



TRƯỜNG ĐẠI HỌC NGOẠI NGỮ - TIN HỌC THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN

□ □ □

BÀI BÁO CÁO KẾT THÚC HỌC PHẦN
CƠ SỞ DỮ LIỆU NÂNG CAO

PHÂN TÍCH, THIẾT KẾ VÀ CÀI ĐẶT CSDL CHO DỊCH VỤ KHÁM CHỮA BỆNH KẾT HỢP TƯ VẤN SỨC KHỎE

Giảng viên hướng dẫn:

Sinh viên thực hiện:

1. Trần Anh Kiệt 22DH111832
2. Phạm Ngọc Duy 21DH113535

Thành phố Hồ Chí Minh, tháng 07/2023

Mục lục

DANH MỤC BẢNG	5
DANH MỤC HÌNH.....	6
CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI	6
1.1. Giới thiệu.....	6
1.1.1. Mở đầu	6
1.1.2. Lý do	6
1.2. Khảo sát thực tế.....	7
1.2.1. Thông tin chung	7
1.2.2. Cơ cấu tổ chức	8
1.3. Mô tả các yêu cầu lưu trữ.....	8
1.4. Các yêu cầu khác	11
1.5. Bố cục báo cáo	12
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT	12
2.1. Mô hình thực thể kết hợp	12
2.2. Phụ thuộc hàm.....	12
2.3. Các dạng chuẩn	13
2.4. Bảo toàn thông tin	13
2.5. Lập trình cơ sở dữ liệu	14
2.6. Ràng buộc toàn vẹn	14
2.7. noSQL và newSQL	15
CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU	16
3.1. Mô hình thực thể kết hợp	16
3.2. Mô tả các thực thể	16
3.2.1 Bảng phòng ban	16

3.2.2 Bảng khoa	17
3.2.3 Bảng chức vụ	17
3.2.4 Bảng danh sách khám	17
3.2.5 Bảng đánh giá	17
3.2.6 Bảng phiếu	18
3.2.7 Bảng nhân viên	18
3.2.8 Bảng lịch đặt	19
3.2.9 Bảng bệnh nhân.....	19
3.2.10 Bảng tài khoản	20
3.2.11 Bảng kho	20
3.2.12 Bảng hồ sơ bệnh nhân.....	20
3.2.13 Bảng thuốc	21
3.2.14 Bảng bệnh án.....	21
3.2.15 Bảng hóa đơn	21
3.2.16 Bảng hóa đơn thuốc	22
3.3. Lược đồ cơ sở dữ liệu quan hệ	22
3.4. Phân tích chuẩn của lược đồ quan hệ	23
3.4.1. Lược đồ Khoa (MaKhoa, TenKhoa).....	23
3.4.2. Lược đồ PhongBan(MaPhong,TenPhong,MaKhoa,MaDS)	23
3.4.3. Lược đồ DSKham(MaDS, NgayKham).....	23
3.4.4. Lược đồ Phieu(MaPhieu,TenPhieu, Loai, Ngay, NoiDung, MaNV).....	24
3.4.5. Lược đồ ChucVu(MaCV, TenCV, Luong)	24
3.4.6. Lược đồ TaiKhoan(MaTK, Email, Password, Active)	24
3.4.7. Lược đồ Kho(MaKho, TenKho, DiaChi, Loai, SL).....	24
3.4.8. Lược đồ DanhGia(MaDG, TieuDe, NoiDung, MaBN)	25

3.4.9. Lược đồ HoSoBN(MaHS, NgayLap, MaNV)	25
3.4.10. Lược đồ BenhAn(MaBA, PPDC, NgayKham, MaHS)	25
3.4.11. Lược đồ LichDat (MaDat, NgayDat, MaNV, MaBN).....	26
3.4.12. Lược đồ HoaDon(MaHD, TongTien, MaBA)	26
3.4.13. Lược đồ Thuoc (MaThuoc, TenThuoc, CachDung, DonGia, SL, MaKho)	26
3.4.14. Lược đồ NhanVien(MaNV, HoTen, DiaChi, GioiTinh, SDT, MaPhong, MaTK, MaCV).....	26
3.4.15. Lược đồ BenhNhan(MaBN, TenBN, DiaChi, GioiTinh, SDT, MaTK, MaDS, MaHS)	27
3.5. Lược đồ cơ sở dữ liệu quan hệ sau khi chuẩn hóa thành 3NF	27
3.6. Ràng buộc toàn vẹn	27
3.6.1. Ràng buộc về giới tính	27
3.6.2. Ràng buộc về tuổi của nhân viên	28
3.6.3. Ràng buộc về ngày đặt trong lịch đặt khám.....	28
3.6.4. Ràng buộc về chức vụ	28
3.6.5. Ràng buộc về lương	29
3.6.6. Ràng buộc về ngày lập trong hồ sơ bệnh nhân	29
3.6.7. Ràng buộc bảng Thuoc với Kho	29
3.6.8. Ràng buộc về bảng bệnh án	30
3.6.9. Ràng buộc về số điện thoại trong bảng nhân viên và bệnh nhân.....	30
3.6.10. Ràng buộc bảng hóa đơn.....	30
3.6.11. Ràng buộc bảng tài khoản.....	31
3.7. Các thao tác xử lý	31
CHƯƠNG 4. TRIỂN KHAI CƠ SỞ DỮ LIỆU TRÊN MS SQL SERVER	31
4.1. Tạo cơ sở dữ liệu	31

4.2. Nhập liệu mẫu	36
4.3. Cài các ràng buộc toàn vẹn.....	38
4.3.1. Ràng buộc về giới tính	38
4.3.2. Ràng buộc về tuổi của nhân viên	39
4.3.3. Ràng buộc về ngày đặt trong lịch đặt khám.....	40
4.3.4. Ràng buộc về chức vụ	40
4.3.5. Ràng buộc về lương	41
4.3.6. Ràng buộc về ngày lập trong hồ sơ bệnh nhân	41
4.3.7. Ràng buộc về số lượng trong thuốc và kho.....	42
4.3.8. Ràng buộc về ngày khám bệnh trong bệnh án không được lớn hơn ngày hiện tại.....	43
4.3.9. Ràng buộc về độ dài điện thoại.....	44
4.3.10. Ràng buộc về hóa đơn.....	45
4.3.11. Ràng buộc về active trong tài khoản.....	45
4.4. Cài các store procedure	46
4.4.1. Store procedure cập nhập thông tin	46
4.4.2. Store procedure tìm kiếm thông tin bệnh nhân theo tên	47
4.4.3. Store procedure danh bệnh nhân khám theo ngày	47
4.4.4. Store procedure đặt lịch khám	48
4.4.5. Store procedure hủy lịch đặt	49
4.4.6. Store procedure sửa bệnh án	49
4.4.7. Store procedure Liệt kê danh sách nhân viên theo chức vụ.....	50
4.4.8. Store procedure sửa bệnh án	50
KẾT LUẬN.....	51
Kết quả đã thực hiện.....	51
Những vấn đề chưa thực hiện.....	52

Các nội dung cần bổ sung thêm.....	52
Tài liệu tham khảo	52

DANH MỤC BẢNG

Bảng 1. 1 Khảo sát thực tế.....	7
Bảng 1. 2 Thông tin lưu trữ của hồ sơ bệnh nhân.....	8
Bảng 1. 3 Thông tin lưu trữ của nhân viên y tế	9
Bảng 1. 4 Thông tin lưu trữ chức vụ.....	9
Bảng 1. 5 Thông tin lưu trữ khoa khám bệnh.....	9
Bảng 1. 6 Thông tin lưu trữ hóa đơn	10
Bảng 1. 7 Thông tin lưu trữ thuốc	10
Bảng 1. 8 Thông tin lưu trữ nhà cung cấp	11
Bảng 2. 1 So sánh newSQL và noSQL.....	15
Bảng 3. 1 Bảng phòng ban.....	16
Bảng 3. 2 Bảng khoa.....	17
Bảng 3. 3 Bảng chức vụ.....	17
Bảng 3. 4 Bảng danh sách khám.....	17
Bảng 3. 5 Bảng đánh giá.....	17
Bảng 3. 6 Bảng phiếu.....	18
Bảng 3. 7 Bảng nhân viên.....	18
Bảng 3. 8 Bảng lịch đặt.....	19
Bảng 3. 9 Bảng bệnh nhân.....	19
Bảng 3. 10 Bảng tài khoản.....	20
Bảng 3. 11 Bảng kho	20
Bảng 3. 12 Bảng hồ sơ bệnh nhân	20
Bảng 3. 13 Bảng thuốc.....	21
Bảng 3. 14 Bảng bệnh án.....	21
Bảng 3. 15 Bảng hóa đơn.....	21
Bảng 3. 16 Bảng hóa đơn thuốc.....	22

DANH MỤC HÌNH

Hình 1. 1 Sơ đồ cơ cấu tổ chức.....	8
Hình 3. 1 Mô hình thực thể kết hợp.....	16
Hình 3. 2 Lược đồ cơ sở dữ liệu quan hệ.....	22
Hình 3. 3 Lược đồ cơ sở dữ liệu quan hệ sau khi chuẩn hóa thành 3NF.....	27

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

1.1. Giới thiệu

1.1.1. Mở đầu

Trong thời đại hiện đại, việc quản lý thông tin y tế đang trở thành một phần quan trọng của hệ thống y tế toàn cầu. Việc xây dựng và quản lý cơ sở dữ liệu (CSDL) cho các dịch vụ khám chữa bệnh và tư vấn sức khỏe không chỉ giúp cải thiện chất lượng chăm sóc sức khỏe mà còn tăng cường hiệu suất làm việc của các nhà cung cấp dịch vụ y tế. Đồng thời, việc tận dụng công nghệ thông tin trong lĩnh vực y tế cũng giúp tạo ra một môi trường phục vụ bệnh nhân tốt hơn và nâng cao trải nghiệm của họ khi sử dụng các dịch vụ y tế.

1.1.2. Lý do

Lý do chính để thực hiện đề tài này là để đáp ứng nhu cầu ngày càng tăng của cộng đồng về dịch vụ y tế và tư vấn sức khỏe. Việc sử dụng CSDL để quản lý thông tin về bệnh nhân và các dịch vụ y tế giúp tăng cường tính đồng nhất và minh bạch trong việc chăm sóc sức khỏe. Bên cạnh đó, việc tích hợp chức năng tư vấn sức khỏe vào hệ thống cũng giúp cung cấp thông tin hữu ích và hỗ trợ cho bệnh nhân trong quá trình chăm sóc sức khỏe hàng ngày, từ việc đưa ra lời khuyên về dinh dưỡng đến việc theo dõi và quản lý bệnh lý. Điều này không chỉ giúp cải thiện chất lượng cuộc sống của bệnh nhân mà còn giảm thiểu nguy cơ mắc các bệnh lý và tăng cường sức khỏe cộng đồng.

1.2. Khảo sát thực tế

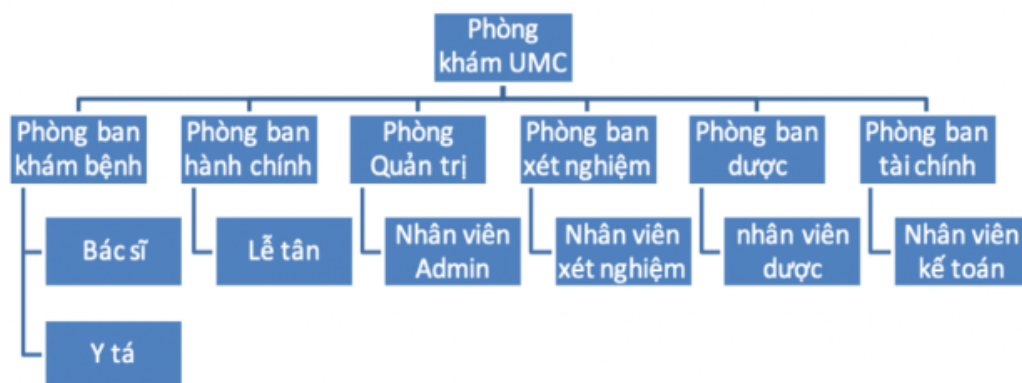
1.2.1. Thông tin chung

Hệ thống hỗ trợ tư vấn, khám, chữa bệnh được phát triển dựa trên nền tảng web và ứng dụng trên điện thoại thông minh. Mục tiêu của ứng dụng này là tăng cường chất lượng công tác tư vấn, khám, chữa bệnh tại tuyến cơ sở, nâng cao tiếp cận dịch vụ y tế cho người dân và nâng cao năng lực cho cán bộ y tế tại tuyến cơ sở. Ứng dụng này cung cấp kết nối giữa cán bộ y tế tại trạm y tế với người dân để phổ biến thông tin y tế, lên lịch hẹn khám tại trạm y tế và cung cấp dịch vụ tư vấn từ xa. Đồng thời, ứng dụng cho phép cán bộ y tế tại trạm y tế nhận sự trợ giúp về chuyên môn từ cán bộ y tế tại bệnh viện, trung tâm y tế huyện hoặc tuyến cao hơn.

Bảng 1. 1 Khảo sát thực tế

Câu hỏi	Trả lời
Câu 1: Phòng khám gồm những ai ?	Phòng khám gồm nhiều phòng ban: Khám bệnh, Hành chính, Quản trị, Xét nghiệm, Dược, Tài chính
Câu 2: Tại sao phải tạo ứng dụng phòng khám ?	Để người dân có thể đặt lịch khám online, theo dõi tình trạng sức khỏe, theo dõi bệnh án
Câu 3: Lợi ích của việc kết hợp khám chữa bệnh và tư vấn sức khỏe là gì?	Giúp người bệnh có cái nhìn tổng quát về sức khỏe, nhận được tư vấn phù hợp từ chuyên gia, cải thiện chất lượng chăm sóc sức khỏe
Câu 4: Bạn có cảm thấy thông tin tư vấn sức khỏe là hữu ích và dễ hiểu không?	Thông tin tư vấn sức khỏe rất hữu ích và dễ hiểu.
Câu 5: Bạn có gặp khó khăn gì khi sử dụng ứng dụng phòng khám không?	Tôi gặp khó khăn khi đăng nhập vào ứng dụng và tìm kiếm thông tin y tế. Hệ thống đặt lịch khám đôi khi bị lỗi, không thể hoàn tất quá trình đặt lịch.

1.2.2. Cơ cấu tổ chức



Hình 1. 1 Sơ đồ cơ cấu tổ chức

1.3. Mô tả các yêu cầu lưu trữ

Bảng 1. 2 Thông tin lưu trữ của hồ sơ bệnh nhân

Stt	Dữ liệu	Kiểu	Hình thức nhập	Ràng buộc
1	Mã bệnh nhân	Text	Nhập từ bàn phím	Không trùng nhau
2	Họ tên	Text	Nhập từ bàn phím	
3	Số điện thoại	Text	Nhập từ bàn phím	
4	Email	Text	Nhập từ bàn phím	
5	Giới tính	Number	Chọn từ danh sách	Nam: 1; Nữ : 2
6	Ngày sinh	Date	Nhập từ bàn phím	
7	Nơi sinh	Text	Nhập từ bàn phím	
8	Địa chỉ	Text	Nhập từ bàn phím	
9	Dân tộc	Text	Nhập từ bàn phím	
10	Mã Khoa	Text	Nhập từ bàn phím	

Bảng 1. 3 Thông tin lưu trữ của nhân viên y tế

Stt	Dữ liệu	Kiểu	Hình thức nhập	Ràng buộc
1	Mã nhân viên	Text	Nhập từ bàn phím	không trùng nhau
2	Họ tên	Text	Nhập từ bàn phím	
3	Số điện thoại	Text	Nhập từ bàn phím	
4	Email	Text	Nhập từ bàn phím	
5	Giới tính	Number	Chọn từ danh sách	Nam: 1; Nữ : 2
6	Ngày sinh	Date	Nhập từ bàn phím	
7	Nơi sinh	Text	Nhập từ bàn phím	
8	Địa chỉ	Text	Nhập từ bàn phím	
9	Mã chức vụ	Text	Nhập từ bàn phím	

Bảng 1. 4 Thông tin lưu trữ chức vụ

Stt	Dữ liệu	Kiểu	Hình thức nhập	Ràng buộc
1	Mã chức vụ	Text	Nhập từ bàn phím	không trùng nhau
2	Tên chức vụ	Text	Nhập từ bàn phím	
3	Hệ số lương	Text	Nhập từ bàn phím	
4	Hệ số thưởng	Text	Nhập từ bàn phím	

Bảng 1. 5 Thông tin lưu trữ khoa khám bệnh

Stt	Dữ liệu	Kiểu	Hình thức nhập	Ràng buộc
-----	---------	------	----------------	-----------

1	Mã khoa	Text	Nhập từ bàn phím	không trùng nhau
2	Tên Khoa	Text	Nhập từ bàn phím	Not null

Bảng 1. 6 Thông tin lưu trữ hóa đơn

Stt	Dữ liệu	Kiểu	Hình thức nhập	Ràng buộc
1	Mã hóa đơn	Text	Nhập từ bàn phím	không trùng nhau
2	Mã nhân viên xuất đơn	Text	Nhập từ bàn phím	không trùng nhau
3	Mã bác sĩ	Text	Nhập từ bàn phím	không trùng nhau
4	Mã bệnh nhân	Text	Nhập từ bàn phím	Not null
5	Ngày xuất đơn	Date	Nhập từ bàn phím	Not null
6	Phương thức thanh toán	Text	Nhập từ bàn phím	Not null
7	Mã số thuế	Text	Nhập từ bàn phím	Not null
8	Mã thuốc	Text	Nhập từ bàn phím	không trùng nhau

Bảng 1. 7 Thông tin lưu trữ thuốc

Stt	Dữ liệu	Kiểu	Hình thức nhập	Ràng buộc
1	Mã thuốc	Text	Nhập từ bàn phím	không trùng nhau
2	Tên thuốc	Text	Nhập từ bàn phím	
3	Mã nhà cung cấp	Text	Nhập từ bàn phím	
4	Đơn giá	number	Nhập từ bàn phím	

5	Số lượng	number	Nhập từ bàn phím	
---	----------	--------	------------------	--

Bảng 1. 8 Thông tin lưu trữ nhà cung cấp

Stt	Dữ liệu	Kiểu	Hình thức nhập	Ràng buộc
1	Mã nhà cung cấp	Text	Nhập từ bàn phím	không trùng nhau
2	Tên nhà cung cấp	Text	Nhập từ bàn phím	
3	Địa chỉ	Text	Nhập từ bàn phím	

1.4. Các yêu cầu khác

- Bảo mật dữ liệu: Thông tin cá nhân của bệnh nhân, nhân viên y tế,... bao gồm tên, địa chỉ, số điện thoại, email và hồ sơ bệnh án, phải được bảo mật và chỉ được tiếp cận bởi người có quyền truy cập
- Khả dụng: Hệ thống quản lý dịch vụ khám chữa bệnh kết hợp với tư vấn sức khỏe phải luôn sẵn sàng 24/7 để quản lý các dịch vụ một cách hiệu quả và liên tục.
- Dễ sử dụng: Giao diện hệ thống phải dễ sử dụng, giúp nhân viên quản lý tiết kiệm thời gian và tăng năng suất trong công việc
- Cập nhật dữ liệu liên tục: Đảm bảo cơ sở dữ liệu quản lý được cập nhật liên tục và chính xác để đảm bảo rằng thông tin về bệnh nhân và các dịch vụ khám chữa bệnh được đầy đủ và không thiếu trong hoạt động quản lý và đánh giá
- Quyền truy cập dữ liệu: Hệ thống quản lý các dịch vụ khám chữa bệnh kết hợp tư vấn sức khỏe cần có cơ chế phân quyền truy cập, chỉ cho phép người có quyền truy cập vào các dữ liệu cần thiết dựa trên vai trò và trách nhiệm của họ
- Tuân thủ quy định pháp luật: đảm bảo rằng quản lý phải tuân thủ các quy định, luật pháp và quyền riêng tư liên quan đến việc bảo vệ thông tin cá nhân của bệnh nhân và các nhân viên y tế

1.5. Bộ cục báo cáo

Chương 1: Giới thiệu đề tài

Chương 2: Cơ sở lý thuyết

Chương 3: Phân tích và thiết kế cơ sở dữ liệu

Chương 4: Triển khai cơ sở dữ liệu trên MS SQL SERVER

Chương 5: Kết luận

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1. Mô hình thực thể kết hợp

- Mô hình thực thể kết hợp (ERD) là một mô hình dữ liệu phổ biến được sử dụng để thiết kế cơ sở dữ liệu cho các ứng dụng. Mô hình này được phát triển nhằm giúp người thiết kế có thể trừu tượng hóa dữ liệu và mô tả quan hệ giữa các thực thể trong hệ thống.
- Một mô hình thực thể được biểu diễn bằng các biểu đồ thực thể, gồm hình vẽ thực thể (Entity), thuộc tính (Attribute) và mối quan hệ (Relationship) giữa chúng. Thiết kế ERD giúp cho người thiết kế có thể hiểu được mối quan hệ giữa các thực thể, thuộc tính và mối quan hệ giữa các thực thể trước khi bắt đầu thiết kế cơ sở dữ liệu.
- Mô hình ERD được sử dụng rộng rãi trong thiết kế cơ sở dữ liệu vì nó cho phép các nhà thiết kế có thể mô tả các thực thể trong hệ thống một cách rõ ràng và trực quan. Việc sử dụng mô hình này sẽ giúp cho người thiết kế cơ sở dữ liệu có được một mô tả chi tiết về dữ liệu cần thiết và mối quan hệ giữa chúng, từ đó giúp cho thiết kế cơ sở dữ liệu trở nên dễ dàng, chính xác và hiệu quả hơn.

2.2. Phụ thuộc hàm

- Phụ thuộc hàm (Functional Dependency) là một khái niệm quan trọng trong cơ sở dữ liệu quan hệ (Relational Database). Nó là mối quan hệ giữa hai tập hợp các thuộc tính trong bảng. Trong đó, việc giá trị của một thuộc tính (hoặc tập hợp các thuộc tính) xác

định một cách duy nhất giá trị của thuộc tính (hoặc tập hợp các thuộc tính) khác trong bảng.

- Phụ thuộc hàm được sử dụng để thiết kế cơ sở dữ liệu quan hệ và đảm bảo tính toàn vẹn dữ liệu trong quá trình thao tác dữ liệu. Nó giúp hạn chế trường hợp dữ liệu trùng lặp và giúp cải thiện hiệu quả của các truy vấn cơ sở dữ liệu. Vì thế, việc phân tích và xác định phụ thuộc hàm là rất quan trọng trong thiết kế và quản lý cơ sở dữ liệu quan hệ.

2.3. Các dạng chuẩn

Trong cơ sở dữ liệu, chuẩn hóa là quá trình tạo ra cấu trúc cơ sở dữ liệu phù hợp, giảm thiểu sự trùng lặp và bảo đảm tính toàn vẹn dữ liệu. Các chuẩn của cơ sở dữ liệu cũng là tiêu chuẩn được áp dụng để đảm bảo tính toàn vẹn cơ sở dữ liệu. Dưới đây là một số chuẩn cơ sở dữ liệu quan trọng:

- First Normal Form (1NF): Đối với một bảng trong cơ sở dữ liệu, thì các giá trị phải được ngăn cách riêng biệt và không có giá trị lặp lại trong một trường nào đó. Điều này giúp loại bỏ các giá trị đơn lẻ hoặc nhóm giá trị lặp lại không cần thiết.
- Second Normal Form (2NF): Bảng cần được phân thành các bảng con để thực hiện phụ thuộc hàm đầy đủ hoàn toàn.
- Third Normal Form (3NF): Một cột trong bảng không nên phụ thuộc vào một hoặc nhiều cột khác nếu chúng không có mối quan hệ chức năng.
- Fourth Normal Form (4NF): Mỗi bảng cần được phân thành hai bảng để loại bỏ các phụ thuộc đa giá trị nếu có.
- Fifth Normal Form (5NF): Loại bỏ các phụ thuộc ngược, hay phụ thuộc đa mức.

2.4. Bảo toàn thông tin

Bảo toàn thông tin trong cơ sở dữ liệu (CSDL) là rất quan trọng để đảm bảo tính toàn vẹn, sự an toàn và quyền riêng tư của thông tin. Dưới đây là một số phương pháp giúp bảo toàn thông tin trong cơ sở dữ liệu:

- Sao lưu dữ liệu thường xuyên: Điều này giúp phục hồi dữ liệu trong trường hợp có sự cố xảy ra như thiết bị hỏng, virus hoặc lỗi hệ thống.
- Chia sẻ dữ liệu an toàn: Quản lý quyền truy để giới hạn quyền truy cập vào dữ liệu. Thông tin được chia sẻ chỉ cho những người được phép truy cập.
- Thiết kế chuẩn hóa CSDL: Sử dụng chuẩn hóa CSDL để giảm thiểu sự trùng lặp và đảm bảo tính toàn vẹn của dữ liệu.
- Xác thực và phân quyền truy cập: Xác định các người dùng được phép truy cập vào dữ liệu và các quyền hạn của họ trong hệ thống CSDL.

2.5. Lập trình cơ sở dữ liệu

- Lập trình cơ sở dữ liệu là quá trình tạo ra các ứng dụng để quản lý và xử lý dữ liệu của một hệ thống cơ sở dữ liệu (CSDL). Nó bao gồm việc thiết kế cơ sở dữ liệu, tạo bảng, quản lý quan hệ giữa các bảng, và các truy vấn để tương tác với dữ liệu trong CSDL.

- Các ngôn ngữ lập trình phổ biến để tương tác với CSDL bao gồm SQL (Structured Query Language), PL/SQL (Procedural Language/Structured Query Language), và T-SQL (Transact-SQL). Tùy vào hệ quản trị CSDL sử dụng, các ngôn ngữ này có thể có các cú pháp khác nhau để tạo các truy vấn, thêm, xóa hoặc cập nhật dữ liệu trong CSDL.

- Lập trình cơ sở dữ liệu cũng liên quan đến các yếu tố khác như bảo mật dữ liệu, đồng bộ dữ liệu, sao lưu và phục hồi dữ liệu, và các công nghệ liên quan đến phụ thuộc hàm. Với sự phát triển nhanh chóng của công nghệ và xu hướng sử dụng dữ liệu Big Data, lập trình cơ sở dữ liệu đang trở thành một lĩnh vực quan trọng và thu hút sự quan tâm của nhiều nhà phát triển và doanh nghiệp.

2.6. Ràng buộc toàn vẹn

Ràng buộc toàn vẹn (integrity constraints) trong hệ quản trị cơ sở dữ liệu (HQCSDL) là các quy tắc quản lý dữ liệu được thiết kế để đảm bảo tính toàn vẹn (integrity) của dữ liệu. Nó đảm bảo rằng những dữ liệu được lưu trong Cơ sở dữ liệu phù hợp với các quy tắc và chuẩn đã được xác định và được đảm bảo độ chính xác, nhất quán và không trái ngược với các ràng buộc đó.

2.7. noSQL và newSQL

- NoSQL (Not Only SQL) là một hệ thống cơ sở dữ liệu phi quan hệ, được thiết kế để xử lý các loại dữ liệu khác nhau và đáp ứng các nhu cầu của các ứng dụng truy xuất dữ liệu lớn, phân tán và không có cấu trúc rõ ràng. NoSQL không sử dụng các bảng, hàng và cột như trong các hệ thống quan hệ truyền thống. Thay vào đó, nó sử dụng các khái niệm như tài liệu (document), cặp khóa giá trị (key-value), và đồ thị (graph).
- NewSQL là một loại hệ thống cơ sở dữ liệu mới cung cấp các tính năng tương tự như các cơ sở dữ liệu quan hệ truyền thống nhưng với khả năng mở rộng phân tán tốt hơn. NewSQL chủ yếu tập trung vào việc giải quyết các vấn đề về hiệu suất và khả năng mở rộng của các hệ thống cơ sở dữ liệu quan hệ.

Ngôn ngữ Query: Structured Query Language Không có ngôn ngữ Query

Khả năng mở rộng: Có thể mở rộng theo chiều dọc Có thể mở rộng theo chiều ngang

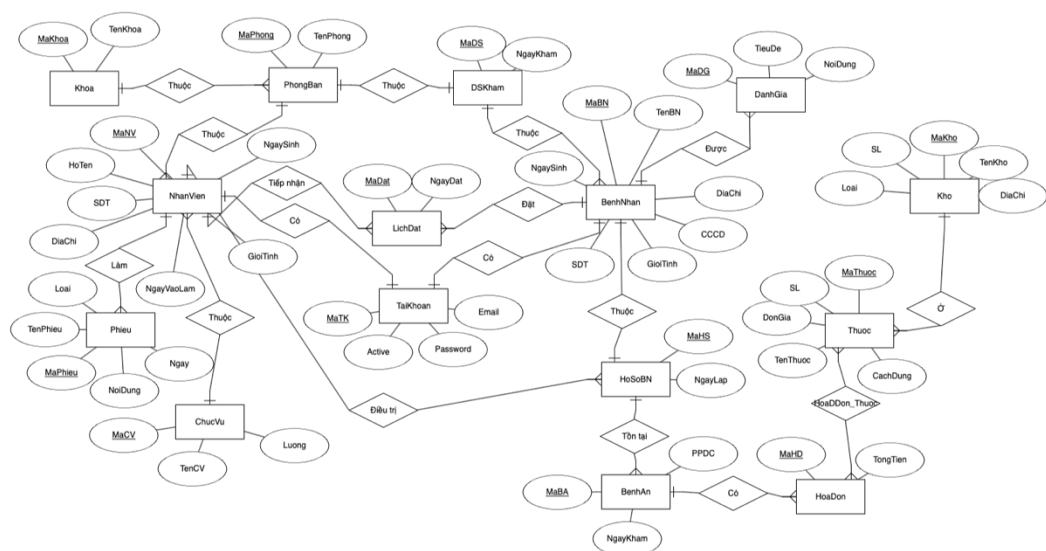
Bảng 2. 1 So sánh newSQL và noSQL

Tiêu chí	new SQL	noSql
Loại hình	SQL database là cơ sở dữ liệu dựa trên bảng	noSQL database có thể dựa trên tài liệu, cặp khóa-giá trị, biểu
Lưu trữ dữ liệu phân cấp	Không thích hợp cho việc lưu trữ dữ liệu phân cấp	Phù hợp cho kho lưu trữ dữ liệu phân cấp vì nó hỗ trợ phương thức cặp khóa giá trị
Mục đích sử dụng	Được thiết kế dành cho các ứng dụng xử lý giao dịch trực tuyến trong giao dịch có độ ổn định cao và thích hợp để xử lý phân tích trực tuyến	Được thiết kế để phục vụ phân tích dữ liệu có cấu trúc chưa hoàn chỉnh
Lựa chọn sử dụng	Dự án đã có yêu cầu dữ liệu rõ ràng xác định quan hệ logic có thể được xác định trước	Dự án yêu cầu dữ liệu không liên quan, khó xác định, đơn giản mềm dẻo khi đang phát triển

Mã nguồn	Một sự kết hợp của mã nguồn mở như Postgres & MySQL, và thương mại như Oracle Database	Open-source
-----------------	--	-------------

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU

3.1. Mô hình thực thể kết hợp



Hình 3. 1 Mô hình thực thể kết hợp

3.2. Mô tả các thực thể

3.2.1 Bảng phòng ban

Bảng 3. 1 Bảng phòng ban

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaPhong	Char	5	Khóa chính	Mã Phòng
2	MaKhoa	Char	5	Khóa ngoại	Mã Khoa
3	TenPhong	Varchar	20		Tên Phòng
4	MaDS	Char	5	Khóa ngoại	Mã Danh Sách Khám

3.2.2 Bảng khoa

Bảng 3. 2 Bảng khoa

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaKhoa	Char	5	Khóa chính	Mã Khoa
2	TenKhoa	Varchar	20		Tên khoa

3.2.3 Bảng chức vụ

Bảng 3. 3 Bảng chức vụ

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaCV	Char	5	Khóa chính	Mã Chức Vụ
2	TenCV	Varchar	20		Tên Chức Vụ
3	Luong	Float			Lương

3.2.4 Bảng danh sách khám

Bảng 3. 4 Bảng danh sách khám

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaDS	Char	5	Khóa chính	Mã Phòng
2	NgayKham	Date			Ngày khám

3.2.5 Bảng đánh giá

Bảng 3. 5 Bảng đánh giá

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaDG	Char	5	Khóa chính	Mã Đánh Giá
2	MaBN	Char	5	Khóa ngoại	Mã Bệnh Nhân
3	TieuDe	Varchar	55		Tiêu đề

4	NoiDung	VarChar	255		Nội dung
---	---------	---------	-----	--	----------

3.2.6 Bảng phiếu

Bảng 3. 6 Bảng phiếu

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaPhieu	Char	5	Khóa chính	Mã Phiếu
2	TenPhieu	Varchar	20		Tên Phiếu
3	Loai	Varchar	20		Loại Phiếu
4	Ngay	Date			Ngày ghi phiếu
5	NoiDung	Varchar	50		Nội dung
6	MaNV	Char	5	Khóa ngoại	Mã Nhân Viên

3.2.7 Bảng nhân viên

Bảng 3. 7 Bảng nhân viên

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaNV	Char	5	Khóa chính	Mã Nhân Viên
2	MaPhong	Char	5	Khóa ngoại	Mã Phòng
3	MaKhoa	Char	5	Khóa ngoại	Mã Khoa
4	MaTK	Char	5	Khóa ngoại	Mã Tài Khoản
5	HoTen	Varchar	20		Họ Tên
6	DiaChi	Varchar	55		Địa Chỉ
7	GioiTinh	Varchar	20		Giới Tính
8	SĐT	Varchar	10		Số Điện Thoại

3.2.8 Bảng lịch đặt

Bảng 3. 8 Bảng lịch đặt

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaDat	Char	5	Khóa chính	Mã Đặt
2	MaNV	Char	5	Khóa ngoại	Mã Nhân Viên
3	MaBN	Char	5	Khóa ngoại	Mã Bệnh Nhân
4	NgayDat	Date			Ngày đặt

3.2.9 Bảng bệnh nhân

Bảng 3. 9 Bảng bệnh nhân

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaBN	Char	5	Khóa chính	Mã bệnh nhân
2	MaTK	Char	5	Khóa ngoại	Mã tài khoản
3	TenBN	Varchar	20		Tên bệnh nhân
4	DiaChi	Varchar	55		Địa chỉ
5	CCCD	Varchar	55		Căn cước công dân
6	GioiTinh	Varchar	20		Giới tính
7	SDT	Varchar	10		Số điện thoại
8	MaDS	Char	5		Mã danh sách khám
9	MaHS	Char	5		Mã hồ sơ

3.2.10 Bảng tài khoản

Bảng 3. 10 Bảng tài khoản

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaTK	Char	5	Khóa chính	Mã tài khoản
2	Email	Varchar	55		Email
3	Password	Varchar	55		Mật khẩu
4	Active	Int			Kích hoạt

3.2.11 Bảng kho

Bảng 3. 11 Bảng kho

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaKho	Char	5	Khóa chính	Mã kho
2	TenKho	Varchar	55		Tên kho
3	DiaChi	Varchar	255		Địa chỉ kho
4	Loai	Varchar	55		Loại
5	SL	Int			Số lượng

3.2.12 Bảng hồ sơ bệnh nhân

Bảng 3. 12 Bảng hồ sơ bệnh nhân

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaHS	Char	5	Khóa chính	Mã hồ sơ
2	MaNV	Char	5	Khóa ngoại	Mã nhân viên
3	NgayLap	Date			Ngày lập hồ sơ

3.2.13 Bảng thuốc

Bảng 3. 13 Bảng thuốc

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaThuoc	Char	5	Khóa chính	Mã thuốc
2	MaKho	Char	5	Khóa ngoại	Mã kho
3	TenThuoc	Varchar	55		Tên thuốc
4	CachDung	Varchar	255		Cách dung
5	DonGia	Float			Đơn giá
6	SL	Int			Số lượng

3.2.14 Bảng bệnh án

Bảng 3.14 Bảng bệnh án

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaBA	Char	5	Khóa chính	Mã bệnh án
2	MaHS	Char	5	Khóa ngoại	Mã hồ sơ
3	PPDC	Varchar	255		
4	NgayKham	Date			Ngày khám

3.2.15 Bảng hóa đơn

Bảng 3. 15 Bảng hóa đơn

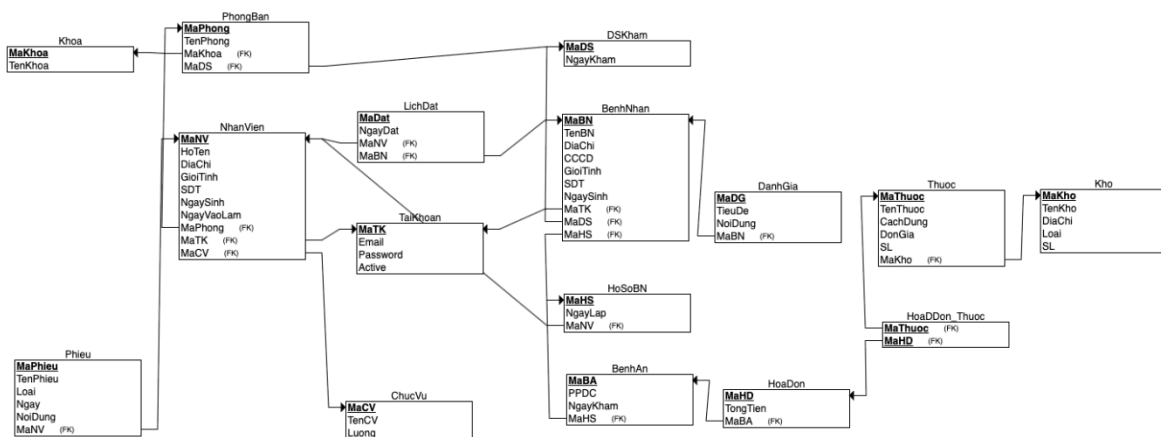
STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaHD	Char	5	Khóa chính	Mã hóa đơn
2	MaBA	Char	5	Khóa ngoại	Mã bệnh án
3	TongTien	Float			Tổng tiền

3.2.16 Bảng hóa đơn thuốc

Bảng 3. 16 Bảng hóa đơn thuốc

STT	Thuộc Tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	MaHD	Char	5	Khóa chính	Mã hóa đơn
2	MaThuoc	Char	5	Khóa Chính	Mã thuốc

3.3. Lược đồ cơ sở dữ liệu quan hệ



Hình 3. 2 Lược đồ cơ sở dữ liệu quan hệ

Biểu diễn dạng chữ:

- ◆ Khoa (MaKhoa, TenKhoa)
- ◆ PhongBan(MaPhong, TenPhong, #MaKhoa, #MaDS)
- ◆ DanhSachKham (MaDS, #MaNV, NgayKham)
- ◆ DanhGia (MaDG, #MaBN, TieuDe, NoiDung)
- ◆ Phieu(MaPhieu, #MaNV, TenPhieu, Loai, Ngay, NoiDung)
- ◆ NhanVien(MaNV, #MaCV, #MaKhoa, #MaTK, HoTen, DiaChi, GioiTinh, SDT)
- ◆ LichDat(MaDat, #MaNV, #MaBN, NgayDat)
- ◆ BenhNhan (MaBN, #MaTK, #MaDS, #MaHS, TenBN, DiaChi, CCCD, GioiTinh, SDT)
- ◆ Kho(MaKho, TenKho, DiaChi, Loai, SL)
- ◆ ChucVu(MaCV, TenCV, Luong)

- ◆ HoSoBN (MaHS, #MaNV, NgayLap)
- ◆ Thuoc(MaThuoc, #MaKho, TenThuoc, CachDung, DonGia, SL)
- ◆ BenhAn(MaBA, #MaHS, PPDC, NgayKham)
- ◆ HoaDon(MaHD, #MaBA, TongTien)
- ◆ HoaDon_Thuoc (MaHD, MaThuoc)

3.4. Phân tích chuẩn của lược đồ quan hệ

3.4.1. Lược đồ Khoa (MaKhoa, TenKhoa)

- Phụ thuộc hàm: MaKhoa -> TenKhoa
- Khóa chính: MaKhoa
- Đánh giá 1NF: Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF: Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF: Đạt do không tồn tại thuộc tính bắt cầu vào khóa
- > Kết luận lược đồ Khoa đạt 3NF

3.4.2. Lược đồ PhongBan(MaPhong,TenPhong,MaKhoa,MaDS)

- Phụ thuộc hàm: MaPhong -> TenPhong, MaKhoa, MaDS
- Khóa chính: MaPhong
- Đánh giá 1NF: Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF: Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF: Đạt do không tồn tại thuộc tính bắt cầu vào khóa
- > Kết luận lược đồ PhongBan đạt 3NF

3.4.3. Lược đồ DSKham(MaDS, NgayKham)

- Phụ thuộc hàm: MaDS -> NgayKham
- Khóa chính: MaDS
- Đánh giá 1NF: Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF: Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF: Đạt do không tồn tại thuộc tính bắt cầu vào khóa

-> Kết luận lược đồ DSKham đạt 3NF

3.4.4. Lược đồ Phieu(MaPhieu,TenPhieu, Loai, Ngay, NoiDung, MaNV)

- Phụ thuộc hàm:MaPhieu -> TenPhieu, Loai, Ngay, NoiDung, MaNV
 - Khóa chính: MaPhieu
 - Đánh giá 1NF: Đạt do tất cả thuộc tính lưu giá trị nguyên tố
 - Đánh giá 2NF: Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
 - Đánh giá 3NF: Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ Phieu đã đạt 3NF

3.4.5. Lược đồ ChucVu(MaCV, TenCV, Luong)

- Phụ thuộc hàm: MaCV -> TenCV, Luong
 - Khóa chính: MaCV
 - Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
 - Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
 - Đánh giá 3NF : Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ ChucVu đã đạt 3NF

3.4.6. Lược đồ TaiKhoan(MaTK, Email, Password, Active)

- Phụ thuộc hàm: MaTK -> Email, Password, Active
 - Khóa chính: MaTK
 - Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
 - Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
 - Đánh giá 3NF : Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ TaiKhoan đã đạt 3NF

3.4.7. Lược đồ Kho(MaKho, TenKho, DiaChi, Loai, SL)

- Phụ thuộc hàm: MaKho -> TenKho, DiaChi, Loai, SL
- Khóa chính: MaKho
- Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố

- Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF : Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ Kho đã đạt 3NF

3.4.8. Lược đồ DanhGia(MaDG, TieuDe, NoiDung, MaBN)

- Phụ thuộc hàm: MaDG -> TieuDe, NoiDung, MaBN
- Khóa chính: MaDG
- Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF : Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ DanhGia đã đạt 3NF

3.4.9. Lược đồ HoSoBN(MaHS, NgayLap, MaNV)

- Phụ thuộc hàm: MaHS -> NgayLap, MaNV
- Khóa chính: MaHS
- Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF : Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ HoSoBN đã đạt 3NF

3.4.10. Lược đồ BenhAn(MaBA, PPDC, NgayKham, MaHS)

- Phụ thuộc hàm: MaBA -> PPDC, NgayKham, MaHS
- Khóa chính: MaBA
- Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF : Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ BenhAn đã đạt 3NF

3.4.11. Lược đồ LichDat (MaDat, NgayDat, MaNV, MaBN)

- Phụ thuộc hàm: MaDat -> NgayDat, MaNV, MaBN
- Khóa chính: MaDat
- Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF : Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ LichDat đã đạt 3NF

3.4.12. Lược đồ HoaDon(MaHD, TongTien, MaBA)

- Phụ thuộc hàm: MaHD -> TongTien, MaBA
- Khóa chính: MaHD
- Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF : Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ HoaDon đã đạt 3NF

3.4.13. Lược đồ Thuoc (MaThuoc, TenThuoc, CachDung, DonGia, SL, MaKho)

- Phụ thuộc hàm: MaThuoc -> TenThuoc, CachDung, DonGia, SL, MaKho
- Khóa chính: MaThuoc
- Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF : Đạt do không tồn tại thuộc tính bất cầu vào khóa
- > Kết luận lược đồ Thuoc đã đạt 3NF

3.4.14. Lược đồ NhanVien(MaNV, HoTen, DiaChi, GioiTinh, SDT, MaPhong, MaTK, MaCV)

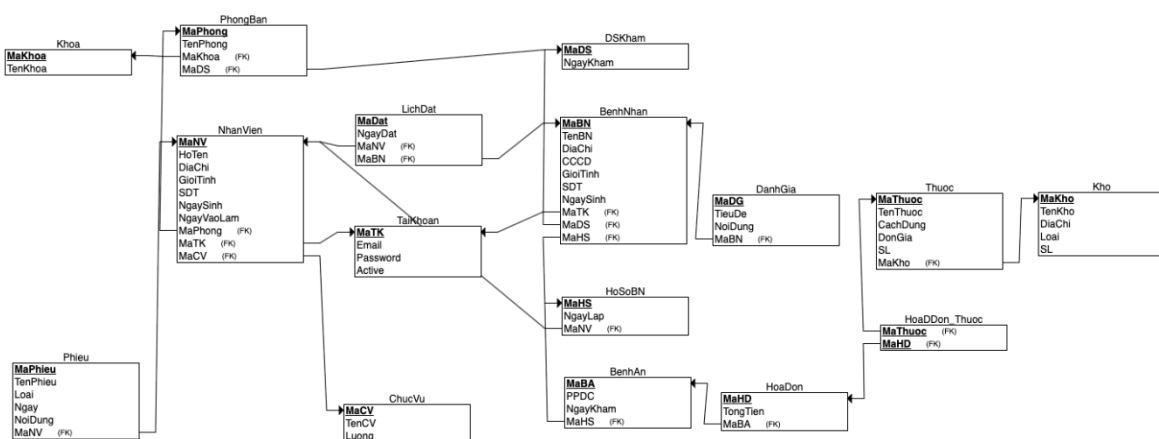
- Phụ thuộc hàm: MaNV -> HoTen, DiaChi, GioiTinh, SDT, MaPhong, MaTK, MaCV
- Khóa chính: MaNV

- Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF : Đạt do không tồn tại thuộc tính bắt cầu vào khóa
- > Kết luận lược đồ NhanVien đã đạt 3NF

3.4.15. Lược đồ BenhNhan(MaBN, TenBN, DiaChi, GioiTinh, SDT, MaTK, MaDS, MaHS)

- Phụ thuộc hàm: MaBN -> HoTen, DiaChi, GioiTinh, SDT, MaTK, MaDS, MaHS
- Khóa chính: MaBN
- Đánh giá 1NF : Đạt do tất cả thuộc tính lưu giá trị nguyên tố
- Đánh giá 2NF : Đạt do không tồn tại thuộc tính không khóa phụ thuộc không đầy đủ vào thuộc tính khóa
- Đánh giá 3NF : Đạt do không tồn tại thuộc tính bắt cầu vào khóa
- > Kết luận lược đồ BenhNhan đã đạt 3NF

3.5. Lược đồ cơ sở dữ liệu quan hệ sau khi chuẩn hóa thành 3NF



Hình 3. 3 Lược đồ cơ sở dữ liệu quan hệ sau khi chuẩn hóa thành 3NF

3.6. Ràng buộc toàn vẹn

3.6.1. Ràng buộc về giới tính

R1: “Giới tính của nhân viên và bệnh nhân chỉ có thể là nam hoặc nữ”

- Bối Cảnh: NhanVien, BenhNhan

- Ngôn ngữ hình thức: $\forall t \in \text{NhanVien} (t.\text{GioiTinh} \in \{\text{'Nam'}, \text{'Nữ'}\}) , \forall t \in \text{BenhNhan} (t.\text{GioiTinh} \in \{\text{'Nam'}, \text{'Nữ'}\})$

Bảng tầm ảnh hưởng

R1	Thêm	Xóa	Sửa
NhanVien	+	-	+ (GioiTinh)
BenhNhan	+	-	+ (GioiTinh)

3.6.2. Ràng buộc về tuổi của nhân viên

R2: “Tuổi của nhân viên phải lớn hơn 18 tuổi”

- Bối Cảnh: NhanVien
- Ngôn ngữ hình thức: $\forall t \in \text{NhanVien} (t.\text{YEAR}(\text{NgayVaoLam}) - t.\text{YEAR}(\text{NgaySinh}) > 18)$

Bảng tầm ảnh hưởng

R2	Thêm	Xóa	Sửa
NhanVien	+	-	+ (NgaySinh, NgayVaoLam)

3.6.3. Ràng buộc về ngày đặt trong lịch đặt khám

R3: “Ngày đặt lịch phải lớn hơn ngày hiện tại”

- Bối Cảnh: LichDat
- Ngôn ngữ hình thức: $\forall t \in \text{LichDat} (t.\text{NgayDat} > \text{CURRENT_DATE})$

Bảng tầm ảnh hưởng

R3	Thêm	Xóa	Sửa
LichDat	+	-	+ (NgayDat)

3.6.4. Ràng buộc về chức vụ

R4: “Mỗi chức vụ phải có ít nhất một nhân viên”

- Bối Cảnh: ChucVu, NhanVien
- Ngôn ngữ hình thức: $\forall t \in \text{ChucVu}, \exists n \in \text{NhanVien} (t.\text{MaCV} = n.\text{MaCV})$

Bảng tầm ảnh hưởng

R4	Thêm	Xóa	Sửa
ChucVu	+	-	-
NhanVien	-	+	+(MaCV)

3.6.5. Ràng buộc về lương

R5: “Lương trong bảng chức vụ phải lớn hơn hoặc 7 triệu”

- Bối Cảnh: ChucVu
- Ngôn ngữ hình thức: $\forall t \in \text{ChucVu} (t.\text{Luong} \geq 7000000)$

Bảng tầm ảnh hưởng

R5	Thêm	Xóa	Sửa
ChucVu	+	-	+(Luong)

3.6.6. Ràng buộc về ngày lập trong hồ sơ bệnh nhân

R6: “Ngày lập hồ sơ bệnh nhân không được lớn hơn ngày hiện tại”

- Bối Cảnh: HoSoBN
- Ngôn ngữ hình thức: $\forall t \in \text{HoSoBN} (t.\text{NgayLap} \geq \text{CURRENT_DATE})$

Bảng tầm ảnh hưởng

R6	Thêm	Xóa	Sửa
HoSoBN	+	-	+(NgayLap)

3.6.7. Ràng buộc bảng Thuoc với Kho

R7: “Số lượng trong bảng thuốc không được lớn hơn số lượng trong bảng kho”

- Bối Cảnh: Thuoc, Kho

- Ngôn ngữ hình thức: $\forall t \in Thuoc, \exists k \in Kho (t.SL \leq k.SL \wedge t.MaKho = k.MaKho)$

Bảng tầm ảnh hưởng

R7	Thêm	Xóa	Sửa
HoSoBN	+	-	+(NgàyLap)

3.6.8. Ràng buộc về bảng bệnh án

R8: “Ngày khám trong bệnh án không được lớn hơn ngày hiện tại”

- Bối Cảnh: BenhAn
- Ngôn ngữ hình thức: $\forall t \in BenhAN(t.NgayKham \leq CURRENT_DATE)$

Bảng tầm ảnh hưởng

R8	Thêm	Xóa	Sửa
BenhAn	+	-	+(NgàyKham)

3.6.9. Ràng buộc về số điện thoại trong bảng nhân viên và bệnh nhân

R9: “Độ dài của số điện thoại phải bằng 10”

- Bối Cảnh: NhanVien, BenhNhan
- Ngôn ngữ hình thức: $\forall n \in NhanVien, BenhNhan (|n.SDT|= 10)$

Bảng tầm ảnh hưởng

R9	Thêm	Xóa	Sửa
NhanVien	+	-	+(SDT)
BenhNhan	+	-	+(SDT)

3.6.10. Ràng buộc bảng hóa đơn

R10: “Tổng tiền trong bảng hóa đơn phải lớn hơn 100.000”

- Bối Cảnh: HoaDon
- Ngôn ngữ hình thức: $\forall h \in \text{HoaDon} (h.\text{TongTien} \geq 100000)$

Bảng tầm ảnh hưởng

R10	Thêm	Xóa	Sửa
HoaDon	+	-	+(TongTien)

3.6.11. Ràng buộc bảng tài khoản

R11: “Active trong tài khoản chỉ được nhận 2 giá trị là 1 và 0”

- Bối cảnh: TaiKhoan
- Ngôn ngữ hình thức: $\forall h \in \text{TaiKhoan} (h.\text{Active} \in \{1, 0\})$

Bảng tầm ảnh hưởng

R11	Thêm	Xóa	Sửa
TaiKhoan	+	-	+(Active)

3.7. Các thao tác xử lý

- Giao tác Thêm: Giao tác này được sử dụng để thêm dữ liệu mới vào các bảng trong hệ thống.
- Giao tác Sửa: Giao tác này được sử dụng để cập nhật thông tin trong các bảng.
- Giao tác Xóa: Giao tác này được sử dụng để xóa dữ liệu khỏi các bảng.
- Giao tác đặt lịch khám: Giao tác này được sử dụng để đặt lịch khám từ bệnh nhân.

CHƯƠNG 4. TRIỂN KHAI CƠ SỞ DỮ LIỆU TRÊN MS SQL SERVER

4.1. Tạo cơ sở dữ liệu

Bảng khoa

```
CREATE TABLE Khoa
(
    MaKhoa CHAR(5) NOT NULL,
    TenKhoa VARCHAR(20) NOT NULL,
    PRIMARY KEY (MaKhoa)
```



```
);
```

Bảng danh sách khám

```
CREATE TABLE DSKham
(
    MaDS CHAR(5) NOT NULL,
    NgayKham DATE NOT NULL,
    PRIMARY KEY (MaDS)
);
```

Bảng chức vụ

```
CREATE TABLE ChucVu
(
    MaCV CHAR(5) NOT NULL,
    TenCV VARCHAR(20) NOT NULL,
    Luong FLOAT NOT NULL,
    PRIMARY KEY (MaCV)
);
```

Bảng tài khoản

```
CREATE TABLE TaiKhoan
(
    MaTK CHAR(5) NOT NULL,
    Email VARCHAR(55) NOT NULL,
    Password VARCHAR(55) NOT NULL,
    Active INT NOT NULL,
    PRIMARY KEY (MaTK)
);
```

Bảng kho

```
CREATE TABLE Kho
(
    MaKho CHAR(5) NOT NULL,
    TenKho VARCHAR(55) NOT NULL,
    DiaChi VARCHAR(255) NOT NULL,
    Loai VARCHAR(55) NOT NULL,
    SL INT NOT NULL,
    PRIMARY KEY (MaKho)
);
```

Bảng phòng ban

```
CREATE TABLE PhongBan
(
    MaPhong CHAR(5) NOT NULL,
    TenPhong VARCHAR(20) NOT NULL,
    MaKhoa CHAR(5) NOT NULL,
    MaDS CHAR(5) NOT NULL,
    PRIMARY KEY (MaPhong),
    FOREIGN KEY (MaKhoa) REFERENCES Khoa(MaKhoa),
    FOREIGN KEY (MaDS) REFERENCES DSKham(MaDS)
);
```

Bảng nhân viên

```
CREATE TABLE NhanVien
(
    MaNV CHAR(5) NOT NULL,
    NgaySinh DATE,
    NgayVaoLam DATE,
    HoTen VARCHAR(20) NOT NULL,
    DiaChi VARCHAR(55) NOT NULL,
    GioiTinh VARCHAR(20) NOT NULL,
    SDT VARCHAR(20) NOT NULL,
    MaPhong CHAR(5) NOT NULL,
    MaTK CHAR(5) NOT NULL,
    MaCV CHAR(5) NOT NULL,
    PRIMARY KEY (MaNV),
    FOREIGN KEY (MaPhong) REFERENCES PhongBan(MaPhong),
    FOREIGN KEY (MaTK) REFERENCES TaiKhoan(MaTK),
    FOREIGN KEY (MaCV) REFERENCES ChucVu(MaCV)
);
```

Bảng hồ sơ bệnh nhân

```
CREATE TABLE HoSoBN
(
    MaHS CHAR(5) NOT NULL,
    NgayLap DATE NOT NULL,
    MaNV CHAR(5) NOT NULL,
    PRIMARY KEY (MaHS),
    FOREIGN KEY (MaNV) REFERENCES NhanVien(MaNV)
);
```

Bảng thuốc

```
CREATE TABLE Thuoc
```

```
(
    MaThuoc CHAR(5) NOT NULL,
    TenThuoc VARCHAR(55) NOT NULL,
    CachDung VARCHAR(255) NOT NULL,
    DonGia FLOAT NOT NULL,
    SL INT NOT NULL,
    MaKho CHAR(5) NOT NULL,
    PRIMARY KEY (MaThuoc),
    FOREIGN KEY (MaKho) REFERENCES Kho(MaKho)
);
```

Bảng bệnh án

```
CREATE TABLE BenhAn
(
    MaBA CHAR(5) NOT NULL,
    PPDC VARCHAR(255) NOT NULL,
    NgayKham DATE NOT NULL,
    MaHS CHAR(5) NOT NULL,
    PRIMARY KEY (MaBA),
    FOREIGN KEY (MaHS) REFERENCES HoSoBN(MaHS)
);
```

Bảng hóa đơn

```
CREATE TABLE HoaDon
(
    MaHD CHAR(5) NOT NULL,
    TongTien FLOAT NOT NULL,
    MaBA CHAR(5) NOT NULL,
    PRIMARY KEY (MaHD),
    FOREIGN KEY (MaBA) REFERENCES BenhAn(MaBA)
);
```

Bảng hoadon_thuoc

```
CREATE TABLE HoaDDon_Thuoc
(
    MaThuoc CHAR(5) NOT NULL,
    MaHD CHAR(5) NOT NULL,
    PRIMARY KEY (MaThuoc, MaHD),
    FOREIGN KEY (MaThuoc) REFERENCES Thuoc(MaThuoc),
    FOREIGN KEY (MaHD) REFERENCES HoaDon(MaHD)
);
```

Bảng phiếu

```
CREATE TABLE Phieu
(
    MaPhieu CHAR(5) NOT NULL,
    TenPhieu VARCHAR(20) NOT NULL,
    Loai VARCHAR(20) NOT NULL,
    Ngay DATE NOT NULL,
    NoiDung VARCHAR(50) NOT NULL,
    MaNV CHAR(5) NOT NULL,
    PRIMARY KEY (MaPhieu),
    FOREIGN KEY (MaNV) REFERENCES NhanVien(MaNV)
);
```

Bảng bệnh nhân

```
CREATE TABLE BenhNhan
(
    MaBN CHAR(5) NOT NULL,
    TenBN VARCHAR(20) NOT NULL,
    NgaySinh DATE,
    DiaChi VARCHAR(55) NOT NULL,
    CCCD VARCHAR(55) NOT NULL,
    GioiTinh VARCHAR(20) NOT NULL,
    SDT VARCHAR(10) NOT NULL,
    MaTK CHAR(5) NOT NULL,
    MaDS CHAR(5) NOT NULL,
    MaHS CHAR(5) NOT NULL,
    PRIMARY KEY (MaBN),
    FOREIGN KEY (MaTK) REFERENCES TaiKhoan(MaTK),
    FOREIGN KEY (MaDS) REFERENCES DSKham(MaDS),
    FOREIGN KEY (MaHS) REFERENCES HoSoBN(MaHS)
);
```

Bảng đánh giá

```
CREATE TABLE DanhGia
(
    MaDG CHAR(5) NOT NULL,
    Tieude VARCHAR(55) NOT NULL,
    NoiDung VARCHAR(255) NOT NULL,
    MaBN CHAR(5) NOT NULL,
    PRIMARY KEY (MaDG),
    FOREIGN KEY (MaBN) REFERENCES BenhNhan(MaBN)
);
```

```
);
```

Bảng lịch đặt

```
CREATE TABLE LichDat
(
    MaDat CHAR(5) NOT NULL,
    NgayDat DATE NOT NULL,
    MaNV CHAR(5) NOT NULL,
    MaBN CHAR(5) NOT NULL,
    PRIMARY KEY (MaDat),
    FOREIGN KEY (MaNV) REFERENCES NhanVien(MaNV),
    FOREIGN KEY (MaBN) REFERENCES BenhNhan(MaBN)
);
```

4.2. Nhập liệu mẫu

Bảng khoa

```
INSERT INTO Khoa (MaKhoa, TenKhoa) VALUES
('K001', 'Khoa A'),
('K002', 'Khoa B');
```

Bảng danh sách khám

```
INSERT INTO DSKham (MaDS, NgayKham) VALUES
('DS001', '2024-01-01'),
('DS002', '2024-02-01');
```

Bảng chức vụ

```
INSERT INTO ChucVu (MaCV, TenCV, Luong) VALUES
('CV001', 'Bac Si', 20000),
('CV002', 'Y Ta', 15000);
```

Bảng tài khoản

```
INSERT INTO TaiKhoan (MaTK, Email, Password, Active) VALUES
('TK001', 'user1@example.com', 'password1', 1),
('TK002', 'user2@example.com', 'password2', 1);
```

Bảng kho

```
INSERT INTO Kho (MaKho, TenKho, DiaChi, Loai, SL) VALUES
```

```
('KH001', 'Kho Thuoc A', '123 Street', 'Thuoc', 100),
('KH002', 'Kho Vat Tu', '456 Avenue', 'Vat Tu', 200);
```

Bảng phòng ban

```
INSERT INTO PhongBan (MaPhong, TenPhong, MaKhoa, MaDS) VALUES
('P001', 'Phong A', 'K001', 'DS001'),
('P002', 'Phong B', 'K002', 'DS002');
```

Bảng nhân viên

```
INSERT INTO NhanVien (MaNV, NgaySinh, NgayVaoLam, HoTen, DiaChi, GioiTinh, SDT,
MaPhong, MaTK, MaCV) VALUES
('NV001', '1980-01-01', '2020-01-01', 'Nguyen Van A', 'HCMC', 'Nam',
'0901234567', 'P001', 'TK001', 'CV001'),
('NV002', '1985-02-02', '2021-02-02', 'Tran Thi B', 'HN', 'Nu', '0912345678',
'P002', 'TK002', 'CV002');
```

Bảng hồ sơ bệnh nhân

```
INSERT INTO HoSoBN (MaHS, NgayLap, MaNV) VALUES
('HS001', '2024-01-01', 'NV001'),
('HS002', '2024-02-01', 'NV002');
```

Bảng thuốc

```
INSERT INTO Thuoc (MaThuoc, TenThuoc, CachDung, DonGia, SL, MaKho) VALUES
('TH001', 'Thuoc A', 'Uong', 500, 50, 'KH001'),
('TH002', 'Thuoc B', 'Tiem', 1000, 30, 'KH002');
```

Bảng bệnh án

```
INSERT INTO BenhAn (MaBA, PPDC, NgayKham, MaHS) VALUES
('BA001', 'Phuong phap A', '2024-01-01', 'HS001'),
('BA002', 'Phuong phap B', '2024-02-01', 'HS002');
```

Bảng hóa đơn

```
INSERT INTO HoaDon (MaHD, TongTien, MaBA) VALUES
('HD001', 5000, 'BA001'),
('HD002', 10000, 'BA002');
```

Bảng phiếu

```
INSERT INTO Phieu (MaPhieu, TenPhieu, Loai, Ngay, NoiDung, MaNV) VALUES
```

```
('PH001', 'Phieu A', 'Loai A', '2024-01-01', 'Noi dung A', 'NV001'),
('PH002', 'Phieu B', 'Loai B', '2024-02-01', 'Noi dung B', 'NV002');
```

Bảng bệnh nhân

```
INSERT INTO BenhNhan (MaBN, TenBN, NgaySinh, DiaChi, CCCD, GioiTinh, SDT, MaTK,
MaDS, MaHS) VALUES
('BN001', 'Nguyen Van C', '1990-03-03', 'DN', '123456789', 'Nam', '0923456789',
'TK001', 'DS001', 'HS001'),
('BN002', 'Le Thi D', '1995-04-04', 'HP', '987654321', 'Nu', '0934567890',
'TK002', 'DS002', 'HS002');
```

Bảng đánh giá

```
INSERT INTO DanhGia (MaDG, TieuDe, NoiDung, MaBN) VALUES
('DG001', 'Danh gia 1', 'Noi dung 1', 'BN001'),
('DG002', 'Danh gia 2', 'Noi dung 2', 'BN002');
```

Bảng lịch đặt

```
INSERT INTO LichDat (MaDat, NgayDat, MaNV, MaBN) VALUES
('LD001', '2024-03-03', 'NV001', 'BN001'),
('LD002', '2024-04-04', 'NV002', 'BN002');
```

4.3. Cài các ràng buộc toàn vẹn

4.3.1. Ràng buộc về giới tính

```
-- Giới tính nhan vien
CREATE TRIGGER tr_GioiTinh_nhanvien
on NhanVien
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @gioitinh VARCHAR(20)
    DECLARE @manv CHAR(5)
    SELECT @gioitinh = GioiTinh, @manv = MaNV FROM inserted
    if(@gioitinh like 'nam' OR @gioitinh like 'nu')
    BEGIN
        UPDATE NhanVien
        Set GioiTinh = @gioitinh
        WHERE MaNV = @manv
    END
    ELSE
```

```

BEGIN
    RAISERROR(N'Yêu cầu nhập lại giới tính', 16, 1);
    ROLLBACK TRANSACTION;
    RETURN
END
END

--Gioi tinh benh nhan
CREATE TRIGGER tr_GioiTinh_benhnhnan
on BenhNhan
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @gioitinh VARCHAR(20)
    DECLARE @mabn CHAR(5)
    SELECT @gioitinh = GioiTinh, @mabn = MaBN FROM inserted
    if(@gioitinh like 'nam' OR @gioitinh like 'nu')
    BEGIN
        UPDATE BenhNhan
        Set GioiTinh = @gioitinh
        WHERE MaBN = @mabn
    END
    ELSE
    BEGIN
        RAISERROR(N'Yêu cầu nhập lại giới tính', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN
    END
END
END

```

4.3.2. Ràng buộc về tuổi của nhân viên

```

CREATE TRIGGER tr_tuoi_nv
ON NhanVien
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @ngaysinh DATE
    select @ngaysinh = NgaySinh FROM inserted
    IF (YEAR(GETDATE()) - YEAR(@ngaysinh) <= 18)
    BEGIN

```



```

        RAISERROR(N'Tuổi của nhân viên phải lớn hơn 18', 16, 1);
    ROLLBACK TRANSACTION;
    RETURN;
END
END

```

4.3.3. Ràng buộc về ngày đặt trong lịch đặt khám

```

CREATE TRIGGER tr_ngaydat
on LichDat
FOR INSERT, UPDATE
AS
BEGIN
    IF EXISTS (
        SELECT 1
        FROM inserted
        WHERE NgayDat <= GETDATE()
    )
    BEGIN
        RAISERROR(N'Thời gian đặt nhỏ hơn thời gian hiện tại !', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END
END

```

4.3.4. Ràng buộc về chức vụ

```

CREATE TRIGGER tr_check_chucvu
ON NhanVien
AFTER INSERT, UPDATE, DELETE
AS
BEGIN
    IF EXISTS (
        SELECT c.MaCV
        FROM ChucVu c
        LEFT JOIN NhanVien n ON c.MaCV = n.MaCV
        GROUP BY c.MaCV
        HAVING COUNT(n.MaNV) = 0
    )
    BEGIN

```

```

        RAISERROR(N'Mỗi chức vụ phải có ít nhất 1 nhân viên', 16, 1);
    ROLLBACK TRANSACTION;
    RETURN;
END
END

```

4.3.5. Ràng buộc về lương

```

CREATE TRIGGER tr_luong
on ChucVu
FOR INSERT, UPDATE
AS
BEGIN
    DECLARE @luong FLOAT

    IF EXISTS(
        SELECT 1 FROM inserted i
        WHERE i.Luong < 7000000
    )
    BEGIN
        RAISERROR(N'Lương của mỗi chức vụ phải lớn hơn hoặc bằng 7 triệu đồng',
16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END
END

```

4.3.6. Ràng buộc về ngày lập trong hồ sơ bệnh nhân

```

CREATE TRIGGER tr_ngaylap_hoso
on HoSoBN
FOR UPDATE, INSERT
AS
BEGIN
    IF EXISTS (
        SELECT 1
        FROM inserted
        WHERE NgayLap > GETDATE()
    )
    BEGIN
        RAISERROR('ngày lon hon ngay hien tai', 16, 1);
    END

```

```

        ROLLBACK TRANSACTION;
        RETURN;
    END
END

```

4.3.7. Ràng buộc về số lượng trong thuốc và kho

```

GO
CREATE TRIGGER tr_so_luong_thuoc
ON Thuoc
AFTER INSERT, UPDATE
AS
BEGIN
    -- Kiểm tra xem có bất kỳ thuốc nào có số lượng vượt quá số lượng trong kho
    không
    IF EXISTS (
        SELECT 1
        FROM inserted i
        JOIN Kho k ON i.MaKho = k.MaKho
        WHERE i.SL > k.SL
    )
    BEGIN
        RAISERROR(N'Số lượng thuốc không được lớn hơn số lượng trong kho', 16,
1);
        ROLLBACK TRANSACTION;
        RETURN;
    END

    DECLARE @makho CHAR(5), @mathuoc CHAR(5), @sl_hientai INT, @sl INT;

    -- Kiểm tra từng hàng trong bảng inserted
    DECLARE inserted_cursor CURSOR FOR
    SELECT MaKho, MaThuoc, SL FROM inserted;

    OPEN inserted_cursor;

    FETCH NEXT FROM inserted_cursor INTO @makho, @mathuoc, @sl;

    WHILE @@FETCH_STATUS = 0
    BEGIN
        -- Kiểm tra nếu thuốc chưa tồn tại trong kho

```

```

        IF EXISTS (SELECT 1 FROM Thuoc WHERE MaThuoc = @mathuoc AND MaKho =
@makho)
        BEGIN
            PRINT(' da co trong database ')
            SELECT @sl_hientai = ISNULL(SUM(SL), 0) FROM Thuoc WHERE MaKho =
@makho and MaThuoc != @mathuoc ;
            IF @sl_hientai + @sl > (SELECT SL FROM Kho WHERE MaKho = @makho)
            BEGIN
                RAISERROR(N'Vui lòng chỉnh lại số lượng thuốc', 16, 1);
                ROLLBACK TRANSACTION;
                CLOSE inserted_cursor;
                DEALLOCATE inserted_cursor;
                RETURN;
            END
        END
    ELSE
    BEGIN
        SELECT @sl_hientai = ISNULL(SUM(SL), 0) FROM Thuoc WHERE MaKho =
@makho;
        IF @sl_hientai + @sl > (SELECT SL FROM Kho WHERE MaKho = @makho)
        BEGIN
            PRINT(' chua co trong database ')
            RAISERROR(N'Vui lòng chỉnh lại số lượng thuốc', 16, 1);
            ROLLBACK TRANSACTION;
            CLOSE inserted_cursor;
            DEALLOCATE inserted_cursor;
            RETURN;
        END
    END

    FETCH NEXT FROM inserted_cursor INTO @makho, @mathuoc, @sl;
END
CLOSE inserted_cursor;
DEALLOCATE inserted_cursor;
END;

```

4.3.8. Ràng buộc về ngày khám bệnh trong bệnh án không được lớn hơn ngày hiện tại

```

CREATE TRIGGER tr_ngaykham_benhan
ON BenhAn
AFTER INSERT, UPDATE
AS

```

```

BEGIN
    SET NOCOUNT ON;

    DECLARE @errorMessage NVARCHAR(100) = N'Ngày khám bệnh không được lớn hơn
ngày hiện tại!';

    IF EXISTS (
        SELECT 1
        FROM inserted
        WHERE NgayKham > GETDATE()
    )
    BEGIN
        RAISERROR(@errorMessage, 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END
END;

```

4.3.9. Ràng buộc về độ dài điện thoại

Nhân viên

```

CREATE TRIGGER tr_so_dien_thoai_nv
ON NhanVien
AFTER INSERT, UPDATE
AS
BEGIN
    IF EXISTS (
        SELECT 1
        FROM inserted
        WHERE SDT NOT LIKE '[0-9]%' OR LEN(SDT) != 10
    )
    BEGIN
        PRINT N'Số điện thoại của khách hàng phải có định dạng hợp lệ (chỉ chứa
chữ số và có độ dài chính xác 10 ký tự).';
        ROLLBACK TRANSACTION
    END
END

```

Bệnh nhân

```

CREATE TRIGGER tr_so_dien_thoai_bn
ON BenhNhan

```

```

AFTER INSERT, UPDATE
AS
BEGIN
    IF EXISTS (
        SELECT 1
        FROM inserted
        WHERE SDT NOT LIKE '[0-9]%' OR LEN(SDT) != 10
    )
    BEGIN
        PRINT N'Số điện thoại của khách hàng phải có định dạng hợp lệ (chỉ chứa
chữ số và có độ dài chính xác 10 ký tự).'
        ROLLBACK TRANSACTION
    END
END

```

4.3.10. Ràng buộc về hóa đơn

```

CREATE TRIGGER tr_tong_tien_hoa_don
ON HoaDon
AFTER INSERT, UPDATE
AS
BEGIN
    SET NOCOUNT ON;
    -- Kiểm tra xem có bất kỳ hóa đơn nào có tổng tiền <= 100000 không
    IF EXISTS (
        SELECT 1
        FROM inserted
        WHERE TongTien <= 100000
    )
    BEGIN
        RAISERROR(N'Tổng tiền trong bảng hóa đơn phải lớn hơn 100', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END
END

```

4.3.11. Ràng buộc về active trong tài khoản

```

CREATE TRIGGER tr_active_taikhoan
ON TaiKhoan

```

```

for UPDATE, INSERT
AS
BEGIN
    IF EXISTS(
        select 1
        from inserted i
        WHERE i.Active Not in (0,1) )
    BEGIN
        ROLLBACK TRANSACTION;
        RAISERROR ('Active must be 0 or 1.', 16, 1);
        RETURN
    END
END

```

4.4. Cài các store procedure

4.4.1. Store procedure cập nhập thông tin

- Mô tả: Dùng để cập nhập thông tin của bệnh nhân
- Tham số đầu vào: Mã bệnh nhân, tên bệnh nhân, ngày sinh, địa chỉ, căn cước công dân, giới tính, số điện thoại
- Tham số đầu ra:

```

CREATE PROCEDURE sp_capnhapthongtinBN
@mabn char(5),
@tenbn varchar(20),
@ngaysinh DATE,
@diachi varchar(55),
@cccd varchar(55),
@gioitinh VARCHAR(20),
@sdt VARCHAR(10)
AS
BEGIN
    IF NOT EXISTS(
        SELECT 1 FROM BenhNhan WHERE MaBN = @mabn
    )
    BEGIN
        RAISERROR('MaBN không tồn tại, vui lòng kiểm tra lại.', 16, 1);
        RETURN;
    END

    UPDATE BenhNhan
    SET TenBN = @tenbn,

```

```

        NgaySinh = @ngaysinh,
        DiaChi = @diachi,
        CCCD = @cccd,
        GioiTinh = @gioitinh,
        SDT = @sdt
    WHERE MaBN = @mabn
    PRINT('cap nhap thanh cong !')
END

```

4.4.2. Store procedure tìm kiếm thông tin bệnh nhân theo tên

- Mô tả: Dùng để truy xuất dữ liệu thông tin cá nhân của bệnh nhân
- Tham số đầu vào: Tên bệnh nhân
- Tham số đầu ra: thông tin của bệnh nhân

```

CREATE PROCEDURE sp_timkiembenhnhantheten
@tenbn VARCHAR(20)
AS
BEGIN
    SELECT * FROM BenhNhan WHERE TenBN like '%' +@tenbn + '%'
END

```

4.4.3. Store procedure danh bệnh nhân khám theo ngày

- Mô tả: Dùng để truy xuất danh sách bệnh nhân có lịch khám theo ngày truyền vào
- Tham số đầu vào: ngày khám bệnh
- Tham số đầu ra:

```

CREATE PROCEDURE sp_danhsach_theongay
@ngaykham DATE
AS
BEGIN
    DECLARE @mads CHAR(5)
    IF NOT EXISTS(
        select 1 FROM DSKham
        WHERE NgayKham = @ngaykham
    )
    BEGIN
        RAISERROR('Ngay kham khong co benh nhan nao !', 16, 1);
        RETURN;
    END

    SELECT @mads = MaDS FROM DSKham WHERE NgayKham = @ngaykham

```



```
SELECT * FROM BenhNhan WHERE MaDS = @mads
END
```

4.4.4. Store procedure đặt lịch khám

- Mô tả: Dùng để đặt lịch khám bệnh
- Tham số đầu vào: ngày đặt lịch, tên bác sĩ, tên bệnh nhân
- Tham số đầu ra:

```
IF NOT EXISTS (SELECT * FROM sys.sequences WHERE name = 'Sequence_LichHen')
BEGIN
    CREATE SEQUENCE Sequence_LichHen
        AS INT
        START WITH 1
        INCREMENT BY 1;
END
GO

GO
CREATE PROCEDURE sp_datlichkham
    @ngaydat DATE,
    @tenbacsi VARCHAR(20),
    @tenbenhnhan varchar(20)
AS
BEGIN
    if not EXISTS(
        SELECT 1 FROM NhanVien
        WHERE HoTen like @tenbacsi
    )
    BEGIN
        RAISERROR('Bac si khong ton tai !', 16, 1);
        RETURN;
    END

    If not EXISTS(
        SELECT 1 FROM BenhNhan
        WHERE TenBN like @tenbenhnhan
    )
    BEGIN
        RAISERROR('Kiem tra lai ten benh nhan !', 16, 1);
        RETURN;
    END
END
```

```

DECLARE @mabs CHAR(5)
DECLARE @mabn char(5)
DECLARE @malichdat CHAR(5)
SELECT @mabs = MaNV FROM NhanVien WHERE HoTen like @tenbacsi
SELECT @mabn = MaBN FROM BenhNhan WHERE TenBN like @tenbenhnhan
SET @malichdat = 'LD' + RIGHT('000' + CAST((NEXT VALUE FOR Sequence_LichHen)
AS VARCHAR), 3);
INSERT INTO LichDat (MaDat, NgayDat, MaNV, MaBN) VALUES
(@malichdat, @ngaydat, @mabs, @mabn)
END

```

4.4.5. Store procedure hủy lịch đặt

- Mô tả: Dùng để hủy đặt lịch khám bệnh
- Tham số đầu vào: mã lịch đặt
- Tham số đầu ra:

```

CREATE PROCEDURE sp_huylichdat
@malichdat char(5)
AS
BEGIN
    if not EXISTS(
        SELECT 1 FROM LichDat
        WHERE MaDat = @malichdat
    )
    BEGIN
        RAISERROR('Lich dat khong ton tai!', 16, 1);
        RETURN;
    END

    DELETE FROM LichDat
    WHERE MaDat = @malichdat
    PRINT('huy thanh cong!')
END

```

4.4.6. Store procedure sửa bệnh án

- Mô tả: Dùng để sửa các thông tin trên bệnh án
- Tham số đầu vào: mã bệnh án, phương pháp điều trị, ngày khám, mã hồ sơ
- Tham số đầu ra:

```

CREATE PROCEDURE sp_UpdateBenhAn

```

```

@maba char(5),
@ppdc VARCHAR(225),
@ngaykham date,
@mahs char(5)
as
BEGIN
    IF NOT EXISTS(
        SELECT * from BenhAn
        WHERE MaBA = @maba
    )
    BEGIN
        RAISERROR('Benh an khong ton tai!', 16, 1);
        RETURN;
    END

    UPDATE BenhAn
    set PPDC = @ppdc, NgayKham = @ngaykham, MaHS = @mahs
    WHERE MaBA = @maba
END

```

4.4.7. Store procedure Liệt kê danh sách nhân viên theo chức vụ

- Mô tả: truy xuất dữ liệu nhân viên theo chức vụ
- Tham số đầu vào: tên chức vụ
- Tham số đầu ra:

```

CREATE PROCEDURE sp_TimKiemBacSiTheoChuyenKhoa
    @chucvu VARCHAR(20)
AS
BEGIN
    DECLARE @macv CHAR(5)
    SELECT @macv = MaCV FROM ChucVu WHERE TenCV like @chucvu

    SELECT *
    FROM NhanVien
    WHERE MaCV = @macv
END;

```

4.4.8. Store procedure sửa bệnh án

- Mô tả: Dùng để sửa các thông tin trên bệnh án

- Tham số đầu vào: mã bệnh án, phương pháp điều trị, ngày khám, mã hồ sơ
- Tham số đầu ra:

```
CREATE PROCEDURE sp_UpdateBenhAn
@maba char(5),
@ppdc VARCHAR(225),
@ngaykham date,
@mahs char(5)
as
BEGIN
    IF NOT EXISTS(
        SELECT * from BenhAn
        WHERE MaBA = @maba
    )
    BEGIN
        RAISERROR('Benh an khong ton tai!', 16, 1);
        RETURN;
    END

    UPDATE BenhAn
    set PPDC = @ppdc, NgayKham = @ngaykham, MaHS = @mahs
    WHERE MaBA = @maba
END
```

KẾT LUẬN

Kết quả đã thực hiện

Đã thực hiện đầy đủ các bước Phân tích, Thiết kế và cài đặt CSDL, cụ thể như sau:

- Bước phân tích đã tạo ra được sơ đồ ERD để biểu diễn các dữ liệu cần lưu trữ, đáp ứng cho các chức năng của hệ thống
- Bước thiết kế mức quan niệm đã tạo ra được Lược đồ CSDL quan hệ đạt dạng chuẩn 3, đảm bảo lưu trữ dữ liệu hiệu quả

- Bước thiết kế mức Logic đã rà soát các phép kết thường xuyên được sử dụng (và bổ sung một số Index) để tăng tốc độ truy xuất
- Bước thiết kế mức Vật lý: đề tài đã chọn cài đặt CSDL trên Hệ quản trị CSDL SQL Server

Bước cài đặt trên SQL Server:

- Đã tạo ra được 16 Table, trong đó có 16 Table được làm thiết kế Vật lý để lưu trữ dữ liệu kết xuất
- Đã cài đặt 11 Trigger cho các RBTV
- Đã cài đặt 8 Procedure để đáp ứng dữ liệu cho các chức năng của hệ thống

Những vấn đề chưa thực hiện

Bảo mật và quyền riêng tư:

- Đảm bảo rằng tất cả các dữ liệu bệnh nhân được bảo vệ theo các quy định bảo mật như HIPAA hoặc GDPR.
- Thiết lập và kiểm tra các cơ chế mã hóa dữ liệu và kiểm soát truy cập.

Tính mở rộng và hiệu suất hệ thống:

- Kiểm tra khả năng mở rộng của hệ thống để xử lý số lượng người dùng lớn hơn.
- Đánh giá và tối ưu hóa hiệu suất của hệ thống để đảm bảo khả năng xử lý nhanh chóng và ổn định.

Tích hợp AI và ML:

- Phát triển và tích hợp các công cụ tư vấn sức khỏe thông minh sử dụng trí tuệ nhân tạo và máy học để cung cấp các lời khuyên y tế chính xác và kịp thời.

Các nội dung cần bổ sung thêm

Có thể chuyển đổi sang các hệ NoSQL,

Tài liệu tham khảo

Các slide bài giảng môn CSDL nâng cao của Khoa CNTT, HUFLIT

Giáo trình và các trang web tham khảo như Studocu,...

