**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**MÔN: HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**ĐỀ TÀI:**

**XÂY DỰNG WEBSITE BÁN THỨC ĂN**

**GVHD: NGUYỄN THÀNH SƠN**

**SVTH:**

**VÕ HOÀNG KHẢ DIỆU 19133014**

**NGUYỄN QUỐC BẢO 19133002**

**DIỆP BẢO TÙNG 19110045**

**PHAN NGUYỄN THANH TRÚC 19110488**

**TP.Hồ Chí Minh, ngày 22 tháng 09 năm 2021**

MỤC LỤC

[Chương 1: ĐẶC TẢ HỆ THỐNG 3](#_Toc85483249)

[1. Mô tả phần mềm: 3](#_Toc85483250)

[1.1. Mục tiêu 3](#_Toc85483251)

[1.2. Mô tả đề tài: 3](#_Toc85483252)

[2. Quy trình xử lý: 4](#_Toc85483253)

[3. Mô hình diagram: 4](#_Toc85483254)

[Chương 2: CẤU HÌNH VÀ CÀI ĐẶT HỆ THỐNG 5](#_Toc85483255)

[1. Tạo cơ sở dữ liệu và ràng buộc: 5](#_Toc85483256)

[2. Mô hình ERD: 7](#_Toc85483257)

[3 . Lược đồ quan hệ: 7](#_Toc85483258)

[4. Mô tả thuộc tính và mối quan hệ của chúng: 7](#_Toc85483259)

[Bảng Users (Người dùng) 7](#_Toc85483260)

[Bảng Transactions (Chi tiết đơn hàng) 8](#_Toc85483261)

[Bảng Orders (Đơn hàng) 8](#_Toc85483262)

[Bảng Products (Sản phẩm) 8](#_Toc85483263)

[Bảng Categories (Thể loại) 9](#_Toc85483264)

[Bảng Suppliers(Nhà phân phối) 9](#_Toc85483265)

[5. Kết nối cơ sở dữ liệu: 10](#_Toc85483266)

[Chương 3: XÂY DỰNG CƠ SỞ DỮ LIỆU 12](#_Toc85483267)

[1. Danh sách các Trigger 12](#_Toc85483268)

[2. Danh sách các View 15](#_Toc85483269)

[3. Danh sách các Procedure 16](#_Toc85483270)

# ĐẶC TẢ HỆ THỐNG

## 1. Mô tả phần mềm:

### 1.1. Mục tiêu

- Chỉnh sửa lại mục tiêu, mô tả đề tài (Chức năng sửa rồi chỉ chỉnh lại cho đẹp thôi ^^)

Có thể thấy, những năm gần đây, việc cập nhật các thông tin cũng như thực hiện các giao dịch qua mạng Internet ngày càng trở nên phổ biến và dễ dàng. Và một loại hình kinh doanh, được khai thác từ khía cạnh trên, hiện nay khá phổ biến đó là buôn bán qua mạng. Hình thức kinh doanh này mang lại rất nhiều lợi ích cho cả người tiêu dùng cũng như nhà cung cấp, giúp rút ngắn thời gian, công sức, chi phí đi lại của người tiêu dùng khi phải mua hàng trực tiếp tại cửa hàng đồng thời giúp doanh nghiệp dễ dàng theo dõi doanh thu, tiết kiệm thời gian, nhân lực trong khâu quản lý sản phẩm tại cửa hang. Bên cạnh đó, đảm bảo được dữ liệu luôn chính xác, đúng đắn, hạn chế sự cố ngoài ý muốn.

Với sự mở rộng cả về quy mô và chất lượng của các dịch vụ mua bán, trao đổi hàng hóa, cũng như nhu cầu của con người theo cách nhanh chóng và thuận tiện, các trang website bán hàng ngày càng đa dạng và đáp ứng được thị hiếu khách hàng.

Nắm bắt được tình hình trên, cùng với mục đích tìm hiểu, tiếp cận và góp phần đẩy mạnh sự phát triển của nền thương mại điện tử ở nước ta, nhằm xây dựng, phát triển phần mềm quản lý chặt chẽ hoạt động kinh doanh thương mại điện tử nhóm em đã chọn thực hiện đề tài “XÂY DỰNG ỨNG DỤNG QUẢN LÍ CỬA HÀNG BÁNH”.

### 1.2. Mô tả đề tài:

“Phần mềm quản lý cửa hàng bánh” là một phần mềm giúp người bán quản lý tình hình kinh doanh của cửa hàng thông qua Database. Phần mềm cho phép người dùng nắm bắt, quản lý được doanh thu, thông tin của khách hàng, sản phẩm cùng các giao dịch của khách hàng. Phần mềm bao gồm 2 giao diện chính với chức năng khác nhau dành cho 2 nhóm người dùng: Admin (người quản trị) và User (khách hàng).

Chức năng:

Admin (người quản trị):

- Đăng nhập bằng tài khỏan quản trị riêng

- Quản lí thông tin người dùng và phân quyền truy cập hệ thống: lưu trữ các thông tin khi người dùng thực hiện tạo tài khoản trên phầm mềm

- Quản lý các danh mục sản phẩm - thực đơn: Thêm mới, cập nhật thực đơn đồng thời giúp phân loại sản phẩm thành các danh mục để dễ dàng quản lí, tìm kiếm

- Quản lý sản phẩm: kiểm tra, thống kê số lượng sản phẩm tồn kho, cập nhật số lượng, sản phẩm mới cho phần mềm

- Quản lý doanh thu: Thống kê doanh thu qua các đơn hàng theo từng ngày, tháng, …

- Quản lý đơn hàng: Thống kê, theo dỗi tiến độ đơn hàng của người.Mỗi người dùng sẽ có một danh sách các đơn hàng đã mua và tiến độ giao hàng của từng đơn.

User (khách hàng):

- Đăng kí tài khoản: Người dùng thực hiện đăng kí tài khỏan bằng SĐT

- Đăng nhập: để có thể tương tác - đặt hàng trên phần mềm người dùng phải đăng nhập bằng tài khỏan đã được đăng ký

- Quản lý thông tin cá nhân: Họ tên, địa chỉ giao hàng, vouchers, phương thức thanh tóan, ...

- Kiểm tra đơn hàng, thanh tóan: Kiểm tra, xác nhận và thực hiện thanh tóan đơn hàng qua nhiều hình thức khác nhau.

## 2. Quy trình xử lý:

- Thêm quy trình sử lý

Để sử dụng các chức năng của phần mềm, yêu cầu người dùng phải đăng nhập vào hệ thống. Việc đăng nhập được thực hiện bằng chức năng **Đăng nhập**.

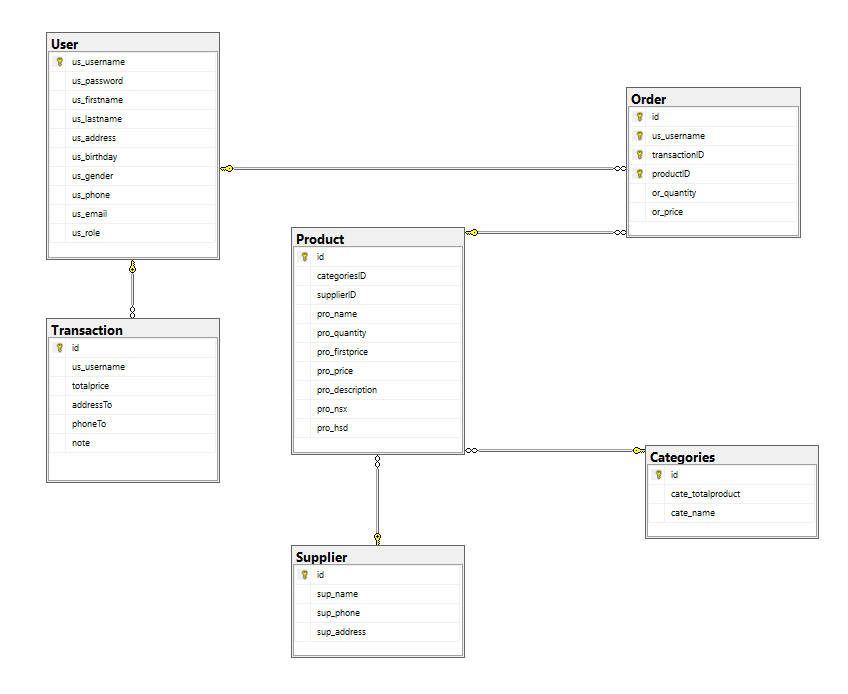
Admin: Đăng nhập bằng tài khoản quản trị

* + - Người dùng chọn Quản lí đơn hàng để xem thông tin chi tiết các đơn hàng, để chuẩn bị đơn hàng hoặc **Huỷ** đơn hàng.
    - Người dùng chọn **Quản lí sản phẩm** để xem, cập nhật thông tin của sản phẩm. Có thể sử dụng các chức năng **Thêm**, **Sửa**, **Xoá** sản phẩm.
    - Người dùng chọn **Thông kê doanh thu** để xem doanh thu của cửa hàng theo ngày, tuần, tháng, năm.
    - Người dùng chọn **Đăng xuất** để kết thúc hoạt động với phầm mềm.

User: Đăng nhâp bằng tài tài khoản đã được đăng kí bằng chức năng **Đăng kí**

* + - Người dụng chọn xem sản phẩm, chọn **Thêm vào giỏ hàng** để chọn mua sản phẩm.
    - Người dùng vào **Giỏ hàng** để xem thông tin đơn hàng để xem chi tiết đơn hàng, chỉnh sửa đơn hàng.
    - Người dùng chọn **Thanh toán** để thanh toán đơn hàng của mình.
    - Người dùng có thể chọn **Đăng xuất** để kết thúc hoặc **Tiếp tục mua sắm**.

## 3. Mô hình diagram:

****

# CẤU HÌNH VÀ CÀI ĐẶT HỆ THỐNG

## 1. Tạo cơ sở dữ liệu và ràng buộc:

Áp dụng các ràng buộc toàn vẹn như: not null, unique, ràng buộc về khoá chính, khoá ngoại

* Tạo cơ sở dữ liệu:

CREATE DATABASE DIEPFAMILY

GO

CREATE TABLE [dbo].[User] (

[us\_username][varchar](20) NOT NULL,

[us\_password][varchar](20) NOT NULL,

[us\_firstname][nvarchar](50) NOT NULL,

[us\_lastname][nvarchar](50) NOT NULL,

[us\_address][nvarchar](50) NOT NULL,

[us\_birthday][date] NOT NULL,

[us\_gender][char](1) NOT NULL,

[us\_phone][varchar](11) NOT NULL,

[us\_email][varchar](50) NOT NULL,

[us\_role][nvarchar](20) NOT NULL

CONSTRAINT PK\_User PRIMARY KEY (us\_username))

GO

CREATE TABLE [dbo].[Product] (

[id][int] IDENTITY(1,1) NOT NULL,

[categoriesID][int] NOT NULL,

[supplierID][int] NOT NULL,

[pro\_name][nvarchar](50) NOT NULL,

[pro\_quantity][int] NOT NULL,

[pro\_firstprice][real] NOT NULL,

[pro\_price][real] NOT NULL,

[pro\_description][nvarchar](100),

[pro\_nsx][date] NOT NULL,

[pro\_hsd][date] NOT NULL

CONSTRAINT PK\_Product PRIMARY KEY (id))

GO

CREATE TABLE [dbo].[Transaction] (

[id][int] IDENTITY(1,1) NOT NULL,

[us\_username][varchar](20) NOT NULL,

[addressTo][nvarchar](MAX) NOT NULL,

[phoneTo][char](11) NOT NULL,

[totalprice][real] DEFAULT 0 NOT NULL,

[createdate][date] NOT NULL,

[note][nvarchar](100)

CONSTRAINT PK\_Transaction PRIMARY KEY (id))

GO

CREATE TABLE [dbo].[Order] (

[id][int] IDENTITY(1,1) NOT NULL,

[us\_username][varchar](20) NOT NULL,

[transactionID][int] NOT NULL,

[productID][int] NOT NULL,

[or\_quantity][int] NOT NULL,

[or\_price][real] DEFAULT 0,

CONSTRAINT PK\_Order PRIMARY KEY (id, us\_username, transactionID, productID))

GO

CREATE TABLE [dbo].[Supplier] (

[id][int] IDENTITY(1,1) NOT NULL,

[sup\_name][nvarchar](50) NOT NULL,

[sup\_phone][char](11) NOT NULL,

[sup\_address][nvarchar](MAX) NOT NULL,

CONSTRAINT PK\_Supplier PRIMARY KEY (id))

GO

CREATE TABLE [dbo].[Categories] (

[id][int] IDENTITY(1,1) NOT NULL,

[cate\_totalproduct][int] DEFAULT 0,

[cate\_name][nvarchar](50) NOT NULL,

CONSTRAINT PK\_Categories PRIMARY KEY (id))

GO

* Ràng buộc khoá ngoại:

ALTER TABLE [dbo].[Product] WITH CHECK ADD CONSTRAINT

[FK\_Product\_Categories] FOREIGN KEY ([categoriesID])

REFERENCES [dbo].[Categories] ([id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[Product] CHECK CONSTRAINT [FK\_Product\_Categories]

GO

ALTER TABLE [dbo].[Product] WITH CHECK ADD CONSTRAINT

[FK\_Product\_Supplier] FOREIGN KEY ([supplierID])

REFERENCES [dbo].[Supplier] ([id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[Product] CHECK CONSTRAINT [FK\_Product\_Supplier]

GO

ALTER TABLE [dbo].[Transaction] WITH CHECK ADD CONSTRAINT

[FK\_Transaction\_User] FOREIGN KEY ([us\_username])

REFERENCES [dbo].[User] ([us\_username])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[Transaction] CHECK CONSTRAINT [FK\_Transaction\_User]

GO

ALTER TABLE [dbo].[Order] WITH CHECK ADD CONSTRAINT

[FK\_Order\_User] FOREIGN KEY ([us\_username])

REFERENCES [dbo].[User] ([us\_username])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[Order] CHECK CONSTRAINT [FK\_Order\_User]

GO

ALTER TABLE [dbo].[Order] WITH CHECK ADD CONSTRAINT

[FK\_Order\_Transaction] FOREIGN KEY ([transactionID])

REFERENCES [dbo].[Transaction] ([id])

ON DELETE CASCADE

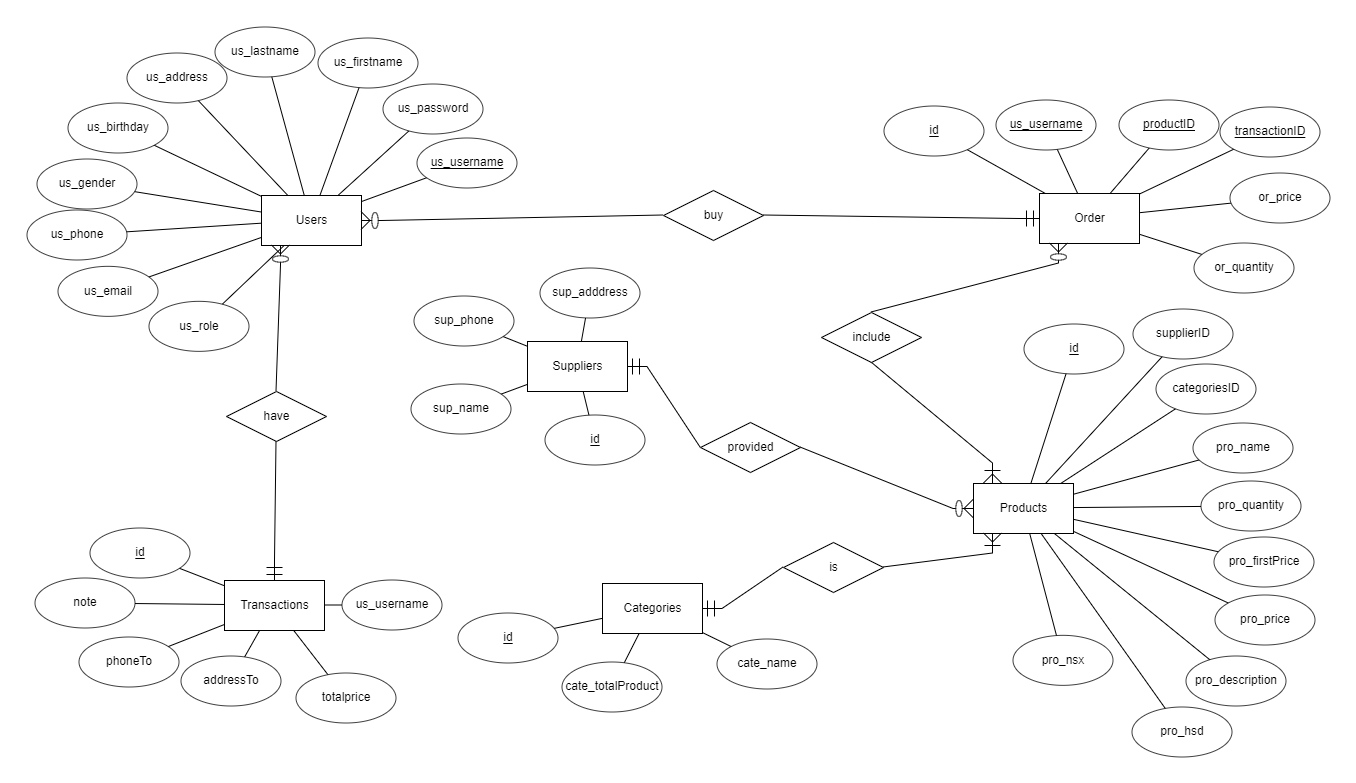
GO

ALTER TABLE [dbo].[Order] CHECK CONSTRAINT [FK\_Order\_Transaction]

GO

## 2. Mô hình ERD:

- Vẽ lại mô hình ERD



## 3 . Lược đồ quan hệ:

## 4. Mô tả thuộc tính và mối quan hệ của chúng:

#### Bảng Users (Người dùng)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** |
| 1 | us\_username | Char(20) | Tên đăng nhập | Khoá chính |
| 2 | us\_password | Char(20) | Mật khẩu |  |
| 3 | us\_firstname | Nvarchar (50) | Tên người dung |  |
| 4 | us\_lastname | Nvarchar (50) | Họ người dung |  |
| 5 | us\_address | Nvarchar (100) | Địa chỉ |  |
| 6 | us\_birthday | date | Ngày sinh |  |
| 7 | us\_gender | Char(1) | Giới tính |  |
| 8 | us\_phone | Char(11) | Số điện thoại |  |
| 9 | us\_email | Char (50) | Địa chỉ email |  |
| 10 | us\_role | Nvarchar(20) | Phân quyền |  |

#### Bảng Transactions (Chi tiết đơn hàng)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** |
| 1 | id | Int | Mã giao hang | Khóa chính |
| 2 | us\_username | Varchar(20) | Mã người dùng | Khoá ngoại |
| 3 | addressTo | Nvarchar(100) | Địa chỉ giao hàng |  |
| 4 | phoneTo | Char(11) | Số điện thoại nhận hàng |  |
| 5 | totalprice | Real | Tổng tiền đơn hàng |  |
| 6 | createdate | date | Ngày thành lập đơn |  |
| 7 | note | Nvarchar(100) | Ghi chú |  |

#### Bảng Orders (Đơn hàng)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** |
| 1 | id | int | Mã đơn hang | Khóa chính |
| 2 | us\_username | Varchar(20) | Mã người dùng | Khoá ngoại |
| 3 | transactionID | Int | Mã giao hàng | Khoá ngoại |
| 4 | product\_ID | Int | Mã sản phẩm | Khoá ngoại |
| 5 | or\_quantity | Int | Số lượng |  |
| 6 | or\_price | real | Giá tiền |  |

#### Bảng Products (Sản phẩm)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** |
| 1 | id | int | Mã sản phẩm | Khóa chính |
| 2 | supplierID | Int | Mã nhà phân phối | Khoá ngoại |
| 3 | categoriesID | Int | Mã thể loại | Khoá ngoại |
| 4 | pro\_name | Nvarchar(50) | Tên sản phẩm |  |
| 5 | pro\_fristprice | Float | Giá gốc |  |
| 6 | pro\_price | Float | Giá bán |  |
| 7 | pro\_description | Nvarchar(100) | Mô tả |  |
| 8 | pro\_nsx | Date | Ngày sản xuất |  |
| 9 | pro\_hsd | date | Hạn sử dụng |  |

#### Bảng Categories (Thể loại)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** |
| 1 | id | Int | Mã thể loại | Khóa chính |
| 2 | cate\_totalproduct | Int | Tổng sản phẩm |  |
| 3 | cate\_name | Char(50) | Tên thể loại |  |

#### Bảng Suppliers(Nhà phân phối)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ràng buộc** |
| 1 | id | int | Mã nhà phân phối | Khóa chính |
| 2 | sup\_name | Nvarchar(50) | Tên nhà phân phối |  |
| 3 | sup\_phone | Char(11) | Số điện thoại |  |
| 4 | sup\_address | Nvarchar(100) | Địa chỉ |  |

## 5. Kết nối cơ sở dữ liệu:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Data;

using System.Data.SqlClient;

namespace DataProvider

{

public class DBLayer

{

SqlConnection cnn;

SqlCommand cmd;

SqlDataAdapter adp;

string strConnect =

@"Data Source=DESKTOP-JS97UTJ\SQL\_BAONGUYEN;Initial Catalog=DIEPFAMILY;Integrated Security = True";

public DBLayer()

{

cnn = new SqlConnection(strConnect);

cmd = cnn.CreateCommand();

}

// Select query

public DataSet ExecuteQueryDataSet(

string strSQL, CommandType ct,

params SqlParameter[] p)

{

cmd.CommandText = strSQL;

cmd.CommandType = ct;

adp = new SqlDataAdapter(cmd);

DataSet ds = new DataSet();

adp.Fill(ds);

return ds;

}

public string ExecuteQueryXML(string strSQL, CommandType ct, params SqlParameter[] p)

{

cmd.CommandText = strSQL;

cmd.CommandType = ct;

adp = new SqlDataAdapter(cmd);

DataSet ds = new DataSet();

adp.Fill(ds);

return ds.GetXml();

}

// action query

public bool MyExecuteNonQuery(string strSQL,

CommandType ct, ref string error,

params SqlParameter[] param)

{

bool f = false;

cnn.Open();

cmd.Parameters.Clear();

cmd.CommandText = strSQL;

cmd.CommandType = ct;

foreach (SqlParameter p in param)

cmd.Parameters.Add(p);

try

{

cmd.ExecuteNonQuery();

f = true;

}

catch (SqlException ex)

{

error = ex.Message;

}

finally

{

cnn.Close();

}

return f;

}

}

}

# XÂY DỰNG CƠ SỞ DỮ LIỆU

## 1. Danh sách các Trigger

1.1. Kiểm tra số lượng sản phẩm khi đặt hàng

CREATE TRIGGER [dbo].[Check\_Order\_quantity] ON [dbo].[Order]

FOR UPDATE, INSERT

AS

BEGIN

DECLARE @quantity int

SELECT @quantity = ne.or\_quantity

FROM inserted ne

IF (@quantity <= 0)

BEGIN

PRINT N'SỐ LƯỢNG KHÔNG HỢP LỆ'

ROLLBACK TRANSACTION

END

END

GO

1.2. Kiểm tra số lượng sản phẩm khi thêm một sản phẩm mới

CREATE TRIGGER [dbo].[Check\_Product\_quantity] ON [dbo].[Product]

FOR UPDATE, INSERT

AS

BEGIN

DECLARE @quantity int

SELECT @quantity = ne.pro\_quantity

FROM inserted ne

IF (@quantity < 0)

BEGIN

PRINT N'SỐ LƯỢNG KHÔNG HỢP LỆ'

ROLLBACK TRANSACTION

END

END

GO

1.3. Kiểm tra trùng user khi thêm một user mới

CREATE TRIGGER [dbo].[Check\_User] ON [dbo].[User]

FOR INSERT

AS

BEGIN

DECLARE @username varchar(20), @tmp int

SELECT @username = ne.us\_username FROM inserted ne

SELECT @tmp = COUNT(\*) FROM [dbo].[User] WHERE us\_username = @username

IF (@tmp > 0)

BEGIN

PRINT N'TÀI KHOẢN ĐÃ TỒN TẠI'

ROLLBACK TRANSACTION

END

END

GO

1.4. Kiểm tra trùng sản phẩm khi thêm, cập nhật một sản phẩm mới

CREATE TRIGGER [dbo].[Check\_Product] ON [dbo].[Product]

FOR INSERT

AS

BEGIN

DECLARE @name nvarchar(50), @tmp int

SELECT @name = ne.pro\_name FROM inserted ne

SELECT @tmp = COUNT(\*)

FROM [dbo].[Product]

WHERE pro\_name = @name

IF(@tmp > 1)

BEGIN

PRINT N'SAN PHAM DA TON TAI'

ROLLBACK TRANSACTION

END

END

GO

1.5. Kiểm tra trùng danh mục khi thêm, cập nhật một danh mục mới

CREATE TRIGGER [dbo].[Check\_Categories] ON [dbo].[Categories]

FOR INSERT, UPDATE

AS

BEGIN

DECLARE @name nvarchar(50), @tmp int

SELECT @name = ne.cate\_name FROM inserted ne

SELECT @tmp = COUNT(\*)

FROM [dbo].[Categories]

WHERE cate\_name = @name

IF (@tmp > 1)

BEGIN

PRINT N'DANH MUC DA TON TAI'

ROLLBACK TRANSACTION

END

END

GO

1.6. Kiểm tra trùng nhà sản xuất khi thêm, cập nhật một nhà sản xuất mới

CREATE TRIGGER [dbo].[Check\_Supplier] ON [dbo].[Supplier]

FOR INSERT, UPDATE

AS

BEGIN

DECLARE @name nvarchar(50), @tmp int

SELECT @name = ne.sup\_name FROM inserted ne

SELECT @tmp = COUNT(\*)

FROM [dbo].[Supplier]

WHERE sup\_name = @name

IF (@tmp > 1)

BEGIN

PRINT N'NHA SAN XUAT DA TON TAI'

ROLLBACK;

END

END

GO

## 2. Danh sách các View

2.1. View báo cáo doanh thu

CREATE VIEW [View\_BaoCaoDoanhThu] AS

SELECT [User].us\_username, totalprice, createdate

FROM [dbo].[User], [dbo].[Transaction]

WHERE [User].us\_username = [Transaction].us\_username

GO

2.2. View thống kê các sản phẩm tồn kho trong cửa hàng

CREATE VIEW [View\_ThongKeSanPhamTheoDanhMuc] AS

SELECT cate\_name, sup\_name, pro\_name, pro\_quantity, pro\_firstprice, pro\_price, pro\_nsx, pro\_hsd, pro\_description

FROM [dbo].[Product], [dbo].[Categories], [dbo].[Supplier]

WHERE Categories.id = Product.categoriesID AND Supplier.id = Product.supplierID

GO

2.3. View thống kê danh sách các mặt hàng mà người dùng đã đặt mua

CREATE VIEW [View\_DanhSachHangDaDat] AS

SELECT us\_firstname, us\_lastname, pro\_name, or\_quantity, us\_address,us\_phone

FROM [dbo].[User], [dbo].[Product], [dbo].[Order]

WHERE [User].us\_username = [Order].us\_username AND [Order].productID = [Product].id

GO

## 3. Danh sách các Procedure

3.1. Thêm, xoá, sửa và các thao tác trên User

* + - * Thêm:

CREATE PROC sp\_InsertUser

@username varchar(20),

@password varchar(20),

@firstname nvarchar(50),

@lastname nvarchar(50),

@address nvarchar(50),

@birthday date,

@gender char(1),

@phone varchar(11),

@email varchar(50),

@role nvarchar(20) = 'KH'

AS

BEGIN

INSERT INTO [dbo].[User]

VALUES(@username, @password, @firstname, @lastname, @address, @birthday, @gender, @phone, @email, @role)

END

GO

* + - * Sửa:
  + Admin thực hiện update

CREATE PROC sp\_AdUpdateUser

@username varchar(20),

@firstname nvarchar(50),

@lastname nvarchar(50),

@address nvarchar(50),

@birthday date,

@gender char(1),

@phone varchar(11),

@email varchar(50),

@role nvarchar(20)

AS

BEGIN

UPDATE [dbo].[User]

SET

us\_firstname = @firstname,

us\_lastname = @lastname,

us\_address = @address,

us\_birthday = @birthday,

us\_gender = @gender,

us\_phone = @phone,

us\_email = @email,

us\_role = @role

WHERE us\_username = @username

END

GO

* + User thực hiện update:

CREATE PROC sp\_UsUpdateInfoUser

@username varchar(20),

@firstname nvarchar(50),

@lastname nvarchar(50),

@address nvarchar(50),

@birthday date,

@gender char(1),

@phone varchar(11),

@email varchar(50)

AS

BEGIN

UPDATE [dbo].[User]

SET

us\_firstname = @firstname,

us\_lastname = @lastname,

us\_address = @address,

us\_birthday = @birthday,

us\_gender = @gender,

us\_phone = @phone,

us\_email = @email

WHERE us\_username = @username

END

GO

* + - * Xoá:

CREATE PROC sp\_AdDeleteUser

@username varchar(20)

AS

BEGIN

DELETE [dbo].[User]

WHERE us\_username = @username

END

GO

* + - * Tìm kiếm thông tin user:

CREATE PROC sp\_SearchInfoUser

@username varchar(20)

AS

BEGIN

SELECT \* FROM [dbo].[User]

WHERE us\_username = @username

END

GO

* Thay đổi password:

CREATE PROC sp\_ChangePassWord

@username varchar(20),

@password\_old varchar(20),

@password\_new varchar(20)

AS

BEGIN

DECLARE @us\_password varchar(20)

SET @us\_password = (SELECT us\_password FROM [dbo].[User] WHERE us\_username = @username)

IF (@us\_password = @password\_old)

BEGIN

IF(@us\_password != @password\_new)

BEGIN

UPDATE [dbo].[User] SET us\_password = @password\_new

WHERE us\_username = @username

END

END

END

GO

* Check Login:

CREATE PROC CheckLogin

@username varchar(20),

@password varchar(20)

AS

BEGIN

SELECT us\_username, us\_password, us\_role

FROM [dbo].[User]

WHERE us\_username = @username and us\_password = @password

END

GO

3.2. Thêm, xoá, sửa và các thao tác trên Sản phẩm

* Thêm sản phẩm:

CREATE PROC sp\_InsertProduct

@categoriesID int,

@supplierID int,

@name nvarchar(50),

@quantity int,

@firstprice real,

@price real,

@nsx date,

@hsd date,

@description nvarchar(100)

AS

BEGIN

INSERT INTO [dbo].[Product]

VALUES(@categoriesID, @supplierID, @name, @quantity,

@firstprice, @price, @description, @nsx, @hsd)

END

GO

* Sửa sản phẩm:

CREATE PROC sp\_UpdateProduct

@id int,

@categoriesID int,

@supplierID int,

@name nvarchar(50),

@quantity int,

@firstprice real,

@price real,

@nsx date,

@hsd date,

@description nvarchar(100)

AS

BEGIN

UPDATE [dbo].[Product]

SET categoriesID = @categoriesID,

supplierID = @supplierID,

pro\_name = @name,

pro\_quantity = @quantity,

pro\_firstprice = @firstprice,

pro\_price = @price,

pro\_nsx = @nsx,

pro\_hsd = @hsd,

pro\_description = @description

WHERE id = @id

END

GO

* Xoá sản phẩm:

CREATE PROC sp\_DeleteProduct

@id int

AS

BEGIN

DELETE [dbo].[Product]

WHERE id = @id

END

GO

* Thêm một giao dịch:

CREATE PROC sp\_InsertTransaction

@us\_username varchar(20),

@address nvarchar(MAX),

@phone char(11),

@totalprice real,

@createdate date,

@note nvarchar(100)

AS

BEGIN

INSERT INTO [dbo].[Transaction]

VALUES(@us\_username, @address, @phone, @totalprice, @createdate, @note)

END

* Thêm một sản phẩm vào giao dịch:

CREATE PROC sp\_InsertOrder

@us\_username varchar(20),

@transactionID int,

@productID int,

@quantity int,

@price real

AS

BEGIN

INSERT INTO [dbo].[Order]

VALUES(@us\_username, @transactionID, @productID, @quantity, @price)

END

* Sửa số lượng của sản phẩm trong giao dịch:

CREATE PROC sp\_UpdateOrder

@id int,

@quantity int

AS

BEGIN

UPDATE [dbo].[Order]

SET or\_quantity = @quantity

WHERE id = @id

END

* Xoá một sản phẩm trong giao dịch:

CREATE PROC sp\_DeleteOrder

@id int

AS

BEGIN

DELETE [dbo].[Order]

WHERE id = @id

END