, UPDATE, DELETE  
AS  
BEGIN  
    DECLARE @CustomerID INT;  
  
    --  
    SELECT @CustomerID = o.Customer_ID  
    FROM Orders o  
    JOIN inserted i ON o.Order_ID = i.Order_ID;  
  
    DECLARE @TotalAmount DECIMAL(18, 2);  
    SELECT  
        @TotalAmount = SUM(od.Quantity * od.UnitPrice * (1 - od.Discount))  
    FROM Orders o  
    JOIN Orders_Details od ON o.Order_ID = od.Order_ID  
    WHERE o.Customer_ID = @CustomerID;  
  
    IF EXISTS (SELECT 1 FROM CustomerSales WHERE Customer_ID = @CustomerID)  
    BEGIN  
        UPDATE CustomerSales  
        SET TotalAmount = @TotalAmount  
        WHERE Customer_ID = @CustomerID;  
    END  
    ELSE  
    BEGIN  
        INSERT INTO CustomerSales (Customer_ID, TotalAmount)  
        VALUES (@CustomerID, @TotalAmount);  
    END  
END;
```

2.4.5. Trigger của Trang

Mục đích: Trigger này được thiết kế để tự động cập nhật số lượng tồn kho của sản phẩm sau khi có một đơn hàng mới được thêm vào bảng OrderDetails. Khi một bản ghi mới được chèn vào, trigger sẽ giảm số lượng tồn kho tương ứng của sản phẩm đó trong bảng Products.

```
--TRIGGER tính số lượng tồn kho của sp sau khi bán
CREATE TABLE OrderDetails (
    Order_ID INT,
    Product_ID INT,
    UnitPrice DECIMAL(10, 2),
    Quantity INT,
    Discount DECIMAL(4, 2)
);

GO
CREATE TRIGGER update_product_stock_after_order
ON OrderDetails
AFTER INSERT
AS
BEGIN
    DECLARE @quantity_sold INT;
    DECLARE @product_id INT;

    SELECT @quantity_sold = Quantity, @product_id = Product_ID FROM inserted;

    UPDATE Products
    SET Quantity = Quantity - @quantity_sold
    WHERE Product_ID = @product_id;
END;
GO

--TEST
INSERT INTO OrderDetails (Order_ID, Product_ID, UnitPrice, Quantity, Discount)
VALUES (1234567, 1, 10.99, 2, 0.1);

--TEST
SELECT *
FROM OrderDetails
WHERE Order_ID = 12345;

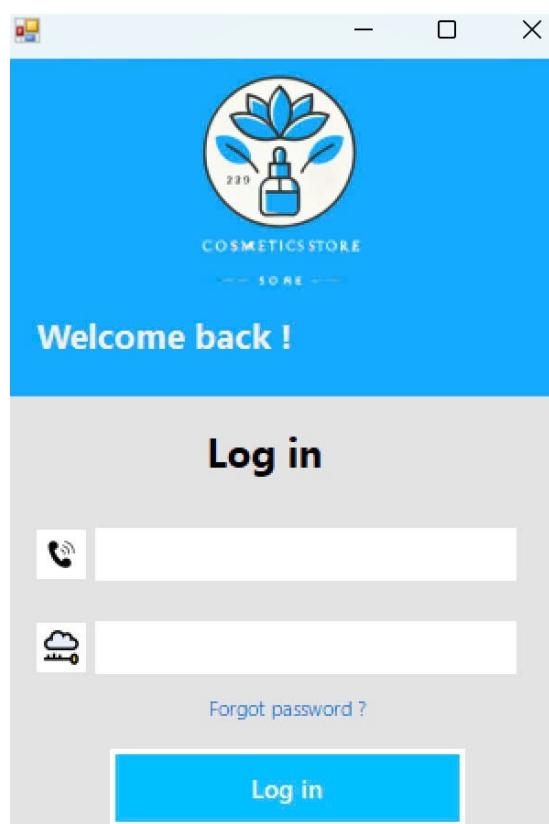
--TEST
SELECT Product_ID, ProductName, Quantity
FROM Products
WHERE Product_ID = 1;
```

CHƯƠNG III: ỨNG DỤNG CÁC ĐỐI TƯỢNG VÀO PHẦN MỀM QUẢN LÝ

3.1. Mô tả chức năng phần mềm quản lý

3.1.1. Chức năng đăng nhập

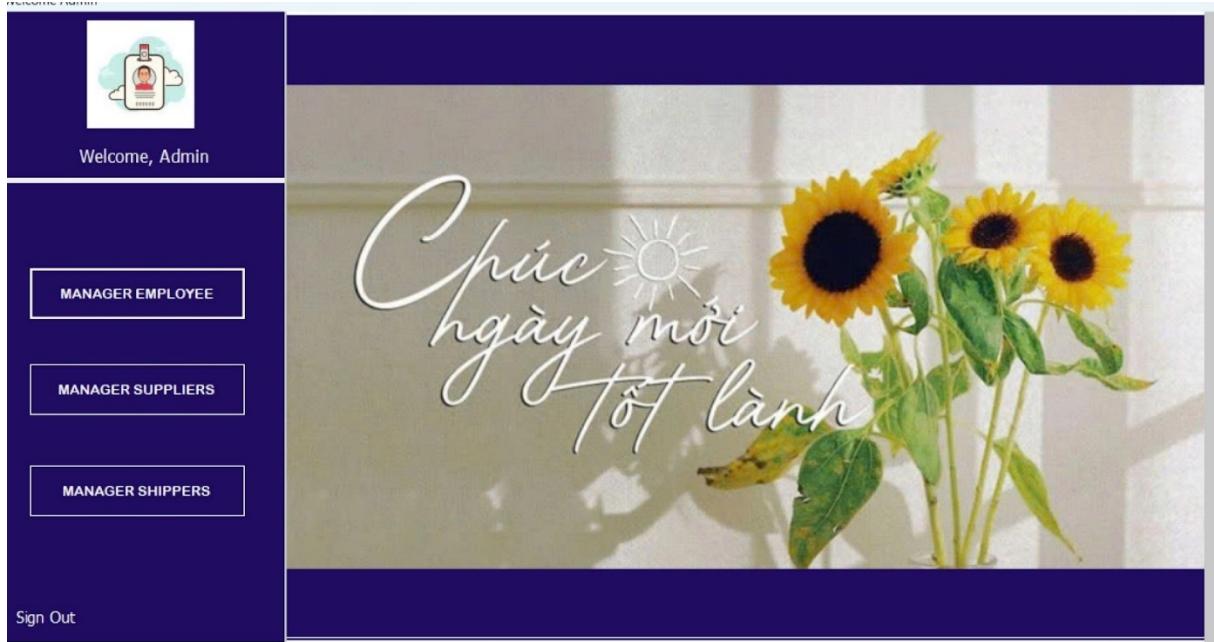
- Mục tiêu: Cho phép Admin và Nhân viên có thể đăng nhập vào hệ thống để thực hiện các tác vụ quản lý và xử lý thông tin với các quyền hạn riêng.
- Tác nhân: Admin và Nhân viên



Hình 3.1 Màn hình hiển thị đăng nhập

3.1.2 Chức năng của Admin

- Màn hình tổng quan của hệ thống cho phép Admin thực hiện các tác vụ quản lý.



Hình 3.2 Màn hình tổng quan

3.1.2.1: Chức năng quản lý nhân viên:

- Mục đích: Hỗ trợ quản lý nhân viên của cửa hàng, tại đây người dùng có thể thực hiện các chức năng: Thêm mới, chỉnh sửa, xóa, tìm kiếm nhân viên.
- Tác nhân tham gia: Admin

The screenshot shows a web-based application for managing employees. At the top left is a logo of a person holding a briefcase. Below it says "Welcome, Admin". On the left sidebar, there are three buttons: "MANAGER EMPLOYEE", "MANAGER SUPPLIERS", and "MANAGER SHIPPERS". At the bottom left is a "Sign Out" link. The main area has a title "Employee 's Data" above a table with columns: employees, firstname, lastname, birthdate, address, phone, email, city, fax, homepage, hiredate, role, and password. A row for employee 1 is selected, showing details for John Doe. Below the table is a form with fields for First name (111), Address (987 Work Rd), Email (sarah.wilson@company.com), Last name (Wilson), Phone (+61298765432), HomePage (www.sarahwilson.com), Birth date (1991/02/14), Password (163928), Fax (+61 2 9876 5433), HireDate (2014/07/18), City (Sydney), Role (staff), and a "Search" button. At the bottom are "Add", "Update", "Delete", and "Clear" buttons.

Hình 3.3 Màn hình trang quản lý nhân viên

3.1.2.2: Chức năng quản lý nhà cung cấp:

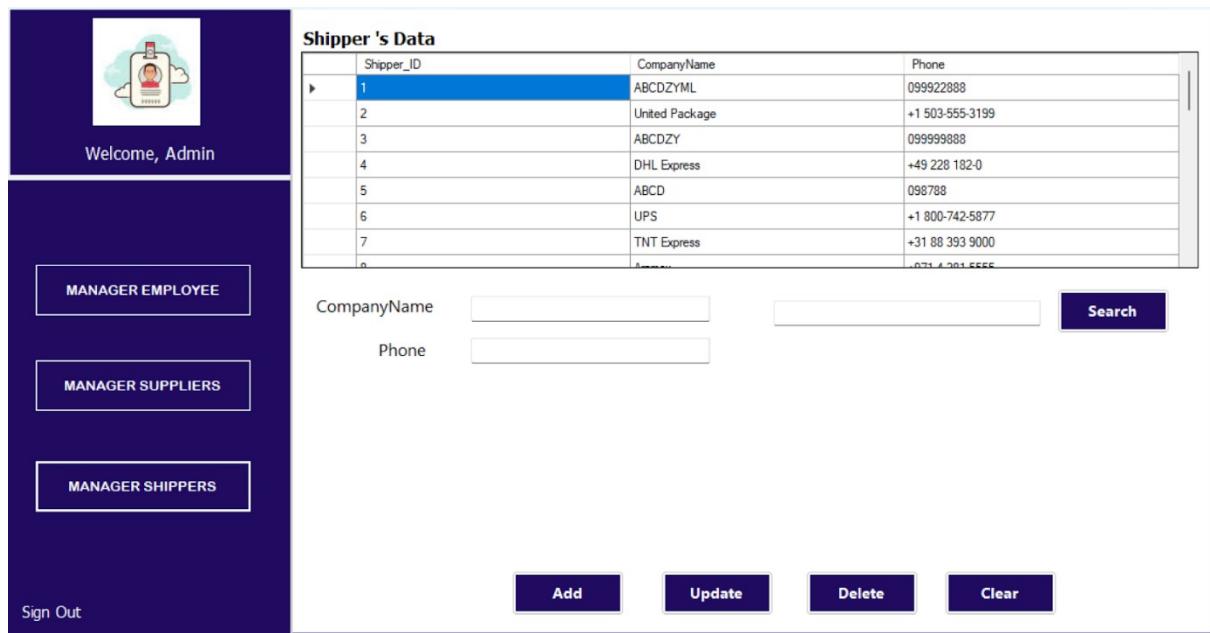
- Mục đích: Hỗ trợ quản lý nhà cung cấp sản phẩm cho cửa hàng, tại đây người dùng có thể thực hiện các chức năng: Thêm mới, chỉnh sửa, xóa, tìm kiếm nhà cung cấp.
- Tác nhân tham gia: Admin

The screenshot shows a web-based application for managing suppliers. At the top left is a logo of a person holding a briefcase. Below it says "Welcome, Admin". On the left sidebar, there are three buttons: "MANAGER EMPLOYEE", "MANAGER SUPPLIERS", and "MANAGER SHIPPERS". At the bottom left is a "Sign Out" link. The main area has a title "Supplier 's Data" above a table with columns: Supplier_ID, CompanyName, ContactName, Address, City, Phone, HomePage, and Fax. A row for supplier 1 is selected, showing details for L'Oréal. Below the table is a form with fields for CompanyName, Phone, ContactName, City, Address, Fax, and HomePage. There is also a "Search" button and a text input field. At the bottom are "Add", "Update", "Delete", and "Clear" buttons.

Hình 3.4 Màn hình trang quản lý nhà cung cấp

3.1.2.3: Chức năng quản lý người giao hàng:

- Mục đích: Hỗ trợ quản lý người giao hàng, tại đây người dùng có thể thực hiện các chức năng: Thêm mới, chỉnh sửa, xóa, tìm kiếm người giao hàng.
- Tác nhân tham gia: Admin

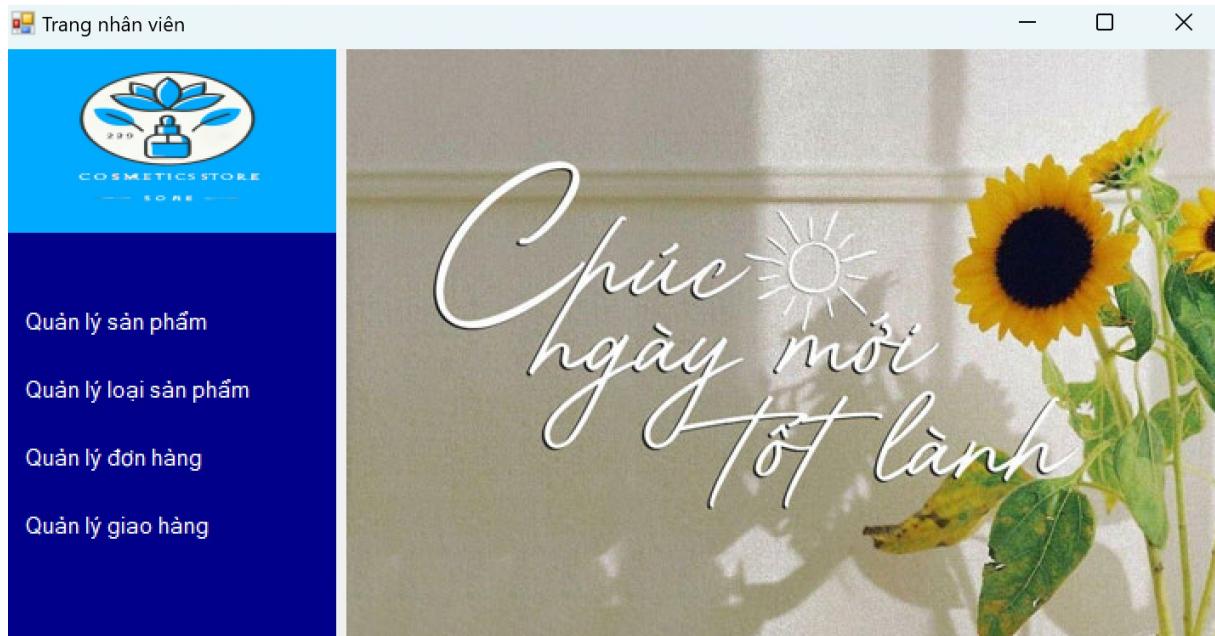


Hình 3.5 Màn hình trang quản lý người giao hàng

3.1.3: Chức năng của nhân viên:

3.1.3.1 Màn hình tổng quan trang nhân viên

- Màn hình tổng quan của hệ thống cho phép nhân viên thực hiện các tác vụ của quản lý.



Hình 3.6 Màn hình tổng quan trang nhân viên

3.1.3.2. Chức năng quản lý sản phẩm:

- Mục đích: Hỗ trợ quản lý sản phẩm của cửa hàng, tại đây người dùng có thể thực hiện các chức năng: Thêm mới, chỉnh sửa, xóa, tìm kiếm sản phẩm.

- Tác nhân tham gia: Employee

The screenshot shows a Windows application window titled "Quản lý sản phẩm". At the top, there is a search bar labeled "Tim kiem sản phẩm" and a "Submit" button. Below the search bar are four input fields: "Đơn giá", "Tên sản phẩm", "Số lượng", and "Loại sản phẩm". To the right of these are buttons for "Thêm" (Add), "Sửa" (Edit), and "Xóa" (Delete). There are also two dropdown menus for "Màu sắc" and "Nhà cung cấp". The main area of the window displays a table with the following data:

	ProductName	CategoryName	CompanyName	UnitPrice	Quantity
▶	L'Oréal Revitalift ...	Skincare	L'Oréal	19.99	100
	Estée Lauder Do...	Makeup	Estée Lauder	42.00	50
	Shiseido Tsubaki...	Haircare	Shiseido	15.50	90
	Axe Body Spray	Fragrance	Unilever	5.99	200
	Olay Body Wash	Bath & Body	Procter & Gamble	7.99	150
	Dior Sauvage EDT	Men's Grooming	IVMH	95.00	30

Hình 3.7 Màn hình quản lý sản phẩm

3.1.3.3. Chức năng quản lý loại sản phẩm (category)

- Mục đích: Hỗ trợ quản lý loại sản phẩm của cửa hàng, tại đây người dùng có thể thực hiện các chức năng: Thêm mới, chỉnh sửa, xóa loại sản phẩm.

- Tác nhân tham gia: Employee

	Category Name	Description
▶	Acne Treatment	Products for treating and preventing acne
	Acne Treatment123	Products for treating and preventing acne
	Anti-Aging	Products designed to reduce signs of aging
	Bath & Body	Products for bathing and body care
	Cruelty-Free	Products not tested on animals
	Dermatologist-Recommended	Products recommended by dermatologists
	Drugstore Beauty	Affordable beauty products
	Eco-Friendly	Environmentally friendly beauty products
	Ethnic Beauty	Products catering to specific ethnic needs
	Fragrance	Perfumes and colognes

Category Name:

Description:

Hình 3.8 Màn hình quản lý loại sản phẩm

3.1.3.4. Chức năng xem thông tin đơn hàng của từng khách hàng

- Mục đích: hiển thị thông tin đơn hàng bao gồm tên khách hàng và tổng tiền bao gồm cả chiết khấu của vị khách hàng đó được tìm kiếm
- Tác nhân: Employee

Order_ID	CustomerName	OrderDate	TotalAmount
3	Olivia Garcia	07/01/2023	97

Hình 3.9 Màn hình xem thông tin đơn hàng

3.1.3.5. Chức năng xem thông tin chi tiết đơn hàng và người giao hàng

- Mục đích: hiển thị toàn bộ thông tin đơn hàng và thông tin shipper của từng đơn hàng
- Tác nhân: Employee

ShipperOrderForm

	Customer_ID	Employee_ID	Order_ID	OrderDate	ShipAddress	ShipCity	
▶	1	1	1	2023-01-05	123 Main St	New York	1
1	1	1	1	2023-01-05	123 Main St	New York	1
1	1	1	1	2023-01-05	123 Main St	New York	1
1	1	1	1	2023-01-05	123 Main St	New York	1
1	1	1	1	2023-01-05	123 Main St	New York	1
1	1	1	1	2023-01-05	123 Main St	New York	1
1	1	1	1	2023-01-05	123 Main St	New York	1
1	1	1	1	2023-01-05	123 Main St	New York	1
1	1	1	1	2023-01-05	123 Main St	New York	1

Hình 3.10 Màn hình xem thông tin chi tiết đơn hàng và shipper

3.1.3.6. Chức năng tạo đơn hàng

- Mục đích: Hỗ trợ nhân viên tạo mới 1 đơn hàng trong hệ thống.
- Tác nhân: Employee

The screenshot shows a Windows application window titled "Tạo đơn hàng". The form consists of several input fields and dropdown menus. At the top left, there are fields for "Họ khách hàng" (Customer Name) and "Số điện thoại" (Phone Number). To the right, there is a field for "Tổng tiền" (Total Amount). Below these are fields for "Ngày bán" (Sale Date) and "Ngày yêu cầu giao" (Delivery Date). Further down are fields for "Địa chỉ giao hàng" (Delivery Address), "Thành phố" (City), "Sản phẩm" (Product), "Giá sản phẩm" (Product Price), "Giảm giá" (Discount), and "Số lượng" (Quantity). A dropdown menu for "Đơn vị vận" (Shipping Unit) is also present. On the right side of the form, there are two buttons: a dark blue "Tạo đơn hàng" (Create Order) button and an orange "Hủy" (Cancel) button.

Hình 3.11 Màn hình xem thông tin tạo đơn hàng

3.2 Mô tả ứng dụng một số đối tượng dữ liệu trong phần mềm

3.2.1 View:

Mục đích: Xem thông tin các đơn hàng

Luồng sự kiện :

1. Nhấn chọn “Quản lý đơn hàng” từ giao diện chính
2. Hệ thống hiển thị form xem đơn hàng
3. Thông tin các đơn hàng được hiển thị lên form

3.2.2 Procedure:

Mục đích: Tìm kiếm sản phẩm theo tên sản phẩm

Luồng sự kiện:

1. Nhập tên sản phẩm
2. Chọn nút “Tìm kiếm”
3. Hệ thống thực hiện truy vấn CSDL

3.1. Nếu không nhập tên sản phẩm

Hệ thống trả về thông tin của tất cả sản phẩm

3.2. Khi nhập tên hoặc ký tự

Hệ thống thực hiện tìm kiếm gần đúng

Ví dụ: Nếu nhập "phone"

Kết quả: "iPhone", "Smartphone", "Telephone"

1. Hệ thống hiển thị thông tin hiển thị bao gồm : Mã sản phẩm, tên sản phẩm, giá bán , số lượng tồn kho, màu sắc, kích thước, loại sản phẩm , mô tả sản phẩm

3.2.3 Trigger:

Mục đích: Cập nhập số lượng của sản phẩm khi có đơn hàng được tạo

Luồng sự kiện:

1. Chọn nút “Tạo đơn hàng”

2. Điền các thông tin cần thiết vào form để tạo đơn hàng

Thông tin gồm : Họ tên khách hàng , ngày đặt hàng , ngày yêu cầu giao , địa chỉ , tên sản phẩm , giảm giá, số lượng , đơn vị vận chuyển ,số điện thoại

- 2.1. Thiếu thông tin thì hiển thị thông báo “Hãy điền đầy đủ thông tin cho đơn hàng ”

2.2. Điền đầy đủ thông tin

- 2.2.1. Kiểm tra trường giảm giá không phải là một số thì đưa ra thông báo “Giảm giá không hợp lệ ”

- 2.2.2. Kiểm tra trường số điện thoại không phải là một chuỗi số thì đưa ra thông báo “Số điện thoại không hợp lệ ! Vui lòng nhập lại ! ”

3. Chọn nút “Tạo đơn hàng”

- 3.1. Nếu số lượng cần bán nhiều hơn số lượng trong kho thì hiển thị thông báo “Số lượng sản phẩm trong kho không đáp ứng được đơn hàng !”

4. Thực hiện thêm mới khách hàng có các thông tin đã nhập khi tạo đơn hàng

5. Cập nhật đơn hàng mới vào bảng Orders
 6. Thêm dữ liệu vào bảng Orders_Details
 7. Hệ thống tự động cập nhật số lượng tồn kho của sản phẩm tương ứng đã được bán
 8. Hiển thị thông báo tạo đơn hàng thành công
- Sự kiện ngoại lệ :
- Thực hiện hủy giao dịch nếu có lỗi xảy ra
 - Nhân viên nhấn chọn “Hủy” để hủy việc tạo đơn hàng

3.2.4 Function:

Mục đích: Xem thông tin đơn hàng của khách hàng

Luồng sự kiện:

1. Chọn khách hàng muốn xem thông tin tại combobox
 2. Chọn nút "Xem đơn hàng" để xem thông tin đơn hàng
- 2.1. Gọi function fn_GetCustomerSales với CustomerID đã chọn
- Khi chưa chọn khách hàng

Hệ thống sẽ hiển thị thông báo: "Please select a customer first"

- 2.2. Truy vấn dữ liệu từ CSDL

Hiển thị kết quả trong DataGridView gồm :

CustomerName: Họ tên khách hàng

OrderDate: Ngày đặt hàng (định dạng dd/MM/yyyy)

TotalAmount: Tổng tiền đơn hàng

- 2.2.1. Khách hàng không có đơn hàng

DataGridView trống, không hiển thị dòng nào

3. Sự kiện ngoại lệ :

- Lỗi kết nối database: Hiện thông báo "Database error: [chi tiết lỗi]"
- Lỗi khác: Hiện thông báo "Error loading orders: [chi tiết lỗi]"

