**TRƯỜNG ĐẠI HỌC NGÂN HÀNG TP. HỒ CHÍ MINH**

**KHOA HỆ THỐNG THÔNG TIN QUẢN LÝ**



**ĐỒ ÁN CHUYÊN NGÀNH HỆ THỐNG THÔNG TIN QUẢN LÝ**

**TÊN ĐỀ TÀI:**

**XÂY DỰNG HỆ THỐNG QUẢN TRỊ CƠ SỞ DỮ LIỆU QUẢN LÝ VIỆC KHÁM CHỮA BỆNH TẠI BỆNH VIỆN   
ĐA KHOA QUỐC TẾ BECAMEX**

**Giảng viên hướng dẫn:** ThS. Trần Đức Tùng

**Sinh viên thực hiện:** Phạm Ngọc Minh Thư

**Mã số sinh viên:** 030236200168

**Lớp:** DH36CDS01

**Tháng 4 - 2023**

LỜI CẢM ƠN

Để hoàn thành bài báo cáo này, em đã may mắn nhận được sự giúp đỡ từ nhiều cá nhân và đơn vị. Trước hết, em xin gửi lời tri ân sâu sắc đến quý thầy, cô khoa Hệ Thống Thông Tin Quản Lý cũng như các quý thầy, cô của các khoa và bộ môn trường Đại học Ngân hàng Thành phố Hồ Chí Minh đã nhiệt tình truyền đạt kiến thức và các kinh nghiệm thực tế. Đặc biệt, em xin cảm ơn thầy Trần Đức Tùng – Giảng viên hướng dẫn đồ án khoa Hệ thống thông tin quản lý đã quan tâm, hỗ trợ em trong suốt quá trình thực hiện.

Đồng thời, em cũng xin gửi lời cảm ơn cô Vân đang công tác tại Bệnh viện Đa khoa Quốc tế Becamex đã nhiệt trình giúp đỡ, tạo điều kiện thuận lợi nhất cho em để em có thể hoàn thành đồ án này.

Vì thời gian có hạn, kiến thức bản thân còn chưa sâu sắc và kinh nghiệm thực tế chưa nhiều, trong quá trình nghiên cứu và làm báo cáo không tránh khỏi những sai sót. Em rất mong nhận được sự đóng góp ý kiến từ các thầy cô nhằm giúp nâng cao kiến thức và giúp cho bài báo cáo này hoàn thiện hơn.

Cuối cùng, em xin gửi lời chúc sức khỏe và thành công đến quý thầy cô trường Đại học Ngân hàng Thành phố Hồ Chí Minh, chúc thầy cô luôn có nhiều bước tiến mới trong tương lai.

*Thành phố Hồ Chí Minh, ngày 25 tháng 04 năm 2022*

Thư

**Phạm Ngọc Minh Thư**

LỜI CAM ĐOAN

Em xin cam đoan đồ án: “Đồ án xây dựng hệ thống quản trị cơ sở dữ liệu quản lý việc khám chữa bệnh tại Bệnh viện Đa khoa Quốc tế Becamex” là một công trình nghiên cứu dưới sự hướng dẫn của giảng viên hướng dẫn: ThS. Trần Đức Tùng. Đồ án, nội dung trong đồ án là sản phẩm mà em đã nỗ lực nghiên cứu trong quá trình học tập tại trường. Ngoài ra, trong bài báo cáo có sử dụng một số nguồn tài liệu tham khảo đã được trích dẫn nguồn và chú thích rõ ràng. Các số liệu, kết quả trình bày trong báo cáo là hoàn toàn trung thực, em xin chịu hoàn toàn trách nhiệm, kỷ luật của khoa và nhà trường đề ra nếu như có vấn đề xảy ra.

*Thành phố Hồ Chí Minh, ngày 25 tháng 04 năm 2022*

Thư

**Phạm Ngọc Minh Thư**

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

*Thành phố Hồ Chí Minh, ngày... tháng 12 năm 2023*

**Giảng viên hướng dẫn**

NHẬN XÉT CỦA GIẢNG VIÊN PHẢN BIỆN

……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

*Thành phố Hồ Chí Minh, ngày... tháng 12 năm 2023*

**Giảng viên phản biện**

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc133526833)

[LỜI CAM ĐOAN ii](#_Toc133526834)

[NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN iii](#_Toc133526835)

[NHẬN XÉT CỦA GIẢNG VIÊN PHẢN BIỆN iv](#_Toc133526836)

[MỤC LỤC v](#_Toc133526837)

[DANH MỤC HÌNH ẢNH viii](#_Toc133526838)

[DANH MỤC SƠ ĐỒ viii](#_Toc133526839)

[DANH MỤC BẢNG BIỂU viii](#_Toc133526840)

[LỜI MỞ ĐẦU 1](#_Toc133526841)

[CHƯƠNG 1: TỔNG QUAN 2](#_Toc133526842)

[1.1. Lý do chọn đề tài 2](#_Toc133526843)

[1.2. Ý nghĩa khoa học và tính thực tiễn của đề tài 2](#_Toc133526844)

[1.3. Mục tiêu nghiên cứu 2](#_Toc133526845)

[1.4. Đối tượng và phạm vi nghiên cứu 3](#_Toc133526846)

[1.5. Phương pháp nghiên cứu 3](#_Toc133526847)

[1.6. Cấu trúc của báo cáo 3](#_Toc133526848)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 5](#_Toc133526849)

[2.1. Hệ thống [1] 5](#_Toc133526850)

[2.2. Hệ thống thông tin [2] 5](#_Toc133526851)

[2.3. Các thành phần của hệ thống thông tin [2] 5](#_Toc133526852)

[2.3.1. Phần cứng 5](#_Toc133526853)

[2.3.2. Phần mềm 5](#_Toc133526854)

[2.3.3. Dữ liệu 6](#_Toc133526855)

[2.3.4. Quy trình 6](#_Toc133526856)

[2.3.5. Con người 6](#_Toc133526857)

[2.4. Mô hình hóa hệ thống 6](#_Toc133526858)

[2.4.1. Functional decomposition diagram (FDD) 6](#_Toc133526859)

[2.4.2. Data Flow Diagram (DFD) [3] 6](#_Toc133526860)

[2.4.3. Entity-Relationship Diagram (ERD) 7](#_Toc133526861)

[2.4.4. Mô hình dữ liệu quan hệ 7](#_Toc133526862)

[2.5. Hệ quản trị cơ sở dữ liệu 8](#_Toc133526863)

[2.6. Mô hình cơ sở dữ liệu quan hệ 8](#_Toc133526864)

[2.7. Hệ quản trị cơ sở dữ liệu SQL Server 8](#_Toc133526865)

[2.7.1. Giới thiệu chung về SQL Server 2014 8](#_Toc133526866)

[2.7.2. Đối tượng cơ sở dữ liệu hệ thống 8](#_Toc133526867)

[2.7.3. Đối tượng cơ sở dữ liệu người dùng 9](#_Toc133526868)

[2.7.4. Stored procedure: 9](#_Toc133526869)

[2.7.5. Giới thiệu câu lệnh SQL 11](#_Toc133526870)

[CHƯƠNG 3: MÔ TẢ BÀI TOÁN VÀ GIẢI PHÁP 12](#_Toc133526871)

[3.1. Giới thiệu tổng quan về Bệnh viện Becamex 12](#_Toc133526872)

[3.1.1. Thông tin chung 12](#_Toc133526873)

[3.1.2. Tổng quan về bệnh viện Đa Khoa Quốc tế Becamex [4] 12](#_Toc133526874)

[3.1.3. Sơ đồ cơ cấu tổ chức của bệnh viện [5] 13](#_Toc133526875)

[3.2. Hiện trạng của bệnh viện đa khoa quốc tế Becamex 13](#_Toc133526876)

[3.3. Những yêu cầu cần phải giải quyết 14](#_Toc133526877)

[3.3.1. Tìm hiểu yêu cầu 14](#_Toc133526878)

[3.3.2. Xác định yêu cầu 15](#_Toc133526879)

[3.4. Hướng phát triển hệ thống quản trị cơ sở dữ liệu quản lý khám chữa bệnh 16](#_Toc133526880)

[CHƯƠNG 4: PHÂN TÍCH HỆ THỐNG 17](#_Toc133526881)

[4.1. Mô tả quy trình khám chữa bệnh tại bệnh viện Becamex 17](#_Toc133526882)

[4.2. Sơ đồ phân rã chức năng (FDD) 17](#_Toc133526883)

[4.3. Sơ đồ luồng dữ liệu (DFD) 18](#_Toc133526884)

[4.3.1. Sơ đồ luồng dữ liệu mức ngữ cảnh 18](#_Toc133526885)

[4.3.2. Sơ đồ luồng dữ liệu mức 0 19](#_Toc133526886)

[CHƯƠNG 5: THIẾT KẾ CƠ SỞ DỮ LIỆU 20](#_Toc133526887)

[5.1. Mô hình thực thể liên kết E-R 20](#_Toc133526888)

[5.1.1. Xác định các thực thể và thuộc tính khóa của thực thể 20](#_Toc133526889)

[5.1.2. Xác định các mối quan hệ giữa các thực thể 21](#_Toc133526890)

[5.1.3. Xây dựng mô hình thực thể liên kết 26](#_Toc133526891)

[5.2. Mô hình dữ liệu quan hệ 27](#_Toc133526892)

[5.2.1. Chuyển đổi từ mô hình thực thể liên kết sang mô hình dữ liệu quan hệ 27](#_Toc133526893)

[5.2.2. Mối quan hệ giữa các bảng 39](#_Toc133526894)

[5.2.3. Mô hình mối quan hệ tổng thể 42](#_Toc133526895)

[5.3. Thiết kế câu lệnh truy vấn SQL 43](#_Toc133526896)

[5.3.1. Tạo cơ sở dữ liệu 43](#_Toc133526897)

[5.3.2. Tạo bảng 43](#_Toc133526898)

[5.3.3. Thêm ràng buộc 44](#_Toc133526899)

[5.3.4. Tạo function 44](#_Toc133526900)

[5.3.5. Tạo stored procedure 46](#_Toc133526901)

[5.3.6. Tạo view 47](#_Toc133526902)

[5.3.7. Tạo Trigger 48](#_Toc133526903)

[5.3.8. Tạo các bản sao lưu 48](#_Toc133526904)

[5.3.9. Một số câu lệnh truy vấn phục vụ cho việc quản lý 49](#_Toc133526905)

[CHƯƠNG 6: KẾT LUẬN 55](#_Toc133526906)

[TÀI LIỆU THAM KHẢO 56](#_Toc133526907)

DANH MỤC HÌNH ẢNH

[Hình 6‑1. Mối quan hệ của các bảng trong quản lý thông tin cá nhân 39](#_Toc133510279)

[Hình 6‑2. Mối quan hệ của các bảng trong quản lý bệnh án 40](#_Toc133510280)

[Hình 6‑3. Mối quan hệ của các bảng trong quản lý nhân sự 41](#_Toc133510281)

[Hình 6‑4. Mối quan hệ của các bảng trong quản lý thuốc 41](#_Toc133510282)

[Hình 6‑5. Mối quan hệ của các bảng trong quản lý thông tin tài chính. 42](#_Toc133510283)

[Hình 6‑6. Mối quan hệ tổng thể giữa toàn bộ các bảng trong cơ sở dữ liệu (1) 42](#_Toc133510284)

[Hình 6‑7. Mối quan hệ tổng thể giữa toàn bộ các bảng trong cơ sở dữ liệu (2) 43](#_Toc133510285)

[Hình 6‑8. Kết quả của câu lệnh function 46](#_Toc133510286)

[Hình 6‑9. Kết quả trả về từ câu lệnh gọi thủ tục 47](#_Toc133510287)

[Hình 6‑10. Kết quả từ câu lệnh gọi view 47](#_Toc133510288)

[Hình 6‑11. Phản hồi từ hệ thống khi Insert dữ liệu mới 48](#_Toc133510289)

[Hình 6‑12. Kết quả trả về cho thủ tục hiển thị bệnh án 53](#_Toc133510290)

DANH MỤC SƠ ĐỒ

[Sơ đồ 3‑1. Sơ đồ tổ chức của bệnh viện đa khoa quốc tế Becamex 13](#_Toc133510554)

[Sơ đồ 5‑1. Sơ đồ phân rã chức năng của Bệnh viện Becamex 17](#_Toc133510555)

[Sơ đồ 5‑2. Sơ đồ DFD mức ngữ cảnh 18](#_Toc133510556)

[Sơ đồ 5‑3. Sơ đồ DFD mức 0 18](#_Toc133510557)

[Sơ đồ 6‑1. Sơ đồ thực thể liên kết 26](#_Toc133510558)

DANH MỤC BẢNG BIỂU

[Bảng 4‑1. Bảng câu hỏi 15](#_Toc133510444)

[Bảng 6‑1. Bảng các thực thể và thuộc tính khóa của thực thể 20](#_Toc133510445)

LỜI MỞ ĐẦU

Cùng với sự phát triển vượt bậc của khoa học kỹ thuật như hiện nay, việc ứng dụng công nghệ thông tin trong hầu hết các lĩnh vực đời sống của con người là điều không còn xa lạ. Công nghệ thông tin đã mang lại rất nhiều giá trị thiết thực nhằm nâng cao hiệu suất cũng như chất lượng quản lý trong kinh doanh, giáo dục, quốc phòng an-ninh. Đặc biệt là trong lĩnh vực y tế. Một trong những thành công của việc ứng dụng công nghệ thông tin vào các lĩnh vực trên đó là ứng dụng cơ sở dữ liệu vào việc quản lý. Nhờ vào tin học hóa mà công tác quản lý các hoạt động trong tổ chức diễn ra hiệu quả hơn, thông tin được lưu trữ và kiểm soát tốt hơn so với quản lý bằng sổ sách như ngày xưa. Do đó, việc xây dựng một mô hình cơ sở dữ liệu là giải pháp hữu hiệu cho các doanh nghiệp có thể tồn tại và phát triển trong tương lai.

Bệnh viện đa khoa quốc tế Becamex được xem là một trong những bệnh viện quy mô lớn nhất tỉnh Bình Dương. Nhận thấy nhu cầu cần có một hệ quản trị cơ sở dữ liệu để quản lý dữ liệu y tế cho bệnh việc là cần thiết. Việc có một cơ sở dữ liệu sẽ giúp cho công tác quản lý bệnh nhân cũng như các hoạt động khám chữa bệnh trở nên thuận lợi hơn. Vì vậy, em đã chọn đề tài “Xây dựng hệ thống quản trị cơ sở dữ liệu quản lý việc khám chữa bệnh tại bệnh viện đa khoa quốc tế Becamex” cho đồ án chuyên ngành. Đề tài hướng đến việc nghiên cứu, tìm hiểu và đề xuất một mô hình quản trị cơ sở dữ liệu phù hợp với nhu cầu đặt ra của bệnh viện.

# TỔNG QUAN

## Lý do chọn đề tài

Ngày nay, nhu cầu khám chữa bệnh tại bệnh viện Becamex ngày càng tăng, đặc biệt là hai năm gần đây kể từ khi dịch Covid-19 xuất hiện. Mỗi ngày tạo ra một lượng dữ liệu khổng lồ liên quan đến bệnh án của bệnh nhân như các xét nghiệm, thuốc, quá trình điều trị,… Điều này đã gây khó khăn cho việc lưu trữ, tìm kiếm, quản lý các vấn đề về chia sẻ thông tin trong bệnh viện. Để giải quyết khó khăn trên, em thực hiện đề tài “Xây dựng hệ thống quản trị cơ sở dữ liệu quản lý việc khám chữa bệnh tại bệnh viện đa khoa quốc tế Becamex”. Ứng dụng hệ quản trị cơ sở dữ liệu giúp bệnh viện tối ưu hóa trong việc quản lý hoạt động khám chữa bệnh, quản lý thuốc, quản lý trang thiết bị, vật tư và quản lý đội ngũ y bác sĩ tại cơ sở bệnh viện. Qua đó, giúp tiết kiệm đáng kể thời gian và chi phí, tăng mức độ hài lòng của bệnh nhân, nâng cao lợi thế cạnh tranh cho bệnh viện.

## Ý nghĩa khoa học và tính thực tiễn của đề tài

* Ý nghĩa khoa học

Nghiên cứu thực trạng thu thập, lưu trữ và xử lý thông tin tại bệnh viện quốc tế Becamex, từ đó đề xuất mô hình cơ sở dữ liệu để quản lý thông tin bệnh nhân bằng việc ứng dụng công nghệ thông tin dựa trên cơ sở quản lý khoa học và tin học hóa.

* Ý nghĩa thực tiễn

Kết quả nghiên cứu giúp tạo ra một cấu trúc cơ sở dữ liệu bao trùm tất cả những thông tin cần thiết trong việc quản lý của bệnh viện như: quản lý thông tin bệnh nhân, dữ liệu bệnh án, kết quả xét nghiệm, kết quả chẩn đoán, dữ liệu về thuốc, thông tin tài chính, thông tin các trang thiết bị tại bệnh viện,…

Kết quả nghiên cứu có thể được ứng dụng để thiết kế cho các công trình như số hóa bệnh viện và làm cơ sở đề xuất cho thiết kế cơ sở dữ liệu y tế mức quốc gia.

## Mục tiêu nghiên cứu

Nghiên cứu về các loại thông tin cần thiết mà một cơ sở y tế cần để lưu trữ cho việc quản lý các hoạt động trong bệnh viện.

Nghiên cứu mối liên kết, sự chuyển hóa giữa các thông tin với nhau trong cơ sở y tế, từ đó làm nền tảng nghiên cứu cách thức tổ chức dữ liệu trong cơ sở dữ liệu.

Nghiên cứu luồng dữ liệu từ khi một bệnh nhân bắt đầu đến khám và điều trị tại bệnh viện đến khi xuất viện.

Nghiên cứu các loại mô hình cơ sở dữ liệu phục vụ cho việc thiết kế một cơ sở dữ liệu phù hợp với yêu cầu đặt ra.

## Đối tượng và phạm vi nghiên cứu

Nghiên cứ và giải quyết các vấn đề về quản lý dữ liệu tại bệnh viện quốc tế Becamex và đưa ra giải pháp xây dựng hệ quản trị cơ sở dữ liệu cho bệnh viện.

* Phạm vi không gian: Bệnh viện đa khoa quốc tế Becamex (Becamex International Hospital – BIH)
* Phạm vi thời gian: Thời gian nghiên cứu từ 20/02/2023 đến 28/04/2023

## Phương pháp nghiên cứu

* Các phương pháp được sử dụng để phân tích và thiết kế cơ sở dữ liệu bao gồm:
* *Phương pháp phân tích tài liệu:* Đọc và phân tích các tài liệu, công trình nghiên cứu khoa học trong và ngoài nước để làm cơ sở lý luận cho đề tài.
* *Phương pháp quan sát:* Phương pháp quan sát nhằm mục đích nghiên cứu, thu thập các thông tin liên quan tới phân tích thiết kế cơ sở dữ liệu cho hệ thống quản lý khám chữa bệnh ở bệnh viện.
* *Phương pháp bảng câu hỏi*: Kỹ thuật thu thập thông tin cho phép các nhà phân tích hệ thống nghiên cứu thái độ, niềm tin, hành vi và đặc điểm của một số người chủ chốt trong tổ chức, những người có thể bị ảnh hưởng bởi các hệ thống hiện tại hoặc đề xuất. Các loại câu hỏi cơ bản được sử dụng trong bảng câu hỏi là câu hỏi đóng và câu hỏi mở. [3]
* *Phương pháp phân rã chức năng:* Sử dụng Functional Decomposition Diagram để thực hiện phân tích chức năng và mô tả sự phân rã các chức năng thành các chức năng nhỏ hơn trong hệ thống*.*
* *Mô phỏng luồng dữ liệu của hệ thống:* Sử Data Flow Diagram để mô hình hóa các quy trình và luồng dữ liệu trong hệ thống.

## Cấu trúc của báo cáo

* Cấu trúc báo cáo gồm các phần như sau:

Chương 1: Giới thiệu lý do, mục tiêu, đối tượng và phạm vi nghiên cứu của đề tài. Giới thiệu tóm tắt nội dung sẽ được trình bày trong các chương tiếp theo.

Chương 2: Cơ sở lý luận của đề tài

Chương 3: Mô tả bài toán và giải pháp

* Giới thiệu tổng quan về bệnh viện đa khoa quốc tế Becamex
* Hiện trạng của bệnh viện đa khoa quốc tế Becamex
* Những yêu cầu cần phải giải quyết
* Hướng phát triển hệ thống quản trị cơ sở dữ liệu quản lý khám chữa bệnh

Chương 4: Phân tích hệ thống

* Mô tả quy trình nghiệp vụ
* Phân tích các chức năng, tác nhân, hồ sơ
* Thiết kế các sơ đồ FDD, DFD

Chương 5: Thiết kế cơ sở dữ liệu

* Thiết kế sơ đồ ERD
* Thiết kế cơ sở dữ liệu
* Thiết kế câu lệnh truy vấn SQL

Chương 6: Kết luận chung cho các chương trong đồ án. Trình bày những vấn đề đã giải quyết, những hạn chế và hướng phát triển trong tương lai.

# CƠ SỞ LÝ THUYẾT

## Hệ thống [1]

Một hệ thống là một tập hợp các thủ tục kinh doanh (hoặc các thành phần) có liên quan với nhau được sử dụng trong một đơn vị kinh doanh, hoạt động cùng nhau vì một mục đích nào đó.

Một hệ thống có chín đặc điểm bao gồm: Thành phần (Component), Liên hệ giữa các thành phần (Interrelated components), Ranh giới (Boundary), Mục đích (Purpose), Môi trường (Environment), Giao diện (Interface), Ràng buộc (Constraints), Đầu vào (Input), Đầu ra (Output).

## Hệ thống thông tin [2]

Hệ thống thông tin là sự kết hợp giữa công nghệ, con người và dữ liệu để hỗ trợ cho các chức năng kinh doanh. Một số hệ thống thông tin xử lý các công việc hàng ngày thông thường, trong khi những hệ thống khác có thể giúp người quản lý đưa ra quyết định tốt hơn, phát hiện xu hướng thị trường và tiết lộ các mẫu có thể ẩn trong dữ liệu được lưu trữ.

## Các thành phần của hệ thống thông tin [2]

Một hệ thống thông tin gồm năm thành phần cơ bản: phần cứng, phần mềm, dữ liệu, quy trình và con người.

### Phần cứng

Phần cứng bao gồm mọi thứ trong lớp vật lý của hệ thống thông tin. Ví dụ: phần cứng có thể bao gồm máy chủ, máy trạm, mạng, thiết bị viễn thông, cáp quang, thiết bị di động, máy quét, thiết bị chụp ảnh kỹ thuật số và cơ sở hạ tầng dựa trên công nghệ khác.

### Phần mềm

Phần mềm đề cập đến các chương trình điều khiển phần cứng và tạo ra thông tin hoặc kết quả mong muốn. Phần mềm bao gồm phần mềm hệ thống và phần mềm ứng dụng.

#### Phần mềm hệ thống

Phần mềm hệ thống quản lý các thành phần của phần cứng, có thể bao gồm một máy tính đơn lẻ hoặc một mạng toàn cầu với hàng nghìn máy khách. Phần mềm hệ thống cũng kiểm soát luồng dữ liệu, cung cấp bảo mật dữ liệu và quản lý các hoạt động của mạng.

#### Phần mềm ứng dụng

Phần mềm ứng dụng bao gồm các chương trình hỗ trợ các chức năng kinh doanh hàng ngày và cung cấp cho người dùng thông tin họ cần. Phần mềm ứng dụng bao gồm hệ thống ngang và dọc. Hệ thống theo chiều ngang là một hệ thống, chẳng hạn như ứng dụng kiểm kê hoặc bảng lương, có thể được điều chỉnh để sử dụng trong nhiều loại hình công ty khác nhau. Hệ thống dọc được thiết kế để đáp ứng các yêu cầu riêng của một doanh nghiệp hoặc ngành cụ thể, chẳng hạn như nhà bán lẻ trực tuyến, cơ sở y tế hoặc đại lý ô tô.

### Dữ liệu

Dữ liệu là nguyên liệu thô mà một hệ thống thông tin chuyển đổi thành thông tin hữu ích.

### Quy trình

Các quy trình mô tả các nhiệm vụ và chức năng kinh doanh mà người dùng, người quản lý và nhân viên công nghệ thông tin thực hiện để đạt được kết quả cụ thể. Các quy trình là các khối xây dựng của một hệ thống thông tin vì chúng đại diện cho các hoạt động kinh doanh thực tế hàng ngày.

### Con người

Con người là những người quan tâm đến một hệ thống thông tin hay còn gọi là các bên liên quan. Các bên liên quan bao gồm nhóm quản lý chịu trách nhiệm về hệ thống, người dùng (đôi khi được gọi là người dùng cuối) trong và ngoài công ty, những người sẽ tương tác với hệ thống và nhân viên công nghệ thông tin, chẳng hạn như nhà phân tích hệ thống, lập trình viên và quản trị viên mạng, người phát triển và hỗ trợ hệ thống.

## Mô hình hóa hệ thống

### Functional decomposition diagram (FDD)

Sơ đồ phân rã chức năng (FDD) là sự biểu diễn từ trên xuống của một chức năng hoặc quy trình. Sử dụng FDD, nhà phân tích có thể hiển thị các chức năng kinh doanh và chia nhỏ chúng thành các chức năng và quy trình cấp thấp hơn. [2]

### Data Flow Diagram (DFD) [3]

Một mô tả đồ họa về các quy trình dữ liệu, luồng dữ liệu và kho dữ liệu trong một hệ thống kinh doanh.

Bốn biểu tượng cơ bản được sử dụng để lập biểu đồ chuyển động dữ liệu trên DFD: hình vuông kép, mũi tên, hình chữ nhật có các góc tròn và hình chữ nhật có đầu mở (đóng ở bên trái và mở ở bên phải)

* Hình vuông kép được sử dụng để mô tả một thực thể bên ngoài (bộ phận khác, doanh nghiệp, người hoặc máy) có thể gửi dữ liệu đến hoặc nhận dữ liệu từ hệ thống.
* Mũi tên hiển thị chuyển động của dữ liệu từ điểm này sang điểm khác, với đầu mũi tên chỉ về đích của dữ liệu.
* Một hình chữ nhật bo góc được sử dụng để hiển thị sự xuất hiện của một quy trình biến đổi. Một quy trình phải được cung đánh một số nhận dạng duy nhất cho biết cấp độ của nó trong sơ đồ.
* Hình chữ nhật có một đầu mở, một đầu đóng đại diện cho một kho lưu trữ dữ liệu.

### Entity-Relationship Diagram (ERD)

Sơ đồ mối quan hệ thực thể (ERD) là một mô hình hiển thị các mối quan hệ logic và tương tác giữa các thực thể hệ thống. ERD cung cấp cái nhìn tổng thể về hệ thống và kế hoạch chi tiết để tạo cấu trúc dữ liệu vật lý. [2]

Các thành phần cơ bản:

* Thực thể (Enity): Là một đối tượng, sự vật, sự việc trong thế giới thực. Thực thể được phân ra là thực thể mạnh (Strong entity) và thực thể yếu (Weak entity). Trong sơ đồ, thực thể được kí hiệu bằng hình chữ nhật.
* Thuộc tính (Attributes): Là các tính chất, đặc trưng của đối tượng. Thuộc tính bao gồm thuộc tính đơn trị, đa trị và thuộc tính phức hợp. Ký hiệu hình elip.
* Mối quan hệ (Relationship): Là sự liên kết giữa hai hoặc nhiều thực thể được biểu diễn bằng hình thoi. Trong mô hình thực thể liên kết có các loại liên kết sau:
* Quan hệ 1-1: Một thực thể từ tập thực thể X có thể được liên kết với nhiều nhất một thực thể của tập thực thể Y và ngược lại.
* Quan hệ 1-n: Nhiều thực thể từ tập thực thể X có thể được liên kết với nhiều nhất một thực thể của tập thực thể Y. Tuy nhiên, một thực thể từ tập thực thể Y có thể được liên kết hoặc không với nhiều thực thể từ tập thực thể X.
* Quan hệ n-n: Một thực thể từ X có thể được liên kết với nhiều thực thể từ Y và ngược lại.

### Mô hình dữ liệu quan hệ

#### Khái niệm

Mô hình Dữ liệu Quan hệ (Relational Data Model – RDM) biểu diễn dữ liệu dưới dạng một tập hợp các bảng hoặc quan hệ có liên quan đến nhau. [1]

Ba thành phần của mô hình dữ liệu quan hệ: Ràng buộc trên dữ liệu, tổ chức dữ liệu và thao tác trên dữ liệu.

#### Các dạng chuẩn của mô hình dữ liệu quan hệ

Một quan hệ R được gọi là ở **dạng chuẩn thứ nhất (1NF)** khi và chỉ khi các thuộc tính chỉ chứa các giá trị nguyên tố.

Một quan hệ R được gọi là ở **dạng chuẩn thứ hai (2NF)** khi và chỉ khi nó ở dạng chuẩn thứ nhất và mọi thuộc tính không khóa của R phụ thuộc hàm đầy đủ vào khóa chính của nó.

Quan hệ R được gọi là ở **dạng chuẩn thứ ba (3NF)** khi và chỉ khi nó ở dạng chuẩn thứ hai và mọi thuộc tính không khóa của nó đều không phụ thuộc bắc cầu vào khóa chính.

Quan hệ R được gọi là ở **dạng chuẩn Boyce – Codd (BCNF)** khi và chỉ khi với mọi phụ thuộc hàm không hiển nhiên X → Y trên R, X là siêu khóa của R.

## Hệ quản trị cơ sở dữ liệu

Hệ quản trị cơ sở dữ liệu (Database Management System - DBMS) là tập hợp các công cụ, tính năng và giao diện cho phép người dùng thêm, cập nhật, quản lý, truy cập và phân tích dữ liệu.

## Mô hình cơ sở dữ liệu quan hệ

Năm 1970, tiến sỹ E.F.Codd đã phát triển mô hình cơ sở dữ liệu mới với tên gọi cơ sở dữ liệu quan hệ (Relational Database). Loại cơ sở dữ liệu này loại bỏ được những vấn đề liên quan tới chuẩn file và các thiết kế cơ sở dữ liệu khác. Mô hình cho cơ sở dữ liệu quan hệ chỉ ra dữ liệu được lưu trữ trên một hoặc nhiều bảng (table). Và mô hình này cũng cho thấy, mỗi bảng có thể có được hình dung như một ma trận hai chiều gồm các hàng (row) và cột (column). Trong thực tế, các hàng và cột của bảng cơ sở dữ liệu quan hệ thường được gọi tên theo thuật ngữ là các bản ghi (record) và trường (field).

## Hệ quản trị cơ sở dữ liệu SQL Server

Microsoft SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relation Database Management System - RDBMS) được phát triển bởi Microsoft.

### Giới thiệu chung về SQL Server 2014

SQL Server 2014 hiện hỗ trợ OLTP trong bộ nhớ, cho phép người dùng chạy các ứng dụng xử lý giao dịch trực tuyến. Thay vì sử dụng các tệp dựa trên bộ đĩa tiêu chuẩn, việc thực thi sẽ dựa trên dữ liệu được lưu trữ trong các bảng được tối ưu hóa bộ nhớ.

SQL Server sử dụng T-SQL (Transact-SQL) làm ngôn ngữ truy vấn dữ liệu giữa máy Client và máy cài SQL Server. T-SQL cung cấp các tính năng như lập trình giúp thao tác với cơ sở dữ liệu như là một ngôn ngữ lập trình thực thụ.

Những thành phần chính trong SQL Server gồm: Database Engine, Intergration Services, Reporting Services, Analysis Services, Service Broker, Replication, Full-Text Search, Notification Services.

### Đối tượng cơ sở dữ liệu hệ thống

* MASTER: Điều khiển các thao tác, thông tin mức hệ thống của SQL Server. Lưu trữ toàn bộ các thông tin tài khoản người dùng, biến môi trường, thông điệp lỗi hệ thống và các thủ tục hệ thống. Ghi lại sự tồn tại của tất cả cơ sở dữ liệu, vị trí của các tập tin dữ liệu.
* MODEL: Cơ sở dữ liệu mẫu cho tất cả các cơ sở dữ liệu mới được tạo ra trong hệ thống, có thể được tùy biến cho phù hợp.
* MSDB: Được sử dụng bởi SQL Server Agent trong việc lập biểu, cảnh báo và ghi nhận các thao tác. Các bảng hệ thống được lưu trữ trong MSDB.
* TEMPDB: Được dùng như là vùng chứa tạm thời đối với các bảng và các thủ tục tạm. Làm việc trong Tempdb rất nhanh vì các hoạt động không ghi nhận lại. Khi client kết thúc kết nối với Server thì toàn bộ các bảng và thủ tục trong Tempdb sẽ bị xóa.

### Đối tượng cơ sở dữ liệu người dùng

#### Table

Bảng là đối tượng lưu trữ dữ liệu của cơ sở dữ liệu. Mỗi bảng có thể chứa từ 1 đến 1.024 cột. Mỗi cột phải có tên duy nhất và phải được gán kiểu dữ liệu. Có thể gán một hoặc nhiều thuộc tính cho cột.

Một số thuộc tính cột phổ biến:

* NULL | NOT NULL: Chỉ ra cột có chấp nhận giá trị NULL hay không. Nếu không sử dụng, Null là thuộc tính mặc định, trừ khi thuộc tính PRIMARY KEY được chỉ định.
* PRIMARY KEY | UNIQUE: Xác định khóa chính hoặc khóa đơn nhất cho bảng. Nếu thuộc tính PRIMARY được chỉ định, thuộc tính NULL không được chấp nhận.
* IDENTITY: Xác định cột identity column. Chỉ một identity column được tạo cho mỗi bảng.
* DEFAULT: Chỉ định giá trị mặc định cho cột.

#### Data type

SQL Server hỗ trợ hầu hết, không phải tất cả, các kiểu dữ liệu chuẩn ANSI. SQL Server định nghĩa 26 kiểu dữ liệu, được chia làm bốn nhóm như sau:

* Chuỗi: Các chuỗi dữ liệu ký tự.
* Số: Các số nguyên, số có dấu phẩy động, tiền tệ và các kiểu dữ liệu số khác.
* Thời gian (ngày/giờ): Ngày, giờ hoặc cả ngày và giờ.
* Kiểu khác: Các giá trị chuỗi và nhị phân lớn, XML, dữ liệu hình học, dữ liệu địa lý, dữ liệu phân cấp.

#### View

View là đối tượng cơ sở dữ liệu chứa các câu lệnh SELECT. Bảng được viết trong câu lệnh SELECT tạo view gọi là bảng cơ sở của view. View còn gọi là khung nhìn, vì nó cung cấp một khung nhìn cho các bảng cơ sở bên dưới.

Dữ liệu trong khung nhìn không được lưu trữ vật lý do nó tham chiếu trở lại các bảng cơ sở của nó và trả về dữ liệu hiện thời. Có thể truy vấn, thậm chí là thay đổi.

### Stored procedure:

Stored procedure là đối tượng cơ sở dữ liệu có thể thực thi, chứa các tập lệnh T-SQL. Stored procedure còn được gọi là thủ tục.

Thủ tục được biên dịch sẵn. Có nghĩa là, kế hoạch thực thi cho mã lệnh SQL được biên dịch trong lần đầu thủ tục được thực thi, sau đó lưu lại dưới dạng được biên dịch.

Store procedure cung cấp một phương pháp hữu ích cho việc thực thi lặp lại cùng một nhiệm vụ:

* Giúp tái sử dụng code
* Sử dụng lời gọi thủ tục mỗi khi thực thi

#### Index:

Chỉ mục được tạo trên một thuộc tính của quan hệ nhằm tăng tốc độ truy xuất dữ liệu liên quan đến thuộc tính đó.

Mỗi khi giá trị tại thuộc tính đó bị thay đổi (Thêm mới, sửa, xóa), toàn bộ chỉ mục sẽ được xác định lại.

#### Constraint:

Ràng buộc dữ liệu, được thiết lập trên một cột hoặc nhiều cột dữ liệu để thiết lập toàn vẹn dữ liệu.

|  |  |  |
| --- | --- | --- |
| **Ràng buộc** | **Sử dụng ràng buộc mức cột** | **Sử dụng ràng buộc mức bảng** |
| NOT NULL | Ngăn giá trị null được đưa vào cột. | Không có |
| PIMARY KEY | Bắt buộc mỗi hàng trong bảng có giá trị duy nhất trên cột này. Giá trị null không được chấp nhận. | Bắt buộc tập giá trị của một hay nhiều cột trên mỗi hàng của bảng phải là duy nhất. Giá trị null không được chấp nhận. |
| UNIQUE | Bắt buộc mỗi hàng trong bảng có giá trị duy nhất trong cột này. | Bắt buộc tập giá trị của một hay nhiều cột trên mỗi hàng của bảng là duy nhất. |
| CHECK | Giới hạn phạm vi giá trị cho một cột. | Giới hạn phạm vi giá trị cho một hoặc nhiều cột. |
| [FOREIGN KEY]  REFERENCES | Đảm bảo toàn vẹn tham chiếu giữa cột trong bảng mới và cột trong bảng liên hệ. | Đảm bảo toàn vẹn tham chiếu giữa một hoặc nhiều cột trong bảng mới và một hoặc nhiều cột trong bảng quan hệ. |

#### Trigger

Bẫy lỗi là một loại Stored procedure đặc biệt ở trạng thái sẵn sàng, được kích hoạt bởi câu lệnh thay đổi dữ liệu trên bảng hoặc khung nhìn.

### Giới thiệu câu lệnh SQL

SQL được chia làm hai loại:

* *Ngôn ngữ thao tác dữ liệu (Data Manipulation Language – DML)* giúp làm việc với dữ liệu trong cơ sở dữ liệu, gồm bốn câu lệnh được sử dụng nhiều nhất sau:
* SELECT: Truy xuất dữ liệu từ một hoặc nhiều bảng.
* INSERT: Thêm một hoặc nhiều hàng mới vào bảng.
* UPDATE: Thay đổi một hoặc nhiều hàng hiện có trong bảng.
* DELETE: Xóa một hoặc nhiều hàng hiện có trong bảng.
* *Ngôn ngữ định nghĩa dữ liệu (Data Definition Language – DDL)* được sử dụng để làm việc với đối tượng cơ sở dữ liệu. Trên những hệ thống lớn, câu lệnh này thường được các nhà quản trị cơ sở dữ liệu sử dụng nhằm bảo trì cơ sở dữ liệu hiện có, nâng cao hiệu suất và tạo các cơ sở dữ liệu mới.

# MÔ TẢ BÀI TOÁN VÀ GIẢI PHÁP

## Giới thiệu tổng quan về Bệnh viện Becamex

### Thông tin chung

* Tên: Bệnh viện Đa Khoa Quốc tế Becamex (Becamex International Hospital – BIH)
* Địa chỉ: Đại lộ Bình Dương, khu Gò Cát, phường Lái Thiêu, thành phố Thuận An, Bình Dương (góc ngã tư cầu Ông Bố).
* Số điện thoại: (0274) 3 681 681
* Website: [www.bih.vn](http://www.bih.vn)
* Logo:



### Tổng quan về bệnh viện Đa Khoa Quốc tế Becamex [4]

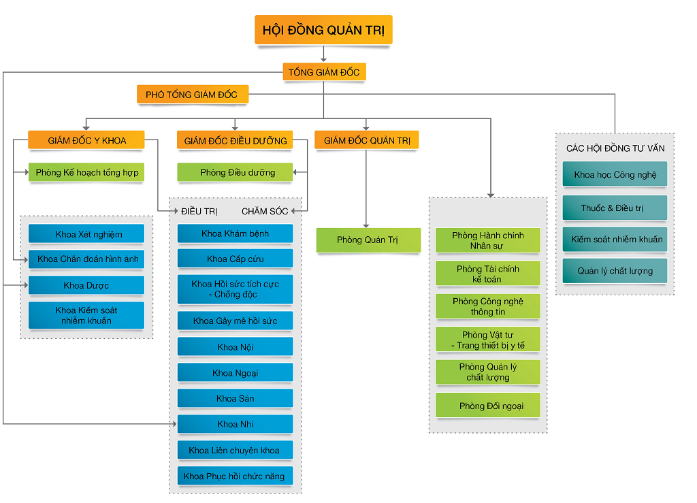
Bệnh viện Đa Khoa Quốc tế Becamex chính thức đưa vào hoạt động từ ngày 30/12/2016, BIH với qui mô 1.200 giường cung cấp dịch vụ khám bệnh, chữa bệnh đa khoa với tôn chỉ phục vụ cho cộng đồng nhân dân tỉnh Bình Dương và các địa bàn lân cận.

BIH được thiết kế, xây dựng và hoạt động theo nguyên tắc lấy bệnh nhân làm trung tâm. Bệnh nhân được chăm sóc toàn diện với dịch vụ y tế chất lượng cao, trang thiết bị hiện đại; được tôn trọng trong môi trường thân thiện; được chọn lựa chế độ thăm khám, chế độ ăn; được nghỉ ngơi trong không gian xanh, gần gũi thiên nhiên.

BIH còn là cơ sở thực hành cho sinh viên y khoa, điều dưỡng của các trường đại học thuộc tỉnh Bình Dương và các địa bàn lân cận. Bệnh viện cũng chú trọng đầu tư nghiên cứu khoa học, ứng dụng các tiến bộ khoa học kỹ thuật vào việc khám bệnh, chữa bệnh và quản lý bệnh viện.

BIH tập trung vào hệ thống quản lý chất lượng và hướng tới được công nhận chất lượng theo tiêu chuẩn quốc tế JCI (Joint Commission International).

### Sơ đồ cơ cấu tổ chức của bệnh viện [5]



Sơ đồ 3‑1. Sơ đồ tổ chức của bệnh viện đa khoa quốc tế Becamex

## Hiện trạng của bệnh viện đa khoa quốc tế Becamex

Bệnh viện đa khoa quốc tế Becamex đã đi vào hoạt động được 7 năm kể từ năm 2016 đến nay. Với quy mô và diện tích rộng, trang thiết bị hiện đại cùng với đội ngũ y bác sĩ có chuyên môn cao, BIH luôn không ngừng cải tiến chất lượng phục vụ để mang lại những trải nghiệm khám chữa bệnh tốt nhất cho bệnh nhân. Bệnh viện cung cấp đầy đủ các chuyên khoa, các khoa Cấp cứu, khoa Hồi sức tích cực, khoa Phẫu thuật, khoa Xét nghiệm, khoa Chẩn đoán hình ảnh được xây dựng và trang bị đầy đủ theo tiêu chuẩn quốc tế.

Theo thống kê báo cáo năm 2019, số lượng khám bệnh ngoại trú ngày càng tăng, đạt 1.000 lượt khám mỗi ngày, công suất giường tăng đặc biệt vào quí II, III tại một số khoa như Nội, Sản, Nhi còn xảy ra tình trạng quá tải, không đủ giường. Cũng trong năm 2019, có gần 5000 ca phẫu thuật ngoại khoa được diễn ra tại bệnh viện.

BIH cung cấp dịch vụ khám sức khỏe định kỳ cho 145 doanh nghiệp, hoàn thành mục tiêu của Tổng Công ty Becamex: gắn bó với cộng đồng doanh nghiệp, là địa chỉ chăm sóc sức khỏe tin cậy để doanh nghiệp trong và ngoài nước yên tâm đầu tư tại Bình Dương.

Có thể thấy, việc lưu trữ thông tin đóng vai trò rất quan trọng trong giải pháp quản lý bệnh viện. Bệnh viện cần một cơ sở dữ liệu giúp lưu trữ khai thác dữ liệu nhanh chóng và chính xác. Qua đó, dễ dàng quản lý các phòng khám, bác sĩ, bệnh nhân, thuốc,... dựa vào thông tin đã có sẵn từ trước nhằm tăng tốc độ cho các hoạt động khám chữa bệnh.

## Những yêu cầu cần phải giải quyết

### Tìm hiểu yêu cầu

Sau đây là bảng câu hỏi được thiết kế dành cho tất cả nhân sự đang làm việc tại Bệnh viện Đa khoa Quốc tế Becamex:

|  |  |
| --- | --- |
| 1. Trong bệnh viện đã có hệ thống quản lý hồ sơ bệnh án của bệnh nhân hay chưa?   🞎 Có 🞎 Chưa   1. Thông tin trong bệnh viện được lưu trữ và quản lý thủ công hay bằng máy tính?   🞎 Thủ công 🞎 Máy tính   1. Nếu đã có sử dụng máy tính trong việc lưu trữ thông tin, anh/chị đánh giá việc quản lý dữ liệu trong bệnh viện như thế nào? (Nếu không có thể bỏ qua)   🞎 Hiệu quả  🞎 Còn đơn sơ, rời rạc  🞎 Kém hiệu quả   1. Đâu là phần mềm anh/chị sử dụng nhiều nhất trong quá trình làm việc?   🞎 Trình xử lý văn bản  🞎 Bản tính  🞎 Cơ sở dữ liệu   1. Bệnh nhân lần đầu đến khám sẽ khai báo thông tin ở đâu? Thông tin đó sẽ được ghi nhận bằng giấy hay bằng máy tính?   ……………………………………………  ……………………………………………   1. Thời gian chờ trung bình của một bệnh nhân đến lượt khám là bao lâu?   ………………………………………….   1. Bệnh nhân có xem số thứ tự khám bằng những hình thức nào?   🞎 Màn hình tại sảnh  🞎 Qua app của bệnh viện  🞎 Qua tin nhắn   1. Bác sĩ có thể xem được thông tin của bệnh nhân mà mình đang khám hay không?   🞎 Có 🞎 Không   1. Bệnh nhân có thể xem kết quả xét nghiệm bằng những hình thức nào?   ……………………………………………..  ……………………………………………..   1. Có hệ thống nội bộ lưu trữ kết quả xét nghiệm của bệnh nhân hay không?   🞎 Có 🞎 Không | 1. Bệnh nhân có thể đóng tiền bằng hình thức nào?   🞎 Tiền mặt 🞎 Chuyển khoản   1. Bệnh nhân có thể xem những khoản tiền mình đã đóng bằng cách nào?   ………………………………………..  ………………………………………..   1. Ai là người ghi nhận bệnh nhân đã đóng tiền hay chưa?   …………………………………………   1. Các hóa đơn viện phí, đơn thuốc đã được kê tự động hay chưa?   🞎 Có 🞎 Chưa   1. Kho thuốc trong bệnh viện được quản lý thủ công hay tự động?   🞎 Thủ công 🞎Tự động   1. Ai là người chịu trách nhiệm quản lý kho thuốc của bệnh viện?   …………………………………………   1. Bệnh viện có thường xuyên xảy ra tình trạng thiếu thuốc và các dụng cụ y tế hay không?   🞎 Thường xuyên  🞎 Thỉnh thoảng  🞎 Rất ít khi xảy ra   1. Hãy liệt kê 3 vấn đề anh/chị cảm thấy bất tiện khi làm việc với hệ thống dữ liệu hiện tại của bệnh viện?   A……………………………………….  B……………………………………….  C……………………………………….   1. Trong số những vấn đề nêu trên, vấn đề nào mang lại nhiều rắc rối nhất?   …………………………………………  …………………………………………   1. Anh/chị có đề xuất nào giúp cải thiện vấn đề trên hay không?   …………………………………………  …………………………………………  …………………………………………  HẾT |

Bảng 4‑1. Bảng câu hỏi

### Xác định yêu cầu

Thông qua bước tìm hiểu yêu cầu, xác định được các yêu cầu cần giải quyết như sau:

* Đối với bệnh nhân: Bệnh nhân cần biết rõ những thông tin như
* Tổng chi phí trong thời gian điều trị tại bệnh viện.
* Số tiền đã đóng cho bệnh viện.
* Số tiền phải đóng cho bệnh viện.
* Chi tiết các khoản đã đóng cho bệnh viện.
* Đối với bác sĩ:
* Nắm được thông tin cá nhân của bệnh nhân mình đang điều trị.
* Biết rõ lịch sử bệnh án và diễn biến bệnh của bệnh nhân mình đang điều trị.
* Có thể dễ dàng tìm kiếm một số trường hợp tương tự để có giải pháp điều trị hiệu quả nhất.
* Thống kê được các loại bệnh, thống kê bệnh nhân theo mùa hoặc theo khu vực,… phục vụ cho việc nghiên cứu.
* Đối với nhân viên tài chính
* Tổng số tiền mà bệnh nhân đã đóng và chưa đóng.
* Thống kê tổng chi phí theo tháng, quý, năm.
* Số bệnh nhân điều trị trong tháng có bảo hiểm y tế và tổng chi phí của những bệnh nhân này để thanh toán với bên bảo hiểm y tế.
* Đối với thân nhân của bệnh nhân
* Biết được diễn biến điều trị, kết quả điều trị của người thân
* Có những trường hợp bác sĩ không thể nói trực tiếp cho bệnh nhân thì có thể thông báo qua người nhà của bệnh nhân.

## Hướng phát triển hệ thống quản trị cơ sở dữ liệu quản lý khám chữa bệnh

Sau khi tìm hiểu, phân tích và xác định các yêu cầu, em đề ra hướng phát triển đối với hệ thống quản trị cơ sở dữ liệu quản lý việc khám chữa bệnh cho Bệnh viện Becamex như sau:

* Đối tượng sử dụng: Đối tượng sử dụng chủ yếu là nhân viên và bác sĩ trong bệnh viện. Vì đây là hệ thống được thiết kế để sử dụng chuyên nghiệp nên những đối tượng sử dụng phải có sự hiểu biết nhất định về chuyên môn cũng như tin học.
* Lựa chọn hệ quản trị cơ sở dữ liệu: Có rất nhiều hệ quản trị cơ sở dữ liệu phổ biến như MySQL, Oracle, PostgreSQL, Access,…Tuy nhiên để phù hợp với cách thức quản lý tại bệnh viện này, em đã lựa chọn SQL Server 2014 là phần mềm được thiết kế bởi Microsoft để thiết kế cơ sở dữ liệu.

# PHÂN TÍCH HỆ THỐNG

## Mô tả quy trình khám chữa bệnh tại bệnh viện Becamex

Khi bệnh nhân có nhu cầu đến khám bệnh sẽ đăng ký trực tiếp tại Quầy đăng ký theo hướng dẫn của nhân viên. Hoặc có thể đặt lịch hẹn trước qua điện thoại và đến quầy đăng ký trước giờ đặt hẹn 15 phút để nhận số vào khám. Tại quầy đăng ký, bệnh nhân cung cấp thông tin cho nhân viên nhập vào máy tính. Đối với bệnh nhân có BHYT và không có BHYT sẽ chi trả chi phí khác nhau. Sau khi đã cung cấp thông tin cá nhân và thông tin bệnh. Bệnh nhân sẽ được phân bố đến phòng khám phù hợp. Bệnh nhân đợi khám và theo dõi màn hình tại sảnh hoặc tin nhắn trên điện thoại. Bệnh nhân đến đợi trước phòng khám, chờ gọi tên mời vào khám. Tại đây, bệnh nhân sẽ cung cấp các thông tin cho bác sĩ bao gồm các thông tin cá nhân, triệu chứng,... để bác sĩ lập Phiếu chỉ định. Bác sĩ sẽ yêu cầu bệnh nhân làm một số xét nghiệm và chỉ định liên quan. Sau khi nhận phiếu chỉ định của bác sĩ, bệnh nhân đến quầy thu ngân để thanh toán và thực hiện các chỉ định theo hướng dẫn. Bác sĩ tại các phòng khám đó thực hiện khám, xét nghiệm và lập kết quả xét nghiệm. Kết quả có được sẽ thông báo đến bệnh nhân qua tin nhắn, đồng thời được gửi đến cho bác sĩ tại phòng khám ban đầu. Sau đó, bệnh nhân được gọi quay trở lại để khám.

Sau khi khám bệnh, bệnh nhân thuộc một trong hai loại: Điều trị tại nhà hoặc điều trị tại bệnh viện.

* Nếu bệnh nhân điều trị tại nhà: Bác sĩ cho một đơn thuốc trong đó ghi đầy đủ tên thuốc, số lượng và cách dùng cũng như căn bệnh mà bác sĩ dự đoán. Bệnh nhân mang đơn thuốc đến quầy thu ngân đóng tiền. Nếu bệnh nhân có BHYT thì nhận thuốc tại quầy thuốc BHYT, ngược lại sẽ sang quầy thuốc thu phí, chờ gọi tên thanh toán và nhận thuốc.
* Nếu bệnh nhân điều trị tại bệnh viện: Bác sĩ sẽ lập một giấy nhập viện ghi đầy đủ thông tin của bệnh nhân và căn bệnh dự đoán sau đó được đưa đến khoa điều trị. Tại khoa điều trị, bệnh nhân sẽ được bác sĩ khám lại và cho đơn thuốc trong đó ghi đầy đủ tên thuốc, số lượng và cách dùng. Bác sĩ sẽ lập lịch khám định kỳ cho bệnh nhân (theo từng loại bệnh).

Trong quá trình điều trị, mỗi bệnh nhân đều có một bệnh án duy nhất bao gồm các thông tin về bệnh nhân, căn bệnh, diễn biến quá trình điều trị. Sau khi kết thúc điều trị tại bệnh viện, nhân viên thu ngân sẽ tính toán tiền với bệnh nhân, tính cả tiền bảo hiểm nếu có. Sau khi thanh toán, thu ngân sẽ lập hóa đơn viện phí cho bệnh nhân.

## Sơ đồ phân rã chức năng (FDD)

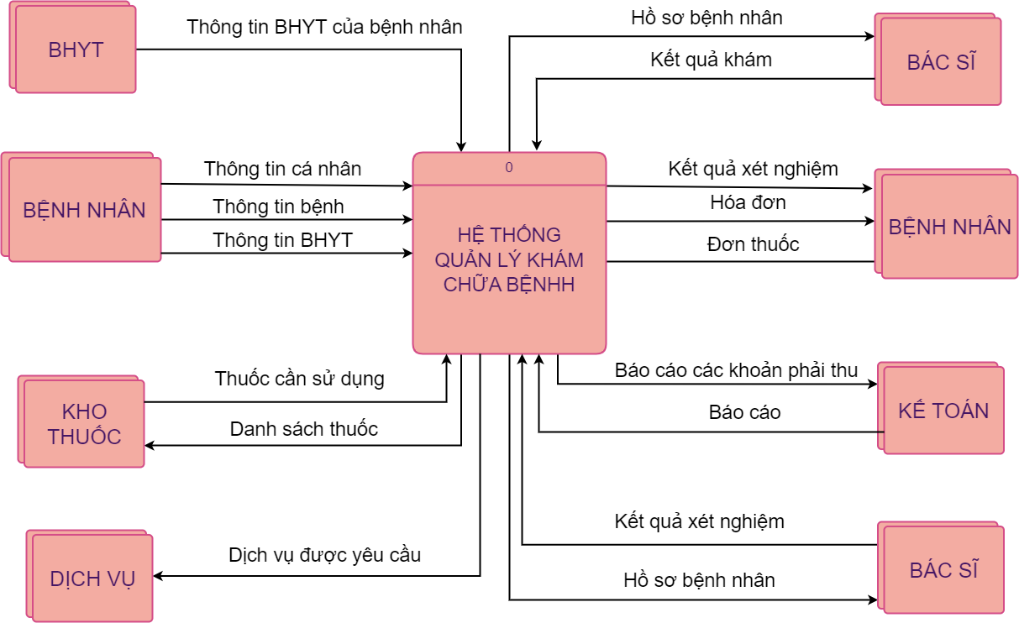
Sau khi tìm hiểu và phân tích, em đưa ra được sơ đồ phân rã chức năng cho Bệnh viện Đa khoa Quốc tế Becamex như sau:

Sơ đồ 5‑1. Sơ đồ phân rã chức năng của Bệnh viện Becamex

## Sơ đồ luồng dữ liệu (DFD)

### Sơ đồ luồng dữ liệu mức ngữ cảnh

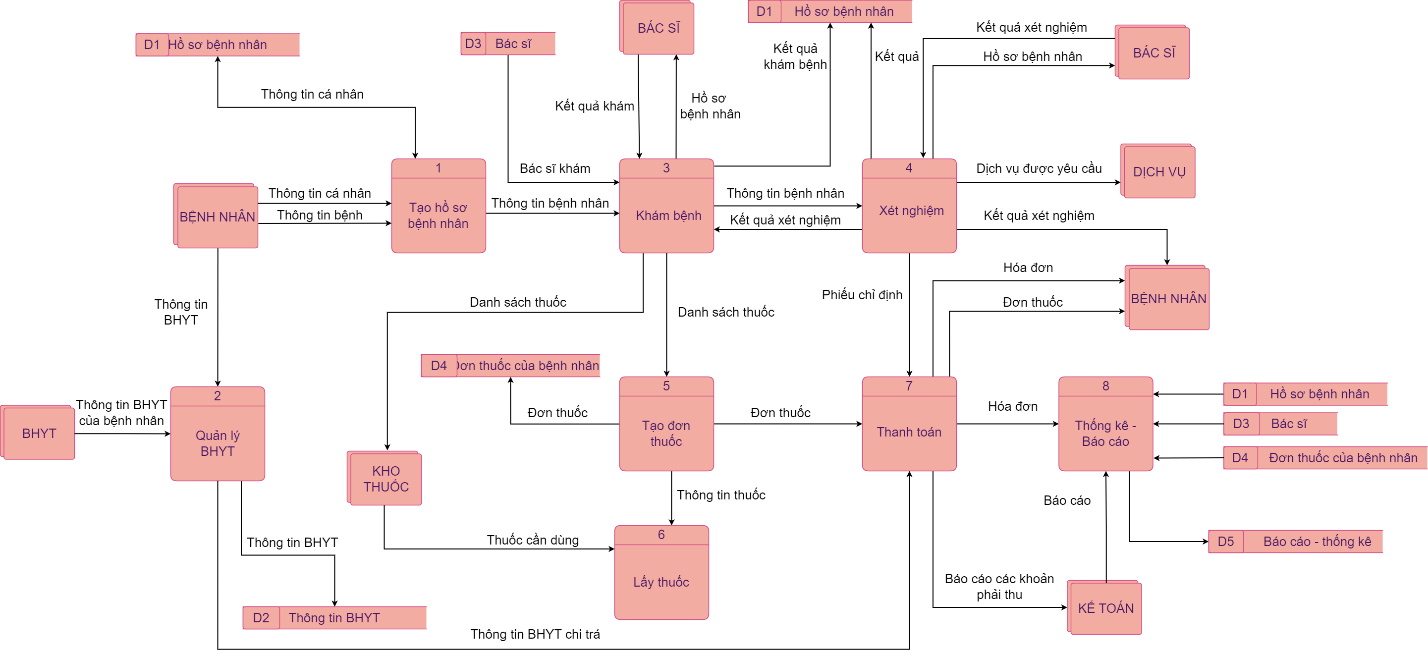
Sơ đồ mức ngữ cảnh là sơ đồ chung nhất, cung cấp cái nhìn toàn cảnh về chuyển động dữ liệu trong hệ thống. Biểu đồ này là mức cao nhất trong DFD và chỉ chứa một quy trình đại diện cho toàn bộ hệ thống (Quy trình số 0). Đầu vào và đầu ra được chỉ định trong sơ đồ ngữ cảnh không đổi trong tất cả các sơ đồ tiếp theo. Các thực thể ngoài của hệ thống bao gồm: Bệnh nhân, Bác sĩ, BHYT, Kho thuốc, Dịch vụ và Kế toán.



Sơ đồ 5‑2. Sơ đồ DFD mức ngữ cảnh

### Sơ đồ luồng dữ liệu mức 0

Tại đây, sơ đồ ban đầu được phân tách thành các quy tình con được đánh số bằng một số nguyên. Sơ đồ DFD mức 0 có thể có từ ba đến chín quy trình con và hiển thị các kho lưu trữ dữ liệu cũng như luồng dữ liệu cấp thấp hơn.



Sơ đồ 5‑3. Sơ đồ DFD mức 0

# THIẾT KẾ CƠ SỞ DỮ LIỆU

## Mô hình thực thể liên kết E-R

### Xác định các thực thể và thuộc tính khóa của thực thể

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Thực thể** | **Thuộc tính** | **Khóa chính** |
| 1 | Bệnh nhân | Mã bệnh nhân, họ và tên, giới tính,  ngày sinh, địa chỉ, số điện thoại, email | Mã bệnh nhân |
| 2 | Bác sĩ | Mã bác sĩ, họ và tên, giới tính, ngày sinh, địa chỉ, số điện thoại, email, chuyên môn, lương | Mã bác sĩ |
| 3 | Nhân viên | Mã nhân viên, họ và tên, giới tính, ngày sinh, địa chỉ, quê quán, số điện thoại, email, lương, chức vụ | Mã nhân viên |
| 4 | Khoa | Mã khoa, tên khoa, số điện thoại, email | Mã khoa |
| 5 | Phòng ban | Mã phòng ban, tên phòng ban ,số điện thoại, email | Mã phòng ban |
| 6 | Phòng khám | Mã phòng, tên phòng | Mã phòng |
| 7 | Bệnh án | Mã hồ sơ, ngày tạo | Mã hồ sơ |
| 8 | Nhóm dịch vụ | Mã nhóm, tên nhóm | Mã nhóm |
| 9 | Dịch vụ | Mã dịch vụ, tên dịch vụ, giá gốc, giá BHYT | Mã dịch vụ |
| 10 | Thuốc | Mã thuốc, tên biệt dược, tên hoạt chất, đơn vị, giá, mô tả | Mã thuốc |
| 11 | Đơn thuốc | Mã đơn thuốc, ngày tạo, tổng tiền. | Mã đơn thuốc |
| 12 | Nhà cung cấp | Mã nhà cung cấp, tên nhà cung cấp, địa chỉ, số điện thoại | Mã nhà cung cấp |
| 13 | Đơn hàng | Mã đơn hàng, ngày lập, tổng tiền | Mã đơn hàng |
| 14 | Phiếu chỉ định | Mã phiếu, ngày tạo | Mã phiếu |
| 15 | Phòng bệnh | Mã phòng, tên phòng,  số lượng giường | Mã phòng |
| 16 | Loại phòng | Mã số, tên loại phòng, số lượng phòng, giá phòng, giá BHYT | Mã số |
| 17 | Giường bệnh | Mã giường, loại giường, trạng thái | Mã giường |
| 18 | BHYT | Mã BHYT, nơi cấp, ngày bắt đầu, ngày kết thúc | Mã BHYT |
| 19 | Thân nhân | Mã thân nhân, họ và tên, giới tính, mối quan hệ | Mã thân nhân |
| 20 | Hóa đơn viện phí | Mã hóa đơn, phí dịch vụ, phí thuốc, phí nằm giường, tạm ứng, ngày tạo, tổng tiền | Mã hóa đơn |

Bảng 6‑1. Bảng các thực thể và thuộc tính khóa của thực thể

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

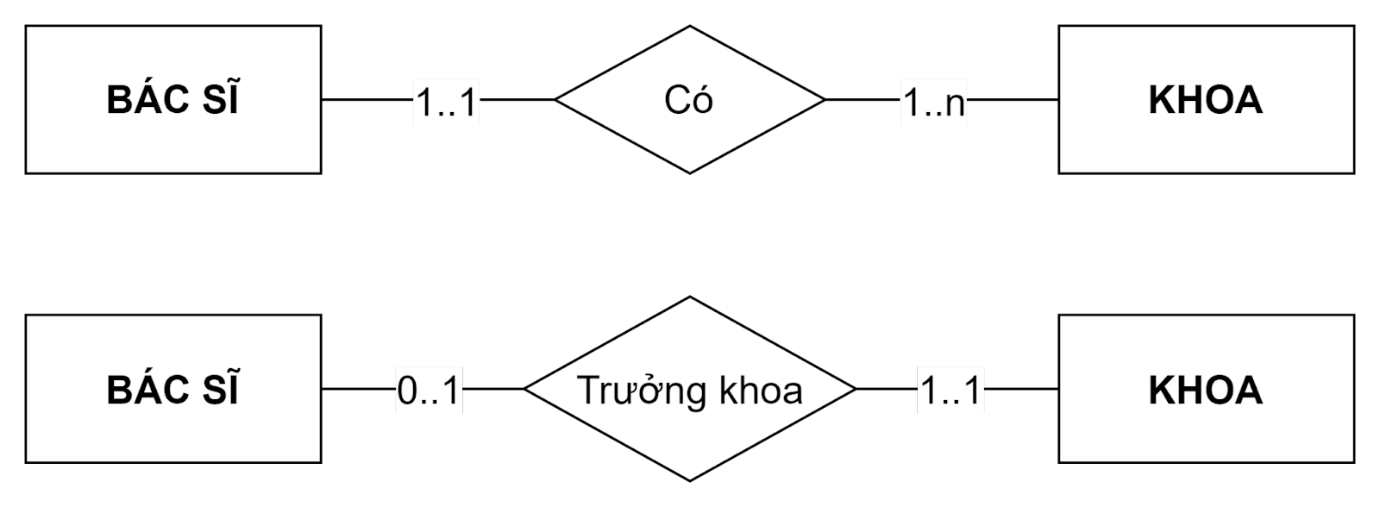
Mối quan hệ giữa thực thể **bác sĩ** và thực thể **bệnh nhân** là mối quan hệ **n-n**: Một bác sĩ có thể khám không hoặc nhiều bệnh nhân, một bệnh nhân được khám bởi một hoặc nhiều bác sĩ.



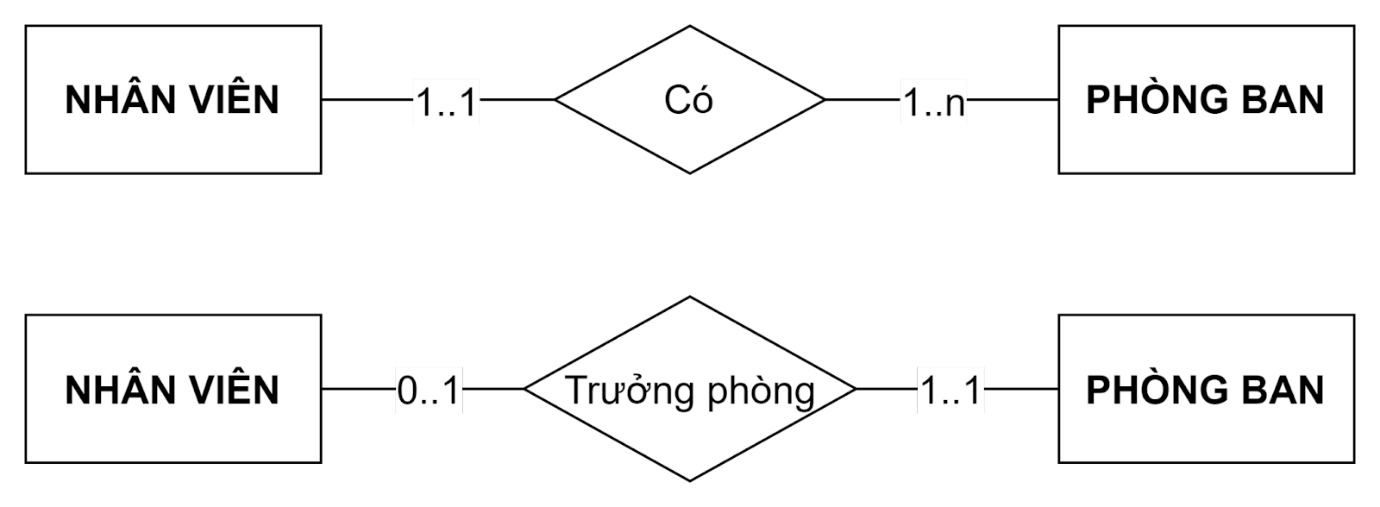
Mối quan hệ giữa thực thể **nhân viên** và thực thể **bệnh nhân** là mối quan hệ **n-n**: Một nhân viên có thể phục trách không hoặc nhiều bệnh nhân, một bệnh nhân được phụ trách bởi một hoặc nhiều nhân viên.



Mối quan hệ giữa thực thể **khoa** và thực thể **bác sĩ** là mối quan hệ **1-n**: Một bác sĩ chỉ thuộc một khoa, một khoa có thể có một hoặc nhiều bác sĩ. Mối quan hệ **1-1**: Một khoa chỉ có một trưởng khoa do bác sĩ khoa đó đảm nhận, một bác sĩ có thể không hoặc làm trưởng khoa của một khoa.



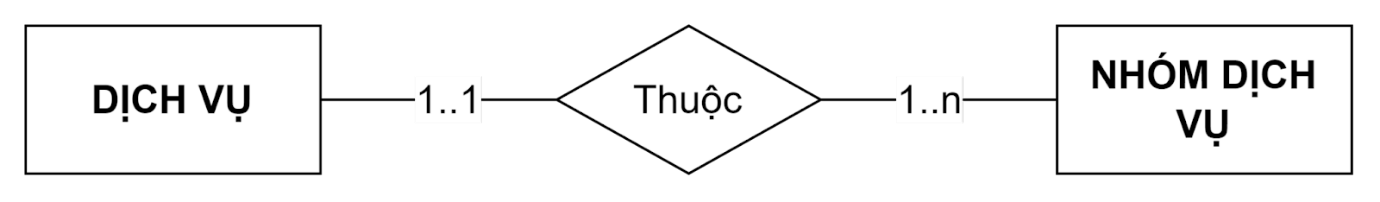
Mối quan hệ giữa thực thể **phòng ban** và thực thể **nhân viên** là mối quan hệ **1-n**: Một nhân viên chỉ thuộc một phòng ban, một phòng ban có thể có một hoặc nhiều nhân viên. Mối quan hệ **1-1**: Một phòng ban chỉ có một trưởng phòng do nhân viên phòng ban đó đảm nhận, một nhân viên có thể không hoặc làm trưởng phòng của một phòng ban.



Mối quan hệ giữa thực thể **bệnh nhân** và thực thể **bệnh án** là mối quan hệ **1-1**: Một bệnh nhân chỉ có một hồ sơ bệnh án và hồ sơ bệnh án chỉ thuộc duy nhất một bệnh nhân.



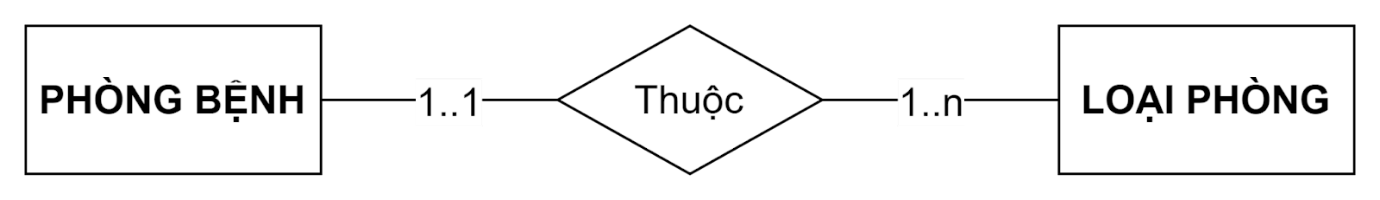
Mối quan hệ giữa thực thể **dịch vụ** và thực thể **nhóm dịch vụ** là mối quan hệ **1-n**: Một dịch vụ chỉ thuộc một nhóm dịch vụ nhất định, nhóm dịch vụ có thể có một hoặc nhiều dịch vụ.



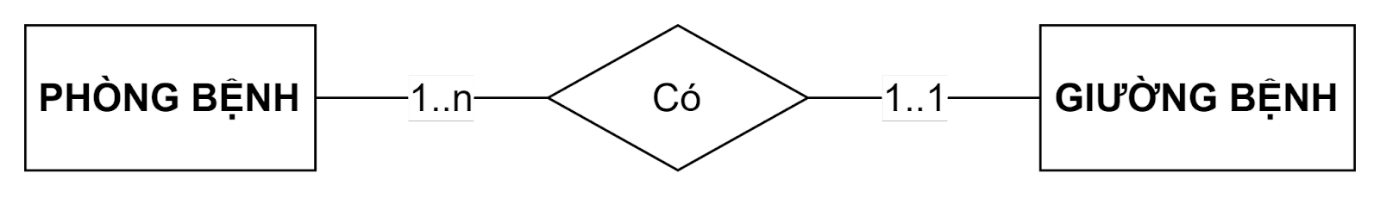
Mối quan hệ giữa thực thể **thuốc** và thực thể **nhà cung cấp** là mối quan hệ **1-n**: Một loại thuốc chỉ do một nhà cung cấp, một nhà cung cấp có thể cung cấp một hoặc nhiều loại thuốc.



Mối quan hệ giữa thực thể **phòng bệnh** và thực thể **loại phòng** là mối quan hệ **1-n**: Một phòng bệnh thuộc một loại phòng nhất định, 1 loại phòng có một hoặc nhiều phòng bệnh.



Mối quan hệ giữa thực thể **phòng bệnh** và thực thể **giường bệnh** là mối quan hệ **1-n**: Một phòng bệnh có một hoặc nhiều giường bệnh, một giường bệnh thuộc một phòng bệnh nào đó.



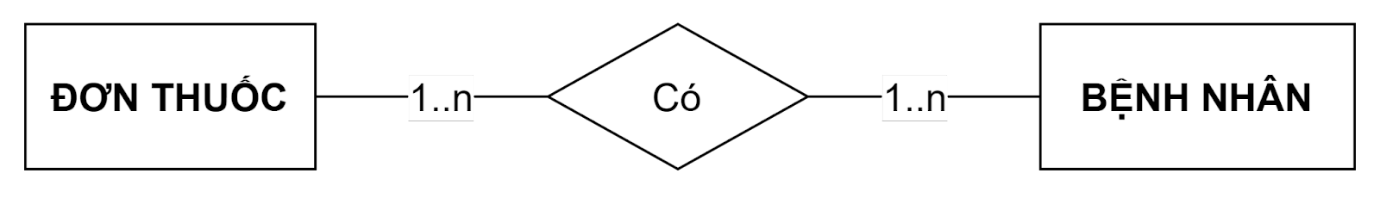
Mối quan hệ giữa thực thể **bệnh nhân** và thực thể **BHYT** là mối quan hệ **1-n**: Một bệnh nhân có thể có nhiều thẻ BHYT trong thời gian khác nhau, một thẻ BHYT chỉ thuộc duy nhất một bệnh nhân.



Mối quan hệ giữa thực thể **bệnh nhân** và thực thể **thân nhân** là mối quan hệ **1-n**: Một bệnh nhân có thể có không hoặc nhiều thân nhân, một thân nhân chỉ thuộc một bệnh nhân.



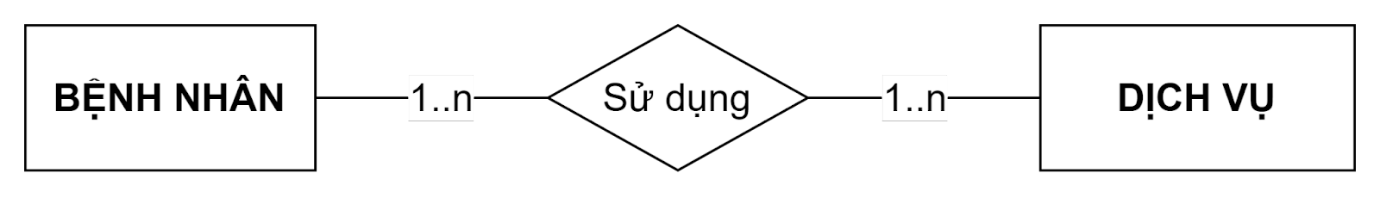
Mối quan hệ giữa thực thể **bệnh nhân** và thực thể **đơn thuốc** là mối quan hệ **1-n**: Một bệnh nhân có thể có nhiều đơn thuốc, một đơn thuốc chỉ thuộc một bệnh nhân



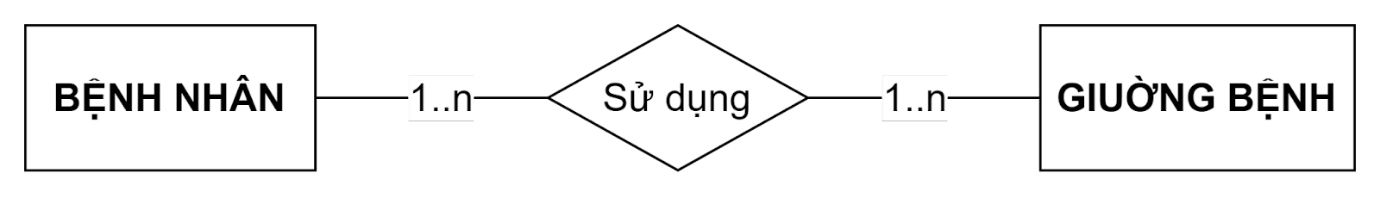
Mối quan hệ giữa thực thể **thuốc** và thực thể **đơn thuốc** là mối quan hệ **n-n**: Một đơn thuốc cho biết một hoặc nhiều loại thuốc, một loại thuốc có thể thuộc một hoặc nhiều đơn thuốc.



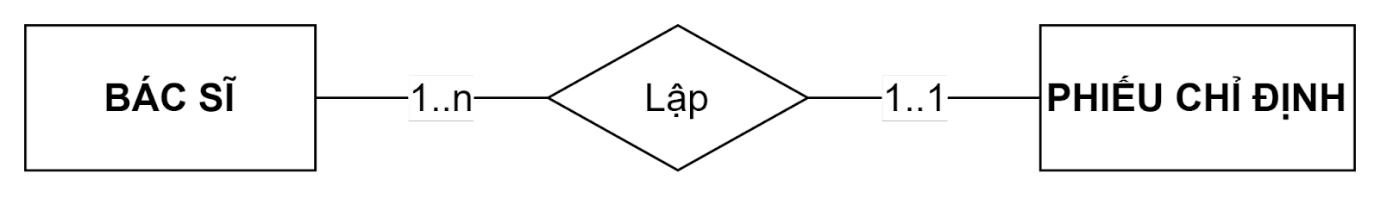
Mối quan hệ giữa thực thể **bệnh nhân** và thực thể **dịch vụ** là mối quan hệ **n-n**: Một bệnh nhân có thể sử dụng một hoặc nhiều dịch vụ, một dịch vụ được sử dụng bởi một hoặc nhiều bệnh nhân



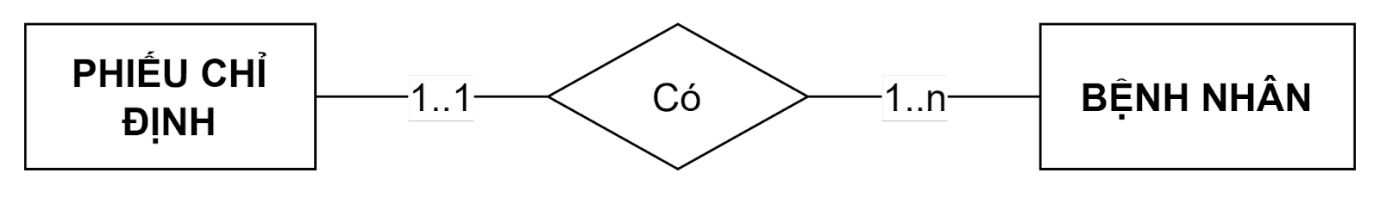
Mối quan hệ giữa thực thể **bệnh nhân** và thực thể **giường bệnh** là mối quan hệ **n-n**: Một bệnh nhân có thể sử dụng một hoặc nhiều giường bệnh, một giường bệnh được sử dụng bởi một hoặc nhiều bệnh nhân.



Mối quan hệ giữa thực thể **bác sĩ** và thực thể **phiếu chỉ định** là mối quan hệ **1-n**: Một bác sĩ có thể lập một hoặc nhiều phiếu chỉ định, một phiếu chỉ định chỉ do một bác sĩ lập.



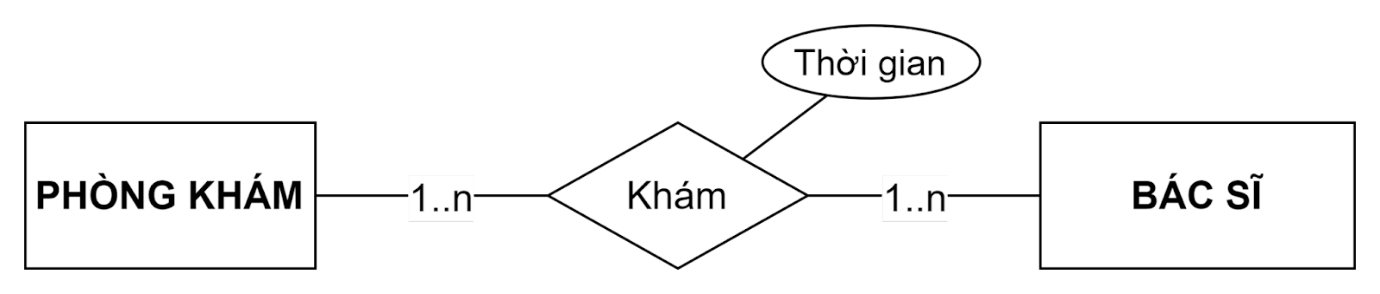
Mối quan hệ giữa thực thể **bệnh nhân** và thực thể **phiếu chỉ định** là mối quan hệ **1-n**: Một bệnh nhân có thể có một hoặc nhiều phiếu chỉ định, một phiếu chỉ định chỉ thuộc một bệnh nhân.



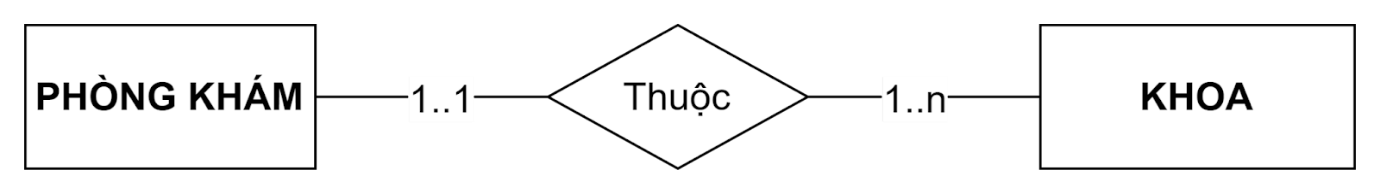
Mối quan hệ giữa thực thể **dịch vụ** và thực thể **phiếu chỉ định** là mối quan hệ **1-n**: Một phiếu chỉ định cho biết một dịch vụ, một dịch vụ có thể có nhiều phiếu chỉ định



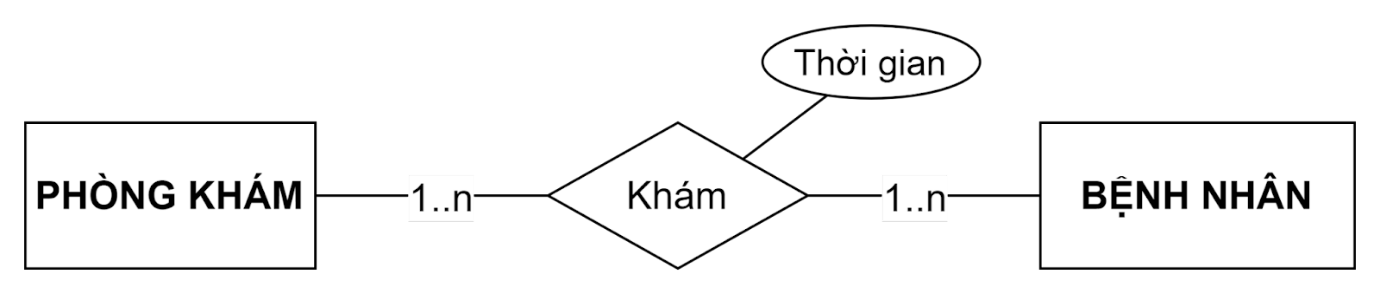
Mối quan hệ giữa thực thể **bác sĩ** và thực thể **phòng khám** là mối quan hệ **n-n**: Một bác sĩ có thể khám bệnh tại một hoặc nhiều phòng khám, một phòng khám có thể có một hoặc nhiều bác sĩ tại thời gian khác nhau.



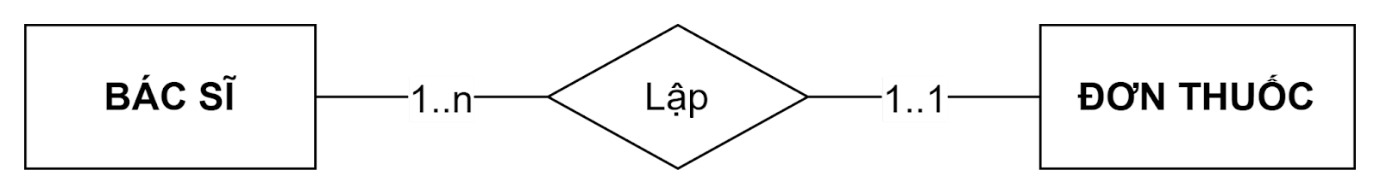
Mối quan hệ giữa thực thể **khoa** và thực thể **phòng khám** là mối quan hệ **1-n**: Một khoa có thể có một hoặc nhiều phòng khám, một phòng khám chỉ thuộc một khoa nhất định.



Mối quan hệ giữa thực thể **bệnh nhân** và thực thể **phòng khám** là mối quan hệ **n-n**: Một bệnh nhân có thể đến khám tại một hoặc nhiều phòng khám, một phòng khám được khám bởi nhiều bệnh nhân.



Mối quan hệ giữa thực thể **bác sĩ** và thực thể **đơn thuốc** là mối quan hệ **1-n**: Một bác sĩ có thể lập một hoặc nhiều đơn thuốc, một đơn thuốc được lập bởi một bác sĩ.



Mối quan hệ giữa thực thể **đơn hàng** và thực thể **thuốc** là mối quan hệ **n-n**: Một đơn hàng cho biết nhiều loại thuốc, một loại thuốc có thể được thể hiện trong nhiều đơn hàng.



Mối quan hệ giữa thực thể **đơn hàng** và thực thể **nhà cung cấp** là mối quan hệ **1-n**: Mỗi đơn hàng chỉ được thực hiện với một nhà cung cấp, một nhà cung cấp có thể thực hiện nhiều đơn hàng.



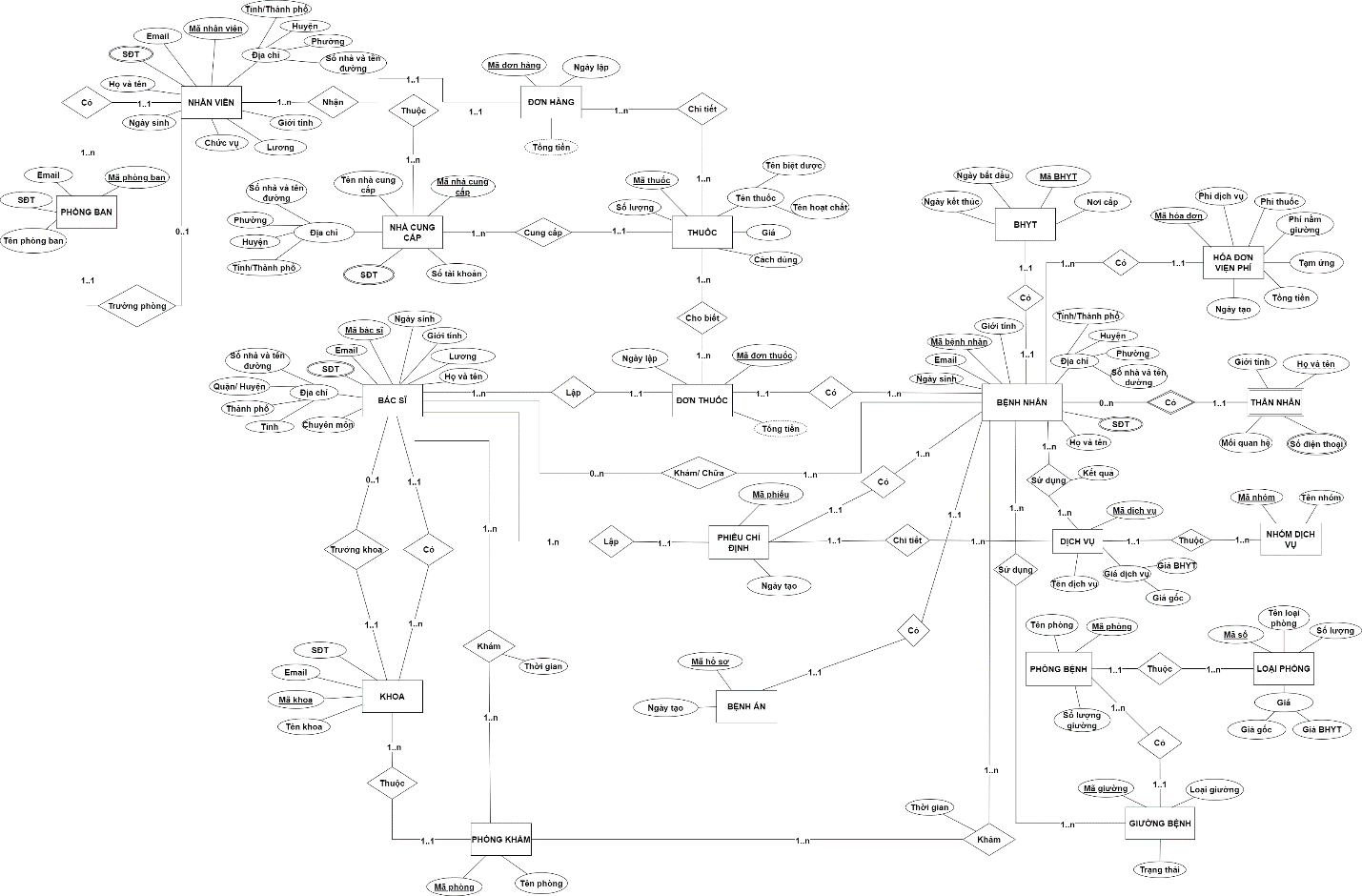
Mối quan hệ giữa thực thể **nhân viên** và thực thể **đơn hàng** là mối quan hệ **1-n**: Mỗi nhân viên có thể nhận một hoặc nhiều đơn hàng, mỗi đơn hàng chỉ được nhận bởi một nhân viên.



Mối quan hệ giữa thực thể **bệnh nhân** và thực thể **hóa đơn viện phí** là mối quan hệ **1-n**: Mỗi bệnh nhân có thể có một hoặc nhiều hóa đơn viện phí, mỗi hóa đơn viện phí chỉ thuộc về một bệnh nhân.



### Xây dựng mô hình thực thể liên kết



Sơ đồ 6‑1. Sơ đồ thực thể liên kết

## Mô hình dữ liệu quan hệ

### Chuyển đổi từ mô hình thực thể liên kết sang mô hình dữ liệu quan hệ

* Các thực thể chuyển thành các bảng.
* Các thuộc tính của thực thể chuyển thành các thuộc tính/cột của bảng.
* Các loại liên kết:
  + Liên kết 1-1: Khóa của bảng bên một bất kỳ trở thành thuộc tính kết nối (Khóa ngoại) của bảng kia.
  + Liên kết 1-n: Khóa của bảng bên 1 trở thành thuộc tính kết nối (Khóa ngoại) của bảng bên n.
  + Liên kết n-n: Tạo thêm một bảng liên kết để liên kết chúng. Nhờ đó, mỗi bảng trong quan hệ n-n sẽ có quan hệ 1-n với bảng liên kết.

Theo cách chuyển đổi, ta xác định được danh sách các bảng như sau:

#### Bảng bác sĩ\_DOCTOR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Doctor\_id | Int | Mã bác sĩ | Primary Key |
| 2 | Doctor\_name | Nvarchar(50) | Họ và tên | Not null |
| 3 | DSex | Char(1) | Giới tính | Not null |
| 4 | DBirthdate | Date | Ngày sinh | Not null |
| 5 | AddressNum | Varchar(50) | Số nhà | Null |
| 6 | Ward | Nvarchar(50) | Phường | Null |
| 7 | District | Nvarchar(50) | Huyện | Null |
| 8 | Province | Nvarchar(50) | Tỉnh | Null |
| 9 | DPhone | Varchar(10) | Số điện thoại | Not null |
| 10 | DEmail | Varchar(50) | Email | Not null |
| 11 | DSalary | Decimal(18,0) | Lương | Not null |
| 12 | DPosition | Nvarchar(30) | Chuyên môn | Not null |
| 13 | DFaculty\_id | Int | Mã khoa | Foreign Key |

**\* Giải thích:** Bảng Doctor lưu trữ tất cả các thông tin về bác sĩ của bệnh viện

* Mỗi bác sĩ có một mã không trùng với bất kỳ bác sĩ nào khác. Mã bác sĩ bắt đầu bằng số 1.
* Trường giới tính DSex là “F” tương ứng với Nữ và “M” tương ứng với Nam.
* Ngày sinh không được sau ngày hiện tại.
* Tuổi bác sĩ từ đủ 26 đến dưới 65.
* Bảng này nhận Mã khoa và Mã trưởng khoa làm khóa ngoại.

#### Bảng nhân viên\_Employee

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Employee\_id | Int | Mã nhân viên | Primary Key |
| 2 | Employee\_name | Nvarchar(50) | Họ và tên | Not null |
| 3 | ESex | Char(1) | Giới tính | Not null |
| 4 | EBirthdate | Datetime | Ngày sinh | Not null |
| 5 | AddressNum | Varchar(50) | Số nhà | Null |
| 6 | Ward | Nvarchar(100) | Phường | Null |
| 7 | District | Nvarchar(100) | Huyện | Null |
| 8 | Province | Nvarchar(100) | Tỉnh | Null |
| 9 | EPhone | Varchar(10) | Số điện thoại | Not null |
| 10 | EEmail | Varchar(50) | Email | Not null |
| 11 | ESalary | Decimal(18,0) | Lương | Not null |
| 12 | EPosition | Nvarchar(30) | Chức vụ | Not null |
| 13 | EDeparment\_id | Int | Mã phòng ban | Foreign Key |

**\* Giải thích:** Bảng Employee lưu trữ tất cả các thông tin về nhân viên của bệnh viện

* Mỗi nhân viên có một mã không trùng với bất kỳ nhân viên nào khác. Mã nhân viên bắt đầu bằng số 2.
* Trường giới tính ESex là “F” tương ứng với Nữ và “M” tương ứng với Nam.
* Ngày sinh không được sau ngày hiện tại.
* Tuổi bác sĩ từ đủ 18 đến dưới 65.
* Bảng này nhận Mã phòng ban và Mã trưởng phòng làm khóa ngoại.

#### Bảng bệnh nhân\_Patient

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Patient\_id | Varchar(10) | Mã bệnh nhân | Primary Key |
| 2 | Patient\_name | Nvarchar(50) | Họ và tên | Not null |
| 3 | PSex | Char(1) | Giới tính | Not null |
| 4 | PBirthdate | Datetime | Ngày sinh | Null |
| 5 | AddressNum | Varchar(50) | Số nhà | Null |
| 6 | Ward | Nvarchar(100) | Phường | Null |
| 7 | District | Nvarchar(100) | Huyện | Null |
| 8 | Province | Nvarchar(100) | Tỉnh | Null |
| 9 | PPhone | Varchar(10) | Số điện thoại | Not null |
| 10 | PEmail | Varchar(50) | Email | Null |
| 11 | Insurance\_id | Decimal(18,0) | Mã BHYT | Foreign Key |

**\* Giải thích:** Bảng Patient lưu trữ tất cả các thông tin về bệnh nhân đến khám chữa bệnh tại bệnh viện.

* Mỗi bệnh nhân có một mã không trùng với bất kỳ bệnh nhân nào khác
* Trường giới tính ESex là “F” tương ứng với Nữ và “M” tương ứng với Nam.
* Ngày sinh không được sau ngày hiện tại.
* Mã bệnh nhân gồm 10 ký tự trong đó 4 ký tự đầu là năm mà bệnh nhân đến khám, 6 ký tự còn lại là thứ tự bệnh nhân đó đến khám trong năm.

#### Bảng khoa\_Faculty

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Faculty\_id | Int | Mã khoa | Primary Key |
| 2 | Faculty\_name | Nvarchar(50) | Tên khoa | Unique |
| 3 | FEmail | Varchar(50) | Email | Null |
| 4 | FPhone | Varchar(10) | Số điện thoại | Not null |
| 5 | FDean\_id | Int | Mã trưởng khoa | Not null |

**\* Giải thích:** Bảng Faculty lưu trữ tất cả các thông tin về khoa mà bác sĩ đang làm việc.

#### Bảng phòng ban\_Department

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Department\_id | Int | Mã phòng ban | Primary Key |
| 2 | Deparment\_name | Nvarchar(50) | Tên phòng ban | Unique |
| 3 | DptEmail | Varchar(50) | Email | Null |
| 4 | DptPhone | Varchar(10) | Số điện thoại | Not null |
| 5 | DptManage\_id | Int | Mã trưởng phòng | Not null |

**\* Giải thích:** Bảng Department lưu trữ tất cả các thông tin về phòng ban mà bác sĩ đang làm việc.

#### Bảng phòng khám\_Clinic

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Clinic\_id | Int | Mã phòng khám | Primary Key |
| 2 | Clinic\_name | Nvarchar(50) | Tên phòng khám | Not null |
| 3 | Faculty\_id | Int | Mã khoa | Foreign Key |

**\* Giải thích:** Bảng Clinic lưu trữ tất cả các thông tin về phòng khám nơi bệnh nhân đến khám bệnh. Mỗi phòng khám thuộc một khoa nhất định nên bảng này nhận Faculty\_id (Mã khoa) làm khóa ngoại.

#### Bảng bệnh án\_Patient\_Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Patient\_report\_id | Decimal(18,0) | Mã hồ sơ | Primary Key |
| 2 | Patient\_id | Varchar(10) | Mã bệnh nhân | Primary Key |
| 3 | Create\_date | Datetime | Ngày tạo | Not null |

**\* Giải thích:** Bảng Patient\_Report lưu trữ danh sách các bệnh án của tất cả bệnh nhân.

#### Bảng thuốc\_Medication

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Medication\_id | Decimal(18,0) | Mã thuốc | Primary Key |
| 2 | Brand\_name | Varchar(100) | Tên biệt dược | Not null |
| 3 | Chemical\_name | Nvarchar(100) | Tên hoạt chất | Not null |
| 4 | Med\_unit | Nvarchar(20) | Đơn vị tính | Not null |
| 5 | Med\_price | Decimal(18,0) | Giá | Not null |
| 6 | Med\_desc | Nvarchar(255) | Mô tả | Null |
| 7 | Med\_supplier\_id | Int | Mã nhà cung cấp | Foreign Key |

**\* Giải thích:** Bảng Medication lưu trữ tất cả các thông tin về tất cả thuốc tại bệnh viện.

* Mỗi thuốc có một mã riêng không trùng với bất kỳ thuốc nào khác.
* Bảng Medication nhận Med\_supplier\_id làm khóa ngoại, trường này nhận giá trị tương ứng từ bảng Supplier.

#### Bảng đơn thuốc\_Prescription

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Prescription\_id | Decimal(18,0) | Mã đơn thuốc | Primary Key |
| 2 | Pres\_doctor\_id | Decimal(18,0) | Mã bác sĩ | Foreign Key |
| 3 | Pres\_patient\_id | Varchar(10) | Mã bệnh nhân | Foreign Key |
| 4 | Create\_date | Datetime | Ngày tạo | Not null |
| 5 | Total\_money | Decimal(18,0) | Tổng tiền | Not null |

**\* Giải thích:** Bảng Prescription lưu trữ tất cả các thông tin về đơn thuốc của bệnh nhân khám chữa bệnh tại bệnh viện.

* Bảng Prescription là mối liên kết giữa các thực thể Patient, Doctor nên nó nhận khóa của các thực thể tham gia vào mối liên kết này làm khóa ngoại.
* Ngoài ra, bảng có khóa chính là Prescription\_id.

#### Bảng nhà cung cấp\_Supplier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | Tên trường | Kiểu dữ liệu | Mô tả | Ràng buộc |
| 1 | Supplier\_id | Int | Mã nhà cung cấp | Primary Key |
| 2 | Supplier\_name | Nvarchar(50) | Tên nhà cung cấp | Not null |
| 3 | SAddress | Nvarchar(255) | Địa chỉ | Null |
| 4 | SPhone | Varchar(10) | Số điện thoại | Not null |

**\* Giải thích:** Bảng Supplier lưu trữ tất cả các thông tin về nhà cung cấp thuốc cho bệnh viện. Mỗi nhà cung cấp có một mã riêng không trùng với nhà cung cấp khác.

#### Bảng đơn hàng\_Orders

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Order\_id | Decimal(18,0) | Mã đơn hàng | Primary Key |
| 2 | Create\_date | Datetime | Ngày tạo | Not null |
| 3 | Total\_money | Decimal(18,0) | Tổng tiền | Not null |
| 4 | Order\_supplier\_id | Int | Mã nhà cung cấp | Foreign Key |
| 5 | Order\_employee\_id | Int | Mã nhân viên | Foreign Key |

**\* Giải thích:** Bảng Order lưu trữ tất cả các thông tin về đơn hàng thuốc mua từ nhà cung cấp.

* Bảng này nhận Supplier\_id từ bảng Supplier làm khóa ngoại. Đồng thời, nhận Employee\_id làm khóa tham chiếu, trường này nhận giá trị tương ứng từ bảng Empolyee cho biết nhân viên nào đã nhận đơn hàng này.
* Total\_money phải lớn hơn không.

#### Bảng chi tiết đơn hàng\_Order\_Details

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Order\_id | Decimal(18,0) | Mã đơn hàng | Primary Key |
| 2 | Medication\_id | Decimal(18,0) | Mã thuốc | Primary Key |
| 3 | Create\_date | Datetime | Ngày tạo | Not null |
| 4 | Quanity | Int | Số lượng | Not null |
| 5 | Total\_price | Decimal(18,0) | Thành tiền | Not null |

**\* Giải thích:** Bảng Order\_details là bảng trung gian từ 2 bảng Medication và Order nên nó nhận Medication\_id và Order\_id làm khóa chính.

* Quanity phải lớn hơn không.
* Total\_Price phải lớn hơn không.
* Total\_price = Quanity x Price của bảng Thuốc.
* Chi tiết của các Total\_price = Total\_money của bảng Order.

#### Bảng dịch vụ\_Service

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Service\_id | Int | Mã dịch vụ | Primary Key |
| 2 | Service\_name | Nvarchar(100) | Tên dịch vụ | Not null |
| 3 | Ser\_price | Decimal(18,0) | giá gốc | Not null |
| 4 | Insurance\_price | Decimal(18,0) | giá BHYT | Not null |
| 5 | Ser\_Service\_categories\_id | Int | Mã nhóm dịch vụ | Foreign Key |

**\* Giải thích:** Bảng Service lưu trữ tất cả các thông tin về các dịch vụ có trong bệnh viện.

* Mỗi dịch vụ có một mã riêng không trùng với dịch vụ khác.
* Bảng Serivce nhận Service\_categories\_id làm khóa ngoại, trường này nhận giá trị tương ứng trong bảng Service\_Categories.
* Giá BHYT phải bé hơn giá gốc.

#### Bảng nhóm dịch vụ\_Service\_Categories

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Service\_categories\_id | Decimal(18,0) | Mã dịch vụ | Primary Key |
| 2 | Service\_categories\_name | Nvarchar(50) | Tên dịch vụ | Not null |

**\* Giải thích:** Bảng Service\_Categories lưu trữ tất cả các thông tin về các dịch vụ có trong bệnh viện.

#### Bảng phiếu chỉ định\_Test\_form

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Test\_form\_id | Decimal(18,0) | Mã phiếu chỉ định | Primary Key |
| 2 | Create\_date | Datetime | Ngày tạo | Not null |
| 3 | Test\_patient\_id | Varchar(10) | Mã bệnh nhân | Foreign Key |
| 4 | Test\_doctor\_id | Int | Mã bác sĩ | Foreign Key |
| 5 | Test\_service\_id | Int | Mã dịch vụ | Foreign Key |

**\* Giải thích:** Bảng Test\_form lưu trữ tất cả các thông tin về các dịch vụ có trong bệnh viện.

* Bảng có Test\_form\_id làm khóa chính.
* Bảng Test\_form là mối liên kết giữa các thực thể Patient, Doctor, Service nên nó nhận khóa của các thực thể tham gia vào mối liên kết này làm khóa ngoại.

#### Bảng phòng bệnh\_Room

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Room\_id | Decimal(18,0) | Mã phòng bệnh | Primary Key |
| 2 | Room\_name | Nvarchar(50) | Tên phòng bệnh | Not null |
| 3 | Bed\_quanity | Int | Số lượng giường | Not null |
| 4 | Room\_categories\_id | Int | Mã loại phòng bệnh | Foreign Key |

**\* Giải thích:** Bảng Room lưu trữ tất cả các thông tin về phòng bệnh trong bệnh viện.

* Bảng có Room\_id là khóa chính.
* Bảng này nhận Room\_categories\_id từ bảng Room\_Categories làm khóa ngoại.
* Bed\_quanity không được nhỏ hơn 0.

#### Bảng loại phòng bệnh\_Room\_Categories

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Room\_categories\_id | Int | Mã loại phòng | Primary Key |
| 2 | Room\_categories\_name | Nvarchar(50) | Tên loại phòng | Not null |
| 3 | Room\_quanity | Int | Số lượng phòng | Not null |
| 4 | Room\_price | Decimal(18,0) | Giá gốc | Not null |
| 5 | Insurrance\_price | Decimal(18,0) | Giá BHYT | Not null |

**\* Giải thích:** Bảng Room\_categories lưu trữ tất cả các thông tin về loại phòng bệnh trong bệnh viện.

* Bảng có Room\_categories\_id là khóa chính.
* Insurrance\_price không lớn hơn Room\_price.

#### Bảng giường\_Bed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Bed\_id | Decimal(18,0) | Mã giường | Primary Key |
| 2 | Bed\_type | Nvarchar(50) | Loại giường | Not null |
| 3 | Bed\_status | Bit | Trạng thái | Null |
| 4 | Room\_id | Decimal(18,0) | Mã phòng | Foreign Key |

**\* Giải thích:** Bảng Bed lưu trữ tất cả các thông tin về giường bệnh trong bệnh viện.

* Bảng có Bed\_id là khóa chính.
* Bảng này nhận Room\_id từ bảng Room làm khóa ngoại.
* Trường Bed\_status nhận giá trị 0 tương ứng với trạng thái “available”, 1 tương ứng với “unavailable”.

#### Bảng BHYT\_Insurance

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Insurance\_id | Decimal(18,0) | Mã BHYT | Primary Key |
| 2 | Pulish\_date | Date | Ngày bắt đầu | Not null |
| 3 | Expire\_date | Date | Ngày kết thúc | Not null |
| 4 | Percentage | Int | Phần trăm | Not null |

**\* Giải thích:** Bảng Insurance lưu trữ tất cả các thông tin về BHYT của mỗi bệnh nhân.

* Bảng có Insurance\_id là khóa chính.
* Ngày bắt đầu phải bé hơn ngày kết thúc => Pulish\_date < Expire\_date
* Trường Percentage là một số nguyên nằm trong khoảng từ (1..100).

#### Bảng thân nhân\_Relative

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Patient\_id | Varchar(10) | Mã bệnh nhân | Primary Key |
| 2 | Relative\_name | Nvarchar(50) | Họ và tên | Not null |
| 3 | Rela\_Sex | Char(1) | Giới tính | Not null |
| 4 | Rela\_Birthdate | Date | Ngày sinh | Null |
| 5 | Rela\_relationship | Nvarchar(50) | Mối quan hệ | Null |

**\* Giải thích:** Bảng Relative lưu trữ tất cả các thông tin về thân nhân của bệnh nhân đến khám chữa bệnh tại bệnh viện.

* Do Thân nhân là thực thể yếu phụ thuộc vào thực thể Bệnh nhân nên nó nhận Patient\_id làm khóa chính.
* Trường giới tính ESex là “F” tương ứng với Nữ và “M” tương ứng với Nam.
* Ngày sinh không được sau ngày hiện tại.

#### Bảng hóa đơn viện phí\_Bill

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Bill\_id | Decimal(18,0) | Mã hóa đơn | Primary Key |
| 2 | Service\_charge | Decimal(18,0) | Chi phí sử dụng dịch vụ | Null |
| 3 | Medication\_charge | Decimal(18,0) | Chi phí thuốc | Null |
| 4 | Room\_charge | Decimal(18,0) | Chi phí phòng bệnh | Null |
| 5 | Advance | Decimal(18,0) | Tạm ứng | Null |
| 6 | Create\_date | Datetime | Ngày tạo | Not null |
| 7 | Total\_bill | Decimal(18,0) | Tổng hóa đơn | Not null |
| 8 | Patient\_id | Varchar(10) | Mã bệnh nhân | Foreign Key |
| 9 | Employee\_id | Int | Mã nhân viên | Foreign Key |

**\* Giải thích:** Bảng Bill lưu trữ tất cả các thông tin về hóa đơn viện phí mà bệnh nhân đã đóng.

* Bảng này có Bill\_id làm khóa chính. Nó  nhận Patient\_id từ bảng Patient và Employee\_id từ bảng Employee làm khóa ngoại.
* Total\_bill bằng tổng các chi tiết.

#### Bảng Dịch vụ sử dụng\_Service\_Using

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Ser\_Using\_id | Decimal(18,0) | Mã sử dụng dv | Primary Key |
| 2 | Patient\_id | Varchar(10) | Mã bệnh nhân | Primary Key |
| 3 | Service\_id | Decimal(18,0) | Mã dịch vụ | Primary Key |
| 4 | Weight | Int | Cân nặng | Null |
| 5 | Height | Int | Chiều cao | Null |
| 6 | Blood\_pressure | Int | Huyết áp | Null |
| 7 | Test\_result | Nvarchar(100) | Kết quả | Not null |
| 8 | Create\_date | Datetime | Ngày | Not null |
| 9 | Test\_doctor\_id | Decimal(18,0) | Mã bác sĩ | Foreign Key |

**\* Giải thích:** Bảng Service\_Using là bảng trung gian từ 2 bảng Service và Patient nên nó nhận Patient\_id và Service\_id làm khóa chính. Bảng nhận Doctor\_id từ bảng Doctor làm khóa ngoại. Và nó có khóa riêng là Ser\_Using\_id.

#### Bảng Luồng khám bệnh\_Patient\_Flow

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Patient\_flow\_id | Decimal(18,0) | Mã khám bệnh | Primary Key |
| 2 | Patient\_id | Varchar(10) | Mã bệnh nhân | Primary Key |
| 3 | Clinic\_id | Int | Mã phòng khám | Primary Key |
| 4 | Doctor\_id | Decimal(18,0) | Mã bác sĩ | Primary Key |
| 5 | Patient\_type | Bit | Đối tượng | Not null |
| 6 | Date | Datetime | Ngày khám | Not null |
| 7 | Diagnose | Nvarchar(255) | Chẩn đoán | Null |

**\* Giải thích:** Bảng luồng khám bệnh là mối kết hợp giữa các thực thể bệnh nhân, phòng khám, bác sĩ nên nó nhận tất cả các khóa của các thực thể tham gia vào mối kết hợp làm khóa chính, các giá trị của các trường này được lấy tương ứng từ các bảng tương ứng.

* Bảng có khóa riêng là Mã khám bệnh.
* Đối tượng: 1 - Có BHYT, 0 - Không có BHYT.

#### Bảng Nằm giường\_Stay

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Stay\_id | Decimal | Mã nằm giường | Primary key |
| 2 | Patient\_id | Varchar(10) | Mã bệnh nhân | Primary Key |
| 3 | Bed\_id | Int | Mã giường | Primary Key |
| 4 | Begin\_date | Datetime | Ngày bắt đầu | Not null |
| 5 | End\_date | Datetime | Ngày kết thúc | Not null |

**\* Giải thích:** Bảng Stay bệnh là mối kết hợp giữa các thực thể bệnh nhân và giường bệnh nên nó nhận Patient\_id và Bed\_id làm khóa chính, các giá trị của các trường này được lấy tương ứng từ các bảng tương ứng. Nó có khóa riêng là Stay\_id

#### Bảng Chi tiết đơn thuốc\_Pres\_Detail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Mô tả** | **Ràng buộc** |
| 1 | Prescription\_id |  | Mã đơn thuốc | Primary Key |
| 2 | Medication\_id | Decimal(18,0) | Mã thuốc | Primary Key |
| 3 | Quanity | Int | Số lượng | Not null |
| 4 | Create\_date | Datetime | Ngày tạo | Not null |

**\* Giải thích:** Bảng chi tiết đơn thuốc là bảng liên kết giữa 2 bảng đơn thuốc và thuốc nên nó nhận 2 khóa của bảng này làm khóa chính.

### Mối quan hệ giữa các bảng

#### Quản lý thông tin bệnh nhân

* Quản lý thông tin cá nhân:



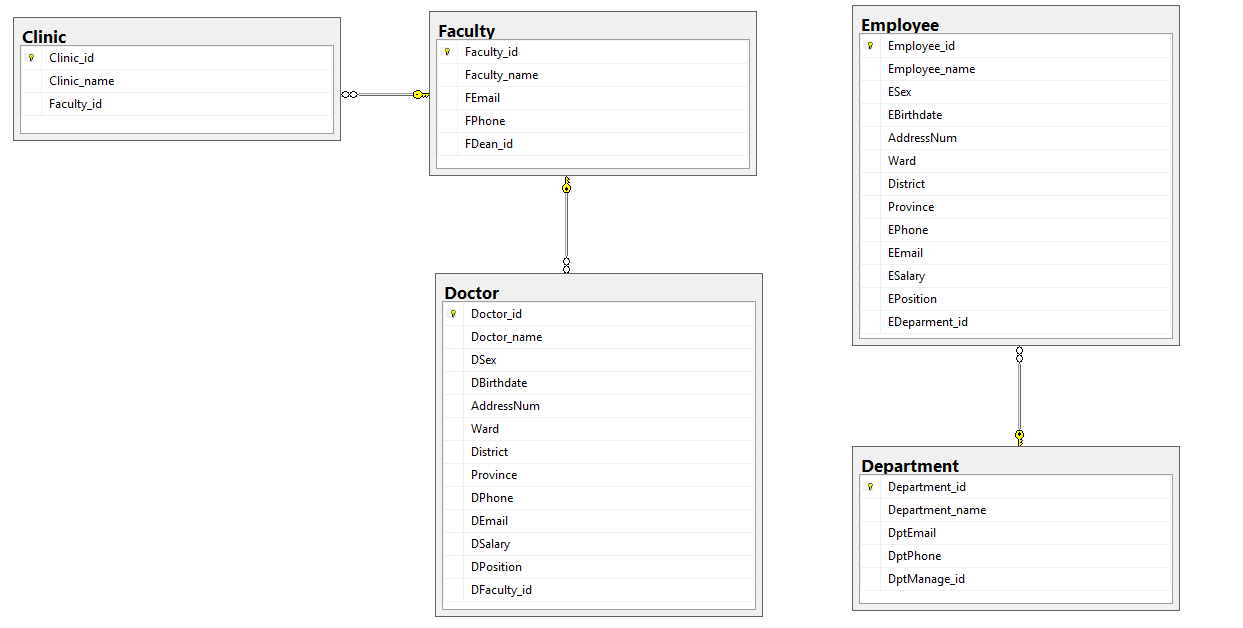
Hình 6‑1. Mối quan hệ của các bảng trong quản lý thông tin cá nhân

* Quản lý thông tin bệnh án:



Hình 6‑2. Mối quan hệ của các bảng trong quản lý bệnh án

#### Quản lý thông tin nhân sự của bệnh viện



Hình 6‑3. Mối quan hệ của các bảng trong quản lý nhân sự

#### Quản lý thuốc



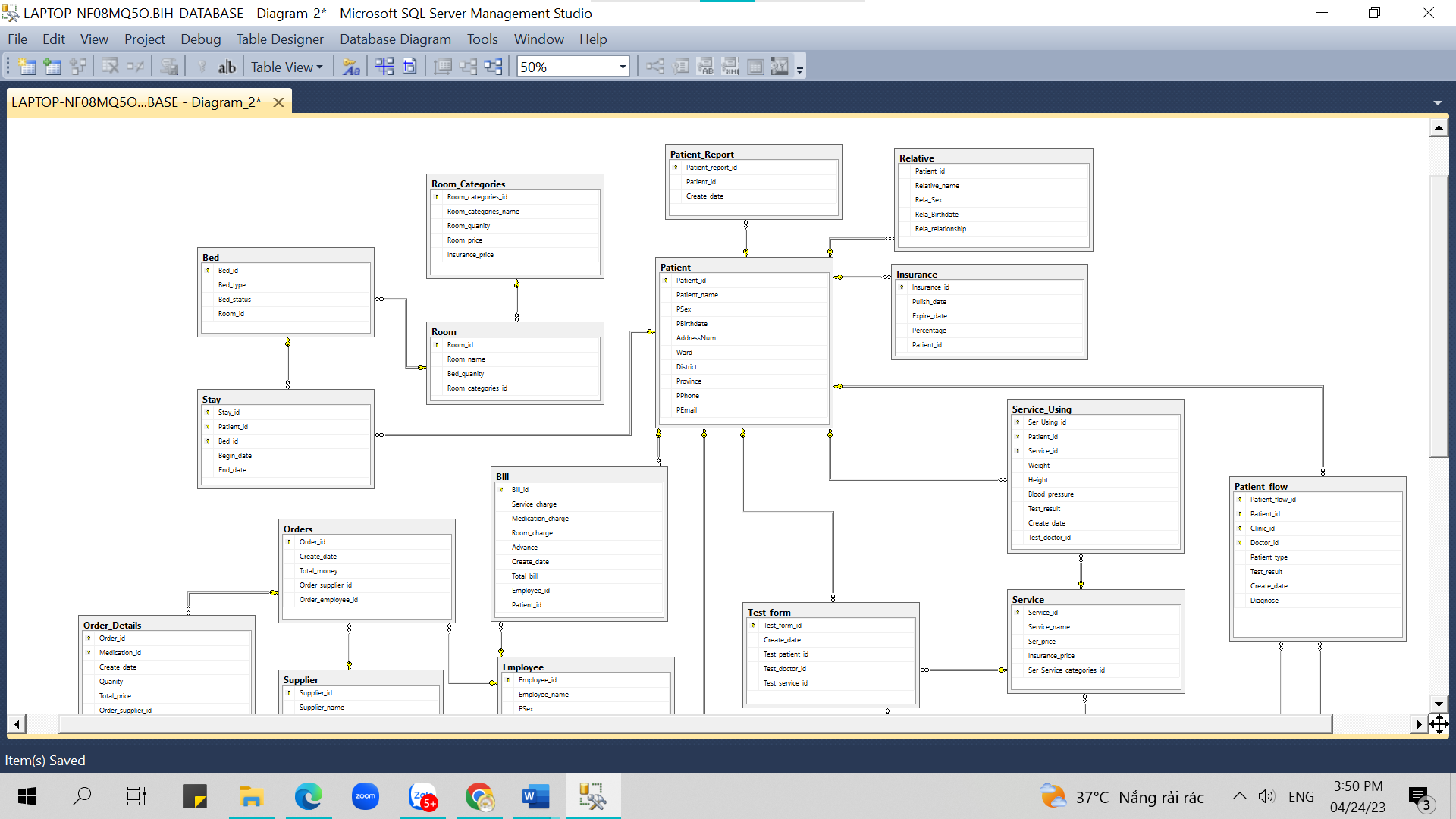
Hình 6‑4. Mối quan hệ của các bảng trong quản lý thuốc

#### Quản lý thông tin tài chính

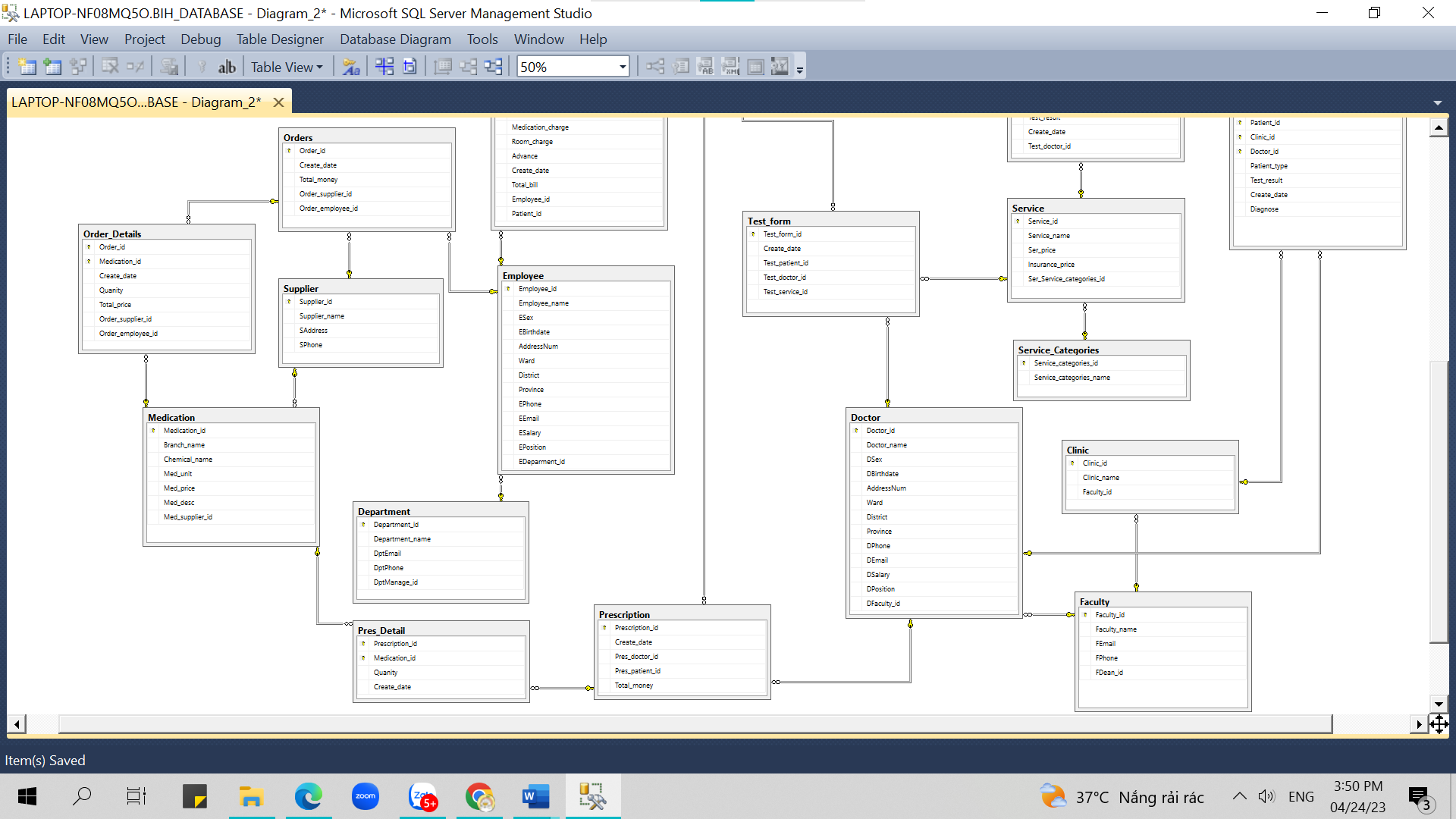


Hình 6‑5. Mối quan hệ của các bảng trong quản lý thông tin tài chính.

### Mô hình mối quan hệ tổng thể



Hình 6‑6. Mối quan hệ tổng thể giữa toàn bộ các bảng trong cơ sở dữ liệu (1)



Hình 6‑7. Mối quan hệ tổng thể giữa toàn bộ các bảng trong cơ sở dữ liệu (2)

## Thiết kế câu lệnh truy vấn SQL

### Tạo cơ sở dữ liệu

CREATE DATABASE BIH\_DATABASE

### Tạo bảng

#### Câu lệnh tạo bảng Bệnh nhân

CREATE TABLE Patient

(Patient\_id Varchar(10) PRIMARY KEY DEFAULT DBO.Auto\_patient\_id(),

Patient\_name Nvarchar(50) NOT NULL,

PSex Char(1) NOT NULL CHECK (PSex in ('F','M')),

PBirthdate Date NULL,

AddressNum varchar(30) NULL,

Ward Nvarchar(30) NULL,

District Nvarchar(30) NULL,

Province Nvarchar(30) NULL,

PPhone Varchar(10) NOT NULL,

PEmail Varchar(50) NULL)

#### Câu lệnh tạo bảng Thuốc

CREATE TABLE Medication

(Medication\_id Decimal(18,0) NOT NULL PRIMARY KEY,

Branch\_name Nvarchar(100) NOT NULL,

Chemical\_name Nvarchar(100) NOT NULL,

Med\_unit Nvarchar(20) NOT NULL,

Med\_price Decimal(18,0) NOT NULL,

Med\_desc Nvarchar(255) Null,

Med\_supplier\_id Int NOT NULL)

### Thêm ràng buộc

Sau đây là câu lệnh thêm ràng buộc khóa ngoại (foreign key constraint) và ràng buộc kiểm tra (check constraint) cho bảng Doctor:

ALTER TABLE Doctor

ADD FOREIGN KEY (DFaculty\_id) REFERENCES Faculty(Faculty\_id) ON UPDATE CASCADE

ALTER TABLE Doctor

ADD CHECK (DBirthdate < GETDATE())

ALTER TABLE Doctor

ADD CHECK (LEFT(Doctor\_id,1) = 1)

### Tạo function

* Tạo function tính các chi phí mà bệnh nhân đã đóng cho bệnh viện với tham số truyền vào là mã bệnh nhân:

---Chi phí dịch vụ

CREATE fUNCTION fun\_Service\_Charge (@Patient\_id VARCHAR(10))

RETURNS INT

AS

BEGIN

DECLARE @Service\_charge INT

IF (NOT EXISTS (SELECT \* FROM Service\_Using s WHERE s.Patient\_id = @Patient\_id))

RETURN 0

ELSE

SET @Service\_charge = (SELECT SUM(s.Ser\_price)

FROM Service s , Service\_Using su

WHERE s.Service\_id = su.Service\_id

AND su.Patient\_id = @Patient\_id

GROUP BY su.Patient\_id)

RETURN @Service\_charge

END

GO

---Chi phí thuốc

CREATE fUNCTION fun\_Medication\_Charge (@Patient\_id VARCHAR(10))

RETURNS INT

AS

BEGIN

DECLARE @Medication\_charge INT

IF (NOT EXISTS (SELECT \* FROM Prescription p WHERE p.Pres\_patient\_id = @Patient\_id))

RETURN 0

ELSE

SET @Medication\_charge = (SELECT SUM(m.Med\_price\*pd.Quanity)

FROM Medication m , Pres\_Detail pd, Prescription p

WHERE m.Medication\_id = pd.Medication\_id

AND p.Pres\_patient\_id = @Patient\_id

AND P.Prescription\_id = pd.Prescription\_id

GROUP BY p.Pres\_patient\_id)

RETURN @Medication\_charge

END

GO

---Chi phí nằm giường

CREATE fUNCTION fun\_Stay\_Charge (@Patient\_id VARCHAR(10))

RETURNS INT

AS

BEGIN

DECLARE @Room\_charge INT

IF (NOT EXISTS (SELECT \* FROM Stay s WHERE s.Patient\_id = @Patient\_id))

RETURN 0

ELSE

SET @Room\_charge = (SELECT SUM(rm.Room\_price\*(DATEDIFF(day, s.Begin\_date, S.End\_date)))

FROM Room\_Categories rm, Room r, Bed b, Stay s

WHERE rm.Room\_categories\_id = r.Room\_categories\_id

AND r.Room\_id = b.Room\_id

AND b.Bed\_id = s.Bed\_id

AND s.Patient\_id = @Patient\_id

GROUP BY s.Patient\_id)

RETURN @Room\_charge

END

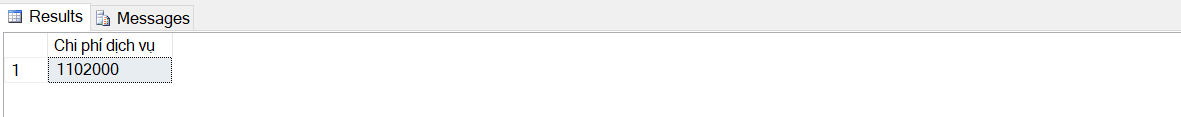
GO

* Câu lệnh gọi hàm

SELECT dbo.fun\_Service\_Charge('2023000018') AS N'Chi phí dịch vụ'

GO

* Tập kết quả được tạo bởi function



Hình 6‑8. Kết quả của câu lệnh function

### Tạo stored procedure

* Tạo thủ tục tính tổng các chi phí bệnh nhân đã đóng cho bệnh viện với tham số truyền vào là mã bệnh nhân

----Tổng chi phí

IF OBJECT\_ID ('sp\_Patient\_charge', 'P') IS NOT NULL

DROP PROCEDURE sp\_Patient\_charge

GO

CREATE PROCEDURE sp\_Patient\_charge

@Patient\_id VARCHAR(10)

AS

IF (NOT EXISTS (SELECT \* FROM Patient p WHERE p.Patient\_id = @Patient\_id))

BEGIN

RAISERROR('Not a valid PATIENT ID',11,1)

END

ELSE

BEGIN

SELECT p.Patient\_name AS N'Họ và tên', CASE WHEN p.PSex ='F' THEN N'Nữ' ELSE N'Nam' END AS N'Giới tính',

YEAR(GETDATE()) - YEAR(p.PBirthdate) AS N'Tuổi',

dbo.fun\_Service\_Charge (@Patient\_id) AS N'Chi phí dịch vụ',

dbo.fun\_Medication\_Charge(@Patient\_id) AS N'Chi phí thuốc',

dbo.fun\_Stay\_Charge(@Patient\_id) AS N'Chi phí nằm giường'

FROM Patient p

WHERE p.Patient\_id = @Patient\_id

END

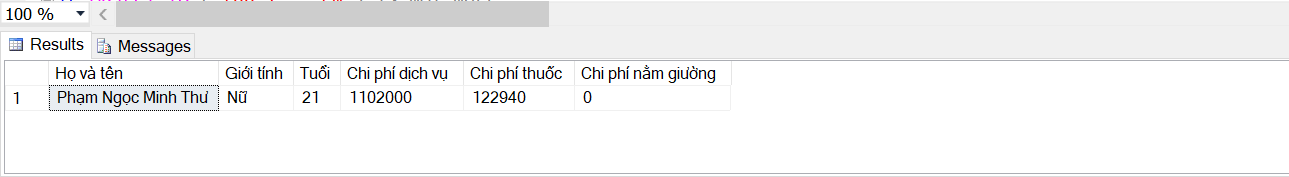
GO

* Câu lệnh gọi thủ tục

EXEC sp\_Patient\_charge'2023000018';

GO

* Tập kết quả được tạo bởi thủ tục



Hình 6‑9. Kết quả trả về từ câu lệnh gọi thủ tục

### Tạo view

* Tạo view hiển thị thông tin cá nhân của một bệnh nhân

IF OBJECT\_ID ('Medication\_list', 'V') IS NOT NULL

DROP VIEW Medication\_list

GO

CREATE VIEW Medication\_list AS

SELECT m.Medication\_id AS N'Mã thuốc', m.Branch\_name AS N'Tên biệt dược',

m.Chemical\_name AS N'Tên hoạt chất', m.Med\_unit AS N'Đơn vị',

m.Med\_desc AS N'Mô tả', s.Supplier\_name AS N'Nhà cung cấp'

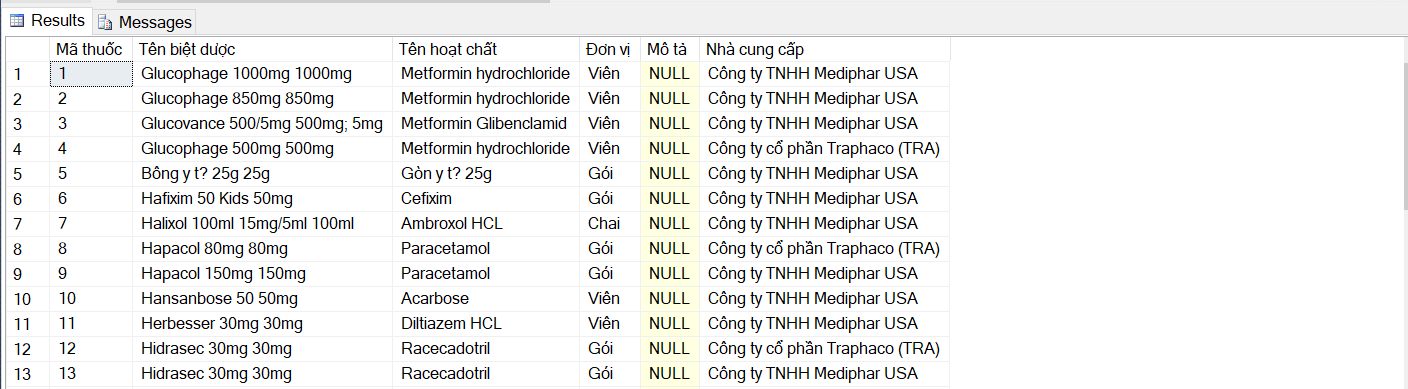
FROM Medication m JOIN Supplier s ON m.Med\_supplier\_id = s.Supplier\_id

GO

* Câu lệnh gọi view

SELECT \* FROM Medication\_list

* Nội dung của view



Hình 6‑10. Kết quả từ câu lệnh gọi view

### Tạo Trigger

* Tạo Trigger kiểm tra tính hợp lệ của tuổi bác sĩ phải từ đủ 26 đến dưới 65 tuổi.

CREATE TRIGGER DoctorAgeTrigger

ON DOCTOR AFTER INSERT

AS

DECLARE @Age INT

SELECT @Age=YEAR(d.DBirthdate) FROM Doctor d JOIN inserted i ON d.Doctor\_id = i.Doctor\_id

IF YEAR(GETDATE()) - @Age < 26 OR YEAR(GETDATE()) - @Age > 64

BEGIN

RAISERROR ('This doctor is not between 26 and 65 years old. We cannot sign a contact with him/her', 16,1)

ROLLBACK TRANSACTION

END

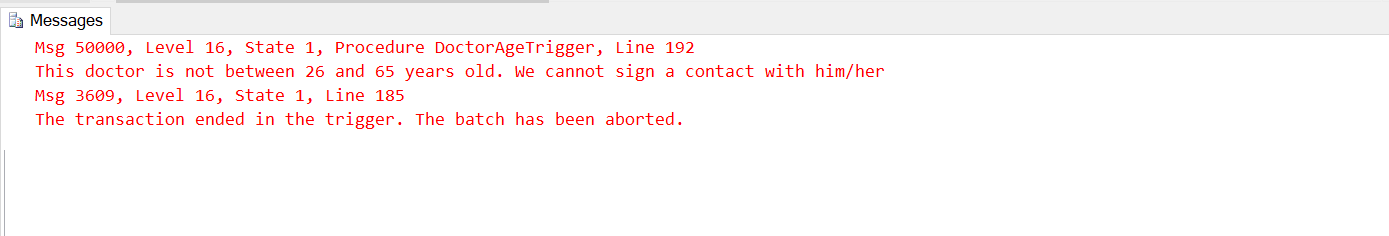
GO

* Câu lệnh Insert làm phát sinh Trigger

INSERT INTO Doctor (Doctor\_id, Doctor\_name, DSex, DBirthdate, AddressNum, Ward, District, Province, DPhone, DEmail, DSalary, DPosition, DFaculty\_id)

VALUES ('10022', N'Nguyễn Văn An', 'M', '08/03/2010', '19A', N'Phường 3', Quận 1', N'TP.Hồ Chí Minh', '0912345678', 'NguyenVanA@gmail.com', 31000, N'Chuyên khoa Xét nghiệm',1)

* Phản hồi từ hệ thống



Hình 6‑11. Phản hồi từ hệ thống khi Insert dữ liệu mới

### Tạo các bản sao lưu

Có 4 loại sao lưu bao gồm:

* Full database backups (Bản sao lưu đầy đủ): Sao chép tất cả data files, user data, database objects trong một cơ sở dữ liệu.
* Differential data backups (Bản sao lưu khác biệt) : Sao chép những thay đổi trong tất cả data files kể từ lần Full Backup gần nhất.
* File or file group backups: Sao chép một data file đơn hoặc một file group.
* Transaction log backups (Sao lưu nhật kí): Sao chép một cách có thứ tự tất cả các thao tác chứa trong LDF kể từ lần Transaction log backup gần nhất.

Sau đây là câu lệnh tạo ba bảng sao lưu đầy đủ, sao lưu khác biệt và sao lưu nhật kí:

---Sao lưu đầy đủ

BACKUP DATABASE [BIH\_DATABASE] TO DISK = N'C:\data\BIH\_FullBK.BAK'

WITH NOFORMAT, NOINIT,

NAME = N'BIH-Full Database Backup',

SKIP, NOREWIND, NOUNLOAD, STATS = 10

GO

--Sao lưu khác biệt

BACKUP DATABASE [BIH\_DATABASE] TO DISK = N'C:\data\BIH\_DiffBK.BAK'

WITH DIFFERENTIAL, NOFORMAT, NOINIT,

NAME = N'BIH-Full Database Backup',

SKIP, NOREWIND, NOUNLOAD, STATS = 10

GO

--Sao lưu nhật kí

BACKUP LOG [BIH\_DATABASE] TO DISK = N'C:\data\BIH\_LogBK.BAK'

WITH NOFORMAT, NOINIT,

NAME = N'BIH-Full Database Backup',

SKIP, NOREWIND, NOUNLOAD, STATS = 10

GO

### Một số câu lệnh truy vấn phục vụ cho việc quản lý

#### Quản lý thông tin cá nhân của bệnh nhân

* Tạo khùng nhìn thống kê danh sách bệnh nhân:

IF OBJECT\_ID ('Patient\_list', 'V') IS NOT NULL

DROP VIEW Patient\_list

GO

CREATE VIEW Patient\_list AS

SELECT Patient\_id AS N'Mã bệnh nhân', Patient\_name AS N'Họ và tên'

FROM Patient

GO

* Tạo khung nhìn thống kê danh sách bệnh nhân ở Thành phố Hồ Chí Minh

IF OBJECT\_ID ('Patient\_list\_dk2', 'V') IS NOT NULL

DROP VIEW Patient\_list\_dk2

GO

CREATE VIEW Patient\_list\_dk2 AS

SELECT Patient\_id AS N'Mã bệnh nhân', Patient\_name AS N'Họ và tên'

FROM Patient

WHERE Province LIKE N'%TP.Hồ Chí Minh%'

GO

* Tạo thủ tục hiển thị thông tin cá nhân của bệnh nhân bất kỳ

IF OBJECT\_ID ('sp\_Patient\_info', 'P') IS NOT NULL

DROP PROCEDURE sp\_Patient\_info

GO

CREATE PROCEDURE sp\_Patient\_info

@Patient\_id VARCHAR(10)

AS

IF (NOT EXISTS (SELECT \* FROM Patient p WHERE p.Patient\_id = @Patient\_id))

BEGIN

RAISERROR('Not a valid PATIENT ID',11,1)

END

ELSE

BEGIN

SELECT p.Patient\_id AS N'Mã bệnh nhân', p.Patient\_name AS N'Họ và tên', CASE

WHEN p.PSex ='F' THEN N'Nữ' ELSE N'Nam' END AS N'Giới tính', p.PBirthdate AS N'Ngày sinh', YEAR(GETDATE()) - YEAR(p.PBirthdate) AS N'Tuổi', p.AddressNum AS N'Số nhà', p.Ward AS N'Phường', p.Province AS N'Tỉnh', p.PPhone AS N'SĐT', p.PEmail AS 'Email'

FROM Patient p

WHERE p.Patient\_id = @Patient\_id

END

GO

* Tạo thủ tục thêm bệnh nhân mới

IF OBJECT\_ID ('sp\_Insert\_Patient', 'P') IS NOT NULL

DROP PROCEDURE sp\_Insert\_Patient

GO

CREATE PROCEDURE sp\_Insert\_Patient

@Patient\_id VARCHAR(10),

@Patient\_name Nvarchar(50),

@PSex Char(1),

@PBirthdate Date,

@AddressNum varchar(30),

@Ward Nvarchar(30),

@District Nvarchar(30),

@Province Nvarchar(30),

@PPhone Varchar(10),

@PEmail Varchar(50)

AS

IF (EXISTS (SELECT \* FROM Patient p WHERE p.Patient\_id = @Patient\_id))

BEGIN

RAISERROR('This patient\_id already exists',11,1)

END

ELSE

BEGIN

INSERT INTO Patient VALUES (@Patient\_id, @Patient\_name, @PSex, @PBirthdate, @AddressNum, @Ward, @District, @Province, @PPhone, @PEmaiL);

END

GO

* Tạo thủ tục chỉnh sửa tên của bệnh nhân

IF OBJECT\_ID ('sp\_Change\_Patient\_Name', 'P') IS NOT NULL

DROP PROCEDURE sp\_Change\_Patient\_Name

GO

CREATE PROCEDURE sp\_Change\_Patient\_Name

@Patient\_id VARCHAR(10),

@Name NVARCHAR(50)

AS

IF (NOT EXISTS (SELECT \* FROM Patient p WHERE p.Patient\_id = @Patient\_id))

BEGIN

RAISERROR('Not a valid PATIENT ID',11,1)

END

ELSE

BEGIN

UPDATE Patient

SET Patient\_name = @Name

WHERE Patient\_id = @Patient\_id

END

GO

#### Quản lý thông tin bác sĩ

Đối với thông tin bác sĩ cũng sẽ có những câu lệnh truy vấn tương tự như với quản lý bệnh nhân. Sau đây là câu lệnh tạo thủ tục xóa thông tin một bác sĩ với tham số truyền vào là mã bác sĩ.

IF OBJECT\_ID ('sp\_Delete\_Doctor', 'P') IS NOT NULL

DROP PROCEDURE sp\_Delete\_Doctor

GO

CREATE PROCEDURE sp\_Delete\_Doctor

@Doctor\_id VARCHAR(10)

AS

IF (NOT EXISTS (SELECT \* FROM Doctor d WHERE d.Doctor\_id = @Doctor\_id))

BEGIN

RAISERROR('Not a valid DOCTOR ID',11,1)

END

ELSE

BEGIN

DELETE FROM Doctor

WHERE Doctor\_id = @Doctor\_id

END

GO

#### Quản lý bệnh án

Bệnh án là cơ sở dữ liệu lớn nhất chứa tất cả các thông tin của bệnh nhân từ khi bắt đầu khám chữa bệnh tại bệnh viện đến thời điểm hiện tại.

Một bệnh án thường có hai phần:

* Hành chính: Thông tin cá nhân + ngày vào viện
* Bệnh sử: Tình trạng bệnh, các xét nghiệm, kết quả xét nghiệm, chẩn đoán, thuốc đã dùng,…

Sau đây là câu lệnh tạo procedure hiển thị thông tin bệnh án với tham số đầu vào là mã bệnh nhân:

CREATE fUNCTION fun\_1 (@Patient\_id VARCHAR(10))

RETURNS table return

SELECT f.Create\_date AS N'Ngày khám', i.Patient\_id AS N'Mã bệnh nhân',

i.Patient\_name AS N'Họ và tên', CASE WHEN i.PSex ='F' THEN N'Nữ' ELSE N'Nam' END AS N'Giới tính', YEAR(GETDATE()) - YEAR(i.PBirthdate) AS N'Tuổi', i.Province AS N'Tỉnh', i.PPhone AS N'SĐT', f.Clinic\_id AS N'Phòng khám', d.Doctor\_name AS N'Bác sĩ phụ trách', CASE WHEN f.Patient\_type ='1' THEN N'Có BHYT' ELSE N'Không BHYT' END AS N'Đối tượng', f.Diagnose AS N'Chẩn đoán'

FROM Patient i JOIN Patient\_flow f ON i.Patient\_id = f.Patient\_id

JOIN Doctor d ON d.Doctor\_id = f.Doctor\_id

AND i.Patient\_id = @Patient\_id

GO

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

CREATE fUNCTION fun\_2 (@Patient\_id VARCHAR(10))

RETURNS table return

SELECT s.Service\_name AS N'Các xét nghiệm', su.Blood\_pressure AS N'Chỉ số

huyết áp', su.Weight AS N'Cân nặng', su.Height AS N'Chiều cao', d.Doctor\_name AS N'Bác sĩ thực hiện', su.Test\_result AS N'Kết quả'

FROM Service s JOIN Service\_Using su ON s.Service\_id = su.Service\_id

JOIN Patient i ON su.Patient\_id = i.Patient\_id

JOIN Doctor d ON su.Test\_doctor\_id = d.Doctor\_id

AND i.Patient\_id = @Patient\_id

GO

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

CREATE fUNCTION fun\_3 (@Patient\_id VARCHAR(10))

RETURNS table return

SELECT m.Chemical\_name AS N'Thuốc đã dùng', pd.Quanity AS N'Số lượng'

FROM Patient i JOIN Prescription p ON i.Patient\_id = p.Pres\_patient\_id

JOIN Pres\_Detail pd ON pd.Prescription\_id = P.Prescription\_id

JOIN Medication m ON m.Medication\_id = pd.Medication\_id

AND i.Patient\_id = @Patient\_id

GO

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

IF OBJECT\_ID ('sp\_Patient\_Report', 'P') IS NOT NULL

DROP PROCEDURE sp\_Patient\_Report

GO

CREATE PROCEDURE sp\_Patient\_Report

@Patient\_id VARCHAR(10)

AS

IF (NOT EXISTS (SELECT \* FROM Patient p WHERE p.Patient\_id = @Patient\_id))

BEGIN

RAISERROR('Not a valid PATIENT ID',11,1)

END

ELSE

BEGIN

SELECT \* FROM dbo.fun\_1(@Patient\_id)

SELECT \* FROM dbo.fun\_2(@Patient\_id)

SELECT \* FROM dbo.fun\_3(@Patient\_id)

END

GO

EXEC sp\_Patient\_Report '2023000018';

GO

Sau khi chạy câu lệnh, hệ thống trả về một tập các kết quả như sau:



Hình 6‑12. Kết quả trả về cho thủ tục hiển thị bệnh án

#### Quản lý thông tin tài chính

* Tạo hàm thống kê doanh thu thuốc theo tháng với tham số truyền vào là tháng:

CREATE fUNCTION fun\_Medication\_Month (@Month INT)

RETURNS INT

AS

BEGIN

RETURN (

SELECT SUM(m.Med\_price\*p.Quanity)

FROM Pres\_Detail p JOIN Medication m

ON p.Medication\_id = m.Medication\_id

WHERE Month(p.Create\_date) = @Month

GROUP BY p.Create\_date)

END

GO

SELECT dbo.fun\_Medication\_Month (4) AS N'Doanh thu thuốc'

* Tạo hàm thống kê top 5 loại thuốc theo tháng với tham số truyền vào là tháng:

CREATE FUNCTION fun\_Medication\_Top5 (@Month INT)

RETURNS TABLE RETURN

SELECT TOP(5) m.Medication\_id AS N'Mã thuốc', m.Chemical\_name AS N'Tên hoạt chất', SUM(m.Med\_price\*p.Quanity) AS N'Doanh thu'

FROM Pres\_Detail p JOIN Medication m ON p.Medication\_id = m.Medication\_id

WHERE Month(p.Create\_date) = @Month

GROUP BY m.Medication\_id, m.Chemical\_name

Order by SUM(m.Med\_price\*p.Quanity) DESC

GO

SELECT \* from dbo.fun\_Medication\_Top5 (4)

# KẾT LUẬN

Sau quá trình nghiên cứu và thực hiện đồ án “Xây dựng hệ thống quản trị cơ sở dữ liệu quản lý việc khám chữa bệnh tại bệnh viện đa khoa quốc tế Becamex”, em đã học hỏi và tiếp thu thêm nhiều kiến thức mới cũng như các phương pháp để xây dựng một hệ thống cơ sở dữ liệu hiệu quả. Qua đó, em nhận thấy việc có một cơ sở dữ liệu để lưu trữ tất cả các thông tin của bệnh viện là điều cần thiết để giúp bệnh viện có thể dễ dàng quản lý, giám sát các hoạt động hằng ngày. Thuận tiện trong việc thống kê báo cáo về tình hình khám chữa bệnh tại bệnh viện.

Thông qua đồ án này, em đã có cái nhìn rõ hơn về cách thức xây dựng một hệ thống quản trị cơ sở dữ liệu, từ các bước xác định yêu cầu cho tới phân tích và thiết kế một hệ thống hoàn chỉnh.

Sau đây là một số đánh giá chung về đồ án:

* Kết quả đạt được:
* Hệ thống này đã giúp cho bệnh viện thực hiện được một số chức năng quản lý như: quản lý thông tin nhân sự, quản lý thông tin bệnh nhân, quản lý bệnh án, quản lý thuốc, quản lý thông tin tài chính mà bệnh nhân đã đóng cho bệnh viện.
* Giảm thiểu công việc tìm kiếm thủ công, tiết kiệm được thời gian và công sức làm việc của nhân viên và y bác sĩ tại bệnh viện.
* Dễ dàng tính toán, thống kê được số lượng bệnh nhân theo khu vực, theo mùa và theo năm.
* Hạn chế và khuyết điểm của đồ án:
* Về mặt thời gian: Do thời gian có giới hạn và kiến thức còn hạn hẹp nên chưa thể hoàn thiện một hệ thống hoàn chỉnh. Hy vọng sẽ nhận được góp ý từ giảng viên để em có thể cải thiện hơn trong tương lai.
* Về bài nghiên cứu: Hệ thống chỉ tập trung vào quản lý thông tin cá nhân và một số thông tin khác từ việc khám chữa bệnh. Do đó, chưa thể đáp ứng đầy đủ các chức năng thực tế như: quản lý trang thiết bị, quản lý nhân sự,…
* Hướng phát triển trong tương lai:
* Xây dựng thêm một số tính năng còn thiếu, phát triển hệ thống lên mức quản lý toàn bệnh viện.
* Ngoài ra, em sẽ không ngừng cập nhật các kiến thức mới về hệ thống để nâng cao hiểu biết của mình.

TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | J. S. J. F. J. A. Valacich, Essentials of systems analysis and design-Pearson, 2015. |
| [2] | Tilley, Systems analysis and design, 2020. |
| [3] | J. E. K. Kenneth E. Kendall, Systems Analysis and Design Global Ed-Pearson, 2020. |
| [4] | Bệnh viện đa khoa quốc tế Becamex, [Online]. Available: https://bih.vn/ve-chung-toi/. [Accessed 20 April 2023]. |
| [5] | Bệnh viện đa khoa quốc tế Becamex, "BIH," [Online]. Available: https://bih.vn/ve-chung-toi/so-do-to-chuc/. [Accessed 20 April 2023]. |
| [6] | Ths. Trịnh Hoàng Nam, Slide bài giảng môn Hệ quản trị cơ sở dữ liệu. |
| [7] | Trường đại học FPT, SQL-Server 2008 - Hướng dẫn học qua ví dụ, NXB Khoa học và Kỹ thuật, 2013. |