**BỘ TÀI CHÍNH**

**TRƯỜNG ĐẠI HỌC TÀI CHÍNH – MARKETING**

**KHOA KHOA HỌC DỮ LIỆU**

**🙤🙤 🕮 🙦🙦**

A blue circle with white text

Description automatically generated

**ĐỒ ÁN MÔN HỌC**

**MÔN HỌC: HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**Đề tài:**

**QUẢN LÝ BỆNH VIỆN**

**Giảng viên hướng dẫn:  ThS. Vũ Thị Thanh Hương**

**Sinh viên thực hiện: Trần Thái Sơn**

**Mã số sinh viên: 2221004283**

**Lớp học phần: 2431101068801**

*Tp. Hồ Chí Minh, tháng 11 năm 2024*

**BỘ TÀI CHÍNH**

**TRƯỜNG ĐẠI HỌC TÀI CHÍNH – MARKETING**

**KHOA KHOA HỌC DỮ LIỆU**

A blue circle with white text

Description automatically generated

**ĐỒ ÁN MÔN HỌC**

**MÔN HỌC: HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**Đề tài:**

**QUẢN LÝ BỆNH VIỆN**

**Giảng viên hướng dẫn:  ThS. Vũ Thị Thanh Hương**

**Sinh viên thực hiện: Trần Thái Sơn**

**Mã số sinh viên: 2221004283**

**Lớp học phần: 2431101068801**

*Tp. Hồ Chí Minh, tháng 11 năm 2024*

NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Điểm số:

Điểm chữ:

*Tp. Hồ Chí Minh, ngày… tháng… năm 2024*

*Ký và ghi rõ họ tên*

LỜI CẢM ƠN

Em xin chân thành cảm ơn Giảng viên phụ trách học phần – Thạc sĩ Vũ Thị Thanh Hương cùng các thầy, cô, anh, chị tại trường Đại học Tài Chính – Marketing đã hướng dẫn, chỉ bảo tận tình cho em và trong quá trình thực hiện đồ án. Đây là một dịp để chúng em tiếp cận với thực tiễn công việc cũng như kiểm chứng và vận dụng những vấn đề lý thuyết được học trên giảng đường vào trong các ngữ cảnh hoạt động của một số đơn vị, tổ chức.

Với vốn kiến thức cũng như kinh nghiệm còn rất khiêm tốn và là bước đầu làm quen với công việc nghiên cứu mang tính thực nghiệm thì chắc chắn kết quả đạt được của chúng em cũng không tránh khỏi những hạn chế nhất định. Chúng em rất mong muốn được các Giảng viên, những bạn Sinh viên đi trước hay bất kỳ độc giả nào quan tâm và góp ý để chúng em hoàn thiện hơn cho các đồ án cũng như các nghiên cứu tiếp theo của mình.

Xin kính chúc Thạc sĩ Vũ Thị Thanh Hương cùng tất cả những người đã hỗ trợ và đóng góp ý kiến cho em cùng những người thân của mình lời chúc sức khỏe, hạnh phúc và thành đạt.

Xin chân thành cảm ơn.

*Sinh viên thực hiện*

***Trần Thái Sơn***

**MỤC LỤC**

[NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN i](#_Toc183626794)

[LỜI CẢM ƠN ii](#_Toc183626795)

[DANH MỤC HÌNH ẢNH vii](#_Toc183626796)

[DANH MỤC BẢNG xiii](#_Toc183626797)

[DANH MỤC TỪ VIẾT TẮT xiv](#_Toc183626798)

[DANH MỤC THUẬT NGỮ ANH VIỆT xv](#_Toc183626799)

[CHƯƠNG I: THIẾT KẾ CƠ SỞ DỮ LIỆU 1](#_Toc183626800)

[1.1. Mô hình dữ liệu mức quan niệm 1](#_Toc183626801)

[1.1.1 Mô tả yêu cầu quản lý 1](#_Toc183626802)

[1.1.2 Xác định các tập thực thể và thuộc tính 1](#_Toc183626803)

[1.1.3 Xác định mối quan hệ giữa các tập thực thể 4](#_Toc183626804)

[1.2. Mô hình thực thể - kết hợp 5](#_Toc183626805)

[1.3. Mô hình dữ liệu quan hệ 5](#_Toc183626806)

[1.4. Ràng buộc dữ liệu 6](#_Toc183626807)

[1.4.1 Ràng buộc miền giá trị 6](#_Toc183626808)

[1.4.2 Ràng buộc toàn vẹn liên thuộc tính 8](#_Toc183626809)

[1.4.3 Ràng Buộc duy nhất 9](#_Toc183626810)

[1.5. Cài đặt cơ sở dữ liệu 9](#_Toc183626811)

[1.5.1 Tạo cơ sở dữ liệu quản lý bệnh viện 9](#_Toc183626812)

[1.5.2 Tạo cấu trúc bảng trong SQL Server 10](#_Toc183626813)

[1.5.3 Sơ Đồ Diagram 14](#_Toc183626814)

[1.5.4 Dữ liệu mẫu 14](#_Toc183626815)

[CHƯƠNG II: ỨNG DỤNG 20](#_Toc183626816)

[2.1. Synonym 20](#_Toc183626817)

[2.1.1 Mô tả ứng dụng của Synonym trong hệ thống 20](#_Toc183626818)

[2.1.2 Tạo Synonym cho bảng BenhNhan 20](#_Toc183626819)

[2.1.3 Tạo Synonym cho bảng BenhAn 21](#_Toc183626820)

[2.2. Index 22](#_Toc183626821)

[2.2.1 Mô tả ứng dụng của Index trong hệ thống 22](#_Toc183626822)

[2.2.2 Tạo Index cho cột HoTenBenhNhan trên bảng BenhNhan 23](#_Toc183626823)

[2.2.3 Tạo Index cho cột NgayKham trên bảng LichKham 23](#_Toc183626824)

[2.3. View 24](#_Toc183626825)

[2.3.1 Mô tả ứng dụng trong hệ thống 24](#_Toc183626826)

[2.3.2 Tạo hiện thị thông tin danh sách bệnh nhân cơ bản 25](#_Toc183626827)

[2.3.3 Tạo hiện thị thống kê số lượng bệnh án theo bác sĩ 26](#_Toc183626828)

[2.3.4 Tạo hiện thị danh sách lịch khám đầy đủ thông tin 27](#_Toc183626829)

[2.3.5 Tạo hiện thị danh sách bác sĩ có chuyên môn nội khoa 28](#_Toc183626830)

[2.3.6 Tạo hiện thị danh sách thống kê thuốc bệnh nhân 29](#_Toc183626831)

[2.4. Function 30](#_Toc183626832)

[2.4.1 Mô tả ứng dụng của Function trong hệ thống 30](#_Toc183626833)

[2.4.2 Tạo hàm tính chi phí khám dựa trên chuyên môn của bác sĩ với tham số chuyền vào là mã bác sĩ 30](#_Toc183626834)

[2.4.3 Tạo hàm thống kê bác sĩ có nhiều bệnh án nhất và xuất ra bảng các thông tin bác sĩ 32](#_Toc183626835)

[2.5. Store Procedure 34](#_Toc183626836)

[2.5.1 Mô tả ứng dụng của Store Procedure trong hệ thống 34](#_Toc183626837)

[2.5.2 Tính tổng chi phí khám của một bệnh nhân với mã bệnh nhân là tham số truyền vào 35](#_Toc183626838)

[2.5.3 Hiện thị lịch khám của bệnh nhân với mã bệnh nhân là tham số truyền vào 35](#_Toc183626839)

[2.5.4 Liệt kê các bác sĩ theo chuyên môn với chuyên môn là tham số được truyền vào 36](#_Toc183626840)

[2.5.5 Tính tổng số bệnh nhân đã khám trong một khoảng thời gian với tham số truyền vào ngày bắt đầu và ngày kết thúc 37](#_Toc183626841)

[2.5.6 Thống kê chẩn đoán và đơn thuốc được dùng cho chẩn đoán đó 38](#_Toc183626842)

[2.5.7 Liệt kê các chẩn đoán theo chuyên môn với chuyên môn là tham số truyền vào 40](#_Toc183626843)

[2.6. Trigger 41](#_Toc183626844)

[2.6.1 Mô tả ứng dụng của Trigger trong hệ thống 41](#_Toc183626845)

[2.6.2 Thêm vào bảng BacSi thì mã phòng khám phải phù hợp với chuyên môn bác sĩ 41](#_Toc183626846)

[2.6.3 Khi thêm hoặc thay đổi lịch khám kiểm tra giờ của lịch khám phải nằm trong giờ làm việc từ 7:00 sáng đến 18:30 tối 43](#_Toc183626847)

[2.6.4 Thông báo xóa thành công lịch khám hoặc không tồn tại lịch khám 45](#_Toc183626848)

[2.7. User 46](#_Toc183626849)

[2.7.1 Ứng dụng của User trong hệ thống 46](#_Toc183626850)

[2.7.2 Tạo User cho nhân viên y tá 47](#_Toc183626851)

[2.7.3 Tạo User cho bác sĩ quản lý 48](#_Toc183626852)

[2.7.4 Tạo User cho nhân viên bán thuốc 50](#_Toc183626853)

[2.8. Transaction 52](#_Toc183626854)

[2.8.1 Ứng dụng của Transaction trong hệ thống 52](#_Toc183626855)

[2.8.2 Tạo thủ tục cập nhật liều dùng, số lượng đơn thuốc trong bảng DonThuoc với tham số truyền vào là mã đơn thuốc, mã bệnh án, liều dùng mới và số lượng mới xác định giao dịch hoàn thành hoặc quay lui khi giao dịch có lỗi 53](#_Toc183626856)

[2.8.3 Tạo thủ tục xem phòng khám có những bác sĩ nào xác định giao dịch hoàn thành hoặc quay lui khi giao dịch có lỗi 54](#_Toc183626857)

[CHƯƠNG III: KẾT LUẬN 57](#_Toc183626858)

[3.1. Những điều đã thực hiện được trong đồ án 57](#_Toc183626859)

[3.2. Ưu điểm 57](#_Toc183626860)

[3.3. Nhược điểm 58](#_Toc183626861)

[TÀI LIỆU THAM KHẢO 60](#_Toc183626862)

DANH MỤC HÌNH ẢNH

[Hình I‑1: Mô hình thực thể - kết hợp quản lý bệnh viện 5](#_Toc183626863)

[Hình I‑2: Câu lệnh tạo ràng buộc 1 6](#_Toc183626864)

[Hình I‑3: Câu lệnh tạo ràng buộc 2 7](#_Toc183626865)

[Hình I‑4: Câu lệnh tạo ràng buộc 3 7](#_Toc183626866)

[Hình I‑5: Câu lệnh tạo ràng buộc 4 8](#_Toc183626867)

[Hình I‑6: Câu lệnh tạo ràng buộc 5 8](#_Toc183626868)

[Hình I‑7: Câu lệnh tạo ràng buộc 6 9](#_Toc183626869)

[Hình I‑8: Câu lệnh tạo ràng buộc 7 9](#_Toc183626870)

[Hình I‑9: Câu lệnh tạo cơ sở dữ liệu 10](#_Toc183626871)

[Hình I‑10: Câu lệnh tạo bảng BenhNhan 10](#_Toc183626872)

[Hình I‑11: Câu lệnh tạo bảng PhongKham 11](#_Toc183626873)

[Hình I‑12: Câu lệnh tạo bảng BacSi 11](#_Toc183626874)

[Hình I‑13: Câu lệnh tạo bảng BenhAn 12](#_Toc183626875)

[Hình I‑14: Câu lệnh tạo bảng DonThuoc 13](#_Toc183626876)

[Hình I‑15: Câu lệnh tạo bảng LichKham 14](#_Toc183626877)

[Hình I‑16: Sơ đồ Diagram 14](#_Toc183626878)

[Hình I‑17: Câu lệnh thêm dữ liệu vào bảng 15](#_Toc183626879)

[Hình I‑18: Dữ liệu mẫu bảng BenhNhan 15](#_Toc183626880)

[Hình I‑19: Dữ liệu mẫu bảng PhongKham 16](#_Toc183626881)

[Hình I‑20: Dữ liệu mẫu bảng BacSi 16](#_Toc183626882)

[Hình I‑21: Dữ liệu mẫu bảng BenhAn 17](#_Toc183626883)

[Hình I‑22: Dữ liệu mẫu bảng DonThuoc 18](#_Toc183626884)

[Hình I‑23: Dữ liệu mẫu bảng LichKham 19](#_Toc183626885)

[Hình II‑1: Câu lệnh tạo Synonym 1 20](#_Toc183626886)

[Hình II‑2: Kiểm thử Synonym 1 21](#_Toc183626887)

[Hình II‑3: Kết quả Synonym 1 21](#_Toc183626888)

[Hình II‑4: Câu lệnh tạo Synonym 2 21](#_Toc183626889)

[Hình II‑5: Kiểm thử Synonym 2 22](#_Toc183626890)

[Hình II‑6: Kết quả Synonym 2 22](#_Toc183626891)

[Hình II‑7: Câu lệnh tạo Index 1 23](#_Toc183626892)

[Hình II‑8: Kiểm thử Index 1 23](#_Toc183626893)

[Hình II‑9: Kết quả kiểm thử Index 1 23](#_Toc183626894)

[Hình II‑10: Câu lệnh tạo Index 2 23](#_Toc183626895)

[Hình II‑11: Kiểm thử Index 2 24](#_Toc183626896)

[Hình II‑12: Kết quả kiểm thử Index 2 24](#_Toc183626897)

[Hình II‑13: Câu lệnh tạo View 1 25](#_Toc183626898)

[Hình II‑14: Kiểm thử View 1 26](#_Toc183626899)

[Hình II‑15: Kết quả kiểm thử View 1 26](#_Toc183626900)

[Hình II‑16: Câu lệnh tạo View 2 26](#_Toc183626901)

[Hình II‑17: Kiểm thử View 2 27](#_Toc183626902)

[Hình II‑18: Kết quả kiểm thử View 2 27](#_Toc183626903)

[Hình II‑19: Câu lệnh tạo View 3 27](#_Toc183626904)

[Hình II‑20: Kiểm thử View 3 28](#_Toc183626905)

[Hình II‑21: Kết quả kiểm thử View 3 28](#_Toc183626906)

[Hình II‑22: Câu lệnh tạo View 4 28](#_Toc183626907)

[Hình II‑23: Kiểm thử View 4 28](#_Toc183626908)

[Hình II‑24: Kết quả kiểm thử View 4 29](#_Toc183626909)

[Hình II‑25: câu lệnh tạo View 5 29](#_Toc183626910)

[Hình II‑26: Kiểm thử View 5 29](#_Toc183626911)

[Hình II‑27: Kết quả kiểm thử View 5 30](#_Toc183626912)

[Hình II‑28: Câu lệnh tạo Function 1 31](#_Toc183626913)

[Hình II‑29: Kiểm thử chi phí khám nội khoa 31](#_Toc183626914)

[Hình II‑30: Kết quả kiểm thử chi phí khám nội khoa 31](#_Toc183626915)

[Hình II‑31: Kiểm thử chi phí khám nhi khoa 31](#_Toc183626916)

[Hình II‑32: Kết quả kiểm thử chi phí khám nhi khoa 32](#_Toc183626917)

[Hình II‑33: Kiểm thử chi phí khám răng hàm mặt 32](#_Toc183626918)

[Hình II‑34: Kết quả kiểm thử chi phí khám răng hàm mặt 32](#_Toc183626919)

[Hình II‑35: Câu lệnh tạo Function 2 33](#_Toc183626920)

[Hình II‑36: Kiểm thử Function 2 33](#_Toc183626921)

[Hình II‑37: Kết quả kiểm thử Function 2 34](#_Toc183626922)

[Hình II‑38: Câu lệnh tạo Store Procedure 1 35](#_Toc183626923)

[Hình II‑39: Kiểm thử Store Procedure 1 35](#_Toc183626924)

[Hình II‑40: Kết quả kiểm thử Store Procedure 1 35](#_Toc183626925)

[Hình II‑41: câu lệnh tạo Store Procedure 2 36](#_Toc183626926)

[Hình II‑42: Kiểm thử Store Procedure 2 36](#_Toc183626927)

[Hình II‑43: Kết quả kiểm thử Store Procedure 2 36](#_Toc183626928)

[Hình II‑44: Câu lệnh tạo Store Procedure 3 37](#_Toc183626929)

[Hình II‑45: Kiểm thử Store Procedure 3 37](#_Toc183626930)

[Hình II‑46: Kết quả kiểm thử Store Procedure 3 37](#_Toc183626931)

[Hình II‑47: Câu lệnh tạo Store Procedure 4 38](#_Toc183626932)

[Hình II‑48: Kiểm thử Store Procedure 4 38](#_Toc183626933)

[Hình II‑49: Kết quả kiểm thử Store Procedure 4 38](#_Toc183626934)

[Hình II‑50: Câu lệnh tạo Store Procedure 5 39](#_Toc183626935)

[Hình II‑51: Kiểm thử Store Procedure 5 39](#_Toc183626936)

[Hình II‑52: Kết quả kiểm thử Store Procedure 5 40](#_Toc183626937)

[Hình II‑53: Câu lệnh tạo Store Procedure 6 40](#_Toc183626938)

[Hình II‑54: Kiểm thử Store Procedure 6 40](#_Toc183626939)

[Hình II‑55: Kết quả kiểm thử Store Procedure 6 41](#_Toc183626940)

[Hình II‑56: Câu lệnh tạo Trigger 1 42](#_Toc183626941)

[Hình II‑57: Kiểm thử trường hợp không thành công Trigger 1 42](#_Toc183626942)

[Hình II‑58: Kết quả kiểm thử không thành công Trigger 1 42](#_Toc183626943)

[Hình II‑59: Kiểm thử trường hợp thành công Trigger 1 42](#_Toc183626944)

[Hình II‑60: Kết quả kiểm thử thành công Trigger 1 43](#_Toc183626945)

[Hình II‑61: Câu lệnh tạo Trigger 2 43](#_Toc183626946)

[Hình II‑62: Kiểm thử thêm thành công Trigger 2 43](#_Toc183626947)

[Hình II‑63: Kết quả kiểm thử thêm thành công Trigger 2 43](#_Toc183626948)

[Hình II‑64: Kiểm thử sửa thành công Trigger 2 44](#_Toc183626949)

[Hình II‑65: Kết quả kiểm thử sửa thành công Trigger 2 44](#_Toc183626950)

[Hình II‑66: Kiểm thử thêm không thành công Trigger 2 44](#_Toc183626951)

[Hình II‑67: Kết quả kiểm thử thêm không thành công Trigger 2 44](#_Toc183626952)

[Hình II‑68: Kiểm thử sửa không thành công Trigger 2 44](#_Toc183626953)

[Hình II‑69: Kết quả kiểm thử sửa không thành công Trigger 2 45](#_Toc183626954)

[Hình II‑70: Câu lệnh tạo Trigger 3 45](#_Toc183626955)

[Hình II‑71: Kiểm thử thành công Trigger 3 45](#_Toc183626956)

[Hình II‑72: Kết quả kiểm thử thành công Trigger 3 46](#_Toc183626957)

[Hình II‑73: Kiểm thử không thành công Trigger 3 46](#_Toc183626958)

[Hình II‑74: Kết quả kiểm thử không thành công Trigger 3 46](#_Toc183626959)

[Hình II‑75: Câu lệnh tạo nhóm người dùng NhanVienYTa 47](#_Toc183626960)

[Hình II‑76: Câu lệnh phân quyền cho nhóm người dùng NhanVienYTa 47](#_Toc183626961)

[Hình II‑77: Câu lệnh tạo login NV\_YTa 47](#_Toc183626962)

[Hình II‑78: Câu lệnh tạo người dùng NV\_YTa 47](#_Toc183626963)

[Hình II‑79: Câu lệnh thêm người dùng NV\_YTa vào nhóm người dùng NhanVienYTa 47](#_Toc183626964)

[Hình II‑80: Kiểm thử phân quyền của nhóm người dùng NhanVienYTa 48](#_Toc183626965)

[Hình II‑81: Kiểm thử người dùng đã trong nhóm người dùng NhanVienYTa 48](#_Toc183626966)

[Hình II‑82: Câu lệnh tạo nhóm người dùng BacSiQL 49](#_Toc183626967)

[Hình II‑83: Câu lệnh phân quyền cho nhóm người dùng BacSiQL 49](#_Toc183626968)

[Hình II‑84: Câu lệnh tạo login cho BacSi 49](#_Toc183626969)

[Hình II‑85: Câu lệnh tạo người dùng BacSi 49](#_Toc183626970)

[Hình II‑86: Câu lệnh thêm người dùng BacSi vào nhóm người dùng BacSiQL 49](#_Toc183626971)

[Hình II‑87: Kiểm thử phân quyền của nhóm người dùng BacSiQL 50](#_Toc183626972)

[Hình II‑88: Kiểm thử người dùng đã trong nhóm người dùng BacSiQL 50](#_Toc183626973)

[Hình II‑89: Câu lệnh tạo nhóm người dùng NhanVienBanThuoc 50](#_Toc183626974)

[Hình II‑90: Câu lệnh phân quyền cho nhóm người dùng NhanVienBanThuoc 51](#_Toc183626975)

[Hình II‑91: Câu lệnh tạo login NV\_BanThuoc 51](#_Toc183626976)

[Hình II‑92: Câu lệnh tạo người dùng NV\_BanThuoc 51](#_Toc183626977)

[Hình II‑93: Câu lệnh thêm người dùng NV\_BanThuoc vào nhóm người dùng NhanVienBanThuoc 51](#_Toc183626978)

[Hình II‑94: Kiểm thử phân quyền của nhóm người dùng NhanVienBanThuoc 52](#_Toc183626979)

[Hình II‑95: Kiểm thử người dùng đã trong nhóm người dùng NhanVienBanThuoc 52](#_Toc183626980)

[Hình II‑96: Câu lệnh tạo Store Procedure CapNhatDonThuoc 53](#_Toc183626981)

[Hình II‑97: Kiểm thử không thành công Store Procedure CapNhatDonThuoc 54](#_Toc183626982)

[Hình II‑98: Kết quả kiểm thử không thành công Store Procedure CapNhatDonThuoc 54](#_Toc183626983)

[Hình II‑99: Kiểm thử thành công Store Procedure CapNhatDonThuoc 54](#_Toc183626984)

[Hình II‑100: Kết quả kiểm thử thành công Store Procedure CapNhatDonThuoc 54](#_Toc183626985)

[Hình II‑101: Câu lệnh tạo Store Procedure XemPhongKham 55](#_Toc183626986)

[Hình II‑102: Kiểm thử không thành công Store Procedure XemPhongKham 55](#_Toc183626987)

[Hình II‑103: Kết quả kiểm thử không thành công Store Procedure XemPhongKham 55](#_Toc183626988)

[Hình II‑104: Kiểm thử thành công Store Procedure XemPhongKham 55](#_Toc183626989)

[Hình II‑105: Kết quả kiểm thử thành công Store Procedure XemPhongKham 56](#_Toc183626990)

DANH MỤC BẢNG

[Bảng I‑1: Tập thực thể bệnh nhân 1](#_Toc183613155)

[Bảng I‑2: Tập thực thể bác sĩ 2](#_Toc183613156)

[Bảng I‑3: Tập thực thể phòng khám 2](#_Toc183613157)

[Bảng I‑4: Tập thực thể bệnh án 3](#_Toc183613158)

[Bảng I‑5: Tập thực thể đơn thuốc 3](#_Toc183613159)

[Bảng I‑6: Tập thực thể lịch khám 4](#_Toc183613160)

[Bảng I‑7: Ràng buộc số điện thoại bệnh nhân có độ dài hợp lý 6](#_Toc183613161)

[Bảng I‑8: Ràng buộc số điện thoại bệnh nhân không được trùng nhau 6](#_Toc183613162)

[Bảng I‑9: Ràng buộc số điện thoại bác sĩ không được trùng nhau 7](#_Toc183613163)

[Bảng I‑10: Ràng buộc chuyên môn của bác sĩ phải nằm trong chuyên môn quy định. 7](#_Toc183613164)

[Bảng I‑11: Tình trạng phòng khám phải là một trong các giá trị cố định 8](#_Toc183613165)

[Bảng I‑12: Ràng buộc số lượng thuốc phải luôn lớn hơn 0 8](#_Toc183613166)

[Bảng I‑13: Ràng buộc cột MaBacSi, NgayKham, và GioKham phải duy nhất 9](#_Toc183613167)

DANH MỤC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt** | **Ý nghĩa đầy đủ** |
| 1 | SQL | Structured Query Language |
| 2 | ERD | Entity-Relationship Diagram |
| 3 | DBMS | Database Management System |
| 4 | NVARCHAR | National Variable Character |
| 5 | INT | Integer |
| 6 | SP | Stored Procedure |
| 7 | PK | Primary Key |
| 8 | FK | Foreign Key |
| 9 | CRUD | Create, Read, Update, Delete |
| 10 | RAM | Random Access Memory |

DANH MỤC THUẬT NGỮ ANH VIỆT

|  |  |  |
| --- | --- | --- |
| **STT** | **Thuật ngữ Anh** | **Nghĩa tiếng Việt** |
| 1 | Entity-Relationship Diagram | Sơ đồ Thực thể - Quan hệ |
| 2 | Attribute | Thuộc tính |
| 3 | Structured Query Language | Ngôn ngữ truy vấn cấu trúc |
| 4 | Database Management System | Hệ quản trị cơ sở dữ liệu |
| 5 | National Variable Character | Kiểu chuỗi biến đổi hỗ trợ Unicode |
| 6 | Integer | Kiểu số nguyên |
| 7 | Stored Procedure | Thủ tục Lưu trữ |
| 8 | Primary Key | Khóa chính |
| 9 | Foreign Key | Khóa ngoại |
| 10 | Random Access Memory | Bộ nhớ truy cập ngẫu nhiên |
| 11 | Trigger | Kích hoạt |
| 12 | Index | Chỉ mục |
| 13 | View | Bảng ảo |
| 14 | Constraint | Ràng buộc |
| 15 | Synonym | Tên đồng nghĩa |
| 16 | Transaction | Giao dịch |

# THIẾT KẾ CƠ SỞ DỮ LIỆU

## Mô hình dữ liệu mức quan niệm

Để xây dựng được mô hình dữ liệu mức quan niệm, đầu tiên cần xác định các tập thực thể từ quy trình nghiệp vụ của hệ thống. Dựa vào phân tích ban đầu, nhóm đã xây dựng sơ đồ thực thể - kết hợp gồm các thành phần sau:

### Mô tả yêu cầu quản lý

Hệ thống quản lý bệnh viện bao gồm các đối tượng chính như sau: Bệnh nhân, Bác sĩ, Phòng khám, Bệnh án, Đơn thuốc, và Lịch khám.

### Xác định các tập thực thể và thuộc tính

**Tập thực thể bệnh nhân:**

Bảng ‑: Tập thực thể bệnh nhân

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** | **Chiều dài** | **Ràng buộc** |
| MaBenhNhan | Mã bệnh nhân | Int |  | Khóa chính |
| HoTenBenhNhan | Họ tên bệnh nhân | Nvarchar | 100 |  |
| NgaySinh | Ngày sinh | Date |  |  |
| SoDienThoai | Số điện thoại | Nvarchar | 15 |  |
| DiaChi | Địa chỉ | Nvarchar | 255 |  |

**Tập thực thể bác sĩ:**

Bảng ‑: Tập thực thể bác sĩ

| **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** | **Chiều dài** | **Ràng buộc** |
| --- | --- | --- | --- | --- |
| MaBacSi | Mã bác sĩ | Int |  | Khóa chính |
| HoTenBacSi | Họ tên bác sĩ | Nvarchar | 100 |  |
| ChuyenMon | Chuyên môn | Nvarchar | 50 |  |
| SoDienThoai | Số điện thoại | Nvarchar | 15 |  |
| MaPhongKham | Mã phòng khám | Int |  | Khóa ngoại |

**Tập thực thể phòng khám:**

Bảng ‑: Tập thực thể phòng khám

| **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** | **Chiều dài** | **Ràng buộc** |
| --- | --- | --- | --- | --- |
| MaPhongKham | Mã phòng khám | Int |  | Khóa chính |
| TenPhongKham | Tên phòng khám | Nvarchar | 100 |  |
| LoaiPhongKham | Loại phòng khám | Nvarchar | 50 |  |
| TinhTrangPhongKham | Tình trạng phòng khám | Nvarchar | 20 |  |

**Tập thực thể bệnh án:**

Bảng ‑: Tập thực thể bệnh án

| **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** | **Chiều dài** | **Ràng buộc** |
| --- | --- | --- | --- | --- |
| MaBenhAn | MaBenhAn | Int |  | Khóa chính |
| MaBenhNhan | Mã bệnh nhân | Int |  | Khóa ngoại |
| MaBacSi | Mã bác sĩ | Int |  | Khóa ngoại |
| ChanDoan | Chẩn đoán | Nvarchar | 200 |  |
| NgayLapBenhAn | Ngày lập bệnh án | Date |  |  |

**Tập thực thể đơn thuốc:**

Bảng ‑: Tập thực thể đơn thuốc

| **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** | **Chiều dài** | **Ràng buộc** |
| --- | --- | --- | --- | --- |
| MaDonThuoc | Mã đơn thuốc | Int |  | Khóa chính |
| MaBenhAn | MaBenhAn | Int |  | Khóa ngoại |
| TenThuoc | Tên thuốc | Nvarchar | 100 |  |
| LieuDung | Liều dùng | Nvarchar | 50 |  |
| SoLuong | Số lượng | Int |  |  |

**Tập thực thể lịch khám:**

Bảng ‑: Tập thực thể lịch khám

| **Tên thuộc tính** | **Diễn giải** | **Kiểu dữ liệu** | **Chiều dài** | **Ràng buộc** |
| --- | --- | --- | --- | --- |
| MaLichKham | Mã đơn thuốc | Int |  | Khóa chính |
| MaBenhNhan | Mã bệnh án | Int |  | Khóa ngoại |
| MaBacSi | Mã bác sĩ | Int |  | Khóa ngoại |
| NgayKham | Ngày khám | Date |  |  |
| GioKham | Giờ khám | Datetime | 50 |  |

### Xác định mối quan hệ giữa các tập thực thể

Bệnh Nhân - Bệnh Án: Một bệnh nhân có thể có nhiều bệnh án, mỗi bệnh án chỉ thuộc về một bệnh nhân.

* Mối quan hệ: Một - Nhiều.

Bác Sĩ - Phòng Khám: Một bác sĩ chỉ thuộc về một phòng khám, nhưng một phòng khám có thể có nhiều bác sĩ.

* Mối quan hệ: Một - Nhiều.

Bệnh Án - Đơn Thuốc: Một bệnh án có thể có nhiều đơn thuốc, mỗi đơn thuốc chỉ thuộc về một bệnh án.

* Mối quan hệ: Một - Nhiều.

Lịch Khám - Bệnh Nhân: Một lịch khám có chỉ thuộc về một bệnh nhân, nhưng một bệnh nhân có thể có nhiều lịch khám.

* Mối quan hệ: Một - Nhiều.

Lịch Khám - Bác sĩ: Một lịch khám có chỉ thuộc về một bác sĩ, nhưng một bác sĩ có thể có nhiều lịch khám.

* Mối quan hệ: Một - Nhiều.

## Mô hình thực thể - kết hợp

A screenshot of a computer

Description automatically generated

Hình ‑: Mô hình thực thể - kết hợp quản lý bệnh viện

## Mô hình dữ liệu quan hệ

Từ mô hình dữ liệu ERD Quản lý bệnh viện mức quan niệm chuyển thành mô hình quan hệ được thể hiện qua lược đồ quan hệ sau đây:

BENHNHAN (**MaBenhNhan**, HoTenBenhNhan, NgaySinh, SoDienThoai, DiaChi)

BACSI (**MaBacSi**, HoTenBacSi, ChuyenMon, SoDienThoai, MaPhongKham)

PHONGKHAM (**MaPhongKham**, TenPhongKham, LoaiPhongKham, TinhTrangPhongKham)

BENHAN (**MaBenhAn**, ChanDoan, NgayLapBenhAn, MaBenhNhan, MaBacSi)

DONTHUOC (**MaDonThuoc**, TenThuoc, LieuDung, SoLuong, MaBenhAn)

LICHKHAM (**MaLichKham**, NgayKham, GioKham, MaBenhNhan, MaBacSi)

**Chú thích:**

+ In đậm, gạch chân: Khóa chính

+ Gạch chân nét đứt: Khóa ngoại

## Ràng buộc dữ liệu

### Ràng buộc miền giá trị

Ràng Buộc: Số điện thoại trong thông tin bệnh nhân phải có độ dài hợp lý (từ 10 đến 15 ký tự).

Bối cảnh: BenhNhan

Biểu diễn:

∀n ∈ BenhNhan (LEN(n.SoDienThoai) ≥ 10 ∧ LEN(n.SoDienThoai) ≤ 15)

Bảng tầm ảnh hưởng:

Bảng ‑: Ràng buộc số điện thoại bệnh nhân có độ dài hợp lý

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên bảng** | **Thêm** | **Xóa** | **Sửa** |
| BenhNhan | + | - | + (SoDienThoai) |

Câu lệnh tạo ràng buộc:



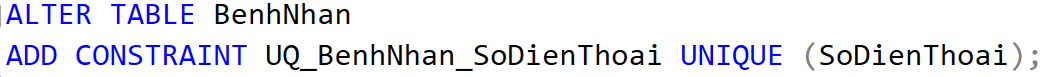
Hình ‑: Câu lệnh tạo ràng buộc 1

Ràng Buộc: Số điện thoại của bệnh nhân không được trùng nhau.  
Bối cảnh: BenhNhan  
Biểu diễn:  
∀n1 ∈ BenhNhan, ∀n2 ∈ BenhNhan (n1 ≠ n2 ∧ n1.SoDienThoai ≠ n2.SoDienThoai)  
Bảng tầm ảnh hưởng:

Bảng ‑: Ràng buộc số điện thoại bệnh nhân không được trùng nhau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên bảng** | **Thêm** | **Xóa** | **Sửa** |
| BenhNhan | + | - | + (SoDienThoai) |

Câu lệnh tạo ràng buộc:



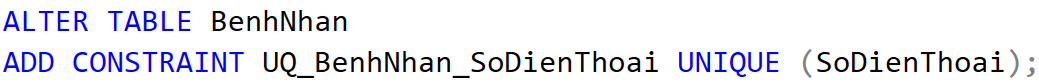
Hình ‑: Câu lệnh tạo ràng buộc 2

Ràng Buộc: Số điện thoại của bác sĩ không được trùng nhau.  
Bối cảnh: BacSi  
Biểu diễn:  
∀n1 ∈ BacSi, ∀n2 ∈ BacSi (n1 ≠ n2 ∧ n1.SoDienThoai ≠ n2.SoDienThoai)  
Bảng tầm ảnh hưởng:

Bảng ‑: Ràng buộc số điện thoại bác sĩ không được trùng nhau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên bảng** | **Thêm** | **Xóa** | **Sửa** |
| BacSi | + | - | + (SoDienThoai) |

Câu lệnh tạo ràng buộc:



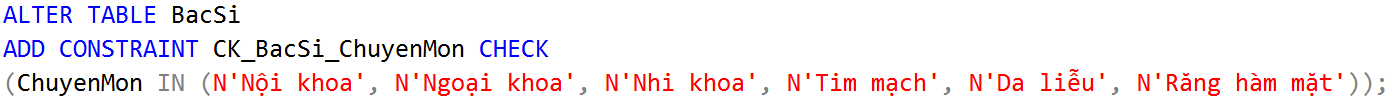
Hình ‑: Câu lệnh tạo ràng buộc 3

Ràng Buộc: Chuyên môn của bác sĩ phải nằm trong danh sách các chuyên môn quy định.  
Bối cảnh: BacSi  
Biểu diễn:  
∀n ∈ BacSi (n.ChuyenMon ∈ {‘Nội khoa’, ‘Ngoại khoa’, ‘Nhi khoa’, ‘Tim mạch’, ‘Da liễu’, ‘Răng hàm mặt’})  
Bảng tầm ảnh hưởng:

Bảng ‑: Ràng buộc chuyên môn của bác sĩ phải nằm trong chuyên môn quy định.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên bảng** | **Thêm** | **Xóa** | **Sửa** |
| BacSi | + | - | + (ChuyenMon) |

Câu lệnh tạo ràng buộc:



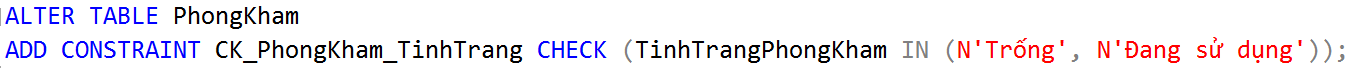
Hình ‑: Câu lệnh tạo ràng buộc 4

Ràng Buộc: Tình trạng phòng khám phải là một trong các giá trị cố định.  
Bối cảnh: PhongKham  
Biểu diễn:  
∀n ∈ PhongKham (n.TinhTrangPhongKham ∈ {‘Trống’, ‘Đang sử dụng’})  
Bảng tầm ảnh hưởng:

Bảng ‑: Tình trạng phòng khám phải là một trong các giá trị cố định

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên bảng** | **Thêm** | **Xóa** | **Sửa** |
| PhongKham | + | - | + (TinhTrangPhongKham) |

Câu lệnh tạo ràng buộc:



Hình ‑: Câu lệnh tạo ràng buộc 5

### Ràng buộc toàn vẹn liên thuộc tính

Ràng Buộc: Trong mỗi đơn thuốc, số lượng thuốc phải luôn lớn hơn 0.  
Bối cảnh: DonThuoc  
Biểu diễn:  
∀n ∈ DonThuoc (n.SoLuong > 0)  
Bảng tầm ảnh hưởng:

Bảng ‑: Ràng buộc số lượng thuốc phải luôn lớn hơn 0

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên bảng** | **Thêm** | **Xóa** | **Sửa** |
| DonThuoc | + | - | + (SoLuong) |

Câu lệnh tạo ràng buộc

A black text with black letters

Description automatically generated

Hình ‑: Câu lệnh tạo ràng buộc 6

### Ràng Buộc duy nhất

**Ràng buộc:** Kết hợp của các cột MaBacSi, NgayKham, và GioKham phải duy nhất

**Bối cảnh:** LichKham.

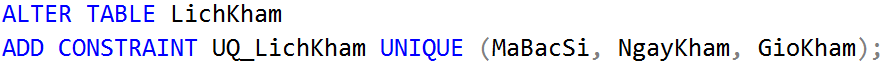
**Biểu diễn:**

∀ n1, n2 ∈ LichKham, n1 ≠ n2 ⟹ (n1.MaBacSi ≠ n2.MaBacSi) ∨ (n1.NgayKham ≠ n2.NgayKham) ∨ (n1.GioKham ≠ n2.GioKham)

Bảng ‑: Ràng buộc cột MaBacSi, NgayKham, và GioKham phải duy nhất

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên bảng** | **Thêm** | **Xóa** | **Sửa** |
| LichKham | + | - | + (MaBacSi, NgayKham, GioKham) |

Câu lệnh tạo ràng buộc:



Hình ‑: Câu lệnh tạo ràng buộc 7

## Cài đặt cơ sở dữ liệu

### Tạo cơ sở dữ liệu quản lý bệnh viện

Tạo cơ sở dữ liệu quản lý bệnh viện

CREATE DATABASE QuanLyBenhVien

USE QuanLyBenhVien

A close up of a sign

Description automatically generated

Hình ‑: Câu lệnh tạo cơ sở dữ liệu

### Tạo cấu trúc bảng trong SQL Server

**Tạo bảng bệnh nhân**

CREATE TABLE BenhNhan (

MaBenhNhan INT constraint PK\_BENHNHAN PRIMARY KEY(MaBenhNhan),

HoTenBenhNhan NVARCHAR(100) NOT NULL,

NgaySinh DATE NOT NULL,

SoDienThoai NVARCHAR(15) NOT NULL,

DiaChi NVARCHAR(255)

);

A close-up of a computer screen

Description automatically generated

Hình ‑: Câu lệnh tạo bảng BenhNhan

**Tạo bảng phòng khám**

CREATE TABLE PhongKham (

MaPhongKham INT constraint PK\_PHONGKHAM PRIMARY KEY(MaPhongKham),

TenPhongKham NVARCHAR(100) NOT NULL,

LoaiPhongKham NVARCHAR(50) NOT NULL,

TinhTrangPhongKham NVARCHAR(20) NOT NULL

);

A close-up of a computer screen

Description automatically generated

Hình ‑: Câu lệnh tạo bảng PhongKham

Tạo bảng bác sĩ

CREATE TABLE BacSi (

MaBacSi INT constraint PK\_BACSI PRIMARY KEY(MaBacSi),

MaPhongKham INT,

HoTenBacSi NVARCHAR(100) NOT NULL,

ChuyenMon NVARCHAR(100) NOT NULL,

SoDienThoai NVARCHAR(15) NOT NULL,

CONSTRAINT FK\_BACSI\_MAPHONGKHAM FOREIGN KEY (MaPhongKham) REFERENCES PhongKham(MaPhongKham)

);

A group of text on a white background

Description automatically generated

Hình ‑: Câu lệnh tạo bảng BacSi

Tạo bảng bệnh án

CREATE TABLE BenhAn (

MaBenhAn INT constraint PK\_BENHAN PRIMARY KEY(MaBenhAn),

MaBenhNhan INT,

MaBacSi INT,

ChanDoan NVARCHAR(200),

NgayLapBenhAn DATE NOT NULL,

CONSTRAINT FK\_BENHAN\_MABENHNHAN FOREIGN KEY (MaBenhNhan) REFERENCES BenhNhan(MaBenhNhan),

CONSTRAINT FK\_BENHAN\_MABACSI FOREIGN KEY (MaBacSi) REFERENCES BacSi(MaBacSi)

);

A close-up of a computer code

Description automatically generated

Hình ‑: Câu lệnh tạo bảng BenhAn

Tạo bảng đơn thuốc

CREATE TABLE DonThuoc (

MaDonThuoc INT constraint PK\_DONTHUOC PRIMARY KEY(MaDonThuoc),

MaBenhAn INT,

TenThuoc NVARCHAR(100) NOT NULL,

LieuDung NVARCHAR(50)NOT NULL,

SoLuong INT NOT NULL,

constraint FK\_DONTHUOC\_MABENHAN FOREIGN KEY (MaBenhAn) REFERENCES BenhAn(MaBenhAn)

);

A close-up of a computer code

Description automatically generated

Hình ‑: Câu lệnh tạo bảng DonThuoc

Tạo bảng lịch khám

CREATE TABLE LichKham (

MaLichKham INT constraint PK\_LICHKHAM PRIMARY KEY(MaLichKham),

MaBenhNhan INT,

MaBacSi INT,

NgayKham DATE NOT NULL,

GioKham TIME NOT NULL,

constraint FK\_LICHKHAM\_MABENHNHAN FOREIGN KEY (MaBenhNhan) REFERENCES BenhNhan(MaBenhNhan),

constraint FK\_LICHKHAM\_MABACSI FOREIGN KEY (MaBacSi) REFERENCES BacSi(MaBacSi)

);

A close-up of a computer code

Description automatically generated

Hình ‑: Câu lệnh tạo bảng LichKham

### Sơ Đồ Diagram

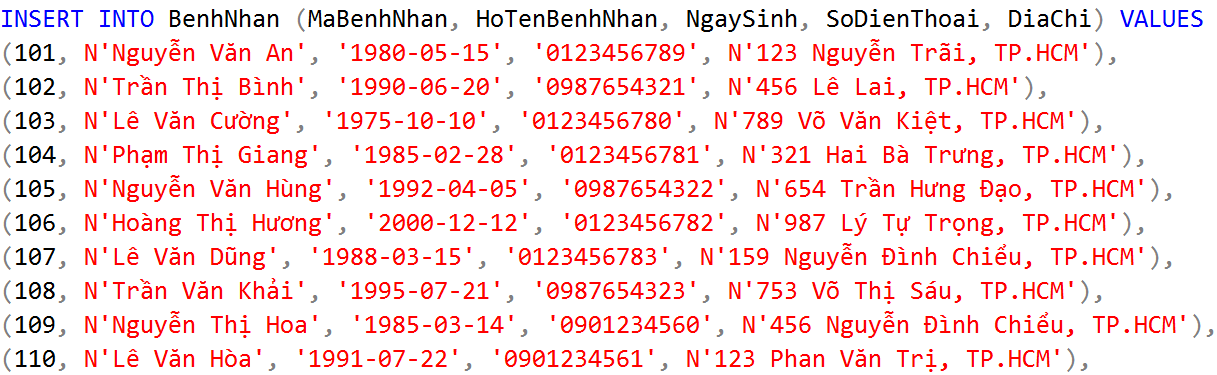
A screenshot of a computer

Description automatically generated

Hình ‑: Sơ đồ Diagram

### Dữ liệu mẫu

**Sử dụng câu lệnh:**

Insert Into tên bảng (tên cột 1, tên cột 2, tên cột 3…) Values (cột 1, cột 2, cột 3…)

Hình ‑: Câu lệnh thêm dữ liệu vào bảng

Tương tự như vậy với các bảng BenhNhan, BacSi, BenhAn, DonThuoc, LichKham, PhongKham

**Bảng bệnh nhân**



Hình ‑: Dữ liệu mẫu bảng BenhNhan

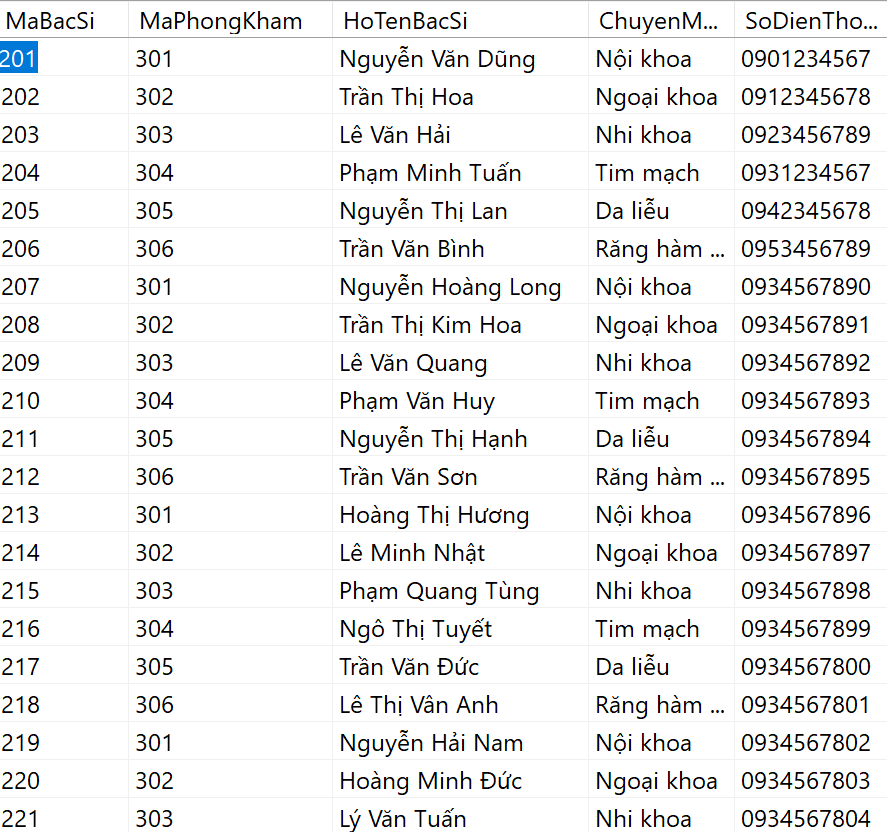
**Bảng phòng khám**

A screenshot of a white background with black text

Description automatically generated

Hình ‑: Dữ liệu mẫu bảng PhongKham

**Bảng bác sĩ**



Hình ‑: Dữ liệu mẫu bảng BacSi

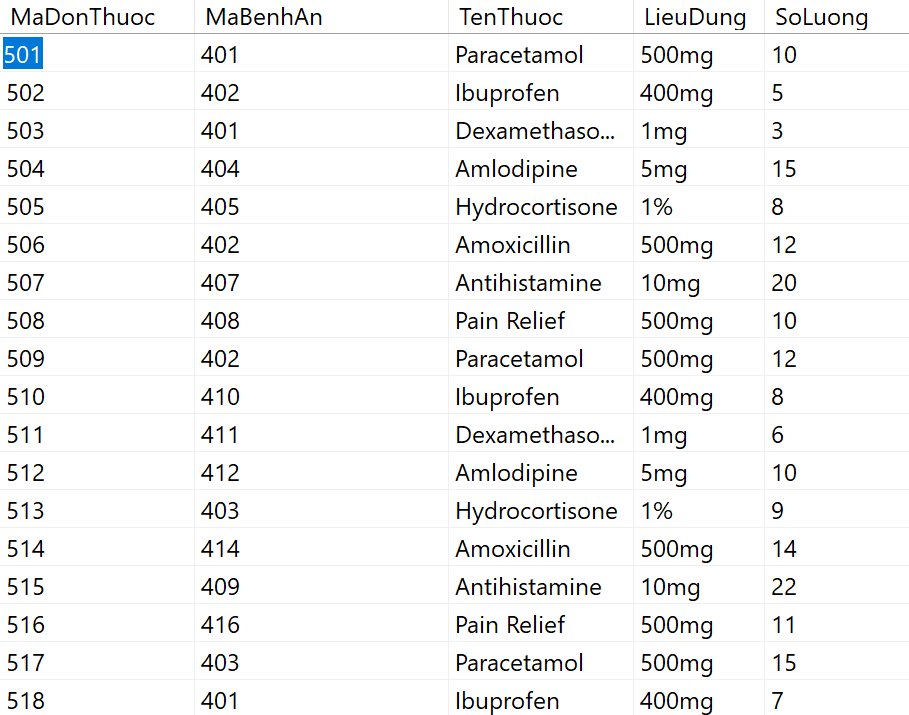
**Bảng bệnh án**

A screenshot of a computer

Description automatically generated

Hình ‑: Dữ liệu mẫu bảng BenhAn

**Bảng đơn thuốc**



Hình ‑: Dữ liệu mẫu bảng DonThuoc

**Bảng lịch khám**

A screenshot of a table

Description automatically generated

Hình ‑: Dữ liệu mẫu bảng LichKham

# ỨNG DỤNG

## Synonym

### Mô tả ứng dụng của Synonym trong hệ thống

Khi người dùng cần truy xuất đến các đối tượng cơ sở dữ liệu thuộc quyền sở hữu của người dùng khác, họ thường phải chỉ định tường minh tên đầy đủ của đối tượng, bao gồm cả tên người sở hữu. Điều này có thể dẫn đến sự phức tạp và dài dòng trong các truy vấn SQL, đặc biệt khi hệ thống cơ sở dữ liệu có nhiều bảng, view hoặc thủ tục lưu trữ với các tên chi tiết. Để đơn giản hóa quá trình này, người dùng có thể tạo các tên đồng nghĩa (synonym) đóng vai trò như một tên thay thế cho các đối tượng đó. Synonym không chỉ giúp rút ngắn truy vấn mà còn cung cấp một cách tiếp cận linh hoạt để tham chiếu đến các đối tượng thuộc sở hữu của người dùng khác mà không cần phải luôn chỉ định tường minh tên người sở hữu.

Hơn nữa, việc sử dụng synonym còn góp phần tăng cường khả năng quản lý và bảo mật dữ liệu. Người dùng có thể thiết lập synonym để kiểm soát việc truy cập vào các đối tượng cơ sở dữ liệu cụ thể, giới hạn quyền trực tiếp trên các đối tượng gốc. Thay vì cho phép truy cập trực tiếp, synonym đóng vai trò như một lớp trung gian, cho phép người dùng truy cập qua tên thay thế mà không làm lộ thông tin chi tiết về cấu trúc hoặc vị trí thực tế của đối tượng. Điều này đặc biệt hữu ích trong các hệ thống cơ sở dữ liệu lớn và phức tạp, nơi các yêu cầu về bảo mật và quản lý dữ liệu được đặt lên hàng đầu.

### Tạo Synonym cho bảng BenhNhan

Tạo Synonym cho bảng BenhNhan với tên Syn\_BenhNhan

A black text with a white background

Description automatically generated

Hình ‑: Câu lệnh tạo Synonym 1

Kiểm thử

A black and blue text

Description automatically generated

Hình ‑: Kiểm thử Synonym 1

Kết quả

A screenshot of a computer

Description automatically generated

Hình ‑: Kết quả Synonym 1

### Tạo Synonym cho bảng BenhAn

Tạo Synonym cho bảng BenhAn với tên Syn\_BenhAn

A close up of a word

Description automatically generated with medium confidence

Hình ‑: Câu lệnh tạo Synonym 2

Kiểm thử

A close-up of a text

Description automatically generated

Hình ‑: Kiểm thử Synonym 2

Kết quả

A screenshot of a computer

Description automatically generated

Hình ‑: Kết quả Synonym 2

## Index

### Mô tả ứng dụng của Index trong hệ thống

Index trong SQL là bảng tra cứu đặc biệt mà công cụ tìm kiếm cơ sở dữ liệu có thể sử dụng để tăng nhanh thời gian và hiệu suất truy xuất dữ liệu.

Hiểu đơn giản, một chỉ mục là một con trỏ chỉ tới từng giá trị xuất hiện trong bảng, cột được đánh chỉ mục. Chỉ mục trong Database có ý nghĩa tương tự như các mục xuất hiện trong Mục lục của một cuốn sách.

INDEX giúp tăng tốc các truy vấn SELECT chứa các mệnh đề WHERE hoặc ORDER, nhưng nó làm chậm việc dữ liệu nhập vào với các lệnh UPDATE và INSERT. Các chỉ mục có thể được tạo hoặc xóa mà không ảnh hưởng tới dữ liệu.

### Tạo Index cho cột HoTenBenhNhan trên bảng BenhNhan

Tạo Index trên cột HoTenBenhNhan để tối ưu hóa tìm kiếm

A close up of a logo

Description automatically generated

Hình ‑: Câu lệnh tạo Index 1

Truy vấn sử dụng Index HoTenBenhNhan

A close up of a text

Description automatically generated

Hình ‑: Kiểm thử Index 1

Kết quả

A number on a white background

Description automatically generated

Hình ‑: Kết quả kiểm thử Index 1

### Tạo Index cho cột NgayKham trên bảng LichKham

Tạo Index trên cột NgayKham của bảng LichKham để tối ưu hóa truy vấn lịch khám

A close up of a text

Description automatically generated

Hình ‑: Câu lệnh tạo Index 2

Truy vấn sử dụng Index NgayKham

A close up of a number

Description automatically generated

Hình ‑: Kiểm thử Index 2

Kết quả

A screenshot of a table

Description automatically generated

Hình ‑: Kết quả kiểm thử Index 2

## View

### Mô tả ứng dụng trong hệ thống

View trong SQL Server được xem như một bảng ảo, nơi dữ liệu của nó không được lưu trữ độc lập mà được truy xuất động từ các bảng nguồn thông qua các truy vấn SQL được định nghĩa trước. Điều này mang lại nhiều lợi ích trong việc tổ chức và xử lý dữ liệu, đặc biệt trong các hệ thống phức tạp. View thường được sử dụng cho nhiều mục đích quan trọng như sau:

* View giúp tập trung vào một số dữ liệu nhất định. Khi làm việc với các hệ thống lớn, việc chỉ hiện thị những dữ liệu mà người dùng quan tâm hoặc chịu trách nhiệm sẽ giúp đơn giản hóa giao diện và tăng hiệu quả làm việc. View có thể được thiết kế để chỉ truy xuất một phần dữ liệu cụ thể, loại bỏ những thông tin không liên quan, giúp người dùng tập trung vào nhiệm vụ của mình mà không bị phân tâm.
* View có thể đơn giản việc xử lý dữ liệu. Những truy vấn phức tạp, bao gồm nhiều phép nối (JOIN), lọc dữ liệu (WHERE), hoặc tính toán, có thể được đóng gói bên trong View. Sau đó, View hoạt động giống như một bảng thông thường, cho phép người dùng thực hiện các thao tác truy vấn hoặc xử lý dữ liệu mà không cần quan tâm đến logic phức tạp của truy vấn gốc. Điều này không chỉ giảm thiểu lỗi mà còn tăng tốc độ phát triển và bảo trì ứng dụng.
* View hỗ trợ tốt cho việc tùy chỉnh dữ liệu. Trong một hệ thống, dữ liệu thường được chia sẻ giữa nhiều người dùng hoặc bộ phận khác nhau, mỗi bên có thể có yêu cầu riêng về cách hiện thị dữ liệu. View cho phép tạo ra các góc nhìn khác nhau trên cùng một nguồn dữ liệu, giúp đáp ứng linh hoạt các nhu cầu đặc thù mà không ảnh hưởng đến dữ liệu gốc. Chẳng hạn, một bộ phận tài chính có thể cần xem các báo cáo chi tiết, trong khi bộ phận bán hàng chỉ cần thấy các tổng quan.
* View còn đóng vai trò quan trọng trong việc tăng tính bảo mật. Bằng cách sử dụng View, người quản trị có thể kiểm soát quyền truy cập dữ liệu, chỉ cho phép người dùng xem những thông tin cần thiết mà không có quyền truy cập trực tiếp vào bảng nguồn. Điều này giúp bảo vệ dữ liệu nhạy cảm và giảm rủi ro bảo mật.

### Tạo hiện thị thông tin danh sách bệnh nhân cơ bản

Tạo View View\_DanhSachBenhNhan hiện thị các thông tin: mã bệnh nhân, họ tên bệnh nhân, chẩn đoán, ngày sinh, số điện thoại

A close up of a text

Description automatically generated

Hình ‑: Câu lệnh tạo View 1

Truy vấn danh sách bệnh nhân

A black text on a white background

Description automatically generated

Hình ‑: Kiểm thử View 1

Kết quả

A screenshot of a white background with black text

Description automatically generated

Hình ‑: Kết quả kiểm thử View 1

### Tạo hiện thị thống kê số lượng bệnh án theo bác sĩ

Tạo View View\_ThongKeBenhAnTheoBacSi hiện thị các thông tin: mã bác sĩ, họ tên bác sĩ, số lượng bệnh án của các bác sĩ

A close up of text

Description automatically generated

Hình ‑: Câu lệnh tạo View 2

Truy vấn vào danh sách thống kê số lượng bệnh án theo bác sĩ

A black text with black letters

Description automatically generated

Hình ‑: Kiểm thử View 2

Kết quả

A screenshot of a computer

Description automatically generated

Hình ‑: Kết quả kiểm thử View 2

### Tạo hiện thị danh sách lịch khám đầy đủ thông tin

Tạo View View\_LichKhamChiTiet hiện thị các thông tin: mã lịch khám, họ tên bệnh nhân, họ tên bác sĩ, ngày khám, giờ khám.

A close-up of a computer code

Description automatically generated

Hình ‑: Câu lệnh tạo View 3

Truy vấn lịch khám chi tiết

A black text on a white background

Description automatically generated

Hình ‑: Kiểm thử View 3

Kết quả

A white background with black text

Description automatically generated

Hình ‑: Kết quả kiểm thử View 3

### Tạo hiện thị danh sách bác sĩ có chuyên môn nội khoa

Tạo View View\_ BacSiNoiKhoa hiện thị các thông tin: mã bác sĩ, họ tên bác sĩ, chuyên môn, số điện thoại.

A close up of a computer screen

Description automatically generated

Hình ‑: Câu lệnh tạo View 4

Truy vấn danh sách bác sĩ nội khoa

A black text on a white background

Description automatically generated

Hình ‑: Kiểm thử View 4

Kết quả

A screenshot of a phone

Description automatically generated

Hình ‑: Kết quả kiểm thử View 4

### Tạo hiện thị danh sách thống kê thuốc bệnh nhân

Tạo View View\_ ThongKeThuocBenhNhan hiện thị các thông tin: mã bệnh nhân, họ tên bệnh nhân, số lượng đơn thuốc, tổng số lượng thuốc.

A screenshot of a computer

Description automatically generated

Hình ‑: câu lệnh tạo View 5

Truy vấn danh sách thống kê thuốc của bệnh nhân

A black text on a white background

Description automatically generated

Hình ‑: Kiểm thử View 5

Kiểm thử

A screenshot of a table

Description automatically generated

Hình ‑: Kết quả kiểm thử View 5

## Function

### Mô tả ứng dụng của Function trong hệ thống

Hàm người dùng tự định nghĩa: là một đối tượng trong CSDL chứa các câu lệnh SQL, được biên dịch sẵn và lưu trữ trong CSDL, thực hiện một hành động như các tính toán phức tạp và trả về kết quả là một giá trị.

Giá trị trả về của hàm:

* Trả về giá trị đơn lẻ: chuỗi, giá trị logic, số...các kiểu dữ liệu của SQL.
* Trả về 1 bảng.

### Tạo hàm tính chi phí khám dựa trên chuyên môn của bác sĩ với tham số chuyền vào là mã bác sĩ

Tạo Function F\_TinhChiPhiKham biết chi phí khám nội khoa là 200000VND, ngoại khoa là 300000VND, nhi khoa là 250000VND và các chuyên môn còn lại là 150000VND

A screen shot of a computer

Description automatically generated

Hình ‑: Câu lệnh tạo Function 1

Kiểm thử với bác sĩ chuyên môn nội khoa



Hình ‑: Kiểm thử chi phí khám nội khoa

Kết quả với bác sĩ chuyên môn nội khoa

A close-up of a number

Description automatically generated

Hình ‑: Kết quả kiểm thử chi phí khám nội khoa

Kiểm thử với bác sĩ chuyên môn nhi khoa



Hình ‑: Kiểm thử chi phí khám nhi khoa

Kết quả với bác sĩ chuyên môn nhi khoa

A close up of a number

Description automatically generated

Hình ‑: Kết quả kiểm thử chi phí khám nhi khoa

Kiểm thử với bác sĩ chuyên môn răng hàm mặt



Hình ‑: Kiểm thử chi phí khám răng hàm mặt

Kết quả với bác sĩ chuyên môn răng hàm mặt

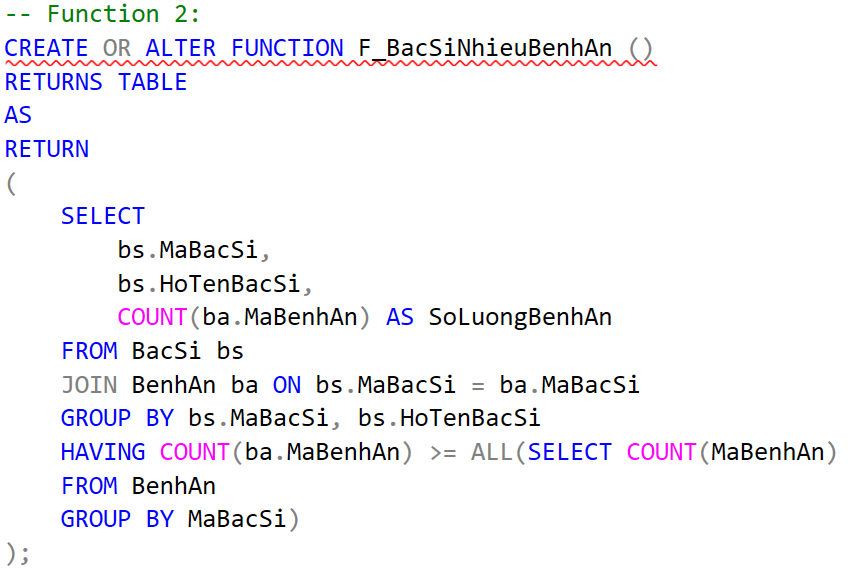
A close up of a number

Description automatically generated

Hình ‑: Kết quả kiểm thử chi phí khám răng hàm mặt

### Tạo hàm thống kê bác sĩ có nhiều bệnh án nhất và xuất ra bảng các thông tin bác sĩ

Tạo Function F\_BacSiNhieuBenhAn



Hình ‑: Câu lệnh tạo Function 2

Kiểm thử



Hình ‑: Kiểm thử Function 2

Kết quả



Hình ‑: Kết quả kiểm thử Function 2

## Store Procedure

### Mô tả ứng dụng của Store Procedure trong hệ thống

Thủ tục (Stored Procedure) là một tập hợp các câu lệnh T-SQL được biên dịch và lưu trữ trong cơ sở dữ liệu dưới một tên duy nhất. Một thủ tục có thể chứa một loạt các câu lệnh SQL, giúp thực hiện các tác vụ cụ thể và phức tạp mà người dùng hoặc hệ thống cần. Khi thủ tục được tạo và lưu trữ, nó sẽ có thể được gọi và thực thi nhiều lần mà không cần phải viết lại mã nguồn SQL mỗi khi cần thực hiện lại tác vụ đó.

Một trong những lợi ích lớn nhất của thủ tục là khả năng tái sử dụng code, điều này giúp giảm thiểu sự trùng lặp trong các câu lệnh SQL khi thực hiện những công việc tương tự ở nhiều nơi trong ứng dụng. Khi cần thực hiện một nhiệm vụ đã được định nghĩa trong một thủ tục, thay vì phải viết lại toàn bộ câu lệnh SQL, người dùng chỉ cần gọi thủ tục đó bằng tên đã định sẵn, giúp tiết kiệm thời gian và công sức.

Thủ tục giúp tăng hiệu suất khi thực hiện các tác vụ lặp đi lặp lại vì chúng được biên dịch và lưu trữ trực tiếp trên máy chủ cơ sở dữ liệu. Điều này có nghĩa là, khi một thủ tục được gọi, nó sẽ không phải trải qua bước biên dịch lại, giúp việc thực thi các câu lệnh SQL trở nên nhanh chóng và hiệu quả hơn. Nhờ đó, tốc độ xử lý các thủ tục thường rất nhanh so với việc thực thi các câu lệnh SQL riêng lẻ.

Thủ tục cũng mang lại một phương pháp thuận tiện cho việc quản lý và tổ chức các nhiệm vụ trong cơ sở dữ liệu. Việc gọi lại thủ tục giúp lập trình viên và quản trị viên cơ sở dữ liệu dễ dàng duy trì, kiểm tra và tối ưu các tác vụ mà không làm gián đoạn đến ứng dụng hoặc các phần khác của hệ thống. Điều này cực kỳ hữu ích trong các môi trường có yêu cầu hiệu suất và khả năng mở rộng cao.

Thủ tục có thể được chia thành thủ tục hệ thống và thủ tục do người dùng tạo. Thủ tục hệ thống là những thủ tục do hệ quản trị cơ sở dữ liệu cung cấp sẵn, phục vụ cho các chức năng quản lý hệ thống hoặc thao tác dữ liệu thông thường. Các thủ tục hệ thống trong SQL Server thường bắt đầu với tiền tố "sp\_". Chúng được đặt trong cơ sở dữ liệu master, nơi chứa các đối tượng quan trọng của hệ thống.

### Tính tổng chi phí khám của một bệnh nhân với mã bệnh nhân là tham số truyền vào

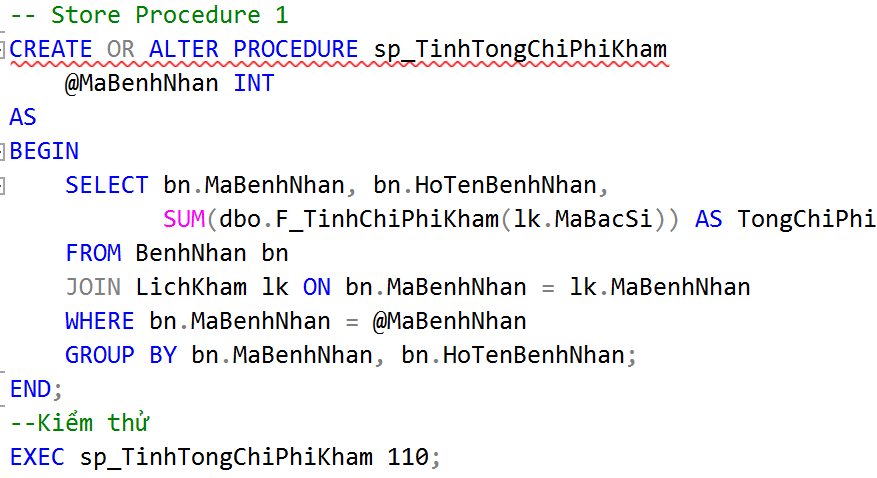
Tạo Store Procedure sp\_TinhTongChiPhiKham

A screenshot of a computer

Description automatically generated

Hình ‑: Câu lệnh tạo Store Procedure 1

Kiểm thử



Hình ‑: Kiểm thử Store Procedure 1

Kết quả



Hình ‑: Kết quả kiểm thử Store Procedure 1

### Hiện thị lịch khám của bệnh nhân với mã bệnh nhân là tham số truyền vào

Tạo Store Procedure sp\_ LayLichKhamBenhNhan

A screenshot of a computer code

Description automatically generated

Hình ‑: câu lệnh tạo Store Procedure 2

Kiểm thử



Hình ‑: Kiểm thử Store Procedure 2

Kết quả

A screenshot of a phone

Description automatically generated

Hình ‑: Kết quả kiểm thử Store Procedure 2

### Liệt kê các bác sĩ theo chuyên môn với chuyên môn là tham số được truyền vào

Tạo Store Procedure sp\_LietKeBacSiTheoChuyenMon

A screen shot of a computer code

Description automatically generated

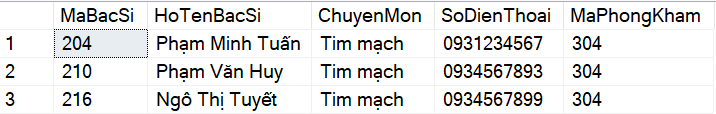
Hình ‑: Câu lệnh tạo Store Procedure 3

Kiểm thử



Hình ‑: Kiểm thử Store Procedure 3

Kết quả



Hình ‑: Kết quả kiểm thử Store Procedure 3

### Tính tổng số bệnh nhân đã khám trong một khoảng thời gian với tham số truyền vào ngày bắt đầu và ngày kết thúc

Tạo Store Procedure sp\_TinhTongBenhNhanTrongKhoangThoiGian

A screenshot of a computer

Description automatically generated

Hình ‑: Câu lệnh tạo Store Procedure 4

Kiểm thử



Hình ‑: Kiểm thử Store Procedure 4

Kết quả

A close-up of a computer screen

Description automatically generated

Hình ‑: Kết quả kiểm thử Store Procedure 4

### Thống kê chẩn đoán và đơn thuốc được dùng cho chẩn đoán đó

Tạo Store Procedure sp\_LietKeChanDoanVaDonThuoc

A screen shot of a computer code

Description automatically generated

Hình ‑: Câu lệnh tạo Store Procedure 5

Kiểm thử



Hình ‑: Kiểm thử Store Procedure 5

Kết quả

A table of prescriptions

Description automatically generated

Hình ‑: Kết quả kiểm thử Store Procedure 5

### Liệt kê các chẩn đoán theo chuyên môn với chuyên môn là tham số truyền vào

Tạo Store Procedure sp\_LietKeChanDoanTheoChuyenMon

A screenshot of a computer code

Description automatically generated

Hình ‑: Câu lệnh tạo Store Procedure 6

Kiểm thử



Hình ‑: Kiểm thử Store Procedure 6

Kết quả

A close-up of a white box

Description automatically generated

Hình ‑: Kết quả kiểm thử Store Procedure 6

## Trigger

### Mô tả ứng dụng của Trigger trong hệ thống

SQL Trigger là một hoặc một tập các câu lệnh SQL đưuọc lưu trữ trong CSDL và sẽ được kích hoạt bởi sự kiện mà nó theo dõi trên bảng dữ liệu. Sự kiện đó gắn liền với thao tác INSERT, UPDATE hay DELETE.

Cài đặt Trigger đối với ngôn ngữu DML có liên quan đến 2 bảng hệ thống là inserted và deleted.

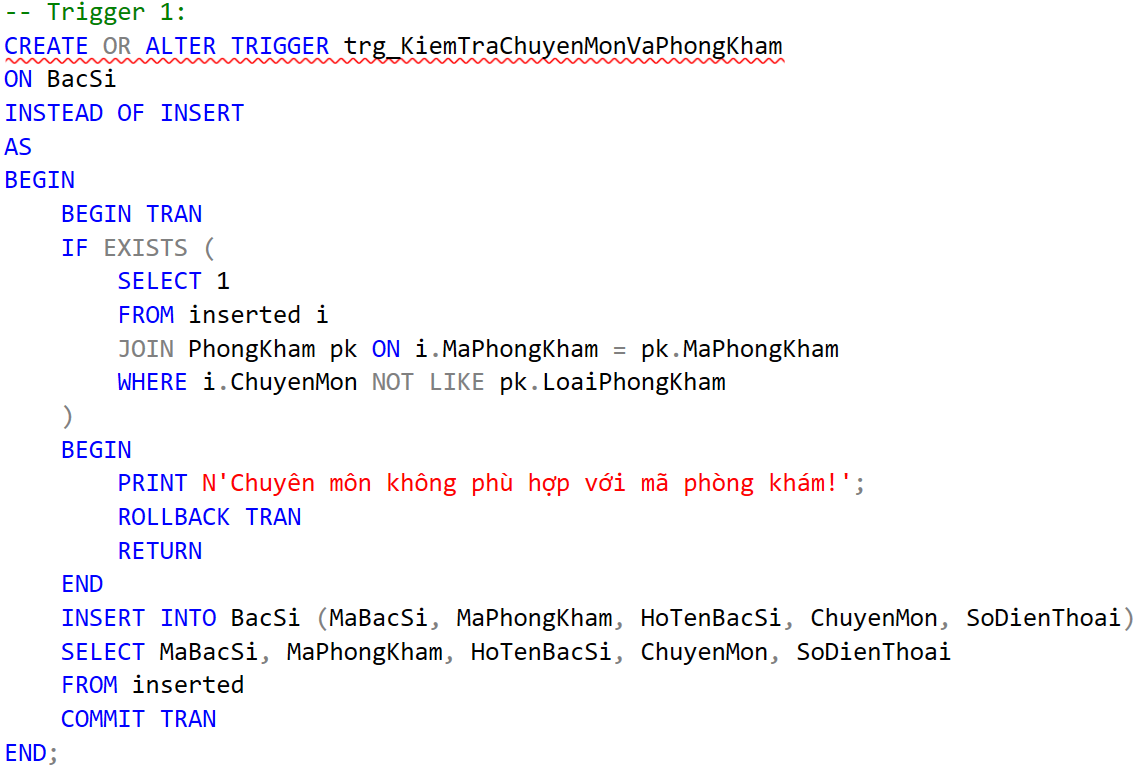
Khi thêm mới 1 mẫu tin vào 1 bảng, thao tác này sẽ kích hoạt trigger tương ứng. Trigger lưu trữ dữ liệu của mẫu tin vừa thêm mới vào bảng có tên là Inserted.

Khi xóa 1 mẫu tin trên 1 bảng, thao tác này sẽ kích hoạt 1 trigger tương ứng. Trigger lưu trữ dữ liệu của mẫu tin vừa xóa vào 1 bảng có tên là Deleted.

Cấu trúc của 2 bảng Inserted và Deleted hoàn toàn giống với cấu trúc bảng dữ liệu liên quan đến trigger khi tạo ra. 2 bảng này chỉ tồn tại trong bộ nhớ RAM, được xem như là 2 bảng luận lý có thể sử dụng trong các xử lý của trigger.

### Thêm vào bảng BacSi thì mã phòng khám phải phù hợp với chuyên môn bác sĩ

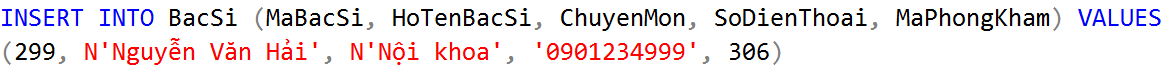
Tạo Trigger trg\_KiemTraChuyenMonVaPhongKham



Hình ‑: Câu lệnh tạo Trigger 1

Kiểm thử

* Trường hợp không thành công



Hình ‑: Kiểm thử trường hợp không thành công Trigger 1

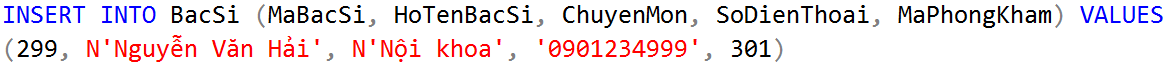
* Kết quả không thành công

A white background with red text

Description automatically generated

Hình ‑: Kết quả kiểm thử không thành công Trigger 1

* Trường hợp thành công



Hình ‑: Kiểm thử trường hợp thành công Trigger 1

* Kết quả thành công



Hình ‑: Kết quả kiểm thử thành công Trigger 1

### Khi thêm hoặc thay đổi lịch khám kiểm tra giờ của lịch khám phải nằm trong giờ làm việc từ 7:00 sáng đến 18:30 tối

Tạo Trigger trg\_KiemTraChuyenMonVaPhongKham

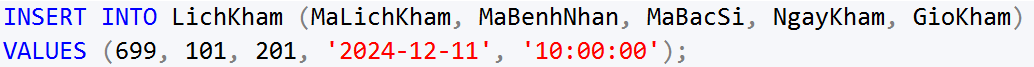
A screenshot of a computer

Description automatically generated

Hình ‑: Câu lệnh tạo Trigger 2

Kiểm thử:

* Kiểm thử thêm thành công



Hình ‑: Kiểm thử thêm thành công Trigger 2

* Kết quả thêm thành công



Hình ‑: Kết quả kiểm thử thêm thành công Trigger 2

* Kiểm thử sửa thành công

A black text with red and grey letters

Description automatically generated

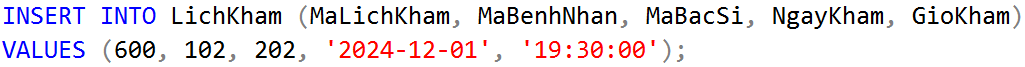
Hình ‑: Kiểm thử sửa thành công Trigger 2

* Kết quả sửa thành công



Hình ‑: Kết quả kiểm thử sửa thành công Trigger 2

* Kiểm thử thêm không thành công



Hình ‑: Kiểm thử thêm không thành công Trigger 2

* Kết quả thêm không thành công

A white background with red text

Description automatically generated

Hình ‑: Kết quả kiểm thử thêm không thành công Trigger 2

* Kiểm thử sửa không thành công

A black text on a white background

Description automatically generated

Hình ‑: Kiểm thử sửa không thành công Trigger 2

* Kết quả sửa không thành công

A white background with black text

Description automatically generated

Hình ‑: Kết quả kiểm thử sửa không thành công Trigger 2

### Thông báo xóa thành công lịch khám hoặc không tồn tại lịch khám

Tạo Trigger trg\_ThongBaoXoaLichKhamThanhCong

A screenshot of a computer program

Description automatically generated

Hình ‑: Câu lệnh tạo Trigger 3

Kiểm thử:

* Kiểm thử thành công



Hình ‑: Kiểm thử thành công Trigger 3

* Kết quả thành công

A white background with black text

Description automatically generated

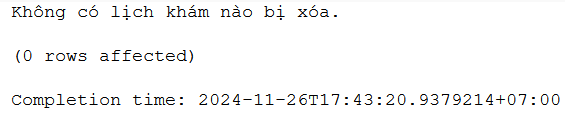
Hình ‑: Kết quả kiểm thử thành công Trigger 3

* Kiểm thử không thành công



Hình ‑: Kiểm thử không thành công Trigger 3

* Kết quả không thành công



Hình ‑: Kết quả kiểm thử không thành công Trigger 3

## User

### Ứng dụng của User trong hệ thống

SQL Server là một hệ quản trị cơ sở dữ liệu mạnh mẽ, cho phép người dùng tạo và quản lý tài khoản để truy cập và làm việc với hệ thống. Mỗi người dùng sẽ sử dụng một tài khoản riêng biệt để đăng nhập vào SQL Server, đảm bảo tính bảo mật và phân quyền trong môi trường quản lý dữ liệu.

Khi tạo một tài khoản người dùng mới, quản trị viên hệ thống sẽ xác định các quyền truy cập (Server Roles) mà tài khoản đó được phép thực hiện, như quyền đọc, ghi, sửa đổi hoặc quản trị cơ sở dữ liệu. Bên cạnh đó, quản trị viên cũng xác định phạm vi truy cập dữ liệu (Database Schemas), tức là những phần dữ liệu cụ thể mà tài khoản được phép thao tác. Điều này giúp kiểm soát quyền truy cập một cách chặt chẽ, đảm bảo rằng mỗi người dùng chỉ có thể truy cập và làm việc trong phạm vi được cho phép, giảm thiểu rủi ro bảo mật và hạn chế các lỗi thao tác không mong muốn trên cơ sở dữ liệu.

### Tạo User cho nhân viên y tá

Tạo nhóm người dùng cho nhân viên y tá với tên NhanVienYTa



Hình ‑: Câu lệnh tạo nhóm người dùng NhanVienYTa

Phân quyền cho nhân viên y tá có thể xem và sửa bảng BenhNhan và LichKham

A group of black and blue text

Description automatically generated

Hình ‑: Câu lệnh phân quyền cho nhóm người dùng NhanVienYTa

Tạo login cho nhân viên y tá

A close up of a sign

Description automatically generated

Hình ‑: Câu lệnh tạo login NV\_YTa

Tạo người dùng cho nhân viên y tá



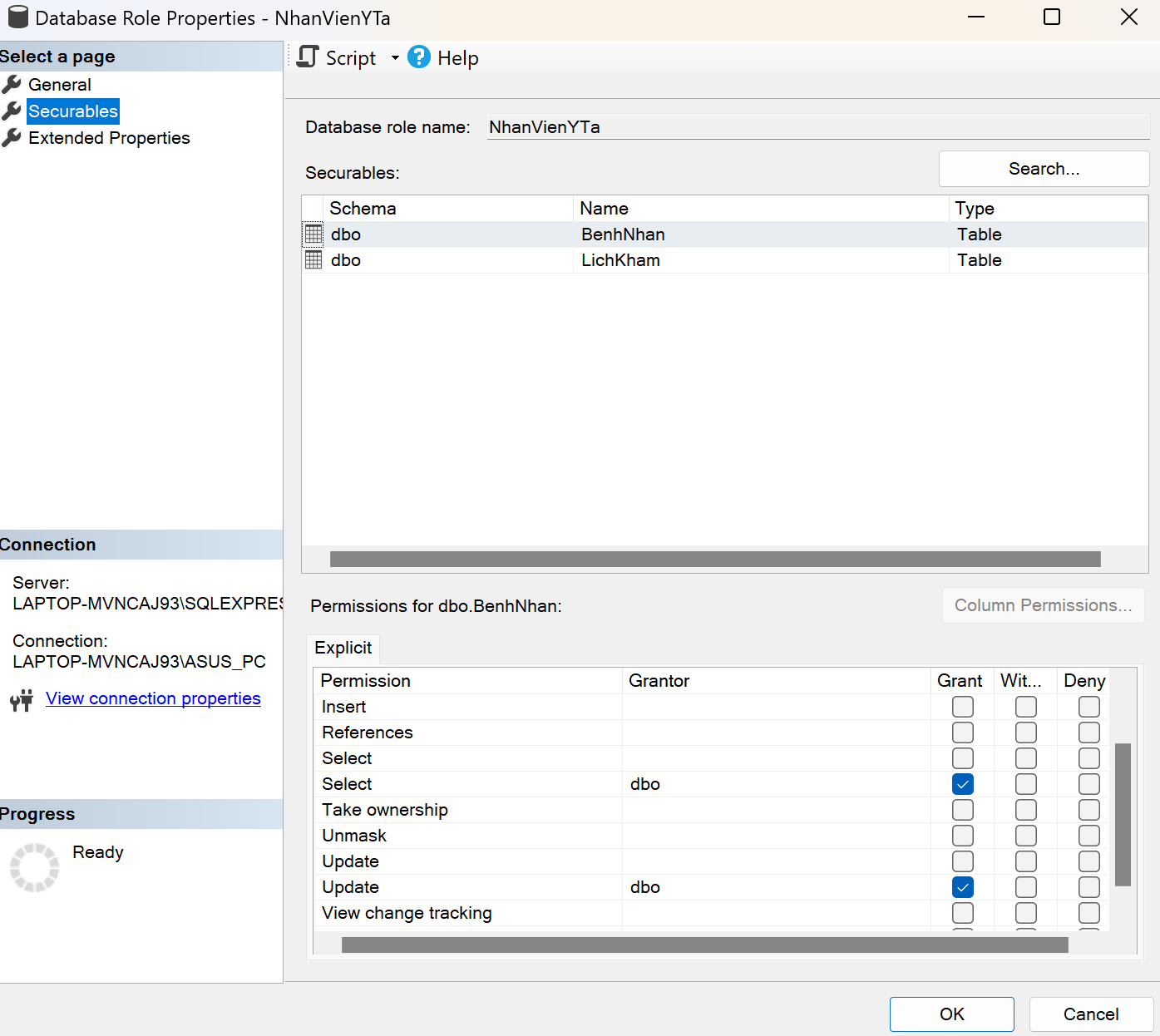
Hình ‑: Câu lệnh tạo người dùng NV\_YTa

Thêm người dùng vào nhóm người dùng NhanVienYTa



Hình ‑: Câu lệnh thêm người dùng NV\_YTa vào nhóm người dùng NhanVienYTa

Kiểm thử



Hình ‑: Kiểm thử phân quyền của nhóm người dùng NhanVienYTa

A screenshot of a computer

Description automatically generated

Hình ‑: Kiểm thử người dùng đã trong nhóm người dùng NhanVienYTa

### Tạo User cho bác sĩ quản lý

Tạo nhóm người dùng cho bác sĩ quản lý với tên BacSiQL



Hình ‑: Câu lệnh tạo nhóm người dùng BacSiQL

Phân quyền cho bác sĩ quản lý có thể xem, xóa và sửa bảng BenhNhan, BacSi, BenhAn, PhongKham, DonThuoc và LichKham

A screenshot of a computer screen

Description automatically generated

Hình ‑: Câu lệnh phân quyền cho nhóm người dùng BacSiQL

Tạo login cho bác sĩ quản lý

A close up of a text

Description automatically generated

Hình ‑: Câu lệnh tạo login cho BacSi

Tạo người dùng cho bác sĩ quản lý



Hình ‑: Câu lệnh tạo người dùng BacSi

Thêm người dùng vào nhóm người dùng BacSiQL



Hình ‑: Câu lệnh thêm người dùng BacSi vào nhóm người dùng BacSiQL

Kiểm thử

A screenshot of a computer

Description automatically generated

Hình ‑: Kiểm thử phân quyền của nhóm người dùng BacSiQL

A screenshot of a computer

Description automatically generated

Hình ‑: Kiểm thử người dùng đã trong nhóm người dùng BacSiQL

### Tạo User cho nhân viên bán thuốc

Tạo nhóm người dùng cho nhân viên bán thuốc với tên NhanVienBanThuoc



Hình ‑: Câu lệnh tạo nhóm người dùng NhanVienBanThuoc

Phân quyền cho bác sĩ quản lý có thể xem và sửa bảng DonThuoc



Hình ‑: Câu lệnh phân quyền cho nhóm người dùng NhanVienBanThuoc

Tạo login cho nhân viên bán thuốc

A close up of a sign

Description automatically generated

Hình ‑: Câu lệnh tạo login NV\_BanThuoc

Tạo người dùng cho nhân viên bán thuốc



Hình ‑: Câu lệnh tạo người dùng NV\_BanThuoc

Thêm người dùng vào nhóm người dùng NhanVienBanThuoc



Hình ‑: Câu lệnh thêm người dùng NV\_BanThuoc vào nhóm người dùng NhanVienBanThuoc

Kiểm thử

A screenshot of a computer

Description automatically generated

Hình ‑: Kiểm thử phân quyền của nhóm người dùng NhanVienBanThuoc

A screenshot of a computer

Description automatically generated

Hình ‑: Kiểm thử người dùng đã trong nhóm người dùng NhanVienBanThuoc

## Transaction

### Ứng dụng của Transaction trong hệ thống

Transaction trong SQL Server là một cách quản lý nhóm các thao tác trên cơ sở dữ liệu để đảm bảo tính toàn vẹn của dữ liệu. Dưới đây là các ứng dụng chính của transaction:

* Đảm bảo tất cả các thao tác trong giao dịch phải hoàn thành, nếu không thì không có thay đổi nào được lưu lại. Ví dụ: Chuyển tiền từ tài khoản này sang tài khoản khác phải hoàn tất cả hai thao tác, nếu không thì không thay đổi gì.
* Dữ liệu phải luôn ở trạng thái hợp lệ trước và sau giao dịch. Nếu có lỗi, hệ thống tự động khôi phục lại trạng thái ban đầu.
* Mỗi giao dịch phải độc lập với nhau, không bị ảnh hưởng bởi các giao dịch khác đang thực hiện đồng thời.
* Sau khi giao dịch được hoàn tất, mọi thay đổi sẽ được lưu lại vĩnh viễn, kể cả khi hệ thống gặp sự cố.

### Tạo thủ tục cập nhật liều dùng, số lượng đơn thuốc trong bảng DonThuoc với tham số truyền vào là mã đơn thuốc, mã bệnh án, liều dùng mới và số lượng mới xác định giao dịch hoàn thành hoặc quay lui khi giao dịch có lỗi

Tạo Store Procedure CapNhatDonThuoc

A screen shot of a computer code

Description automatically generated

Hình ‑: Câu lệnh tạo Store Procedure CapNhatDonThuoc

Kiểm thử không thành công



Hình ‑: Kiểm thử không thành công Store Procedure CapNhatDonThuoc

Kết quả kiểm thử không thành công

A screenshot of a computer

Description automatically generated

Hình ‑: Kết quả kiểm thử không thành công Store Procedure CapNhatDonThuoc

Kiểm thử thành công



Hình ‑: Kiểm thử thành công Store Procedure CapNhatDonThuoc

Kết quả kiểm thử thành công

A number and text on a white background

Description automatically generated

Hình ‑: Kết quả kiểm thử thành công Store Procedure CapNhatDonThuoc

### Tạo thủ tục xem phòng khám có những bác sĩ nào xác định giao dịch hoàn thành hoặc quay lui khi giao dịch có lỗi

Tạo Store Procedure XemPhongKham

A screenshot of a computer program

Description automatically generated

Hình ‑: Câu lệnh tạo Store Procedure XemPhongKham

Kiểm thử không thành công



Hình ‑: Kiểm thử không thành công Store Procedure XemPhongKham

Kết quả kiểm thử không thành công

A white background with black text

Description automatically generated

Hình ‑: Kết quả kiểm thử không thành công Store Procedure XemPhongKham

Kiểm thử thành công



Hình ‑: Kiểm thử thành công Store Procedure XemPhongKham

Kết quả kiểm thử thành công

A white background with black text

Description automatically generated

Hình ‑: Kết quả kiểm thử thành công Store Procedure XemPhongKham

# KẾT LUẬN

## Những điều đã thực hiện được trong đồ án

Đồ án đã thực hiện thành công nhiều công việc, đạt được những kết quả sau:

* Xây dựng mô hình cơ sở dữ liệu: Đồ án đã xây dựng một mô hình cơ sở dữ liệu hoàn chỉnh theo phương pháp mô hình thực thể - kết hợp (ERD). Sau đó, mô hình này được chuyển đổi sang mô hình dữ liệu quan hệ để áp dụng vào hệ thống quản lý bệnh viện. Mô hình dữ liệu này đã phản ánh đầy đủ các mối quan hệ giữa các thực thể trong môi trường bệnh viện, như bệnh nhân, bác sĩ, phòng khám, bệnh án, đơn thuốc, và lịch khám.
* Triển khai cơ sở dữ liệu: Các bảng dữ liệu đã được tạo ra một cách khoa học, bao gồm các bảng như bệnh nhân, bác sĩ, phòng khám, bệnh án, đơn thuốc, và lịch khám. Tất cả các bảng này đều được thiết lập với các ràng buộc toàn vẹn dữ liệu để đảm bảo tính chính xác và nhất quán khi xử lý thông tin.
* Sử dụng các kỹ thuật nâng cao: Để tối ưu hóa hiệu suất và bảo mật, đồ án đã áp dụng nhiều kỹ thuật nâng cao trong quản lý cơ sở dữ liệu như: synonym, index, view, function, stored procedure, và trigger. Những kỹ thuật này không chỉ giúp cải thiện hiệu suất xử lý dữ liệu mà còn bảo vệ dữ liệu khỏi các truy cập không hợp lệ.
* Phân quyền người dùng: Hệ thống đã thiết lập các user và nhóm user với các quyền hạn cụ thể, từ đó đảm bảo an toàn và bảo mật dữ liệu trong quá trình vận hành. Việc phân quyền rõ ràng giúp hệ thống dễ dàng quản lý quyền truy cập và theo dõi các hoạt động của người dùng.
* Kiểm thử và đánh giá: Các chức năng của hệ thống đã được kiểm thử đầy đủ và đã thành công trong việc xác nhận tính ổn định và khả năng đáp ứng các yêu cầu nghiệp vụ của bệnh viện. Việc kiểm thử giúp phát hiện và sửa chữa các lỗi còn tồn tại, đảm bảo hệ thống hoạt động trơn tru.

## Ưu điểm

Hệ thống cơ sở dữ liệu đã mang lại các ưu điểm sau:

* Nghiên cứu và phân tích nhu cầu của hệ thống quản lý bệnh viện, từ đó đề xuất các giải pháp tối ưu cho việc xử lý và giải quyết các vấn đề liên quan đến quản lý thông tin bệnh nhân, bác sĩ, và các dịch vụ y tế.
* Phân tích các nghiệp vụ của bệnh viện và đưa ra các tình huống cần giải quyết, giúp thiết kế một hệ thống cơ sở dữ liệu đáp ứng yêu cầu thực tế.
* Thiết kế và triển khai cơ sở dữ liệu quản lý các hoạt động của bệnh viện, bao gồm các bảng dữ liệu liên quan đến bệnh nhân, bác sĩ, phòng khám, bệnh án, đơn thuốc và lịch khám.
* Áp dụng kiến thức từ môn học Hệ quản trị cơ sở dữ liệu vào thực tiễn, tạo ra một hệ thống quản lý cơ sở dữ liệu chuyên nghiệp và hiệu quả.
* Thực hiện các lệnh truy vấn SQL để khai thác và hỗ trợ người dùng truy xuất dữ liệu từ hệ thống, đảm bảo tính chính xác và nhanh chóng.
* Thiết lập các hàm, thủ tục lưu trữ, và các ràng buộc dữ liệu, đảm bảo hệ thống hoạt động ổn định và đáp ứng các yêu cầu nghiệp vụ của bệnh viện.
* Phân quyền người dùng và cấp quyền cho từng nhóm người dùng, giúp quản lý thông tin bệnh viện một cách tối ưu, bảo mật và hiệu quả.

## Nhược điểm

Đồ án những nhược điểm mắc phải như sau:

* Qua 2 chương phân tích và tìm hiểu, xây dựng hệ thống quản lý bệnh viện, nhóm chúng em đã đưa ra những phân tích về hệ thống ở cả hai phương diện chức năng và dữ liệu. Tuy nhiên do còn hạn chế về nghiên cứu cũng như tiếp cận thực tế, chính vì vậy việc phân tích không tránh khỏi việc sai sót. Việc phân chia dòng dữ liệu còn bất cập. Chức năng còn trùng lặp và hạn chế. Thiết kế hệ thống còn chưa đúng với thực tế.
* Các sơ đồ thiết kế và công cụ phân tích chủ yếu mới chỉ dừng lại ở mức cơ bản, chưa khai thác hết các tính năng nâng cao của công cụ, ảnh hưởng đến hiệu quả thiết kế và quản lý cơ sở dữ liệu
* Thiết kế cơ sở dữ liệu chưa triệt để, các ràng buộc dữ liệu chưa tối ưu. Bảo mật hệ thống còn yếu, thiếu các biện pháp bảo vệ thông tin nhạy cảm như mã hóa hay kiểm soát truy cập chặt chẽ.
* Do giới hạn thời gian, nhóm chưa thể nghiên cứu và áp dụng các giải pháp nâng cao như tối ưu hiệu suất, bảo mật phức tạp hay các công nghệ mới. Việc chưa triển khai các giải pháp này có thể ảnh hưởng đến khả năng mở rộng và tính linh hoạt của hệ thống trong tương lai, đặc biệt khi hệ thống phải đối mặt với khối lượng dữ liệu lớn hơn hoặc yêu cầu bảo mật cao hơn trong môi trường thực tế.

TÀI LIỆU THAM KHẢO

[1]. ThS. Vũ Thị Thanh Hương (2024), Bài giảng Hệ quản trị cơ sở dữ liệu, Lưu hành nội bộ.

[2]. Trần Văn Hòa (2023), SQL Server - Giáo Trình Cơ Bản Và Nâng Cao, Nhà xuất bản Khoa học và Kỹ thuật.

[3]. Nguyễn Mạnh Tường (2022), Nhập Môn Cơ Sở Dữ Liệu và SQL, Trường Đại học Công nghệ.