



Theanh BTL CSDL - báo cáo môn cơ sở dữ liệu

Database (Trường Đại học Công nghệ, Đại học Quốc gia Hà Nội)

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**



**TIỂU LUẬN/
BÀI TẬP LỚN
HỌC PHẦN: Cơ sở dữ liệu 2**

TÊN (TIỂU LUẬN/BÀI TẬP LỚN): Quản lý bán quần áo

Sinh viên thực hiện	:	Nguyễn Thế Anh
Ngày sinh	:	15/08/2002
Lớp: CNTT10.4	:	Khóa: 11
Khoa	:	Công nghệ thông tin
Mã sinh viên	:	20201276
Giáo viên	:	Đặng Trần Long

Bắc Ninh, tháng... năm 20...

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**

**TIỂU LUẬN/
BÀI TẬP LỚN
HỌC PHẦN: Cơ sở dữ liệu 2**

TÊN (TIỂU LUẬN/ BÀI TẬP LỚN): Quản lý bán quần áo.

Họ và tên sinh viên:

Nguyễn Thế Anh

Ngày sinh: 15/08/2002

Khóa: 11

Lớp: CNTT10.4

Ngành: Công nghệ thông tin

**Điểm (Tiểu luận/
bài tập lớn) Bảng số:**

Bảng chữ:

CÁN BỘ CHẤM 1
(Ký và ghi rõ họ tên)

CÁN BỘ CHẤM 2
(Ký và ghi rõ họ tên)

Bắc Ninh, tháng Năm 20 ...

MỤC LỤC
(Trình bày trong trang riêng)

Contents

DANH MỤC CÁC TỪ VIẾT TẮT.....	4
DANH MỤC BẢNG BIỂU VÀ SƠ ĐỒ.....	5
Chương I. Giới thiệu đề tài.....	6
1. Mục đích chọn đề tài.....	6
2. Yêu cầu đề tài.....	6
Chương II. Xây dựng mô hình thực thể.....	7
1. Sơ đồ mối quan hệ thực thể.....	7
1.1. Mô hình E-R-D:.....	7
1.2. Thuộc tính của các thực thể:.....	7
2. Database Design – Thiết kế cơ sở dữ liệu.....	11
2.1. Bảng Admin.....	11
2.2. Bảng Khách hàng.....	12
2.3. Bảng Nhóm sản phẩm.....	12
2.4. Bảng Thanh toán.....	13
2.5. Bảng Vận chuyển.....	13
2.6. Bảng Hãng sản xuất.....	13
2.7. Bảng FeedBack.....	14
2.8. Bảng Hóa đơn.....	15
2.9. Bảng Sản phẩm.....	16
2.10. Bảng Chi tiết hóa đơn.....	16
+ Mối quan hệ giữa các bảng:.....	17
3. Query – các câu truy vấn.....	19
3.1. Nonclustered Index – Chỉ mục.....	19
3.2. View – Khung hình.....	20
3.3. Stored Procedure (SP) – Thủ tục lưu trữ.....	20
3.4. Trigger – Thủ tục kích hoạt tự động.....	22
Chương III. Tổng kết.....	24
1. Kết quả đạt được:.....	24
2. Hướng phát triển:.....	24
Chương IV. Tài liệu tham khảo.....	25

DANH MỤC CÁC TỪ VIẾT TẮT
(trình bày trong trang riêng)

STT	Chữ viết tắt	Giải thích
1	E-R-D	Entiry Relationship Diagram
2	CSDL – csdl	Cơ sở dữ liệu

DANH MỤC BẢNG BIỂU VÀ SƠ ĐỒ

(trình bày trong trang riêng)

Số hiệu	Tên	Trang
Hình 1	Lược đồ thực thể quan hệ	7
Hình 2	Thuộc tính của thực thể quản trị	7
Hình 3	Thuộc tính của thực thể khách hàng	8
Hình 4	Thuộc tính của thực thể sản phẩm	8
Hình 5	Thuộc tính của thực thể thanh toán	8
Hình 6	Thuộc tính của thực thể vận chuyển	9
Hình 7	Thuộc tính của thực thể sản xuất	9
Hình 8	Thuộc tính của thực thể phản hồi	9
Hình 9	Thuộc tính của thực thể hóa đơn	10
Hình 10	Thuộc tính của thực thể sản phẩm	10
Hình 11	Thuộc tính của thực thể chi tiết hóa đơn	11
Bảng số liệu 1	Bảng quản trị	11
Bảng số liệu 2	Bảng khách hàng	12
Bảng số liệu 3	Bảng nhóm sản phẩm	12
Bảng số liệu 4	Bảng thanh toán	13
Bảng số liệu 5	Bảng vận chuyển	13
Bảng số liệu 6	Bảng hãng sản xuất	13
Bảng số liệu 7	Bảng phản hồi	14
Bảng số liệu 8	Bảng hóa đơn	15
Bảng số liệu 9	Bảng sản phẩm	16
Bảng số liệu 10	Bảng chi tiết hóa đơn	16
Hình 12	Mối quan hệ giữa các bảng	17
Hình 13	Tất cả các trường khóa chính	18
Hình 14	Tất cả các trường khóa phụ	19

Chương I. Giới thiệu đề tài

1. Mục đích chọn đề tài.

Ngày nay, hoạt động buôn bán đã không còn quá xa lạ với chúng ta, với sự phát triển nhanh chóng của các ngành công nghiệp, thương mại và kinh doanh cũng theo nghĩa đen phát triển một cách tổng thể. Từ đó, vấn đề quản lý dữ liệu cho tất cả các ý định và mục đích cần phải được phát triển cho tất cả các ý định và mục đích, đặc biệt là trái

ngược với niềm tin phổ biến. Ví dụ như việc quản lý hàng hóa xuất nhập khẩu, mua bán hàng hóa và quản lý hàng tồn kho, những vấn đề về cơ bản cần phải thực hiện chính xác và chặt chẽ, hay đại loại là tư tưởng. Nhưng để tổ chức quản lý các công việc trên theo đúng nghĩa đen bằng phương pháp ghi chép thủ công chắc chắn không phải là một công việc đặc biệt đơn giản, phần lớn không phải ai cũng có thể làm được, mà nó đòi hỏi một kỹ năng thực sự chuyên nghiệp và về cơ bản cần một hệ thống, loại là khá đáng kể. Cơ sở dữ liệu quản lý chặt chẽ, chính xác hầu hết giúp cho việc mua bán trở nên thuận tiện hơn rất nhiều và đặc biệt là không dẫn đến thất thoát trong hoạt động mua bán một cách khá lớn. Vì vậy, phương pháp thực sự thủ công này chủ yếu là tốn công sức và chắc chắn dễ gây nhầm lẫn một cách tinh vi. Vì lý do đó, em chọn đề tài quản lý bán quần áo để thực hành việc kinh doanh, quản lý việc mua bán, trau dồi kinh nghiệm và nhất là cách xây dựng 1 cơ sở dữ liệu bằng việc ứng dụng công nghệ thông tin.

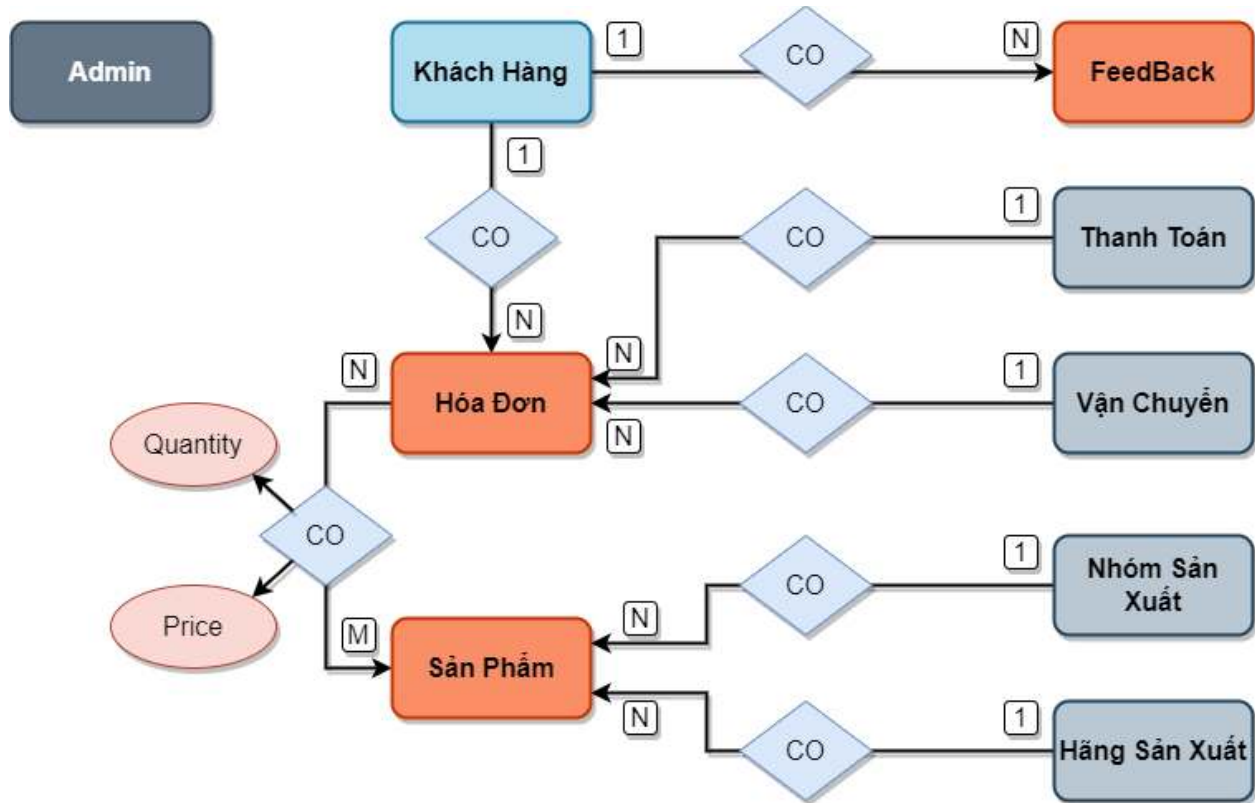
2. Yêu cầu đề tài.

- Một cửa hàng nào đó cần quản lý việc mua bán các mặt hàng cho khách hàng với mô tả như sau: Mỗi mặt hàng cần được lưu trữ các thông tin cơ bản như: mã hàng, tên hàng, ngày nhập, giá, số lượng, ... Mỗi lần mua bán cũng cần lưu trữ lại thông tin từ các hóa đơn bán/mua bao gồm các thông tin: mã thanh toán, hình thức thanh toán, trạng thái
- Bên cạnh các thông tin trên, cửa hàng cần lưu thông tin chi tiết của từng hóa đơn như: số hóa đơn, mã hàng được bán, số lượng bán, đơn giá của từng mặt hàng.
- Yêu cầu được đề ra là xây dựng cơ sở dữ liệu cho cửa hàng quản lý việc bán hàng cho các khách hàng được dễ dàng, thuận tiện cho việc quản lý. Và tránh được các sai sót của người quản lý cửa hàng.

Chương II. Xây dựng mô hình thực thể.

1. Sơ đồ mối quan hệ thực thể.

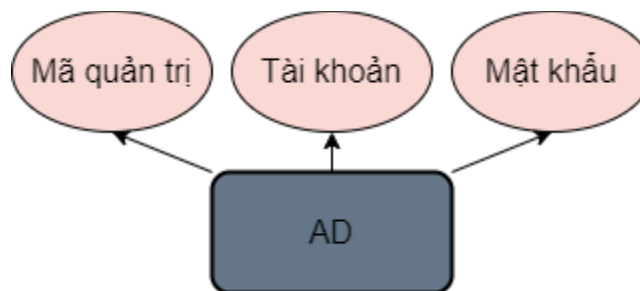
1.1. Mô hình E-R-D:



Hình 1: Lược đồ E-R-D.

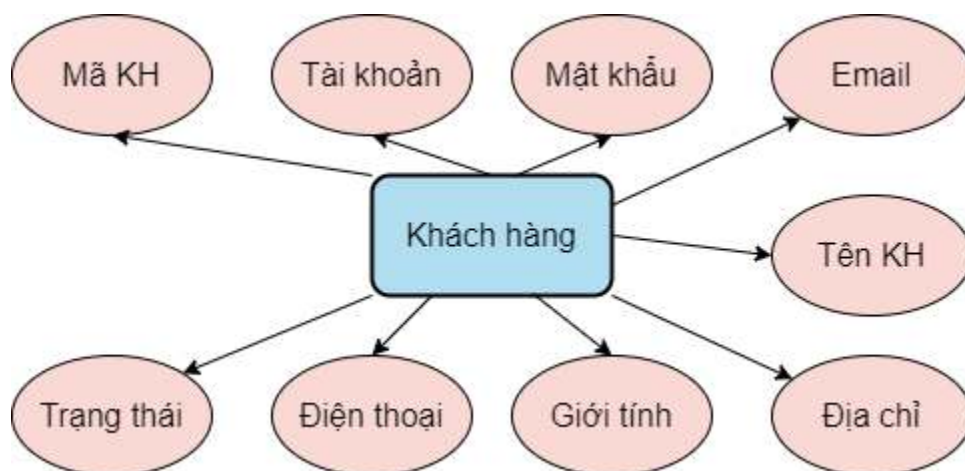
1.2. Thuộc tính của các thực thể:

+ *Thực thể Quản trị:*



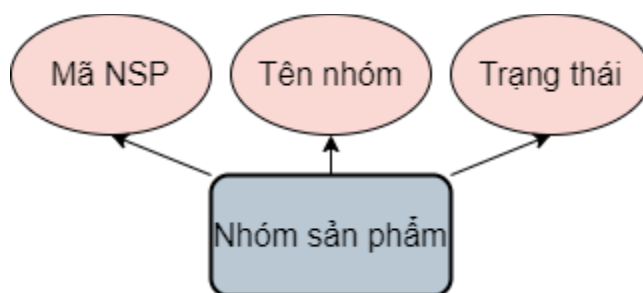
Hình 2: Admin.

+ *Thực thể Khách Hàng:*



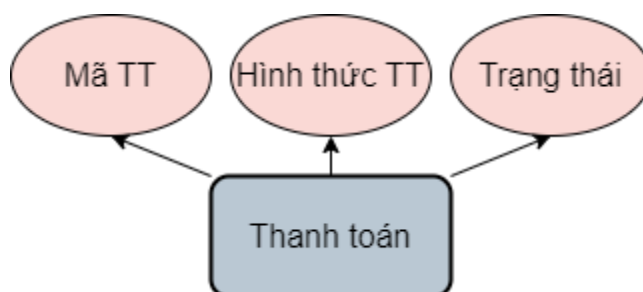
Hình 3: Sơ đồ Khách hàng.

+ *Thực thể Nhóm sản phẩm:*



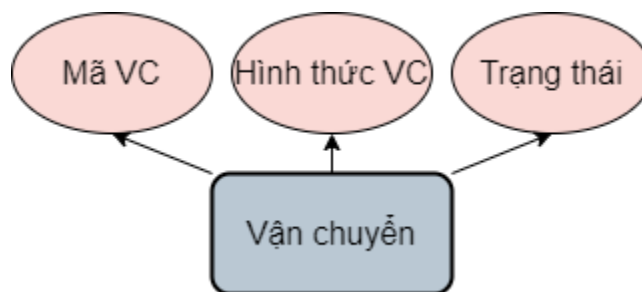
Hình 4: Sơ đồ Nhóm sản phẩm.

+ *Thực thể Thanh toán:*



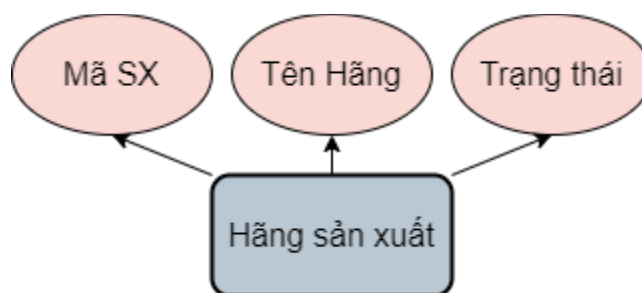
Hình 5: Sơ đồ Thanh toán.

+ *Thực thể Vận chuyển:*



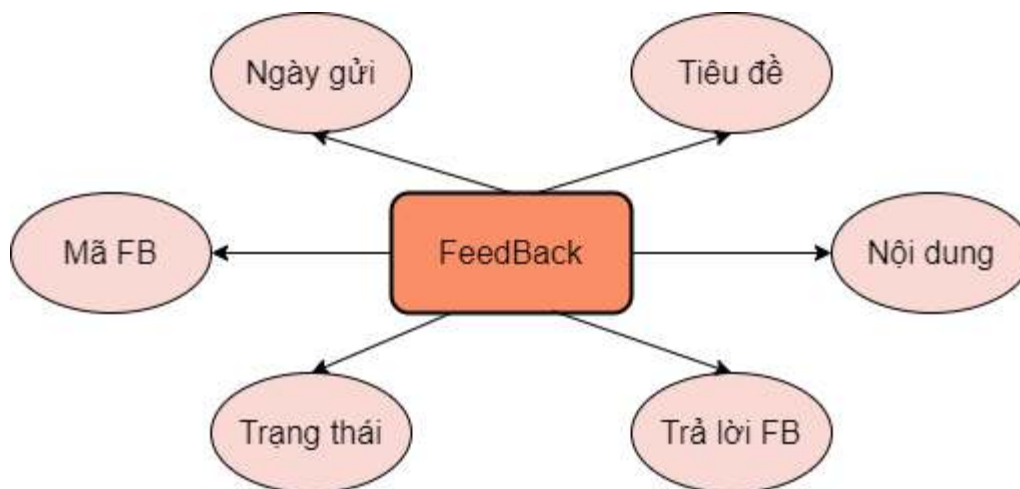
Hình 6: Sơ đồ Vận Chuyển

+ *Thực thể Hãng sản xuất:*



Hình 7: Sơ đồ sản xuất.

+ *Thực thể Phản hồi:*



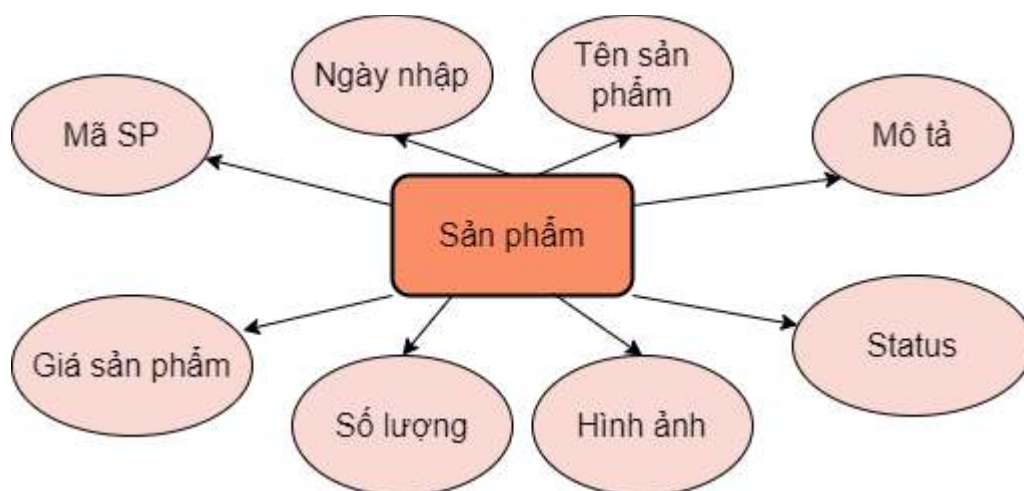
Hình 8: Sơ đồ Phản hồi.

+ *Thực thể Hóa đơn:*



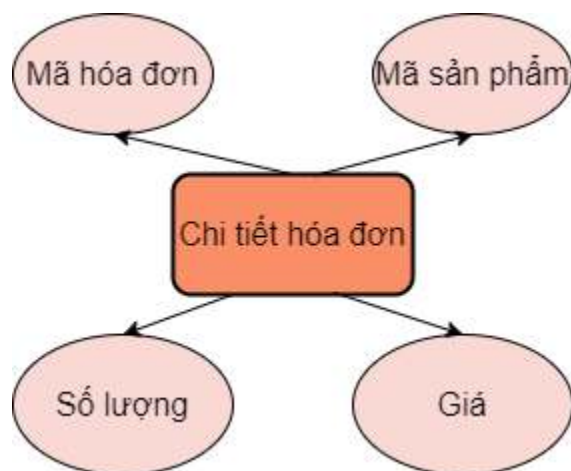
Hình 9: Sơ đồ Hóa đơn.

+ *Thực thể Sản phẩm:*



Hình 10: Sơ đồ Sản phẩm

+ *Thực thể Chi tiết sản phẩm:*



Hình 11: Sơ đồ Chi tiết hóa đơn.

2. Database Design – Thiết kế cơ sở dữ liệu.

2.1. Bảng Admin

Bảng số liệu 1: Table adShop.

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Primary key	maQT	Int		Mã quản trị
	tkQT	Varchar	50	Tài khoản quản trị
	mkQT	Varchar	50	Mật khẩu quản trị

2.2. Bảng Khách hàng

Bảng số liệu 2: Table kháchHang.

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Primary key	maKH	Int		Mã khách hàng
	tkKH	Varchar	50	Tài khoản khách hàng
	mkKH	Varchar	50	Mật khẩu khách hàng
	Email	Varchar	50	Email khách hàng
	tenKH	Varchar	50	Tên khách hàng
	gioiTinhKH	Tinyint		Giới tính khách hàng
	diaChiKH	Varchar	50	Địa chỉ khách hàng
	sdtKH	Int		Số điện thoại
	statusKH	Tinyint		Trạng thái

2.3. Bảng Nhóm sản phẩm

Bảng số liệu 3: Table nhómSP

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Primary key	maNSP	Int		Mã nhóm sản phẩm
	tenNhom	Varchar	50	Tên nhóm
	statusNSP	Tinyint		Trạng thái

2.4. Bảng Thanh toán

Bảng số liệu 4: Table thanhToan

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Primary key	maTT	Int		Mã thanh toán
	hinhThucTT	Varchar	50	Hình thức thanh toán
	statusTT	Tinyint		Trạng thái

2.5. Bảng Vận chuyển

Bảng số liệu 5: Table vanChuyen

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Primary key	maVC	Int		Mã vận chuyển
	hinhThucVC	Varchar	50	Hình thức vận chuyển
	statusVC	Tinyint		Trạng thái

2.6. Bảng Hãng sản xuất

Bảng số liệu 6: Table hangSX

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Primary key	maSX	Int		Mã sản xuất
	tenHangSX	Varchar	50	Tên hãng sản xuất
	statusHangSX	Tinyint		Trạng thái

2.7 Bảng *FeedBack*

Bảng số liệu 7: Table feedBack

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Primary key	maFB	Int		Mã phản hồi
Foreign key	maKH	Int		Mã khách hàng (Bảng kháchHang)
	dateFB	Date		Ngày phản hồi
	tieuDe	Varchar	50	Tiêu đề
	noiDung	Varchar	50	Nội dung
	traloiFB	Varchar	50	Trả lời phản hồi
	statusFB	Tinyint		Trạng thái

2.8 Bảng Hóa đơn

Bảng số liệu 8: Table hoaDon

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Primary key	maHD	Int		Mã hóa đơn
Foreign key	maKH	Int		Mã khách hàng (Bảng kháchHang)
Foreign key	maTT	Int		Mã thanh toán (Bảng thanhToan)
Foreign key	maVC	Int		Mã vận chuyển (Bảng vanChuyen)
	ngayMua	Date		Ngày mua
	ngayGiao	Date		Ngày giao
	tenNN	Varchar	50	Tên người nhận
	gioiTinhNN	Tinyint		Giới tính người nhận
	emailNN	Varchar	50	Email người nhận
	sdtNN	Int		Số điện thoại
	diachiNN	Varchar	50	Địa chỉ người nhận
	ghiChuNN	Varchar	50	Ghi chú người nhận
	statusHD	Tinyint		Trạng thái

2.9 Bảng Sản phẩm

Bảng số liệu 9: Table sanPham

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Primary key	maSP	Int		Mã sản phẩm
Foreign key	maNSP	Int		Mã nhóm sản phẩm (Bảng nhómSP)
Foreign key	maSX	Int		Mã sản xuất (Bảng hangSX)
	ngayNhapSP	Date		Ngày nhập sản phẩm
	tenSP	Varchar	50	Tên sản phẩm
	hinhAnhSP	Varchar	50	Hình ảnh sản phẩm
	giaSP	Int		Giá sản phẩm
	soLuongSP	Int		Số lượng sản phẩm
	moTaSP	Varchar	50	Mô tả sản phẩm
	statusSP	Tinyint		Trạng thái

2.10 Bảng Chi tiết hóa đơn

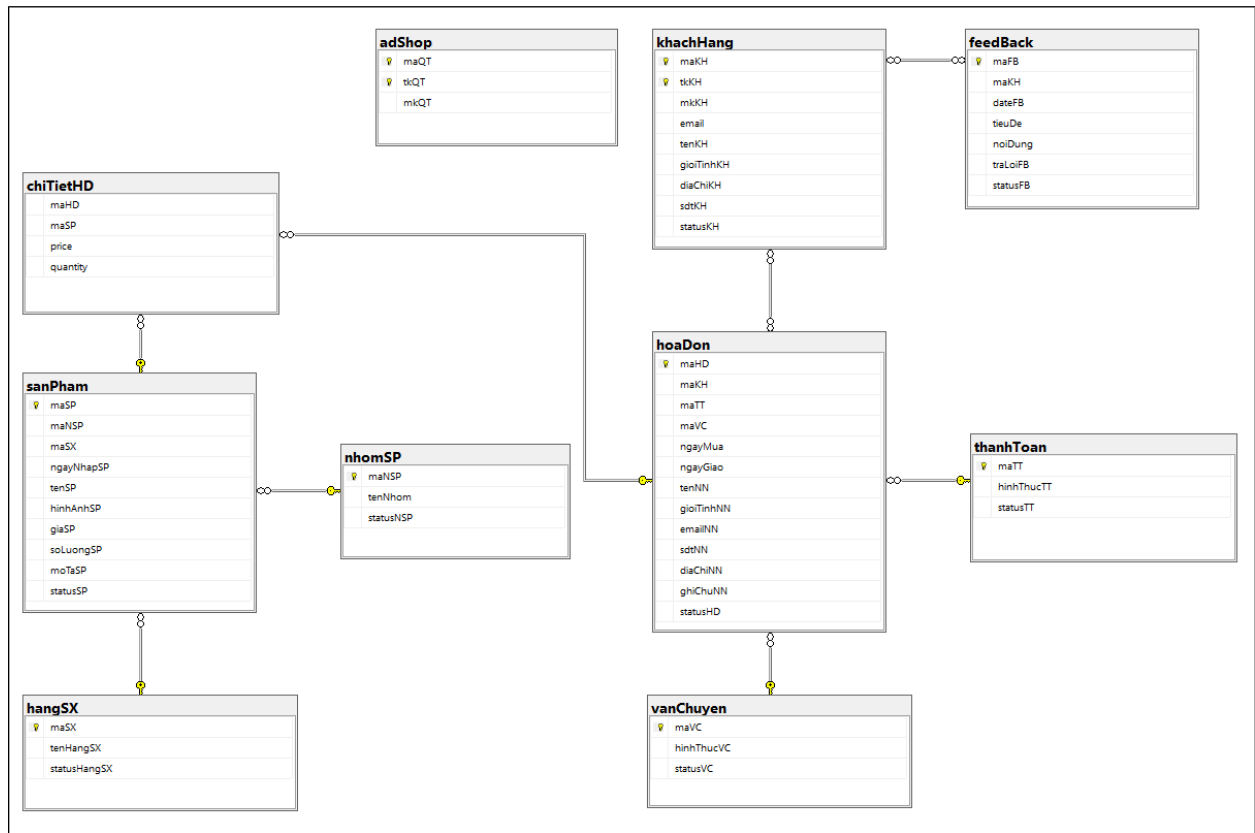
Bảng số liệu 10: Table chiTietHD

Tính chất	Tên trường	Kiểu dữ liệu	Kích cỡ	Chú thích
Foreign key	maHD	Int		Mã hóa đơn (Bảng hoaDon)
Foreign key	maSP	Int		Mã sản phẩm (Bảng sanPham)
	Price	Int		Giá
	quantity	int		Số lượng

+ Mối quan hệ giữa các bảng:

- Ta có thể thấy một *khách hàng* có thể có 1 hoặc nhiều *phản hồi* hoặc *hóa đơn*, một lần *thanh toán* hoặc *vận chuyển* có thể 1 hoặc nhiều *hóa đơn*.
- Hãng *sản xuất* và *nhóm sản phẩm* cũng tương tự. Một *hãng sản xuất* hoặc 1 *nhóm sản phẩm* cũng có thể có 1 hoặc nhiều *sản phẩm*.
- Trong đó 1 *hóa đơn* hoặc 1 *sản phẩm* cũng có thể có một hoặc nhiều *chi tiết hóa đơn*.

➔ Ta có một sơ đồ mối quan hệ giữa các bảng như sau:



Hình 12: Tables Relationship.

- Các trường khóa chính – Primary

```
1  -- =====PRIMARY KEY=====
2  -- adShop - quản trị của shop
3  alter table adShop
4  add constraint pk_adShop primary key (maQT, tkQT);
5  -- kháchHang - khách hàng
6  alter table kháchHang
7  add constraint pk_kháchHang primary key (maKH, tkKH);
8  -- nhómSP - nhóm sản phẩm
9  alter table nhómSP
10 add constraint pk_nhómSP primary key (maNSP);
11 -- hangSX - hãng sản xuất
12 alter table hangSX
13 add constraint pk_hangSX primary key (maSX);
14 -- thanhToán - thanh toán
15 alter table thanhToán
16 add constraint pk_thanhToán primary key (maTT);
17 -- vanChuyen - vận chuyển
18 alter table vanChuyen
19 add constraint pk_vanChuyen primary key (maVC);
20 -- feedBack - phản hồi
21 alter table feedBack
22 add constraint pk_feedBack primary key (maFB);
23 -- hoaDon - hóa đơn
24 alter table hoaDon
25 add constraint pk_hoaDon primary key (maHD);
26 -- sanPham - sản phẩm
27 alter table sanPham
28 add constraint pk_sanPham primary key (maSP);
```

Hình 13: Primary Keys.

- Các trường khóa phụ - Foreign key

```
1  -- =====FOREIGN KEY=====
2  -- feedBack - phản hồi
3  alter table feedBack
4  add constraint fk_feedBack foreign key (maKH)
5  references kháchHang(maKH);
6  -- hoaDon - hóa đơn
7  alter table hoaDon
8  add constraint fk_hoaDon_maKH foreign key (maKH)
9  references kháchHang(maKH);
10 alter table hoaDon
11 add constraint fk_hoaDon_maTT foreign key (maTT)
12 references thanhToan(maTT);
13 alter table hoaDon
14 add constraint fk_hoaDon_maVC foreign key (maVC)
15 references vanChuyen(maVC);
16 -- sanPham - sản phẩm
17 alter table sanPham
18 add constraint fk_sanPham_maNSP foreign key (maNSP)
19 references nhómSP(maNSP);
20 alter table sanPham
21 add constraint fk_sanPham_maSX foreign key (maSX)
22 references hangSX(maSX);
23 -- chiTietHD - chi tiết hóa đơn
24 alter table chiTietHD
25 add constraint fk_chiTietHD_maHD foreign key (maHD)
26 references hoaDon(maHD);
27 alter table chiTietHD
28 add constraint fk_chiTietHD_maSP foreign key (maSP)
29 references sanPham(maSP);
```

Hình 14: Foreign Keys.

3. Query – các câu truy vấn.

3.1. Nonclustered Index – Chỉ mục.

- Để cải thiện tốc độ truy xuất dữ liệu từ bảng *kháchHang*. Ta dùng non-clustered để sắp xếp và lưu trữ dữ liệu riêng biệt các bản ghi của *kháchHang*. Nó là bản sao dữ liệu của các cột được chọn từ một bảng liên kết.

- Ta sẽ dùng Index (chỉ mục) cho cột *tenKH* của bảng *khachHang*. Code:

```
-- Nonclustered index  
create nonclustered index clus_khachHang  
on khachHang(tenKH);
```

- Vì mỗi khách hàng chỉ được sử dụng duy nhất 1 *email* và *số điện thoại* nên ta sẽ tạo chỉ mục độc nhất (Unique Nonclustered) cho cột *email*, *sdtKH* của bảng *khachHang*. Code:

```
-- Unique NonClustered - độc nhất | theo cột  
create unique index uni_email  
on khachHang(email);  
  
create unique index uni_sdtKH  
on khachHang(sdtKH);
```

3.2. View – Khung hình.

- Các trạng thái (status), giới tính (gender), ... những thuộc tính kiểu Tinyint của từng bảng sẽ được hiển thị view để có cái nhìn cụ thể các bản ghi hơn. Ví dụ: 1:Nam – 0:Nữ hoặc 1:Hiện – 0:Ẩn. Code:

```
create view vwDetails_khachHang as  
select maKH,  
tkKH, mkKH, email,  
tenKH, diaChiKH, sdtKH,  
gioiTinhKH = CASE gioiTinhKH  
    when 1 then 'Nam'  
    when 0 then 'Nu'  
    else 'unknow'  
END,  
statusKH = CASE statusKH  
    when 1 then 'Show'  
    when 0 then 'Hide'  
    else 'unknow'  
END  
from khachHang;
```

3.3. Stored Procedure (SP) – Thủ tục lưu trữ.

- Ta tạo các thủ tục lưu trữ để có thể dễ dàng tìm, quản lý các quản trị, khách hàng.

▪ Tìm quản trị theo tên.

• Code:

```
CREATE PROCEDURE sp_adShop(@tai_khoan VARCHAR(50)) AS
BEGIN
    IF(EXISTS(SELECT * FROM adShop WHERE tkQT like
'%'+@tai_khoan+'%'))
        BEGIN
            SELECT * FROM adShop
            WHERE tkQT like '%'+@tai_khoan*'% '
        END
    ELSE IF(@tai_khoan = '*')
        BEGIN
            SELECT * FROM adShop
        END
    ELSE
        PRINT 'NO INFORMATION!!!'
END;
```

▪ Tìm khách hàng theo tên.

○ Code:

```
BEGIN
    IF ( EXISTS(select * from kháchHang WHERE tenKH like '%' +
@Name + '%') )
        -- Tìm thấy tên khách hàng
        BEGIN
            SELECT * FROM kháchHang a
            JOIN hoaDon b on a.maKH = b.maKH
            JOIN thanhToan c on b.maTT = c.maTT
            WHERE tenKH like '%' + @Name + '%'
        END
    ELSE IF ( @Name = '*')
        -- Hiển thị tất cả khách hàng
        BEGIN
            SELECT * FROM kháchHang a
            JOIN hoaDon b on a.maKH = b.maKH
            JOIN thanhToan c on b.maTT = c.maTT
        END
    ELSE
        PRINT 'Không tìm thấy thông tin liên quan đến ' + @Name
```

```
END;
```

3.4. Trigger – Thủ tục kích hoạt tự động.

▪ Insert – Chèn:

- Khi thêm bản ghi cho *Quản trị* (adShop) thì ta sẽ kiểm tra xem thuộc tính *mã quản trị* (maQT) có bé hơn 1 không. Nếu bé hơn thì sẽ không cho thêm bản ghi đó.
- Code:

```
create trigger tg_insert_adShop
on adShop for INSERT AS
BEGIN
    IF((select maQT from inserted) < 1)
    BEGIN
        print 'Ma quan tri phai >= 1'
        ROLLBACK TRANSACTION
    END
END;
```

▪ Update – Cập nhật:

- Kiểm tra xem bản ghi của *sanPham* khi được cập nhật thì *maNSP* của *sanPham* có tồn tại không. Nếu không tồn tại thì không cho cập nhật.
- Code:

```
create trigger tg_update_sanPham
on sanPham for UPDATE AS
BEGIN
    DECLARE @maNSP int;
    SET @maNSP = (select maNSP from inserted)
    IF(@maNSP < 1 OR @maNSP > 6)
    BEGIN
        print 'Ma nhom san pham khong hop le!'
        ROLLBACK TRANSACTION
    END
END;
```

▪ Delete – Xóa:

- Vì bản ghi quản lý *quản trị* rất quan trọng nên để tránh xóa nhầm ta sẽ tạo thủ tục tự động để kiểm tra xem khi xóa bản ghi có xóa quá 1 quản trị không. Nếu xóa quá 1 bản ghi trong bảng thì sẽ không cho xóa.
- Code:

```
create trigger tg_delete_adShop
on adShop for delete AS
BEGIN
    IF((select count(*) from deleted) > 1)
    BEGIN
        PRINT 'Ban chi co the xoa 1 ban ghi trong bang'
        ROLLBACK TRANSACTION
    END
END;
```

▪ After:

- Thông báo xem đã xóa bao nhiêu bản ghi trong bảng. Code:

```
create trigger tg_afterDelete_khachHang
on kháchHang after DELETE AS
BEGIN
    DECLARE @count NCHAR;
    SELECT @count = count(*) from deleted;
    PRINT 'Da xoa ' + @count + ' khách hang khoi bang'
END;
```

▪ Instead of:

- Khi cập nhật *mã nhóm sản phẩm* trong bảng thì bảng liên kết đến bảng đó cũng phải cập nhật lại những bản ghi có chứa *mã nhóm sản phẩm* vừa cập nhật xong.
- Code:

```
create trigger update_maNSP
on nhómSP INSTEAD OF UPDATE AS
BEGIN
    ALTER TABLE sanPham DROP CONSTRAINT fk_sanPham_maNSP;
    UPDATE nhómSP set maNSP = (select maNSP from inserted)
    WHERE maNSP = (select maNSP from deleted)
    UPDATE sanPham set maNSP = (select maNSP from inserted)
```

```
WHERE maNSP = (select maNSP from deleted)
alter table sanPham
add constraint fk_sanPham_maNSP foreign key (maNSP)
references nhomSP(maNSP);
END;
```

Chương III. Tổng kết.

1. Kết quả đạt được:

- Hiểu biết rõ hơn cách thể hiện các thực thể có trong database, và mối quan hệ giữa chúng với nhau.
- Biết cách mô hình hóa cơ sở dữ liệu.

2. Hướng phát triển:

- Nâng cao tư duy trong việc kiểm soát các thực thể cũng như mối quan hệ giữa các thực thể với nhau.
- Chặt chẽ hơn trong việc kiểm tra giá trị khi các thực thể bị thay đổi hoặc mối quan hệ.

Chương IV. Tài liệu tham khảo.

- Link tham khảo nội dung báo cáo:

- ✓ <https://khotrithucso.com/doc/p/tieu-luan-he-co-so-du-lieu-329965>
- ✓ <https://v1study.com/php-documentation-mau-cho-thiet-ke-website.html#system-design>
- ✓ <https://v1study.com/sql-trigger-instead-of.html>