**Bộ Giáo Dục Và Đào Tạo**

**Trường Đại Học Ngoại Ngữ - Tin Học TP-Hồ Chí Minh**

**---o0o---**

Logo

Description automatically generated

**BÁO CÁO KẾT THÚC MÔN HỌC**

**HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

***ĐỀ TÀI: XÂY DỰNG ỨNG DỤNG QUẢN LÝ***

***PHÒNG KHÁM BỆNH CỦA PHÒNG KHÁM ĐẠI HỌC Y***

Khoa: Công Nghệ Thông Tin

Giảng viên hướng dẫn:  ThS. Lê Thị Minh Nguyện

Sinh viên thực hiện:

Tạ Thúy Lam MSSV: 19DH110139

Nguyễn Thị Thúy Hằng MSSV: 19DH110432

Ngô Nhựt Trường MSSV: 19DH110998

Tp. Hồ Chí Minh - Tháng 12/2022

[DANH MỤC HÌNH ẢNH 4](#_Toc122144040)

[DANH MỤC BẢNG 5](#_Toc122144041)

[GIỚI THIỆU 1](#_Toc122144042)

[CHƯƠNG 1 KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU 1](#_Toc122144043)

[1.1 Khảo sát nghiệp vụ 1](#_Toc122144044)

[1.1.1 Quy trình đăng nhập: 1](#_Toc122144045)

[1.1.2 Quy trình tiếp nhận bệnh nhân: 1](#_Toc122144046)

[1.1.3 Quy trình tạo khám lâm sàng: 1](#_Toc122144047)

[1.1.4 Quy trình lập bảng kê dịch vụ: 1](#_Toc122144048)

[1.1.5 Quy trình lập đơn thuốc: 1](#_Toc122144049)

[1.1.6 Quy trình thanh toán: 1](#_Toc122144050)

[1.2 Yêu cầu 2](#_Toc122144051)

[1.2.1 Yêu cầu chức năng 2](#_Toc122144052)

[1.2.2 Yêu cầu phi chức năng 2](#_Toc122144054)

[1.2.2.1 Bảo mật 2](#_Toc122144055)

[1.2.2.2 Hiệu suất 2](#_Toc122144056)

[1.2.2.3 Tiện dụng 2](#_Toc122144057)

[1.2.2.4 An toàn 2](#_Toc122144058)

[CHƯƠNG 2 CƠ SỞ LÝ THUYẾT 3](#_Toc122144059)

[2.1. Giới thiệu 3](#_Toc122144060)

[2.2. Cấu trúc cơ bản SQL Server 4](#_Toc122144062)

[2.3. Một số phiên bản hệ quản trị cơ sở dữ liệu – SQL Server 4](#_Toc122144064)

[2.4. Các chức năng mà SQL server thực hiện 5](#_Toc122144065)

[2.5. Ngôn ngữ sử dụng: Transact SQL 5](#_Toc122144066)

[CHƯƠNG 3. PHÂN TÍCH – THIẾT KẾ DỮ LIỆU 8](#_Toc122144071)

[3.1. Phân tích yêu cầu: 8](#_Toc122144072)

[3.1.1. Sơ đồ Usecase tổng quát 8](#_Toc122144073)

[3.1.2. Sơ đồ Usecase phân rã chức năng: 9](#_Toc122144075)

[3.2. Phân tích dữ liệu 13](#_Toc122144083)

[3.2.1. Mô hình thực thể kết hợp 13](#_Toc122144084)

[3.2.2. Mô hình quan hệ: 13](#_Toc122144086)

[3.2.2.1. NHANVIEN 14](#_Toc122144087)

[3.2.2.2. BENHNHAN 14](#_Toc122144089)

[3.2.2.3. LOAIDICHVU 15](#_Toc122144091)

[3.2.2.4. PHONG 15](#_Toc122144093)

[3.2.2.5. DICHVU 15](#_Toc122144095)

[3.2.2.6. THUOC 16](#_Toc122144097)

[3.2.2.7. PHIEUKHAMBENH 16](#_Toc122144099)

[3.2.2.8. PHIEUDICHVU 17](#_Toc122144101)

[3.2.2.9. CHITIETPHIEUDICHVU 17](#_Toc122144103)

[3.2.2.10. DONTHUOC 18](#_Toc122144105)

[3.2.2.11. CHITIETDONTHUOC 18](#_Toc122144107)

[3.3. Ràng buộc trọn vẹn 19](#_Toc122144109)

[CHƯƠNG 4: THIẾT KẾ XỬ LÝ TRONG CLIENT/SERVER 20](#_Toc122144110)

[4.1. Giao tác 20](#_Toc122144111)

[4.1.1. Thêm hồ sơ bệnh nhân 20](#_Toc122144114)

[4.1.2. Xóa hồ sơ bệnh nhân 20](#_Toc122144116)

[4.1.3. Cập nhật hồ sơ bệnh nhân 21](#_Toc122144120)

[4.1.4. Tìm kiếm hồ sơ bệnh nhân 21](#_Toc122144122)

[4.1.5. Thêm phiếu khám bệnh 22](#_Toc122144124)

[4.1.6. Cập nhật phiếu khám bệnh 22](#_Toc122144126)

[4.1.7. Thêm chi tiết dịch vụ: 23](#_Toc122144128)

[4.1.8. Xóa chi tiết dịch vụ 23](#_Toc122144130)

[4.1.9. Thêm chi tiết đơn thuốc: 24](#_Toc122144132)

[4.1.10. Xóa chi tiết đơn thuốc 24](#_Toc122144134)

[4.2. Phân quyền: 25](#_Toc122144136)

[4.2.1. Các đối tượng phân quyền: 25](#_Toc122144139)

[4.2.2. Phân quyền trên các bảng: 25](#_Toc122144140)

[4.2.3. Phân quyền trên các giao tác: 26](#_Toc122144142)

[CHƯƠNG 5: TRIỂN KHAI THỰC NGHIỆM 27](#_Toc122144144)

[5.1 Phía Server 27](#_Toc122144145)

[5.2 Phía Client 45](#_Toc122144152)

[5.2.1. Giao diện đăng nhập: 45](#_Toc122144154)

[5.2.2. Giao diện Trang chủ: 45](#_Toc122144156)

[5.2.3. Giao diện Quản lý hồ sơ bệnh nhân: 46](#_Toc122144158)

[5.2.4. Giao diện Tiếp đón bệnh nhân 46](#_Toc122144160)

[5.2.5. Giao diện Tra cứu phiếu khám 47](#_Toc122144162)

[5.2.6. Giao diện Khám bệnh: 47](#_Toc122144164)

[5.2.7. Giao diện Lập phiếu dịch vụ: 48](#_Toc122144166)

[5.2.8. Giao diện Lập đơn thuốc: 48](#_Toc122144168)

[5.2.9. Giao diện Quản lý dịch vụ: 49](#_Toc122144170)

[5.2.10. Giao diện Quản lý thuốc: 49](#_Toc122144172)

[5.2.11. Giao diện Quản lý nhân viên: 50](#_Toc122144174)

[CHƯƠNG 6: KẾT LUẬN 51](#_Toc122144176)

[6.1. Phía server 51](#_Toc122144178)

[6.2 Kết quả chưa đạt được 51](#_Toc122144181)

[6.3 Hướng phát triển trong tương lai 51](#_Toc122144182)

[TÀI LIỆU THAM KHẢO 52](#_Toc122144183)

# DANH MỤC HÌNH ẢNH

[*Hình 1: Sơ đồ chức năng 2*](#_Toc122144053)

[*Hình 2: Hình mô tảMicrosoft SQL Server 3*](#_Toc122144061)

[*Hình 3: Mô hình cấu trúc SQL Server 4*](#_Toc122144063)

[*Hình 4: Trạng thái của giao tác 6*](file:///D:\Lập%20trình\Hệ%20quản%20trị%20CSDL\HQTCSDL_QLKhamBenh2%20(2).docx#_Toc122144068)

[*Hình 5: Sơ đồ Use case Tổng quát 8*](#_Toc122144074)

[*Hình 6: Sơ đồ Use case Quản lý đơn thuốc 9*](#_Toc122144076)

[*Hình 7: Sơ đồ Use case quản lý phiếu dịch vụ 9*](#_Toc122144077)

[*Hình 8: Sơ đồ Use case Quản lý phiếu khám 10*](#_Toc122144078)

[*Hình 9: Sơ đồ Use case quản lý hồ sơ bệnh nhân 10*](#_Toc122144079)

[*Hình 10: Sơ đồ Use case Quản lý dịch vụ 11*](#_Toc122144080)

[*Hình 11: Sơ đồ Use case quản lý thuốc 11*](#_Toc122144081)

[*Hình 12: Sơ đồ Use case quản lý nhân viên 12*](#_Toc122144082)

[*Hình 13: Mô hình dữ liệu vật lý 13*](#_Toc122144085)

[*Hình 14: Giao diện đăng nhập 45*](#_Toc122144155)

[*Hình 15: Giao diện trang chủ 45*](#_Toc122144157)

[*Hình 16: Giao diện quản lý hồ sơ bệnh nhân 46*](#_Toc122144159)

[*Hình 17: Giao diện tiếp đón bệnh nhân 46*](#_Toc122144161)

[*Hình 18: Giao diện tra cứu phiếu khám 47*](#_Toc122144163)

[*Hình 19: Giao diện khám bệnh 47*](#_Toc122144165)

[*Hình 20: Giao diện Lập phiếu dịch vụ 48*](#_Toc122144167)

[*Hình 21: Giao diện Lập đơn thuốc 48*](#_Toc122144169)

[*Hình 22: Giao diện Lập phiếu dịch vụ 49*](#_Toc122144171)

[*Hình 23: Giao diện Quản lý thuốc 49*](#_Toc122144173)

[*Hình 24: Giao diện Quản lý nhân viên 50*](#_Toc122144175)

# DANH MỤC BẢNG

[*Bảng 1: Bảng NhanVien 14*](#_Toc122144088)

[*Bảng 2: Bảng BenhNhan 14*](#_Toc122144090)

[*Bảng 3: Bảng LoaiDichVu 15*](#_Toc122144092)

[*Bảng 4: Bảng Phong 15*](#_Toc122144094)

[*Bảng 5: Bảng DichVu 15*](#_Toc122144096)

[*Bảng 6: Bảng Thuoc 16*](#_Toc122144098)

[*Bảng 7: Bảng PhieuKhamBenh 16*](#_Toc122144100)

[*Bảng 8: Bảng PhieuDichVu 17*](#_Toc122144102)

[*Bảng 9: Bảng ChiTietPhieuDichVu 17*](#_Toc122144104)

[*Bảng 10: Bảng DonThuoc 18*](#_Toc122144106)

[*Bảng 11: Bảng ChiTietDonThuoc 18*](#_Toc122144108)

[*Bảng 12: Bảng các bước xử lý giao tác Thêm hồ sơ bệnh nhân 20*](#_Toc122144115)

[*Bảng 13: Bảng các bước xử lý giao tác Xóa hồ sơ bệnh nhân 20*](#_Toc122144117)

[*Bảng 14: Bảng các bước xử lý giao tác Cập nhật hồ sơ bệnh nhân 21*](#_Toc122144121)

[*Bảng 15: Bảng các bước xử lý giao tác Tìm kiếm hồ sơ bệnh nhân 21*](#_Toc122144123)

[*Bảng 16: Bảng các bước xử lý giao tác Thêm phiếu khám bệnh 22*](#_Toc122144125)

[*Bảng 17: Bảng các bước xử lý giao tác Cập nhật phiếu khám bệnh 22*](#_Toc122144127)

[*Bảng 18: Bảng các bước xử lý giao tác Thêm chi tiết dịch vụ 23*](#_Toc122144129)

[*Bảng 19: Bảng các bước xử lý giao tác Xóa chi tiết dịch vụ 23*](#_Toc122144131)

[*Bảng 20: Bảng các bước xử lý giao tác Thêm chi tiết đơn thuốc 24*](#_Toc122144133)

[*Bảng 21: Bảng các bước xử lý giao tác Xóa chi tiết đơn thuốc 24*](#_Toc122144135)

[*Bảng 22: Bảng phân quyền đối tượng trên các bảng 25*](#_Toc122144141)

[*Bảng 23: Bảng phân quyền các đối tượng trên giao tác 26*](#_Toc122144143)

# GIỚI THIỆU

Nhu cầu sử dụng các dịch vụ khám bệnh ngày càng cao, các phòng khám lớn nhỏ càng phải đáp ứng đầy đủ các dịch vụ và chất lượng để phục vụ các bệnh nhân và khách hàng một cách hiệu quả và tối ưu nhất. Việc đó đòi hỏi phòng khám cần phải có một hệ thống thông tin quản lý toàn bộ dữ liệu liên quan đến dịch vụ và thuốc mà các bệnh nhân được sử dụng trong mục đích khám và chữa bệnh. Các công việc như tìm kiếm, lưu trữ thông tin, phân loại các dịch vụ, thuốc,…tạo và quản lý các thông tin cần thiết liên quan đến phòng khám sẽ được thực hiện trực tiếp. Điều này giúp cho công việc quản lý thông tin dữ liệu của phòng khám được thực hiện một cách thuận tiện, dễ dàng và không làm mất quá nhiều thời gian như trước.

Ngày này cùng với sự phát triển vượt bậc của khoa học kỹ thuật thì công nghệ thông tin là một trong những lĩnh vực có nhiều đóng góp thiết thực nhất, công nghệ thông tin có mặt trong hầu hết các lĩnh vực của cuộc sống đặc biệt công nghệ thông tin là một công cụ hỗ trợ rất đắc lực trong công tác quản lý. Chúng ta dễ dàng thấy được việc đưa tin học vào trong quản lý kinh doanh là một trong những ứng dụng quan trọng trong rất nhiều ứng dụng cơ sở dữ liệu. Nhờ vào công tác tin học hóa mà công việc quản lý và điều hành doanh nghiệp tỏ ra rất nhanh chóng và hiệu quả. Chính vì lẽ đó mà cơ sở dữ liệu như là một giải pháp hữu hiệu nhất cho các doanh nghiệp có thể tồn tại và phát triển một cách nhanh chóng.

Phòng khám đại học Y là một trong những trung tâm khám chữa bệnh theo dịch vụ lớn nhất ở TP HCM. Do đó nhu cầu ứng dụng cơ sở dữ liệu trong việc quản lý bệnh nhân là hết sức thiết thực. Vì vậy chúng em chọn đề tài xây dựng ứng dụng quản lý khám bệnh của đại học Y. Hệ thống được phân tích và thiết kế theo phương pháp có cấu trúc và được cài đặt trên môi trường Visual Studio và được thiết kế dữ liệu dựa trên ngôn ngữ Oracle, các chương tiếp theo sẽ nói rõ hơn về vấn đề này

# Chương 1 KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU

## Khảo sát nghiệp vụ

### Quy trình đăng nhập:

* Nhân viên đăng nhập vào hệ thống bằng tài khoản và mật khẩu.
* Tùy vào chức vụ mà chức năng bị hạn chế.

### Quy trình tiếp nhận bệnh nhân:

* Bệnh nhân đến nơi tiếp nhận bệnh nhân để đăng ký khám chữa bệnh, nhân viên sẽ lập phiếu khám bệnh cho bệnh nhân.
* Nếu bệnh nhân đã từng khám thì tra từ danh sách, nếu là bệnh nhân mới thì nhập thông tin bệnh nhân.

### Quy trình tạo khám lâm sàng:

* Bệnh nhân được phân vào phòng khám theo yêu cầu.
* Bác sĩ sẽ khám và cập nhật phiếu khám (chẩn đoán)

### Quy trình lập bảng kê dịch vụ:

* Bác sĩ sẽ kê những loại dịch vụ bệnh nhân cần thực hiện để kiểm tra sức khỏe (nếu có).

### Quy trình lập đơn thuốc:

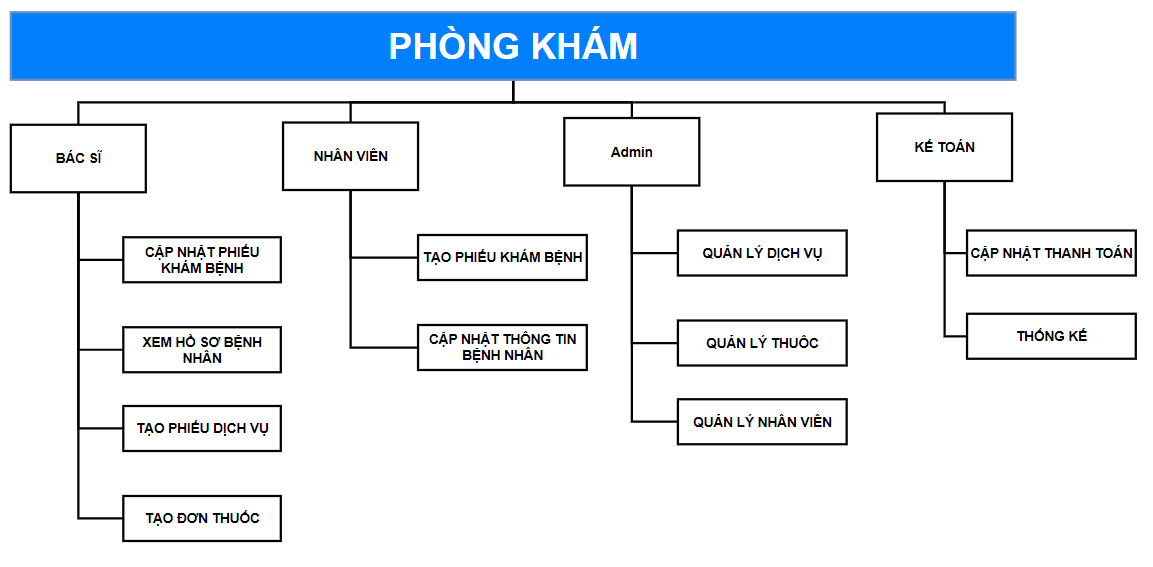
* Bác sĩ sẽ kê những loại thuốc điều trị.

### Quy trình thanh toán:

* Bệnh nhân phải thanh toán theo từng phiếu.
* Phiếu khám bệnh là 100 nghìn.
* Phiếu dịch vụ phải được thanh toán trước khi thực hiện.
* Đơn thuốc phải được thanh toán trước khi nhận thuốc.

## Yêu cầu

### Yêu cầu chức năng



### Yêu cầu phi chức năng

#### Bảo mật

* Đảm bảo tính bảo mật của hệ thống với người ngoài hệ thống
* Giới hạn các thông tin người dùng được truy cập theo vai trò của người dùng
* Giới hạn các giao tác người dùng có thể thực hiện theo vai trò của người dùng

#### Hiệu suất

* Thời gian phản hồi, thời gian xử lý và thời gian trả kết quả nhanh

#### Tiện dụng

* Cung cấp đầy đủ các chức năng mà người dùng yêu cầu
* Dễ học cách sử dụng
* Hạn chế lỗi nhập liệu

#### An toàn

* Hoạt động ổn định, chính xác
* Có thể khôi phục hệ thống, khôi phục dữ liệu sau các sự cố

# Chương 2 CƠ SỞ LÝ THUYẾT

## Giới thiệu



SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS)) sử dụng câu lệnh SQL (Transact-SQL) để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

SQL Server được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user. SQL Server có thể kết hợp “ăn ý” với các server khác như Microsoft Internet Information Server (IIS), E-Commerce Server, Proxy Server….

## Cấu trúc cơ bản SQL Server

Diagram

Description automatically generated

**SQLOS**

SQLOS là viết tắt của hệ điều hành SQL server. Đây là tầng cuối cùng trong kiến trúc tổng thể của SQL server. Tại đây sẽ chịu trách nhiệm xử lý các nhiệm vụ như quản lý bộ nhớ, lên lịch nhiệm vụ, khoá dữ liệu nhằm tránh các xung đột ngoài ý muốn có thể xảy ra mỗi khi thực hiện các thao tác cập nhật.

**Database engine**

Đây là một công cụ có chức năng quản lý việc lưu trữ, xử lý và bảo mật dữ liệu. Trong đây sẽ bao gồm rất nhiều các công cụ khác nhau như một công cụ lưu trữ quản lý các tệp, bảng, trang, chỉ mục, bộ đệm dữ liệu và giao dịch cơ sở dữ liệu.

**External protocol**

Đây là các giao thức được sử dụng để giao tiếp với Database engine. Nó bao gồm TCP/IP hay VIA (Virtual Interface Adapter),...

## Một số phiên bản hệ quản trị cơ sở dữ liệu – SQL Server

**Enterprise (Bản Doanh nghiệp):** Phiên bản này chứa gần như đầy đủ nhất các tinh hoa của SQL Server. Trong đó phải kể đến như nhân bộ máy cơ sở dữ liệu và các dịch vụ đi kèm cùng với các công cụ cho tạo và quản lý phân cụm SQL Server. Với phiên bản Enterprise, nó hướng đến các CSDL lớn tới 524 [petabytes](https://en.wikipedia.org/wiki/Petabyte) và hỗ trợ tới 640 bộ vi xử lý.

**Standard (Bản Tiêu chuẩn):** Phiên bản này rất thích hợp cho các công ty vừa và nhỏ. Một phần vì giá thành rẻ hơn nhiều so với phiên bản Enterprise, một phần vì những công ty vừa và nhỏ cũng chưa có lượng dữ liệu lớn đến thế.

**Developer (Bản Developer):** Phiên bản này có đầy đủ các tính năng của bản Enterprise nhưng nó được tuỳ biến để nhằm phục vụ cho việc phát triển và kiểm tra ứng dụng.

**Workgroup (Bản Workgroup):** Phiên bản SQL Server này chỉ bao gồm chức năng lõi cơ sở dữ liệu nhưng không có các dịch vụ đi kèm. Đặc biệt hơn từ phiên bản SQL server 2012, phiên bản này đã bị Microsoft khai tử.

**Express (Bản Express):** SQL Server phiên bản này được đánh giá là dễ sử dụng. Có thể nó hướng đến dùng cho việc quản trị cơ sở dữ liệu đơn giản. Microsoft tích hợp nó cùng với Microsoft Visual Studio, nên giúp dễ dàng để phát triển các ứng dụng dữ liệu. Đặc biệt hơn nữa đây là một phiên bản miễn phí. Tuy nhiên nhược điểm là  nó chỉ dùng cho 1 CPU với 1 GB bộ nhớ và 10 GB file cơ sở dữ liệu.

## Các chức năng mà SQL server thực hiện

Dưới đây là một số lý do cho phép người sử dụng ưa chuộng sử dụng SQL để thực hiện đa dạng các chức năng như sau:

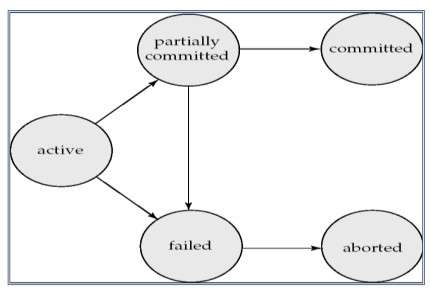
* Cho phép người sử dụng có thể dùng để truy cập dữ liệu bên trong hệ thống quản lý cơ sở dữ liệu quan hệ.
* Cho phép người dùng mô tả dữ liệu.
* Cho phép người dùng thực hiện xác định dữ liệu bên trong cơ sở dữ liệu và thực hiện các thao tác dữ liệu.
* Cho phép người dùng tạo và thả cho các cơ sở dữ liệu cũng như bảng.
* SQL cho phép người sử dụng để thực hiện tạo ra các chế độ view, các thủ tục lưu trữ và chức năng trong cơ sở dữ liệu.
* Cho phép người dùng để thực hiện thiết lập quyền trên các bảng, view và thủ tục.



## Ngôn ngữ sử dụng: Transact SQL

T-SQL là việc thực thi ngôn ngữ cấu trúc chuẩn của Microsoft được dùng trong hệ thống [SQL Server](https://vietnix.vn/sql-server-la-gi/). Ngôn ngữ đặc biệt này có thể được dùng để thực hiện các thao tác với bảng như định nghĩa, chèn, xóa, cập nhật, truy vấn [dữ liệu](https://vietnix.vn/du-lieu-la-gi/).

Cấu trúc của ngôn ngữ T-SQL được xây dựng theo từng khối lệnh được bắt đầu bằng BEGIN và kết thúc là END. Một khối lệnh có thể được đặt lồng trong một khối lệnh bao trùm khác, trong mỗi khối lệnh là các câu lệnh, các câu lệnh này được ngăn cách với nhau bởi dấu chấm phẩy.

* **Quản lí giao tác**
* Giao tác là 1 đơn vị xử lý nguyên tố gồm 1 chuỗi các hành động tương tác lên CSDL. Khi thực hiện một giao tác hoặc phải thực hiện tất cả các hành động của nó hoặc không thực hiện hành động nào hết
* Tính chất ACID của giao tác
  + **Tính Nguyên tố (Atomicity):** Hoặc là toàn bộ hoạt động của giao dịch được phản ánh đúng đắn trong CSDL hoặc không có hoạt động nào cả. Nói cách khác, tác dụng của các câu lệnh trong một giao tác phải như là một câu lệnh đơn. Không chia nhỏ ra được.
  + **Tính Nhất quán (Consistency):** Bất kỳ CSDL nào thì mọi ràng buộc tòan vẹn phải thỏa. Tại bất kỳ thời điểm mà mọi RBTV được thỏa gọi là tính nhất quán. Một giao tác phải biến CSDL từ trạng thái nhất quán này sang trạng thái nhất quán khác không được phá vở trạng thái nhất quán. E1 →T E2 (E1nhất quán thì E2 phải nhất quán)
  + **Tính Cô lập (Isolation):** Một giao tác không quan tâm đến các giao tác khác xử lý đồng thời với nó Khi có n giao tác xử lý đồng thời phải làm sao bảo đảm là tôi có tính độc lập của riêng tôi.
  + **Tính Bền vững (Durability):** Tính bền vững đảm bảo rằng một khi giao dịch được hoàn thành, tất cả các cập nhật trên CSDL là bền vững thậm chí nếu có lỗi hệ thống sau khi giao dịch hoàn thành thực hiện. Giả sử một lỗi hệ thống có thể dẫn đến mất mát dữ liệu trong bộ nhớ chính, nhưng dữ liệu được ghi vào đĩa không bao giờ bị mất.
* Trạng thái của giao tác:
* **Các vấn đề trong truy xuất đồng thời:**
* **Mất dữ liệu đã cập nhật (lost updated):** Tình trạng xảy ra khi hai hay nhiều thao tác của các giao tác khác nhau cùng yêu cầu truy cập một mục dữ liệu. Các dữ liệu đã được các thao tác trước cập nhật nhưng lại bị các thao tác sau cập nhật lại làm thay đổi kết quả mong muốn.
* **Không thể đọc lại (unrepeatable read):**  Tình trạng này xảy ra khi có nhiều hơn một giao tác cùng thực hiện cập nhật trên 1 đơn vị dữ liệu. Khi đó, tác dụng của giao tác cập nhật thực hiện sau sẽ đè lên tác dụng của thao tác cập nhật trước.
* **“Bóng ma” (phantom):** Là tình trạng mà một giao tác đang thao tác trên một tập dữ liệu nhưng giao tác khác lại chèn thêm các dòng dữ liệu vào tập dữ liệu mà giao tác kia quan tâm
* **Đọc dữ liệu chưa chính xác (dirty read):** Xảy ra khi một giao tác thực hiện đọc trên một đơn vị dữ liệu mà đơn vị dữ liệu này đang bị cập nhật bởi một giao tác khác nhưng việc cập nhật chưa được xác nhận.
* **Cơ chế phục hồi dữ liệu sau sự cố**
* Khôi phục CSDL là tiến trình phục hồi CSDL về trạng thái nhất quán cuối cùng trước khi có sự cố xảy ra. Việc khôi phục dữ liệu được thực hiện bởi bộ quản lý khôi phục dữ liệu (RM- Recovery Manager). Khôi phục dữ liệu tự động giúp giảm thiểu việc yêu cầu người sử dụng thực hiện lại công việc (khi xảy ra sự cố). Giao tác là đơn vị cơ bản khi khôi phục CSDL
* Phân loại sự cố:
  + Sự cố do nhập liệu sai
  + Sự cố của giao tác (transaction failure)
  + Sự cố liên quan đến hệ thống (system failure)
  + Sự cố trên thiết bị lưu trữ (media failure)
* Khôi phục sự cố sau một giao tác:

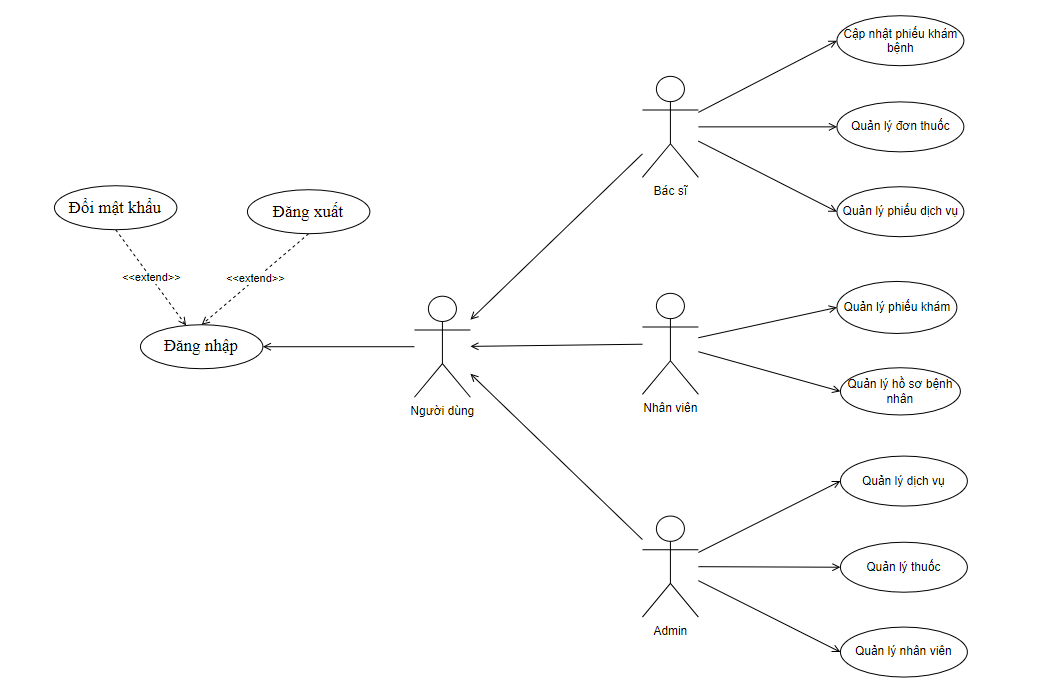
Nếu một giao tác T bị sự cố thì hệ thống sẽ thực hiện các việc sau:

* + Giải phóng các đơn vị dữ liệu đang bị khóa bởi T (UL(B))
  + Reset lại giá trị các đơn vị dữ liệu mà T đã làm thay đổi (A) (khó???)
  + Xét xem có giao tác nào(#T) đã đọc dữ liệu do T ghi không? Nếu có: -T’ chưa commit →hủy - T’ commit rồi → thực hiện lại
* Khôi phục sau sự cố hệ thống:
  + **Nhật ký giao tác:** Nhật ký giao tác là một chuỗi các mẫu tin (log record) ghi nhận lại các hành động của DBMS. Nhật ký là một tập tin tuần tự được lưu trữ trên bộ nhớ chính, và sẽ được ghi xuống đĩa ngay khi có thể
  + **Nghi thức Undo:** Thực hiện cho những giao tác chưa kết thúc nhưng xảy ra sự cố hay nhưng giao tác bị rollback
  + **Nghi thức REDO:** Thực hiện cho những giao tác đã commit nhưng chưa được ghi nhận trên CSDL Vật lí
  + **Phục hồi nóng:** Khi hệ thống đang hoạt động. Cơ chế này dựa trên nhật ký, các bản sao dữ liệu và điểm phục hồi hệ thống
  + **Phục hồi lạnh:** Xảy ra do lỗi thiết bị (thiết bị hư, thay đĩa khác). Thông thường người ta có một phiên bản CSDL và nhật ký nằm trên một đĩa khác. Cơ chế này dựa trên “nhật ký sau”. Cơ chế phục hồi lạnh tùy thuộc vào người quản trị CSDL

# Chương 3. PHÂN TÍCH – THIẾT KẾ DỮ LIỆU

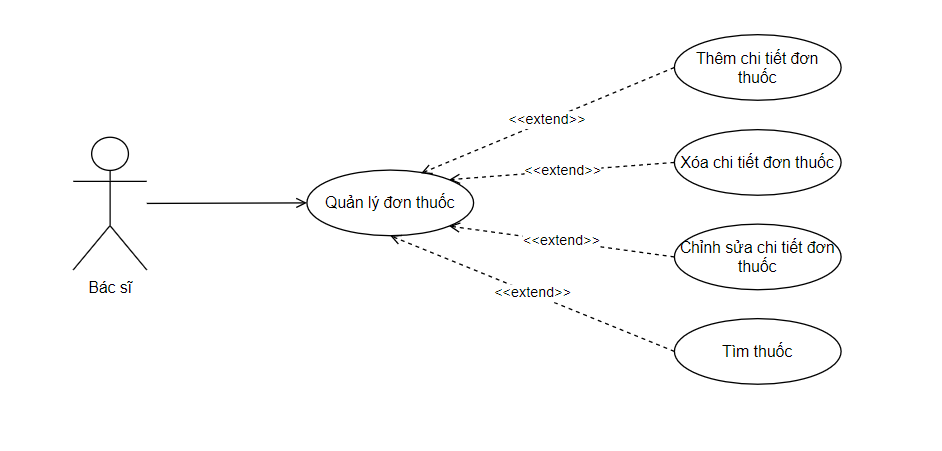
## Phân tích yêu cầu:

### Sơ đồ Usecase tổng quát

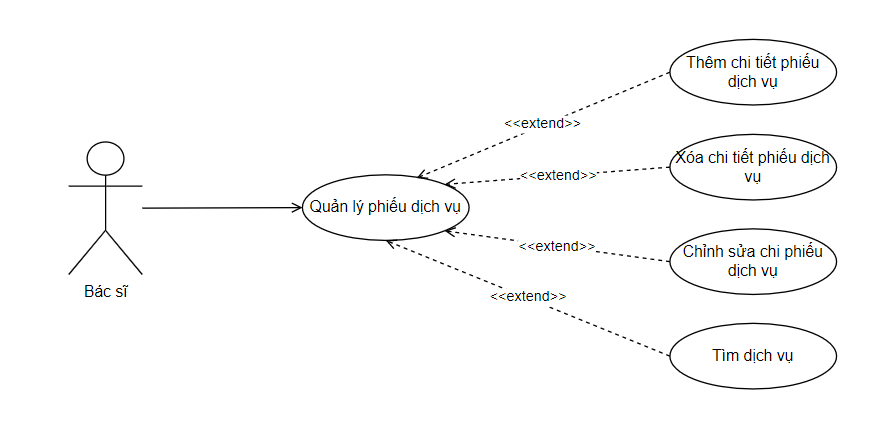


### Sơ đồ Usecase phân rã chức năng:

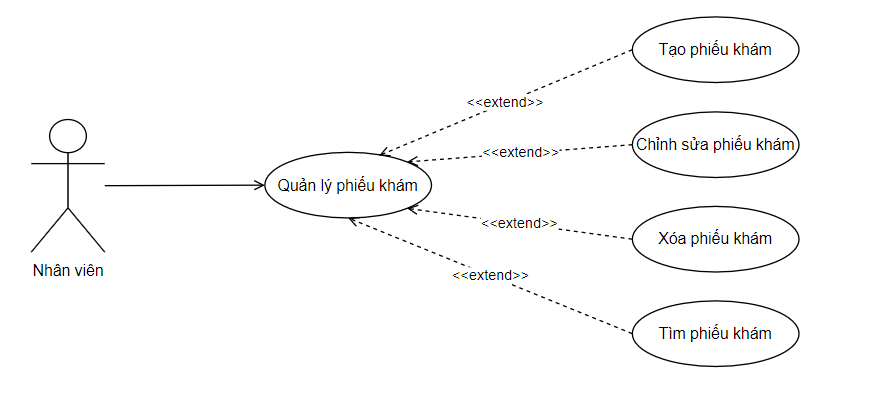
* Sơ đồ usecase quản lý đơn thuốc:



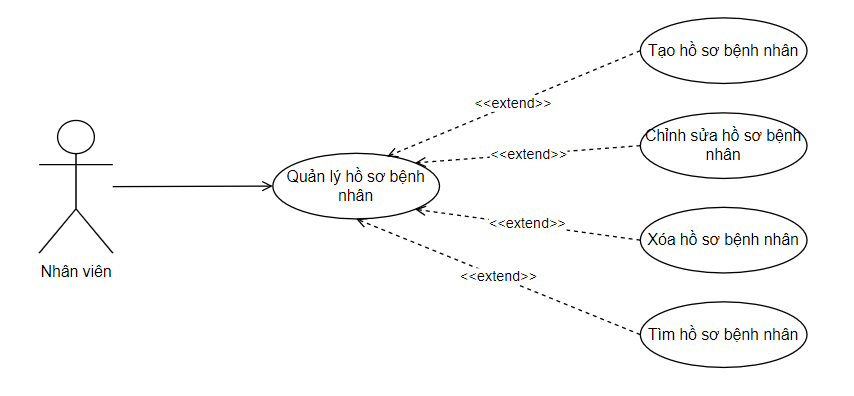
* Sơ đồ usecase quản lý phiếu dịch vụ:



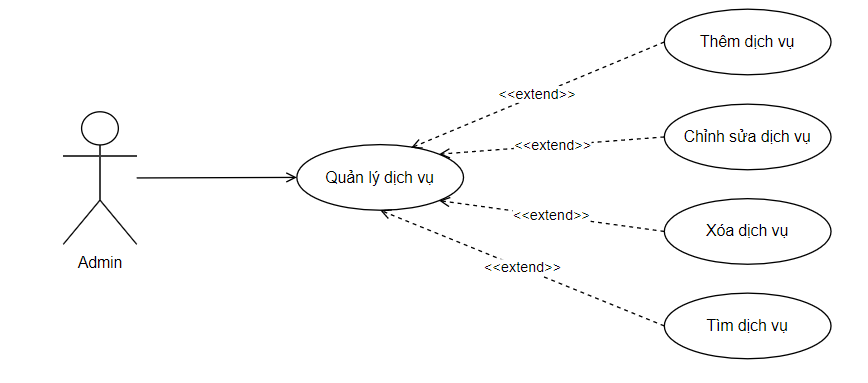
* Sơ đồ usecase quản lý phiếu khám:



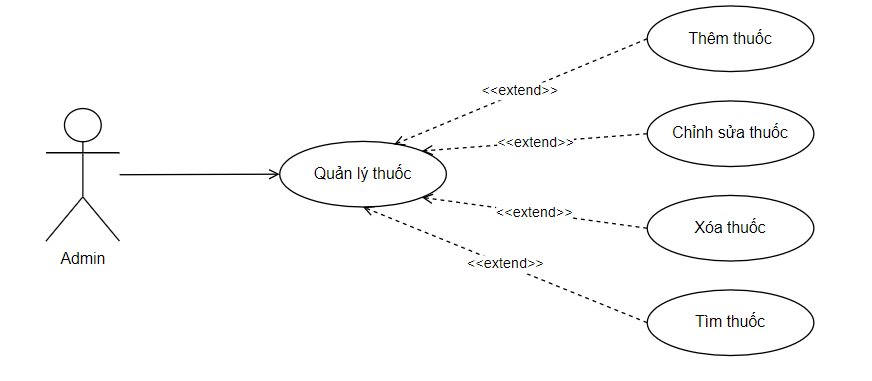
* Sơ đồ usecase quản lý hồ sơ bệnh nhân:



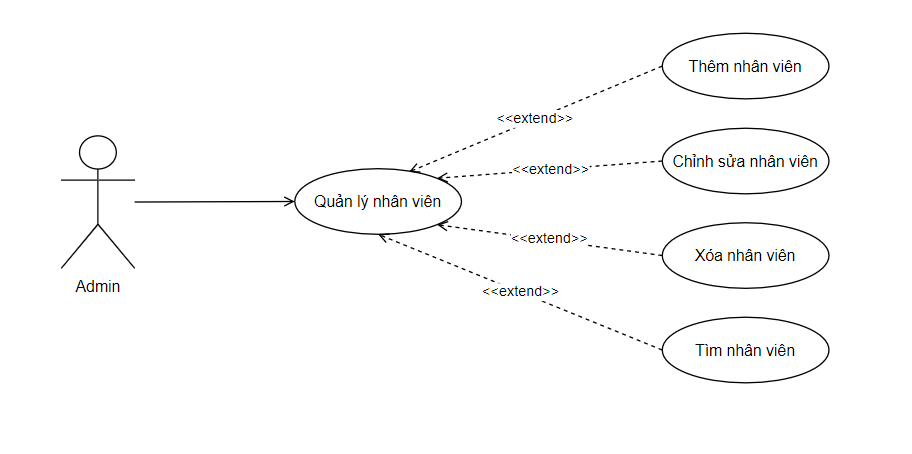
* Sơ đồ usecase quản lý dịch vụ:



* Sơ đồ usecase quản lý thuốc:

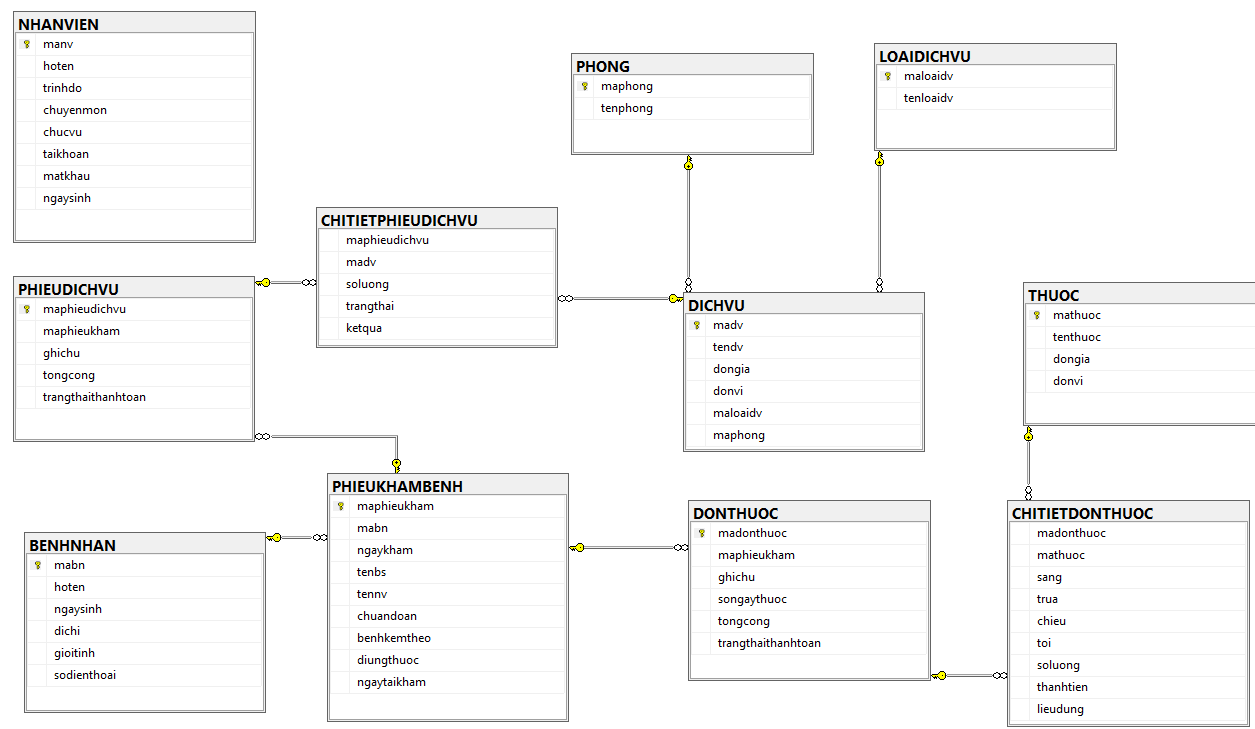


* Sơ đồ usecase quản lý nhân viên:



## Phân tích dữ liệu

### Mô hình thực thể kết hợp



### Mô hình quan hệ:

NHANVIEN (**manv**, hoten, trinhdo, chuyenmon, chucvu, taikhoan, matkhau, ngaysinh)

THUOC (**mathuoc**, tenthuoc, donvi, dongia)

DICHVU (**madv**, tendv, dongia, donvi, maloaidv, maphong)

LOAIDICHVU (**maloaidv**, tenloaidv)

PHONG (**maphong**, tenphong)

BENHNHAN (**mabn**, hoten, ngaysinh, diachi, gioitinh, sodienthoai)

PHIEUKHAMBENH (**maphieukham**, mabn, ngaykham, tenbs, tennv, chuandoan, benhkemtheo, diungthuoc, ngaytaikham)

DONTHUOC (**madonthuoc**, maphieukham, ghichu, songaythuoc, tongcong, trangthaithanhtoan)

CHITIETDONTHUOC (**madonthuoc**, mathuoc, sang, trua, chieu, toi, luong, thanhtien, lieudung)

PHIEUDICHVU (**maphieudichvu**, maphieukham, ghichu, tongcong, trangthaithanhtoan)

CHITIETPHIEUDICHVU (**maphieudichvu**, madv, soluong, trangthai, ketqua)

#### NHANVIEN

|  |  |  |  |
| --- | --- | --- | --- |
| **NHANVIEN** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| manv | int | PRIMARY KEY | Mã nhân viên |
| hoten | nvarchar(50) |  | Họ tên |
| trinhdo | nvarchar(50) |  | Trình độ |
| chuyenmon | nvarchar(50) |  | Chuyên môn |
| chucvu | nvarchar(50) |  | Chức vụ |
| taikhoan | char(20) |  | Tài khoản |
| matkhau | char(10) |  | Mật khẩu |
| ngaysinh | date |  | Ngày sinh |

#### BENHNHAN

|  |  |  |  |
| --- | --- | --- | --- |
| **BENHNHAN** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| mabn | int | PRIMARY KEY | Mã bệnh nhân |
| hoten | nvarchar(50) |  | Họ tên |
| ngaysinh | date |  | Ngày sinh |
| diachi | nvarchar(50) |  | Địa chỉ |
| gioitinh | bit |  | Giới tính |
| sodienthoai | char(10) |  | Số điện thoại |

#### LOAIDICHVU

|  |  |  |  |
| --- | --- | --- | --- |
| **LOAIDICHVU** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| maloaidv | int | PRIMARY KEY | Mã loại dịch vụ |
| tenloaidv | nvarchar(50) |  | Tên loại dịch vụ |

#### PHONG

|  |  |  |  |
| --- | --- | --- | --- |
| **PHONG** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| maphong | int | PRIMARY KEY | Mã phòng |
| tenphong | nvarchar(50) |  | Tên phòng |

#### DICHVU

|  |  |  |  |
| --- | --- | --- | --- |
| **DICHVU** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| madv | int | PRIMARY KEY | Mã dịch vụ |
| tendv | nvarchar(50) |  | Tên dịch vụ |
| dongia | int |  | Đơn giá |
| donvi | nvarchar(10) |  | Đơn vị tính |
| maloaidv | int | FOREIGN KEY | Mã loại dịch vụ |
| maphong | int | FOREIGN KEY | Mã phòng |

#### THUOC

|  |  |  |  |
| --- | --- | --- | --- |
| **THUOC** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| mathuoc | int | PRIMARY KEY | Mã thuốc |
| tenthuoc | nvarchar2(50) |  | Tên thuốc |
| dongia | int |  | Đơn giá |
| donvi | char(10) |  | Đơn vị tính |

#### PHIEUKHAMBENH

|  |  |  |  |
| --- | --- | --- | --- |
| **PHIEUKHAMBENH** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| maphieukham | int | PRIMARY KEY | Mã phiếu khám |
| mabn | int | FOREIGN KEY | Mã bệnh nhân |
| ngaykham | date |  | Ngày khám |
| tenbs | nvarchar(50) |  | Tên bác sĩ |
| tennv | nvarchar(50) |  | Tên nhân viên |
| chuandoan | nvarchar(50) |  | Chuẩn đoán |
| benhkemtheo | nvarchar(50) |  | Bệnh kèm theo |
| diungthuoc | nvarchar(50) |  | Dị ứng thuốc |
| ngaytaikham | nvarchar(50) |  | Ngày tái khám |

#### PHIEUDICHVU

|  |  |  |  |
| --- | --- | --- | --- |
| **PHIEUDICHVU** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| maphieudichvu | int | PRIMARY KEY | Mã phiếu dịch vụ |
| maphieukham | int | FOREIGN KEY | Mã phiếu khám |
| ghichu | nvarchar(50) |  | Ghi chú |
| tongcong | int |  | Tổng cộng |
| trangthaithanhtoan | bit |  | Trạng thái thanh toán |

#### CHITIETPHIEUDICHVU

|  |  |  |  |
| --- | --- | --- | --- |
| **CHITIETPHIEUDICHVU** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| maphieudichvu | int | PRIMARY KEY | Mã phiếu dịch vụ |
| madichvu | int | FOREIGN KEY | Mã dịch vụ |
| soluong | nvarchar(50) |  | Số lượng |
| trangthai | bit |  | Trạng thái |
| ketqua | nvarchar(50) |  | Kết quả |

#### DONTHUOC

|  |  |  |  |
| --- | --- | --- | --- |
| **DONTHUOC** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| madonthuoc | int | FOREIGN KEY | Mã đơn thuốc |
| maphieukham | int | FOREIGN KEY | Mã phiếu khám |
| ghichu | nvarchar(50) |  | Ghi chú |
| tongcong | int |  | Tổng cộng |
| trangthaithanhtoan | bit |  | Trạng thái thanh toán |

#### CHITIETDONTHUOC

|  |  |  |  |
| --- | --- | --- | --- |
| **CHITIETDONTHUOC** | | | |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| madonthuoc | int | FOREIGN KEY | Mã đơn thuốc |
| mathuoc | int | FOREIGN KEY | Mã thuoc |
| soluong | nvarchar(50) |  | Số lượng |
| lieudung | nvarchar(50) |  | Liều dùng |

## Ràng buộc trọn vẹn

### **TaiKhoan** không được trùng

* Bối cảnh: NhanVien
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **NhanVien** | + | - | + (taikhoan) |

### Tên thuốc không được trùng

* Bối cảnh: Thuoc
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **Thuoc** | + | - | + (tenthuoc) |

### Nhân viên không được trùng tên và ngày sinh

* Bối cảnh: NhanVien
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **NhanVien** | + | - | + (hoten, ngaysinh) |

### Bệnh nhân không được trùng

* Bối cảnh: BenhNhan
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **BenhNhan** | + | - | + (hoten, ngaysinh, gioitinh, sodienthoai) |

### Ngày sinh bệnh nhân không được sau ngày hiện tại

* Bối cảnh: BenhNhan
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **BenhNhan** | + | - | + (ngaysinh) |

### Ngày sinh nhân viên không được sau ngày hiện tại

* Bối cảnh: NhanVien
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **NhanVien** | + | - | + (ngaysinh) |

### Kiểm tra giá trị giới tính bệnh nhân, chỉ được nhập “Nam”, “Nữ”, “Khác”

* Bối cảnh: BenhNhan
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **BenhNhan** | + | - | + (gioitinh) |

### Số lượng trong bảng ChiTietDonThuoc phải lớn hơn 0

* Bối cảnh: ChiTietDonThuoc
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **ChiTietDonThuoc** | + | - | + (soluong) |

### Kiểm tra trạng thái thanh toán đơn thuốc:

* Bối cảnh: DonThuoc
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **DonThuoc** | + | - | + (trangthai) |

### Kiểm tra trạng thái thanh toán đơn thuốc:

* Bối cảnh: PhieuDichVu
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **PhieuDichVu** | + | - | + (trangthai) |

### Kiểm tra ngày tái khám

* Bối cảnh: PhieuKhamBenh
* Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Thêm** | **Xóa** | **Sửa** |
| **PhieuKhamBenh** | + | - | + (ngaytaikham) |

# Chương 4: THIẾT KẾ XỬ LÝ TRONG CLIENT/SERVER

## Giao tác



### Thêm hồ sơ bệnh nhân

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tạo tài khoản và truyền tham số |  |
|  | Tạo hồ sơ bệnh nhân   * Đọc BENHNHAN để tạo **mabn** * Thêm bệnh nhân vào bảng BENHNHAN * Tạo user trong database * Thêm user vào Role thích hợp |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Xóa hồ sơ bệnh nhân

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server xóa hồ sơ bệnh nhân và truyền tham số |  |
|  | Xóa hồ sơ bệnh nhân   * Xóa bệnh nhân khỏi bảng BENHNHAN * Xóa user trong database |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |



### Cập nhật hồ sơ bệnh nhân

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server cập nhật hồ sơ bệnh nhân và truyền tham số |  |
|  | Kiểm tra mã bệnh nhân   * Hiển thị thông tin hồ sơ BENHNHAN |
| Nhận thông tin và yêu cầu cập hồ sơ |  |
|  | Kiểm tra thông tin   * Cập nhật lại bảng BENHNHAN |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Tìm kiếm hồ sơ bệnh nhân

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tìm hồ sơ bệnh nhân và truyền tham số |  |
|  | Kiểm tra tên bệnh nhân   * Hiển thị danh sách kết quả |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Thêm phiếu khám bệnh

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server tạo phiếu khám bệnh và truyền tham số |  |
|  | Tạo phiếu khám bệnh   * Đọc PHIEUKHAM để tạo **maphieukham** * Thêm phiếu khám bệnh vào bảng PHIEUKHAM |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Cập nhật phiếu khám bệnh

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server cập nhật phiếu khám bệnh và truyền tham số |  |
|  | Kiểm tra mã phiếu khám   * Hiển thị thông tin hồ sơ PHIEUKHAM |
| Nhận thông tin và yêu cầu cập nhật phiếu khám |  |
|  | Kiểm tra thông tin   * Cập nhật lại bảng PHIEUKHAM |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Thêm chi tiết dịch vụ:

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server thêm chi tiết dịch vụ bệnh và truyền tham số |  |
|  | Kiểm tra mã phiếu khám   * Hiển thị thông tin hồ sơ PHIEUKHAM |
| Nhận thông tin và yêu cầu thêm chi tiết dịch vụ |  |
|  | Thêm vào bảng CHITIETDICHVU   * Insert vào bảng CHITIETDICHVU |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Xóa chi tiết dịch vụ

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server xóa chi tiết dịch vụ và truyền mã phiếu khám |  |
|  | Xóa các chi tiết dịch vụ   * Xóa danh sách chi tiết dịch vụ của mã phiếu khám |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Thêm chi tiết đơn thuốc:

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server thêm chi tiết đơn thuốc và truyền tham số |  |
|  | Kiểm tra mã phiếu khám   * Hiển thị thông tin hồ sơ PHIEUKHAM |
| Nhận thông tin và yêu cầu thêm chi tiết đơn thuốc |  |
|  | Thêm vào bảng CHITIETDONTHUOC   * Insert vào bảng CHITIETDONTHUOC |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

### Xóa chi tiết đơn thuốc

|  |  |
| --- | --- |
| Client | Server |
| Yêu cầu server xóa chi tiết đơn thuốc và truyền mã phiếu khám |  |
|  | Xóa các chi tiết dịch vụ   * Xóa danh sách chi tiết đơn thuốc của mã phiếu khám |
| Nhận kết quả giao tác |  |
| Thông báo kết quả giao tác cho người dùng |  |

## Phân quyền:



### Các đối tượng phân quyền:

* Nhân viên: các nhân viên tiếp nhận bệnh nhân
* Bác sĩ: các nhân viên cập nhật phiếu khám bệnh
* Admin: nhân viên quản trị hệ thống

### Phân quyền trên các bảng:

***Ghi chú:* S – Select, I – Insert, U – Update, D – Delete**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Nhân viên** | **Bác sĩ** | **Admin** | **Kế toán** |
| NHANVIEN |  |  | S,I,U,D |  |
| BENHNHAN | S,I,U,D | S |  |  |
| THUOC | S | S | S,I,U,D |  |
| DICHVU | S | S | S,I,U,D |  |
| PHIEUKHAM | S,I,U,D | S,U |  |  |
| PHIEUDICHVU | S,U | S,I,U,D |  | S,U |
| CHITIETPHIEUDICHVU | S | S,I,U,D |  | S |
| DONTHUOC | S,U | S,I,U,D |  | S,U |
| CHITIETDONTHUOC | S | S,I,U,D |  | S |

### Phân quyền trên các giao tác:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Nhân viên** | **Bác sĩ** | **Admin** |
| Thêm hồ sơ bệnh nhân | X |  |  |
| Xóa hồ sơ bệnh nhân | X |  |  |
| Cập nhật hồ sơ bệnh nhân | X |  |  |
| Thêm phiếu khám bệnh | X |  |  |
| Chỉnh sửa phiếu khám bệnh | X |  |  |
| Xóa phiếu khám bệnh | X |  |  |
| Cập nhật quá trình khám bệnh |  | X |  |
| Thêm chi tiết dịch vụ |  | X |  |
| Xóa chi tiết dịch vụ |  | X |  |
| Thêm chi tiết đơn thuốc |  | X |  |
| Xóa chi tiết đơn thuốc |  | X |  |
| Thêm dịch vụ |  |  | X |
| Xóa dịch vụ |  |  | X |
| Cập nhật dịch vụ |  |  | X |
| Thêm thuốc |  |  | X |
| Xóa thuốc |  |  | X |
| Cập nhật thuốc |  |  | X |
| Thêm nhân viên |  |  | X |
| Xóa nhân viên |  |  | X |
| Cập nhật nhân viên |  |  | X |

# Chương 5: TRIỂN KHAI THỰC NGHIỆM

## Phía Server

### Chức năng

#### Procedure Kiểm tra đăng nhập:

CREATE PROC [dbo].[sp\_KiemTraDangNhap] @username char(50),

@pass char(20)

AS

BEGIN

if(not exists( Select \* From NHANVIEN where taikhoan=@username))

raiserror(N'Người dùng không tồn tại!!!',16,1)

else if(( Select matkhau From NHANVIEN where taikhoan=@username)!=@pass)

raiserror(N'Sai mật khẩu!!!',16,1)

else

Select \* From NHANVIEN where taikhoan=@username and matkhau=@pass

END

#### Procedure Xem danh sách hồ sơ bệnh nhân:

Create proc [dbo].[sp\_DanhSachHoSoBenhNhan]

as

begin

select mabn as N'Mã bệnh nhân',

hoten as N'Họ và tên',

ngaysinh as N'Ngày sinh',

dichi as N'Địa chỉ',

gioitinh as N'Giới tính',

sodienthoai as N'Số điện thoại'

from BENHNHAN

end

#### Procedure Thêm bệnh nhân:

Create proc [dbo].[sp\_ThemBenhNhan] @hoten nvarchar(50),

@ngaysinh date,

@diachi nvarchar(50),

@gioitinh nvarchar(3),

@sodienthoai char(10)

as

begin

declare @dem int

set @dem=1

begin transaction

begin try

while( exists (select \* from BENHNHAN with (updlock) where mabn=@dem))

set @dem=@dem+1

insert into BENHNHAN

values(@dem,@hoten,@ngaysinh,@diachi,@gioitinh,@sodienthoai)

commit

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

#### Procedure Xóa hồ sơ bệnh nhân:

Create proc [dbo].[sp\_XoaBenhNhan] @mabn int

as

begin

if(not exists(select \* from BENHNHAN where mabn=@mabn))

Raiserror('Bệnh nhân không tồn tại',16,1)

else

begin

if(exists (select \* from PHIEUKHAMBENH where mabn=@mabn))

begin

delete from PHIEUKHAMBENH where mabn=@mabn

end

Delete from BENHNHAN where mabn=@mabn

end

end

#### Procedure Tìm kiếm hồ sơ bệnh nhân:

Create proc [dbo].[sp\_TimBenhNhan] @hoten nvarchar(50)

as

begin

select mabn as N'Mã bệnh nhân',

hoten as N'Họ và tên',

ngaysinh as N'Ngày sinh',

dichi as N'Địa chỉ',

gioitinh as N'Giới tính',

sodienthoai as N'Số điện thoại'

from BENHNHAN

where hoten like '%'+@hoten+'%'

end

#### Procedure Chỉnh sửa hồ sơ bênh nhân:

Create proc [dbo].[sp\_ChinhSuaBenhNhan] @mabn int,

@hoten nvarchar(50),

@ngaysinh date,

@diachi nvarchar(50),

@gioitinh nvarchar(3),

@sodienthoai char(10)

as

begin

if not exists (select \* from BENHNHAN where mabn=@mabn)

begin

raiserror(N'Bệnh nhân không tồn tại!!!',16,1)

return

end

else

begin

begin transaction

begin try

update BENHNHAN

set hoten =@hoten, ngaysinh=@ngaysinh,dichi=@diachi,gioitinh=@gioitinh,sodienthoai=@sodienthoai

where mabn=@mabn

commit

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

end

#### Procedure Tạo phiếu khám:

Create proc [dbo].[sp\_TaoPhieuKhamBenh] @hoten nvarchar(50),

@ngaysinh date,

@diachi nvarchar(50),

@gioitinh nvarchar(3),

@sodienthoai char(10),

@bacsi nvarchar(50),

@nhanvien nvarchar(50)

as

begin transaction

declare @mapkb int

declare @mabn int

set @mapkb=1

if not exists (select \* from BENHNHAN where hoten=@hoten and ngaysinh=@ngaysinh and gioitinh=@gioitinh and sodienthoai=@sodienthoai)

begin

exec sp\_ThemBenhNhan @hoten, @ngaysinh, @diachi, @gioitinh, @sodienthoai

end

else

begin try

set @mabn= (select mabn from BENHNHAN where hoten=@hoten and ngaysinh=@ngaysinh and gioitinh=@gioitinh and sodienthoai=@sodienthoai)

while( exists (select \* from PHIEUKHAMBENH with (updlock) where maphieukham=@mapkb))

set @mapkb=@mapkb+1

insert into PHIEUKHAMBENH

values(@mapkb,@mabn,GETDATE(),@bacsi,@nhanvien,'','','','')

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

commit

#### Procedure Xem danh sách phiếu khám bệnh:

Create proc [dbo].[sp\_DanhSachPhieuKham]

as

begin

select maphieukham as N'Mã phiếu khám',

ngaykham as N'Ngày khám',

hoten as N'Họ và tên',

ngaysinh as N'Ngày sinh',

dichi as N'Địa chỉ',

gioitinh as N'Giới tính',

tenbs as N'Bác sĩ',

tennv as N'Nhân viên'

from PHIEUKHAMBENH join BENHNHAN on PHIEUKHAMBENH.mabn=BENHNHAN.mabn

end

#### Procedure Xem danh sách phiếu khám bệnh theo ngày:

Create proc [dbo].[sp\_DanhSachPhieuKhamTheoNgay] @datefrom date,

@dateto date

as

begin

if(@datefrom>@dateto)

begin

Raiserror('Ngày bắt đầu phải trước hoặc cùng ngày với ngày kết thúc!!!',16,1)

return

end

else

begin

select maphieukham as N'Mã phiếu khám',

ngaykham as N'Ngày khám',

hoten as N'Họ và tên',

ngaysinh as N'Ngày sinh',

dichi as N'Địa chỉ',

gioitinh as N'Giới tính',

tenbs as N'Bác sĩ',

tennv as N'Nhân viên'

from PHIEUKHAMBENH join BENHNHAN on PHIEUKHAMBENH.mabn=BENHNHAN.mabn

where ngaykham>=@datefrom and ngaykham<=@dateto

end

end

#### Procedure Lấy danh sách bác sĩ:

Create PROC [dbo].[sp\_LayDanhSachTenBacSi]

as

begin

select hoten +' - BS.'+chuyenmon

from NHANVIEN

where chucvu=N'Bác Sĩ'

end

#### Procedure Lấy danh sách nhân viên:

CREATE PROC [dbo].[sp\_LayDanhSachTenNhanVien]

as

begin

select hoten +' - '+chucvu

from NHANVIEN

where chucvu=N'Nhân viên tiếp nhận'

end

#### Procedure Xóa phiếu khám bệnh:

CREATE proc [dbo].[sp\_XoaPhieuKhamBenh] @mapk int

as

begin

if(exists(select \* from PHIEUKHAMBENH where maphieukham=@mapk))

Delete from PHIEUKHAMBENH where maphieukham=@mapk

else

Raiserror('Phiếu khám không tồn tại',16,1)

end

#### Procedure Chỉnh sửa phiếu khám bệnh:

CREATE proc [dbo].[sp\_ChinhSuaPhieuKhamBenh] @maphieukham int,

@mabn int,

@hoten nvarchar(50),

@ngaysinh date,

@diachi nvarchar(50),

@gioitinh nvarchar(3),

@sodienthoai char(10),

@bacsi nvarchar(50),

@nhanvien nvarchar(50)

as

begin

if not exists (select \* from PHIEUKHAMBENH where maphieukham=@maphieukham)

begin

raiserror(N'Phiếu khám không tồn tại!!!',16,1)

return

end

else

begin

begin transaction

begin try

exec sp\_ChinhSuaBenhNhan @mabn,@hoten,@ngaysinh,@diachi,@gioitinh,sodienthoai

UPDATE PHIEUKHAMBENH

SET tenbs=@bacsi,tennv =@nhanvien

where maphieukham=@maphieukham

commit

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

end

#### Procedure Cập nhật quá trình khám bệnh:

CREATE proc [dbo].[sp\_CapNhatPhieuKhamBenh] @maphieukham int,

@chuandoan nvarchar(50),

@benhkemtheo nvarchar(50),

@diungthuoc nvarchar(50),

@ngaytaikham date

as

begin

if not exists (select \* from PHIEUKHAMBENH where maphieukham=@maphieukham)

begin

raiserror(N'Phiếu khám không tồn tại!!!',16,1)

return

end

else

begin

begin transaction

begin try

update PHIEUKHAMBENH set chuandoan=@chuandoan, benhkemtheo=@benhkemtheo,diungthuoc=@diungthuoc,ngaytaikham=@ngaytaikham where maphieukham=@maphieukham

commit

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

end

#### Procedure Lấy danh sách dịch vụ:

CREATE PROCEDURE [dbo].[sp\_DanhSachDichVu]

AS

BEGIN

select madv as N'Mã dịch vụ',

tendv as N'Tên dịch vụ',

dongia as N'Đơn giá',

donvi as N'Đơn vị',

tenloaidv as N'Loại dịch vụ',

tenphong N'Tên phòng'

from DICHVU Join LOAIDICHVU on DICHVU.maloaidv=LOAIDICHVU.maloaidv

join PHONG on DICHVU.maphong=PHONG.maphong

END

#### Procedure Lấy danh sách phiếu khám:

CREATE proc [dbo].[sp\_LayDanhSachPhieuKham]

as

begin

select \*

from PHIEUKHAMBENH join BENHNHAN on PHIEUKHAMBENH.mabn=BENHNHAN.mabn

end

#### Procedure Thêm chi tiết dịch vụ:

CREATE proc [dbo].[sp\_ThemChiTietDichVu] @maphieukham int,

@madichvu int,

@soluong int

as

begin

if not exists (select \* from PHIEUDICHVU where maphieudichvu=@maphieukham)

begin

INSERT INTO PHIEUDICHVU VALUES (@maphieukham,@maphieukham,'',0,'N')

end

begin

begin transaction

begin try

if not exists (select \* from CHITIETPHIEUDICHVU where maphieudichvu=@maphieukham and madv=@madichvu)

INSERT INTO CHITIETPHIEUDICHVU VALUES(@maphieukham,@madichvu,@soluong,N'Chưa thực hiện','')

else

UPDATE CHITIETPHIEUDICHVU SET soluong= @soluong where maphieudichvu=@maphieukham and madv=@madichvu

declare @tongtien int

set @tongtien=(select SUM(soluong\*dongia) from CHITIETPHIEUDICHVU join DICHVU on CHITIETPHIEUDICHVU.madv=DICHVU.madv where DICHVU.madv=@madichvu)

UPDATE PHIEUDICHVU SET tongcong=@tongtien

commit

end try

begin catch

DECLARE @ErrMsg VARCHAR(max)

ROLLBACK

SELECT @ErrMsg='Lỗi'+ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

end

#### Procedure Xóa chi tiết dịch vụ:

CREATE proc [dbo].[sp\_XoaChiTietDichVu] @maphieukham int

as

begin

delete from CHITIETPHIEUDICHVU where maphieudichvu=@maphieukham

end

#### Procedure Xem chi tiết dịch vụ theo mã phiếu khám

CREATE PROCEDURE [dbo].[sp\_DanhSachChiTietDichVu] @maphieudv int

AS

BEGIN

select madv,soluong,trangthai,ketqua

from CHITIETPHIEUDICHVU where maphieudichvu=@maphieudv

END

#### Procedure Lấy danh sách thuốc:

CREATE PROCEDURE [dbo].[sp\_DanhSachThuoc]

AS

BEGIN

select mathuoc as N'Mã thuốc',tenthuoc as N'Tên thuốc',donvi as N'Đơn vị',dongia as N'Đơn giá'

from THUOC

END

#### Procedure Thêm chi tiết đơn thuốc:

CREATE proc [dbo].[sp\_ThemChiTietDonThuoc] @maphieukham int,

@mathuoc int,

@sang int,

@trua int,

@chieu int,

@toi int,

@soluong int,

@lieudung nvarchar(50)

as

begin

if not exists (select \* from DONTHUOC where madonthuoc=@maphieukham)

begin

INSERT INTO DONTHUOC VALUES (@maphieukham,@maphieukham,'',0,0,'N')

end

begin

begin transaction

begin try

DECLARE @thanhtien FLOAT

SET @thanhtien=(SELECT dongia FROM THUOC where mathuoc=@mathuoc)\*@soluong\*1.0

if not exists (select \* from CHITIETDONTHUOC where madonthuoc=@maphieukham and mathuoc=@mathuoc)

INSERT INTO CHITIETDONTHUOC VALUES(@maphieukham,@mathuoc,@sang,@trua,@chieu,@toi,@soluong,@thanhtien,@lieudung)

else

UPDATE CHITIETDONTHUOC SET soluong= @soluong, sang=@sang,trua=@trua,chieu=@chieu,toi=@toi,lieudung=@lieudung,thanhtien=@thanhtien where madonthuoc=@maphieukham and mathuoc=@mathuoc

UPDATE DONTHUOC

set tongcong =(select(select SUM(thanhtien) from CHITIETDONTHUOC where madonthuoc=@maphieukham)\* 2 from DONTHUOC where madonthuoc=@maphieukham) where madonthuoc=@maphieukham

commit

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

end

#### Procedure Cập nhật số lượng ngày thuốc:

CREATE proc [dbo].[sp\_CapNhatSoNgayThuoc] @maphieukham int,

@soluong int

as

begin

UPDATE DONTHUOC SET songaythuoc=@soluong where madonthuoc=@maphieukham

end

#### Procedure Xóa chi tiết đơn thuốc theo mã đơn thuốc:

CREATE proc [dbo].[sp\_XoaChiTietDonThuoc] @maphieukham int

as

begin

delete from CHITIETDONTHUOC where madonthuoc=@maphieukham

end

#### Procedure Xem chi tiết đơn thuốc theo mã đơn thuốc:

CREATE PROCEDURE [dbo].[sp\_DanhSachChiTietDonThuoc] @madonthuoc int

AS

BEGIN

select madonthuoc,mathuoc,sang,trua,chieu,toi,soluong,lieudung

from CHITIETDONTHUOC where madonthuoc=@madonthuoc

END

#### Procedure Xóa dịch vụ theo mã dịch vụ:

CREATE proc [dbo].[sp\_XoaDichVu] @madv int

as

begin

delete from DICHVU where madv=@madv

end

#### Procedure Tìm kiếm dịch vụ:

CREATE PROCEDURE [dbo].[sp\_TimKiemDichVu] @ten nvarchar(50)

AS

BEGIN

select madv as N'Mã dịch vụ',

tendv as N'Tên dịch vụ',

dongia as N'Đơn giá',

donvi as N'Đơn vị',

tenloaidv as N'Tên loại dịch vụ',

tenphong N'Tên phòng'

from DICHVU Join LOAIDICHVU on DICHVU.maloaidv=LOAIDICHVU.maloaidv

join PHONG on DICHVU.maphong=PHONG.maphong

where tendv Like '%'+@ten+'%'

END

#### Procedure Cập nhật dịch vụ:

CREATE proc [dbo].[sp\_CapNhatDichVu] @madv int,

@ten nvarchar(50),

@dongia int,

@donvi nvarchar(10),

@maloai int,

@maphong int

as

begin

if not exists (select \* from DICHVU where madv=@madv)

begin

raiserror(N'Dịch vụ không tồn tại!!!',16,1)

return

end

else

begin try

UPDATE DICHVU SET tendv=@ten,dongia=@dongia,donvi=@donvi,maloaidv=@maloai,maphong=@maphong where madv=@madv

end try

begin catch

DECLARE @ErrMsg VARCHAR(max)

ROLLBACK

SELECT @ErrMsg='Lỗi'+ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

#### Procedure Thêm thuốc:

CREATE proc [dbo].[sp\_ThemThuoc] @ten nvarchar(50),

@dongia int,

@donvi nvarchar(10)

as

begin

declare @dem int

set @dem=1

if exists (select \* from THUOC where tenthuoc=@ten and donvi=@donvi)

begin

raiserror(N'Thuốc đã tồn tại!!!',16,1)

return

end

else

begin transaction

begin try

while( exists (select \* from THUOC with (updlock) where mathuoc=@dem))

set @dem=@dem+1

insert into THUOC

values(@dem,@ten,@dongia,@donvi)

commit

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

#### Procedure Xóa thuốc:

CREATE proc [dbo].[sp\_XoaThuoc] @mathuoc int

as

begin

begin try

delete from THUOC where mathuoc=@mathuoc

end try

begin catch

RAISERROR (N'Không thể xóa thuốc này!!!',16,1)

end catch

end

#### Procedure Tìm thuốc:

CREATE PROCEDURE [dbo].[sp\_TimKiemThuoc] @ten nvarchar(50)

AS

BEGIN

select mathuoc as N'Mã thuốc',tenthuoc as N'Tên thuốc',donvi as N'Đơn vị',dongia as N'Đơn giá'

from THUOC

where tenthuoc Like '%'+@ten+'%'

END

#### Procedure Cập nhật thuốc:

CREATE proc [dbo].[sp\_ChinhSuaThuoc] @mathuoc int,

@ten nvarchar(50),

@dongia int,

@donvi nvarchar(10)

as

begin

if not exists (select \* from THUOC where mathuoc=@mathuoc)

begin

raiserror(N'Thuốc không không tồn tại!!!',16,1)

return

end

else

begin

begin try

update THUOC

set tenthuoc =@ten, dongia=@dongia,donvi=@donvi

where mathuoc=@mathuoc

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

end

#### Procedure Xem danh sách nhân viên:

CREATE PROCEDURE [dbo].[sp\_DanhSachNhanVien]

AS

BEGIN

select manv as N'Mã nhân viên',hoten as N'Họ tên',chucvu as N'Chức vụ',chuyenmon as N'Chuyên môn',trinhdo as N'Trình độ',taikhoan as N'Tài khoản',ngaysinh as N'Ngày sinh'

from NHANVIEN

END

#### Procedure Thêm nhân viên:

CREATE proc [dbo].[sp\_ThemNhanVien] @ten nvarchar(50),

@trinhdo nvarchar(50),

@chuyenmon nvarchar(50),

@chucvu nvarchar(50),

@taikhoan char(50),

@ngaysinh date

as

begin

declare @dem int

set @dem=1

begin transaction

begin try

while( exists (select \* from NHANVIEN with (updlock) where manv=@dem))

set @dem=@dem+1

declare @matkhau char(10)

set @matkhau=(select RTRIM(CONVERT(char(2),DAY(@ngaysinh)))+RTRIM(CONVERT(char(2),month(@ngaysinh)))+CONVERT(char(4),year(@ngaysinh)))

insert into NHANVIEN

values(@dem,@ten,@trinhdo,@chuyenmon,@chucvu,@taikhoan,@matkhau,@ngaysinh)

commit

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE()

RAISERROR (@ErrMsg,16,1)

end catch

end

#### Procedure Xóa nhân viên theo mã nhân viên:

CREATE proc [dbo].[sp\_XoaNhanVien] @manv int

as

begin

begin try

delete from NHANVIEN where manv=@manv

end try

begin catch

RAISERROR (N'Không thể xóa nhân viên này!!!',16,1)

end catch

end

#### Procedure Cập nhật nhân viên:

CREATE proc [dbo].[sp\_ChinhSuaNhanVien] @manv int,

@ten nvarchar(50),

@trinhdo nvarchar(50),

@chuyenmon nvarchar(50),

@chucvu nvarchar(50),

@taikhoan char(50),

@ngaysinh date

as

begin

if not exists (select \* from NHANVIEN where manv=@manv)

begin

raiserror(N'Nhân viên không không tồn tại!!!',16,1)

return

end

else

begin

begin transaction

begin try

update NHANVIEN

set hoten =@ten, trinhdo=@trinhdo,chuyenmon=@chuyenmon,chucvu=@chucvu,taikhoan=@taikhoan,ngaysinh=@ngaysinh

where manv=@manv

commit

end try

begin catch

DECLARE @ErrMsg NVARCHAR(max)

ROLLBACK

SELECT @ErrMsg=ERROR\_MESSAGE();

RAISERROR (@ErrMsg,16,1)

end catch

end

end



### Ràng buộc

#### Trigger kiểm tra tài khoản nhân viên

Create TRIGGER [dbo].[trg\_TaiKhoanNhanVien]

ON [dbo].[NHANVIEN]

FOR INSERT, UPDATE

AS

BEGIN

DECLARE @TaiKhoan nvarchar(100)

SELECT @TaiKhoan = taikhoan FROM inserted

IF (SELECT COUNT(\*) FROM NHANVIEN WHERE taikhoan = @TaiKhoan) > 1

BEGIN

RAISERROR(N'Tài khoản đã tồn tại.', 16, 1)

ROLLBACK

END

END

#### Trigger kiểm tra tên thuốc

Create TRIGGER [dbo].[trg\_TenThuoc]

ON [dbo].THUOC

FOR INSERT

AS

BEGIN

DECLARE @Tenthuoc nvarchar(100)

SELECT @Tenthuoc = tenthuoc FROM inserted

IF (SELECT COUNT(\*) FROM THUOC WHERE tenthuoc = @Tenthuoc) > 1

BEGIN

RAISERROR(N'tên thuốc đã tồn tại.', 16, 1)

ROLLBACK

END

END

#### Trigger kiểm tra nhập nhân viên:

Create TRIGGER [dbo].[trg\_KiemTraNhanVien]

ON [dbo].NHANVIEN

FOR INSERT

AS

BEGIN

DECLARE @ten nvarchar(50),@ngaysinh date

SELECT @ten =hoten,@ngaysinh=ngaysinh FROM inserted

IF (SELECT COUNT(\*) from NHANVIEN where hoten=@ten and @ngaysinh=@ngaysinh) > 1

begin

raiserror(N'Nhân viên đã đã tồn tại!!!',16,1)

return

end

END

#### Trigger kiểm tra bệnh nhân:

create TRIGGER [dbo].[trg\_KiemTraBenhNhan]

ON [dbo].BENHNHAN

FOR INSERT

AS

BEGIN

DECLARE @hoten nvarchar(50),@ngaysinh date,@gioitinh nvarchar(3),@sodienthoai char(10)

SELECT @hoten =hoten,@ngaysinh=ngaysinh,@gioitinh=gioitinh,@sodienthoai=sodienthoai FROM inserted

IF (SELECT COUNT(\*) from BENHNHAN where hoten=@hoten and ngaysinh=@ngaysinh and gioitinh=@gioitinh and sodienthoai=@sodienthoai) > 1

begin

raiserror(N'Bệnh nhân đã được tạo!!!',16,1)

return

end

END

#### Trigger kiểm tra ngày sinh bệnh nhân:

create TRIGGER [dbo].[trg\_KiemTraNgaySinhBenhNhan]

ON [dbo].BENHNHAN

FOR INSERT,UPDATE

AS

BEGIN

DECLARE @ngaysinh date

SELECT @ngaysinh=ngaysinh FROM inserted

if(@ngaysinh>GETDATE())

begin

raiserror(N'Ngày sinh không được lớn hơn ngày hiện tại',16,1)

return

end

END

#### Trigger kiểm tra ngày sinh nhân viên:

create TRIGGER [dbo].[trg\_KiemTraNgaySinhNhanVien]

ON [dbo].NHANVIEN

FOR INSERT,UPDATE

AS

BEGIN

DECLARE @ngaysinh date

SELECT @ngaysinh=ngaysinh FROM inserted

if(@ngaysinh>GETDATE())

begin

raiserror(N'Ngày sinh không được lớn hơn ngày hiện tại',16,1)

return

end

END

#### Trigger kiểm tra giới tính bệnh nhân:

create TRIGGER [dbo].[trg\_KiemTraGioiTinhBenhNhan]

ON [dbo].BENHNHAN

FOR INSERT,UPDATE

AS

BEGIN

    IF EXISTS (SELECT \* FROM inserted WHERE gioitinh != 'Nam' AND gioitinh != 'Nu' AND gioitinh != 'Khac')

    BEGIN

        RAISERROR (N'Vui lòng nhập lại, giới tính không hợp lệ', 16, 1)

        ROLLBACK

    END

END

#### Trigger kiểm tra số lượng chi tiết đơn thuốc:

create TRIGGER [dbo].[trg\_KiemTraSoLuongCTDonThuoc]

ON [dbo].ChiTietDonThuoc

FOR INSERT,UPDATE

AS

BEGIN

    IF EXISTS(SELECT \* FROM inserted WHERE soluong = 0)

    BEGIN

        RAISERROR(N'Số lượng phải lớn hơn 0', 16, 1)

        ROLLBACK

    END

END

#### Trigger kiểm tra trạng thái thanh toán đơn thuốc:

CREATE TRIGGER [dbo].[trg\_KiemTraTrangThaiThanhToan]

ON [dbo].DONTHUOC

FOR INSERT, UPDATE

AS

BEGIN

    IF EXISTS(SELECT \* FROM inserted WHERE trangthaithanhtoan != 'Y' AND trangthaithanhtoan != 'N')

    BEGIN

        RAISERROR(N'Trạng thái chỉ được Y hoặc N', 16, 1)

        ROLLBACK

    END

END

#### Trigger kiểm tra trạng thái thanh toán dịch vụ:

CREATE TRIGGER [dbo].[trg\_KiemTraTrangThaiThanhToanDV]

ON [dbo].PHIEUDICHVU

FOR INSERT, UPDATE

AS

BEGIN

    IF EXISTS(SELECT \* FROM inserted WHERE trangthaithanhtoan != 'Y' AND trangthaithanhtoan != 'N')

    BEGIN

        RAISERROR(N'Trạng thái chỉ được Y hoặc N', 16, 1)

        ROLLBACK

    END

END

#### Trigger kiểm tra ngày tái khám

create TRIGGER [dbo].[trg\_KiemTraNgayTaiKham]

ON [dbo].PHIEUKHAMBENH

FOR INSERT, UPDATE

AS

BEGIN

    DECLARE @mapk int

    SELECT @mapk = maphieukham FROM inserted

    IF (SELECT day(ngaytaikham) FROM PHIEUKHAMBENH WHERE maphieukham = @mapk) < GETDATE()

    BEGIN

        RAISERROR(N'Ngày tái khám phải sau ngày hôm nay.', 16, 1)

        ROLLBACK

    END

END

### Phân quyền

* Tạo tài khoản login

create login Hang with password ='123456789'

create login TranThiAnhThu with password ='123456789'

create login BuiThanhXuan with password ='123456789'

create login NguyenThiThuyHang with password ='123456789'

create login DoanNgocThanhVy with password ='123456789'

create login CaoNhatToan with password ='123456789'

create login LuongNgocHaiDang with password ='123456789'

* Tạo user cho từng login

create user Hang for login Hang

create user TranThiAnhThu for login TranThiAnhThu

create user BuiThanhXuan for login BuiThanhXuan

create user NguyenThiThuyHang for login NguyenThiThuyHang

create user DoanNgocThanhVy for login DoanNgocThanhVy

create user CaoNhatToan for login CaoNhatToan

create user LuongNgocHaiDang for login LuongNgocHaiDang

* Tạo role Admin, gán quyền và add user

create role Admin

grant select, insert, update, delete on NhanVien to Admin

grant select, insert, update, delete on Thuoc to Admin

grant select, insert, update, delete on DichVu to Admin

exec sp\_addrolemember Admin, Hang

exec sp\_addrolemember Admin, CaoNhatToan

* Tạo role NhanVien, gán quyền và add user

create role NhanVien

grant select, insert, update, delete on BenhNhan to NhanVien

grant select, insert, update, delete on PhieuKhamBenh to NhanVien

grant select on Thuoc to NhanVien

grant select on DichVu to NhanVien

grant select on ChiTietPhieuDichVu to NhanVien

grant select on ChiTietDonThuoc to NhanVien

grant select, update on PhieuDichVu to NhanVien

grant select, update on DonThuoc to NhanVien

exec sp\_addrolemember NhanVien, TranThiAnhThu

exec sp\_addrolemember NhanVien, LuongNgocHaiDang

* Tạo role Bacsi, gán quyền và add user

create role BacSi

grant select on BenhNhan to BacSi

grant select on Thuoc to BacSi

grant select on DichVu to BacSi

grant select, update on PhieuKhamBenh to BacSi

grant select, insert, update, delete on PhieuDichVu to BacSi

grant select, insert, update, delete on ChiTietPhieuDichVu to BacSi

grant select, insert, update, delete on DonThuoc to BacSi

grant select, insert, update, delete on ChiTietDonThuoc to BacSi

exec sp\_addrolemember BacSi, DoanNgocThanhVy

* Tạo role KeToan, gán quyền và add user

create role KeToan

grant select, update on PhieuDichVu to KeToan

grant select, update on DonThuoc to KeToan

grant select on ChiTietPhieuDichVu to BacSi

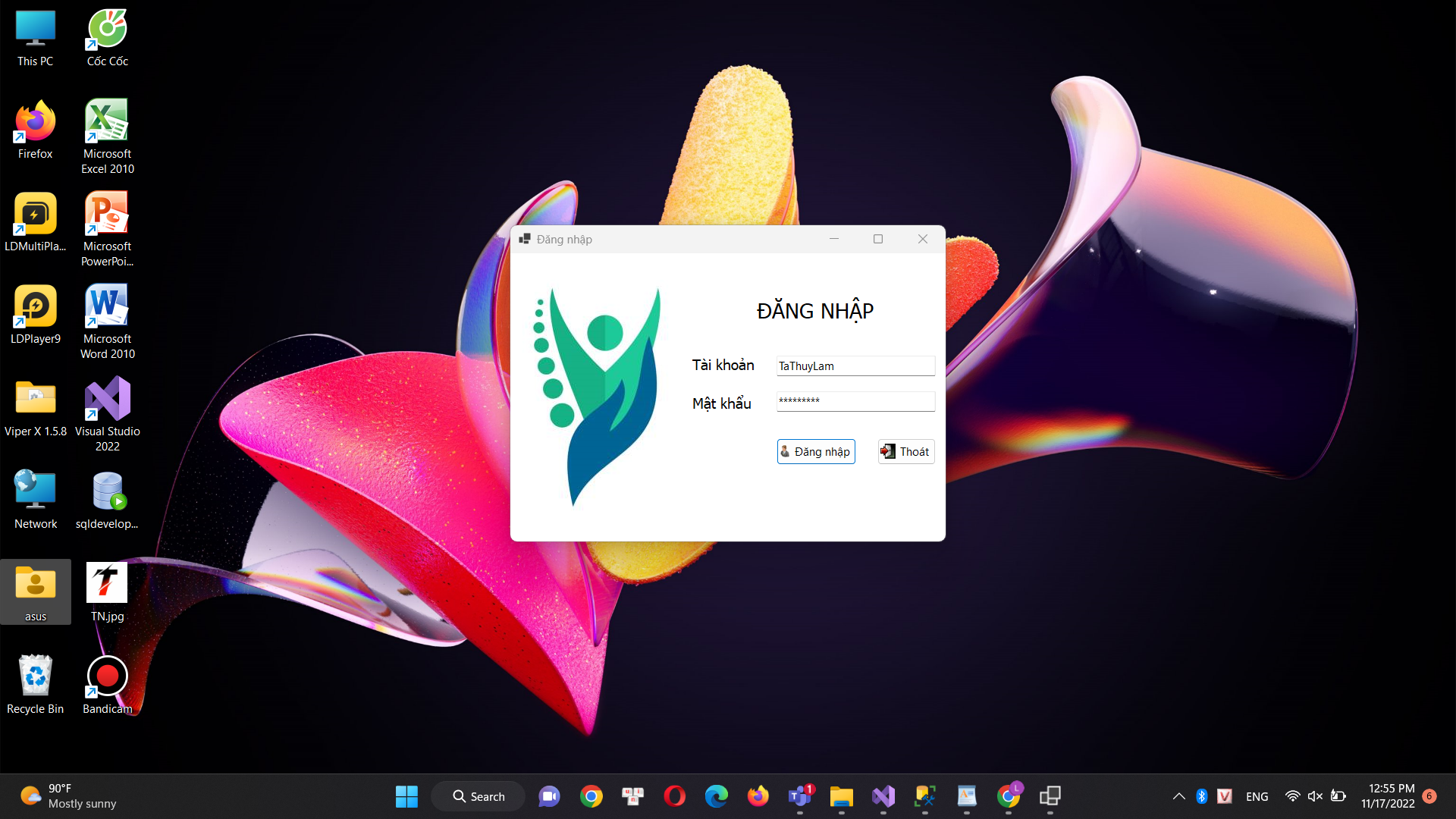
grant select on ChiTietDonThuoc to BacSi

exec sp\_addrolemember KeToan, BuiThanhXuan

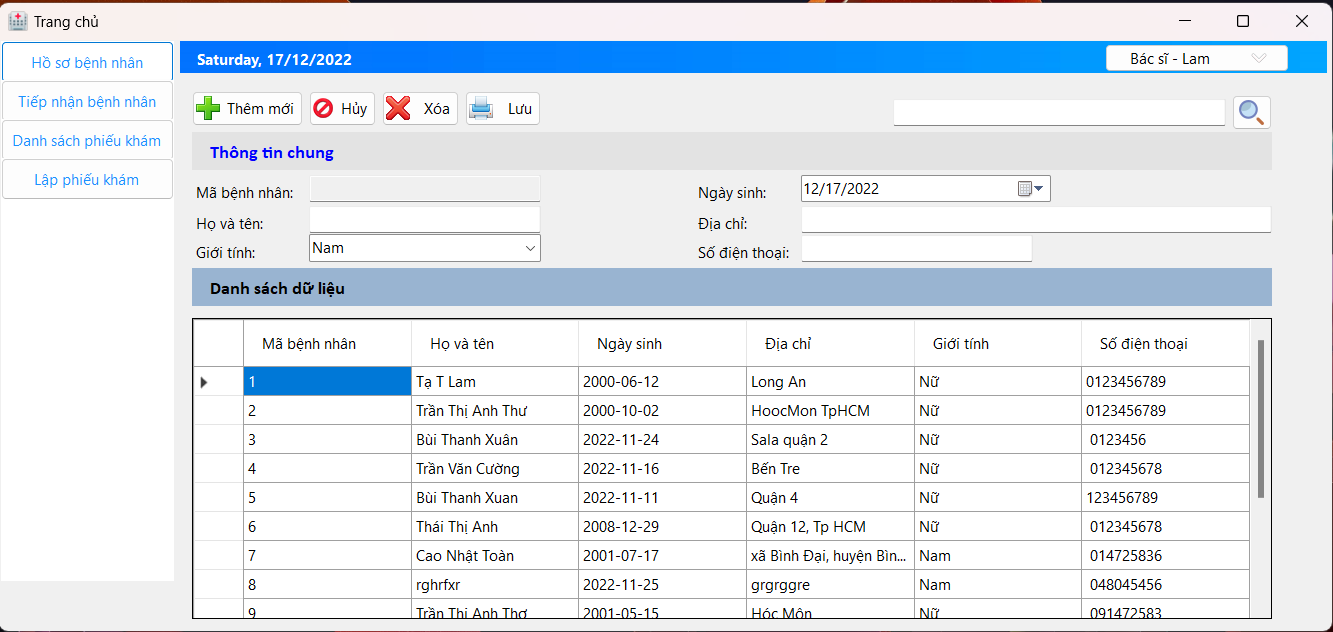
exec sp\_addrolemember KeToan, NguyenThiThuyHang

## Phía Client

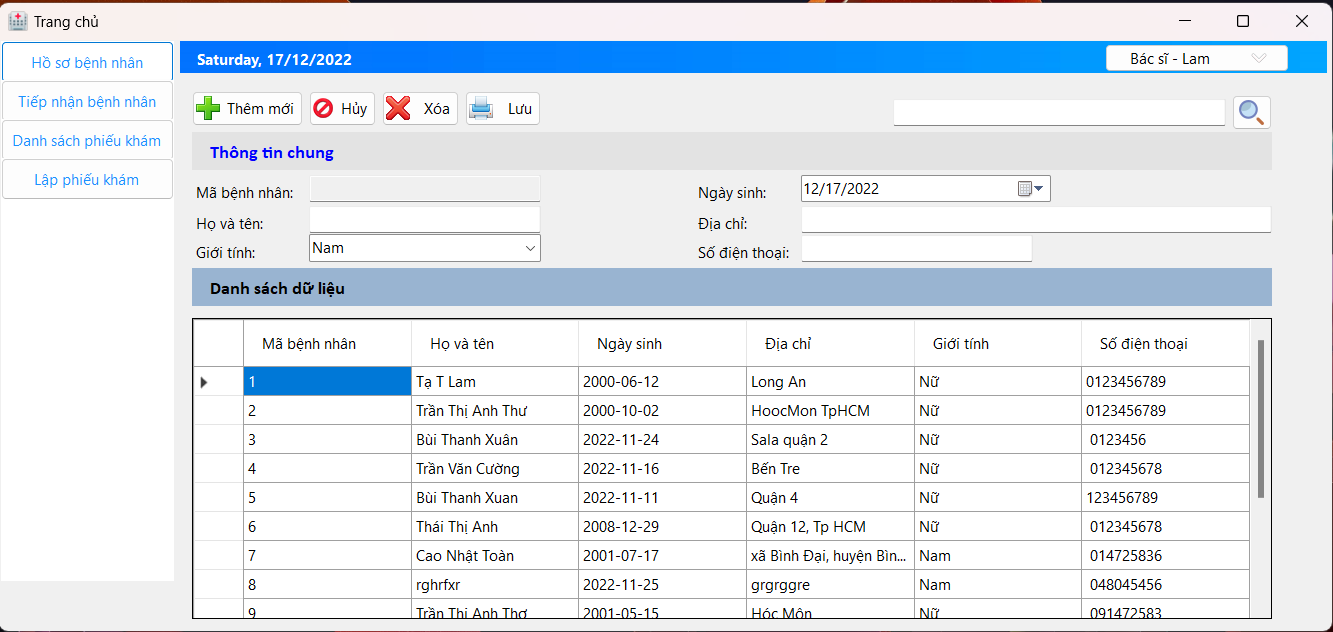
### Giao diện đăng nhập:



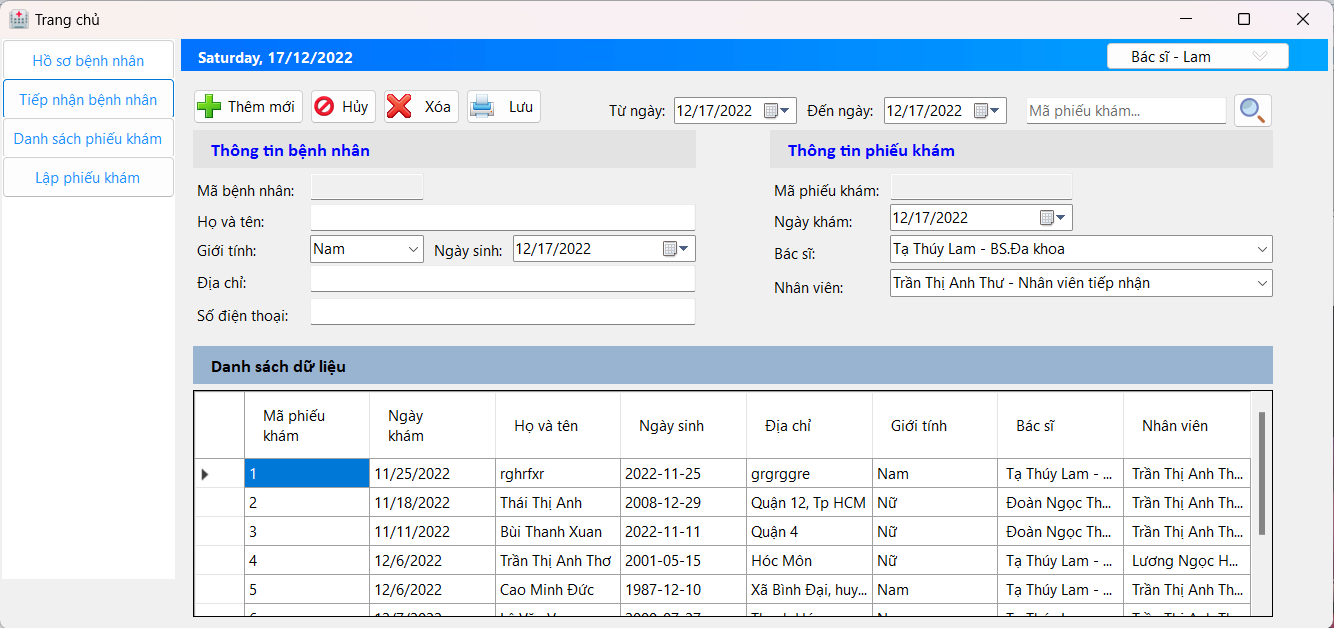
### Giao diện Trang chủ:



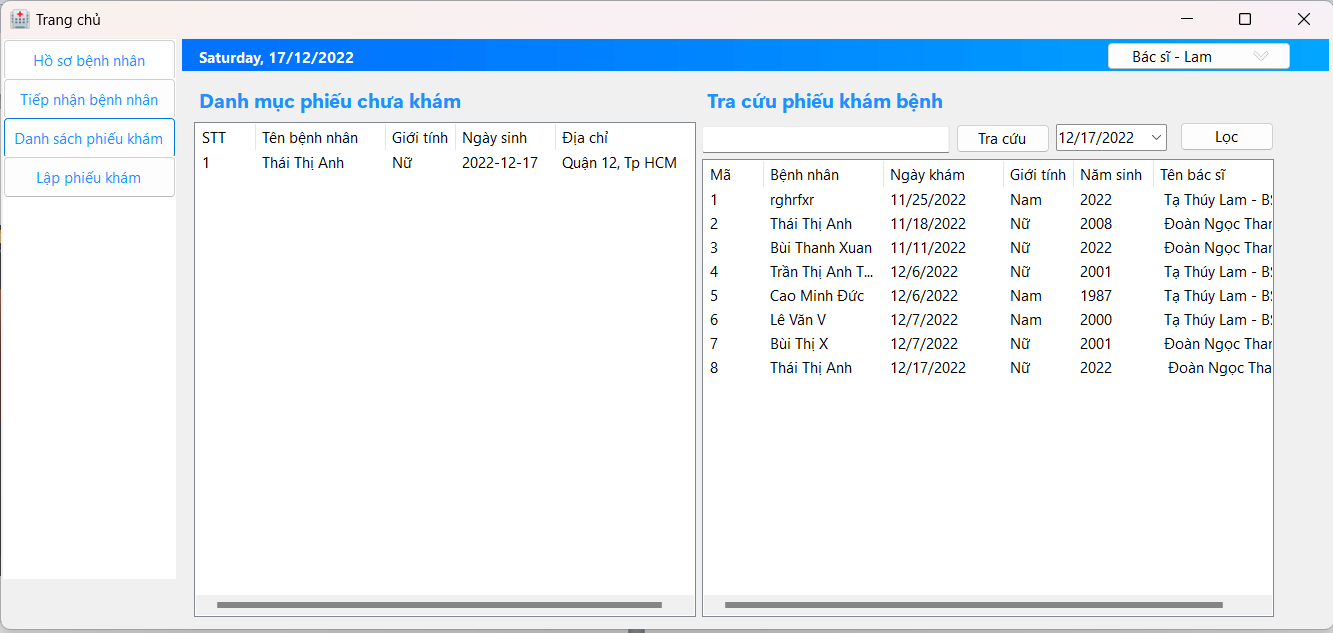
### Giao diện Quản lý hồ sơ bệnh nhân:



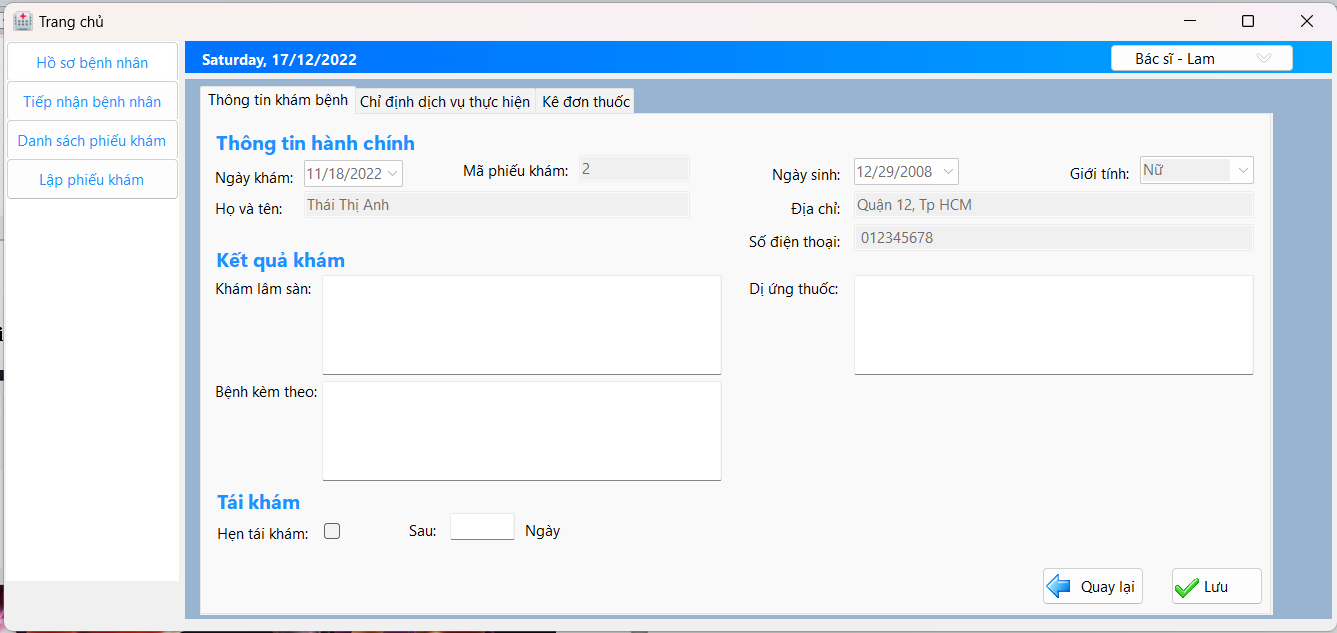
### Giao diện Tiếp đón bệnh nhân



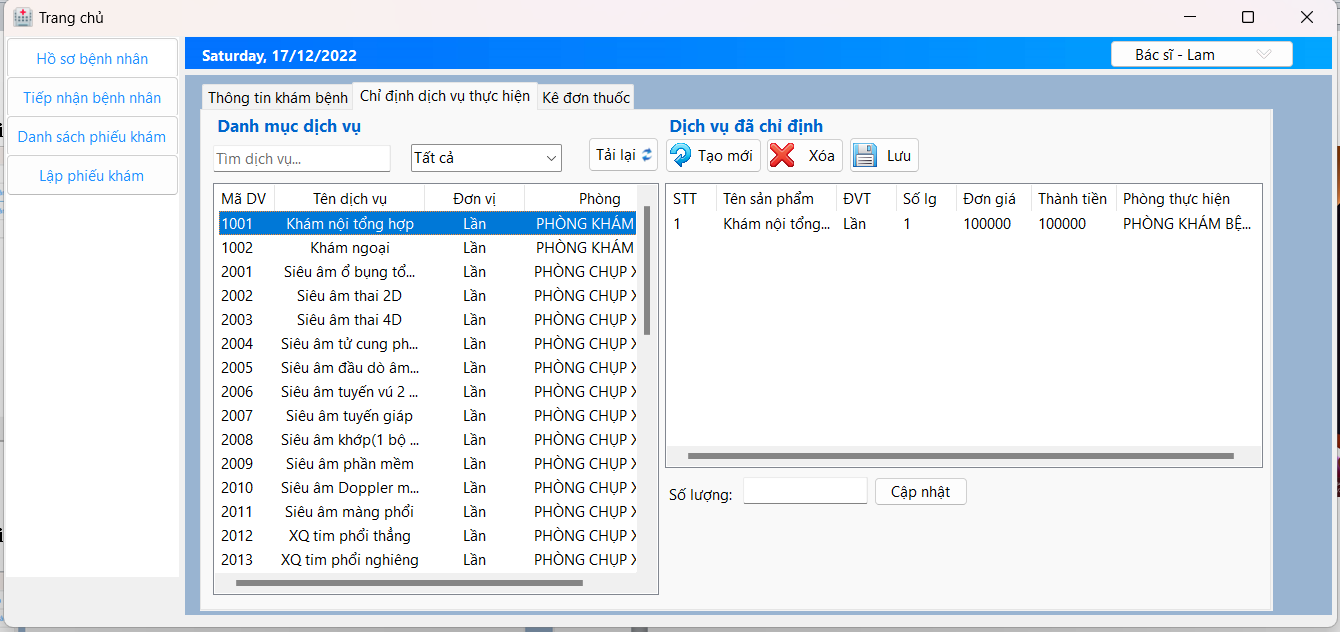
### Giao diện Tra cứu phiếu khám



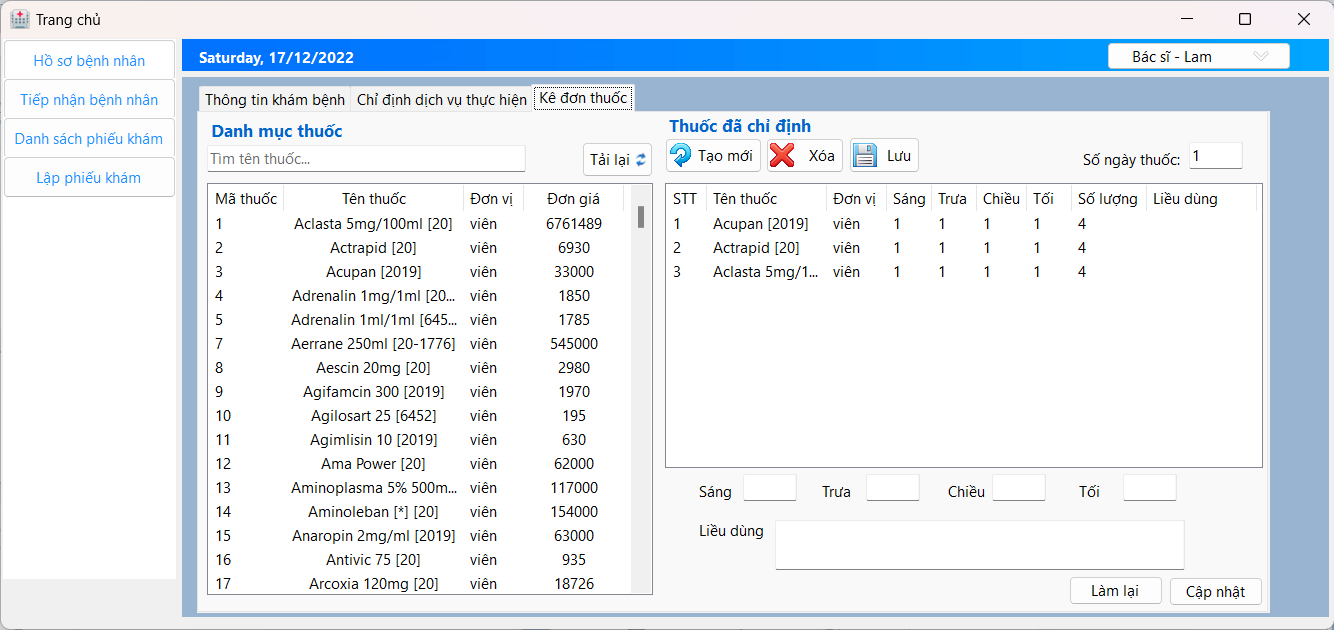
### Giao diện Khám bệnh:



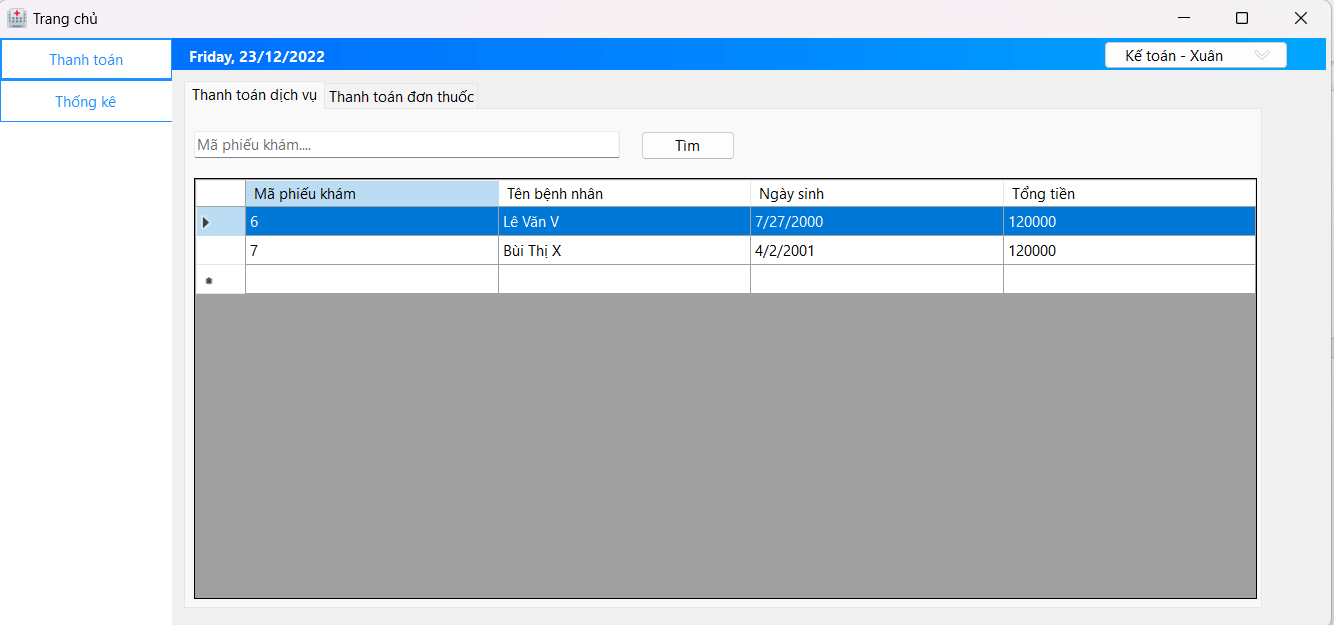
### Giao diện Lập phiếu dịch vụ:



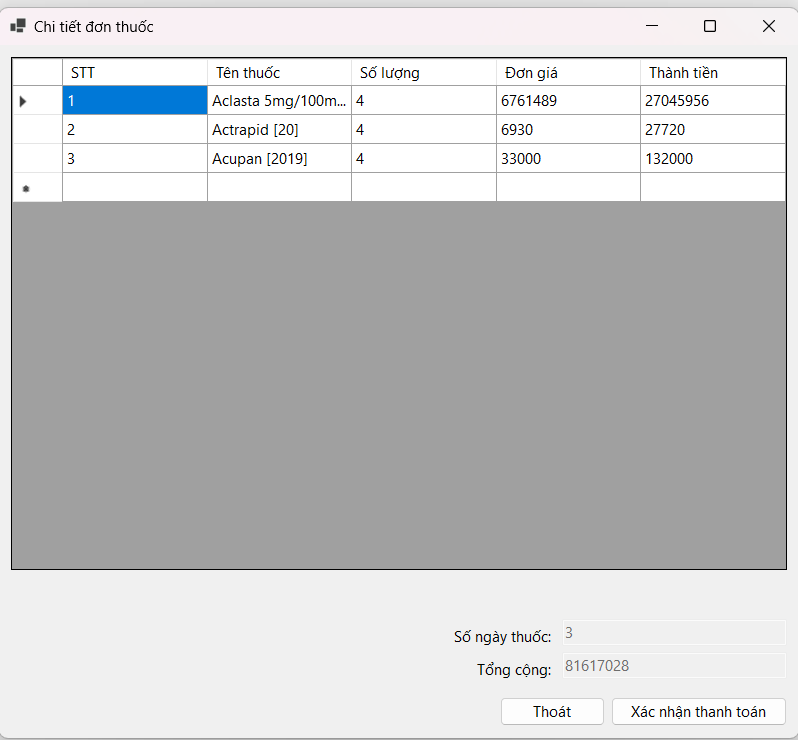
### Giao diện Lập đơn thuốc:



### Giao diện Quản lý Thanh toán:



### Giao diện Xác nhận thanh toán hóa đơn:

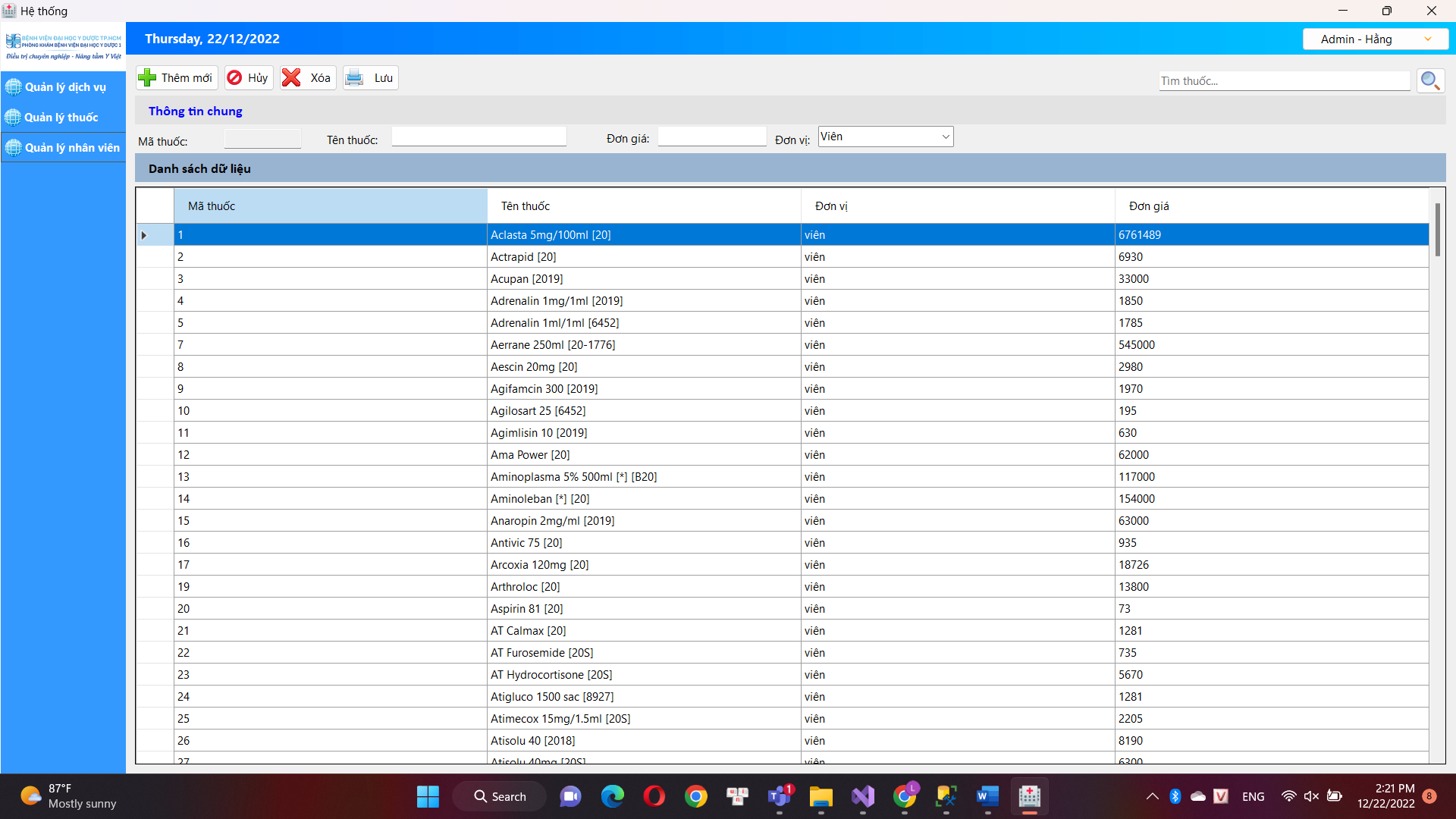


### Giao diện Quản lý dịch vụ:

Ảnh có chứa bàn

Mô tả được tạo tự động

### Giao diện Quản lý thuốc:



### Giao diện Quản lý nhân viên:

Ảnh có chứa bàn

Mô tả được tạo tự động

# Chương 6 KẾT LUẬN

## Kết quả đạt được

### Phía server

* Xây dựng cấu trúc dữ liệu phù hợp với yêu cầu của người dùng.
* Xây dựng các stored procedure và function đáp ứng yêu cầu truy vấn và giao tác của người dùng, đảm bảo việc xử lý giao tác được thực hiện nhanh và có thể xử lý đồng thời. Phía client

Xây dựng trang web phục vụ đầy đủ các chức năng tùy loại người dùng yêu cầu.

### Phía client

## Kết quả chưa đạt được

* Mã hóa dữ liệu nhằm tăng tính bảo mật.
* Giao diện chưa thực sự bắt mắt.

## Hướng phát triển trong tương lai

* Mã hóa dữ liệu để tăng tính bảo mật.
* Thiết kế giao diện bắt mắt hơn.
* Cho phép học viên đăng ký lớp online.

# TÀI LIỆU THAM KHẢO

* Microsoft, Microsoft SQL documentation   
  <https://docs.microsoft.com/vi-vn/sql/?view=sql-server-ver15>
* Microsoft, ASP.NET documentation   
  <https://docs.microsoft.com/vi-vn/aspnet/core/?view=aspnetcore-5.0>
* Stack Exchange, stackoverflow.com  
  <https://stackoverflow.com/>