

# NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

## TÀI LIỆU THIẾT KẾ

*Yêu cầu nhóm sinh viên hoàn thành tài liệu thiết kế cho đề án đã được giao theo biểu mẫu đính kèm.*



Bộ môn Công nghệ phần mềm  
Khoa Công nghệ thông tin  
Đại học Khoa học tự nhiên TP HCM

# MỤC LỤC

1	Bảng đánh giá thành viên .....	2
2	Mô hình quan niệm.....	3
3	Thiết kế kiến trúc .....	4
	3.1. Kiến Trúc Client-Server .....	6
	3.2. Kiến Trúc MVC (Model-View-Controller) .....	6
	3.3. Sử Dụng Mẫu Thiết Kế (Design Patterns) .....	6
	3.4. Cơ Chế Plug-in .....	7
4	Thiết kế dữ liệu .....	8
	4.1 Sơ đồ dữ liệu .....	8
	4.2 Đặc tả dữ liệu .....	9

# TÀI LIỆU THIẾT KẾ

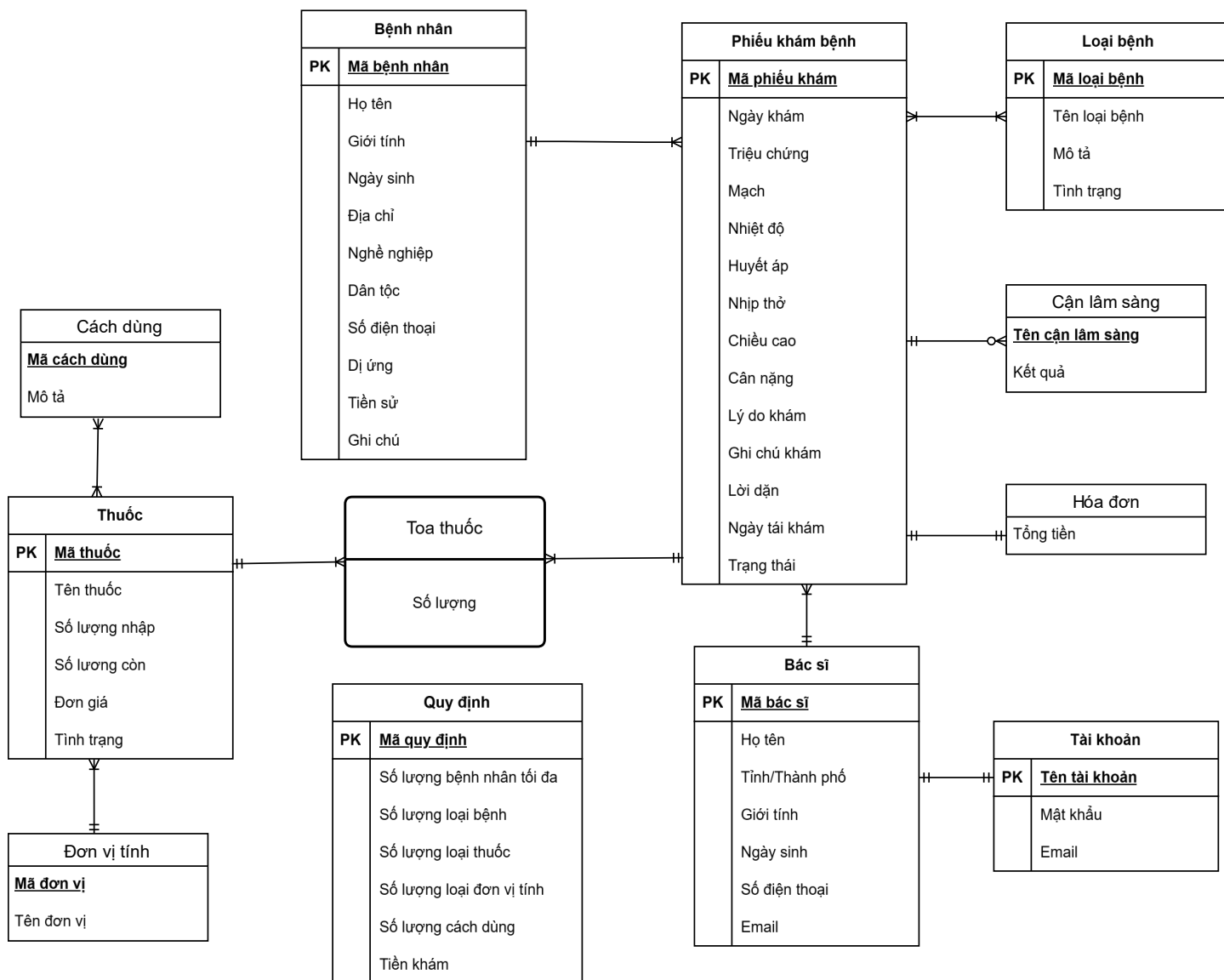
Tài liệu tập trung vào các chủ đề:

- ✓ Tạo ra tài liệu thiết kế phần mềm.
- ✓ Hoàn chỉnh tài liệu thiết kế phần mềm với các nội dung:
  - Mô hình quan niệm
  - Thiết kế kiến trúc
  - Thiết kế dữ liệu
  - Thiết kế giao diện người dùng
- ✓ Đọc hiểu tài liệu thiết kế phần mềm.

# 1 Bảng đánh giá thành viên

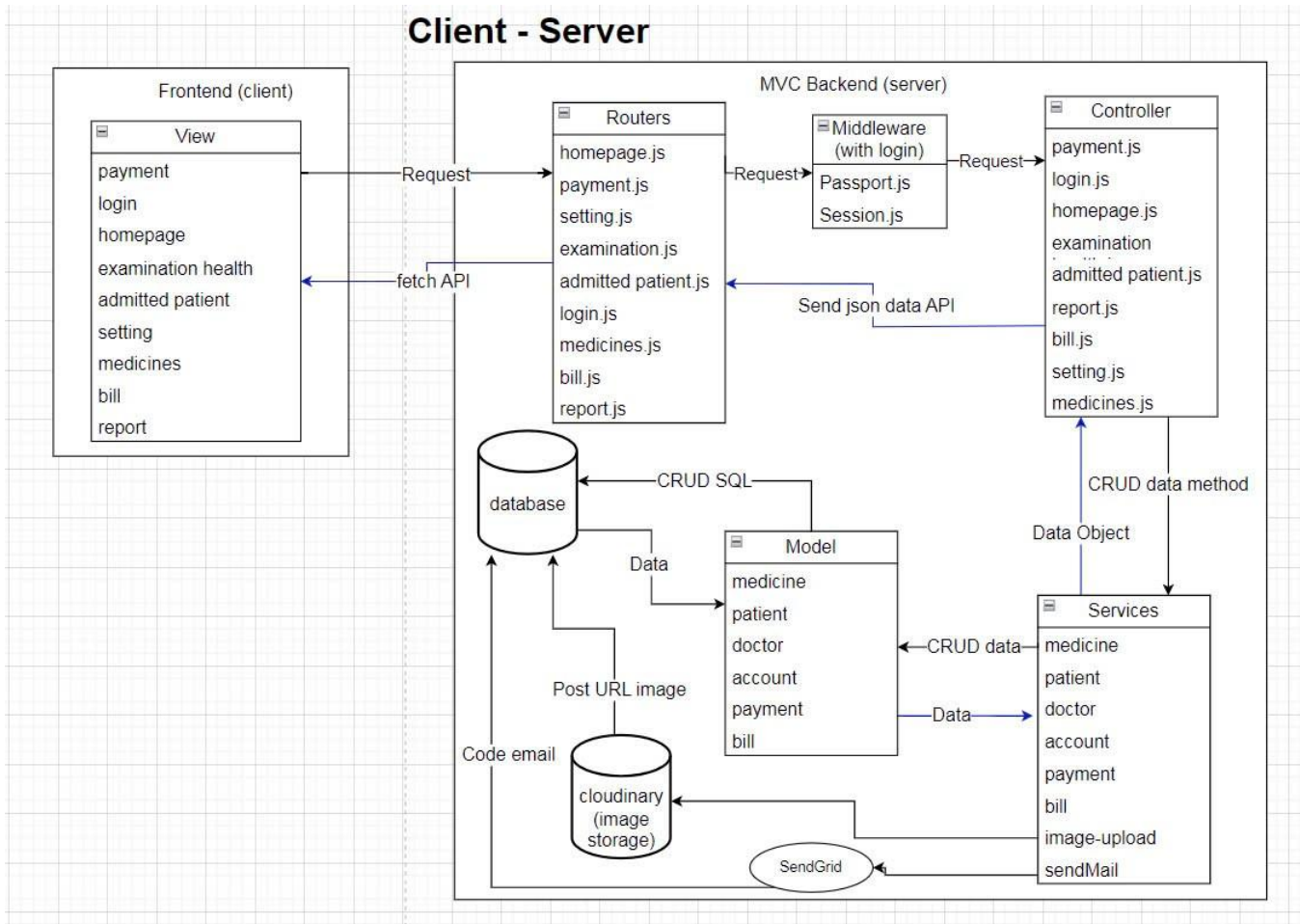
MSSV	Họ Tên	% đóng góp (tối đa 100%)	Chữ ký
22120091	Phạm Khánh Hân	100%	
22120093	Lê Bảo Hồng Hạnh	100%	
22120192	Nguyễn Đăng Long	100%	
22120199	Trần Lượng	100%	
22120200	Hoàng Thanh Mẫn	100%	
22120233	Nguyễn Thị Tú Ngọc	100%	

# 2 Mô hình quan niệm

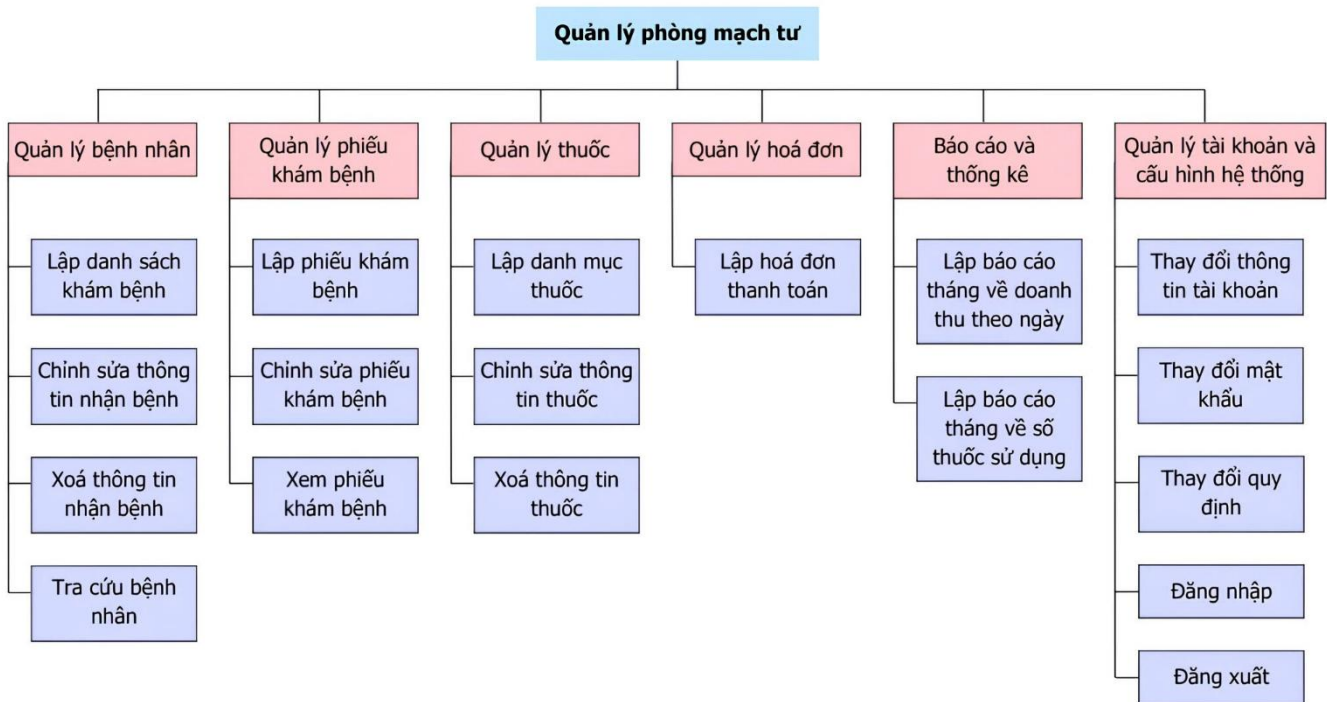


Sơ đồ thể hiện các thực thể ngữ nghĩa trong phần mềm phòng mạch tư

# 3 Thiết kế kiến trúc



Mô hình Client-Server



Sơ đồ bố cục phần mềm

Hệ thống quản lý phòng mạch, kết hợp giữa **mô hình Client-Server** và **kiến trúc MVC** cho backend, sẽ giúp tổ chức hệ thống một cách hiệu quả và dễ bảo trì. Dưới đây là cách thức hoạt động của hệ thống, cùng các thành phần và các công nghệ được sử dụng.

### 3.1. Kiến Trúc Client-Server

- **Frontend (Client):** Giao diện người dùng sẽ sử dụng **Tailwind CSS** để thiết kế và hiển thị thông tin như danh sách bệnh nhân, lịch khám, thông tin bệnh nhân, phiếu khám bệnh, lịch sử khám bệnh, hóa đơn, báo cáo về thuốc và doanh thu, cùng các thao tác chỉnh sửa thông tin phòng khám. Frontend sẽ gửi yêu cầu API (sử dụng **ajax**) đến backend để lấy và gửi dữ liệu và lưu trữ thông tin vào cơ sở dữ liệu.
- **Backend (Server):** Backend sẽ sử dụng **Node.js** với **Express.js**, nơi các API RESTful được xây dựng để xử lý yêu cầu từ frontend, thực hiện logic nghiệp vụ, và trả dữ liệu về cho frontend. Backend cũng sẽ xử lý các tác vụ như xác thực người dùng (bác sĩ) thông qua **Passport** và **Passport-local**, và bảo mật mật khẩu với **bcrypt**. **MySQL** sẽ được dùng để lưu trữ dữ liệu.

### 3.2. Kiến Trúc MVC (Model-View-Controller)

- **Model:** Quản lý dữ liệu của hệ thống, chẳng hạn như thông tin bác sĩ, bệnh nhân, thuốc, hóa đơn, doanh thu, v.v. Các mô hình này được định nghĩa trong Sequelize, ORM của Node.js cho MySQL, để thực hiện các thao tác CRUD (Create, Read, Update, Delete) trên cơ sở dữ liệu.
- **View:** Là cách dữ liệu được trả về cho frontend, dưới dạng gửi **JSON** qua các API. Frontend sẽ chịu trách nhiệm tiếp nhận, xử lý và hiển thị dữ liệu này cho người dùng.
- **Controller:** Xử lý các yêu cầu từ người dùng, tương tác với **Model** để lấy hoặc cập nhật dữ liệu, và trả kết quả về cho **View**. *Ví dụ:* khi người dùng đăng nhập, Controller sẽ liên hệ Services xác thực thông tin người dùng, và nếu hợp lệ, trả về dữ liệu người dùng cho frontend.

### 3.3. Sử Dụng Mẫu Thiết Kế (Design Patterns)

- **Singleton:** Mẫu Singleton sẽ được sử dụng để quản lý các đối tượng có thể có một thể hiện duy nhất trong suốt vòng đời của ứng dụng, ví dụ như kết nối đến cơ sở dữ liệu. Điều này giúp tối ưu tài nguyên và tránh việc tạo kết nối mới mỗi khi có yêu cầu (vì hosting miễn phí nên bị hạn chế khi tạo quá nhiều kết nối).
- **Module Pattern:** Tổ chức mã nguồn theo các module độc lập, dễ bảo trì. Trong Node.js, sử dụng `module.exports` để chia mã thành các phần.



- **MVC Pattern:** Tách biệt các thành phần trong ứng dụng theo mô hình ba lớp: Model (dữ liệu), View (giao diện người dùng), và Controller (xử lý logic nghiệp vụ).

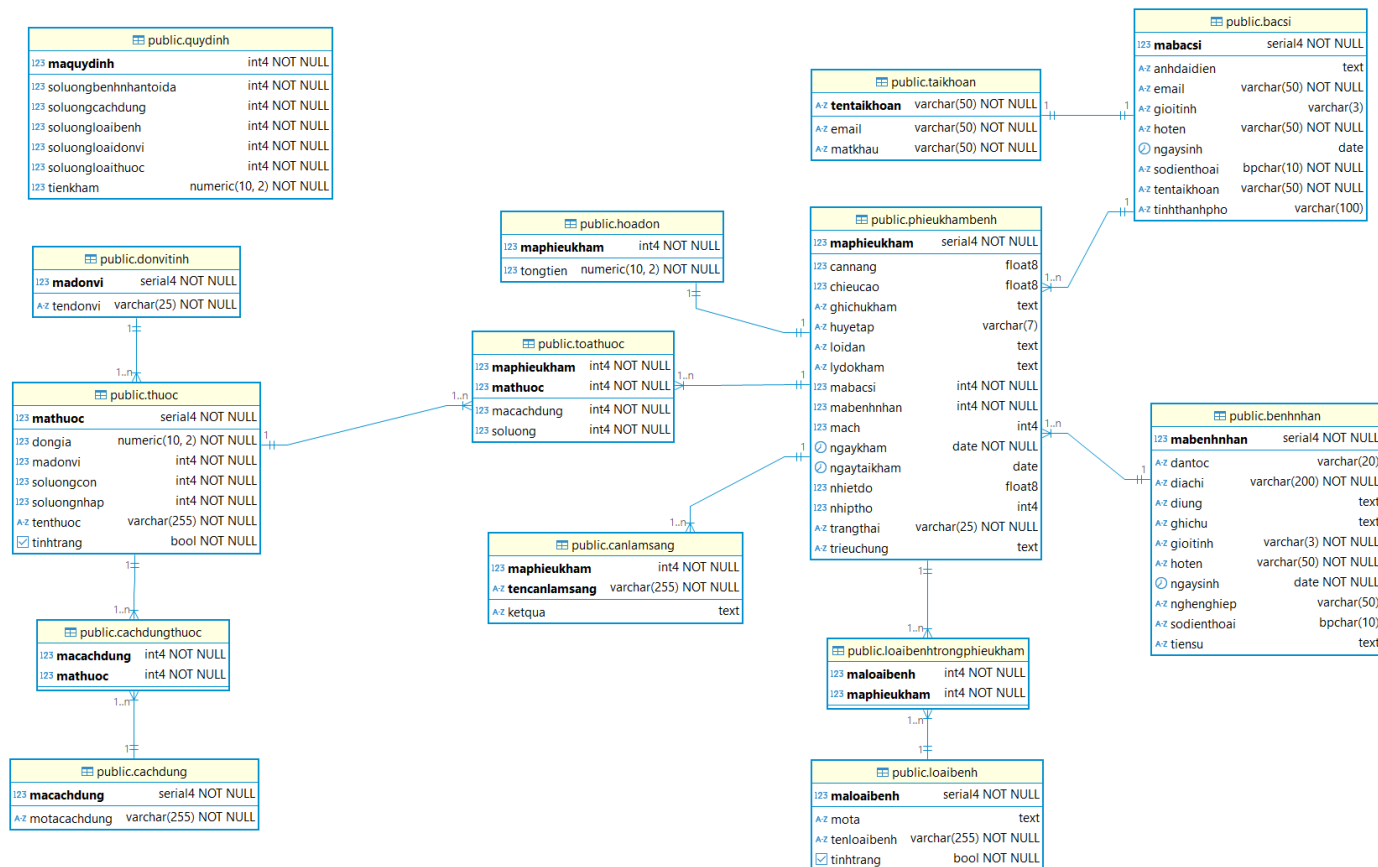
### 3.4. Cơ Chế Plug-in

Cơ chế plug-in cho phép mở rộng hệ thống một cách dễ dàng mà không thay đổi cấu trúc chính:

- *Passport:* Xử lý xác thực bác sĩ với Passport để quản lý việc đăng nhập
- *Sequelize:* Là ORM (Object Relational Mapping) cho MySQL, giúp dễ dàng thao tác với cơ sở dữ liệu mà không cần viết SQL thủ công.
- *Chart.js:* Dùng để vẽ biểu đồ trực quan, ví dụ như thống kê doanh thu, số lượng bệnh nhân, các loại thuốc bán chạy. Các báo cáo hoặc thống kê có thể được hiển thị trên frontend dưới dạng biểu đồ.
- *Tailwind CSS:* Framework CSS giúp bạn xây dựng giao diện người dùng nhanh chóng và dễ dàng.
- *Session:* Dùng để lưu thông tin trạng thái của người dùng (chẳng hạn như khi người dùng đăng nhập, thông tin sẽ được lưu trữ trong session để theo dõi người dùng trong suốt phiên làm việc).
- *Bcrypt:* Dùng để mã hóa mật khẩu, đảm bảo an toàn cho việc lưu trữ và xử lý mật khẩu của bác sĩ
- *Handlebars:* Một thư viện template engine giúp dễ dàng render các dữ liệu động lên giao diện frontend, ví dụ như danh sách bệnh nhân, lịch khám, hóa đơn,...
- *Cloudinary:* Dùng để lưu trữ và quản lý hình ảnh hoặc tệp tải lên (ví dụ như hình ảnh bác sĩ,...) đồng thời có tính năng tối ưu hóa hình ảnh với khả năng tự động thay đổi kích thước.
- *SendGrid:* Dùng để gửi email từ hệ thống, ví dụ như gửi email xác nhận quên mật khẩu.

# 4 Thiết kế dữ liệu

## 4.1 Sơ đồ dữ liệu



Sơ đồ dữ liệu hệ thống

## 4.2 Đặc tả dữ liệu

❖ TaiKhoan (TenTaiKhoan, MatKhau, Email)

Thuộc tính	Mô tả
TenTaiKhoan	Là chuỗi ký tự để định danh một tài khoản duy nhất
	Kiểu dữ liệu: <i>Varchar(50)</i>
	Độ dài tối đa: 50
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>
MatKhau	Là mật khẩu của một tài khoản
	Kiểu dữ liệu: <i>Varchar(50)</i>
	Độ dài tối đa: 50
	Ràng buộc: - <i>NOT NULL</i>
Email	Là tài khoản email của tài khoản
	Kiểu dữ liệu: <i>Varchar(50)</i>
	Độ dài tối đa: 50
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>

❖ DonViTinh (MaDonVi, TenDonVi)

Thuộc tính	Mô tả
MaDonVi	Là mã số định danh một đơn vị tính duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>
TenDonVi	Mô tả loại đơn vị cho các loại thuốc
	Kiểu dữ liệu: <i>Varchar(25)</i>
	Độ dài tối đa: 25
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>

- ❖ BacSi (**MaBacSi**, HoTen, Email, TinhThanhPho, GioiTinh, NgaySinh, SoDienThoai, TenTaiKhoan, AnhDaiDien)

Thuộc tính	Mô tả
MaBacSi	Là mã số để định danh một bác sĩ duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc:
	- <i>NOT NULL</i> - <i>UNIQUE</i>
HoTen	Là họ và tên đầy đủ của bác sĩ
	Kiểu dữ liệu: <i>Varchar(50)</i>
	Độ dài tối đa: 50
	Ràng buộc: - <i>NOT NULL</i>
Email	Là tài khoản email của bác sĩ
	Kiểu dữ liệu: <i>Varchar(50)</i>
	Độ dài tối đa: 50
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>
TinhThanhPho	Là tên thành phố bác sĩ đang sinh sống
	Kiểu dữ liệu: <i>Varchar(100)</i>
	Độ dài tối đa: 100
GioiTinh	Là giới tính của bác sĩ
	Kiểu dữ liệu: <i>Varchar(3)</i>
	Độ dài tối đa: 3
	Ràng buộc: - Giá trị: <i>Nam</i> hoặc <i>Nữ</i>
NgaySinh	Là ngày tháng năm sinh của bác sĩ
	Kiểu dữ liệu: <i>Date</i>
SoDienThoai	Là số điện thoại của bác sĩ
	Kiểu dữ liệu: <i>Char(10)</i>
	Độ dài: 10
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>
TenTaiKhoan	Là tài khoản của bác sĩ trong hệ thống

	Kiểu dữ liệu: <i>Varchar(50)</i>
	Độ dài tối đa: <i>50</i>
	Ràng buộc: - <i>FOREIGN KEY</i> - <i>NOT NULL</i>
AnhDaiDien	Là đường dẫn ảnh đại diện của bác sĩ
	Kiểu dữ liệu: <i>Text</i>

❖ CachDung (**MaCachDung**, MoTaCachDung)

Thuộc tính	Mô tả
MaCachDung	Là mã để định danh một cách dùng duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>
MoTaCachDung	Là nội dung chi tiết của một cách dùng
	Kiểu dữ liệu: <i>Varchar(255)</i>
	Ràng buộc: - <i>NOT NULL</i>

❖ Thuoc (**MaThuoc**, TenThuoc, SoLuongNhap, SoLuongCon, DonGia, MaDonVi)

Thuộc tính	Mô tả
MaThuoc	Là mã số để định danh một loại thuốc duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>
TenThuoc	Là tên của một loại thuốc
	Kiểu dữ liệu: <i>Varchar(50)</i>
	Độ dài tối đa: <i>50</i>
	Ràng buộc: - <i>NOT NULL</i>
SoLuongNhap	Là số lượng nhập vào của một loại thuốc
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc:

	<ul style="list-style-type: none"> <li>- Giá trị: <i>lớn hơn 0</i></li> <li>- <i>NOT NULL</i></li> </ul>
SoLuongCon	Là số lượng tồn của một loại thuốc Kiểu dữ liệu: <i>Int</i> Ràng buộc: <ul style="list-style-type: none"> <li>- Giá trị: <i>lớn hơn 0</i></li> <li>- <i>NOT NULL</i></li> </ul>
DonGia	Là giá hiện tại của một loại thuốc Kiểu dữ liệu: <i>Numeric(10,2)</i> Ràng buộc: <ul style="list-style-type: none"> <li>- Giá trị: <i>lớn hơn 0</i></li> <li>- <i>NOT NULL</i></li> </ul>
MaDonVi	Là mã đơn vị tính của một loại thuốc Kiểu dữ liệu: <i>Int</i> Ràng buộc: <ul style="list-style-type: none"> <li>- <i>FOREIGN KEY</i></li> <li>- <i>NOT NULL</i></li> </ul>

❖ **CachDungThuoc** (**MaThuoc**, **MaCachDung**)

Thuộc tính	Mô tả
MaThuoc	Là mã số để định danh một loại thuốc duy nhất Kiểu dữ liệu: <i>Int</i> Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- <i>FOREIGN KEY</i></li> </ul>
MaCachDung	Là mã số để định danh một cách dùng duy nhất Kiểu dữ liệu: <i>Int</i> Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- <i>FOREIGN KEY</i></li> </ul>

❖ **QuyDinh** (**MaQuyDinh**, SoLuongBenhNhanToiDa, SoLuongLoaiBenh, SoLuongLoaiThuoc, SoLuongLoaiDonVi, SoLuongCachDung, TienKham, TinhTrang)

Thuộc tính	Mô tả
------------	-------

MaQuyDinh	Là mã số để xác định một lần thay đổi quy định
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>
SoLuongBenhNhanToiDa	Là số lượng bệnh nhân tối đa được khám trong một ngày
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - Giá trị: $> 0$ - Mặc định: 40
SoLuongLoaiBenh	Là số lượng loại bệnh hỗ trợ trong phòng khám
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - Giá trị: $> 0$ - Mặc định: 5
SoLuongLoaiThuoc	Là số lượng loại thuốc có trong phòng khám
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - Giá trị: $> 0$ - Mặc định: 30
SoLuongLoaiDonVi	Là số lượng đơn vị có trong phòng khám
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - Giá trị: $> 0$ - Mặc định: 2
SoLuongCachDung	Là số lượng cách dùng của các loại thuốc có trong phòng khám
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - Giá trị: $> 0$

	- Mặc định: 4
TienKham	Tiền khám cho mỗi lần khám ở phòng khám
	Kiểu dữ liệu: <i>Numeric(10,2)</i>
	Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- Giá trị: <i>&gt; 0</i></li> <li>- Mặc định: <i>30000</i></li> </ul>
TinhTrang	Là tình trạng của thuốc (để không bị lỗi khóa ngoại khi xóa)
	Kiểu dữ liệu: <i>Bool</i>
	Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- Giá trị: <i>1: đang hoạt động, 0: bị khóa</i></li> </ul>

❖ ToaThuoc (MaPhieuKham, MaThuoc, MaCachDung, SoLuong)

Thuộc tính	Mô tả
MaPhieuKham	Là mã số để định danh một phiếu khám bệnh duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- <i>FOREIGN KEY</i></li> </ul>
MaThuoc	Là mã số để định danh một loại thuốc duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- <i>FOREIGN KEY</i></li> </ul>
SoLuong	Là số lượng của từng loại thuốc trong toa thuốc
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> </ul>
MaCachDung	Là mã số để xác định cách dùng trong toa thuốc
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc:



	<ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- <i>FOREIGN KEY</i></li> </ul>
--	---

❖ BệnhNhan (**MaBệnhNhan**, HoTen, GioiTinh, DiaChi, NgaySinh, NgheNghiep, DanToc, SoDienThoai, TienSu, DiUng, GhiChu)

Thuộc tính	Mô tả
MaBệnhNhan	Là mã số để định danh một bệnh nhân duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc:
	<ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- <i>UNIQUE</i></li> </ul>
HoTen	Là họ và tên đầy đủ của bệnh nhân
	Kiểu dữ liệu: <i>Varchar(50)</i>
	Độ dài tối đa: 50
	Ràng buộc:
GioiTinh	<ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> </ul>
	Là giới tính của bệnh nhân
	Kiểu dữ liệu: <i>Varchar(3)</i>
	Độ dài tối đa: 3
DiaChi	Ràng buộc:
	<ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- Giá trị: <i>Nam</i> hoặc <i>Nữ</i></li> </ul>
	Là địa chỉ nơi ở của bệnh nhân
	Kiểu dữ liệu: <i>Varchar(200)</i>
NgaySinh	Độ dài tối đa: 200
	Ràng buộc:
	<ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> </ul>
	Là ngày tháng năm sinh của bệnh nhân
NgheNghiep	Kiểu dữ liệu: <i>Date</i>
	Ràng buộc:
	<ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> </ul>
	Là nghề nghiệp của bệnh nhân
DanToc	Kiểu dữ liệu: <i>Varchar(50)</i>
	Độ dài tối đa: 50
	Là tên dân tộc của bệnh nhân
	Kiểu dữ liệu: <i>Varchar(20)</i>

	Độ dài tối đa: 20
SoDienThoai	Là số điện thoại của bệnh nhân
	Kiểu dữ liệu: <i>Char(10)</i>
	Độ dài: 10
TienSu	Là tiền sử bệnh của bệnh nhân
	Kiểu dữ liệu: <i>Text</i>
DiUng	Lưu thông tin xem bệnh nhân có bị dị ứng với thứ gì không.
	Kiểu dữ liệu: <i>Text</i>
GhiChu	Ghi chú thêm về bệnh nhân
	Kiểu dữ liệu: <i>Text</i>

❖ HoaDon (**MaPhieuKham**, TongTien)

Thuộc tính	Mô tả
MaPhieuKham	Là mã số để định danh một phiếu khám bệnh duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - <i>FOREIGN KEY</i>
TongTien	Là tổng tiền của hóa đơn
	Kiểu dữ liệu: <i>Numeric(10,2)</i>
	Ràng buộc: - <i>NOT NULL</i>

❖ LoaiBenh (**MaLoaiBenh**, TenLoaiBenh, MoTa, TinhTrang)

Thuộc tính	Mô tả
MaLoaiBenh	Là mã số để định danh một loại bệnh duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: - <i>NOT NULL</i> - <i>UNIQUE</i>
TenLoaiBenh	Là tên của loại bệnh
	Kiểu dữ liệu: <i>Varchar(255)</i>
	Độ dài tối đa: 255
	Ràng buộc:

	- <i>NOT NULL</i>
MoTa	Là mô tả về loại bệnh
	Kiểu dữ liệu: <i>Text</i>
TinhTrang	Là tình trạng của loại bệnh (để không bị lỗi khóa ngoại khi xóa)
	Kiểu dữ liệu: <i>Bool</i>
	Ràng buộc:
	- <i>NOT NULL</i>
	- Giá trị: 1: đang hoạt động, 0: bị khóa

❖ LoaiBenhTrongPhieuKham (**MaPhieuKham**, **MaLoaiBenh**)

Thuộc tính	Mô tả
MaPhieuKham	Là mã số để định danh một phiếu khám bệnh duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc:
	- <i>NOT NULL</i>
	- <i>FOREIGN KEY</i>
MaLoaiBenh	Là mã số để định danh một loại bệnh duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc:
	- <i>NOT NULL</i>
	- <i>FOREIGN KEY</i>

❖ CanLamSang (**MaPhieuKham**, **TenCanLamSang**, KetQua)

Thuộc tính	Mô tả
MaPhieuKham	Là mã số để định danh một phiếu khám bệnh duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc:
	- <i>NOT NULL</i>
	- <i>FOREIGN KEY</i>
TenCanLamSang	Là tên của cận lâm sàng duy nhất
	Kiểu dữ liệu: <i>Varchar(255)</i>
	Độ dài tối đa: 255
	Ràng buộc:

	- <i>NOT NULL</i>
KetQua	Là kết quả của cận lâm sàng
	Kiểu dữ liệu: <i>Text</i>

- ❖ PhieuKhamBenh (**MaPhieuKham**, NgayKham, TrieuChung, Mach, NhietDo, HuyetAp, NhipTho, ChieuCao, CanNang, LyDoKham, GhiChuKham, LoiDan, NgayTaiKham, TrangThai, MaBenhNhan, MaBacSi)

Thuộc tính	Mô tả
MaPhieuKham	Là mã số để định danh một phiếu khám bệnh duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc:
	- <i>NOT NULL</i>
	- <i>UNIQUE</i>
NgayKham	Là ngày bệnh nhân khám bệnh
	Kiểu dữ liệu: <i>Date</i>
	Ràng buộc:
	- <i>NOT NULL</i>
TrieuChung	Là triệu chứng bệnh nhân mắc phải
	Kiểu dữ liệu: <i>Text</i>
Mach	Là chỉ số mạch của bệnh nhân
	Kiểu dữ liệu: <i>Int</i>
NhietDo	Là nhiệt độ của bệnh nhân
	Kiểu dữ liệu: <i>Float</i>
HuyetAp	Là chỉ số huyết áp của bệnh nhân
	Kiểu dữ liệu: <i>Varchar(7)</i>
NhipTho	Là chỉ số nhịp thở của bệnh nhân
	Kiểu dữ liệu: <i>Int</i>
ChieuCao	Là chiều cao của bệnh nhân
	Kiểu dữ liệu: <i>Float</i>
CanNang	Là cân nặng của bệnh nhân
	Kiểu dữ liệu: <i>Float</i>
LyDoKham	Lý do khám bệnh của bệnh nhân
	Kiểu dữ liệu: <i>Text</i>
LoiDan	Lời dặn của bác sĩ
	Kiểu dữ liệu: <i>Text</i>

NgàyTaiKham	Ngày tái khám đã được thỏa thuận giữa bệnh nhân và bác sĩ
	Kiểu dữ liệu: <i>Date</i>
TrangThai	Trạng thái khám bệnh của bệnh nhân khi được nhận bệnh
	Kiểu dữ liệu: <i>Varchar(25)</i>
	Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- Giá trị: <i>Chưa khám, Đang khám, Đã khám</i></li> </ul>
MaBenhNhan	Là mã số để định danh một bệnh nhân duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- <i>FOREIGN KEY</i></li> </ul>
MaBacSi	Là mã số để định danh một bác sĩ duy nhất
	Kiểu dữ liệu: <i>Int</i>
	Ràng buộc: <ul style="list-style-type: none"> <li>- <i>NOT NULL</i></li> <li>- <i>FOREIGN KEY</i></li> </ul>