BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

---------------------------



**XÂY DỰNG CHƯƠNG TRÌNH QUẢN LÝ BỆNH VIỆN**

**BÁO CÁO BÀI TẬP LỚN**

Môn học: Hệ quản trị cơ sở dữ liệu

**GVHD: Nguyễn Thị Thanh Thủy**

**SINH VIÊN THỰC HIỆN**

1. 2001223664, Lương Liêm Phong

2. 2001225914, Trần Khánh Vũ

3. 2001225547, Trần Minh Trí

4. 2001215670, Huỳnh Trương Bảo Duy

5. 2001223138, Đoàn Lê Anh Nguyên

TP. HỒ CHÍ MINH, tháng 10 năm 2024

**MỤC LỤC**

Mục Lục

[CHƯƠNG 1: GIỚI THIỆU 4](#_Toc184687161)

[1. Giới thiệu tổng quan đề tài 4](#_Toc184687162)

[2. Lý do chọn đề tài 4](#_Toc184687163)

[CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT CƠ SỞ DỮ LIỆU 5](#_Toc184687164)

[2.1. Xây dựng cơ sở dữ liệu: 5](#_Toc184687165)

[2.1.1 Tạo một CSDL khoảng 10 bảng theo đề tài thiết lập các ràng buộc khóa chính, khóa ngoại 5](#_Toc184687166)

[2.1.2 Thiết lập các ràng buộc check, unique, default 6](#_Toc184687167)

[2.1.3 Tạo lược đồ diagram 15](#_Toc184687168)

[2.1.4 Nhập dữ liệu mẫu (phù hợp với thực tế) 15](#_Toc184687169)

[2.2. Cài đặt yêu cầu xử lý: 22](#_Toc184687170)

[2.2.1 Câu lệnh Trigger 22](#_Toc184687171)

[2.2.2 Câu lệnh Procedure 32](#_Toc184687172)

[2.2.3 Câu lệnh Function 44](#_Toc184687173)

[2.2.4 Câu lệnh Cursor 51](#_Toc184687174)

[2.2.5 Câu lệnh phân quyền 55](#_Toc184687175)

[2.2.6 Sao lưu cơ sở dữ liệu 57](#_Toc184687176)

[2.3. Quản trị người dùng: 58](#_Toc184687177)

[CHƯƠNG 3: XÂY DỰNG ỨNG DỤNG 58](#_Toc184687178)

[3.1 Nghiệp vụ chia lịch trực cho bác sĩ 58](#_Toc184687179)

[3.2 Nghiệp vụ Khám Bệnh 63](#_Toc184687180)

[3.3 Nghiệp vụ Quản lý thuốc 73](#_Toc184687181)

[3.4 Nghiệp vụ quản lý hồ sơ bệnh nhân 82](#_Toc184687182)

[3.5 Nghiệp vụ nhập xuất viện 87](#_Toc184687183)

[DANH MỤC TÀI LIỆU THAM KHẢO 90](#_Toc184687184)

Mục lục hình ảnh

[HÌnh 1 Diagram 16](#_Toc184687137)

[HÌnh 2 Giao diện của chia lịch trực 59](#_Toc184687138)

[HÌnh 3 Giao diện quản lí bệnh nhân 64](#_Toc184687139)

[HÌnh 4 Giao diện tìm kiến bệnh nhân 64](#_Toc184687140)

[HÌnh 5 Giao diện xem chi tiết thông tin bệnh nhân 65](#_Toc184687141)

[HÌnh 6 Giao diện toa thuốc 65](#_Toc184687142)

[Hình 7 Giao diện tìm kiếm toa thuốc 66](#_Toc184687143)

[Hình 8 Giao diện xem chi tiết toa thuốc 66](#_Toc184687144)

[Hình 9 Giao diện phiếu khám 67](#_Toc184687145)

[Hình 10 Giao diện tìm kiếm phiếu khám 67](#_Toc184687146)

[Hình 11 Giao diện phiếu xét nghiệm 68](#_Toc184687147)

[Hình 12 Giao diện phiếu thanh toán 68](#_Toc184687148)

[Hình 13 Giao diện thông tin chi tiết thuốc 74](#_Toc184687149)

[Hình 14 Giao diện Thống kê phiếu nhập 74](#_Toc184687150)

[HÌnh 15 Giao diện lấy thuốc theo toa 75](#_Toc184687151)

[HÌnh 16 Giao diện nhập thuốc 75](#_Toc184687152)

[HÌnh 17 Giao diện Hồ sơ bệnh án 83](#_Toc184687153)

[HÌnh 18 Giao diện tìm kiếm hồ sơ bệnh án 84](#_Toc184687154)

[HÌnh 19 Giao diện form kiểm tra thông tin BHYT và Tổng số bệnh nhân đang điều trị ở bệnh viện 88](#_Toc184687155)

[HÌnh 20 Giao diện kiểm tra lịch sử nhập viện của bệnh nhân 88](#_Toc184687156)

[HÌnh 21 Giao diện nhập viện 89](#_Toc184687157)

[HÌnh 22 Giao diện xuất viện 89](#_Toc184687158)

MỤC LỤC BẢNG DỮ LIỆU

[Bảng 1 BENHNHAN 6](#_Toc184687221)

[Bảng 2 KHOA 6](#_Toc184687222)

[Bảng 3 BACSI 6](#_Toc184687223)

[Bảng 4 TOATHUOC 7](#_Toc184687224)

[Bảng 5 THUOC 7](#_Toc184687225)

[Bảng 6 CHITIETTOA 7](#_Toc184687226)

[Bảng 7 PHIEUKHAM 8](#_Toc184687227)

[Bảng 8 NHAPVIEN 9](#_Toc184687228)

[Bảng 9 RAVIEN 10](#_Toc184687229)

[Bảng 10 HOSOBENHNHAN 11](#_Toc184687230)

[Bảng 11 CATRUC 11](#_Toc184687231)

[Bảng 12 LICHTRUC 12](#_Toc184687232)

[Bảng 13 CHIDINHXETNGHIEM 12](#_Toc184687233)

[Bảng 14 THANHTOAN 13](#_Toc184687234)

[Bảng 15 LUUKHO 14](#_Toc184687235)

[Bảng 16 NHACCTHUOC 14](#_Toc184687236)

[Bảng 17 PHIEU\_NHAP\_THUOC 14](#_Toc184687237)

[Bảng 18 CHITIET\_PN\_THUOC 15](#_Toc184687238)

1. GIỚI THIỆU

## 1. Giới thiệu tổng quan đề tài

Đề tài "Quản lý bệnh viện" là một chủ đề có tính ứng dụng cao, tập trung vào việc xây dựng một hệ thống cơ sở dữ liệu nhằm quản lý các hoạt động chính của bệnh viện một cách hiệu quả. Hệ thống quản lý bệnh viện không chỉ lưu trữ và xử lý thông tin về bệnh nhân, bác sĩ, lịch khám chữa bệnh mà còn hỗ trợ quản lý các nguồn lực như phòng khám, thuốc men, trang thiết bị y tế và tài chính.

Mục tiêu chính của đề tài là thiết kế một cơ sở dữ liệu tối ưu, dễ sử dụng và có khả năng hỗ trợ đưa ra các quyết định quan trọng trong việc vận hành bệnh viện. Hệ thống này sẽ bao gồm các chức năng như:

* Quản lý thông tin bệnh nhân.
* Quản lý hồ sơ bệnh án.
* Quản lý ca trực.
* Quản lý thuốc.
* Quản lý xuất nhập viện.

## 2. Lý do chọn đề tài

Việc lựa chọn đề tài "Quản lý bệnh viện" xuất phát từ các lý do sau:

Tầm quan trọng của quản lý thông tin trong y tế:

* Ngành y tế đóng vai trò quan trọng trong đời sống con người. Việc quản lý hiệu quả thông tin trong bệnh viện không chỉ giúp cải thiện chất lượng dịch vụ mà còn giảm thiểu sai sót trong chăm sóc sức khỏe.

Ứng dụng thực tế và tính cấp thiết:

* Các bệnh viện hiện nay ngày càng chuyển đổi số, do đó nhu cầu xây dựng hệ thống quản lý dữ liệu chuyên nghiệp là rất cao. Việc nghiên cứu và phát triển hệ thống cơ sở dữ liệu quản lý bệnh viện sẽ đáp ứng yêu cầu thực tiễn của ngành.

Cơ hội học hỏi và phát triển kỹ năng:

* Đề tài này cung cấp cơ hội để áp dụng các kiến thức về hệ quản trị cơ sở dữ liệu, từ thiết kế mô hình quan hệ, tối ưu hóa truy vấn, đến triển khai trên các hệ quản trị như MySQL, SQL Server. Đồng thời, việc phân tích và giải quyết các bài toán thực tế trong bệnh viện giúp sinh viên nâng cao tư duy logic và khả năng làm việc nhóm.

Thách thức mang tính học thuật:

* Đề tài "Quản lý bệnh viện" đòi hỏi sự tổng hợp kiến thức từ nhiều lĩnh vực như quản lý thông tin, lập trình, bảo mật dữ liệu và tối ưu hóa hệ thống. Đây là cơ hội để sinh viên rèn luyện kỹ năng phân tích, thiết kế và phát triển các giải pháp công nghệ.

1. thiết kế VÀ CÀI ĐẶT cơ sở dữ liệu

## 2.1. Xây dựng cơ sở dữ liệu:

### 2.1.1 Tạo một CSDL khoảng 10 bảng theo đề tài thiết lập các ràng buộc khóa chính, khóa ngoại

**BENHNHAN** (MaBenhNhan, HoTen, NgaySinh, GioiTinh, DiaChi, SoDienThoai, TrangThai, MaBHYT, Tuoi, Tukhoa, Denkhoa)

**KHOA** (MaKhoa, TenKhoa)

**BACSI** (MaBacSi, HoTen, MaKhoa, SoDienThoai, SoCaTruc, MatKhau, ChucVu)

**TOATHUOC** (MaToa, NgayKetoa, SoNgayDung, NgayKetThuc, SoLoaiThuoc, MaBsKetoa)

**THUOC** (MaThuoc, TenThuoc, ThanhPhan, Gia)

**CHITIETTOA** (MaToa, MaThuoc, SoLuong, CachDung)

**PHIEUKHAM** (MaPhieuKham, MaBenhNhan, NgayKham, ChanDoan, MaBacSi, KetQua, GiaKham, MaToa)

**NHAPVIEN** (MaNhapVien, MaBenhNhan, NgayNhapVien, MaBacSi, MaPhong, LyDoNhapVien)

**RAVIEN** (MaRaVien, MaNhapVien, NgayRaVien, TinhTrangRaVien, MaBenhNhan, SoNgayNamVien)

**HOSOBENHAN** (MaHoSo, MaBenhNhan, NgayLapHoSo, MaBacSi, NoiDungHoSo)

**CATRUC** (MaCaTruc, ThoiGianCa)

**LICHTRUC** (MaBacSi, NgayTruc, MaCaTruc)

**CHIDINHXETNGHIEM** (MaChiDinh, MaPhieuKham, LoaiChiDinh, NgayChiDinh)

**THANHTOAN** (MaThanhToan, MaPhieuKham, MaRaVien, SoTien, NgayThanhToan)

**LUUKHO** (MaThuoc, TenThuoc, SoLuongTon, DonViTinh, NgayNhapKho, HanSuDung)

**NHACCTHUOC** (MaNCC, TenNCC, DiaChi)

**PHIEU\_NHAP\_THUOC** (MaPN, MaNCC, NgayNhap, ThanhTien)

**CHITIET\_PN\_THUOC** (MaPN, MaThuoc, SoLuong, Gia)

### 2.1.2 Thiết lập các ràng buộc check, unique, default

**Bảng dữ liệu BENHNHAN**

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not Null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaBenhNhan** | CHAR(10) | Not null | PRIMARY KEY | Mã Bệnh Nhân |
| 2 | **HoTen** | NVARCHAR(100) | Not null |  | Họ Tên Bệnh Nhân |
| 3 | **NgaySinh** | DATE | Null |  | Ngày Sinh |
| 4 | **GioiTinh** | NVARCHAR(10) | Not null |  | Giới Tính |
| 5 | **DiaChi** | NVARCHAR(255) | Null | DEFAULT ' ' | Địa Chỉ |
| 6 | **SoDienThoai** | NVARCHAR(15) | Null |  | Số Điện Thoại |
| 7 | **TrangThai** | NVARCHAR(50) | Null |  | Trạng Thái |
| 8 | **MaBHYT** | CHAR(15) | Null |  | Mã Bảo Hiểm Y Tế |
| 9 | **Tuoi** | INT | Null |  | Tuổi |
| 10 | **Tukhoa** | NVARCHAR(50) | Null |  | Từ Khoa |
| 11 | **Denkhoa** | NVARCHAR(50) | Null |  | Đến Khoa |

Bảng 1 BENHNHAN

Bảng dữ liệu KHOA

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not Null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaKhoa** | CHAR(10) | Not null | PRIMARY KEY | Mã Khoa |
| 2 | **TenKhoa** | NVARCHAR(50) | Not null |  | Tên Khoa |

Bảng 2 KHOA

Bảng dữ liệu BACSI

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not Null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaBacSi** | CHAR(10) | Not null | PRIMARY KEY | Mã Bác Sĩ |
| 2 | **HoTen** | NVARCHAR(100) | Not null |  | Họ tên bác sĩ |
| 3 | **MaKhoa** | CHAR(10) | Not null | FOREIGN KEY | Mã khoa (liên kết với bảng KHOA) |
| 4 | **SoDienThoai** | NVARCHAR(15) | Not null |  | Số điện thoại của bác sĩ |
| 5 | **SoCaTruc** | INT | Null | DEFAULT 0 | Số ca trực của bác sĩ (mặc định là 0) |
| 6 | **MatKhau** | CHAR(20) | Not null |  | Mật khẩu của bác sĩ |
| 7 | **ChucVu** | NVARCHAR(30) | Null |  | Chức vụ của bác sĩ |

Bảng 3 BACSI

Bảng dữ liệu TOATHUOC

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not Null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaToa** | CHAR(10) | Not null | PRIMARY KEY | Mã toa thuốc |
| 2 | **NgayKetoa** | DATE | Null |  | Ngày bác sĩ kê toa |
| 3 | **SoNgayDung** | INT | Null |  | Số ngày sử dụng thuốc |
| 4 | **NgayKetThuc** | DATE | Null |  | Ngày kết thúc sử dụng thuốc |
| 5 | **SoLoaiThuoc** | INT | Null |  | Số loại thuốc trong toa |
| 6 | **MaBsKetoa** | CHAR(10) | Not null | FOREIGN KEY | Mã bác sĩ kê toa, liên kết với bảng **BACSI** |

Bảng 4 TOATHUOC

Bảng dữ liệu THUOC

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaThuoc** | CHAR(10) | Not null | PRIMARY KEY | Mã thuốc, khóa chính |
| 2 | **TenThuoc** | NVARCHAR(50) | Null |  | Tên thuốc |
| 3 | **ThanhPhan** | NVARCHAR(50) | Null |  | Thành phần của thuốc |
| 4 | **Gia** | FLOAT | Null |  | Giá của thuốc |

Bảng 5 THUOC

Bảng dữ liệu CHITIETTOA

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaToa** | CHAR(10) | Not null | PRIMARY KEY (MaToa, MaThuoc) | Mã toa thuốc, khóa chính |
| 2 | **MaThuoc** | CHAR(10) | Not null | PRIMARY KEY (MaToa, MaThuoc) | Mã thuốc, khóa chính (phối hợp với MaToa) |
| 3 | **SoLuong** | INT | Null |  | Số lượng thuốc trong toa |
| 4 | **CachDung** | NVARCHAR(50) | Null |  | Cách dùng thuốc |

Bảng 6 CHITIETTOA

Bảng dữ liệu PHIEUKHAM

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaPhieuKham** | CHAR(10) | Not null | PRIMARY KEY | Mã phiếu khám, khóa chính |
| 2 | **MaBenhNhan** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES BENHNHAN(MaBenhNhan) | Mã bệnh nhân, khóa ngoại |
| 3 | **NgayKham** | DATE | Not null |  | Ngày khám |
| 4 | **ChanDoan** | NVARCHAR(255) | Null |  | Chẩn đoán |
| 5 | **MaBacSi** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES BACSI(MaBacSi) | Mã bác sĩ, khóa ngoại |
| 6 | **KetQua** | NVARCHAR(255) | Null |  | Kết quả |
| 7 | **GiaKham** | FLOAT | Null |  | Giá khám |
| 8 | **MaToa** | CHAR(10) | Null | FOREIGN KEY, REFERENCES TOATHUOC(MaToa) | Mã toa thuốc, khóa ngoại |

Bảng 7 PHIEUKHAM

Bảng dữ liệu NHAPVIEN

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaNhapVien** | CHAR(10) | Not null | PRIMARY KEY | Mã nhập viện, khóa chính |
| 2 | **MaBenhNhan** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES BENHNHAN(MaBenhNhan) | Mã bệnh nhân, khóa ngoại |
| 3 | **NgayNhapVien** | DATE | Not null |  | Ngày nhập viện |
| 4 | **MaBacSi** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES BACSI(MaBacSi) | Mã bác sĩ, khóa ngoại |
| 5 | **MaPhong** | CHAR(10) | Not null |  | Mã phòng |
| 6 | **LyDoNhapVien** | NVARCHAR(255) | Null |  | Lý do nhập viện |

Bảng 8 NHAPVIEN

Bảng dữ liệu RAVIEN

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaRaVien** | CHAR(10) | Not null | PRIMARY KEY | Mã ra viện, khóa chính |
| 2 | **MaNhapVien** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES NHAPVIEN(MaNhapVien) | Mã nhập viện, khóa ngoại |
| 3 | **NgayRaVien** | DATE | Not null |  | Ngày ra viện |
| 4 | **TinhTrangRaVien** | NVARCHAR(255) | Null |  | Tình trạng ra viện |
| 5 | **MaBenhNhan** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES BENHNHAN(MaBenhNhan) | Mã bệnh nhân, khóa ngoại |
| 6 | **SoNgayNamVien** | INT | Null |  | Số ngày nằm viện |

Bảng 9 RAVIEN

Bảng dữ liệu HOSOBENHNHAN

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaHoSo** | CHAR(10) | Not null | PRIMARY KEY | Mã hồ sơ, khóa chính |
| 2 | **MaBenhNhan** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES BENHNHAN(MaBenhNhan) | Mã bệnh nhân, khóa ngoại |
| 3 | **NgayLapHoSo** | DATE | Not null |  | Ngày lập hồ sơ |
| 4 | **MaBacSi** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES BACSI(MaBacSi) | Mã bác sĩ, khóa ngoại |
| 5 | **NoiDungHoSo** | NVARCHAR(255) | Null |  | Nội dung hồ sơ |

Bảng 10 HOSOBENHNHAN

Bảng dữ liệu CATRUC

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaCaTruc** | CHAR(10) | Not null | PRIMARY KEY | Mã ca trực, khóa chính |
| 2 | **ThoiGianCa** | NVARCHAR(50) | Not null |  | Thời gian ca (Ca Sáng, Ca Chiều, Ca Tối) |

Bảng 11 CATRUC

Bảng dữ liệu LICHTRUC

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaBacSi** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES BACSI(MaBacSi) | Mã bác sĩ, khóa ngoại từ bảng BACSI |
| 2 | **NgayTruc** | DATE | Not null |  | Ngày trực |
| 3 | **MaCaTruc** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES CATRUC(MaCaTruc) | Mã ca trực, khóa ngoại từ bảng CATRUC |

Bảng 12 LICHTRUC

Bảng dữ liệu CHIDINHXETNGHIEM

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaChiDinh** | CHAR(10) | Not null | PRIMARY KEY | Mã chỉ định, khóa chính |
| 2 | **MaPhieuKham** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES PHIEUKHAM(MaPhieuKham) | Mã phiếu khám, khóa ngoại từ bảng PHIEUKHAM |
| 3 | **LoaiChiDinh** | NVARCHAR(255) | Not null |  | Loại chỉ định xét nghiệm |
| 4 | **NgayChiDinh** | DATE | Not null |  | Ngày chỉ định xét nghiệm |

Bảng 13 CHIDINHXETNGHIEM

Bảng dữ liệu THANHTOAN

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaThanhToan** | CHAR(10) | Not null | PRIMARY KEY | Mã thanh toán, khóa chính |
| 2 | **MaPhieuKham** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES PHIEUKHAM(MaPhieuKham) | Mã phiếu khám, khóa ngoại từ bảng PHIEUKHAM |
| 3 | **MaRaVien** | CHAR(10) | Not null | FOREIGN KEY, REFERENCES RAVIEN(MaRaVien) | Mã ra viện, khóa ngoại từ bảng RAVIEN |
| 4 | **SoTien** | FLOAT | Not null |  | Số tiền thanh toán |
| 5 | **NgayThanhToan** | DATE | Not null |  | Ngày thanh toán |

Bảng 14 THANHTOAN

Bảng dữ liệu LUUKHO

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaThuoc** | CHAR(10) | Not null | PRIMARY KEY | Mã thuốc, khóa chính |
| 2 | **TenThuoc** | NVARCHAR(50) | Not null |  | Tên thuốc |
| 3 | **SoLuongTon** | INT | Not null |  | Số lượng tồn trong kho |
| 4 | **DonViTinh** | NVARCHAR(20) | Not null |  | Đơn vị tính của thuốc |
| 5 | **NgayNhapKho** | DATE | Not null |  | Ngày nhập kho của thuốc |
| 6 | **HanSuDung** | DATE | Not null |  | Hạn sử dụng của thuốc |
| 7 | **MaThuoc** | CHAR(10) | Not null | FOREIGN KEY REFERENCES THUOC(MaThuoc) | Khóa ngoại tham chiếu từ bảng **THUOC** |

Bảng 15 LUUKHO

Bảng dữ liệu NHACCTHUOC

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaNCC** | CHAR(10) | Not null | PRIMARY KEY | Mã nhà cung cấp, khóa chính |
| 2 | **TenNCC** | NVARCHAR(50) | Not null |  | Tên nhà cung cấp |
| 3 | **DiaChi** | NVARCHAR(50) | Not null |  | Địa chỉ nhà cung cấp |

Bảng 16 NHACCTHUOC

Bảng dữ liệu PHIEU\_NHAP\_THUOC

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaPN** | CHAR(10) | Not null | PRIMARY KEY | Mã phiếu nhập, khóa chính |
| 2 | **MaNCC** | CHAR(10) | Not null | FOREIGN KEY REFERENCES NHACCTHUOC(MaNCC) | Mã nhà cung cấp, khóa ngoại từ bảng **NHACCTHUOC** |
| 3 | **NgayNhap** | DATE | Not null |  | Ngày nhập thuốc |
| 4 | **ThanhTien** | FLOAT | Not null |  | Thành tiền của phiếu nhập |

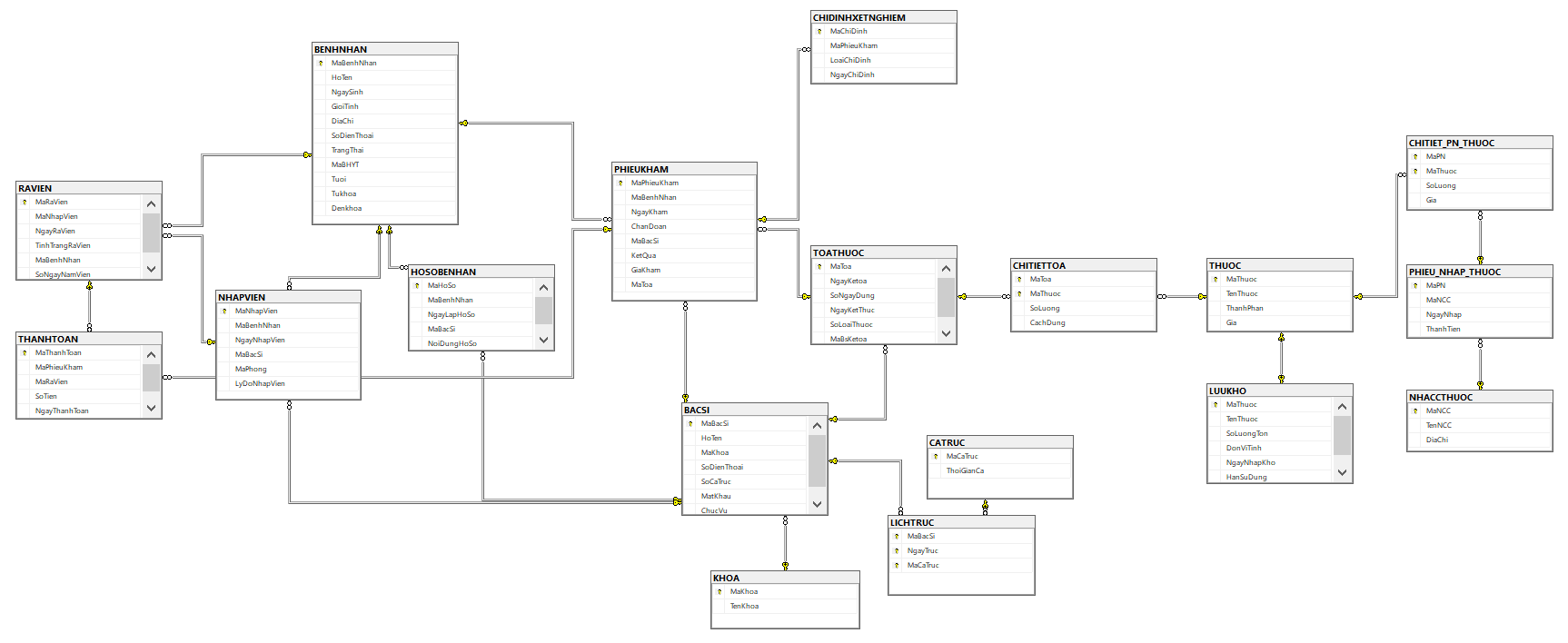
Bảng 17 PHIEU\_NHAP\_THUOC

Bảng dữ liệu CHITIET\_PN\_THUOC

| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Null / Not null** | **Ràng buộc** | **Mô tả** |
| --- | --- | --- | --- | --- | --- |
| 1 | **MaPN** | CHAR(10) | Not null | FOREIGN KEY REFERENCES PHIEU\_NHAP\_THUOC(MaPN) | Mã phiếu nhập, khóa ngoại từ bảng **PHIEU\_NHAP\_THUOC** |
| 2 | **MaThuoc** | CHAR(10) | Not null | FOREIGN KEY REFERENCES THUOC(MaThuoc) | Mã thuốc, khóa ngoại từ bảng **THUOC** |
| 3 | **SoLuong** | INT | Not null |  | Số lượng thuốc nhập |
| 4 | **Gia** | FLOAT | Not null |  | Giá của mỗi thuốc |

Bảng 18 CHITIET\_PN\_THUOC

### 2.1.3 Tạo lược đồ diagram



HÌnh 1 Diagram

### 2.1.4 Nhập dữ liệu mẫu (phù hợp với thực tế)

Nhập dữ liệu cho BENHNHAN

INSERT INTO BENHNHAN (MaBenhNhan, HoTen, NgaySinh, GioiTinh, DiaChi, SoDienThoai, TrangThai, MaBHYT, Tukhoa, Denkhoa)

VALUES

('BN00000001', N'Nguyễn Văn A', '1990-01-15', N'Nam', N'Hà Nội', '0905123456', N'Đang điều trị', 'BHYT0001', N'Nội khoa', N'Ngoại khoa'),

('BN00000002', N'Lê Thị B', '1985-05-21', N'Nữ', N'TP Hồ Chí Minh', '0917123456', N'Xuất viện', 'BHYT0002', N'Nhi khoa', N'Tai mũi họng'),

('BN00000003', N'Phạm Văn C', '1979-12-12', N'Nam', N'Đà Nẵng', '0938123456', N'Tái khám', 'BHYT0003', N'Nội khoa', N'Tim mạch'),

('BN00000004', N'Hoàng Thị D', '1992-07-23', N'Nữ', N'Hải Phòng', '0949123456', N'Đang điều trị', NULL, N'Nhi khoa', N'Sản phụ khoa'),

('BN00000005', N'Vũ Minh E', '1988-03-14', N'Nam', N'Cần Thơ', '0970123456', N'Chờ xét nghiệm', 'BHYT0005', N'Nội khoa', N'Ngoại khoa'),

('BN00000006', N'Đỗ Thị F', '1994-11-11', N'Nữ', N'Nha Trang', '0961123456', N'Xuất viện', NULL, N'Tim mạch', N'Hô hấp'),

('BN00000007', N'Trần Văn G', '1983-08-19', N'Nam', N'Vinh', '0952123456', N'Đang điều trị', 'BHYT0007', N'Nội khoa', N'Ngoại khoa'),

('BN00000008', N'Ngô Thị H', '1991-10-30', N'Nữ', N'Quảng Ninh', '0923123456', N'Chờ phẫu thuật', 'BHYT0008', N'Sản phụ khoa', N'Tai mũi họng'),

('BN00000009', N'Lương Minh I', '1975-09-05', N'Nam', N'Bình Dương', '0914123456', N'Tái khám', 'BHYT0009', N'Nội khoa', N'Ngoại khoa'),

('BN00000010', N'Phan Thị K', '1987-04-28', N'Nữ', N'Quảng Nam', '0906123456', N'Xuất viện', NULL, N'Nhi khoa', N'Răng hàm mặt'),

('BN00000011', N'Huỳnh Văn L', '1970-06-14', N'Nam', N'Huế', '0907123456', N'Chờ xét nghiệm', 'BHYT0011', N'Nội khoa', N'Tim mạch'),

('BN00000012', N'Lý Thị M', '1993-02-02', N'Nữ', N'Tây Ninh', '0908123456', N'Đang điều trị', NULL, N'Nhi khoa', N'Hô hấp'),

('BN00000013', N'Hồ Văn N', '1981-11-22', N'Nam', N'Hậu Giang', '0913123456', N'Tái khám', 'BHYT0013', N'Tim mạch', N'Nội tiết'),

('BN00000014', N'Tạ Thị O', '1996-07-16', N'Nữ', N'Hòa Bình', '0920123456', N'Xuất viện', NULL, N'Nhi khoa', N'Răng hàm mặt'),

('BN00000015', N'Cao Văn P', '1984-10-11', N'Nam', N'Bắc Giang', '0930123456', N'Đang điều trị', 'BHYT0015', N'Tai mũi họng', N'Nội khoa');

Nhập dữ liệu cho bảng KHOA

INSERT INTO KHOA (MaKhoa, TenKhoa) VALUES

('K01', N'Nội tổng hợp'),

('K02', N'Ngoại tổng hợp'),

('K03', N'Nhi khoa'),

('K04', N'Sản khoa'),

('K05', N'Tim mạch'),

('K06', N'Tai mũi họng'),

('K07', N'Tiêu hóa'),

('K08', N'Nhãn khoa'),

('K09', N'Da liễu'),

('K10', N'Phục hồi chức năng');

Nhập dữ liệu cho bảng BACSI

INSERT INTO BACSI (MaBacSi, HoTen, MaKhoa, SoDienThoai, SoCaTruc, MatKhau, ChucVu)

VALUES

('B001', N'Nguyễn Văn A', 'K01', '0901234567', 3, 'password123', N'Bác sĩ'),

('B002', N'Phan Thị B', 'K02', '0902345678', 4, 'password123', N'Bác sĩ'),

('B003', N'Hoàng Minh C', 'K03', '0903456789', 2, 'password123', N'Bác sĩ'),

('B004', N'Trần Thị D', 'K01', '0904567890', 5, 'password123', N'Bác sĩ'),

('B005', N'Lê Văn E', 'K04', '0905678901', 1, 'password123', N'Bác sĩ'),

('B006', N'Ngô Thị F', 'K02', '0906789012', 6, 'password123', N'Bác sĩ'),

('B007', N'Vũ Minh G', 'K03', '0907890123', 0, 'password123', N'Bác sĩ'),

('B008', N'Phạm Thị H', 'K04', '0908901234', 7, 'password123', N'Bác sĩ'),

('B009', N'Toàn Hoàng I', 'K01', '0909012345', 8, 'password123', N'Bác sĩ'),

('B010', N'Hồng Thị J', 'K02', '0900123456', 9, 'password123', N'Bác sĩ');

Nhập dữ liệu cho bảng TOATHUOC

INSERT INTO TOATHUOC (MaToa, NgayKetoa, SoNgayDung, NgayKetThuc, SoLoaiThuoc, MaBsKetoa)

VALUES

('T001', '2024-11-01', 7, '2024-11-08', 3, 'B001'),

('T002', '2024-11-02', 5, '2024-11-07', 2, 'B002'),

('T003', '2024-11-03', 10, '2024-11-13', 4, 'B003'),

('T004', '2024-11-04', 14, '2024-11-18', 5, 'B004'),

('T005', '2024-11-05', 3, '2024-11-08', 1, 'B005'),

('T006', '2024-11-06', 7, '2024-11-13', 6, 'B006'),

('T007', '2024-11-07', 8, '2024-11-15', 2, 'B007'),

('T008', '2024-11-08', 12, '2024-11-20', 3, 'B008'),

('T009', '2024-11-09', 4, '2024-11-13', 4, 'B009'),

('T010', '2024-11-10', 6, '2024-11-16', 3, 'B010');

Nhập dữ liệu cho bảng THUOC

INSERT INTO THUOC (MaThuoc, TenThuoc, ThanhPhan, Gia)

VALUES

('T001', 'Paracetamol', 'Paracetamol 500mg', 2000),

('T002', 'Ibuprofen', 'Ibuprofen 200mg', 2500),

('T003', 'Amoxicillin', 'Amoxicillin 500mg', 3000),

('T004', 'Vitamin C', 'Vitamin C 1000mg', 1500),

('T005', 'Aspirin', 'Aspirin 500mg', 1800),

('T006', 'Doxycycline', 'Doxycycline 100mg', 3500),

('T007', 'Cetirizine', 'Cetirizine 10mg', 2200),

('T008', 'Omeprazole', 'Omeprazole 20mg', 2800),

('T009', 'Loratadine', 'Loratadine 10mg', 2400),

('T010', 'Diphenhydramine', 'Diphenhydramine 25mg', 2100);

Nhập dữ liệu cho bảng CHITIETTOA

INSERT INTO CHITIETTOA (MaToa, MaThuoc, SoLuong, CachDung)

VALUES

('T001', 'T001', 2, N'Uống 1 viên mỗi 4 giờ'),

('T001', 'T002', 1, N'Uống 1 viên mỗi 6 giờ'),

('T002', 'T003', 3, N'Uống 1 viên mỗi 8 giờ'),

('T002', 'T004', 2, N'Uống 1 viên mỗi ngày'),

('T003', 'T005', 1, N'Uống 1 viên mỗi 6 giờ'),

('T003', 'T006', 2, N'Uống 1 viên mỗi 8 giờ'),

('T004', 'T007', 1, N'Uống 1 viên mỗi ngày'),

('T004', 'T008', 1, N'Uống 1 viên vào buổi sáng'),

('T005', 'T009', 3, N'Uống 1 viên mỗi tối'),

('T005', 'T010', 2, N'Uống 1 viên mỗi ngày');

Nhập dữ liệu cho PHIEUKHAM

INSERT INTO PHIEUKHAM (MaPhieuKham, MaBenhNhan, NgayKham, ChanDoan, MaBacSi, KetQua, GiaKham, MaToa)VALUES

('PK001', 'BN00000001', '2024-11-01', N'Viêm họng cấp', 'B002', N'Đã điều trị, khỏe mạnh', 100000, 'T001')

INSERT INTO PHIEUKHAM (MaPhieuKham, MaBenhNhan, NgayKham, ChanDoan, MaBacSi, KetQua, GiaKham, MaToa)VALUES

('PK002', 'BN00000001', '2024-11-02', N'Cảm cúm', 'B003', N'Điều trị khỏi, tiếp tục theo dõi', 120000, 'T002')

INSERT INTO PHIEUKHAM (MaPhieuKham, MaBenhNhan, NgayKham, ChanDoan, MaBacSi, KetQua, GiaKham, MaToa)VALUES

('PK003', 'BN00000003', '2024-11-03', N'Đau bụng', 'B004', N'Khám và điều trị ổn', 110000, 'T003')INSERT INTO PHIEUKHAM (MaPhieuKham, MaBenhNhan, NgayKham, ChanDoan, MaBacSi, KetQua, GiaKham, MaToa)VALUES

('PK004', 'BN00000004', '2024-11-04', N'Viêm phổi', 'B005', N'Điều trị khỏi, đã hồi phục', 150000, 'T004')

INSERT INTO PHIEUKHAM (MaPhieuKham, MaBenhNhan, NgayKham, ChanDoan, MaBacSi, KetQua, GiaKham, MaToa)VALUES

('PK005', 'BN00000005', '2024-11-05', N'Viêm dạ dày', 'B006', N'Điều trị, tái khám sau 1 tuần', 130000, 'T005')

INSERT INTO PHIEUKHAM (MaPhieuKham, MaBenhNhan, NgayKham, ChanDoan, MaBacSi, KetQua, GiaKham, MaToa)VALUES

('PK006', 'BN00000006', '2024-11-06', N'Chấn thương đầu gối', 'B007', N'Điều trị phục hồi chức năng', 140000, 'T006')

INSERT INTO PHIEUKHAM (MaPhieuKham, MaBenhNhan, NgayKham, ChanDoan, MaBacSi, KetQua, GiaKham, MaToa)VALUES

('PK007', 'BN00000008', '2024-11-07', N'Viêm gan', 'B008', N'Điều trị và theo dõi lâu dài', 160000, 'T007')

INSERT INTO PHIEUKHAM (MaPhieuKham, MaBenhNhan, NgayKham, ChanDoan, MaBacSi, KetQua, GiaKham, MaToa)VALUES

('PK008', 'BN00000008', '2024-11-08', N'Thử nghiệm máu', 'B009', N'Kết quả bình thường', 180000, 'T008')

Nhập liệu cho bảng NHAPVIEN

INSERT INTO NHAPVIEN (MaNhapVien, MaBenhNhan, NgayNhapVien, MaBacSi, MaPhong, LyDoNhapVien)

VALUES

('NV001', 'BN00000001', '2024-01-10', 'B001', 'P001', N'Cảm cúm nặng'),

('NV002', 'BN00000005', '2024-01-15', 'B002', 'P002', N'Viêm phổi cấp'),

('NV003', 'BN00000002', '2024-02-01', 'B003', 'P001', N'Tai biến mạch máu não'),

('NV004', 'BN00000003', '2024-02-05', 'B004', 'P003', N'Chấn thương sọ não'),

('NV005', 'BN00000004', '2024-02-10', 'B005', 'P004', N'Viêm dạ dày mãn tính'),

('NV006', 'BN00000006', '2024-02-12', 'B001', 'P001', N'Huyết áp cao'),

('NV007', 'BN00000007', '2024-02-15', 'B002', 'P002', N'Xương khớp'),

('NV008', 'BN00000008', '2024-03-01', 'B003', 'P003', N'Viêm gan cấp tính'),

('NV009', 'BN00000009', '2024-03-05', 'B004', 'P004', N'Bệnh tim mạch'),

('NV010', 'BN00000010', '2024-03-10', 'B005', 'P001', N'Cảm cúm, sốt cao');

Nhập liệu cho bảng RAVIEN

INSERT INTO RAVIEN (MaRaVien, MaNhapVien, NgayRaVien, TinhTrangRaVien, MaBenhNhan, SoNgayNamVien)VALUES

('RV001', 'NV001', '2024-01-20', N'Khỏi bệnh, sức khỏe ổn định', 'BN00000001', 10),

('RV002', 'NV002', '2024-01-25', N'Điều trị ổn định, sức khỏe cải thiện', 'BN00000005', 11),

('RV003', 'NV003', '2024-02-10', N'Tình trạng ổn định, ra viện sau phẫu thuật', 'BN00000002', 30),

('RV004', 'NV004', '2024-02-12', N'Phẫu thuật thành công, sức khỏe phục hồi', 'BN00000003', 20),

('RV005', 'NV005', '2024-02-15', N'Khỏi bệnh, ra viện', 'BN00000004', 5),

('RV006', 'NV006', '2024-02-18', N'Khỏi bệnh, cần theo dõi thêm', 'BN00000006', 7),

('RV007', 'NV007', '2024-02-20', N'Tình trạng ổn định, xuất viện sau điều trị', 'BN00000007', 15),

('RV008', 'NV008', '2024-03-05', N'Thực hiện phẫu thuật thành công, sức khỏe tốt', 'BN00000008', 25),

('RV009', 'NV009', '2024-03-08', N'Ra viện sau điều trị tim mạch', 'BN00000009', 18),

('RV010', 'NV010', '2024-03-12', N'Sức khỏe ổn định, đã khỏi bệnh', 'BN00000010', 7);

Nhập liệu cho bảng HOSOBENHAN

INSERT INTO HOSOBENHAN (MaHoSo, MaBenhNhan, NgayLapHoSo, MaBacSi, NoiDungHoSo) VALUES

('HS001', 'BN00000001', '2024-01-10', 'B001', N'Khám bệnh sốt cao, chẩn đoán cảm cúm'),

('HS002', 'BN00000002', '2024-02-15', 'B002', N'Khám bệnh đau dạ dày, chẩn đoán viêm loét dạ dày'),

('HS003', 'BN00000003', '2024-03-01', 'B003', N'Khám bệnh mệt mỏi kéo dài, chẩn đoán thiếu máu'),

('HS004', 'BN00000004', '2024-04-05', 'B004', N'Khám bệnh ho kéo dài, chẩn đoán viêm phế quản'),

('HS005', 'BN00000005', '2024-05-10', 'B005', N'Khám bệnh đau lưng, chẩn đoán thoái hóa đốt sống'),

('HS006', 'BN00000006', '2024-06-12', 'B006', N'Khám bệnh huyết áp cao, chẩn đoán tăng huyết áp'),

('HS007', 'BN00000007', '2024-07-20', 'B007', N'Khám bệnh khó thở, chẩn đoán bệnh phổi tắc nghẽn mãn tính'),

('HS008', 'BN00000008', '2024-08-22', 'B008', N'Khám bệnh đau đầu dữ dội, chẩn đoán đau nửa đầu'),

('HS009', 'BN00000009', '2024-09-30', 'B009', N'Khám bệnh chóng mặt, chẩn đoán rối loạn tiền đình'),

('HS010', 'BN00000010', '2024-10-15', 'B010', N'Khám bệnh sưng khớp, chẩn đoán viêm khớp dạng thấp');

Nhập liệu cho bảng CATRUC

insert into CATRUC values ('SA', N'Ca Sáng');

insert into CATRUC values ('CH',N'Ca Chiều');

insert into CATRUC values ('TO', N'Ca Tối');

insert into CATRUC values ('DE', N'Ca Đêm');

Nhập liệu cho bảng LICHTRUC

INSERT INTO LICHTRUC (MaBacSi, NgayTruc, MaCaTruc) VALUES

('B001', '2024-01-01', 'SA'), -- Bác sĩ B001, ngày trực 2024-01-01, Ca sáng

('B002', '2024-01-01', 'CH'), -- Bác sĩ B002, ngày trực 2024-01-01, Ca chiều

('B003', '2024-01-02', 'TO'), -- Bác sĩ B003, ngày trực 2024-01-02, Ca tối

('B004', '2024-01-02', 'DE'), -- Bác sĩ B004, ngày trực 2024-01-02, Ca sáng

('B005', '2024-01-03', 'SA'), -- Bác sĩ B005, ngày trực 2024-01-03, Ca chiều

('B006', '2024-01-03', 'CH'), -- Bác sĩ B006, ngày trực 2024-01-03, Ca tối

('B007', '2024-01-04', 'TO'), -- Bác sĩ B007, ngày trực 2024-01-04, Ca sáng

('B008', '2024-01-04', 'DE'), -- Bác sĩ B008, ngày trực 2024-01-04, Ca chiều

('B009', '2024-01-05', 'SA'), -- Bác sĩ B009, ngày trực 2024-01-05, Ca tối

('B010', '2024-01-05', 'CH'); -- Bác sĩ B010, ngày trực 2024-01-05, Ca sáng

Nhập liệu cho bảng CHIDINHXETNGHIEM

INSERT INTO CHIDINHXETNGHIEM (MaChiDinh, MaPhieuKham, LoaiChiDinh, NgayChiDinh)

VALUES

('CD001', 'PK001', N'Xét nghiệm máu', '2024-01-10'),

('CD002', 'PK002', N'Xét nghiệm nước tiểu', '2024-01-12'),

('CD003', 'PK003', N'Chụp X-quang', '2024-02-05'),

('CD004', 'PK004', N'Siêu âm', '2024-02-20'),

('CD005', 'PK005', N'Nội soi dạ dày', '2024-03-10'),

('CD006', 'PK006', N'Xét nghiệm đường huyết', '2024-04-01'),

('CD007', 'PK007', N'Soi cổ tử cung', '2024-05-15'),

('CD008', 'PK008', N'Điện tâm đồ', '2024-06-18');

Nhập liệu cho bảng THANHTOAN

INSERT INTO THANHTOAN (Mathanhtoan, MaPhieuKham, SoTien, NgayThanhToan) VALUES

('TT001', 'PK001', 500000, '2024-11-01'),

('TT002', 'PK002', 300000, '2024-11-02'),

('TT003', 'PK003', 450000, '2024-11-03'),

('TT004', 'PK004', 600000, '2024-11-04'),

('TT005', 'PK005', 700000, '2024-11-05'),

('TT006', 'PK006', 350000, '2024-11-06'),

('TT007', 'PK007', 800000, '2024-11-07');

Nhập liệu cho bảng LUUKHO

INSERT INTO LUUKHO (MaThuoc, TenThuoc, SoLuongTon, DonViTinh, NgayNhapKho, HanSuDung) VALUES

('T001', N'Paracetamol', 100, N'Viên', '2024-01-01', '2026-01-01'),

('T002', N'Ibuprofen', 150, N'Viên', '2024-01-05', '2026-01-05'),

('T003', N'Amoxicillin', 200, N'Viên', '2024-01-10', '2025-12-10'),

('T004', N'Vitamin C', 120, N'Viên', '2024-01-12', '2026-01-12'),

('T005', N'Aspirin', 80, N'Viên', '2024-01-15', '2026-01-15'),

('T006', N'Doxycycline', 90, N'Viên', '2024-01-20', '2025-12-20'),

('T007', N'Cetirizine', 50, N'Viên', '2024-01-22', '2025-12-22'),

('T008', N'Omeprazole', 70, N'Viên', '2024-01-25', '2026-01-25'),

('T009', N'Loratadine', 300, N'Viên', '2024-01-28', '2026-01-28'),

('T010', N'Diphenhydramine', 110, N'Viên', '2024-02-01', '2026-02-01');

Nhập liệu cho bảng NHACCTHUOC

INSERT INTO NHACCTHUOC (MaNCC, TenNCC, DiaChi) VALUES

('NCC001', N'Nhà Cung Cấp Vnsite', N'Hà Nội, Việt Nam'),

('NCC002', N'Nhà Cung Cấp BioMax', N'TP. Hồ Chí Minh, Việt Nam'),

('NCC003', N'Nhà Cung Cấp CenTerPill', N'Đà Nẵng, Việt Nam'),

('NCC004', N'Nhà Cung Cấp Diaxienon', N'Hải Phòng, Việt Nam'),

('NCC005', N'Nhà Cung Cấp Exidon', N'Vũng Tàu, Việt Nam'),

('NCC006', N'Nhà Cung Cấp FreckPill', N'Cần Thơ, Việt Nam'),

('NCC007', N'Nhà Cung Cấp Gironaxin', N'Thái Nguyên, Việt Nam'),

('NCC008', N'Nhà Cung Cấp HibrigthPill', N'Bình Dương, Việt Nam'),

('NCC009', N'Nhà Cung Cấp Igabonich', N'Quảng Ninh, Việt Nam'),

('NCC010', N'Nhà Cung Cấp JabonPill', N'An Giang, Việt Nam');

Nhập dữ liệu cho bảng PHIEU\_NHAP\_THUOC

INSERT INTO PHIEU\_NHAP\_THUOC (MaPN, MaNCC, NgayNhap, ThanhTien) VALUES

('PN001', 'NCC001', '2024-01-10', 150000),

('PN002', 'NCC002', '2024-01-15', 200000),

('PN003', 'NCC003', '2024-01-20', 180000),

('PN004', 'NCC004', '2024-01-25', 210000),

('PN005', 'NCC005', '2024-02-01', 250000),

('PN006', 'NCC006', '2024-02-05', 300000),

('PN007', 'NCC007', '2024-02-10', 220000),

('PN008', 'NCC008', '2024-02-15', 270000),

('PN009', 'NCC009', '2024-02-20', 320000),

('PN010', 'NCC010', '2024-02-25', 350000);

Nhập dữ liệu cho bảng CHITIET\_PN\_THUOC

INSERT INTO CHITIET\_PN\_THUOC (MaPN, MaThuoc, SoLuong, Gia) VALUES

('PN001', 'T001', 100, 1500),

('PN001', 'T002', 50, 3000),

('PN001', 'T003', 30, 5000),

('PN002', 'T004', 120, 1200),

('PN002', 'T005', 60, 2200),

('PN003', 'T006', 110, 2500),

('PN003', 'T007', 80, 1500),

('PN004', 'T008', 70, 1800),

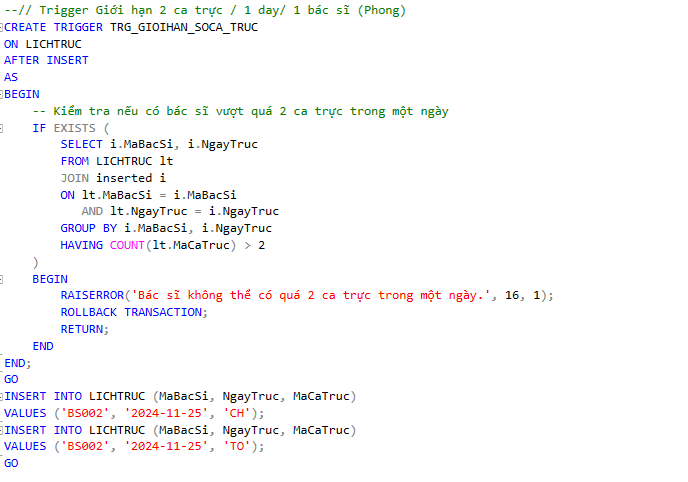
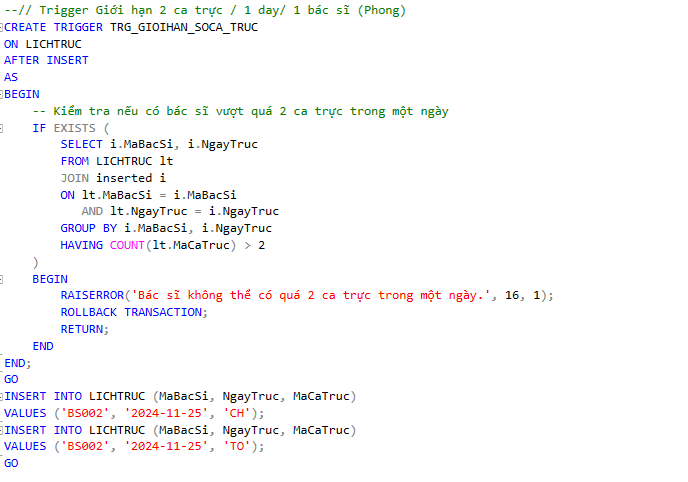
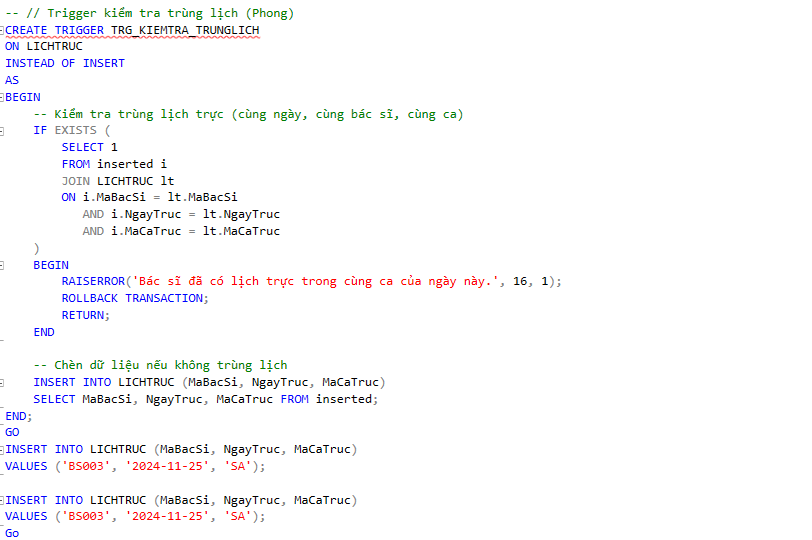
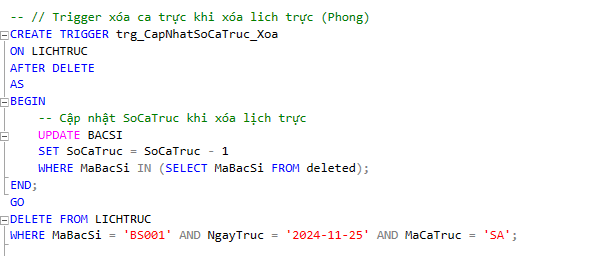
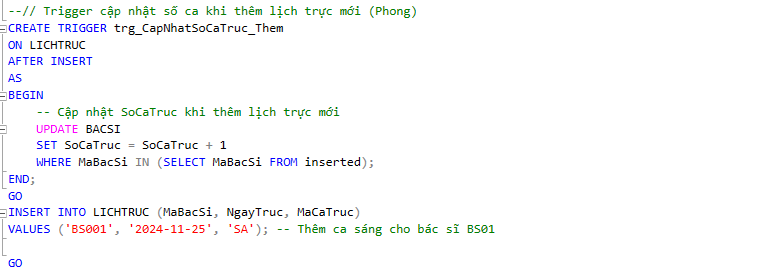
('PN005', 'T009', 130, 2000),

('PN005', 'T010', 90, 3500);

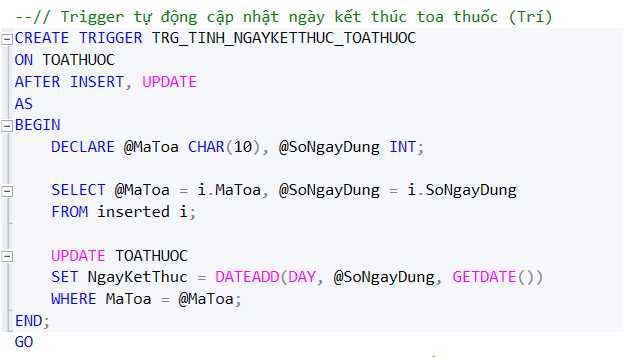
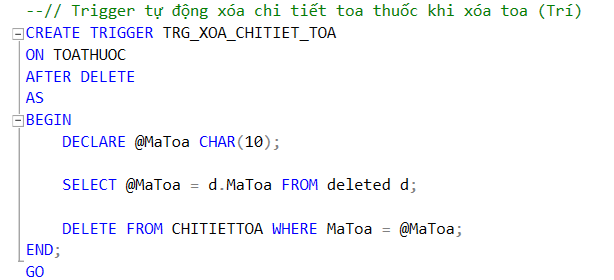
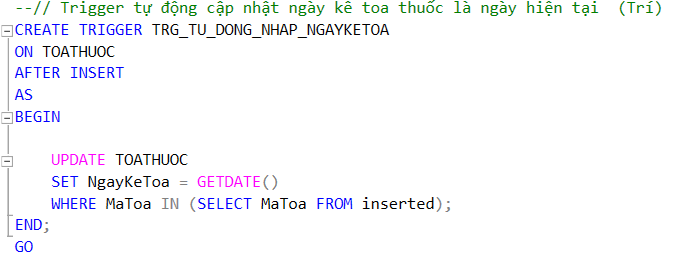
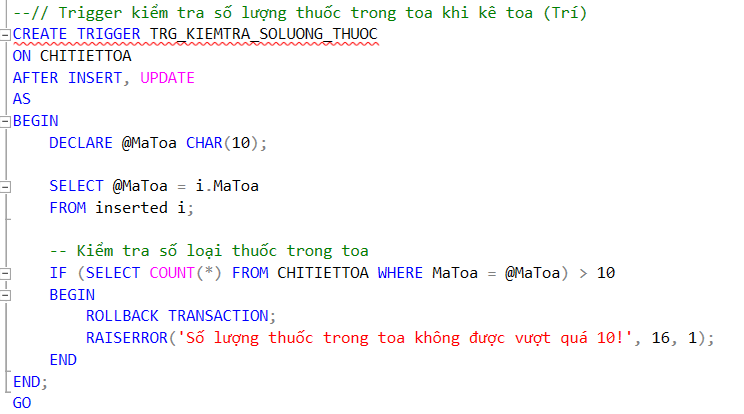
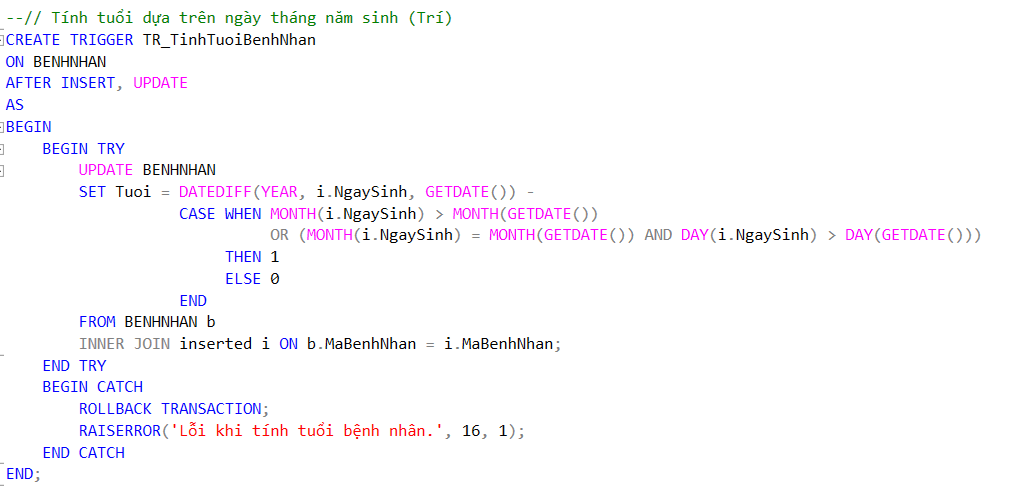
## 2.2. Cài đặt yêu cầu xử lý:

### 2.2.1 Câu lệnh Trigger

**2.2.1.1 Phong**



**2.2.1.2 Trí**

**2.2.1.3 Vũ**

----------------Vũ-------------------------

------------------CẬP NHẬT SLTONKHI KHI NHẬP THUỐC MỚI-----------Vũ-------

CREATE TRIGGER trg\_UpdateSoLuongTon

ON CHITIET\_PN\_THUOC

AFTER INSERT

AS

BEGIN

-- Cập nhật SOLUONGTON trong bảng THUOC

UPDATE LUUKHO

SET SOLUONGTON = SOLUONGTON + i.SoLuong

FROM inserted i

WHERE LUUKHO.MATHUOC = i.MATHUOC;

END;

----------------Vũ-------------------------

CREATE TRIGGER CN\_THANHTIEN\_PHIEUNHAP ON CHITIET\_PN\_THUOC

FOR INSERT

AS

BEGIN

UPDATE PHIEU\_NHAP\_THUOC

SET THANHTIEN=THANHTIEN+(SELECT SUM(SoLuong\*GIA)FROM CHITIET\_PN\_THUOC WHERE MAPN=(SELECT MAPN FROM inserted))

WHERE MAPN=(SELECT MAPN FROM inserted)

END

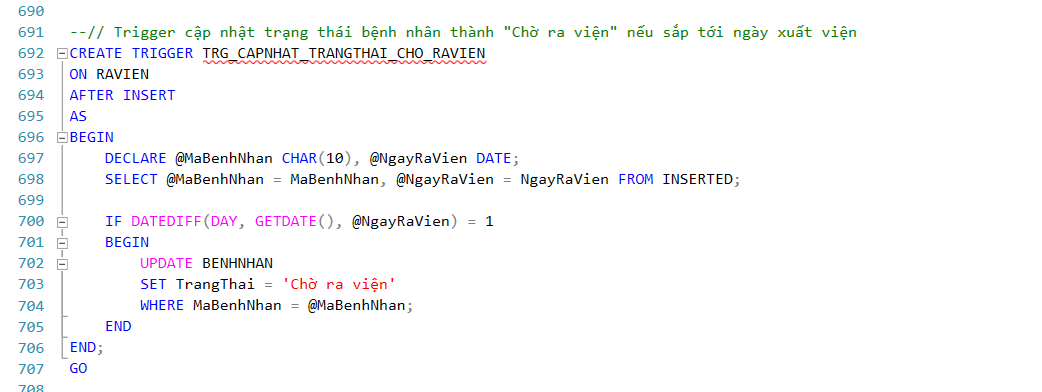
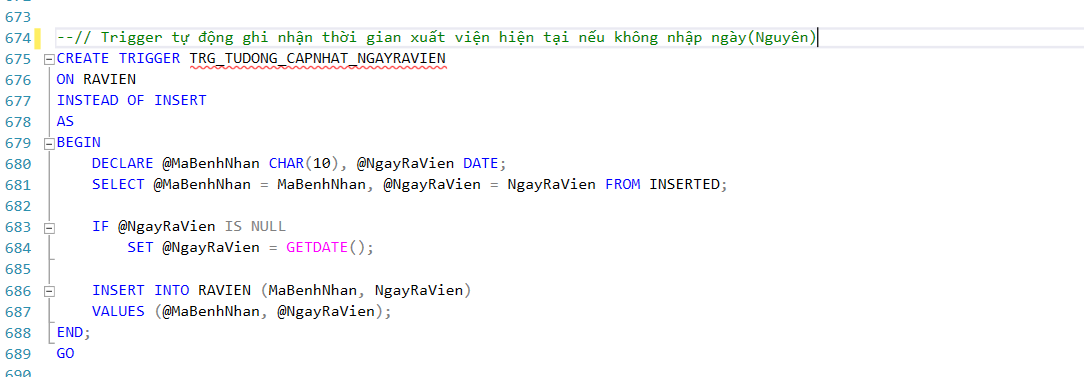
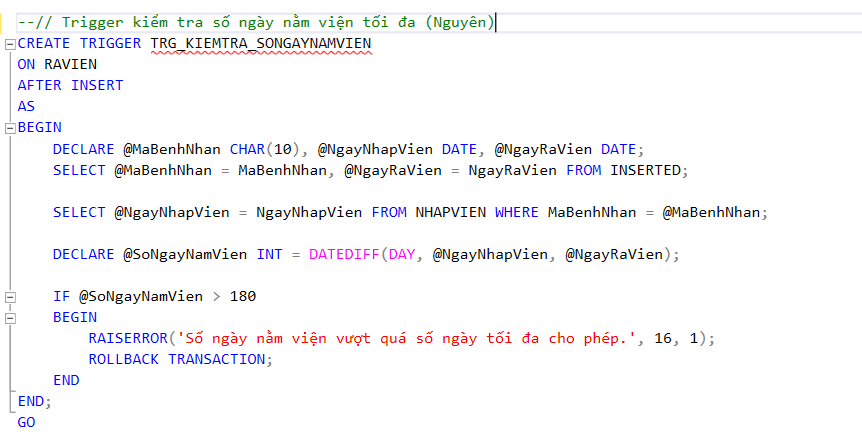
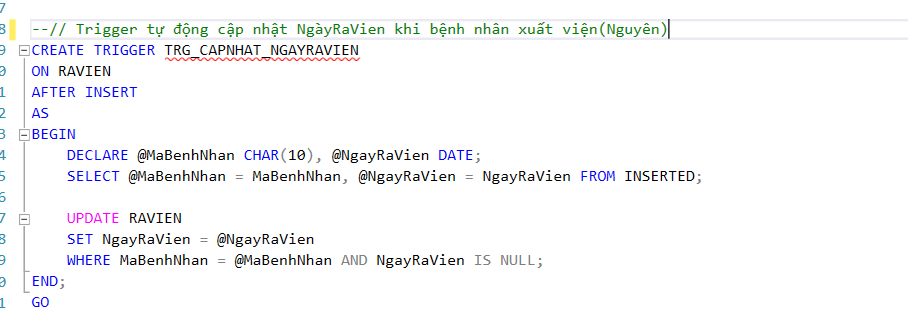
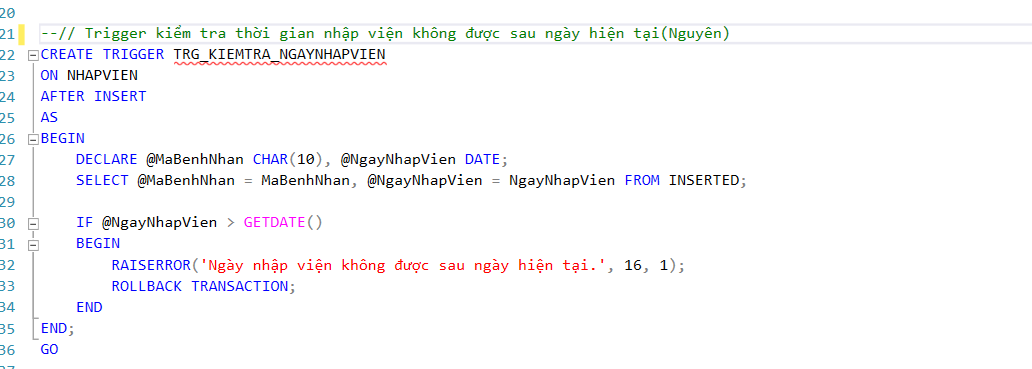
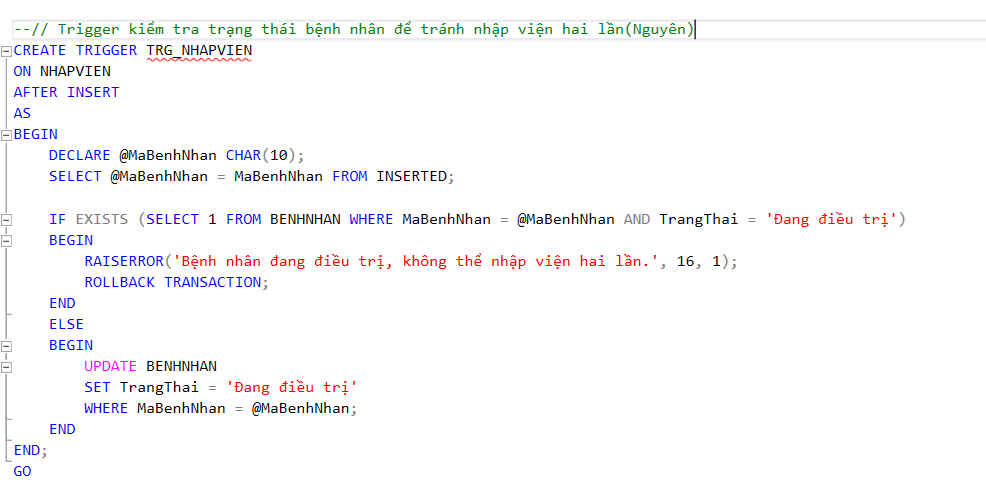
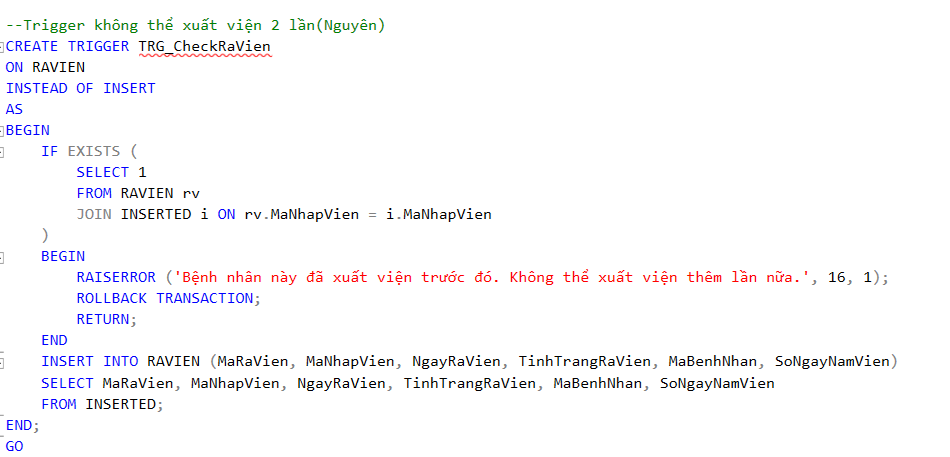
**2.2.1.4 Bảo Duy**A screenshot of a computer code

Description automatically generatedA screenshot of a computer

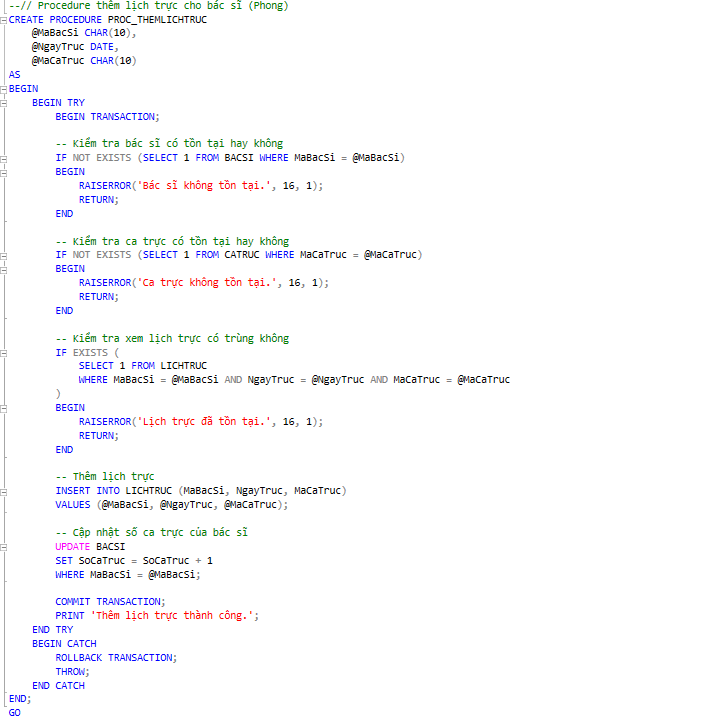
Description automatically generatedA screenshot of a computer code

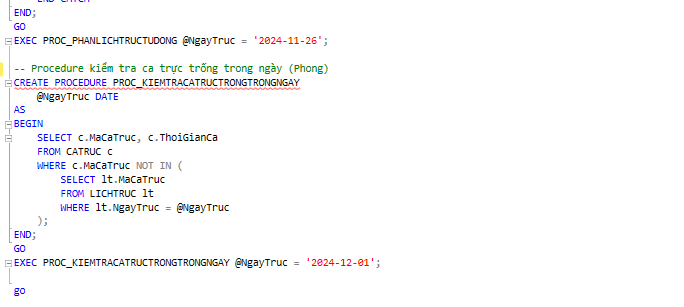
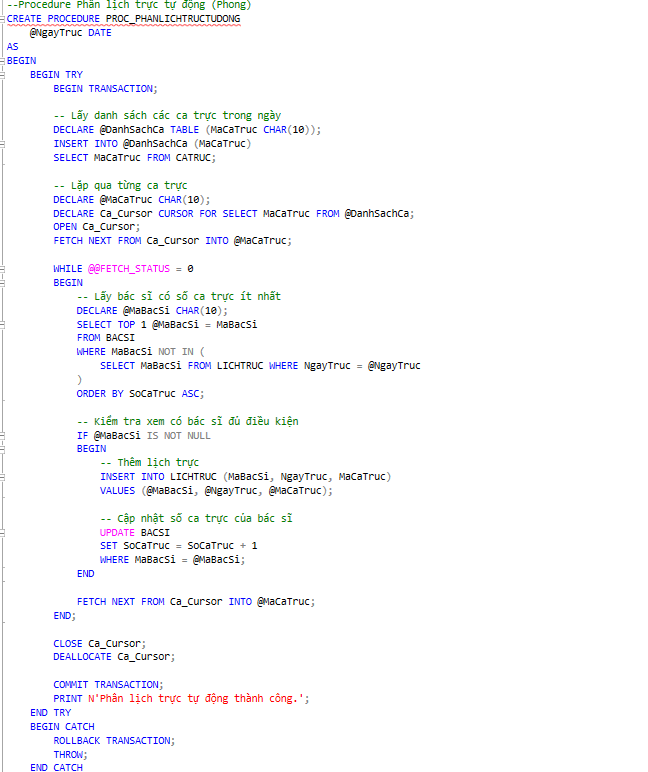
Description automatically generated

**2.2.1.4 Nguyên**



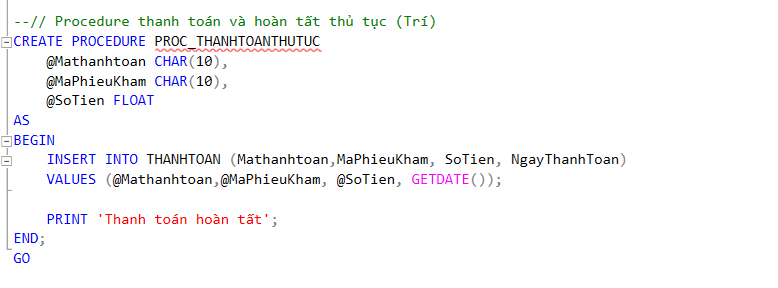
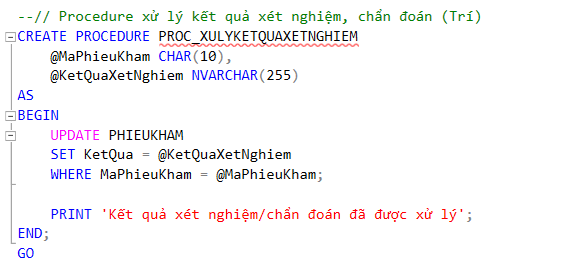
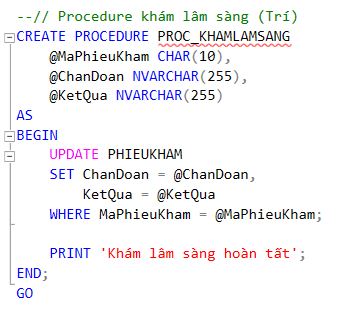
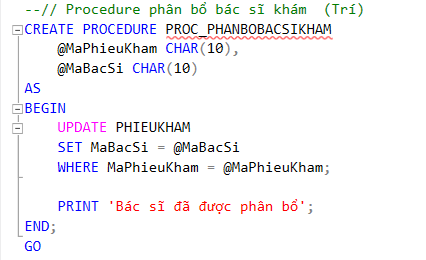
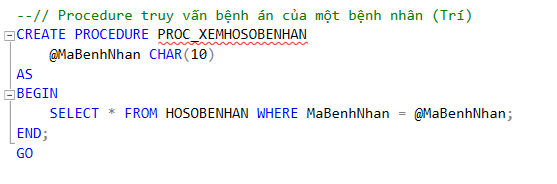
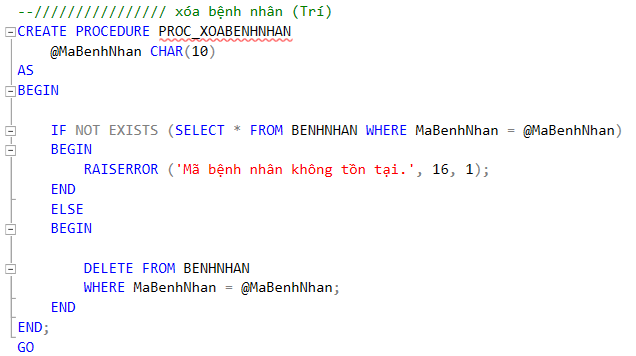
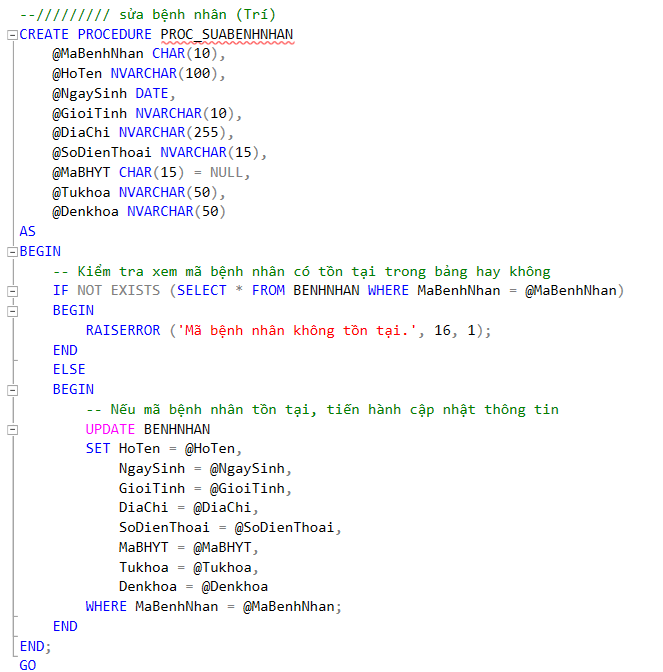
### 2.2.2 Câu lệnh Procedure

**2.2.2.1 Phong**



**2.2.2.2 Trí**





**2.2.2.3 Vũ**

----------------------Vũ----------------------------------------

-- --------------------PROC THÊM PHIẾU NHẬP THUỐC MỚI------------------------------Vũ---------------

CREATE PROCEDURE ThemPhieuNhapThuoc

@MAPN CHAR(10),

@MANCC CHAR(10),

@NGAYNHAP DATE,

@THANHTIEN FLOAT

AS

BEGIN

-- Kiểm tra xem nhà cung cấp có tồn tại hay không

IF NOT EXISTS (SELECT 1 FROM NHACCTHUOC WHERE MANCC = @MANCC)

BEGIN

PRINT 'Nhà cung cấp không tồn tại.';

RETURN;

END

-- Thêm phiếu nhập thuốc

INSERT INTO PHIEU\_NHAP\_THUOC (MAPN, MANCC, NGAYNHAP, THANHTIEN)

VALUES (@MAPN, @MANCC, @NGAYNHAP, @THANHTIEN);

END;

----------------------Vũ----------------------------------------

-----------------------THÊM NHÀ CUNG CẤP---------------------------------Vũ--------------------------

CREATE PROCEDURE ThemNhaCungCap

@MANCC CHAR(10),

@TENNCC NVARCHAR(50),

@DIACHI NVARCHAR(50)

AS

BEGIN

-- Kiểm tra xem mã nhà cung cấp đã tồn tại chưa

IF EXISTS (SELECT 1 FROM NHACCTHUOC WHERE MANCC = @MANCC)

BEGIN

PRINT 'Mã nhà cung cấp đã tồn tại.';

RETURN;

END

-- Thêm nhà cung cấp mới

INSERT INTO NHACCTHUOC (MANCC, TENNCC, DIACHI)

VALUES (@MANCC, @TENNCC, @DIACHI);

END;

----------------------Vũ----------------------------------------

---------------------- THÊM CHI TIẾT PN-----------------------Vũ----------------------------

CREATE PROCEDURE ThemChiTietPhieuNhapThuoc

@MAPN CHAR(10),

@MATHUOC CHAR(10),

@SL INT,

@GIA FLOAT

AS

BEGIN

-- Kiểm tra xem phiếu nhập có tồn tại hay không

IF NOT EXISTS (SELECT 1 FROM PHIEU\_NHAP\_THUOC WHERE MAPN = @MAPN)

BEGIN

PRINT 'Phiếu nhập không tồn tại.';

RETURN;

END

-- Kiểm tra xem thuốc có tồn tại hay không

IF NOT EXISTS (SELECT 1 FROM THUOC WHERE MATHUOC = @MATHUOC)

BEGIN

PRINT 'Thuốc không tồn tại.';

RETURN;

END

-- Thêm chi tiết phiếu nhập thuốc

INSERT INTO CHITIET\_PN\_THUOC (MaPN, MaThuoc, SoLuong, Gia)

VALUES (@MAPN, @MATHUOC, @SL, @GIA);

END;

----------------------Vũ----------------------------------------

----------------------THÊM THUỐC MỚI-----------------Vũ----------------------

CREATE PROCEDURE ThemThuoc

@MATHUOC CHAR(10),

@TENTHUOC NVARCHAR(50),

@THANHPHAN NVARCHAR(50),

@GIA FLOAT,

@DONVI NVARCHAR(20)

AS

BEGIN

-- Kiểm tra xem mã thuốc đã tồn tại chưa

IF EXISTS (SELECT 1 FROM THUOC WHERE MATHUOC = @MATHUOC)

BEGIN

PRINT 'Mã thuốc đã tồn tại.';

RETURN;

END

-- Thêm thuốc mới

INSERT INTO THUOC (MATHUOC, TENTHUOC, THANHPHAN, GIA)

VALUES (@MATHUOC, @TENTHUOC, @THANHPHAN, @GIA);

INSERT INTO LUUKHO VALUES(@MATHUOC,@TENTHUOC,0,@DONVI,GETDATE(),DATEADD(YEAR,3,GETDATE()))

END;

**2.2.2.4 Bảo Duy**

A screenshot of a computer code

Description automatically generatedA white screen with black text

Description automatically generated

A white background with black text

Description automatically generated

A screenshot of a computer

Description automatically generatedA white background with text

Description automatically generatedA screenshot of a computer code

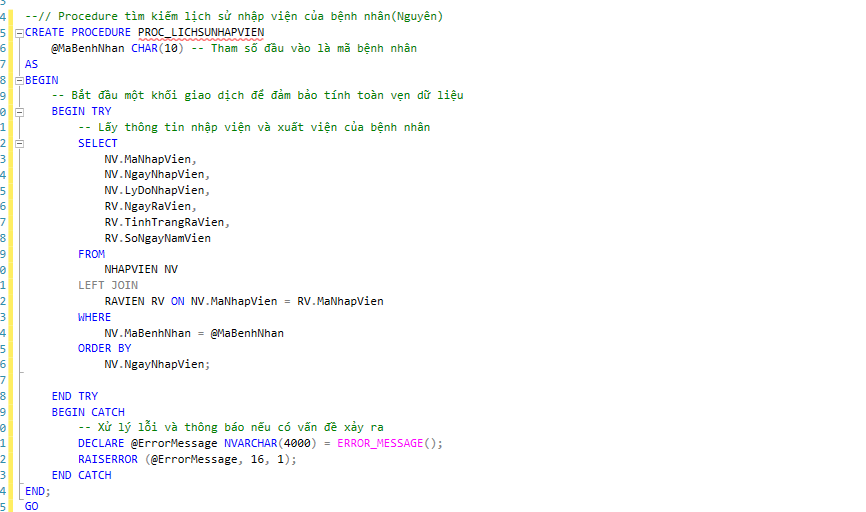
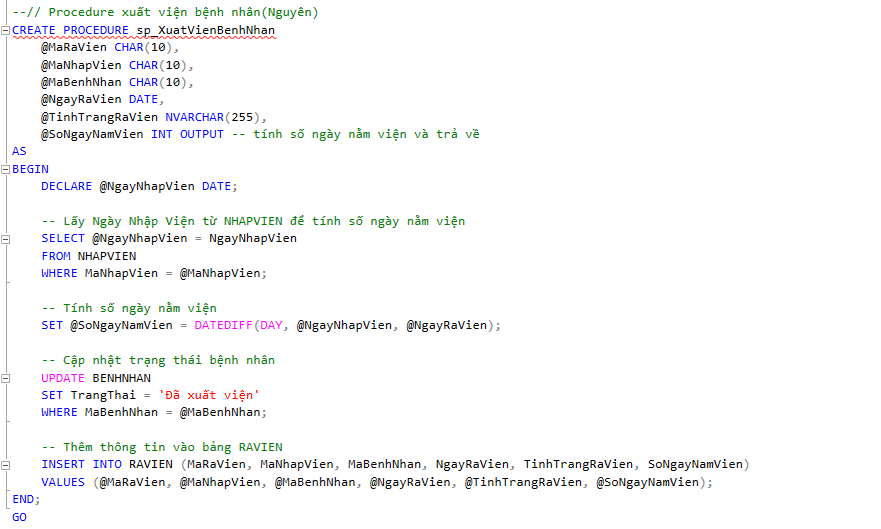
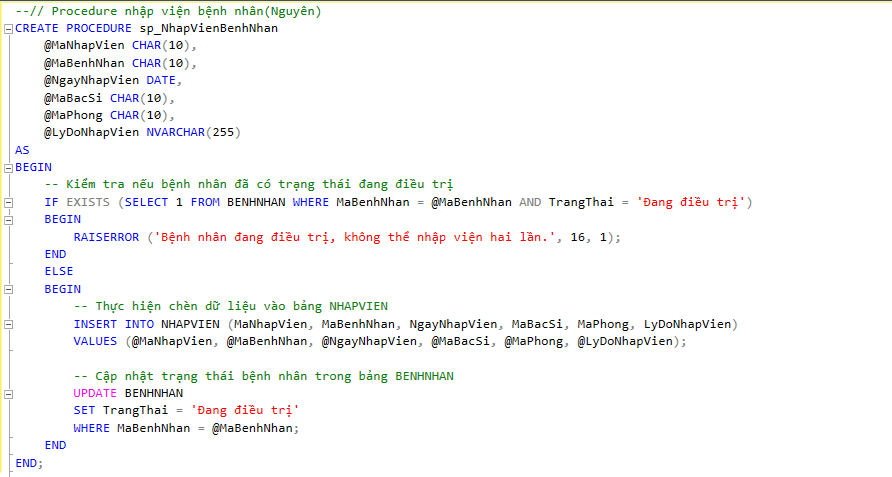
Description automatically generated

A close up of text

Description automatically generatedA white background with black and pink text

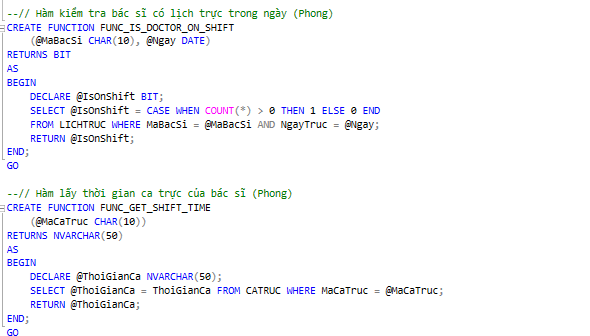
Description automatically generated

**2.2.2.5 Nguyên**



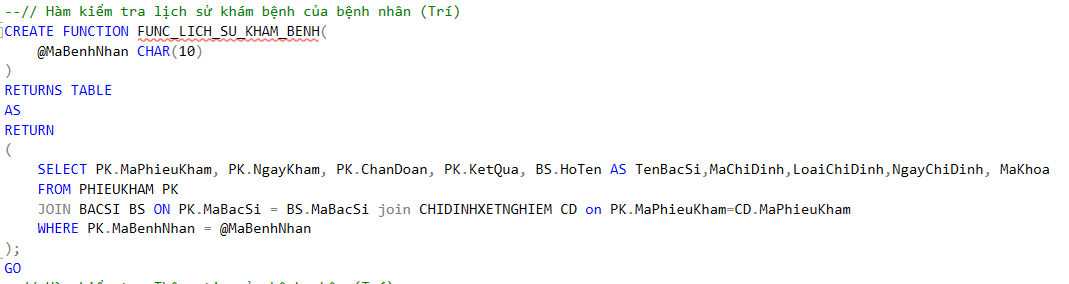
### 2.2.3 Câu lệnh Function

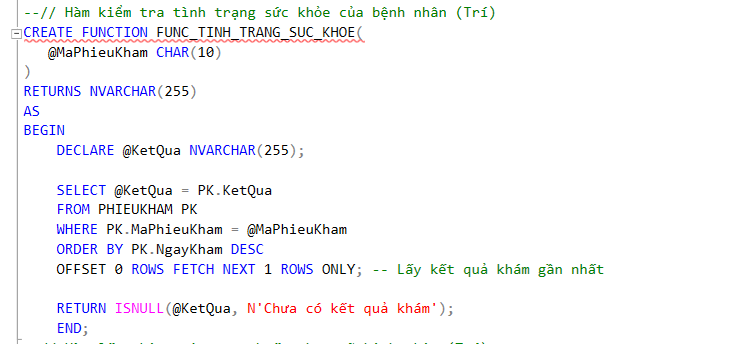
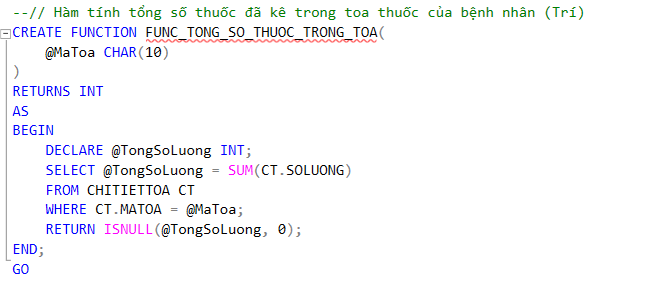
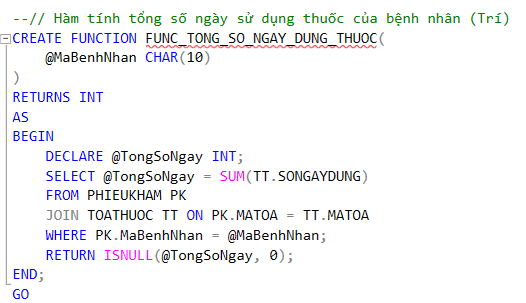
**2.2.3.1 Phong**





**2.2.3.2 Trí**





**2.2.3.3 Vũ**

----------------Vũ-------------------------

--------------------FUNCTION---------------------------Vũ--------------

CREATE FUNCTION ThongKePhieuNhapTheoThang

(

@Thang INT,

@Nam INT

)

RETURNS @KetQua TABLE

(

MAPN CHAR(10),

MANCC CHAR(10),

NGAYNHAP DATE,

THANHTIEN FLOAT

)

AS

BEGIN

INSERT INTO @KetQua

SELECT MAPN, MANCC, NGAYNHAP, THANHTIEN

FROM PHIEU\_NHAP\_THUOC

WHERE MONTH(NGAYNHAP) = @Thang AND YEAR(NGAYNHAP) = @Nam;

RETURN;

END;

----------------Vũ-------------------------

----------------------FUNCTION XEM CHI TIẾT PH THEO MAPN-----Vũ------

CREATE FUNCTION XemChiTietPhieuNhap

(

@MAPN CHAR(10)

)

RETURNS @KetQua TABLE

(

MATHUOC CHAR(10),

SL INT,

GIA FLOAT

)

AS

BEGIN

INSERT INTO @KetQua

SELECT MATHUOC, SoLuong, GIA

FROM CHITIET\_PN\_THUOC

WHERE MAPN = @MAPN;

RETURN;

END;

----------------Vũ-------------------------

--------------------------KIỂM TRA THÔNG TIN THUỐC THEO MATHUOC----Vũ----

CREATE FUNCTION KiemTraThongTinThuoc

(

@MATHUOC CHAR(10)

)

RETURNS @KetQua TABLE

(

TENTHUOC NVARCHAR(50),

THANHPHAN NVARCHAR(50),

GIA FLOAT

)

AS

BEGIN

INSERT INTO @KetQua

SELECT TENTHUOC, THANHPHAN, GIA

FROM THUOC

WHERE MATHUOC = @MATHUOC;

RETURN;

END;

**2.2.3.4 Bảo Duy**

A white background with black text

Description automatically generatedA white background with text

Description automatically generatedA white background with text

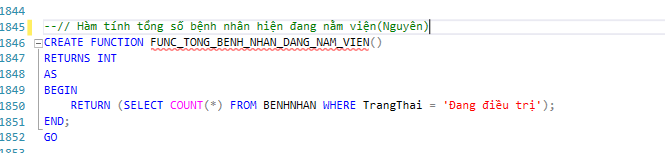
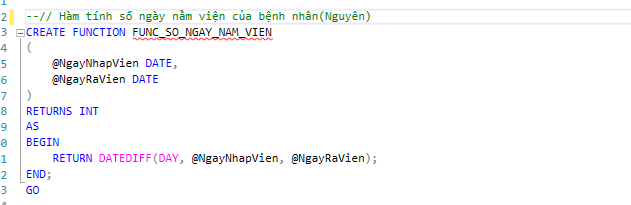
Description automatically generated

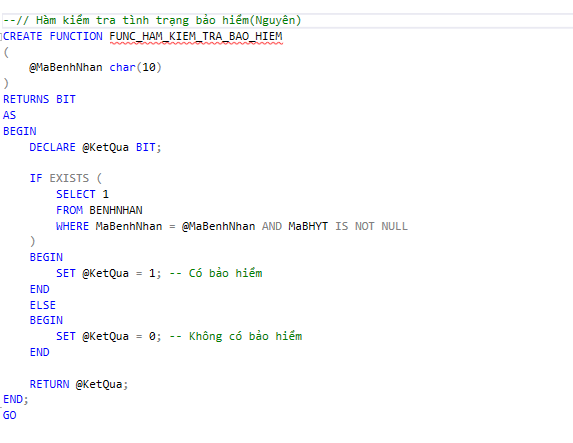
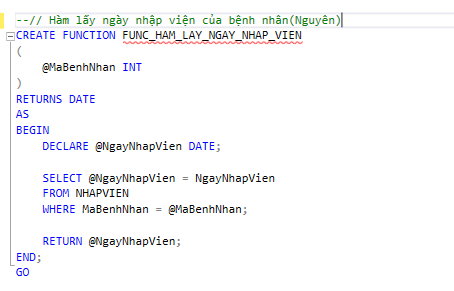
A white background with black and green text

Description automatically generatedA white background with black text

Description automatically generated

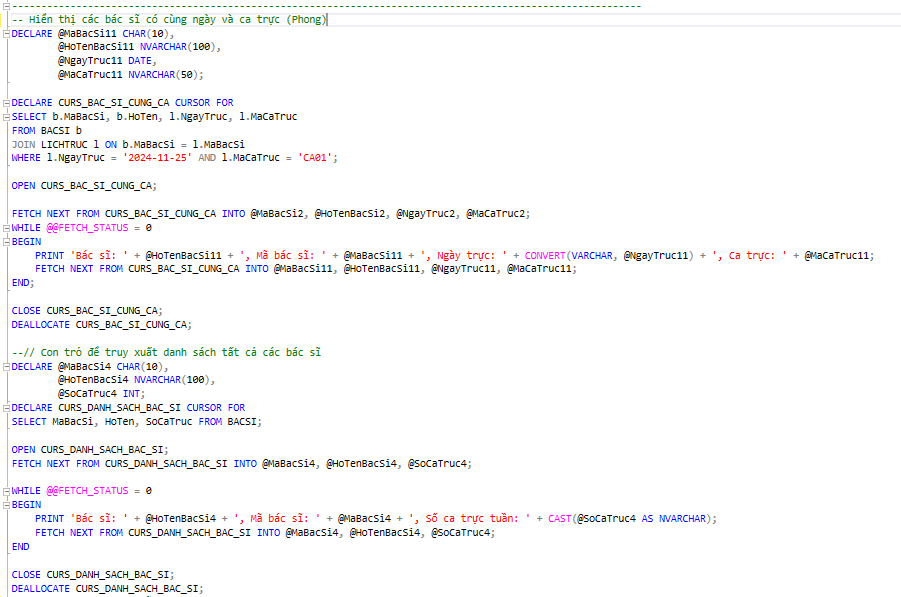
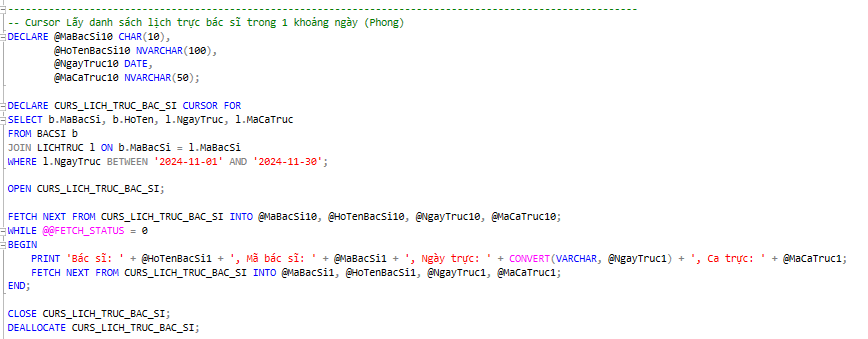
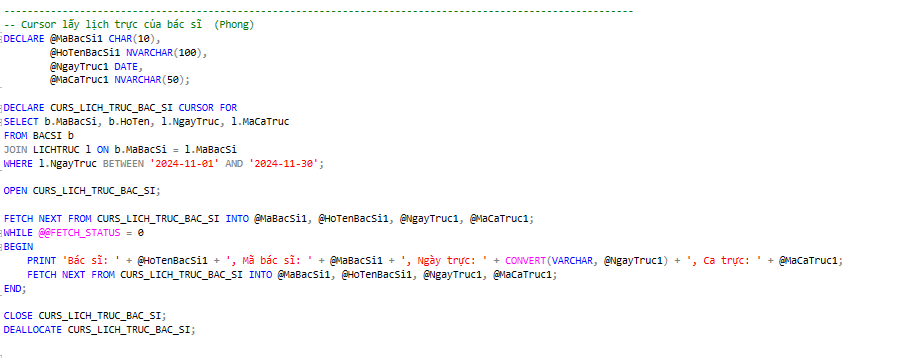
**2.2.3.5 Nguyên**





### 2.2.4 Câu lệnh Cursor

**2.2.4.1 Phong**



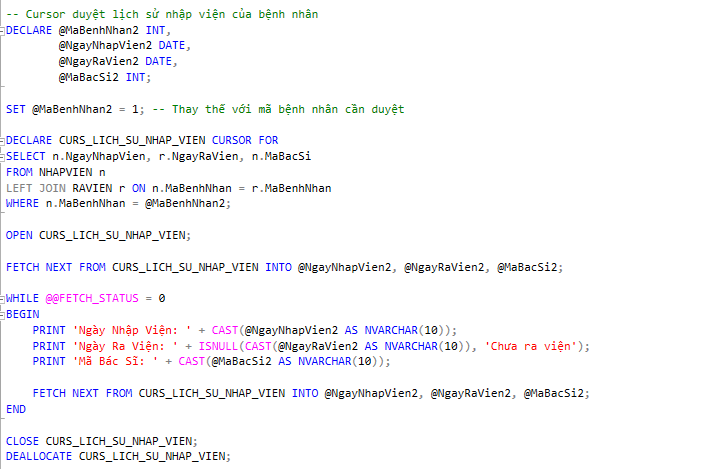
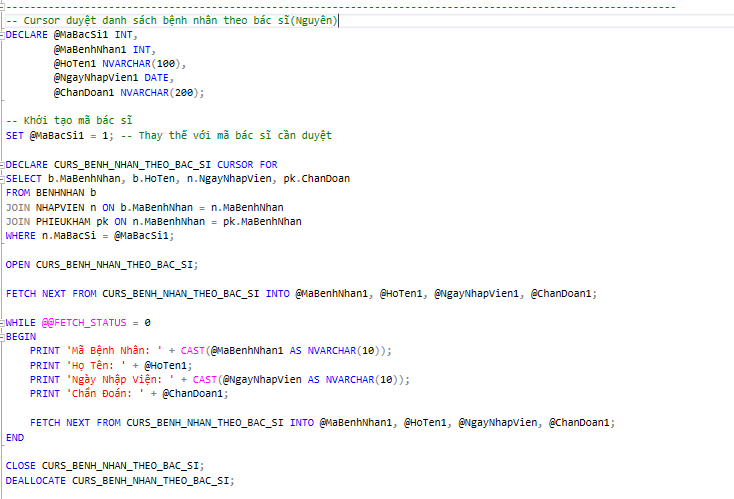
**2.2.4.2 Bảo Duy**

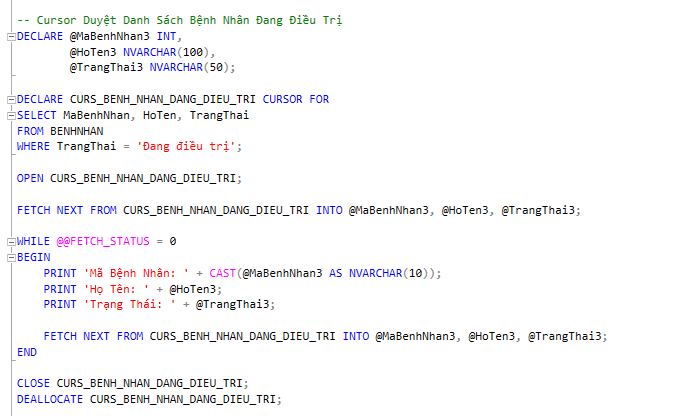
A computer screen shot of text

Description automatically generatedA white background with black and green text

Description automatically generated

**2.2.4.3 Nguyên**





### 2.2.5 Câu lệnh phân quyền

CREATE ROLE DIEUDUONG\_ROLE;

CREATE ROLE BACSI\_ROLE;

CREATE ROLE THUNGAN\_ROLE;

--ĐIỀU DƯỠNG

--XÁC ĐỊNH NHU CẦU PHÂN QUYỀN

GRANT SELECT, INSERT ON BENHNHAN TO DIEUDUONG\_ROLE;

GRANT SELECT, INSERT ON TOATHUOC TO DIEUDUONG\_ROLE;

GRANT SELECT, INSERT ON CHITIETTOATHUOC TO DIEUDUONG\_ROLE;

GRANT SELECT , UPDATE ON NHAPVIEN TO DIEUDUONG\_ROLE;

GRANT SELECT , UPDATE ON RAVIEN TO DIEUDUONG\_ROLE;

GRANT SELECT, INSERT ON HOSOBENHNHAN TO DIEUDUONG\_ROLE;

GRANT INSERT ,UPDATE ON PHIEUKHAM TO DIEUDUONG\_ROLE;

--TẠO LOGIN CHO CÁC ĐIỀU DƯỠNG

CREATE LOGIN [NGUYENVANA] WITH PASSWORD = 'A123';

CREATE LOGIN [NGUYENVANB] WITH PASSWORD = 'B124';

CREATE LOGIN [NGUYENVANC] WITH PASSWORD = 'C125';

CREATE LOGIN [NGUYENVAND] WITH PASSWORD = 'D126';

-- TẠO TÀI KHOẢN CHO CÁC ĐIỀU DƯỠNG

CREATE USER [NGUYENVANA123] FOR LOGIN [NGUYENVANA] ;

CREATE USER [NGUYENVANB123] FOR LOGIN [NGUYENVANB] ;

CREATE USER [NGUYENVANC123] FOR LOGIN [NGUYENVANC] ;

CREATE USER [NGUYENVAND123] FOR LOGIN [NGUYENVAND] ;

--GÁN ROLE CHO ĐIỀU DƯỠNG

EXEC sp\_addrolemember 'DIEUDUONG\_ROLE','NGUYENVANA123';

EXEC sp\_addrolemember 'DIEUDUONG\_ROLE','NGUYENVANB123';

EXEC sp\_addrolemember 'DIEUDUONG\_ROLE','NGUYENVANC123';

EXEC sp\_addrolemember 'DIEUDUONG\_ROLE','NGUYENVAND123';

--CẤP QUYỀN KẾT NỐI CHO ĐIỀU DƯỠNG

GRANT CONNECT TO [NGUYENVANA123];

GRANT CONNECT TO [NGUYENVANB123];

GRANT CONNECT TO [NGUYENVANC123];

GRANT CONNECT TO [NGUYENVAND123];

--//////

---BÁC SĨ

--XÁC ĐỊNH NHU CẦU PHÂN QUYỀN

GRANT SELECT, INSERT ,UPDATE ON BENHNHAN TO BACSI\_ROLE;

GRANT SELECT, INSERT, UPDATE ON TOATHUOC TO BACSI\_ROLE;

GRANT SELECT, INSERT, UPDATE ON CHITIETTOATHUOC TO BACSI\_ROLE;

GRANT SELECT , UPDATE ON NHAPVIEN TO BACSI\_ROLE;

GRANT SELECT , UPDATE ON RAVIEN TO BACSI\_ROLE;

GRANT SELECT, INSERT , UPDATE ON HOSOBENHNHAN TO BACSI\_ROLE;

GRANT INSERT ON PHIEUKHAM TO BACSI\_ROLE;

GRANT SELECT, INSERT ON CHIDINHXETNGHIEM TO BACSI\_ROLE;

--TẠO LOGIN CHO CÁC BÁC SĨ

CREATE LOGIN [TRANVANA] WITH PASSWORD = 'A1234';

CREATE LOGIN [TRANVANB] WITH PASSWORD = 'B1244';

CREATE LOGIN [TRANVANC] WITH PASSWORD = 'C1254';

CREATE LOGIN [TRANVAND] WITH PASSWORD = 'D1264';

-- TẠO TÀI KHOẢN CHO CÁC BÁC SĨ

CREATE USER [TRANVANA] FOR LOGIN [TRANVANA] ;

CREATE USER [TRANVANB] FOR LOGIN [TRANVANB] ;

CREATE USER [TRANVANC] FOR LOGIN [TRANVANC] ;

CREATE USER [TRANVAND] FOR LOGIN [TRANVAND] ;

--GÁN ROLE CHO BÁC SĨ

EXEC sp\_addrolemember 'BACSI\_ROLE','TRANVANA';

EXEC sp\_addrolemember 'BACSI\_ROLE','TRANVANB';

EXEC sp\_addrolemember 'BACSI\_ROLE','TRANVANC';

EXEC sp\_addrolemember 'BACSI\_ROLE','TRANVAND';

--CẤP QUYỀN KẾT NỐI CHO BÁC SĨ

GRANT CONNECT TO [TRANVANA];

GRANT CONNECT TO [TRANVANB];

GRANT CONNECT TO [TRANVANC];

GRANT CONNECT TO [TRANVAND];

--//////

---THU NGÂN

--XÁC ĐỊNH NHU CẦU PHÂN QUYỀN

GRANT SELECT, INSERT ,UPDATE ON THANHTOAN TO THUNGAN\_ROLE;

--TẠO LOGIN CHO CÁC THU NGÂN

CREATE LOGIN [TRANVANE] WITH PASSWORD = 'E1234';

CREATE LOGIN [TRANVANF] WITH PASSWORD = 'F1244';

CREATE LOGIN [TRANVANG] WITH PASSWORD = 'G1254';

CREATE LOGIN [TRANVANH] WITH PASSWORD = 'H1264';

-- TẠO TÀI KHOẢN CHO CÁC THU NGÂN

CREATE USER [TRANVANE] FOR LOGIN [TRANVANE] ;

CREATE USER [TRANVANF] FOR LOGIN [TRANVANF] ;

CREATE USER [TRANVANG] FOR LOGIN [TRANVANG] ;

CREATE USER [TRANVANH] FOR LOGIN [TRANVANH] ;

--GÁN ROLE CHO THU NGÂN

EXEC sp\_addrolemember 'THUNGAN\_ROLE','TRANVANE';

EXEC sp\_addrolemember 'THUNGAN\_ROLE','TRANVANF';

EXEC sp\_addrolemember 'THUNGAN\_ROLE','TRANVANG';

EXEC sp\_addrolemember 'THUNGAN\_ROLE','TRANVANH';

--CẤP QUYỀN KẾT NỐI CHO THU NGÂN

GRANT CONNECT TO [TRANVANE];

GRANT CONNECT TO [TRANVANF];

GRANT CONNECT TO [TRANVANG];

GRANT CONNECT TO [TRANVANH];

### 2.2.6 Sao lưu cơ sở dữ liệu

--full backuP

BACKUP DATABASE QLBV

TO DISK = 'D:\HE\_QUAN\_TRI\_CSDL\TMT\TMT\TMT\QLBV\_FULL.bak'

WITH INIT;

--DIFFERENTIAL backuP

BACKUP DATABASE QLBV

TO DISK = 'D:\HE\_QUAN\_TRI\_CSDL\TMT\TMT\TMT\QLBV\_DIFF.bak'

WITH DIFFERENTIAL , INIT;

--TRANSACTION LOG backuP

BACKUP LOG QLBV

TO DISK = 'D:\HE\_QUAN\_TRI\_CSDL\TMT\TMT\TMT\QLBV\_LOG.bak'

WITH INIT;

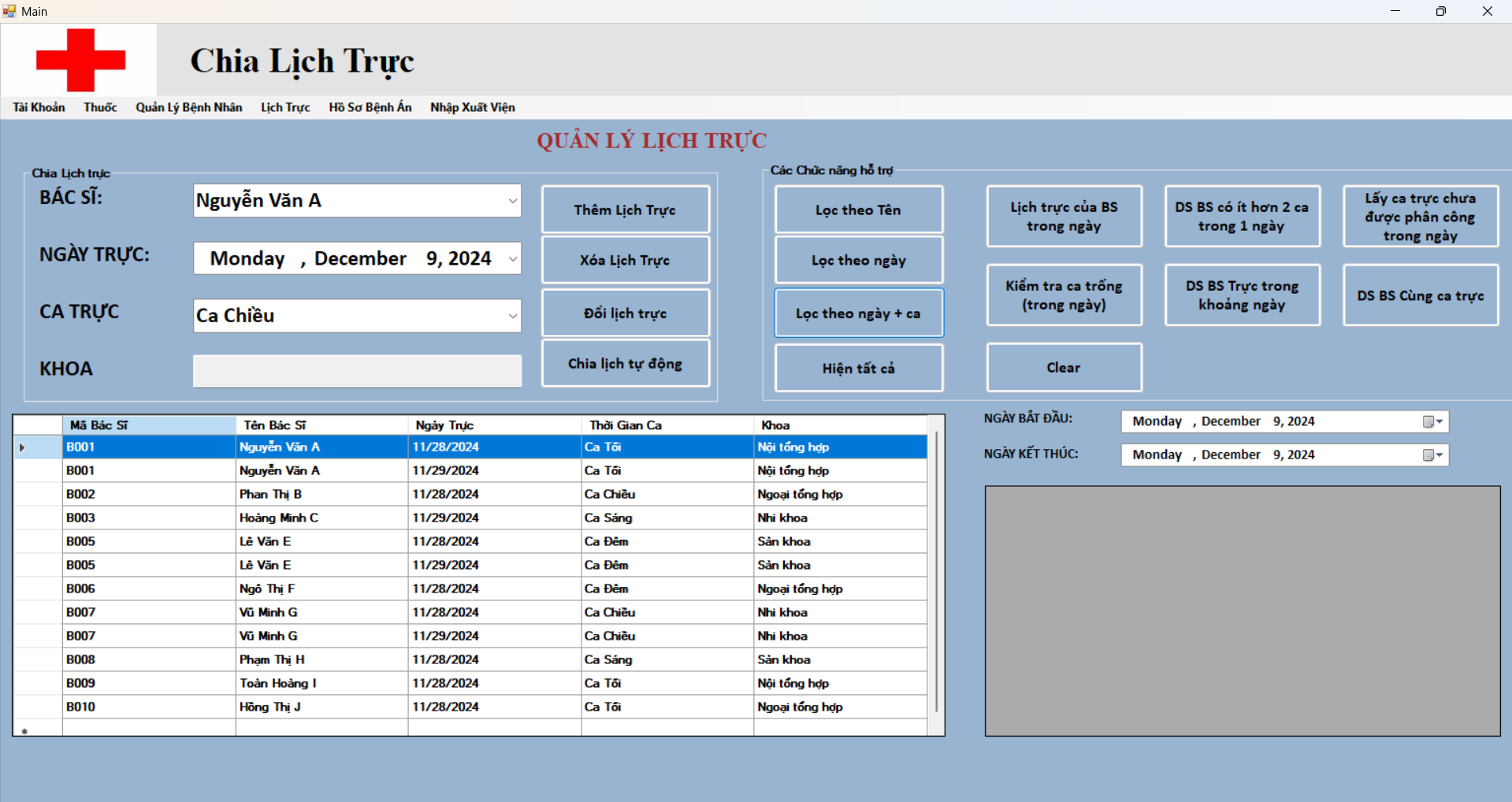
### 2.3. Quản trị người dùng:

- Mô tả giải pháp quản trị người dùng của nhóm, chọn lựa cơ chế xác thực. Tạo các tài khoản login, user, nhóm quyền role, cấp, hủy quyền người dùng.

- Lập lịch trình sao lưu định kỳ cho cơ sở dữ liệu và thiết lập sao lưu tự động.

1. Xây dựng ứng dụng

## 3.1 Nghiệp vụ chia lịch trực cho bác sĩ



HÌnh 2 Giao diện của chia lịch trực

Thêm dữ liệu

// Thêm lịch trực

public bool ThemLichTruc(LICH\_TRUC\_DTO lichTruc)

{

using (SqlConnection conn = new SqlConnection(dbContext.strcon))

{

conn.Open();

using (SqlCommand cmd = new SqlCommand("PROC\_THEMLICHTRUC", conn))

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@MaBacSi", lichTruc.MaBacSi);

cmd.Parameters.AddWithValue("@NgayTruc", lichTruc.NgayTruc);

cmd.Parameters.AddWithValue("@MaCaTruc", lichTruc.MaCaTruc);

try

{

cmd.ExecuteNonQuery();

return true; // Chèn thành công

}

catch (SqlException ex)

{

throw new Exception("Lỗi SQL: " + ex.Message, ex);

}

}

}

}

public void PhanLichTrucTuDong(DateTime ngayTruc)

{

using (SqlConnection conn = new SqlConnection(dbContext.strcon))

{

conn.Open();

using (SqlCommand cmd = new SqlCommand("PROC\_PHANLICHTRUCTUDONG", conn))

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@NgayTruc", ngayTruc);

try

{

cmd.ExecuteNonQuery();

}

catch (SqlException ex)

{

throw new Exception("Lỗi SQL: " + ex.Message, ex);

}

}

}

}

public DataTable KiemTraCaTrucTrongTrongNgay(DateTime ngayTruc)

{

using (SqlConnection conn = new SqlConnection(dbContext.strcon))

{

conn.Open();

using (SqlCommand cmd = new SqlCommand("PROC\_KIEMTRACATRUCTRONGTRONGNGAY", conn))

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@NgayTruc", ngayTruc);

using (SqlDataAdapter da = new SqlDataAdapter(cmd))

{

DataTable dt = new DataTable();

da.Fill(dt);

return dt;

}

}

}

}

// Gọi function FUNC\_LAYLICHTRUCCUABACSI

public DataTable LayLichTrucCuaBacSi(string maBacSi, DateTime ngayTruc)

{

using (SqlConnection conn = new SqlConnection(dbContext.strcon))

{

conn.Open();

using (SqlCommand cmd = new SqlCommand("SELECT \* FROM dbo.FUNC\_LAYLICHTRUCCUABACSI(@MaBacSi, @NgayTruc)", conn))

{

cmd.Parameters.AddWithValue("@MaBacSi", maBacSi);

cmd.Parameters.AddWithValue("@NgayTruc", ngayTruc);

using (SqlDataAdapter da = new SqlDataAdapter(cmd))

{

DataTable dt = new DataTable();

da.Fill(dt);

return dt;

}

}

}

}

// Gọi function FUNC\_LAYDANHSACHBACSI

public DataTable LayDanhSachBacSi(DateTime ngayTruc)

{

using (SqlConnection conn = new SqlConnection(dbContext.strcon))

{

conn.Open();

using (SqlCommand cmd = new SqlCommand("SELECT \* FROM dbo.FUNC\_LAYDANHSACHBACSI(@NgayTruc)", conn))

{

cmd.Parameters.AddWithValue("@NgayTruc", ngayTruc);

using (SqlDataAdapter da = new SqlDataAdapter(cmd))

{

DataTable dt = new DataTable();

da.Fill(dt);

return dt;

}

}

}

}

// Gọi function FUNC\_LAYCATRUCCHUADUOCPHANCONG\_TRONGNGAY

public DataTable LayCaTrucChuaDuocPhanCongTrongNgay(DateTime ngayTruc)

{

using (SqlConnection conn = new SqlConnection(dbContext.strcon))

{

conn.Open();

using (SqlCommand cmd = new SqlCommand("SELECT \* FROM dbo.FUNC\_LAYCATRUCCHUADUOCPHANCONG\_TRONGNGAY(@NgayTruc)", conn))

{

cmd.Parameters.AddWithValue("@NgayTruc", ngayTruc);

using (SqlDataAdapter da = new SqlDataAdapter(cmd))

{

DataTable dt = new DataTable();

da.Fill(dt);

return dt;

}

}

}

}

// Phương thức để lấy danh sách lịch trực bác sĩ trong khoảng ngày

public DataTable LayDanhSachLichTruc(DateTime startDate, DateTime endDate)

{

DataTable dt = new DataTable();

using (SqlConnection conn = new SqlConnection(dbContext.strcon))

{

conn.Open();

string query = @"

SELECT b.MaBacSi, b.HoTen, l.NgayTruc, l.MaCaTruc

FROM BACSI b

JOIN LICHTRUC l ON b.MaBacSi = l.MaBacSi

WHERE l.NgayTruc BETWEEN @StartDate AND @EndDate";

using (SqlCommand cmd = new SqlCommand(query, conn))

{

cmd.Parameters.AddWithValue("@StartDate", startDate);

cmd.Parameters.AddWithValue("@EndDate", endDate);

using (SqlDataAdapter da = new SqlDataAdapter(cmd))

{

da.Fill(dt);

}

}

}

return dt;

}

// Phương thức để lấy bác sĩ cùng ca trực

public DataTable LayBacSiCungCa(DateTime ngayTruc, string maCaTruc)

{

DataTable dt = new DataTable();

using (SqlConnection conn = new SqlConnection(dbContext.strcon))

{

conn.Open();

string query = @"

SELECT b.MaBacSi, b.HoTen, l.NgayTruc, l.MaCaTruc

FROM BACSI b

JOIN LICHTRUC l ON b.MaBacSi = l.MaBacSi

WHERE l.NgayTruc = @NgayTruc AND l.MaCaTruc = @MaCaTruc";

using (SqlCommand cmd = new SqlCommand(query, conn))

{

cmd.Parameters.AddWithValue("@NgayTruc", ngayTruc);

cmd.Parameters.AddWithValue("@MaCaTruc", maCaTruc);

using (SqlDataAdapter da = new SqlDataAdapter(cmd))

{

da.Fill(dt);

}

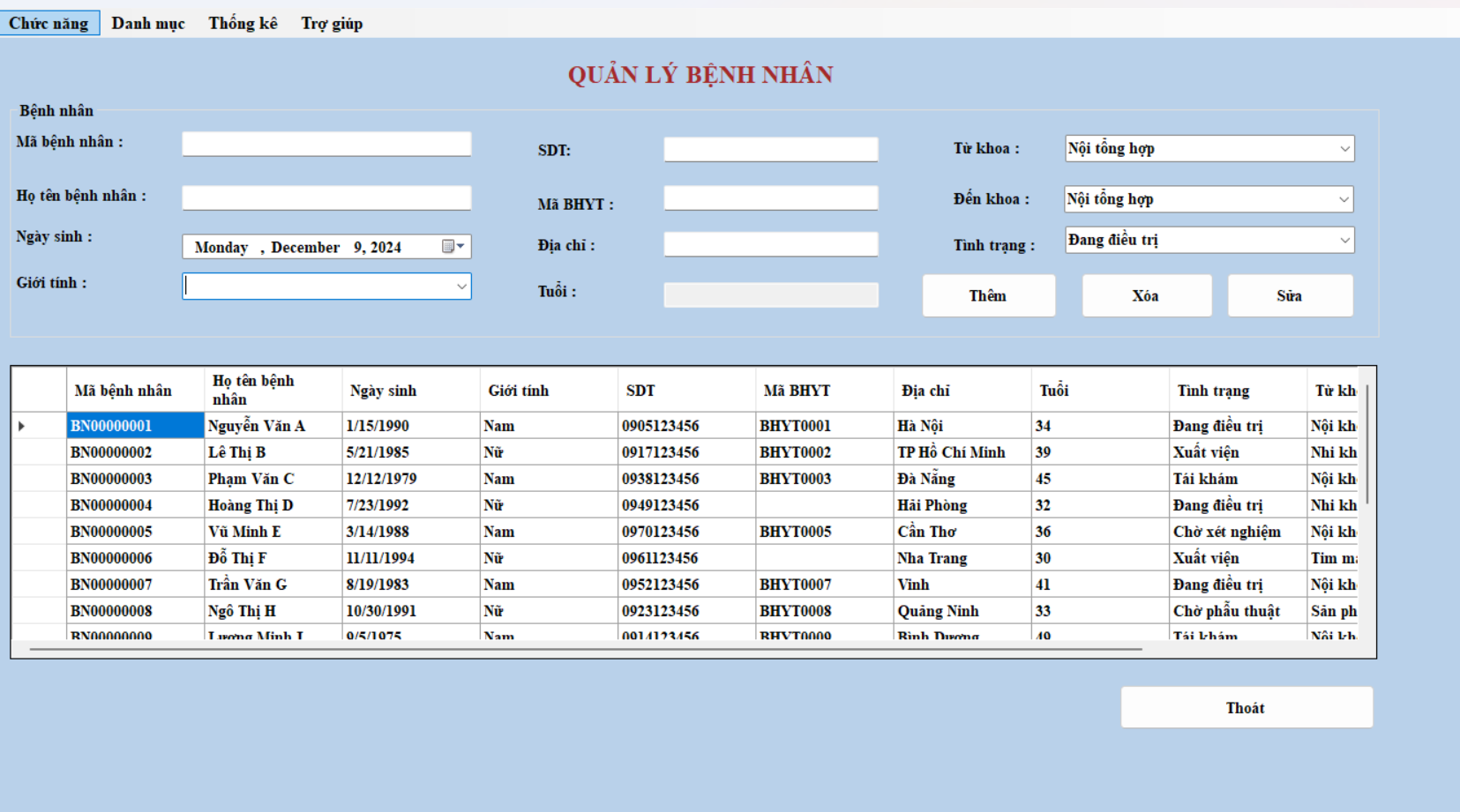
}

}

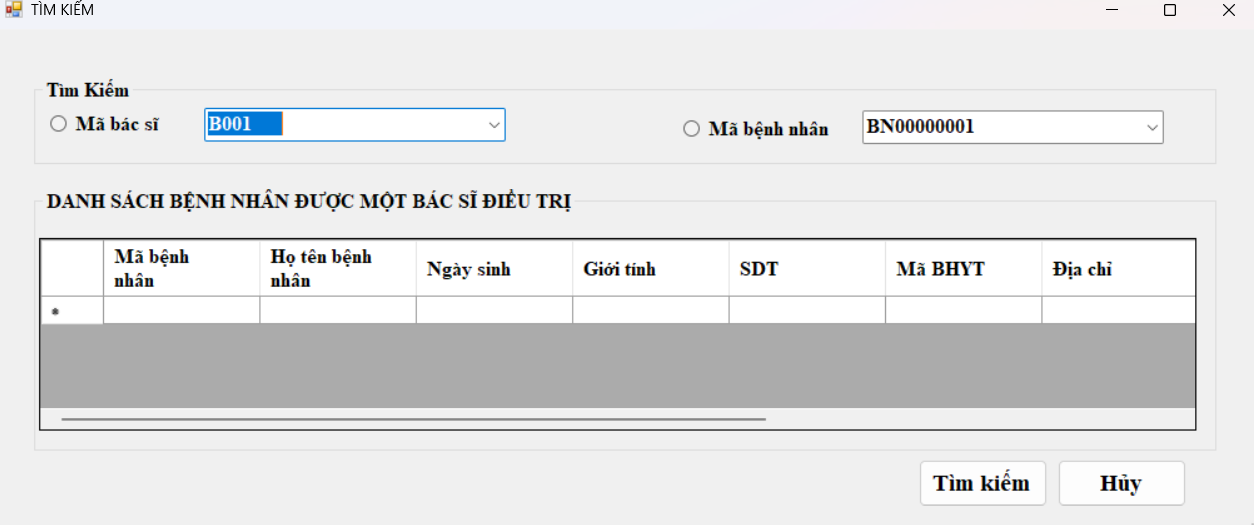
return dt;

}

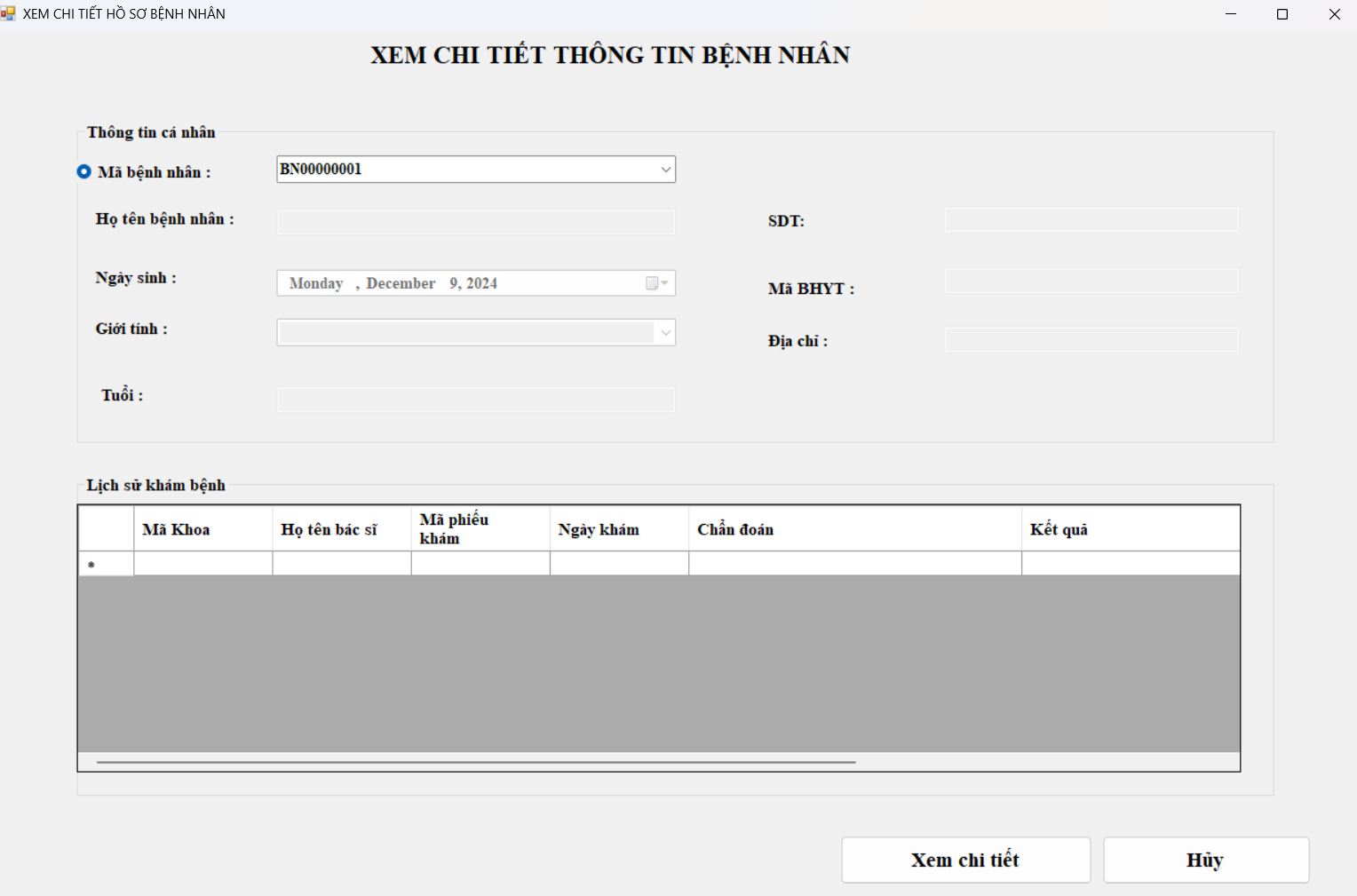
## 3.2 Nghiệp vụ Khám Bệnh



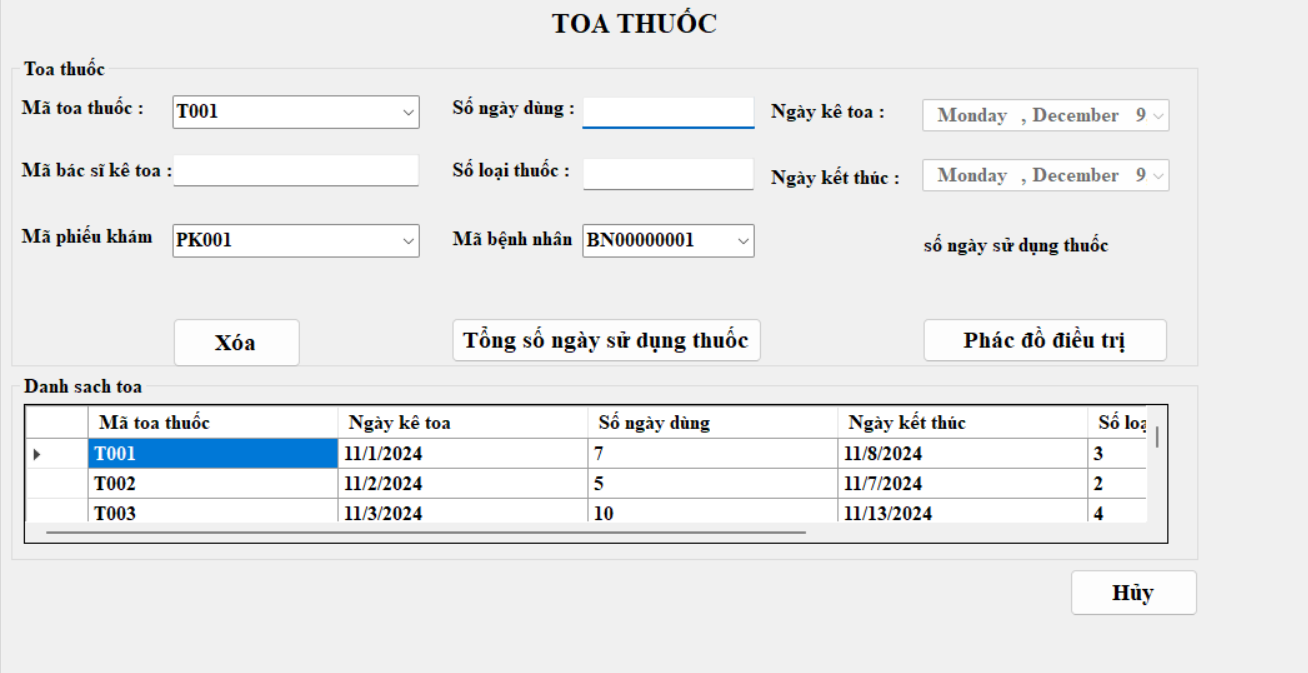
HÌnh 3 Giao diện quản lí bệnh nhân



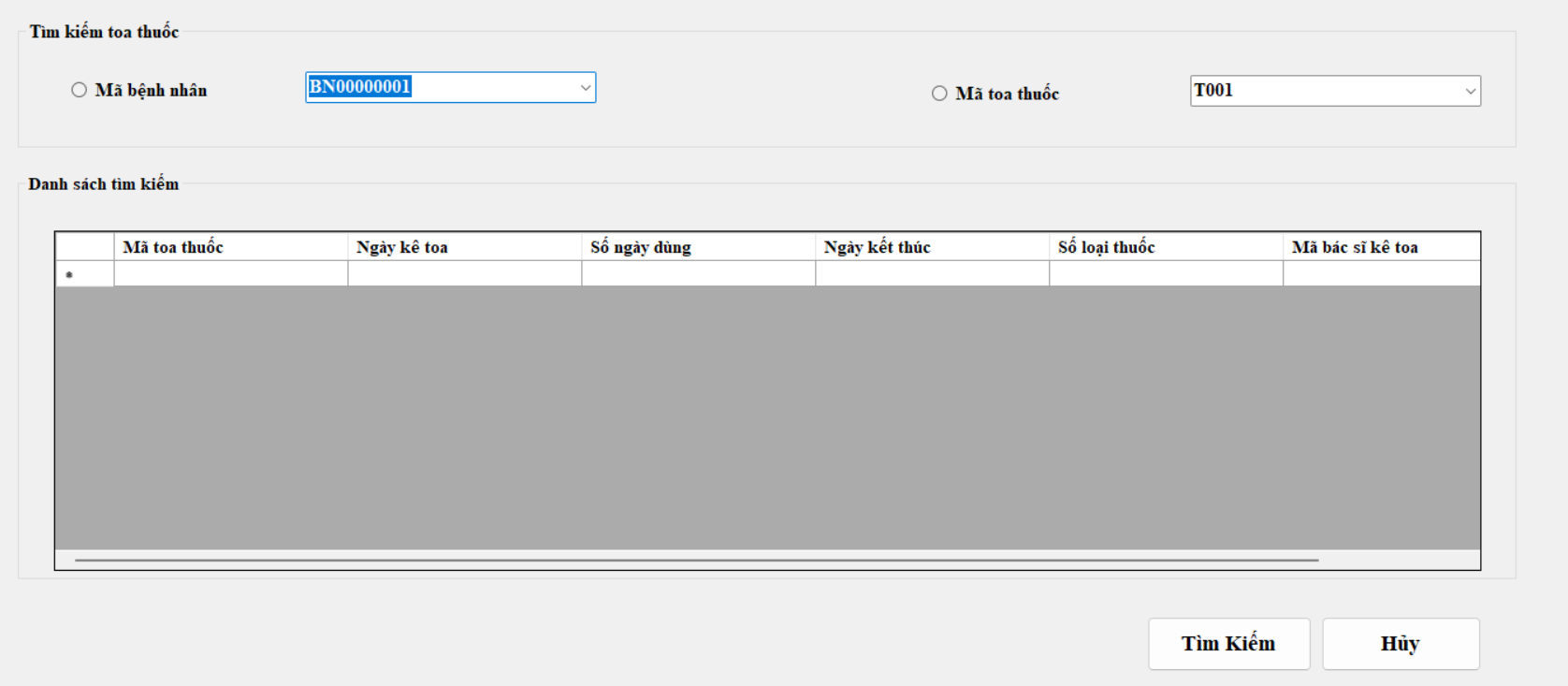
HÌnh 4 Giao diện tìm kiến bệnh nhân

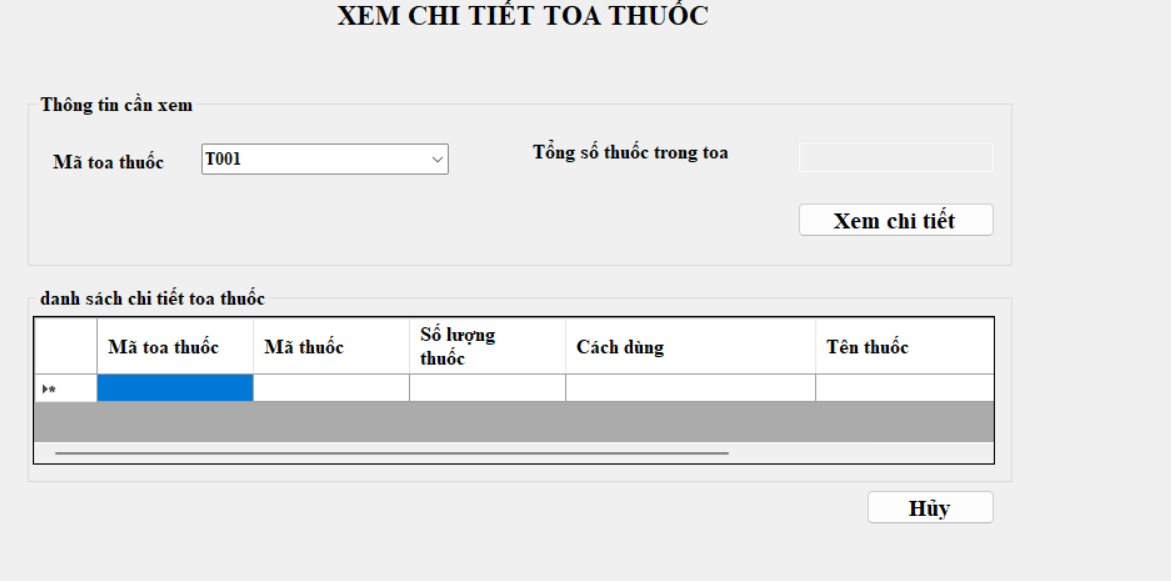


HÌnh 5 Giao diện xem chi tiết thông tin bệnh nhân

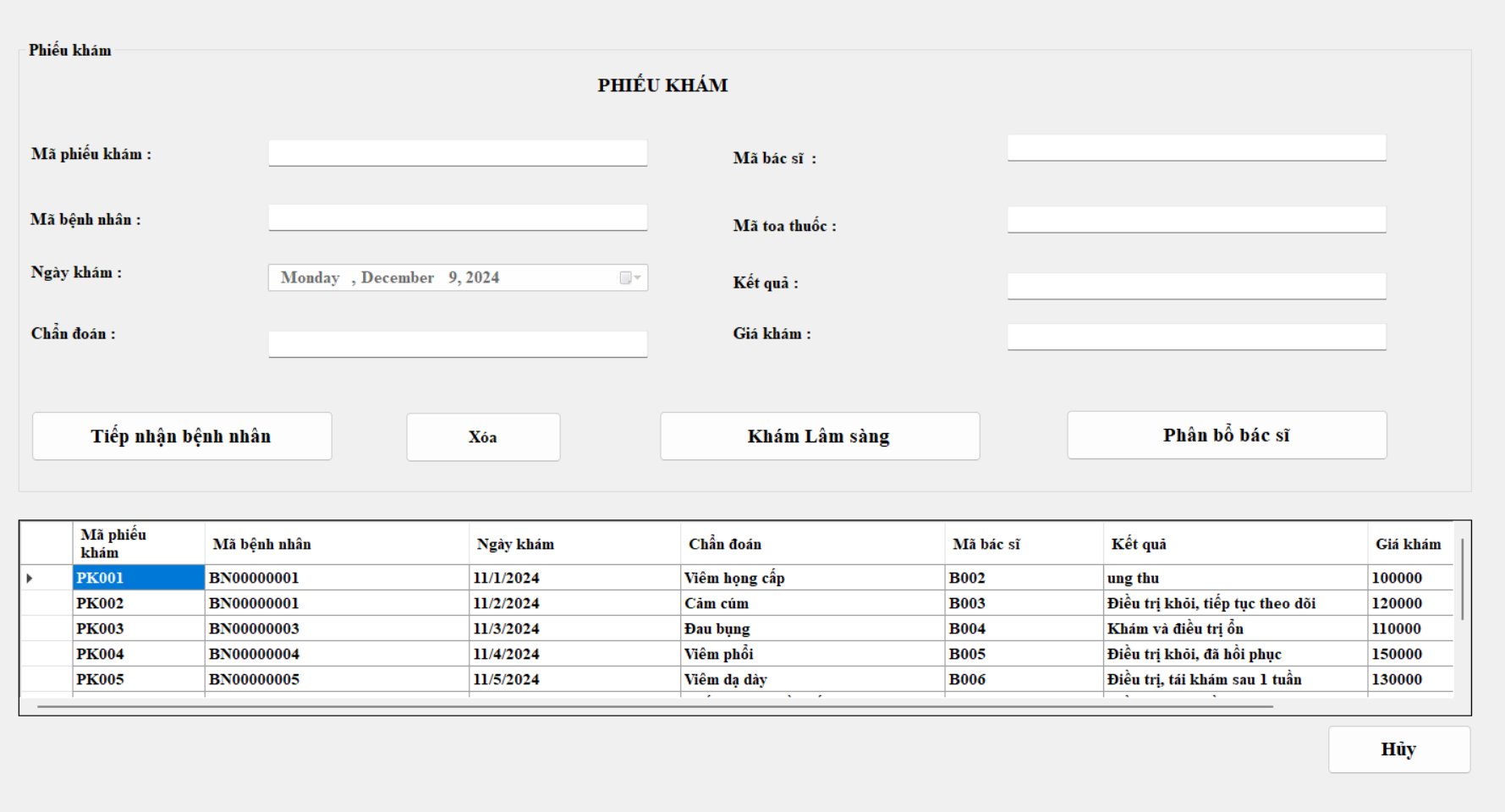


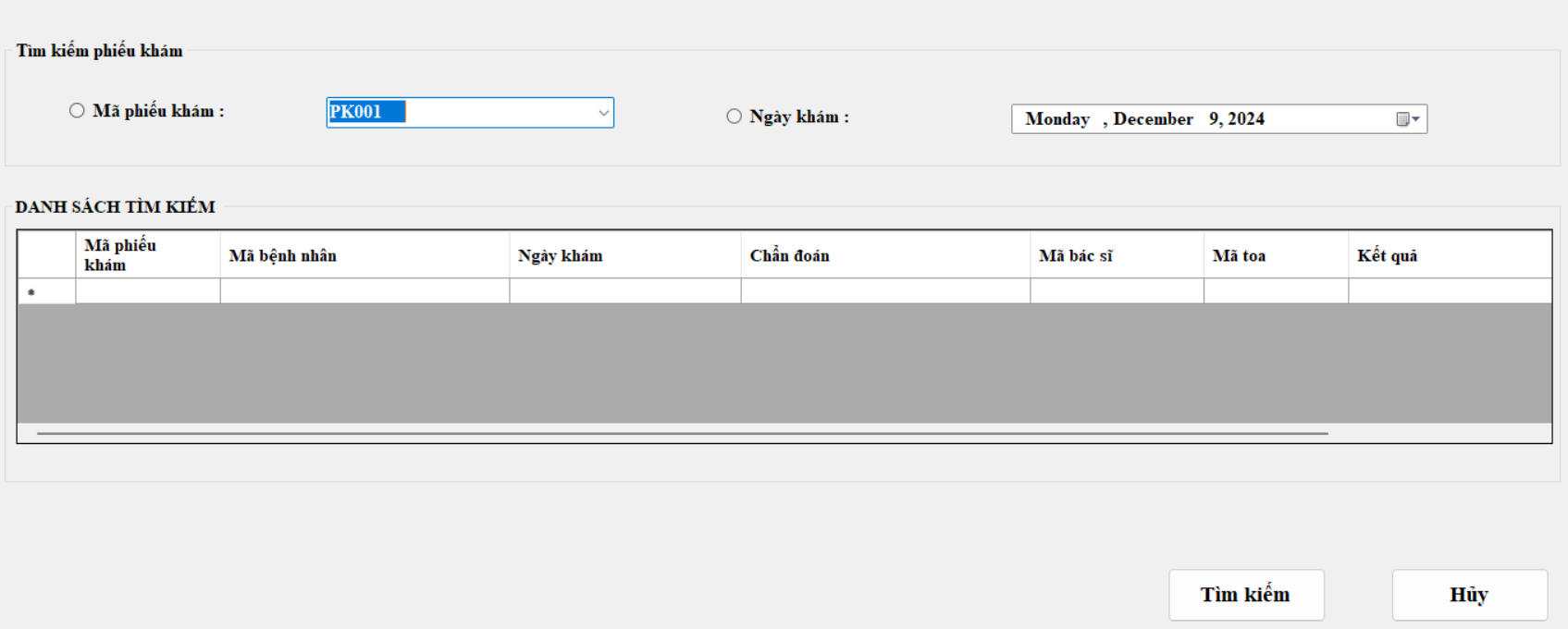
HÌnh 6 Giao diện toa thuốc



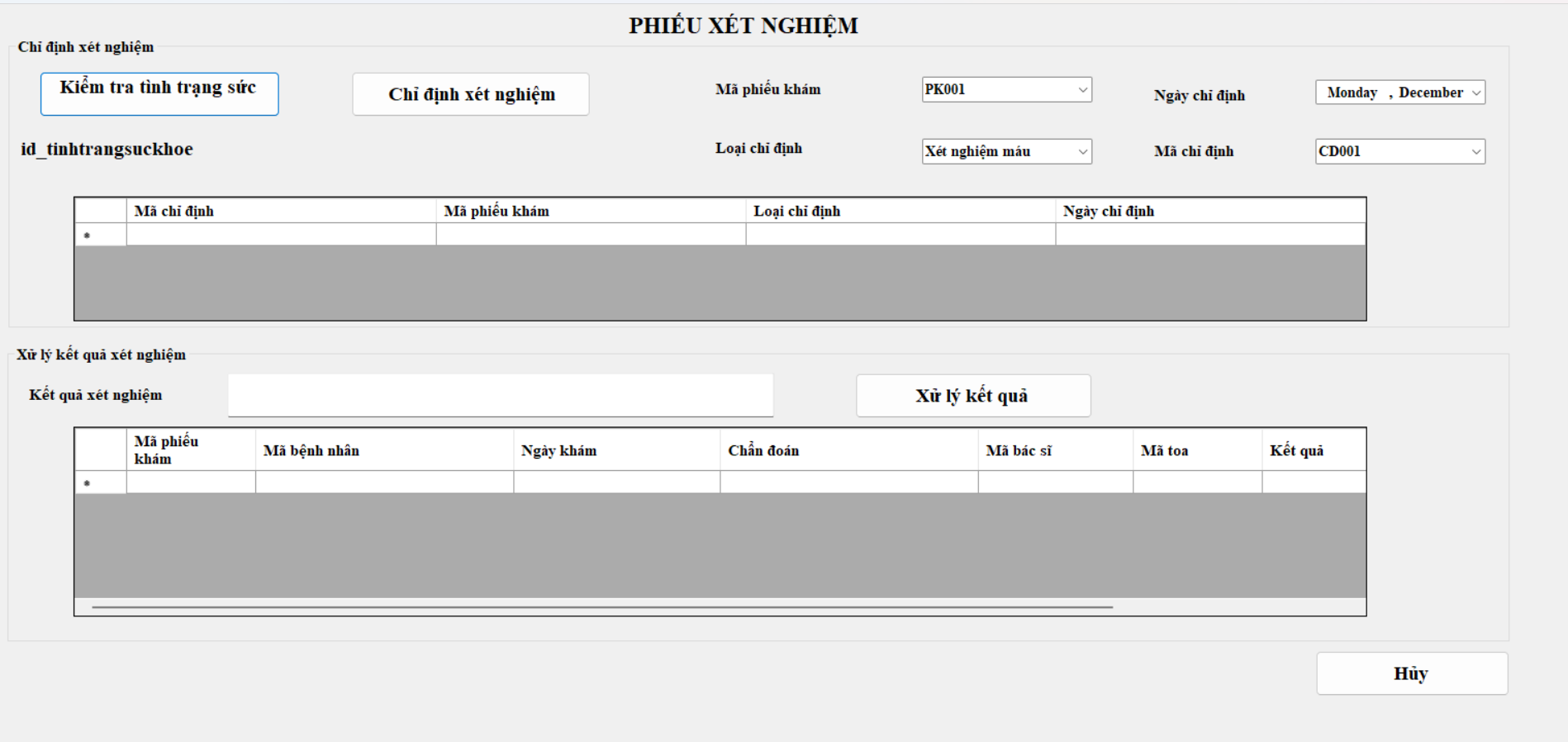
Hình 7 Giao diện tìm kiếm toa thuốc

Hình 8 Giao diện xem chi tiết toa thuốc

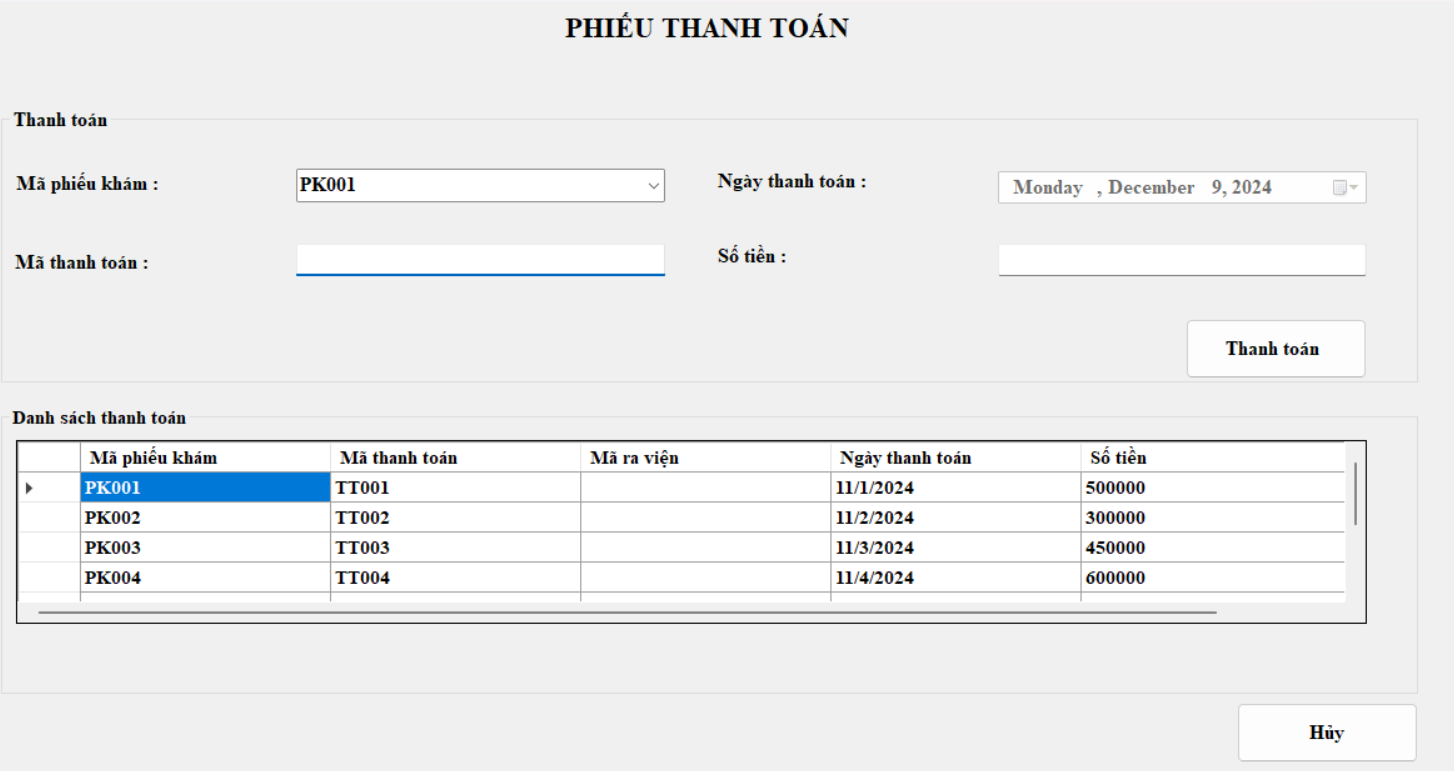


Hình 9 Giao diện phiếu khám

Hình 10 Giao diện tìm kiếm phiếu khám



Hình 11 Giao diện phiếu xét nghiệm



Hình 12 Giao diện phiếu thanh toán

CODE

Thêm bệnh nhân bằng thủ tục

public int thutucthemBN(string maBenhNhan, string hoTen, DateTime ngaySinh, string gioiTinh, string diaChi, string soDienThoai, string maBHYT, string TrangThai ,string tukhoa,string denkhoa)

{

try

{

openconnect();

using (SqlCommand cmd = new SqlCommand("PROC\_THEMBENHNHAN", conn))

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@MaBenhNhan", maBenhNhan);

cmd.Parameters.AddWithValue("@HoTen", hoTen);

cmd.Parameters.AddWithValue("@NgaySinh", ngaySinh);

cmd.Parameters.AddWithValue("@GioiTinh", gioiTinh);

cmd.Parameters.AddWithValue("@DiaChi", diaChi);

cmd.Parameters.AddWithValue("@SoDienThoai", soDienThoai);

cmd.Parameters.AddWithValue("@MaBHYT",maBHYT);

cmd.Parameters.AddWithValue("@TrangThai", TrangThai);

cmd.Parameters.AddWithValue("@Tukhoa",tukhoa);

cmd.Parameters.AddWithValue("@Denkhoa", string.IsNullOrEmpty(denkhoa) ? (object)DBNull.Value : denkhoa);

// Thực thi thủ tục

int kq = cmd.ExecuteNonQuery();

return kq;

}

}

catch (SqlException ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi SQL", MessageBoxButtons.OK, MessageBoxIcon.Error);

return -1;

}

finally

{

closeconnect();

}

}

Xem lịch sử khám

public DataTable Getlichsukhambenh(string maBN)

{

try

{

openconnect();

string query = "SELECT \* FROM FUNC\_LICH\_SU\_KHAM\_BENH(@MaBenhNhan)";

using (SqlCommand cmd = new SqlCommand(query, conn))

{

cmd.CommandType = CommandType.Text;

cmd.Parameters.AddWithValue("@MaBenhNhan", maBN);

// Tạo DataAdapter để lưu kết quả trả về từ câu lệnh

SqlDataAdapter da = new SqlDataAdapter(cmd);

// Tạo DataTable để chứa dữ liệu trả về

DataTable dt = new DataTable();

da.Fill(dt);

return dt;

}

}

catch (SqlException ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi SQL", MessageBoxButtons.OK, MessageBoxIcon.Error);

return null;

}

finally

{

closeconnect();

}

}

Phân bổ bác sĩ

public void PhanBoBacSiKham(string maPhieuKham, string maBacSi)

{

try

{

openconnect();

string query = "PROC\_PHANBOBACSIKHAM";

using (SqlCommand cmd = new SqlCommand(query, conn))

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@MaPhieuKham", maPhieuKham);

cmd.Parameters.AddWithValue("@MaBacSi", maBacSi);

cmd.ExecuteNonQuery();

MessageBox.Show("Bác sĩ đã được phân bổ");

}

}

catch (SqlException ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi SQL", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

finally

{

closeconnect();

}

}

Khám lâm sàn

public void KhamLamSang(string maPhieuKham, string chanDoan, string ketQua)

{

try

{

openconnect();

string query = "PROC\_KHAMLAMSANG";

using (SqlCommand cmd = new SqlCommand(query, conn))

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@MaPhieuKham", maPhieuKham);

cmd.Parameters.AddWithValue("@ChanDoan", chanDoan);

cmd.Parameters.AddWithValue("@KetQua", ketQua);

cmd.ExecuteNonQuery();

MessageBox.Show("Khám lâm sàng hoàn tất");

}

}

catch (SqlException ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi SQL", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

finally

{

closeconnect();

}

}

Xem tình trạng sức khỏe gần nhất

public string GetTinhTrangSucKhoe(string maPhieuKham)

{

try

{

openconnect();

string query = "SELECT dbo.FUNC\_TINH\_TRANG\_SUC\_KHOE(@MaPhieuKham)";

using (SqlCommand cmd = new SqlCommand(query, conn))

{

cmd.CommandType = CommandType.Text;

cmd.Parameters.AddWithValue("@MaPhieuKham", maPhieuKham);

object result = cmd.ExecuteScalar();

return result != null ? result.ToString() : "Chưa có kết quả khám";

}

}

catch (SqlException ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi SQL", MessageBoxButtons.OK, MessageBoxIcon.Error);

return null;

}

finally

{

closeconnect();

}

}

Tính tổng số ngày sử dụng thuốc

public int GetTongSoNgayDungThuoc(string maBenhNhan)

{

try

{

openconnect();

string query = "SELECT dbo.FUNC\_TONG\_SO\_NGAY\_DUNG\_THUOC(@MaBenhNhan) AS TongSoNgayDung";

using (SqlCommand cmd = new SqlCommand(query, conn))

{

cmd.Parameters.AddWithValue("@MaBenhNhan", maBenhNhan);

object result = cmd.ExecuteScalar();

return result != null ? Convert.ToInt32(result) : 0;

}

}

catch (SqlException ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi SQL", MessageBoxButtons.OK, MessageBoxIcon.Error);

return -1;

}

finally

{

closeconnect();

}

}

Tính tổng số thuốc trong toa

public int GetTongSoThuocTrongToa(string maToa)

{

try

{

openconnect();

string query = "SELECT dbo.FUNC\_TONG\_SO\_THUOC\_TRONG\_TOA(@MaToa) AS TongSoThuoc";

using (SqlCommand cmd = new SqlCommand(query, conn))

{

cmd.Parameters.AddWithValue("@MaToa", maToa);

object result = cmd.ExecuteScalar();

return result != null ? Convert.ToInt32(result) : 0;

}

}

catch (SqlException ex)

{

MessageBox.Show("Lỗi: " + ex.Message, "Lỗi SQL", MessageBoxButtons.OK, MessageBoxIcon.Error);

return -1;

}

finally

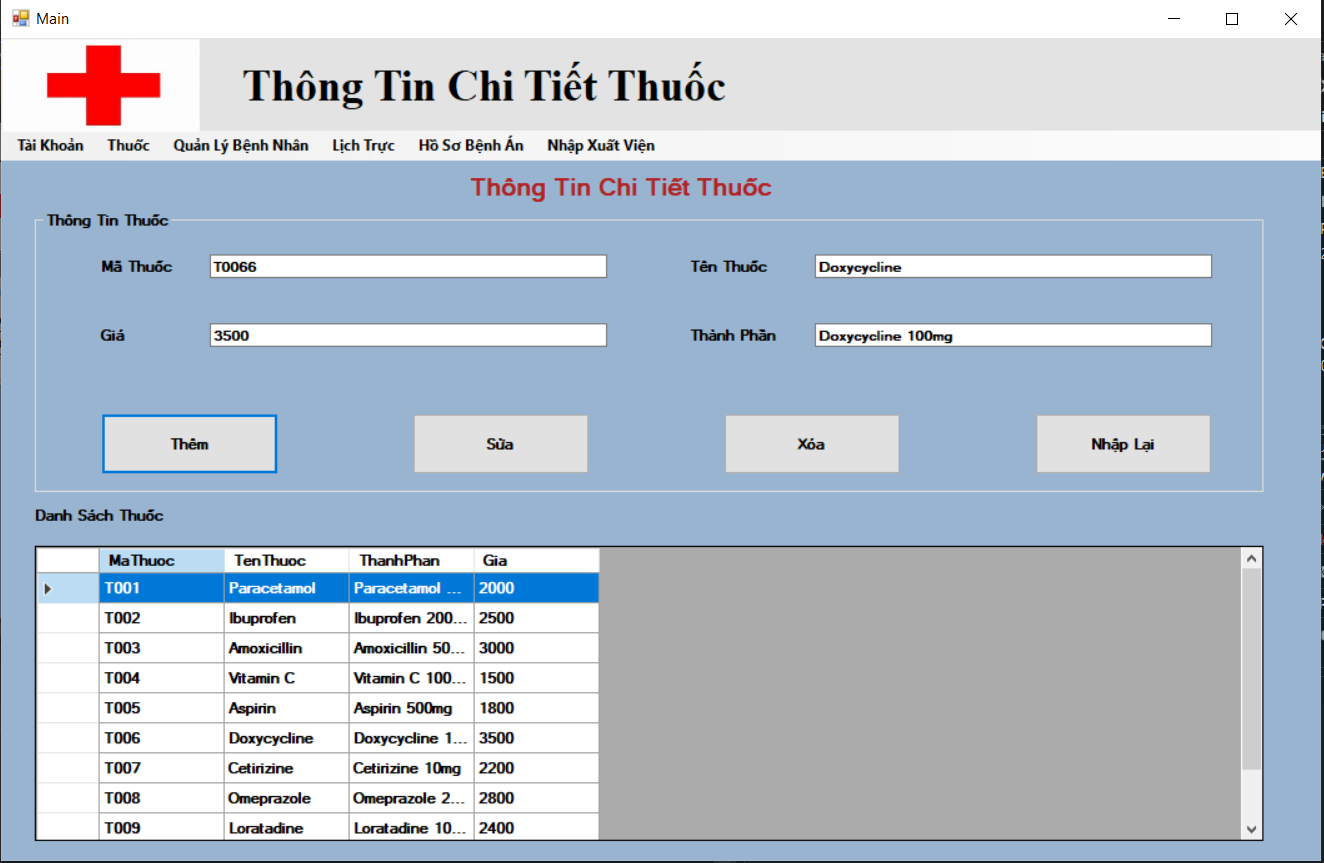
{

closeconnect();

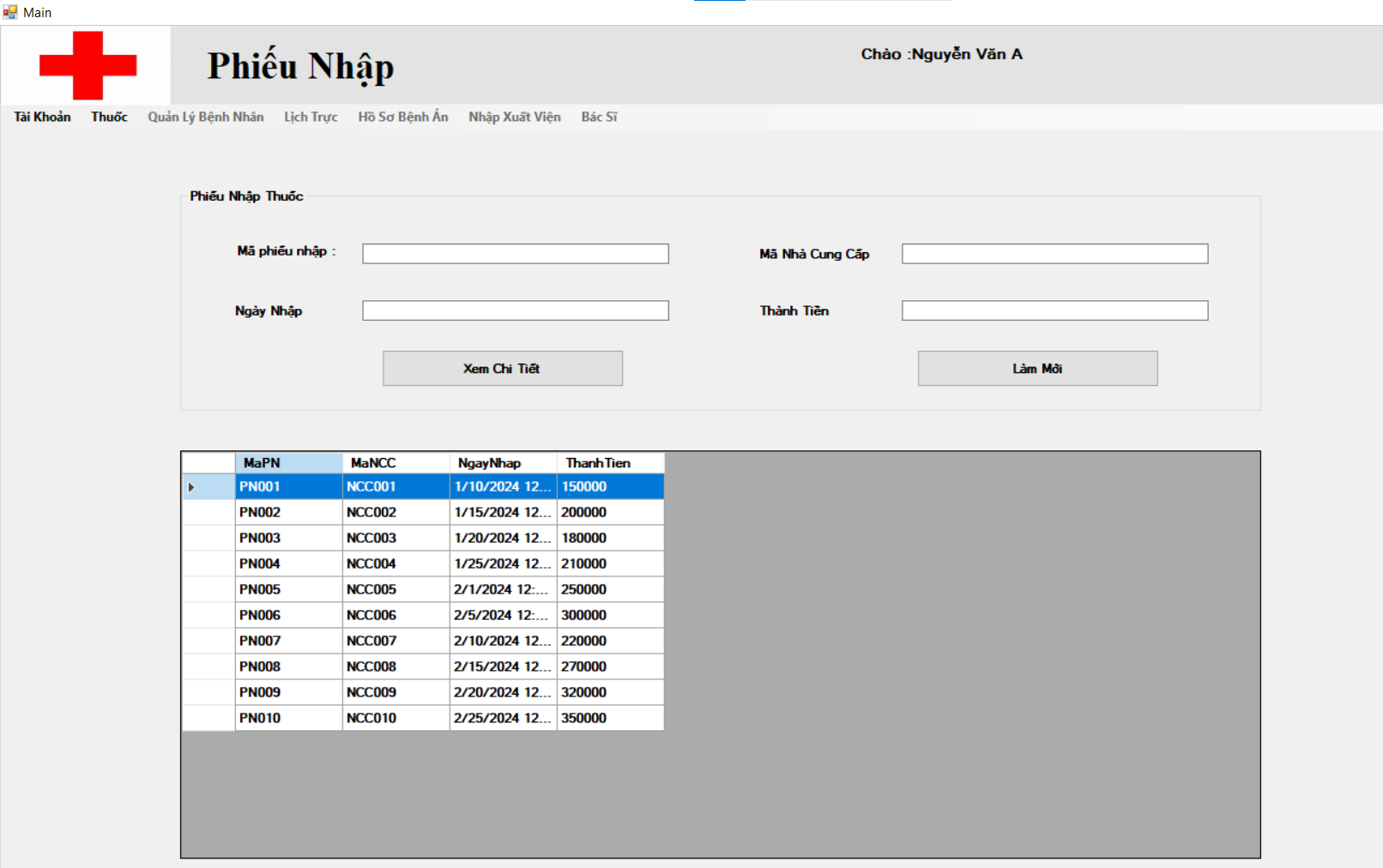
}

}

## 3.3 Nghiệp vụ Quản lý thuốc



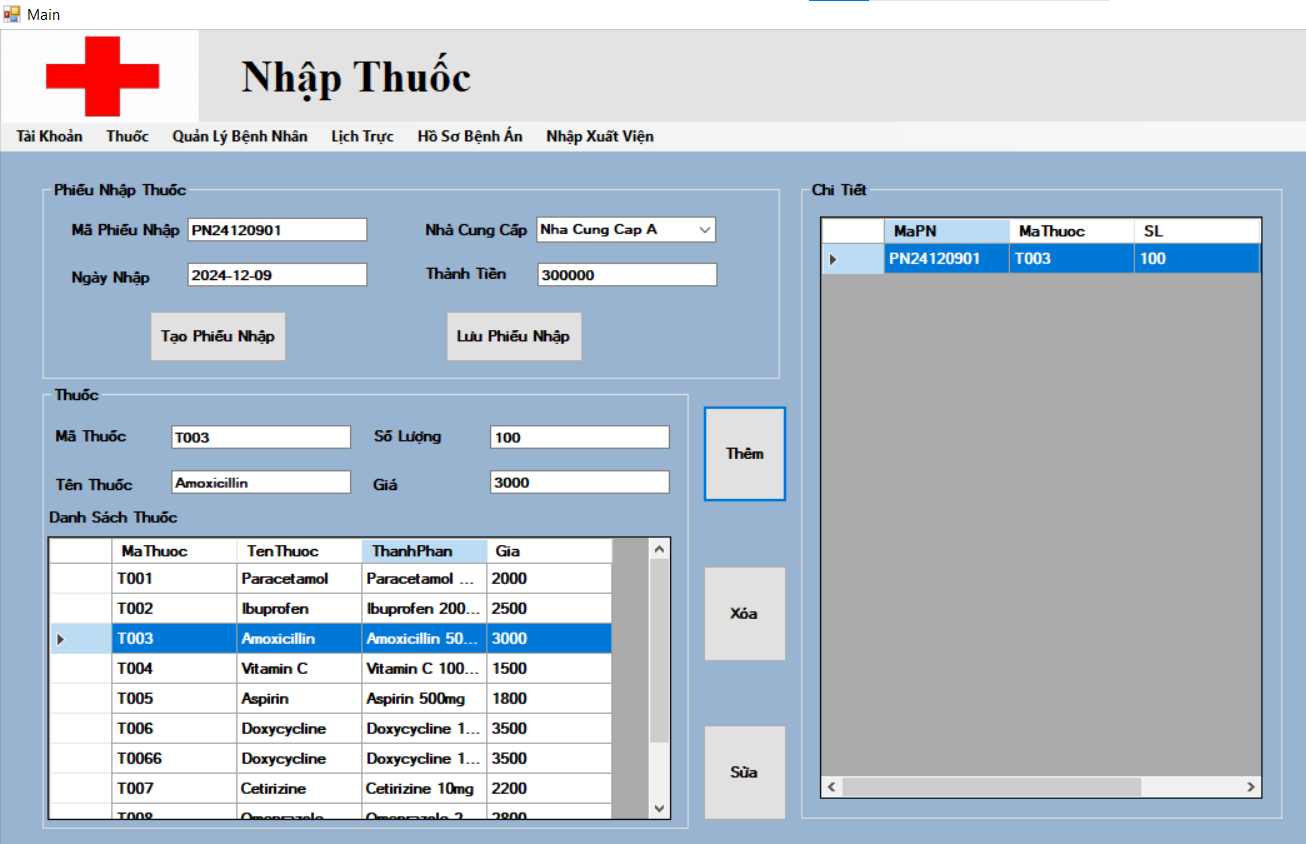
Hình 13 Giao diện thông tin chi tiết thuốc



Hình 14 Giao diện Thống kê phiếu nhập



HÌnh 15 Giao diện lấy thuốc theo toa



HÌnh 16 Giao diện nhập thuốc

Thêm phiếu nhập thuốc

// Vũ

public void ThemPhieuNhapThuoc(string maPN, string maNCC, DateTime ngayNhap, float thanhTien)

    {

        using (SqlConnection conn = new SqlConnection(connectionString))

        {

            try

            {

                conn.Open();

                using (SqlCommand cmd = new SqlCommand("ThemPhieuNhapThuoc", conn))

                {

                    cmd.CommandType = CommandType.StoredProcedure;

                    cmd.Parameters.AddWithValue("@MAPN", maPN);

                    cmd.Parameters.AddWithValue("@MANCC", maNCC);

                    cmd.Parameters.AddWithValue("@NGAYNHAP", ngayNhap);

                    cmd.Parameters.AddWithValue("@THANHTIEN", thanhTien);

                    cmd.ExecuteNonQuery();

                }

                MessageBox.Show("Thêm phiếu nhập thuốc thành công!");

            }

            catch (SqlException ex)

            {

                MessageBox.Show($"Lỗi SQL: {ex.Message}");

            }

            catch (Exception ex)

            {

                MessageBox.Show($"Lỗi: {ex.Message}");

            }

        }

    }

}

Thêm nhà cung cấp

//Vũ

public void ThemNhaCungCap(string maNCC, string tenNCC, string diaChi)

    {

        using (SqlConnection conn = new SqlConnection(connectionString))

        {

            try

            {

                conn.Open();

                using (SqlCommand cmd = new SqlCommand("ThemNhaCungCap", conn))

                {

                    cmd.CommandType = CommandType.StoredProcedure;

                    cmd.Parameters.AddWithValue("@MANCC", maNCC);

                    cmd.Parameters.AddWithValue("@TENNCC", tenNCC);

                    cmd.Parameters.AddWithValue("@DIACHI", diaChi);

                    cmd.ExecuteNonQuery();

                }

                MessageBox.Show("Thêm nhà cung cấp thành công!");

            }

            catch (SqlException ex)

            {

                if (ex.Message.Contains("Mã nhà cung cấp đã tồn tại"))

                {

                    MessageBox.Show("Lỗi: Mã nhà cung cấp đã tồn tại.");

                }

                else

                {

                    MessageBox.Show($"Lỗi SQL: {ex.Message}");

                }

            }

            catch (Exception ex)

            {

                MessageBox.Show($"Lỗi: {ex.Message}");

            }

        }

    }

Thêm chi tiết phiếu nhập thuốc

// Vũ

public void ThemChiTietPhieuNhapThuoc(string maPN, string maThuoc, int soLuong, float gia)

    {

        using (SqlConnection conn = new SqlConnection(connectionString))

        {

            try

            {

                conn.Open();

                // Prepare the SqlCommand to call the stored procedure

                using (SqlCommand cmd = new SqlCommand("ThemChiTietPhieuNhapThuoc", conn))

                {

                    cmd.CommandType = CommandType.StoredProcedure;

                    cmd.Parameters.AddWithValue("@MAPN", maPN);

                    cmd.Parameters.AddWithValue("@MATHUOC", maThuoc);

                    cmd.Parameters.AddWithValue("@SL", soLuong);

                    cmd.Parameters.AddWithValue("@GIA", gia);

                    cmd.ExecuteNonQuery();

                }

                MessageBox.Show("Thêm chi tiết phiếu nhập thuốc thành công!");

            }

            catch (SqlException ex)

            {

                if (ex.Message.Contains("Phiếu nhập không tồn tại"))

                {

                    MessageBox.Show("Lỗi: Phiếu nhập không tồn tại.");

                }

                else if (ex.Message.Contains("Thuốc không tồn tại"))

                {

                    MessageBox.Show("Lỗi: Thuốc không tồn tại.");

                }

                else

                {

                    MessageBox.Show($"Lỗi SQL: {ex.Message}");

                }

            }

            catch (Exception ex)

            {

                MessageBox.Show($"Lỗi: {ex.Message}");

            }

        }

    }

Thêm thuốc

// Vũ

public void ThemThuoc(string maThuoc, string tenThuoc, string thanhPhan, float gia, string donVi)

    {

        using (SqlConnection conn = new SqlConnection(connectionString))

        {

            try

            {

                conn.Open();

                using (SqlCommand cmd = new SqlCommand("ThemThuoc", conn))

                {

                    cmd.CommandType = CommandType.StoredProcedure;

                    cmd.Parameters.AddWithValue("@MATHUOC", maThuoc);

                    cmd.Parameters.AddWithValue("@TENTHUOC", tenThuoc);

                    cmd.Parameters.AddWithValue("@THANHPHAN", thanhPhan);

                    cmd.Parameters.AddWithValue("@GIA", gia);

                    cmd.Parameters.AddWithValue("@DONVI", donVi);

                    cmd.ExecuteNonQuery();

                }

                MessageBox.Show("Thêm thuốc thành công!");

            }

            catch (SqlException ex)

            {

                if (ex.Message.Contains("Mã thuốc đã tồn tại"))

                {

                    MessageBox.Show("Lỗi: Mã thuốc đã tồn tại.");

                }

                else

                {

                    MessageBox.Show($"Lỗi SQL: {ex.Message}");

                }

            }

            catch (Exception ex)

            {

                MessageBox.Show($"Lỗi: {ex.Message}");

            }

        }

    }

Thống kê phiếu nhập theo tháng

// Vũ  
public DataTable ThongKePhieuNhapTheoThang(int thang, int nam)

    {

        DataTable resultTable = new DataTable();

        using (SqlConnection conn = new SqlConnection(connectionString))

        {

            try

            {

                conn.Open();

                string query = @"

                    SELECT \*

                    FROM dbo.ThongKePhieuNhapTheoThang(@Thang, @Nam)";

                using (SqlCommand cmd = new SqlCommand(query, conn))

                {

                    cmd.CommandType = CommandType.Text;

                    cmd.Parameters.AddWithValue("@Thang", thang);

                    cmd.Parameters.AddWithValue("@Nam", nam);

                    using (SqlDataAdapter adapter = new SqlDataAdapter(cmd))

                    {

                        adapter.Fill(resultTable);

                    }

                }

            }

            catch (SqlException ex)

            {

                MessageBox.Show($"Lỗi SQL: {ex.Message}");

            }

            catch (Exception ex)

            {

                MessageBox.Show($"Lỗi: {ex.Message}");

            }

        }

        return resultTable;

    }

Xem Chi tiết phiếu nhập

// Vũ

public DataTable XemChiTietPhieuNhap(string maPN)

    {

        DataTable resultTable = new DataTable();

        using (SqlConnection conn = new SqlConnection(connectionString))

        {

            try

            {

                conn.Open();

                string query = @"

                    SELECT \*

                    FROM dbo.XemChiTietPhieuNhap(@MAPN)";

                using (SqlCommand cmd = new SqlCommand(query, conn))

                {

                    cmd.CommandType = CommandType.Text;

                    cmd.Parameters.AddWithValue("@MAPN", maPN);

                    using (SqlDataAdapter adapter = new SqlDataAdapter(cmd))

                    {

                        adapter.Fill(resultTable);

                    }

                }

            }

            catch (SqlException ex)

            {

                MessageBox.Show($"Lỗi SQL: {ex.Message}");

            }

            catch (Exception ex)

            {

                MessageBox.Show($"Lỗi: {ex.Message}");

            }

        }

        return resultTable;

    }

Xem Thông tin thuốc

// VŨ

public DataTable KiemTraThongTinThuoc(string maThuoc)

    {

        DataTable resultTable = new DataTable();

        using (SqlConnection conn = new SqlConnection(connectionString))

        {

            try

            {

                conn.Open();

                // Prepare the SQL query to call the function

                string query = @"

                    SELECT \*

                    FROM dbo.KiemTraThongTinThuoc(@MATHUOC)";

                using (SqlCommand cmd = new SqlCommand(query, conn))

                {

                    cmd.CommandType = CommandType.Text;

                    // Add the parameter

                    cmd.Parameters.AddWithValue("@MATHUOC", maThuoc);

                    // Execute the query and fill the result into a DataTable

                    using (SqlDataAdapter adapter = new SqlDataAdapter(cmd))

                    {

                        adapter.Fill(resultTable);

                    }

                }

            }

            catch (SqlException ex)

            {

                MessageBox.Show($"Lỗi SQL: {ex.Message}");

            }

            catch (Exception ex)

            {

                MessageBox.Show($"Lỗi: {ex.Message}");

            }

        }

        return resultTable;

## 3.4 Nghiệp vụ quản lý hồ sơ bệnh nhân

A screenshot of a computer

Description automatically generated

HÌnh 17 Giao diện Hồ sơ bệnh án

A screenshot of a computer

Description automatically generated

HÌnh 18 Giao diện tìm kiếm hồ sơ bệnh án

// Lấy danh sách hồ sơ bệnh án

public DataTable GetAllHoSoBenhAn()

{

string query = "SELECT \* FROM HOSOBENHAN";

return dbConnection.GetDataTable(query);

}

// Thêm hồ sơ bệnh án mới

public void AddHoSoBenhAn(HO\_SO\_BENH\_AN\_DTO hoSoBenhAn)

{

string query = "EXEC PROC\_THEMHOSOBENHAN @MaHoSo, @MaBenhNhan, @NgayLapHoSo, @MaBacSi, @NoiDungHoSo";

var parameters = new List<SqlParameter>

{

new SqlParameter("@MaHoSo", SqlDbType.Char) { Value = hoSoBenhAn.MaHoSo },

new SqlParameter("@MaBenhNhan", SqlDbType.Char) { Value = hoSoBenhAn.MaBenhNhan },

new SqlParameter("@NgayLapHoSo", SqlDbType.Date) { Value = hoSoBenhAn.NgayLapHoSo },

new SqlParameter("@MaBacSi", SqlDbType.Char) { Value = hoSoBenhAn.MaBacSi },

new SqlParameter("@NoiDungHoSo", SqlDbType.NVarChar) { Value = hoSoBenhAn.NoiDungHoSo }

};

dbConnection.ExecuteQuery(query, parameters);

}

// Sửa hồ sơ bệnh án

public void UpdateHoSoBenhAn(HO\_SO\_BENH\_AN\_DTO hoSoBenhAn)

{

string query = "EXEC PROC\_CAPNHATHOSOBENHAN @MaHoSo, @MaBenhNhan, @MaBacSi, @NoiDungHoSo";

var parameters = new List<SqlParameter>

{

new SqlParameter("@MaHoSo", SqlDbType.Char) { Value = hoSoBenhAn.MaHoSo },

new SqlParameter("@MaBenhNhan", SqlDbType.Char) { Value = hoSoBenhAn.MaBenhNhan },

new SqlParameter("@MaBacSi", SqlDbType.Char) { Value = hoSoBenhAn.MaBacSi },

new SqlParameter("@NoiDungHoSo", SqlDbType.NVarChar) { Value = hoSoBenhAn.NoiDungHoSo }

};

dbConnection.ExecuteQuery(query, parameters);

}

//Xóa hồ sơ bệnh án

public void DeleteHoSoBenhAn(string maHoSo)

{

string query = "EXEC PROC\_XOAHOSOBENHAN @MaHoSo";

var parameters = new List<SqlParameter>

{

new SqlParameter("@MaHoSo", SqlDbType.Char) { Value = maHoSo }

};

dbConnection.ExecuteQuery(query, parameters);

}

//Tổng hồ sơ

public int GetTotalHoSoBenhAn()

{

string query = "SELECT dbo.FUNC\_LAY\_TONG\_HO\_SO()";

object result = dbConnection.ExecuteScalar(query);

// Kiểm tra và chuyển đổi giá trị trả về

if (result != null && int.TryParse(result.ToString(), out int total))

{

return total;

}

// Nếu không lấy được kết quả, trả về giá trị mặc định

return 0;

}

// Lấy nội dung hồ sơ bệnh án theo mã hồ sơ

public DataTable GetHoSoByMaHoSo(string maHoSo)

{

var parameters = new List<SqlParameter>

{

new SqlParameter("@MaHoSo", maHoSo)

};

// Gọi hàm FUNC\_LAY\_HO\_SO\_THEO\_MA\_HO\_SO

return dbConnection.GetDataTable("SELECT \* FROM dbo.FUNC\_LAY\_HO\_SO\_THEO\_MA\_HO\_SO(@MaHoSo)", parameters);

}

// Lấy hồ sơ bệnh án theo mã bệnh nhân

public DataTable GetHoSoBenhAnByMaBenhNhan(string maBenhNhan)

{

var parameters = new List<SqlParameter>

{

new SqlParameter("@MaBenhNhan", maBenhNhan)

};

// Gọi thủ tục PROC\_LAYDANHSACHHOSOBENHAN

return dbConnection.GetDataTable("EXEC PROC\_LAYDANHSACHHOSOBENHAN @MaBenhNhan", parameters);

}

// Lấy hồ sơ bệnh án theo mã bác sĩ

public DataTable GetHoSoBenhAnByMaBaSi(string maBacSi)

{

var parameters = new List<SqlParameter>

{

new SqlParameter("@MaBacSi", maBacSi)

};

// Gọi hàm FUNC\_KIEM\_TRA\_HO\_SO\_CUNG\_BAC\_SI

return dbConnection.GetDataTable("SELECT \* FROM dbo.FUNC\_KIEM\_TRA\_HO\_SO\_CUNG\_BAC\_SI(@MaBacSi)", parameters);

}

// Lấy hồ sơ bệnh án theo ngày lập

public DataTable GetHoSoBenhAnByNgayLap(DateTime ngayLap)

{

var parameters = new List<SqlParameter>

{

new SqlParameter("@NgayLapHoSo", ngayLap)

};

// Gọi thủ tục PROC\_LAYDANHSACHHOSOTHEONGAY

return dbConnection.GetDataTable("EXEC PROC\_LAYDANHSACHHOSOTHEONGAY @NgayLapHoSo", parameters);

}

// Lấy hồ sơ bệnh án theo khoảng thời gian

public DataTable GetHoSoBenhAnByKhoangThoiGian(DateTime tuNgay, DateTime denNgay)

{

var parameters = new List<SqlParameter>

{

new SqlParameter("@TuNgay", tuNgay),

new SqlParameter("@DenNgay", denNgay)

};

// Gọi hàm FUNC\_LAY\_HO\_SO\_BENH\_AN\_THEO\_KHOANG\_THOI\_GIAN

return dbConnection.GetDataTable("SELECT \* FROM dbo.FUNC\_LAY\_HO\_SO\_BENH\_AN\_THEO\_KHOANG\_THOI\_GIAN(@TuNgay, @DenNgay)", parameters);

}

// Lấy thông tin chi tiết hồ sơ bệnh án theo mã bệnh nhân

public DataTable GetThongTinChiTietByMaBenhNhan(string maBenhNhan)

{

string query = @"

SELECT PK.MaPhieuKham, PK.NgayKham, BN.HoTen AS TenBenhNhan, BS.HoTen AS TenBacSi

FROM PHIEUKHAM PK

INNER JOIN BENHNHAN BN ON PK.MaBenhNhan = BN.MaBenhNhan

INNER JOIN BACSI BS ON PK.MaBacSi = BS.MaBacSi

WHERE BN.MaBenhNhan = @MaBenhNhan";

var parameters = new List<SqlParameter>

{

new SqlParameter("@MaBenhNhan", maBenhNhan)

};

return dbConnection.GetDataTable(query, parameters);

}

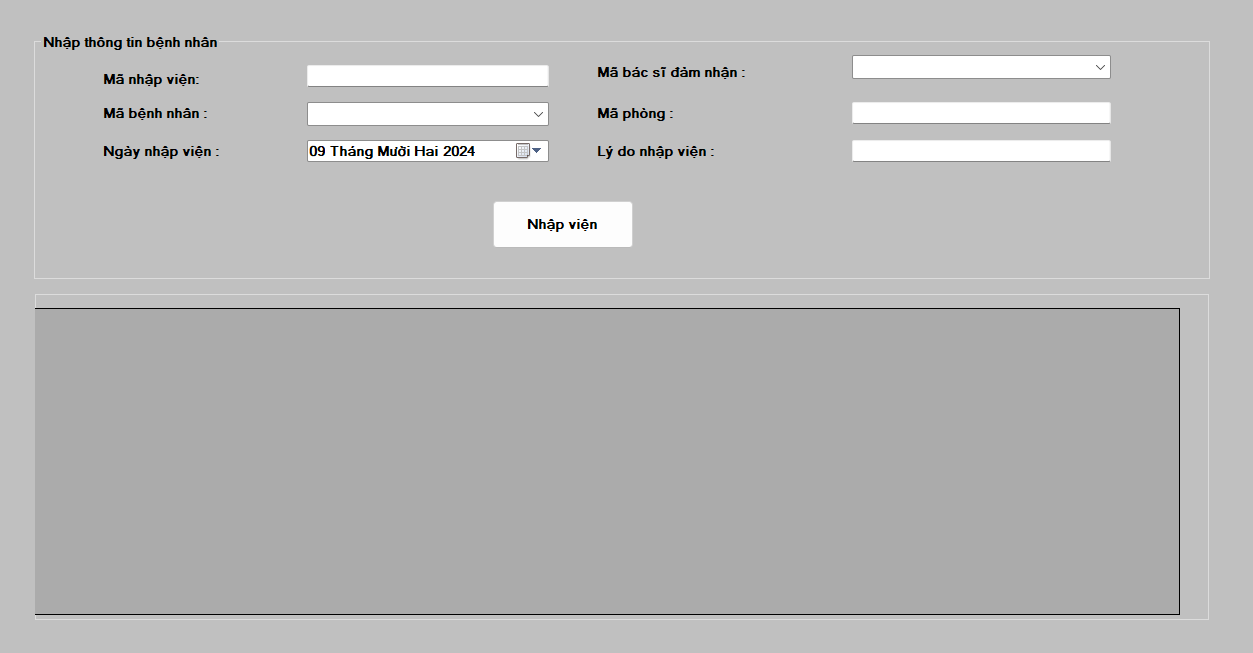
## 3.5 Nghiệp vụ nhập xuất viện



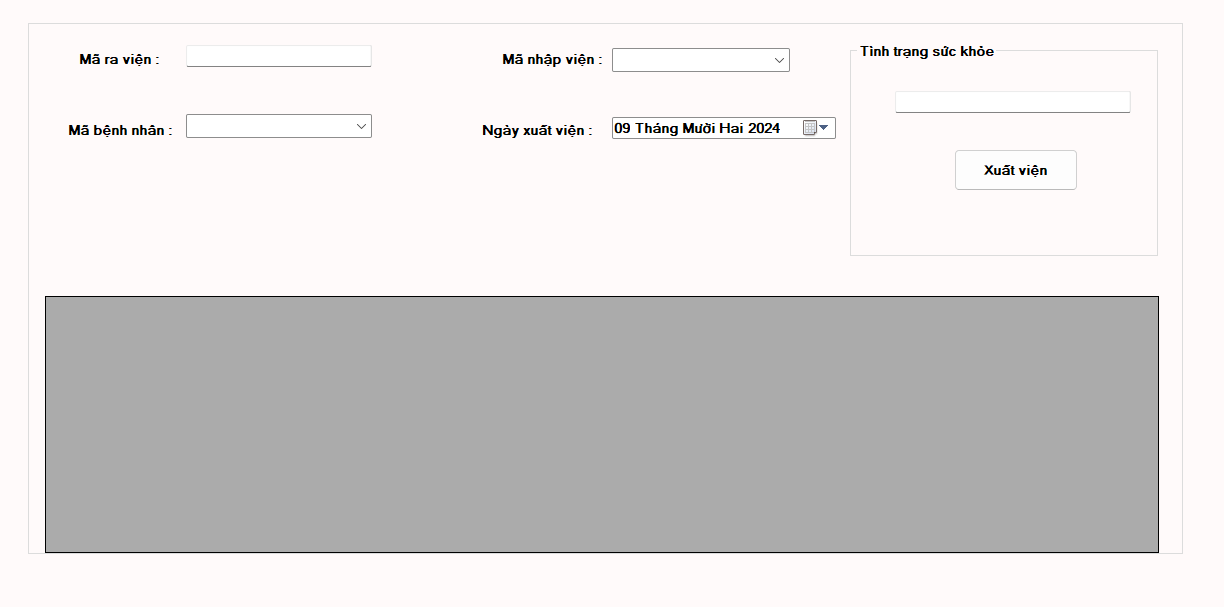
HÌnh 19 Giao diện form kiểm tra thông tin BHYT và Tổng số bệnh nhân đang điều trị ở bệnh viện



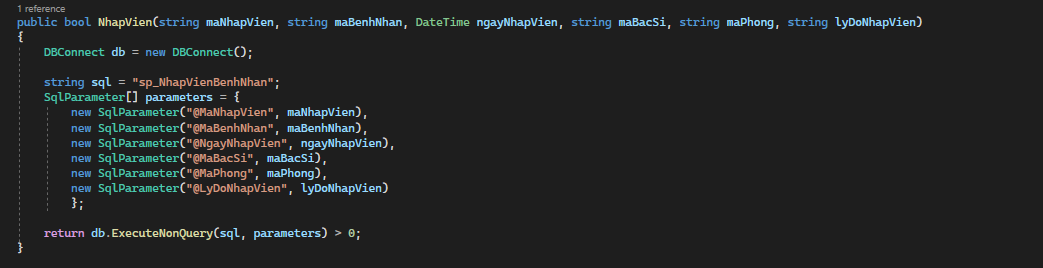
HÌnh 20 Giao diện kiểm tra lịch sử nhập viện của bệnh nhân

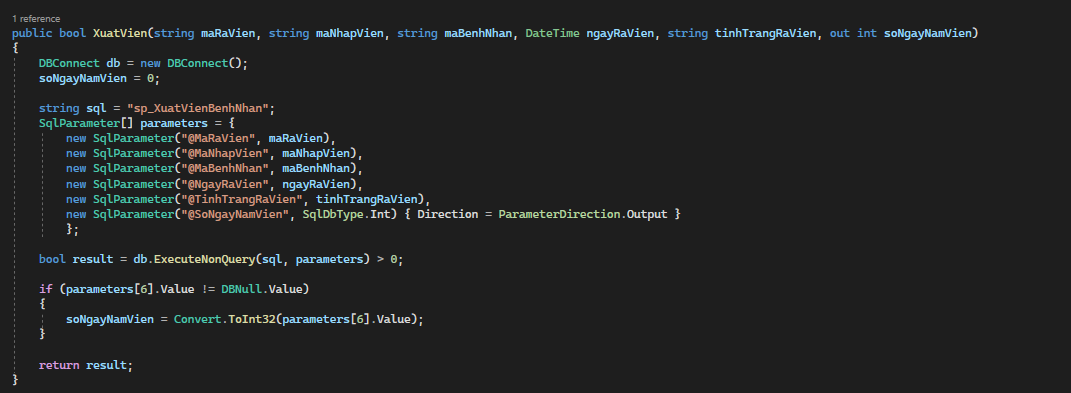


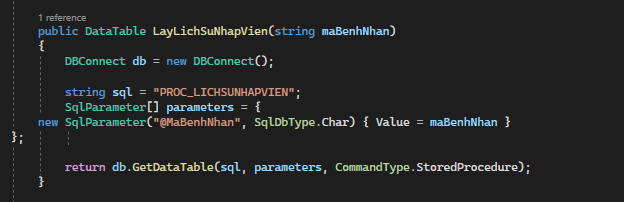
HÌnh 21 Giao diện nhập viện

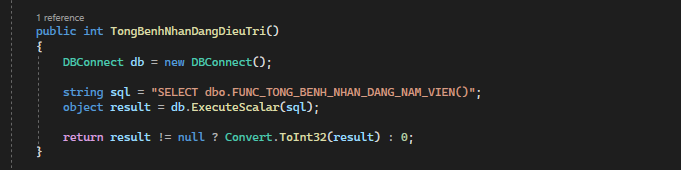


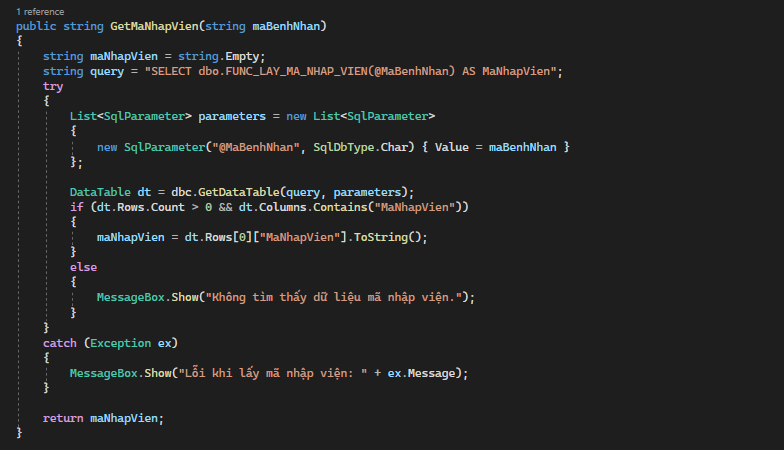
HÌnh 22 Giao diện xuất viện

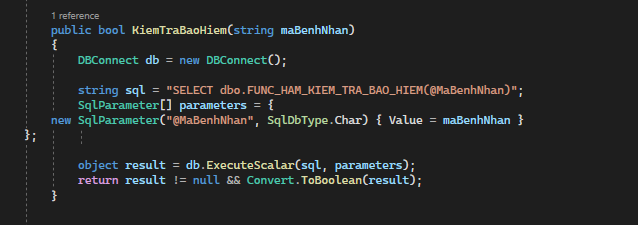












# DANH MỤC TÀI LIỆU THAM KHẢO

* SÁCH
* NGUỒN TỪ INTERNET VÀ WEBSITE