

**TRƯỜNG ĐẠI HỌC KINH TẾ - ĐẠI HỌC ĐÀ NẴNG**  
**KHOA THỐNG KÊ - TIN HỌC**



**BÁO CÁO DỰ ÁN**  
**HỆ THỐNG**  
**QUẢN LÝ BÁN TRÀ SỮA**  
**MÔN: QUẢN TRỊ CƠ SỞ DỮ LIỆU**

GV hướng dẫn: **Th.S Cao Thị Nhâm**

Sinh viên thực hiện: **Trần Thị Diệu Linh**

**Hà Thị Mỹ Hạnh**

**Trương Thị Kiều Diễm**

**Lương Ngọc Hoàng**

Lớp: **44k14**

*Đà Nẵng, tháng 11/2020*

## MỤC LỤC

<b>BÁO CÁO DỰ ÁN</b>	<b>1</b>
<b>MỤC LỤC</b>	<b>2</b>
<b>CHƯƠNG I: MÔ TẢ BÀI TOÁN</b>	<b>3</b>
<b>I. Tóm tắt bài toán</b>	<b>3</b>
1. Tóm tắt hoạt động của hệ thống	3
2. Các thông tin sử dụng trong hệ thống	3
<b>II. Dự định của hệ thống</b>	<b>3</b>
1. Tên hệ thống	3
2. Môi trường triển khai	3
<b>III. Mô tả các yêu cầu của hệ thống</b>	<b>4</b>
1. Cơ cấu quản lý của hệ thống	4
<b>CHƯƠNG II: THIẾT KẾ CƠ SỞ DỮ LIỆU</b>	<b>6</b>
<b>I. Thiết kế cơ sở dữ liệu:</b>	<b>6</b>
1. Thiết kế CSDL ở mức khái niệm:	6
a. Chọn lọc thông tin:	6
b. Xác định thực thể và thuộc tính:	7
c. Xác định mối quan hệ:	7
d. Vẽ mô hình ER:	8
2. Thiết kế CSDL mức logic:	8
<b>II. Xây dựng CSDL:</b>	<b>9</b>
1. Tạo database:	9
<b>CHƯƠNG 3: XÂY DỰNG CƠ CHẾ BẢO MẬT CHO CSDL</b>	<b>11</b>
<b>1. Phân quyền người dùng:</b>	<b>11</b>
<b>2. Tạo các hàm, thủ tục, trigger:</b>	<b>12</b>
a. Thủ tục thêm tài khoản mới trong TaiKhoan	12
b. Thủ tục sửa tài khoản trong TaiKhoan	13
c. Trigger xóa tài khoản trong TaiKhoan	13
d. Hàm tạo mã món mới trong Mon	14
e. Thủ tục thêm món mới trong	15
f. Thủ tục sửa món trong MON	16
g. Trigger xóa món trong MON	16
<b>CHƯƠNG 4: SAO LƯU VÀ PHỤC HỒI</b>	<b>17</b>

# CHƯƠNG I: MÔ TẢ BÀI TOÁN

## I. Tóm tắt bài toán

### 1. Tóm tắt hoạt động của hệ thống

- Xây dựng hệ thống quản lý bán trà sữa.
- Mô tả môi trường: hệ thống sẽ thực hiện quản lý thông tin các tài khoản, các loại nước có trong menu, hóa đơn bán hàng và chi tiết hóa đơn bán hàng.
- Việc quản lý bán trà sữa như sau: lấy mã hóa đơn bán hàng làm khóa chính để quản lý thông tin liên quan đến hóa đơn như tài khoản nhân viên bán hàng, chi tiết hóa đơn bán hàng (số lượng và thành tiền của các loại nước), thông tin về loại nước mà khách hàng đã đặt.
- Các chức năng chính của hệ thống:
  - + Xem (hiển thị)
  - + Thêm mới
  - + Chỉnh sửa
  - + Tìm kiếm
  - + Xóa

### 2. Các thông tin sử dụng trong hệ thống

Các thông tin chi tiết về hóa đơn bán hàng, các loại nước, tài khoản của nhân viên.

## II. Dự định của hệ thống

- Đây là phần mềm tiện ích trong việc quản lý bán trà sữa trong các tiệm trà sữa, cà phê.
- Xây dựng hệ thống với chức năng chính là chức năng quản trị và tạo hóa đơn bán hàng. Người quản lý phải đăng nhập vào hệ thống để sử dụng các chức năng quản trị, có quyền thêm, sửa, xóa, tìm kiếm các thông tin liên quan đến hóa đơn, tài khoản nhân viên, các loại đồ uống. Nhân viên chỉ được chỉnh sửa, tạo và xóa hóa đơn, đồ uống.

### 1. Tên hệ thống

- Hệ thống quản lý bán trà sữa.

### 2. Môi trường triển khai

- Visual Studio 2013.
- Hệ quản trị cơ sở dữ liệu: SQL Server.
- Ngôn ngữ lập trình C#.
- **LINK DỰ ÁN CỦA NHÓM:** <https://bom.to/cHpa2Mh>

### III. Mô tả các yêu cầu của hệ thống

#### 1. Cơ cấu quản lý của hệ thống

##### ***Chức năng quản lý chính: quản lý hóa đơn bán hàng***

- Mục đích: tạo hóa đơn bán hàng và quản lý bán hàng một cách hiệu quả hơn.
- Tóm lược: quản trị viên hoặc nhân viên đăng nhập vào hệ thống bằng tài khoản và mật khẩu của mình. Hệ thống kiểm tra thấy mật khẩu là đúng, sau đó chọn vào form HÓA ĐƠN. Nhân viên hoặc quản lý có thể thêm, sửa, xóa, tìm kiếm thông tin theo mã hóa đơn.
- Đầu vào:
  - Mã hóa đơn bán
  - Mã tài khoản
  - Tên món
  - Số lượng
  - Ngày bán
- Đầu ra:
  - Thông tin chi tiết về hóa đơn.
- Sự cố xảy ra:
  - Mật khẩu đưa vào không đúng. Nhân viên hoặc người quản lý nhập lại tài khoản và mật khẩu hoặc là kết thúc sử dụng.

##### ***Quản lý tài khoản***

- Mục đích: quản lý các tài khoản đăng nhập của các nhân viên. Theo dõi quá trình làm việc dựa vào các hóa đơn được nhân viên lập ra.
- Tóm lược: quản trị viên bằng tài khoản và mật khẩu của mình. Sau khi đăng nhập thành công thì hệ thống cho phép quản trị viên thêm, sửa, xóa, tìm kiếm thông tin theo mã tài khoản.
- Đầu vào:
  - Mã tài khoản
  - Mật khẩu
  - Họ tên
  - Số điện thoại
  - Chức vụ
  - Địa chỉ
- Đầu ra:
  - Danh sách các tài khoản của nhân viên.
  - Thông tin chi tiết về nhân viên.
- Sự cố xảy ra:
  - Mật khẩu đưa vào không đúng. Người quản lý nhập lại tài khoản và mật khẩu hoặc là kết thúc sử dụng.

##### ***Quản lý món***

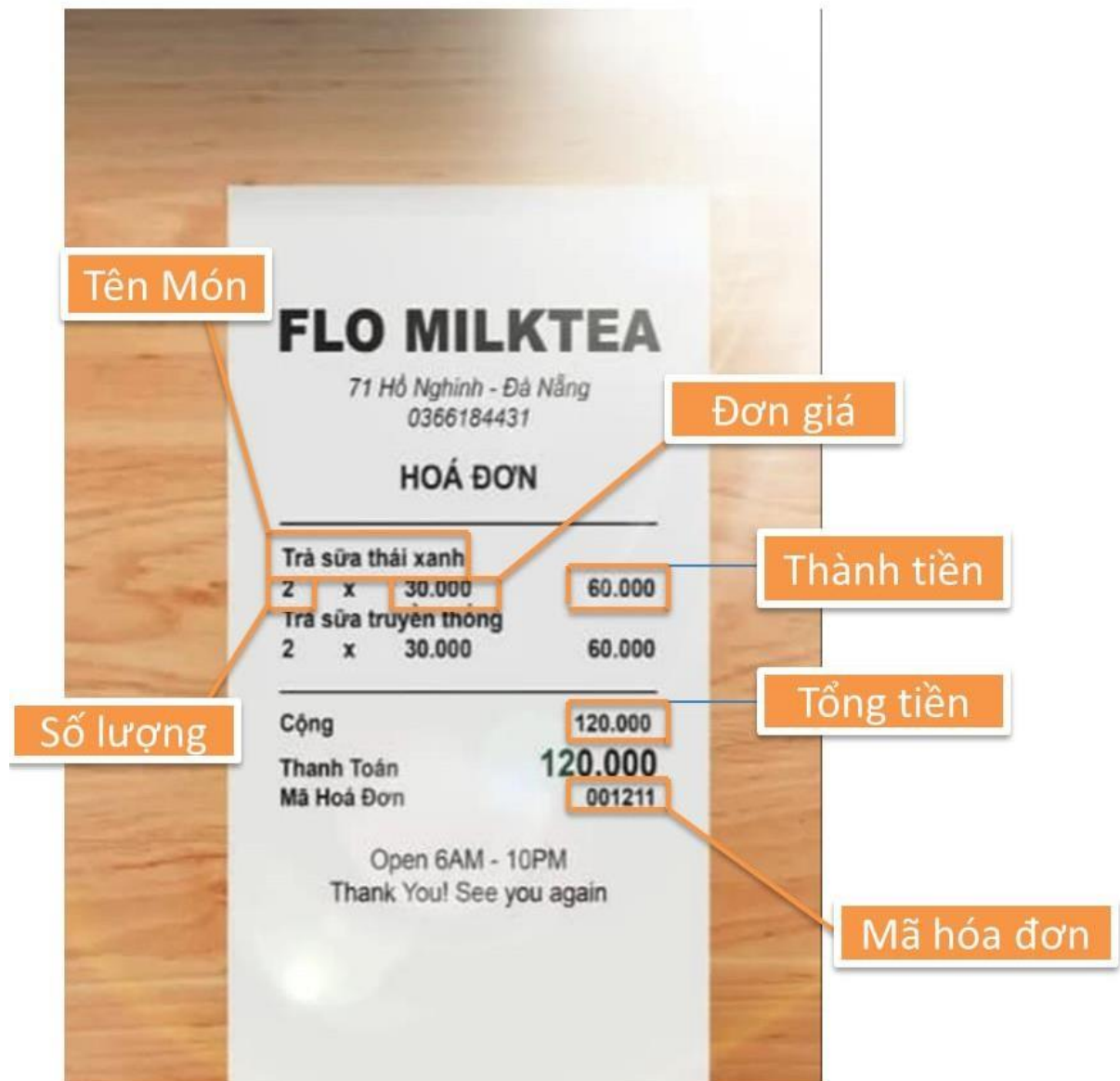
- Mục đích: quản lý các đồ uống có trong quán.
- Tóm lược: Khi nhà quản trị và nhân viên đăng nhập thành công, hệ thống sẽ cho phép thao tác với các chức năng thêm, sửa, xóa, tìm kiếm thông tin món ăn.
- Đầu vào:
  - Mã món
  - Tên món
  - Đơn vị tính
  - Đơn giá
- Đầu ra:
  - Thông tin chi tiết về các đồ uống trong thực đơn.
- Sự cố xảy ra:
  - Mật khẩu đưa vào không đúng. Nhân viên hoặc người quản lý nhập lại tài khoản và mật khẩu hoặc là kết thúc sử dụng.

## CHƯƠNG II: THIẾT KẾ CƠ SỞ DỮ LIỆU

### I. Thiết kế cơ sở dữ liệu:

#### 1. Thiết kế CSDL ở mức khái niệm:

##### a. Chọn lọc thông tin:



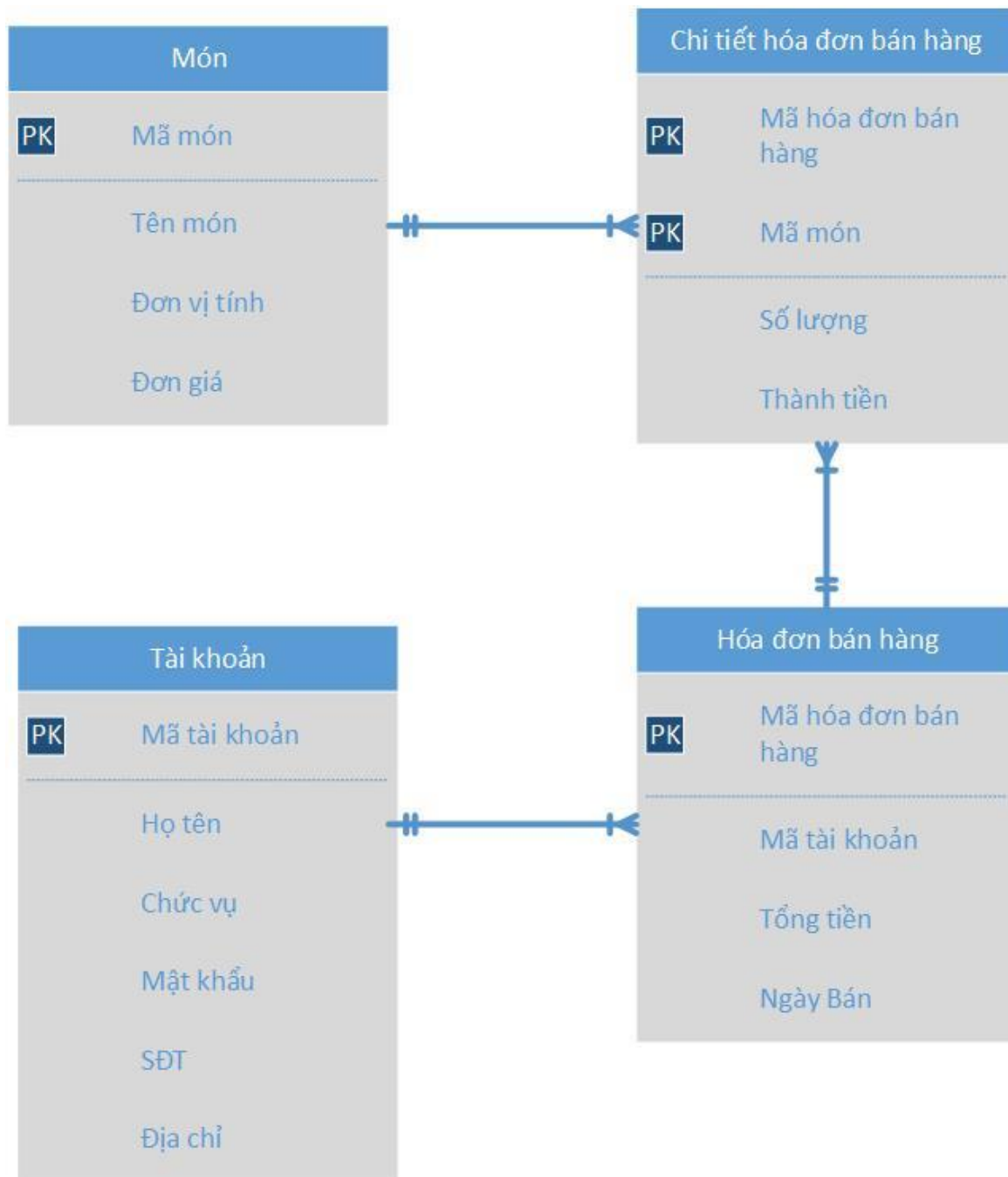
**b. Xác định thực thể và thuộc tính:**

- Tài Khoản:
  - + Mã tài khoản
  - + Mật khẩu
  - + Họ tên
  - + Số điện thoại
  - + Chức vụ
  - + Địa chỉ: nvarchar(200)
- Món:
  - + Mã món
  - + Tên món
  - + Đơn vị tính
  - + Đơn giá
- Hóa Đơn Bán Hàng:
  - + Mã hóa đơn bán
  - + Mã tài khoản
  - + Tổng tiền
  - + Ngày bán

**c. Xác định mối quan hệ:**

- Do bảng hóa Món và bảng Hoá Đơn Bán Hàng có mối quan hệ nhiều - nhiều nên sinh ra một bảng mới là bảng Chi Tiết Hóa Đơn Bán Hàng.
- Bảng Chi tiết hóa đơn Bán hàng gồm các thực thể và thuộc tính sau:
  - + Mã hóa đơn bán: char(6)
  - + Mã món: char(6)
  - + Số lượng: int
  - + Thành tiền: numeric(15,0)

**d. Vẽ mô hình ER:**

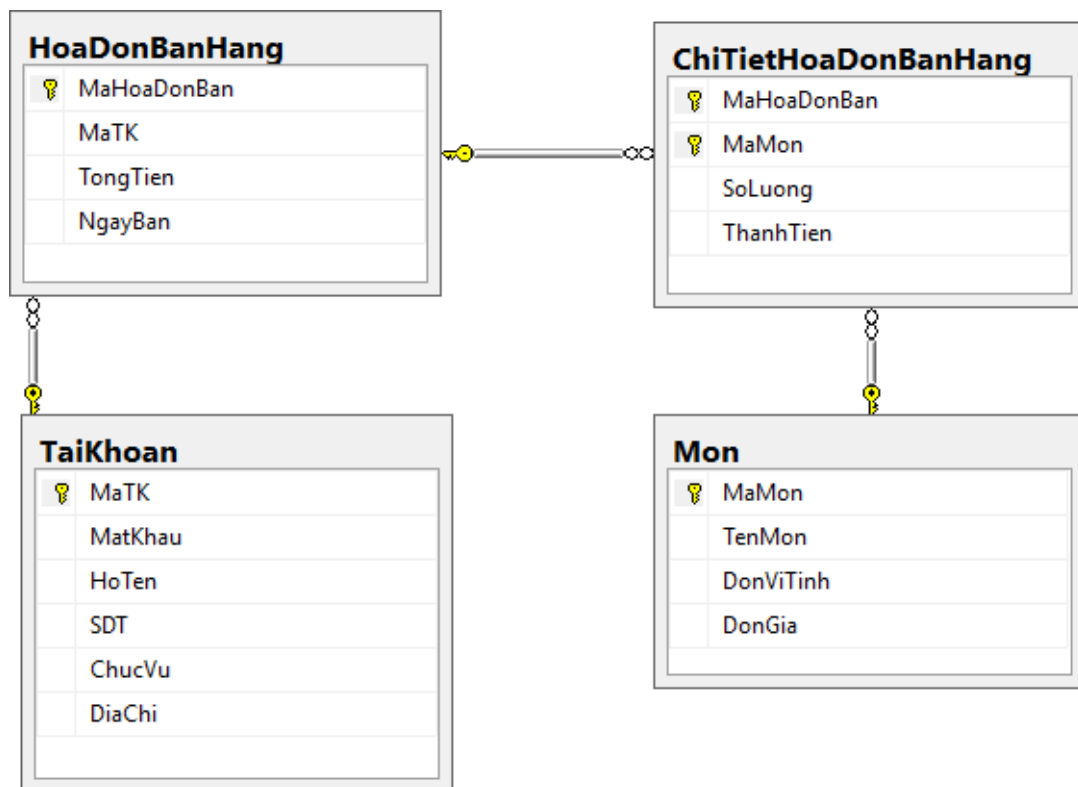


**2. Thiết kế CSDL mức logic:**

- Chuyển thực thể, thu được:
  - + TaiKhoan (**MaTK**, MatKhau, HoTen, SĐT, ChucVu, DiaChi)
  - + Mon (**MaMon**, TenMon, DonViTinh, DonGia)
  - + HoaDonBanHang (**MaHoaDonBan**, MaTK, TongTien, NgayBan)
- Chuyển mối quan hệ, thu được:
  - + ChiTietHoaDonBanHang ( **MaHoaDonBan**, **MaMon**, SoLuong, ThanhTien)



- Vẽ sơ đồ quan hệ:



## II. Xây dựng CSDL:

- **Tạo database:**

Create database BanTraSua

Use BanTraSua

- **Tạo bảng Mon:**

Create table Mon

```
(
    MaMon char(6) not null primary key,
    TenMon nvarchar(100) not null,
    DonViTinh nvarchar(50) not null,
    DonGia numeric(15,0) not null,
)
```

- **Tạo bảng TaiKhoan:**

Create table TaiKhoan

```
(
    MaTK varchar(50) not null primary key,
    MatKau varchar(100) not null,
    HoTen nvarchar(50) not null,
    SDT varchar(15) not null,
    ChucVu nvarchar(50) not null,
```

```

        DiaChi nvarchar(200) not null
    )

    ● Tạo bảng HoaDonBanHang:
    Create table HoaDonBanHang
    (
        MaHoaDonBan char(6) not null primary key,
        MaTK varchar(50) not null,
        TongTien numeric(15,0),
        NgayBan datetime,
        foreign key (MaTK) references TaiKhoan
    )

```

```

    ● Tạo bảng ChiTietHoaDonBanHang:
    Create table ChiTietHoaDonBanHang
    (
        MaHoaDonBan char(6) not null,
        MaMon char (6) not null,
        SoLuong int not null,
        ThanhTien numeric(15,0) ,
        Constraint Pk_ChiTietHoaDonBanHang Primary key (MaHoaDonBan,
        MaMon),
        foreign key (MaMon) references Mon,
        foreign key (MaHoaDonBan) references HoaDonBanHang
    )

```

## CHƯƠNG 3: XÂY DỰNG CƠ CHẾ BẢO MẬT CHO CSDL

Xây dựng cơ chế bảo mật cho CSDL bằng cách xác thực người dùng khi kết nối vào CSDL và phân quyền cho người dùng khi đã vào được database. Ngoài ra sử dụng các hàm, thủ tục và trigger để phòng chống sql injection

### Ví dụ:

Một chức năng đăng nhập.

Request gửi một username và password tới server.

Bạn dùng username và password này để truy vấn trực tiếp trong CSDL.

Câu lệnh như sau: `SELECT * FROM user WHERE username='username' AND password='password'`

Nếu kẻ tấn công thay username thành `' OR '1'='1'`.

Câu truy vấn lúc này sẽ thành `SELECT * FROM user WHERE username='' OR '1'='1' AND password='abc'`.

Lúc này mọi điều kiện đều hợp lệ và thế là truy vấn trả về kết quả đúng.

Hãy tưởng tượng kẻ phá hoại chen những đoạn mã xóa bảng, CSDL thì đó thật nguy hiểm

=> Đây là cách phòng chống sql injection

### 1. Phân quyền người dùng:



## **2. Tạo các hàm, thủ tục, trigger:**

### **a. Thủ tục thêm tài khoản mới trong TaiKhoan**

```
create proc spInsertTaiKhoan(  
    @MaTK varchar(50),  
    @MatKhau varchar(50),  
    @HoTen nvarchar(100),  
    @SDT varchar(15),  
    @ChucVu nvarchar(50),  
    @DiaChi nvarchar(200))  
  
as  
  
begin  
    declare @checkMaTK int = 0  
  
    Set @checkMaTK = (Select COUNT(MaTK) from TaiKhoan where MaTK =  
@MaTK)  
  
    if(@checkMaTK != 0)  
    begin  
        return  
    end  
  
    insert into TaiKhoan(MaTK,MatKhau,HoTen,SDT,ChucVu,DiaChi)  
    values (@MaTK,@MatKhau,@HoTen,@SDT,@ChucVu,@DiaChi)  
  
end
```

### **b. Thủ tục sửa tài khoản trong TaiKhoan**

```
create proc TaiKhoan_Update(  
    @MaTK varchar(50),  
    @MatKhau varchar(100),  
    @HoTen nvarchar(50),  
    @SDT varchar(15),  
    @ChucVu nvarchar(50),  
    @DiaChi nvarchar(200))  
  
as  
  
begin  
    update TaiKhoan set MatKhau=@MatKhau, HoTen=@HoTen, SDT=@SDT,  
    DiaChi=@DiaChi, ChucVu=@ChucVu where MaTK=@MaTK  
  
end
```

### **c. Trigger xóa tài khoản trong TaiKhoan**

```
Create trigger tDeleteTaiKhoan  
  
on TaiKhoan  
  
Instead of delete  
  
as  
  
begin  
    update TaiKhoan  
    set ChucVu = N'Không'  
  
    from deleted  
  
    Where TaiKhoan.MaTK = deleted.MaTK  
  
end
```

**d. Hàm tạo mã món mới trong Mon**

Create function fTaoMaMon()

returns char(6)

As

Begin

Declare @MaMon char(6)

Set @MaMon = (Select MAX(MaMon) from Mon)

if(@MaMon is Null)

begin

Set @MaMon = 1

set @MaMon=( REPLICATE('0',6-LEN(RTRIM(@MaMon))) +  
RTRIM(@MaMon))

return @MaMon

end

Set @MaMon = @MaMon + 1

set @MaMon=( REPLICATE('0',6-LEN(RTRIM(@MaMon))) +  
RTRIM(@MaMon))

return @MaMon

end

**e. Thủ tục thêm món mới trong**

```
Create proc spInsertMon(@TenMon nvarchar(100),@DonGia  
numeric(15,0),@DonViTinh nvarchar(50))
```

```
as
```

```
begin
```

```
    declare @checkMaMon int, @MaMon char(6)
```

```
    set @MaMon = (Select dbo.fTaoMaMon())
```

```
    Set @checkMaMon = (select COUNT(MaMon) from Mon where MaMon =  
@MaMon)
```

```
    if(@checkMaMon > 1)
```

```
    begin
```

```
        print N'Mã món đã tồn tại !'
```

```
        return
```

```
    end
```

```
    if(@DonGia <= 0)
```

```
    begin
```

```
        print N'Dơn giá không hợp lệ !'
```

```
        return
```

```
    end
```

```
    insert into Mon(MaMon,TenMon,DonGia,DonViTinh)
```

```
    values (@MaMon,@TenMon,@DonGia,@DonViTinh)
```

```
    if(@@ROWCOUNT <= 0)
```

```
    begin
```

```
        return
    end
end
```

**f. Thủ tục sửa món trong MON**

```
alter proc Menu_Update(
    @MaMon char(6),
    @TenMon nvarchar(100),
    @DonViTinh nvarchar(50),
    @DonGia numeric(15,0))
as
begin
    update Mon set TenMon=@TenMon, DonGia=@DonGia,
    DonViTinh=@DonViTinh where MaMon=@MaMon
end
```

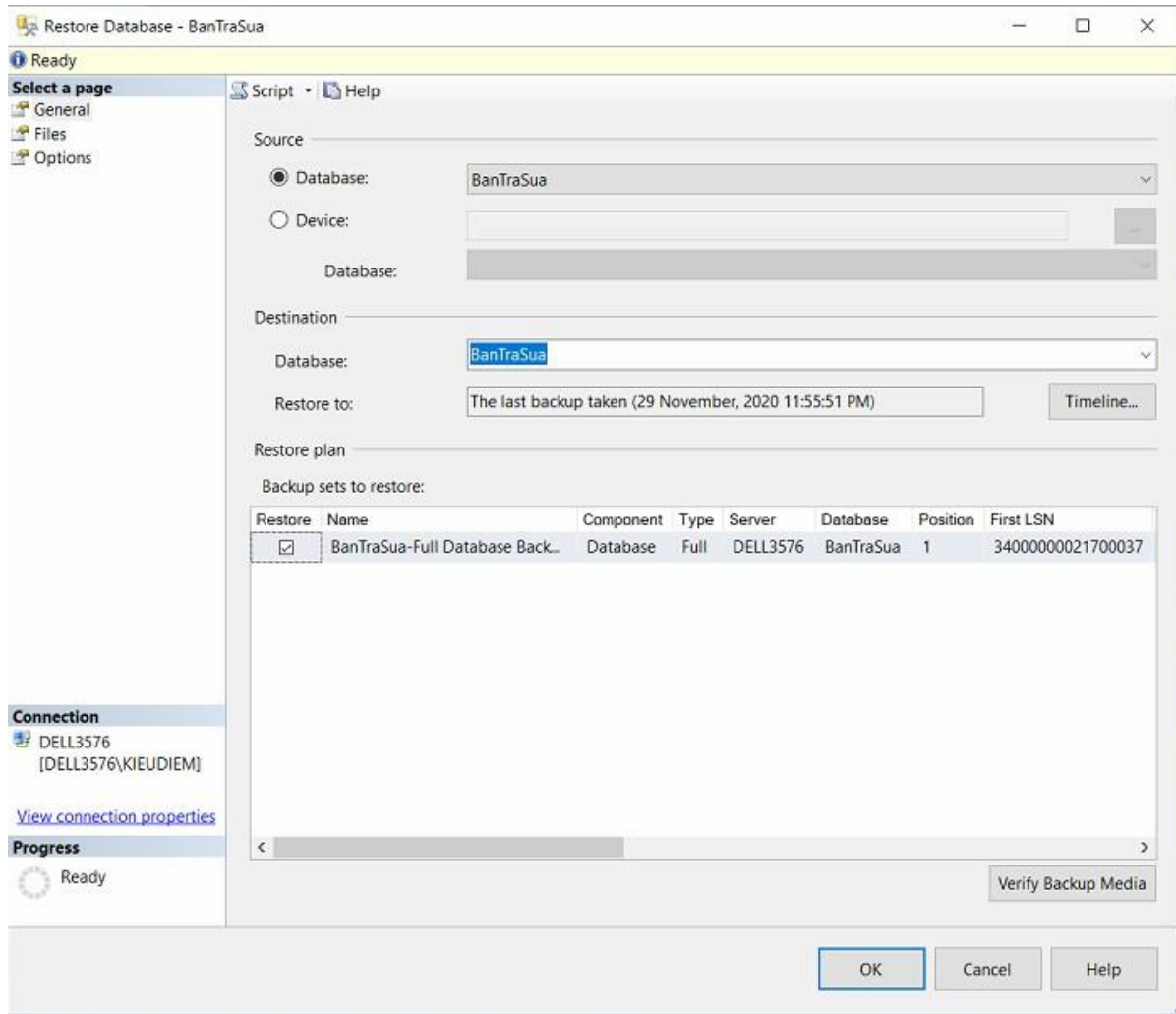
**g. Trigger xóa món trong MON**

```
Create trigger tDeleteMon
on Mon
Instead of delete
as
begin
    update Mon
    set DonGia = -1
    from deleted
    Where Mon.MaMon = deleted.MaMon
```



end

## CHƯƠNG 4: SAO LƯU VÀ PHỤC HỒI



New Job

Select a page

General

Steps

Schedules

Alerts

Notifications

Targets

Connection

Server:

DELL3576

Connection:

DELL3576\KIEUDIEM

View connection

Progress

Ready

Script Help

Job step list:

S...	Name	Type	On Su...	On Fail...
1	BackupBTS	Trans...	Go to t...	Quit th...

Move step:

Start step:

1:BackupBTS

New...

Insert...

Edit

Delete

OK

Cancel

**Job Schedule Properties - BackupBanTraSua**

Name:  Jobs in Schedule

Schedule type: Recurring ☒ Enabled

One-time occurrence  
 Date:  Time:

Frequency  
 Occurs: Daily  
 Recurs every:  day(s)

Daily frequency  
☒ Occurs once at:   
☐ Occurs every:  hour(s) Starting at:   
 Ending at:

Duration  
 Start date:  ☐ End date:   
☒ No end date

Summary  
 Description:

- Trong trường hợp không đủ không gian lưu trữ cho cơ sở dữ liệu vì ổ đĩa đầy thì nhóm thực hiện tự động backup database lên google drive.

