**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: HỆ QUẢN TRỊ CSDL**

**ĐỀ TÀI: QUẢN LÝ BÁN HÀNG TẠI SIÊU THỊ**

**Giáo viên hướng dẫn: Ths. Nguyễn Ngọc Ân**

**Sinh viên thực hiện:**  **Nhóm 10**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Mã sv** | **Họ và tên** | **Lớp** |
| **1** | **1771020255** | **Trịnh Công Hiệp** | **CNTT 1709** |
| **2** | **1771020287** | **Phạm Thế Hoán** | **CNTT 1709** |

**Hà Nội, năm 2025**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: HỆ QUẢN TRỊ CSDL**

**ĐỀ TÀI: QUẢN LÝ BÁN HÀNG TẠI SIÊU THỊ**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mã Sinh Viên | Họ và Tên | Ngày Sinh | Điểm | |
| Bằng Số | Bằng Chữ |
| 1 | 1771020255 | Trịnh Công Hiệp | 10/05/2005 |  |  |
| 2 | 1771020287 | Phạm Thế Hoán | 10/04/2005 |  |  |

### 

### CÁN BỘ CHẤM THI

**Hà Nội, năm 2025**

**LỜI NÓI ĐẦU**

(Nếu có)

**MỤC LỤC**

[**LỜI NÓI ĐẦU** 3](#_Toc193219463)

[**MỤC LỤC** 4](#_Toc193219464)

[**CHƯƠNG 1. MỞ ĐẦU** 5](#_Toc193219465)

[**1.1 Giới thiệu tổng quan về đề tài** 5](#_Toc193219466)

[**1.2 Mục đích và lý do lựa chọn đề tài** 5](#_Toc193219467)

[**1.3 Phạm vi nghiên cứu** 5](#_Toc193219468)

[**1.4 Phương pháp nghiên cứu** 6](#_Toc193219469)

[**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ CSDL** 7](#_Toc193219470)

[**2.1 Xác định các yêu cầu của hệ thống** 7](#_Toc193219471)

[**2.2 Thiết kế mô hình dữ liệu (ERD)** 7](#_Toc193219474)

[**2.3 Chuyển đổi mô hình dữ liệu sang mô hình vật lý** 8](#_Toc193219475)

[**CHƯƠNG 3: TRIỂN KHAI CƠ SỞ DỮ LIỆU** 11](#_Toc193219485)

[**3.1 Tạo Cơ Sở Dữ Liệu và Các Bảng** 11](#_Toc193219486)

[**3.2 Nhập Dữ Liệu Mẫu** 11](#_Toc193219487)

[**3.3 Truy Vấn Cơ Bản** 12](#_Toc193219488)

[**Truy Vấn Nâng Cao** 12](#_Toc193219489)

[**CHƯỜNG 4: LẬP TRÌNH T-SQL** 14](#_Toc193219490)

[**4.1 Tạo View** 14](#_Toc193219491)

[**4.2 Tạo Index** 15](#_Toc193219492)

[**4.3 Xây Dựng Stored Procedure** 16](#_Toc193219493)

[**4.4 Tạo Function** 18](#_Toc193219494)

[**4.5 Tạo Trigger** 21](#_Toc193219495)

[**4.6 Kết Luận** 23](#_Toc193219496)

[**CHƯƠNG 5: BẢO MẬT VÀ QUẢN TRỊ** 24](#_Toc193219497)

[**5.1. Tạo người dùng** 24](#_Toc193219498)

[**5.2. Thiết lập quyền truy cập và phân quyền người dùng** 24](#_Toc193219501)

[**5.3. Quản lý sao lưu và phục hồi dữ liệu** 25](#_Toc193219503)

[**CHƯƠNG 6: KẾT LUẬN** 26](#_Toc193219504)

[**6.1. Nhận xét về quá trình thực hiện** 26](#_Toc193219505)

[**6.2. Đánh giá ưu và nhược điểm của hệ thống** 26](#_Toc193219506)

[**6.3. Đề xuất hướng phát triển tiếp theo** 27](#_Toc193219507)

[**PHÂN CÔNG NHIỆM VỤ** 28](#_Toc193219508)

[**DANH MỤC TÀI LIỆU THAM KHẢO** 29](#_Toc193219509)

**CHƯƠNG 1. MỞ ĐẦU**

## **1.1 Giới thiệu tổng quan về đề tài**

Siêu thị là một mô hình kinh doanh bán lẻ hiện đại, cung cấp đa dạng các mặt hàng từ thực phẩm, thời trang, điện tử đến đồ gia dụng. Với sự phát triển của công nghệ và xu hướng mua sắm hiện nay, việc xây dựng một hệ thống quản lý bán hàng hiệu quả là điều cần thiết nhằm nâng cao trải nghiệm mua sắm, tối ưu quy trình vận hành và quản lý dữ liệu một cách chặt chẽ.

Hệ thống quản lý bán hàng tại siêu thị không chỉ hỗ trợ nhân viên bán hàng và kế toán trong việc ghi nhận giao dịch, mà còn giúp ban quản lý nắm bắt tình hình kinh doanh, tồn kho, chương trình khuyến mãi và quản lý khách hàng thân thiết. Với sự trợ giúp của công nghệ thông tin, việc số hóa quy trình bán hàng giúp giảm thiểu sai sót, nâng cao hiệu suất làm việc và cải thiện chất lượng dịch vụ.

## **1.2 Mục đích và lý do lựa chọn đề tài**

Hệ thống quản lý bán hàng tại siêu thị được xây dựng nhằm đạt được các mục tiêu chính sau:

* **Tối ưu hóa quy trình bán hàng**: Hỗ trợ nhân viên xử lý nhanh chóng các giao dịch mua bán, quản lý hóa đơn, và cập nhật dữ liệu tồn kho theo thời gian thực.
* **Quản lý khách hàng hiệu quả**: Theo dõi thông tin khách hàng, đặc biệt là khách hàng VIP để áp dụng các chương trình ưu đãi phù hợp.
* **Hỗ trợ kiểm soát tồn kho**: Cập nhật và quản lý dữ liệu nhập, xuất hàng hóa giúp siêu thị nắm bắt chính xác lượng hàng tồn để đặt hàng kịp thời.
* **Nâng cao khả năng phân tích và dự báo**: Hệ thống có thể giúp ban quản lý theo dõi xu hướng mua sắm của khách hàng, từ đó đưa ra các quyết định kinh doanh hợp lý.
* **Tăng cường bảo mật và quản lý dữ liệu**: Hệ thống đảm bảo dữ liệu được bảo vệ, phân quyền hợp lý để tránh truy cập trái phép.

Lý do lựa chọn đề tài xuất phát từ nhu cầu thực tế của các siêu thị hiện nay. Nhiều doanh nghiệp vẫn sử dụng các phương pháp thủ công hoặc hệ thống cũ chưa đáp ứng được nhu cầu quản lý hiện đại. Việc xây dựng một hệ thống quản lý bán hàng tối ưu sẽ giúp cải thiện chất lượng dịch vụ, giảm chi phí vận hành và gia tăng sự hài lòng của khách hàng.

## **1.3 Phạm vi nghiên cứu**

Hệ thống quản lý bán hàng tại siêu thị sẽ tập trung vào các chức năng chính sau:

* **Quản lý sản phẩm**: Bao gồm thông tin chi tiết về sản phẩm như tên, mã hàng, giá bán, số lượng tồn kho, danh mục hàng hóa.
* **Quản lý khách hàng**: Lưu trữ thông tin khách hàng thường xuyên, khách hàng VIP và theo dõi lịch sử mua sắm.
* **Quản lý hóa đơn bán hàng**: Ghi nhận giao dịch mua bán, in hóa đơn, tính toán khuyến mãi cho khách hàng VIP.
* **Quản lý tồn kho**: Cập nhật số lượng hàng hóa, theo dõi hàng tồn tối thiểu và đề xuất nhập hàng khi cần thiết.
* **Hậu mãi và đổi trả hàng**: Quản lý chính sách đổi trả, theo dõi hóa đơn liên quan và xử lý các giao dịch hoàn tiền hoặc đổi hàng.

## **1.4 Phương pháp nghiên cứu**

Trong quá trình thực hiện đề tài, nhóm đã áp dụng các phương pháp nghiên cứu sau:

* **Phương pháp thu thập tài liệu**: Tham khảo các tài liệu chuyên ngành về quản lý bán hàng, hệ thống cơ sở dữ liệu và quy trình vận hành siêu thị.
* **Phương pháp phân tích hệ thống**: Tiến hành khảo sát thực tế, tìm hiểu các yêu cầu từ siêu thị và đề xuất mô hình phù hợp.
* **Phương pháp thiết kế và lập trình**: Sử dụng mô hình ERD để thiết kế cơ sở dữ liệu, xây dựng bảng, khóa, ràng buộc dữ liệu và lập trình các truy vấn SQL phục vụ quản lý dữ liệu.
* **Phương pháp kiểm thử**: Tiến hành nhập dữ liệu mẫu, chạy thử hệ thống và kiểm tra kết quả để đảm bảo tính chính xác và hiệu quả của hệ thống.

# **CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ CSDL**

## **2.1 Xác định các yêu cầu của hệ thống**

a. Yêu cầu chức năng

* Quản lý khách hàng: Lưu trữ thông tin khách hàng bao gồm mã khách hàng, họ tên, số điện thoại, email, địa chỉ và loại khách hàng (thường/VIP).
* Quản lý thẻ VIP: Theo dõi thông tin thẻ VIP của khách hàng như mã thẻ, số lần sử dụng, ngày cấp thẻ và mức chiết khấu.
* Quản lý sản phẩm: Lưu trữ thông tin sản phẩm bao gồm mã sản phẩm, tên sản phẩm, giá, loại sản phẩm và số lượng tồn kho.
* Quản lý hóa đơn: Theo dõi các hóa đơn mua hàng với thông tin khách hàng, ngày mua và tổng tiền.
* Quản lý chi tiết hóa đơn: Ghi nhận thông tin các sản phẩm trong từng hóa đơn bao gồm mã sản phẩm, số lượng và giá bán.
* Quản lý phiếu chi: Lập phiếu chi khi khách hàng trả hàng, ghi nhận lý do và số tiền hoàn trả.
* Quản lý phiếu đổi hàng: Theo dõi các giao dịch đổi hàng, bao gồm sản phẩm trả, sản phẩm nhận và số tiền chênh lệch.
* Quản lý nhân viên: Theo dõi thông tin nhân viên, bao gồm mã nhân viên, họ tên, chức vụ và số điện thoại.
* Quản lý kho hàng: Theo dõi số lượng tồn kho và ngưỡng tối thiểu cần duy trì của từng sản phẩm.

b. Yêu cầu phi chức năng

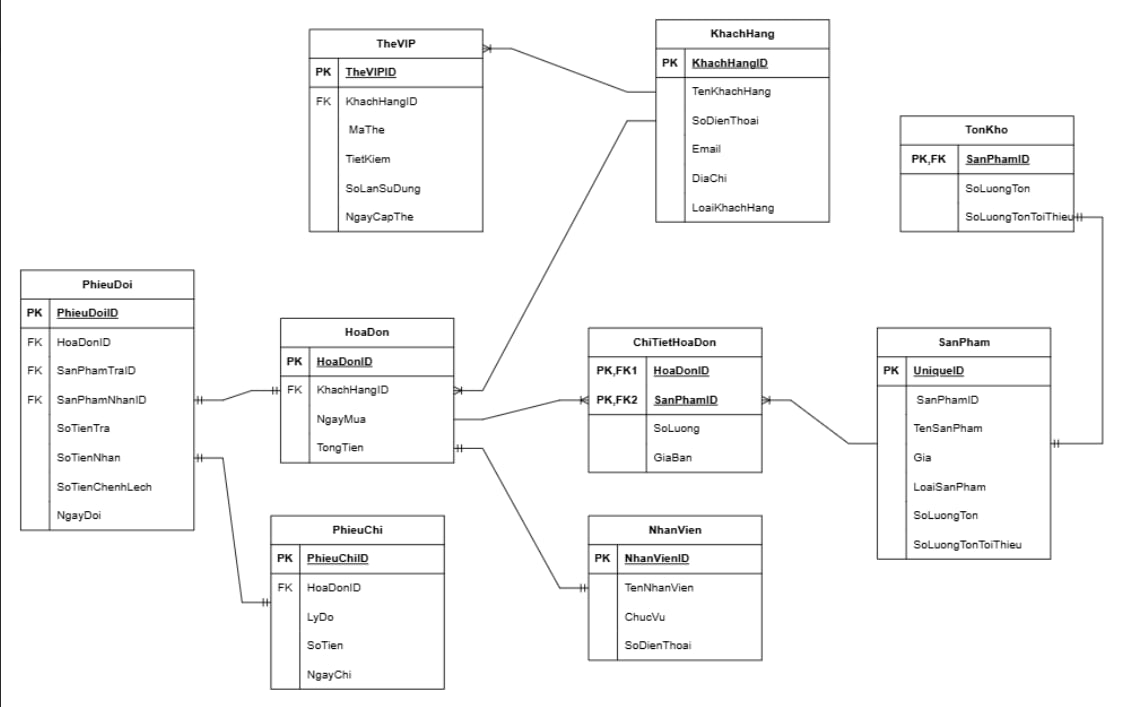
* Hệ thống phải đảm bảo tính chính xác và bảo mật dữ liệu.
* Giao diện trực quan, dễ sử dụng cho nhân viên bán hàng.
* Hỗ trợ báo cáo doanh thu, thống kê hàng tồn kho và hoạt động giao dịch.
* Đáp ứng tốc độ xử lý nhanh, tối ưu hiệu suất khi truy vấn dữ liệu.

## **2.2 Thiết kế mô hình dữ liệu (ERD)**

Dựa trên các yêu cầu đã xác định, mô hình thực thể quan hệ (ERD) của hệ thống được thiết kế như sau:

* Thực thể KhachHang: Chứa thông tin khách hàng.
* Thực thể TheVIP: Liên kết với KhachHang, lưu thông tin thẻ VIP.
* Thực thể SanPham: Lưu trữ thông tin sản phẩm.
* Thực thể TonKho: Theo dõi số lượng tồn kho của sản phẩm.
* Thực thể HoaDon: Ghi nhận thông tin hóa đơn của khách hàng.
* Thực thể ChiTietHoaDon: Liên kết giữa hóa đơn và sản phẩm, chi tiết số lượng mua.
* Thực thể PhieuChi: Lưu thông tin phiếu chi khi khách hàng trả hàng.
* Thực thể PhieuDoi: Ghi nhận thông tin đổi hàng.
* Thực thể NhanVien: Lưu trữ thông tin nhân viên.

(ERD sẽ được minh họa bằng sơ đồ, bạn có thể đính kèm hình ảnh ERD vào tài liệu)



## **2.3 Chuyển đổi mô hình dữ liệu sang mô hình vật lý**

Dưới đây là các bảng cơ sở dữ liệu cùng khóa chính (PK), khóa ngoại (FK) và các ràng buộc:

Bảng KhachHang

| Tên cột | Kiểu dữ liệu | Ràng buộc |
| --- | --- | --- |
| KhachHangID | INT | PK |
| TenKhachHang | VARCHAR(100) | NOT NULL |
| SoDienThoai | VARCHAR(15) | UNIQUE |
| Email | VARCHAR(100) | UNIQUE |
| DiaChi | TEXT | NULL |
| LoaiKhachHang | VARCHAR(10) | CHECK ('Thường' OR 'VIP') |

Bảng TheVIP

| Tên cột | Kiểu dữ liệu | Ràng buộc |
| --- | --- | --- |
| TheVIPID | INT | PK |
| KhachHangID | INT | FK -> KhachHang(KhachHangID) |
| MaThe | VARCHAR(50) | UNIQUE |
| TietKiem | FLOAT | DEFAULT 0 |
| SoLanSuDung | INT | DEFAULT 0 |
| NgayCapThe | DATE | NOT NULL |

Bảng SanPham

| Tên cột | Kiểu dữ liệu | Ràng buộc |
| --- | --- | --- |
| SanPhamID | INT | PK |
| TenSanPham | VARCHAR(100) | NOT NULL |
| Gia | FLOAT | NOT NULL |
| LoaiSanPham | VARCHAR(50) | NOT NULL |
| SoLuongTon | INT | DEFAULT 0 |
| SoLuongTonToiThieu | INT | DEFAULT 0 |

Bảng TonKho

| Tên cột | Kiểu dữ liệu | Ràng buộc |
| --- | --- | --- |
| SanPhamID | INT | PK, FK -> SanPham(SanPhamID) |
| SoLuongTon | INT | DEFAULT 0 |
| SoLuongTonToiThieu | INT | DEFAULT 0 |

Bảng HoaDon

| Tên cột | Kiểu dữ liệu | Ràng buộc |
| --- | --- | --- |
| HoaDonID | INT | PK |
| KhachHangID | INT | FK -> KhachHang(KhachHangID) |
| NgayMua | DATE | NOT NULL |
| TongTien | FLOAT | NOT NULL |

Bảng ChiTietHoaDon

| Tên cột | Kiểu dữ liệu | Ràng buộc |
| --- | --- | --- |
| HoaDonID | INT | PK, FK -> HoaDon(HoaDonID) |
| SanPhamID | INT | PK, FK -> SanPham(SanPhamID) |
| SoLuong | INT | NOT NULL |
| GiaBan | FLOAT | NOT NULL |

Bảng PhieuChi

| Tên cột | Kiểu dữ liệu | Ràng buộc |
| --- | --- | --- |
| PhieuChiID | INT | PK |
| HoaDonID | INT | FK -> HoaDon(HoaDonID) |
| LyDo | TEXT | NOT NULL |
| SoTien | FLOAT | NOT NULL |
| NgayChi | DATE | NOT NULL |

Bảng PhieuDoi

| Tên cột | Kiểu dữ liệu | Ràng buộc |
| --- | --- | --- |
| PhieuDoiID | INT | PK |
| HoaDonID | INT | FK -> HoaDon(HoaDonID) |
| SanPhamTraID | INT | FK -> SanPham(SanPhamID) |
| SanPhamNhanID | INT | FK -> SanPham(SanPhamID) |
| SoTienTra | FLOAT | NOT NULL |
| SoTienNhan | FLOAT | NOT NULL |
| SoTienChenhLech | FLOAT | NOT NULL |
| NgayDoi | DATE | NOT NULL |

Bảng NhanVien

| Tên cột | Kiểu dữ liệu | Ràng buộc |
| --- | --- | --- |
| NhanVienID | INT | PK |
| TenNhanVien | VARCHAR(100) | NOT NULL |
| ChucVu | VARCHAR(50) | NOT NULL |
| SoDienThoai | VARCHAR(15) | UNIQUE |

# **CHƯƠNG 3: TRIỂN KHAI CƠ SỞ DỮ LIỆU**

## **3.1 Tạo Cơ Sở Dữ Liệu và Các Bảng**

Cơ sở dữ liệu quản lý bán hàng tại siêu thị được triển khai với các bảng chính như sau:

* **KhachHang** (Mã khách hàng, Tên khách hàng, Số điện thoại, Email, Địa chỉ, Loại khách hàng)
* **TheVIP** (Mã thẻ VIP, Mã khách hàng, Mã thẻ, Tiết kiệm, Số lần sử dụng, Ngày cấp thẻ)
* **HoaDon** (Mã hóa đơn, Mã khách hàng, Ngày mua, Tổng tiền)
* **ChiTietHoaDon** (Mã hóa đơn, Mã sản phẩm, Số lượng, Giá bán)
* **SanPham** (Mã sản phẩm, Tên sản phẩm, Giá, Loại sản phẩm, Số lượng tồn, Số lượng tồn tối thiểu)
* **TonKho** (Mã sản phẩm, Số lượng tồn, Số lượng tồn tối thiểu)
* **NhanVien** (Mã nhân viên, Tên nhân viên, Chức vụ, Số điện thoại)
* **PhieuDoi** (Mã phiếu đổi, Mã hóa đơn, Sản phẩm trả, Sản phẩm nhận, Số tiền trả, Số tiền nhận, Số tiền chênh lệch, Ngày đổi)
* **PhieuChi** (Mã phiếu chi, Mã hóa đơn, Lý do, Số tiền, Ngày chi)

#### Định nghĩa khóa và ràng buộc:

* **Khóa chính (PK)**: Được xác định trên các cột quan trọng như Mã khách hàng, Mã hóa đơn, Mã sản phẩm.
* **Khóa ngoại (FK)**: Liên kết giữa các bảng để đảm bảo tính toàn vẹn dữ liệu.
* **Ràng buộc**: Các điều kiện kiểm tra dữ liệu như NOT NULL, UNIQUE, CHECK.

## **3.2 Nhập Dữ Liệu Mẫu**

Sau khi tạo bảng, nhập một số dữ liệu mẫu:

Khách hàng

INSERT INTO KhachHang (KhachHangID, TenKhachHang, SoDienThoai, Email, DiaChi, LoaiKhachHang)

VALUES

(1, 'Nguyen Minh Tu', '0912345678', 'nguyenminhtu@example.com', 'Hanoi', N'VIP'),

(2, 'Le Thi Lan', '0987654321', 'lethilan@example.com', 'Ho Chi Minh', N'Thường'),

(3, 'Tran Thanh Hoa', '0978654321', 'tranthanhhoa@example.com', 'Da Nang', N'VIP'),

(4, 'Nguyen Quang Hieu', '0936547890', 'nguyenquanghieu@example.com', 'Hue', N'Thường'),

(5, 'Pham Quyen', '0901234567', 'phamquyen@example.com', 'Can Tho', N'Thường'),

(6, 'Hoang Minh Thu', '0977654321', 'hoangminhthu@example.com', 'Hai Phong', N'VIP'),

(7, 'Bui Thanh Son', '0912345679', 'bui.thanhson@example.com', 'Nha Trang', N'Thường');

Sản phẩm

INSERT INTO SanPham (SanPhamID, TenSanPham, Gia, LoaiSanPham, SoLuongTon, SoLuongTonToiThieu)

VALUES

(1, N'Bánh Kẹo', 25.000, N'Tiêu dùng', 200, 50),

(2, N'Sữa Tươi', 30.000, N'Thực phẩm', 100, 30),

(3, N'Áo Thun', 150.000, N'Thời trang', 50, 10),

(4, N'Quần Jean', 200.000, N'Thời trang', 70, 20),

(5, N'Nước Ngọt', 15.000, N'Thực phẩm', 500, 100),

(6, N'Chè Khúc Bạch', 50.000, N'Tiêu dùng', 120, 20),

(7, N'Giày Thể Thao', 300.000, N'Thời trang', 80, 20);

## **3.3 Truy Vấn Cơ Bản**

* **Chọn dữ liệu:**

**SELECT \* FROM KhachHang**

* **Thêm dữ liệu:**

INSERT INTO HoaDon (HoaDonID, KhachHangID, NgayMua, TongTien)

VALUES (8, 1, '2025-03-01', 500.000);

* **Cập nhật dữ liệu:**

UPDATE SanPham SET Gia = 27000 WHERE SanPhamID = 1;

* **Xóa dữ liệu:**

DELETE FROM HoaDon WHERE HoaDonID = 8;

## Truy Vấn Nâng Cao

* **INNER JOIN giữa hóa đơn và khách hàng:**

SELECT HoaDon.HoaDonID, KhachHang.TenKhachHang, HoaDon.NgayMua, HoaDon.TongTien

FROM HoaDon

INNER JOIN KhachHang ON HoaDon.KhachHangID = KhachHang.KhachHangID;

* **Tính tổng tiền của từng khách hàng (GROUP BY):**

SELECT KhachHang.TenKhachHang, SUM(HoaDon.TongTien) AS TongTienMua

FROM HoaDon

INNER JOIN KhachHang ON HoaDon.KhachHangID = KhachHang.KhachHangID

GROUP BY KhachHang.TenKhachHang;

* **Truy vấn con (SUBQUERY) để lấy sản phẩm có giá cao nhất:**

# **CHƯỜNG 4: LẬP TRÌNH T-SQL**

## **4.1 Tạo View**

Các view giúp truy vấn dữ liệu nhanh hơn và dễ dàng hơn. Dưới đây là một số view từ cơ bản đến nâng cao:

* **View danh sách khách hàng:**

CREATE VIEW v\_DanhSachKhachHang AS

SELECT KhachHangID, TenKhachHang, SoDienThoai, Email, DiaChi

FROM KhachHang;

* **View tổng số lượng hàng tồn theo loại sản phẩm:**

CREATE VIEW v\_TongSoLuongTon AS

SELECT LoaiSanPham, SUM(SoLuongTon) AS TongSoLuong

FROM SanPham

GROUP BY LoaiSanPham;

* **View hóa đơn của khách hàng VIP:**

CREATE VIEW v\_HoaDonVIP AS

SELECT hd.HoaDonID, kh.TenKhachHang, hd.NgayMua, hd.TongTien

FROM HoaDon hd

JOIN KhachHang kh ON hd.KhachHangID = kh.KhachHangID

WHERE kh.LoaiKhachHang = 'VIP';

* **View danh sách nhân viên**

CREATE VIEW v\_DanhSachNhanVien AS

SELECT NhanVienID, TenNhanVien, ChucVu FROM NhanVien;

* **View thống kê doanh thu**

CREATE VIEW v\_ThongKeDoanhThu AS

SELECT KhachHangID, SUM(TongTien) AS DoanhThu FROM HoaDon GROUP BY KhachHangID;

* **View sản phẩm bán chạy**

CREATE VIEW v\_SanPhamBanChay AS

SELECT TOP 5 SanPhamID, SUM(SoLuong) AS TongSoLuong FROM ChiTietHoaDon GROUP BY SanPhamID ORDER BY TongSoLuong DESC;

* **View chi tiết hóa đơn**

CREATE VIEW v\_HoaDonChiTiet AS

SELECT hd.HoaDonID, hd.NgayMua, kh.TenKhachHang, ct.SanPhamID, ct.SoLuong, ct.GiaBan

FROM HoaDon hd JOIN ChiTietHoaDon ct ON hd.HoaDonID = ct.HoaDonID

JOIN KhachHang kh ON hd.KhachHangID = kh.KhachHangID;

## **4.2 Tạo Index**

Index giúp tăng tốc độ truy vấn dữ liệu. Một số index quan trọng:

* **Index trên cột KhachHangID để tối ưu tìm kiếm khách hàng:**

CREATE INDEX idx\_KhachHangID ON KhachHang(KhachHangID);

* **Index trên cột SanPhamID để tăng tốc độ truy vấn sản phẩm:**

CREATE INDEX idx\_SanPhamID ON SanPham(SanPhamID);

* **Index trên cột NgayMua trong bảng HoaDon để tối ưu hóa truy vấn theo ngày:**

CREATE INDEX idx\_NgayMua ON HoaDon(NgayMua);

* **Index trên cột KhachHangID trong bảng HoaDon để tối ưu tìm kiếm khách hàng:**

CREATE INDEX idx\_KhachHangID ON HoaDon(KhachHangID);

* **Index trên cột GiaBan trong bảng ChiTietHoaDon để tăng tốc độ truy vấn theo giá bán:**

CREATE INDEX idx\_GiaBan ON ChiTietHoaDon(GiaBan);

* **Index trên cột TongTien trong bảng HoaDon để tối ưu hóa truy vấn theo tổng tiền:**

CREATE INDEX idx\_TongTien ON HoaDon(TongTien);

* **Index trên cột LoaiSanPham trong bảng SanPham để tăng tốc độ truy vấn theo loại sản phẩm:**

CREATE INDEX idx\_LoaiSanPham ON SanPham(LoaiSanPham);

## **4.3 Xây Dựng Stored Procedure**

Stored Procedure giúp thực hiện các thao tác nhanh chóng và có thể tái sử dụng.

* **Stored Procedure đơn giản (không tham số) - Lấy danh sách sản phẩm:**

CREATE PROCEDURE sp\_LayDanhSachSanPham

AS

BEGIN

SELECT \* FROM SanPham;

END;

* **Stored Procedure có tham số - Lấy thông tin khách hàng theo ID:**

CREATE PROCEDURE sp\_LayThongTinKhachHang @KhachHangID INT

AS

BEGIN

SELECT \* FROM KhachHang WHERE KhachHangID = @KhachHangID;

END;

* **Stored Procedure có OUTPUT - Tính tổng tiền hóa đơn của khách hàng:**

CREATE PROCEDURE sp\_TinhTongTienHoaDon

@KhachHangID INT,

@TongTien FLOAT OUTPUT

AS

BEGIN

SELECT @TongTien = SUM(TongTien) FROM HoaDon WHERE KhachHangID = @KhachHangID;

END;

* **Stored Procedure có tham số - Lấy hóa đơn theo ngày:**

CREATE PROCEDURE sp\_LayHoaDonTheoNgay @Ngay DATE

AS

BEGIN

SELECT \* FROM HoaDon WHERE NgayMua = @Ngay;

END;

EXEC sp\_LayHoaDonTheoNgay @Ngay = '2024-03-17';

* **Stored Procedure có tham số - Lấy sản phẩm có số lượng tồn kho thấp:**

CREATE PROCEDURE sp\_LaySanPhamTonKhoThap @SoLuong INT

AS

BEGIN

SELECT \* FROM SanPham WHERE SoLuongTon < @SoLuong;

END;

EXEC sp\_LaySanPhamTonKhoThap @SoLuong = 10;

* **Stored Procedure có tham số - Cập nhật giá sản phẩm:**

CREATE PROCEDURE sp\_CapNhatGiaSanPham @SanPhamID INT, @GiaMoi FLOAT

AS

BEGIN

UPDATE SanPham SET Gia = @GiaMoi WHERE SanPhamID = @SanPhamID;

END;

EXEC sp\_CapNhatGiaSanPham @SanPhamID = 1, @GiaMoi = 500000;

* **Stored Procedure có tham số - Xóa khách hàng:**

CREATE PROCEDURE sp\_XoaKhachHang @KhachHangID INT

AS

BEGIN

DELETE FROM KhachHang WHERE KhachHangID = @KhachHangID;

END;

EXEC sp\_XoaKhachHang @KhachHangID = 5;

* **Stored Procedure đơn giản - Lấy danh sách khách hàng:**

CREATE PROCEDURE sp\_LayDanhSachKhachHang

AS

BEGIN

SELECT \* FROM KhachHang;

END;

* **Stored Procedure có tham số - Cập nhật số lượng tồn kho của sản phẩm:**

CREATE PROCEDURE sp\_CapNhatSoLuongTon @SanPhamID INT, @SoLuong INT

AS

BEGIN

UPDATE SanPham SET SoLuongTon = @SoLuong WHERE SanPhamID = @SanPhamID;

END;

EXEC sp\_CapNhatSoLuongTon @SanPhamID = 2, @SoLuong = 100;

* **Stored Procedure đơn giản - Lấy danh sách hóa đơn:**

CREATE PROCEDURE sp\_LayDanhSachHoaDon

AS

BEGIN

SELECT \* FROM HoaDon;

END;

* **Stored Procedure không có tham số - Tính tổng doanh thu:**

CREATE PROCEDURE sp\_TinhTongDoanhThu

AS

BEGIN

SELECT SUM(TongTien) AS TongDoanhThu FROM HoaDon;

END;

EXEC sp\_TinhTongDoanhThu;

## **4.4 Tạo Function**

Function giúp xử lý và trả về dữ liệu một cách linh hoạt.

* **Function trả về kiểu vô hướng - Lấy tổng số đơn hàng của khách hàng:**

CREATE FUNCTION fn\_TongDonHang (@KhachHangID INT)

RETURNS INT

AS

BEGIN

DECLARE @Tong INT;

SELECT @Tong = COUNT(\*) FROM HoaDon WHERE KhachHangID = @KhachHangID;

RETURN @Tong;

END;

* **Function trả về bảng - Danh sách hóa đơn theo khách hàng:**

CREATE FUNCTION fn\_HoaDonTheoKhachHang (@KhachHangID INT)

RETURNS TABLE

AS

RETURN (

SELECT HoaDonID, NgayMua, TongTien

FROM HoaDon

WHERE KhachHangID = @KhachHangID

);

* **Function trả về tổng số tiền hóa đơn của khách hàng**

CREATE FUNCTION fn\_TongHoaDonKhachHang (@KhachHangID INT)

RETURNS FLOAT

AS

BEGIN

DECLARE @Tong FLOAT;

SELECT @Tong = SUM(TongTien) FROM HoaDon WHERE KhachHangID = @KhachHangID;

RETURN @Tong;

END;

* **Function tính trung bình giá sản phẩm**

CREATE FUNCTION fn\_TinhTrungBinhGiaSanPham ()

RETURNS FLOAT

AS

BEGIN

DECLARE @TrungBinh FLOAT;

SELECT @TrungBinh = AVG(Gia) FROM SanPham;

RETURN @TrungBinh;

END;

* **Function đếm số hóa đơn của khách hàng**

CREATE FUNCTION fn\_DemSoHoaDonKhachHang (@KhachHangID INT)

RETURNS INT

AS

BEGIN

DECLARE @SoLuong INT;

SELECT @SoLuong = COUNT(\*) FROM HoaDon WHERE KhachHangID = @KhachHangID;

RETURN @SoLuong;

END;

* **Function tính tổng số lượng tồn theo loại sản phẩm**

CREATE FUNCTION fn\_SoLuongTonTheoLoai (@LoaiSanPham VARCHAR(50))

RETURNS INT

AS

BEGIN

DECLARE @TongSoLuong INT;

SELECT @TongSoLuong = SUM(SoLuongTon) FROM SanPham WHERE LoaiSanPham = @LoaiSanPham;

RETURN @TongSoLuong;

END;

* **Function lấy tên sản phẩm theo ID**

CREATE FUNCTION fn\_LayTenSanPham (@SanPhamID INT)

RETURNS VARCHAR(100)

AS

BEGIN

DECLARE @TenSanPham VARCHAR(100);

SELECT @TenSanPham = TenSanPham FROM SanPham WHERE SanPhamID = @SanPhamID;

RETURN @TenSanPham;

END;

* **Function tính tổng tiền của một hóa đơn**

CREATE FUNCTION fn\_TinhTongTienHoaDon (@HoaDonID INT)

RETURNS FLOAT

AS

BEGIN

DECLARE @TongTien FLOAT;

SELECT @TongTien = SUM(GiaBan \* SoLuong) FROM ChiTietHoaDon WHERE HoaDonID = @HoaDonID;

RETURN @TongTien;

END;

* **Function tính tổng số lượng sản phẩm tồn kho**

CREATE FUNCTION fn\_TinhTongSoLuongSanPham ()

RETURNS INT

AS

BEGIN

DECLARE @Tong INT;

SELECT @Tong = SUM(SoLuongTon) FROM SanPham;

RETURN @Tong;

END;

* **Function đếm tổng số khách hàng**

CREATE FUNCTION fn\_TongSoKhachHang ()

RETURNS INT

AS

BEGIN

DECLARE @SoKhachHang INT;

SELECT @SoKhachHang = COUNT(\*) FROM KhachHang;

RETURN @SoKhachHang;

END;

## **4.5 Tạo Trigger**

Trigger giúp kiểm soát và bảo vệ dữ liệu tự động khi có sự kiện xảy ra.

* **Trigger kiểm tra khi thêm sản phẩm, không cho phép số lượng tồn nhỏ hơn 0:**

CREATE TRIGGER trg\_KiemTraSoLuongTon

ON SanPham

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS (SELECT 1 FROM inserted WHERE SoLuongTon < 0)

BEGIN

RAISERROR ('Số lượng tồn không thể nhỏ hơn 0', 16, 1);

ROLLBACK TRANSACTION;

END;

END;

* **Trigger tự động cập nhật số lần sử dụng thẻ VIP khi khách hàng thanh toán:**

CREATE TRIGGER trg\_CapNhatSoLanSuDungTheVIP

ON HoaDon

AFTER INSERT

AS

BEGIN

UPDATE TheVIP

SET SoLanSuDung = SoLanSuDung - 1

FROM TheVIP t

JOIN inserted i ON t.KhachHangID = i.KhachHangID

WHERE SoLanSuDung > 0;

END;

* **Trigger kiểm tra khi thêm sản phẩm, không cho phép số lượng tồn nhỏ hơn 0:**

CREATE TRIGGER trg\_KiemTraSoLuongTon

ON SanPham

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS (SELECT 1 FROM inserted WHERE SoLuongTon < 0)

BEGIN

RAISERROR ('Số lượng tồn không thể nhỏ hơn 0', 16, 1);

ROLLBACK TRANSACTION;

END;

END;

* **Trigger tự động cập nhật tồn kho sau khi khách hàng mua hàng:**

CREATE TRIGGER trg\_CapNhatTonKhoSauMua

ON ChiTietHoaDon

AFTER INSERT

AS

BEGIN

UPDATE SanPham

SET SoLuongTon = SoLuongTon - i.SoLuong

FROM SanPham sp

JOIN inserted i ON sp.SanPhamID = i.SanPhamID;

END;

* **Trigger kiểm tra giá bán, không cho phép giá sản phẩm âm:**

CREATE TRIGGER trg\_KiemTraGiaBan

ON SanPham

FOR INSERT, UPDATE

AS

BEGIN

IF EXISTS (SELECT 1 FROM inserted WHERE Gia < 0)

BEGIN

RAISERROR ('Giá sản phẩm không được âm!', 16, 1);

ROLLBACK TRANSACTION;

END;

END;

* **Trigger cập nhật ngày mua khi tạo hóa đơn:**

CREATE TRIGGER trg\_CapNhatNgayMuaHoaDon

ON HoaDon

AFTER INSERT

AS

BEGIN

UPDATE HoaDon

SET NgayMua = GETDATE()

FROM HoaDon h

JOIN inserted i ON h.HoaDonID = i.HoaDonID;

END;

* **Trigger cảnh báo khi sản phẩm hết hàng:**

CREATE TRIGGER trg\_CanhBaoSanPhamHetHang

ON SanPham

AFTER UPDATE

AS

BEGIN

IF EXISTS (SELECT 1 FROM inserted WHERE SoLuongTon = 0)

BEGIN

PRINT N'Cảnh báo: Một sản phẩm đã hết hàng!';

END;

END;

* **Trigger cảnh báo khi số lượng tồn kho quá lớn (>1000):**

CREATE TRIGGER trg\_CanhBaoTonKhoCao

ON SanPham

AFTER UPDATE

AS

BEGIN

IF EXISTS (SELECT 1 FROM inserted WHERE SoLuongTon > 1000)

BEGIN

PRINT N'Cảnh báo: Số lượng tồn kho vượt quá 1000!';

END;

END;

## **4.6 Kết Luận**

Chương này đã triển khai các chức năng quan trọng của T-SQL để hỗ trợ quản lý siêu thị, bao gồm tạo View, Index, Stored Procedure, Function và Trigger, giúp tối ưu hóa hiệu suất và tự động hóa các thao tác dữ liệu.

# **CHƯƠNG 5: BẢO MẬT VÀ QUẢN TRỊ**

## **5.1. Tạo người dùng**

Trong hệ thống quản lý bán hàng tại siêu thị, cần tạo nhiều loại người dùng với các quyền khác nhau:

* **Admin:** Quản trị toàn bộ hệ thống, có quyền cao nhất.
* **Nhân viên bán hàng:** Chỉ được truy cập và thao tác với các bảng liên quan đến bán hàng, hóa đơn, khách hàng.
* **Nhân viên kho:** Được truy cập dữ liệu về hàng tồn kho, lập phiếu nhập hàng nhưng không được sửa dữ liệu bán hàng.
* **Kế toán:** Có quyền xem và tổng hợp dữ liệu doanh thu, báo cáo tài chính nhưng không được sửa dữ liệu bán hàng.

### Tạo tài khoản đăng nhập cho nhân viên

CREATE LOGIN NhanVienBanHang WITH PASSWORD = 'BanHang@123';

CREATE LOGIN QuanLyKho WITH PASSWORD = 'Kho@123';

CREATE LOGIN AdminQL WITH PASSWORD = 'Admin@123';

### Tạo người dùng trong cơ sở dữ liệu

CREATE USER NhanVienBanHang FOR LOGIN NhanVienBanHang;

CREATE USER QuanLyKho FOR LOGIN QuanLyKho;

CREATE USER AdminQL FOR LOGIN AdminQL;

## **5.2. Thiết lập quyền truy cập và phân quyền người dùng**

Quyền hạn của mỗi nhóm người dùng sẽ được giới hạn theo vai trò:

* **Admin:** Có toàn quyền (SELECT, INSERT, UPDATE, DELETE) trên tất cả các bảng.
* **Nhân viên bán hàng:** Chỉ có quyền INSERT, UPDATE, SELECT trên bảng **HoaDon**, **ChiTietHoaDon**, **KhachHang**.
* **Nhân viên kho:** Chỉ có quyền SELECT trên bảng **SanPham** và INSERT trên bảng **PhieuNhap**.
* **Kế toán:** Chỉ có quyền SELECT trên các bảng liên quan đến báo cáo tài chính.

### Phân quyền cho nhân viên bán hàng

Nhân viên bán hàng chỉ có quyền **đọc dữ liệu khách hàng**, **tạo hóa đơn**, **xem sản phẩm**

GRANT SELECT, INSERT ON HoaDon TO NhanVienBanHang;

GRANT SELECT ON SanPham TO NhanVienBanHang;

GRANT SELECT ON KhachHang TO NhanVienBanHang;

**Thu hồi quyền truy cập nếu cần**

REVOKE INSERT ON HoaDon FROM NhanVienBanHang;

## **5.3. Quản lý sao lưu và phục hồi dữ liệu**

Nhằm đảm bảo tính toàn vẹn của dữ liệu, hệ thống cần thực hiện sao lưu định kỳ theo các mức độ sau:

* **Sao lưu toàn bộ (Full Backup):** Thực hiện mỗi tuần.
* **Sao lưu vi sai (Differential Backup):** Thực hiện hàng ngày để lưu trữ các thay đổi so với lần sao lưu toàn bộ trước đó.
* **Sao lưu giao dịch (Transaction Log Backup):** Thực hiện mỗi giờ để giảm thiểu mất dữ liệu trong trường hợp lỗi hệ thống.

# **CHƯƠNG 6: KẾT LUẬN**

## **6.1. Nhận xét về quá trình thực hiện**

Trong quá trình thực hiện đề tài **"Quản lý bán hàng tại siêu thị"**, nhóm đã phân tích và thiết kế hệ thống dữ liệu từ khâu mô hình hóa, triển khai cơ sở dữ liệu, xây dựng truy vấn, cho đến thiết lập các cơ chế bảo mật và quản trị.

Các bước thực hiện bao gồm:

* **Xác định yêu cầu hệ thống:** Đã thu thập và phân tích dữ liệu để xác định các bảng và mối quan hệ.
* **Thiết kế mô hình ERD và chuyển đổi sang mô hình vật lý:** Đảm bảo sự chặt chẽ của dữ liệu, tránh dư thừa.
* **Triển khai cơ sở dữ liệu:** Tạo bảng, nhập dữ liệu mẫu, xây dựng các truy vấn SQL từ cơ bản đến nâng cao.
* **Lập trình T-SQL:** Viết stored procedure, function, trigger để tự động hóa các thao tác.
* **Thiết lập bảo mật và quản trị:** Phân quyền người dùng, sao lưu dữ liệu đảm bảo an toàn hệ thống.

## **6.2. Đánh giá ưu và nhược điểm của hệ thống**

**Ưu điểm**

Hệ thống giúp **tự động hóa quy trình bán hàng**, giảm sai sót do nhập liệu thủ công.

CSDL được thiết kế **chuẩn hóa, tối ưu hóa**, tránh trùng lặp dữ liệu.

Các chức năng bảo mật và phân quyền **bảo vệ dữ liệu**, tránh truy cập trái phép.

Hệ thống hỗ trợ **truy vấn nâng cao**, giúp quản lý báo cáo doanh thu, hàng tồn kho một cách hiệu quả.

**Nhược điểm:**

Chưa có giao diện đồ họa, chỉ mới tập trung vào hệ thống backend.

Chưa tích hợp công cụ phân tích dữ liệu nâng cao như BI hoặc AI để dự đoán xu hướng mua sắm.

Việc sao lưu dữ liệu cần phải có cơ chế tự động hóa hơn.

## **6.3. Đề xuất hướng phát triển tiếp theo**

Trong tương lai, hệ thống có thể được cải tiến theo các hướng sau:

* **Phát triển giao diện web hoặc ứng dụng di động** để giúp nhân viên dễ dàng sử dụng hệ thống.
* **Ứng dụng AI/ML** để dự đoán nhu cầu nhập hàng, tối ưu quản lý hàng tồn kho.
* **Tích hợp hệ thống với các phương thức thanh toán điện tử** như ví điện tử, thẻ tín dụng.
* **Tự động hóa quy trình sao lưu dữ liệu** bằng cách thiết lập lịch trình tự động chạy backup và gửi thông báo khi có lỗi.

# **PHÂN CÔNG NHIỆM VỤ**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **MSV** | **Họ và tên** | **Nhiệm vụ** |
| 1 | 1771020255 | Trịnh Công Hiệp | Sơ đồ,Tạo bảng, Nhập dữ liệu,Truy vấn T-SQL |
| 2 | 1771020287 | Phạm Thế Hoán | Truy Vấn, T-SQL, Word |

**DANH MỤC TÀI LIỆU THAM KHẢO**

**Nguyễn Xuân Huy** (2020). *Giáo trình Hệ quản trị cơ sở dữ liệu*. Nhà xuất bản Khoa học và Kỹ thuật.

**Phạm Văn Ất** (2017). *Giáo trình Cơ sở dữ liệu*. Nhà xuất bản Khoa học và Kỹ thuật.

**Lê Sỹ Vinh, Phạm Công Hoàng** (2019). *Nhập môn cơ sở dữ liệu*. Nhà xuất bản Đại học Quốc gia Hà Nội.

**Nguyễn Kim Khánh** (2018). *Quản trị cơ sở dữ liệu với SQL Server*. Nhà xuất bản Bách Khoa Hà Nội.

**Trần Ngọc Bảo** (2021). *Lập trình SQL Server từ cơ bản đến nâng cao*. Nhà xuất bản Hồng Đức.