

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

XÂY DỰNG WEBSITE ĐẶT LỊCH KHÁM CHO
PHÒNG KHÁM NHA KHOA

Người hướng dẫn: **Th.S NGUYỄN THỊ VŨ THẢO**

Sinh viên thực hiện: **HUỲNH QUỐC DŨNG**

Mã số sinh viên: **2051220062**

Lớp: **20CT2**

Đà Nẵng, 12 /2025

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

XÂY DỰNG WEBSITE ĐẶT LỊCH KHÁM CHO PHÒNG KHÁM NHA KHOA

Người hướng dẫn: **Th.S NGUYỄN THỊ VŨ THẢO**

Sinh viên thực hiện: **HUỲNH QUỐC DŨNG**

Mã số sinh viên: **2051220062**

Lớp: **20CT2**

Đà Nẵng, 12 /2025

TÓM TẮT

Tên đề tài: Xây dựng website đặt lịch khám cho phòng khám nha khoa

Sinh viên thực hiện: Huỳnh Quốc Dũng

Mã số SV: 2051220062

Lớp: 20CT2

Xây dựng website đặt lịch khám cho phòng khám nha khoa viết bằng Lavarel và Visual Studio Code. Xây dựng hệ thống với 4 tác nhân chính là admin, bác sĩ, trợ lý bác sĩ và bệnh nhân. Phần mềm có các chức năng cơ bản như: xem thông tin lịch khám, xem thông tin bệnh nhân, , hủy hoặc thay đổi lịch hẹn, đặt lịch khám.

LỜI CẢM ƠN

Trước hết, em xin bày tỏ lòng biết ơn sâu sắc đến tất cả các Thầy Cô giáo trong bộ môn, những người đã luôn tận tâm truyền đạt tri thức, khơi dậy cảm hứng học tập và nghiên cứu trong chúng em. Sự nhiệt huyết, tâm huyết và những bài giảng bổ ích của Thầy Cô đã trang bị cho chúng em không chỉ là kiến thức mà còn là tinh thần trách nhiệm và khát vọng chinh phục những đỉnh cao tri thức mới.

Đặc biệt, em xin dành lời cảm ơn chân thành nhất đến cô ThS. Nguyễn Thị Vũ Thảo, người đã tận tình hướng dẫn và đồng hành cùng em trong suốt quá trình thực hiện đề tài này. Sự quan tâm, động viên của cô đã giúp em vượt qua những khó khăn và thử thách trong học tập và nghiên cứu. Em vô cùng biết ơn cô vì đã dành nhiều thời gian quý báu để hướng dẫn, sửa chữa, và cung cấp những lời khuyên sâu sắc, giúp em hoàn thiện đề tài một cách tốt nhất.

Những đóng góp của cô không chỉ đơn thuần là những kiến thức học thuật mà còn là những bài học quý báu về sự kiên nhẫn, sáng tạo và tinh thần không ngừng học hỏi. Sự tận tụy và lòng nhiệt thành của cô đã để lại trong em niềm cảm kích lớn lao, là động lực để em cố gắng hơn nữa trên con đường học tập và sự nghiệp sau này.

Em xin chân thành cảm ơn và kính chúc Quý Thầy Cô dồi dào sức khỏe, hạnh phúc và tiếp tục thành công trong sự nghiệp trồng người cao quý!

Xin chân thành cảm ơn!

CAM ĐOAN

Tôi xin cam đoan rằng đồ án tốt nghiệp này là công trình nghiên cứu của cá nhân tôi, được thực hiện một cách trung thực, nghiêm túc và tuân thủ đúng các quy định về liêm chính học thuật. Toàn bộ nội dung, dữ liệu, kết quả, và tài liệu tham khảo được trình bày trong đồ án này đều được tôi tìm hiểu, nghiên cứu và tổng hợp một cách khách quan, minh bạch.

Tôi xin khẳng định rằng không có bất kỳ hành vi sao chép, vi phạm bản quyền, hay gian lận học thuật nào được thực hiện trong quá trình hoàn thành đồ án này. Nếu có bất kỳ sự sai sót hay vi phạm nào, tôi xin hoàn toàn chịu trách nhiệm trước quy định của nhà trường và pháp luật.

Tôi cũng xin gửi lời cảm ơn chân thành đến các Thầy Cô và tất cả những người đã hỗ trợ, góp ý và tạo điều kiện cho tôi trong suốt quá trình thực hiện đồ án tốt nghiệp này. Xin trân trọng cảm ơn!

Huỳnh Quốc Dũng

MỤC LỤC

TÓM TẮT	v
LỜI CẢM ƠN	ii
CAM ĐOAN	ii
MỤC LỤC	iii
DANH SÁCH CÁC BẢNG	vi
DANH MỤC HÌNH ẢNH	viii
DANH MỤC CÁC TỪ VIẾT TẮT	x
MỞ ĐẦU	1
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT	3
1.1 Hệ quản trị cơ sở Sql	3
1.1.1 Điểm mạnh và điểm yếu	3
1.1.2 Cấu trúc hệ quản trị cơ sở dữ liệu	4
1.1.3 Tại sao phải cần hệ quản trị cho đề tài	5
1.2 Visual Studio Code	5
1.2.1 Visual Studio Code là gì	5
1.2.2 Những ưu điểm nổi bật của Visual Studio Code	6
1.2.3 Tại sao nên sử dụng Visual Studio Code	6
1.3 Ngôn ngữ lập trình	7
1.3.1 Ngôn ngữ HTML	7
1.3.2 Ngôn ngữ CSS	8
1.3.3 Ngôn ngữ PHP	9

1.4. Laravel framework.....	10
CHƯƠNG 2: KHẢO SÁT THIẾT KẾ HỆ THỐNG	12
2.1 Giới thiệu chung về bài toán.....	12
2.2 Phân tích yêu cầu.....	12
2.1.1 Bác sĩ.....	12
2.1.2 Trợ lý bác sĩ.....	13
2.1.2 Bệnh nhân.....	13
2.2 Các biểu đồ Use Case	14
2.2.1 Use Case Admin.....	14
2.2.2 Use Case bác sĩ	18
2.2.3 Use Case trợ lý bác sĩ.....	21
2.2.4 Use Case bệnh nhân	24
2.3 Biểu đồ tuần tự.....	27
2.3.1 Biểu đồ tuần tự chức năng “ Đăng nhập ”	27
2.3.2 Biểu đồ tuần tự chức năng “ Đăng ký”	28
2.3.2 Biểu đồ tuần tự chức năng “ Chỉnh sửa thông tin cá nhân”	29
2.3.3 Biểu đồ tuần tự chức năng “ Xem thông tin dịch vụ khám ”	30
2.3.4 Biểu đồ tuần tự chức năng “ Tìm kiếm bác sĩ”	31
2.3.5 Biểu đồ tuần tự chức năng “ Đặt lịch khám”	32
2.3.6 Biểu đồ tuần tự chức năng “Quản lý thông tin bệnh nhân”	33
2.3.7 Quản lý lịch làm việc bác sĩ.....	34
2.3.9 Quản lý banner	35
2.3.8 Quản lý liên hệ.....	36
2.3.10 Quản lý bài viết	37

2.4 Cơ sở dữ liệu	38
2.4.1 Các thực thể có trong hệ thống.....	38
Mô tả về giá trị của các trường	38
2.4.2 Chi tiết cho từng thực thể.....	39
2.4.3 Sơ đồ ERD tổng thể	44
CHƯƠNG 3 . KẾT QUẢ THỰC NGHIỆM ĐỀ TÀI.....	45
3. 1 Môi trường	45
3. 2 Giao diện kết quả.....	45
3.2.1 Giao diện trang chủ	45
3.2.2 Giao diện đăng nhập	46
3.2.3 Giao diện đăng ký	46
3.2.4 Giao diện các dịch vụ phòng khám	47
3.2.5 Giao diện trang tin tức	48
3.2.6 Giao diện trang liên hệ	48
3.2.7 Giao diện chatbox	49
3.2.8 Giao diện đặt lịch khám	49
3.2.9 Giao diện quản lý phòng khám(Admin).....	50
3.2.10 Giao diện trang quản lý bệnh nhân	50
3.2.11 Giao diện trang quản lý bác sĩ.....	51
3.2.12 Giao diện trang quản lý lịch làm việc	51
3.2.13 Giao diện trang quản lý đặt lịch khám.....	52
3.2.14 Giao diện trang quản lý tin tức	52
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	53
TÀI LIỆU THAM KHẢO.....	54

DANH SÁCH CÁC BẢNG

Bảng 2.1 Đặt tả Use Case quản lý người dùng.....	14
Bảng 2.2 Đặt tả Use Case quản lý bác sĩ.....	15
Bảng 2.3 Đặt tả Use Case quản lý lịch hẹn	15
Bảng 2.4 Đặt tả Use Case quản lý dịch vụ	16
Bảng 2.5 Đặt tả Use Case quản lý báo cáo thống kê.....	17
Bảng 2.6 Đặt tả Use Case xem lịch làm việc	18
Bảng 2.7 Đặt tả Use Case xem hồ sơ bệnh nhân.....	19
Bảng 2.8 Đặt tả Use Case xác nhận lịch hẹn.....	19
Bảng 2.9 Đặt tả Use Case cập nhật tình trạng khám	20
Bảng 2.10 Đặt tả Use Case ghi chú, chuẩn đoán, lưu kết quả khám.....	20
Bảng 2.11 Đặt tả Use Case nhận thông báo khi lịch thay đổi	21
Bảng 2.12 Đặt tả Use Case quản lý thông tin bác sĩ	22
Bảng 2.13 Đặt tả Use Case liên hệ và nhắc lịch cho bệnh nhân	22
Bảng 2.14 Đặt tả Use Case duyệt hoặc hủy lịch hẹn.....	23
Bảng 2.15 Đặt tả Use Case đăng ký tài khoản	24
Bảng 2.16 Đặt tả Use Case đặt lịch khám	25
Bảng 2.17 Đặt tả Use Case xem lịch sử khám	25
Bảng 2.18 Đặt tả Use Case xem/sửa thông tin cá nhân.....	26
Bảng 2.19 Bảng thực thể có trong hệ thống	38
Bảng 2.20 Thực thể articles.....	39
Bảng 2.21 Thực thể bookings.....	40
Bảng 2.22 Thực thể bot_knowledge.....	40

Bảng 2.23 Thực thể chat_messages.....	41
Bảng 2.24 Thực thể clinics	42
Bảng 2.25 Thực thể categories	43

DANH MỤC HÌNH ẢNH

Hình 2.1 Biểu đồ tuần tự chức năng đăng nhập	27
Hình 2.2 Biểu đồ tuần tự chức năng đăng ký	28
Hình 2.3 Biểu đồ tuần tự chức chỉnh sửa thông tin cá nhân	29
Hình 2. 4 Biểu đồ tuần tự chức xem thông tin dịch vụ khám.....	30
Hình 2. 5 Biểu đồ tuần tự chức năng tìm kiếm bác sĩ	31
Hình 2. 6 Biểu đồ tuần tự chức năng đặt lịch khám	32
Hình 2.7 Biểu đồ tuần tự chức năng quản lý thông tin bệnh nhân	33
Hình 2. 8 Biểu đồ tuần tự chức năng quản lý việc làm việc bác sĩ	34
Hình 2. 9 Biểu đồ tuần tự chức năng quản lý banner	35
Hình 2. 10 Biểu đồ tuần tự chức năng quản lý liên hệ	36
Hình 2. 11 Biểu đồ tuần tự chức năng quản lý bài viết	37
Hình 2. 12 Sơ đồ ERD tổng thể.....	44
Hình 3.1 Giao diện trang chủ	Error! Bookmark not defined.
Hình 3.2 Giao diện đăng nhập.....	46
Hình 3.3 Giao diện đăng ký.....	46
Hình 3.4 Giao diện dịch vụ.....	47
Hình 3.5 Giao diện trang tin tức	48
Hình 3.6 Giao diện liên hệ.....	Error! Bookmark not defined.
Hình 3.7 Giao diện chatbox.....	49
Hình 3.8 Giao diện đặt lịch khám.....	49
Hình 3.9 Giao diện trang chủ(Admin).....	50
Hình 3.10 Giao diện trang quản lý bệnh nhân.....	50
Hình 3.11 Giao diện trang quản lý bác sĩ	51
Hình 3.12 Giao diện trang quản lý làm việc.....	51

Hình 3.13 Giao diện trang quản lý đặt lịch khám	52
Hình 3.14 Giao diện trang quản lý tin tức	52

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Diễn giải
API	Application Programming Interface
CRUD	Create - Read - Update - Delete
CSS	Cascading Style Sheets
DB	Database
ERD	Entity Relationship Diagram
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
ID	Identifier
JS	JavaScript
JSON	JavaScript Object Notation
MVC	Model - View - Controller
MySQL	My Structured Query Language (Hệ quản trị CSDL)
PHP	Hypertext Preprocessor (ngôn ngữ lập trình)
SQL	Structured Query Language
UI	User Interface
URL	Uniform Resource Locator
UX	User Experience
VS Code	Visual Studio Code
WWW	World Wide Web

MỞ ĐẦU

1. Lý do chọn đề tài

Trong bối cảnh chuyển đổi số diễn ra mạnh mẽ, nhu cầu sử dụng các dịch vụ y tế trực tuyến ngày càng tăng. Tuy nhiên, nhiều phòng khám nha khoa vẫn quản lý lịch hẹn theo cách truyền thống như ghi chép sổ sách hoặc liên hệ qua điện thoại, dẫn đến các vấn đề như trùng lịch, khó kiểm soát số lượng bệnh nhân, mất nhiều thời gian trao đổi và gây bất tiện cho cả bệnh nhân lẫn bác sĩ.

Việc xây dựng website đặt lịch khám cho phòng khám nha khoa giúp tự động hóa quy trình đặt lịch, giảm tải công việc cho nhân viên, tăng độ chính xác khi quản lý lịch hẹn, đồng thời mang lại trải nghiệm tiện lợi cho bệnh nhân khi có thể đặt lịch mọi lúc, mọi nơi. Bên cạnh đó, hệ thống còn hỗ trợ phòng khám quảng bá dịch vụ, nâng cao tính chuyên nghiệp và đáp ứng xu hướng công nghệ trong lĩnh vực chăm sóc sức khỏe.

Chính vì những lý do trên nên em đã chọn đề tài “Xây dựng website đặt lịch khám cho phòng khám nha khoa” nhằm góp phần tối ưu hóa hoạt động quản lý và nâng cao chất lượng phục vụ tại phòng khám.

2. Mục tiêu đề tài

Xây dựng một website đặt lịch khám trực tuyến cho phòng khám nha khoa giúp bệnh nhân, bác sĩ, trợ lý bác sĩ và quản trị viên có thể tương tác, quản lý lịch hẹn và thông tin khám chữa bệnh một cách hiệu quả, thuận tiện và hiện đại, góp phần số hóa quy trình quản lý phòng khám.

3. Đối tượng và phạm vi nghiên cứu

3.1 Đối tượng nghiên cứu

- Bệnh nhân: Đăng ký, đăng nhập và đặt lịch khám trực tuyến.
- Bác sĩ: Xem, quản lý và cập nhật lịch khám bệnh nhân.
- Trợ lý bác sĩ: Xác nhận, sắp xếp và điều phối lịch khám cho bác sĩ.
- Quản trị viên: Quản lý người dùng, dịch vụ và toàn bộ hệ thống.

- Hệ thống website: Gồm giao diện, cơ sở dữ liệu và chức năng xử lý, giúp tự động hóa quy trình đặt lịch và quản lý khám chữa bệnh.

3.2 Phạm vi nghiên cứu

Đề tài tập trung nghiên cứu và xây dựng website đặt lịch khám cho phòng khám nha khoa với các chức năng chính:

- Đăng ký, đăng nhập và đặt lịch khám trực tuyến cho bệnh nhân.
- Quản lý lịch khám, bệnh nhân và dịch vụ cho bác sĩ, trợ lý và quản trị viên.
- Giao diện web thân thiện, hoạt động trên máy tính và điện thoại.
- Phân quyền người dùng rõ ràng: bệnh nhân – bác sĩ – trợ lý – quản trị viên.

4. Phương pháp nghiên cứu

Đề tài sử dụng các phương pháp nghiên cứu sau:

- Phương pháp khảo sát thực tế: Tìm hiểu quy trình đặt lịch và quản lý lịch khám tại phòng khám nha khoa.
- Phương pháp phân tích – thiết kế hệ thống: Xây dựng sơ đồ chức năng, use case, cơ sở dữ liệu và luồng xử lý.
- Phương pháp lập trình và thử nghiệm: Thiết kế giao diện, viết mã nguồn website, kiểm thử và hoàn thiện chức năng.
- Phương pháp tổng hợp – đánh giá: Thu thập phản hồi, đánh giá tính hiệu quả và khả năng ứng dụng của hệ thống.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

1.1 Hệ quản trị cơ sở Sql

1.1.1 Điểm mạnh và điểm yếu

❖ Điểm mạnh:

Quản lý dữ liệu hiệu quả

- Dễ dàng lưu trữ, truy xuất, cập nhật và xóa dữ liệu bằng ngôn ngữ SQL chuẩn.

Tính toàn vẹn và bảo mật cao

- Có cơ chế kiểm soát truy cập, phân quyền người dùng, và ràng buộc dữ liệu (constraints) để đảm bảo dữ liệu chính xác, an toàn.

Hỗ trợ giao dịch (Transaction)

- Đảm bảo tính ACID (Atomicity, Consistency, Isolation, Durability), giúp dữ liệu không bị lỗi khi thực hiện nhiều thao tác cùng lúc.

Dễ mở rộng và quản lý

- Có thể làm việc với lượng dữ liệu lớn, hỗ trợ sao lưu (backup), khôi phục (restore) dễ dàng.

Chuẩn hóa và phổ biến

- SQL là ngôn ngữ chuẩn, được hầu hết các hệ quản trị lớn như MySQL, SQL Server, Oracle, PostgreSQL sử dụng.

❖ Điểm yếu:

Khó mở rộng theo chiều ngang

- SQL thường khó chia nhỏ dữ liệu trên nhiều máy chủ (so với NoSQL).

Tốc độ giảm khi dữ liệu cực lớn

- Với dữ liệu rất lớn (big data), SQL có thể chậm hơn các hệ thống NoSQL.

Cấu trúc dữ liệu cố định

- Phải định nghĩa bảng và kiểu dữ liệu trước; khó linh hoạt khi thay đổi cấu trúc.

Chi phí bản quyền (với hệ thống lớn)

- Một số hệ quản trị SQL như Oracle, SQL Server có chi phí cao.

Cần kỹ năng chuyên sâu

- Người dùng cần hiểu rõ về thiết kế cơ sở dữ liệu, câu lệnh SQL, và tối ưu truy vấn để đạt hiệu suất cao.

1.1.2 Cấu trúc hệ quản trị cơ sở dữ liệu

❖ Lớp giao tiếp người dùng (User Interface Layer)

Chức năng: Là phần mà người dùng hoặc ứng dụng tương tác trực tiếp.

- Thành phần:
 - Ngôn ngữ truy vấn SQL: Dùng để thao tác dữ liệu (SELECT, INSERT, UPDATE, DELETE...).
 - Công cụ quản trị (GUI): Ví dụ: *phpMyAdmin*, *SQL Server Management Studio (SSMS)*, *Oracle SQL Developer*...
 - API hoặc driver: Cho phép các chương trình (Python, Java, C#...) kết nối với cơ sở dữ liệu.
 -

❖ Lớp xử lý (Database Engine / Processing Layer)

- Chức năng: Là “bộ não” của hệ quản trị, xử lý tất cả các yêu cầu từ người dùng.
- Thành phần:
 - Query Processor (Bộ xử lý truy vấn):
 - Phân tích, tối ưu và thực thi các câu lệnh SQL.
 - Transaction Manager (Quản lý giao dịch):
 - Đảm bảo tính toàn vẹn và nhất quán dữ liệu theo nguyên tắc ACID.
 - Concurrency Control (Kiểm soát đồng thời):
 - Cho phép nhiều người dùng truy cập dữ liệu cùng lúc mà không xung đột.
 - Security Manager (Bảo mật):
 - Quản lý người dùng, mật khẩu, quyền truy cập dữ liệu.

❖ Lớp lưu trữ dữ liệu (Storage Layer / Data Storage Layer)

- Chức năng: Chịu trách nhiệm lưu trữ vật lý dữ liệu trên ổ đĩa hoặc bộ nhớ.
- Thành phần:
 - Database Files (Tập cơ sở dữ liệu): Chứa bảng, chỉ mục, dữ liệu, log...
 - File Manager: Quản lý việc đọc/ghi dữ liệu.
 - Index Manager: Quản lý các chỉ mục để tăng tốc tìm kiếm.
 - Backup & Recovery System: Hỗ trợ sao lưu và phục hồi dữ liệu khi có sự cố.

1.1.3 Tại sao phải cần hệ quản trị cho dữ liệu

❖ Lưu trữ và quản lý dữ liệu tập trung

- Hệ quản trị giúp lưu toàn bộ dữ liệu của hệ thống (ví dụ: thông tin bệnh nhân, lịch hẹn, bác sĩ, dịch vụ nha khoa...) một cách khoa học và an toàn.
- Tránh việc thất lạc, trùng lặp hoặc sai sót dữ liệu.

❖ Dễ dàng truy xuất và cập nhật thông tin

- Người dùng hoặc chương trình có thể tìm kiếm, thêm, sửa, xóa dữ liệu bằng ngôn ngữ SQL nhanh chóng, chính xác.

❖ Đảm bảo tính toàn vẹn và bảo mật dữ liệu

- DBMS hỗ trợ phân quyền người dùng, ràng buộc dữ liệu và sao lưu – khôi phục, giúp dữ liệu luôn an toàn.

❖ Hỗ trợ nhiều người dùng truy cập cùng lúc

- Cho phép nhiều người (admin, bác sĩ, bệnh nhân) cùng thao tác mà không gây xung đột hoặc mất dữ liệu.

❖ Tăng tính linh hoạt và mở rộng của hệ thống

- Dễ dàng thêm bảng, cột, hoặc tính năng mới (ví dụ thêm chức năng “trợ lý bác sĩ”) mà không ảnh hưởng toàn hệ thống.

❖ Tương thích với nhiều công nghệ

- Hầu hết các ngôn ngữ lập trình và nền tảng web (PHP, Python, .NET, Java...) đều kết nối dễ dàng với SQL DBMS như MySQL, SQL Server, PostgreSQL.

1.2 Visual Studio Code

1.2.1 Visual Studio Code là gì

Visual Studio Code chính là ứng dụng cho phép biên tập, soạn thảo các đoạn code để hỗ trợ trong quá trình thực hiện xây dựng, thiết kế website một cách nhanh

chóng. Visual Studio Code hay còn được viết tắt là VS Code. Trình soạn thảo này vận hành mượt mà trên các nền tảng như Windows, macOS, Linux. Hơn thế nữa, VS Code còn cho khả năng tương thích với những thiết bị máy tính có cấu hình tầm trung vẫn có thể sử dụng dễ dàng.

1.2.2 Những ưu điểm nổi bật của Visual Studio Code

Visual Studio Code là gì được rất nhiều người tìm hiểu. Đây cũng là một trong các ứng dụng được dân IT “săn đón” và tải về và sử dụng rất nhiều. Visual Studio Code cũng luôn có những cải tiến và tạo ra đa dạng các tiện ích đi kèm từ đó giúp cho các lập trình viên sử dụng dễ dàng hơn. Trong đó có thể kể đến những ưu điểm sau:

- Đa dạng ngôn ngữ lập trình giúp người dùng thỏa sức sáng tạo và sử dụng như HTML, CSS, JavaScript, C++,...
- Ngôn ngữ, giao diện tối giản, thân thiện, giúp các lập trình viên dễ dàng định hình nội dung.
- Các tiện ích mở rộng rất đa dạng và phong phú.
- Tích hợp các tính năng quan trọng như tính năng bảo mật (Git), khả năng tăng tốc xử lý vòng lặp (Debug),...
- Đơn giản hóa việc tìm quản lý hết tất cả các Code có trên hệ thống.

1.2.3 Tại sao nên sử dụng Visual Studio Code

Không phải ngẫu nhiên mà Visual Studio Code được các lập trình viên ưa chuộng sử dụng. Visual Studio Code mang rất nhiều ưu điểm vượt trội so với bất kỳ IDE nào khác:

- Hỗ trợ đa nền tảng: Linux, Mac, Windows,...
- Hỗ trợ đa ngôn ngữ: C/C++, C#, F#, JavaScript, JSON, Visual Basic, HTML, CSS,...
- Ít dung lượng
- Tính năng mạnh mẽ
- Intellisense chuyên nghiệp
- Giao diện thân thiện
- Kiến trúc mạnh mẽ và người dùng có thể khai thác mở rộng
- Số lượng người sử dụng lớn tạo nên ộng đồng hỗ trợ rộng rãi

1.3 Ngôn ngữ lập trình

1.3.1 Ngôn ngữ HTML

Ngôn ngữ HTML (HyperText Markup Language – ngôn ngữ siêu văn bản) là một trong các loại ngôn ngữ được sử dụng trong lập trình web. Khi truy cập một trang web cụ thể là click vào các đường link, chúng ta sẽ được dẫn tới nhiều trang các nhau, và các trang này được gọi là một tài liệu HTML (tập tin HTML).

Một trang HTML như vậy được cấu thành bởi nhiều phần tử HTML nhỏ và được quy định bằng các thẻ tag. Chúng ta có thể phân biệt một trang web được viết bằng ngôn ngữ HTML hay PHP thông qua đường link của nó. Ở cuối các trang HTML thường hay có đuôi là .HTML hoặc .HTM

HTML là ngôn ngữ lập trình web được đánh giá là đơn giản. Mọi trang web, mọi trình duyệt web đều có thể hiển thị tốt ngôn ngữ HTML. Hiện nay, phiên bản mới nhất của HTML là HTML 5 với nhiều tính năng tốt và chất lượng hơn so với các phiên bản cũ của HTML.

Vai trò của HTML trong lập trình web:

HTML là một loại ngôn ngữ đánh dấu siêu văn bản nên các chức năng của nó cũng xoay quanh yếu tố này. Cụ thể, HTML giúp cấu thành các cấu trúc cơ bản trên một website (chia khung sườn, bố cục các thành phần trang web) và góp phần hỗ trợ khai báo các tập tin kỹ thuật số như video, nhạc, hình ảnh.

Ưu điểm nổi trội nhất và cũng là thế mạnh của HTML là khả năng xây dựng cấu trúc và khiến trang web đi vào quy củ một hệ thống hoàn chỉnh. HTML chứa những yếu tố cần thiết mà dù website có thuộc thể loại nào, giao tiếp với ngôn ngữ lập trình nào để xử lý dữ liệu thì nó vẫn phải cần đến *ngôn ngữ HTML* để hiển thị nội dung cho người truy cập.

Đối với các lập trình viên hay nhà phát triển web, họ đều phải học HTML như một loại ngôn ngữ cơ bản trước khi bắt tay vào thiết kế trang web nào.

Các thẻ trong HTML:

Các trang HTML được quy định bằng các thẻ tag. Những thẻ này được chứa trong các dấu ngoặc đơn dạng: **<tên thẻ>**. Trừ một vài thẻ đặc biệt, hầu hết các thẻ cơ bản đều có các thẻ đóng tương ứng với nó. Ví dụ, thẻ **<html>** có thẻ đóng tương ứng

là `</html>`, thẻ `<body>` có thể đóng tương ứng là `</body>` ... Bảng 1.1 là các mẫu thẻ tag thường gặp trong HTML.

Bảng 1.1 Các mẫu thẻ thường gặp trong HTML

Tag	Giải thích
<code><!DOCTYPE...></code>	Còn gọi là thẻ khai báo một tài liệu HTML. Thẻ này xác định loại tài liệu và phiên bản HTML.
<code><html></code>	Thẻ này chứa đựng các tài liệu HTML đầy đủ. Ở đầu trang sẽ xuất hiện các thẻ <code><head></code> , <code></head></code> và thân tài liệu là các thẻ <code><body></code> , <code></body></code> .
<code><head></code>	Thẻ này đại diện cho đầu trang tài liệu mà có thể giữ các thẻ HTML như <code><title></code> , <code><link></code> ...
<code><title></code>	Thẻ <code><title></code> được sử dụng trong thẻ <code><head></code> chỉ tiêu đề tài liệu.
<code><body></code>	Thẻ này đại diện cho thân tài liệu và giữ các thẻ như <code><h1></code> , <code><div></code> , <code><p></code> ...
<code><h1></code>	Thẻ tag này đại diện cho các tiêu đề trang.
<code><p></code>	Thẻ tag này đại diện cho định dạng các đoạn văn trong trang web.

1.3.2 Ngôn ngữ CSS

Định nghĩa

CSS là viết tắt của Cascading Style Sheets. Đây là một ngôn style sheet được sử dụng để mô tả giao diện và định dạng của một tài liệu viết bằng ngôn ngữ đánh dấu (markup). CSS cung cấp một tính năng bổ sung cho HTML. CSS thường được sử dụng với HTML

để thay đổi phong cách của trang web và giao diện người dùng. Ngôn ngữ này cũng có thể được sử dụng với bất kỳ loại tài liệu XML nào bao gồm cả XML đơn giản, SVG và XUL.

CSS được sử dụng cùng với HTML và JavaScript trong hầu hết các trang web để tạo giao diện người dùng cho các ứng dụng web và giao diện người dùng cho nhiều ứng dụng di động.

Ba lợi ích chính của CSS

- Giải quyết một vấn đề lớn:

Trước khi có CSS, các thẻ như phông chữ, màu sắc, kiểu nền, các sắp xếp phân tử, đường viền và kích thước phải được lặp lại trên mọi trang web. Đây là một quá trình rất dài tốn thời gian và công sức.

- Tiết kiệm rất nhiều thời gian:

Định nghĩa kiểu CSS được lưu trong các tệp CSS bên ngoài vì vậy có thể thay đổi toàn bộ trang web bằng cách thay đổi chỉ một tệp.

- Cung cấp thêm các thuộc tính:

CSS cung cấp các thuộc tính chi tiết hơn HTML để định nghĩa giao diện của trang web.

1.3.3 Ngôn ngữ PHP

❖ Giới thiệu chung:

PHP khởi đầu như là một dự án mã nguồn mở nhỏ, nhưng theo đà phát triển, ngày càng nhiều người thấy rằng nó càng ngày càng hữu ích. PHP được phát triển từ một sản phẩm có tên là PHP/FI. PHP/FI do Rasmus Lerdorf tạo ra năm 1994, ban đầu được xem như là một tập con đơn giản của các mã kịch bản Perl để theo dõi tình hình truy cập đến bản sơ yếu lý lịch của ông trên mạng. Ông đã đặt tên cho bộ mã kịch bản này là "Personal Home Page Tools". Khi cần đến các chức năng rộng hơn, Rasmus đã viết ra một bộ thực thi bằng C lớn hơn để có thể truy vấn tới Database và giúp cho người sử dụng phát triển các ứng dụng web đơn giản. Rasmus đã quyết định công bố mã nguồn của PHP/FI cho mọi người xem, sử dụng cũng như sửa các lỗi có trong nó, đồng thời cải tiến mã nguồn.

PHP là ngôn ngữ lập trình kịch bản viết cho máy chủ mà được nhúng trong HTML. Nó được sử dụng để quản lý nội dung động, Database, Session tracking, ...

PHP được tích hợp với một số Database thông dụng như MySQL, PostgreSQL, Oracle, Sybase, Informix, và Microsoft SQL Server.

PHP thực thi rất tuyệt vời, đặc biệt khi được biên dịch như là một Apache Module trên Unix side. MySQL Server, khi được khởi động, thực thi các truy vấn phức tạp với các tập hợp kết quả không lờ trong thời gian Record-setting.

PHP hỗ trợ một số lượng rộng rãi các giao thức lớn như POP3, IMAP, và LDAP. PHP4 bổ sung sự hỗ trợ cho Java và các cấu trúc đối tượng phân phối (COM và CORBA).

Cú pháp PHP là giống C.

❖ *Chức năng của PHP*

PHP thực hiện các hàm hệ thống, ví dụ: từ các file trên một hệ thống, nó có thể tạo, mở, đọc, ghi và đóng chúng.

PHP có thể xử lý các form, ví dụ: thu thập dữ liệu từ file, lưu dữ liệu vào một file, thông qua email có thể gửi dữ liệu, trả về dữ liệu tới người dùng.

Chúng ta có thể thêm, xóa, sửa đổi các phần tử bên trong Database thông qua PHP.

Truy cập các biến Cookie và thiết lập Cookie.

Sử dụng PHP, có thể hạn chế người dùng truy cập vào một số trang trong Website.

Nó có thể mã hóa dữ liệu.

❖ *Đặc trưng của PHP*

Các đặc trưng quan trọng làm PHP trở thành ngôn ngữ khá tiện lợi:

- Đơn giản hóa
- Hiệu quả
- Bảo mật cao
- Linh động
- Thân thiện

1.4. Laravel framework

Framework là một thư viện các hàm xây dựng sẵn, phục vụ cho lập trình PHP.

Cho tới năm 2015, thì Framework laravel hiện đang đứng top 1 thế giới về mức độ phổ biến và ưu dùng.

Được cộng đồng hỗ trợ phát triển mạnh mẽ. Tháng 8 năm 2014, Laravel Framework được xem như là một dự án PHP phổ biến nhất trên Github.

Sau nhiều lần được cộng đồng Laravel thế giới hỗ trợ phát triển thì phiên bản mới nhất hiện nay là Laravel 5.2 và sắp tới là phiên bản 5.3.

Các tính năng cơ bản của Laravel Framework

- **Bundles:** Ở laravel phiên bản 3.x, cung cấp một hệ thống đóng gói các module, với rất nhiều tính năng đi kèm.
- **Composer:** Ở laravel phiên bản 4.x, được sử dụng như một công cụ quản lý với tính năng như thêm các gói cài đặt, các chức năng PHP phụ trợ cho Laravel có trong kho Packagist.
- **Eloquent ORM** (object relation mapping): ánh xạ các đối tượng và quan hệ cơ sở dữ liệu, cung cấp các phương thức nội bộ để thực thi đồng thời cũng bổ sung các tính năng hạn chế về mối quan hệ giữa các đối tượng cơ sở dữ liệu. Eloquent ORM trình bày các bảng trong cơ sở dữ liệu dưới dạng các lớp, cung cấp thêm lựa chọn truy cập cơ sở dữ liệu trực tiếp mới mẻ hơn, chuyên nghiệp hơn.
- **Application logic:** Là một phần của phát triển ứng dụng, được sử dụng bởi bộ điều khiển controllers.
- **Routes:** Định nghĩa mối quan hệ giữa các đường dẫn (url), các liên kết (link) . Khi một liên kết được tạo ra bằng cách sử dụng tên của routes, thì một định danh liên kết thống nhất sẽ được tạo ra bởi laravel.
- **Restful Controller:** cung cấp các tùy chọn để tách các logic phía sau các request HTTP POST, GET.
- **Class auto loading:** cung cấp việc tải tự động các class trong PHP, mà không cần include các class vào. Tùy thuộc vào yêu cầu các class cần thiết sẽ được nạp vào, hạn chế các class không cần thiết.
- **View:** chứa các mã html, hiển thị dữ liệu được chỉ định bởi controller
- **Migrations:** cung cấp một hệ thống kiểm soát các phiên bản lược đồ cơ sở dữ liệu (database cheme), làm cho web ứng dụng có khả năng tương tác phù hợp những thay đổi logic, các đoạn mã code của ứng dụng và những thay đổi cần thiết trong việc bố trí cơ sở dữ liệu, triển khai nói lỏng và cập nhật các ứng dụng.
- **Unit Testing:** đóng một vai trò quan trọng trong Laravel, Unit testing chứa rất nhiều các hệ thống unit testing, giúp phát hiện và ngăn chặn lỗi trong khuôn khổ.

CHƯƠNG 2: KHẢO SÁT THIẾT KẾ HỆ THỐNG

2.1 Giới thiệu chung về bài toán

Bài toán đặt ra là xây dựng một hệ thống website đặt lịch khám cho phòng khám nha khoa nhằm hỗ trợ bệnh nhân và phòng khám trong việc quản lý, tổ chức và thực hiện các lịch hẹn khám một cách hiệu quả và tiện lợi. Hệ thống sẽ cung cấp các chức năng cần thiết cho bệnh nhân, bác sĩ, trợ lý bác sĩ và người quản trị hệ thống, bao gồm việc đặt lịch khám trực tuyến, xem thông tin dịch vụ và bác sĩ, quản lý lịch làm việc, xác nhận lịch hẹn và quản lý người dùng.

Hệ thống cần đảm bảo khả năng xử lý nhiều người dùng truy cập đồng thời, lưu trữ và bảo mật thông tin bệnh nhân một cách an toàn, đồng thời cung cấp trải nghiệm người dùng mượt mà với giao diện trực quan, dễ sử dụng. Ngoài ra, việc quản lý lịch khám chính xác, cập nhật thông tin theo thời gian thực và gửi thông báo nhanh chóng cho bệnh nhân và bác sĩ là một trong những yêu cầu quan trọng của hệ thống.

2.2 Phân tích yêu cầu

2.2.1 Bác sĩ

- ❖ Đăng nhập

Người dùng đăng nhập theo tài khoản đã tạo. Nếu đăng nhập sai, yêu cầu nhập lại.

- ❖ Quản lý các lịch khám

Người dùng đăng nhập tài khoản bác sĩ có thể thực hiện các thao tác như xem lịch khám, chấp nhận lịch khám hoặc hủy lịch khám.

- ❖ Xem thông tin bệnh nhân

Người dùng có thể xem tên, tuổi, lý do khám của bệnh nhân.

- ❖ Quản lý lịch làm việc, ngày nghỉ

Người dùng có thể xem lịch làm việc hoặc thay đổi lịch làm việc khi có công việc đột xuất, giảm tình trạng trùng lịch.

- ❖ Ghi chú, chẩn đoán và lưu kết quả điều trị

Người dùng khám cho bệnh nhân ghi rõ chuẩn đoán, phàn điều trị và lưu kết quả khám trên hệ thống.

2.1.2 Trợ lý bác sĩ

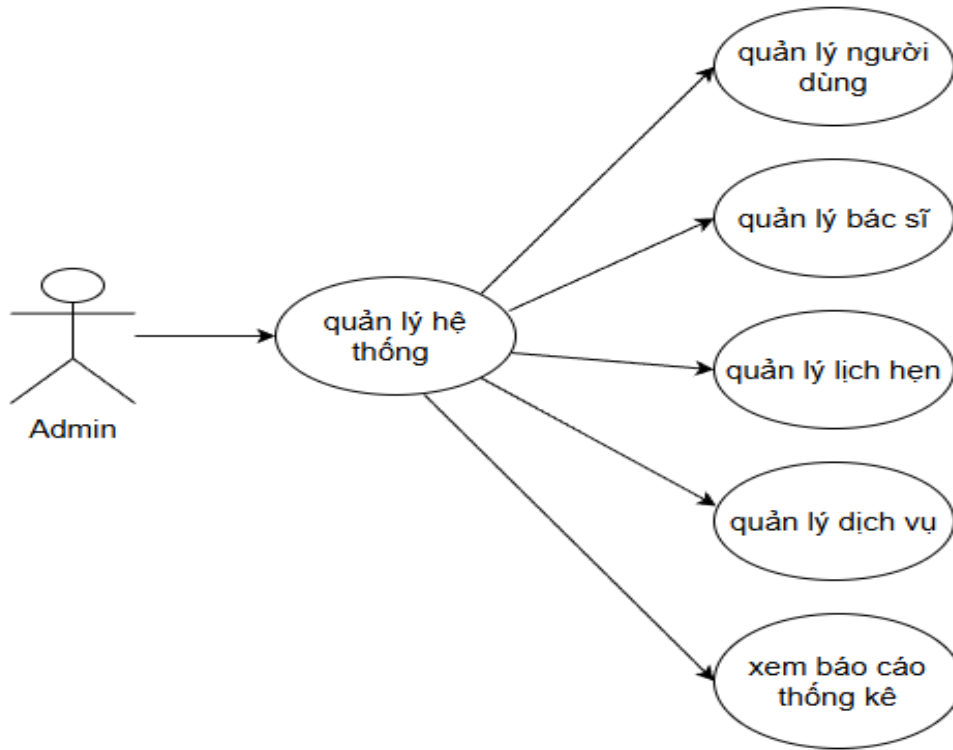
- ❖ Đăng nhập
Người dùng đăng nhập theo tài khoản đã tạo. Nếu đăng nhập sai, yêu cầu nhập lại.
- ❖ Xem danh sách lịch hẹn của bệnh nhân
Người dùng đăng nhập tài khoản trợ lý có thể thực hiện thao tác xem danh sách bệnh đặt lịch hẹn và hủy lịch hẹn.
- ❖ Xác nhận, sắp xếp hoặc hủy lịch theo yêu cầu bác sĩ.
Người dùng có thể sắp xếp lịch hẹn khi bác sĩ yêu cầu để tránh trùng lịch hẹn với bệnh nhân.
- ❖ Cập nhật trạng thái cuộc hẹn
Người dùng có thể theo dõi trạng thái bệnh nhân hoặc bác sĩ hoạt động để cập nhật lại lịch hẹn.
- ❖ Liên hệ và nhắc lịch cho bệnh nhân
Người dùng có thể thông báo cho bệnh nhân bằng cách liên hệ bằng điện thoại khi sắp đến lịch khám để giảm tránh tình trạng lỡ lịch hẹn.
- ❖ Quản lý thông tin bác sĩ và phòng khám
Người dùng có thể thay đổi thông tin bác sĩ, hoặc phòng khám.

2.1.2 Bệnh nhân

- ❖ Đăng ký
Người dùng đăng nhập theo tài khoản đã tạo. Nếu đăng nhập sai, yêu cầu nhập lại.
- ❖ Xem danh sách bác sĩ và dịch vụ nha khoa
Người dùng có thể xem danh sách bác sĩ và dịch vụ muốn khám trên trang chủ.
- ❖ Đặt lịch khám, chọn ngày và khung giờ phù hợp
Người dùng khi xem xong thông tin về dịch vụ, bác sĩ phù hợp với mong muốn thì có thể đặt lịch khám chọn ngày, thời gian phù hợp.
- ❖ Hủy hoặc đổi lịch khi cần
Tăng tính linh hoạt và thuận tiện cho bệnh nhân.
- ❖ Xem lịch sử khám và kết quả điều trị
Người dùng có thể xem lại lịch sử khám về kết quả bác sĩ đã ghi cho mình.
- ❖ Cập nhật thông tin cá nhân
Người dùng có thể thay đổi thông tin cá nhân khi nhập sai địa chỉ, điện thoại hoặc sai tên.

2.2 Các biểu đồ Use Case

2.2.1 Use Case Admin



Hình 2. 1 Biểu đồ Use Case Admin

Mô tả Use Case Admin

Bảng 2.1 Đặt tả Use Case quản lý người dùng

Nội dung	Mô tả
Tên chức năng	Quản lý người dùng
Luồng xử lý chính	1. Admin truy cập mục “Quản lý người dùng”. 2. Hệ thống hiển thị danh sách người dùng. 3. Admin có thể thêm mới người dùng. 4. Admin sửa thông tin người dùng. 5. Admin khóa/mở khóa tài khoản người dùng. 6. Hệ thống lưu thay đổi.

Luồng thay thế	- Người dùng nhập thông tin không hợp lệ → Hệ thống báo lỗi. - Tài khoản đã tồn tại → Hệ thống yêu cầu đổi email/tên đăng nhập. - Mất kết nối CSDL → Hiện thị thông báo thất bại.
Kết quả	Danh sách người dùng được cập nhật đúng.

Bảng 2.2 Đặt tả Use Case quản lý bác sĩ

Nội dung	Mô tả
Tên chức năng	Quản lý bác sĩ
Luồng xử lý chính	1. Admin mở mục “Quản lý bác sĩ”. 2. Xem danh sách bác sĩ. 3. Thêm mới bác sĩ (thông tin, chuyên môn, lịch làm việc). 4. Sửa hoặc xóa bác sĩ. 5. Hệ thống lưu thay đổi.
Luồng thay thế	- Thông tin không hợp lệ → báo lỗi. - Xóa bác sĩ đang có lịch hẹn → bị chặn. - Lỗi kết nối hệ thống.
Kết quả	Cập nhật danh sách bác sĩ, hồ sơ hợp lệ.

Bảng 2.3 Đặt tả Use Case quản lý lịch hẹn

Nội dung	Mô tả
Tên chức năng	Quản lý lịch hẹn

Luồng xử lý chính	1. Admin truy cập “Quản lý lịch hẹn”. 2. Xem danh sách theo ngày/bác sĩ/bệnh nhân. 3. Xem chi tiết lịch hẹn. 4. Thay đổi thông tin hoặc hủy lịch. 5. Hệ thống gửi thông báo cho bệnh nhân.
Luồng thay thế	- Tên dịch vụ trùng → báo lỗi. - Giá hoặc dữ liệu nhập không hợp lệ. - Không thể xóa dịch vụ đang được dùng trong lịch hẹn. - Tên dịch vụ trùng → báo lỗi. - Giá hoặc dữ liệu nhập không hợp lệ. - Không thể xóa dịch vụ đang được dùng trong lịch hẹn.
Kết quả	Lịch hẹn được tổ chức hợp lý, đồng bộ cho bệnh nhân/bác sĩ.

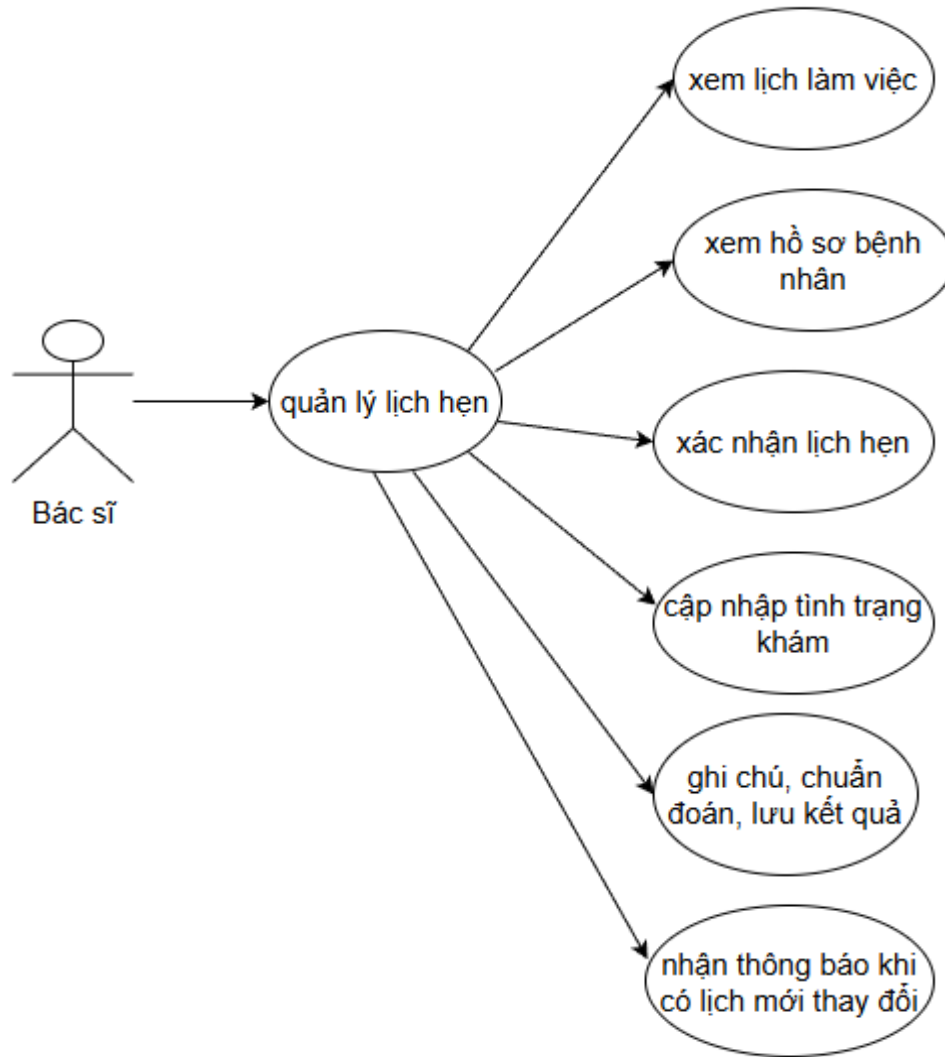
Bảng 2.4 Đặt tả Use Case quản lý dịch vụ

Nội dung	Mô tả
Tên chức năng	Quản lý dịch vụ
Luồng xử lý chính	1. Admin mở “Quản lý dịch vụ”. 2. Xem danh sách dịch vụ nha khoa. 3. Thêm mới dịch vụ (tên, giá, mô tả...). 4. Chỉnh sửa hoặc xóa dịch vụ. 5. Hệ thống lưu cập nhật.
Luồng thay thế	- Tên dịch vụ trùng → báo lỗi. - Giá hoặc dữ liệu nhập không hợp lệ. - Không thể xóa dịch vụ đang được dùng trong lịch hẹn.
Kết quả	Danh sách dịch vụ chuẩn hóa, hiển thị đúng cho bệnh nhân.

Bảng 2.5 Đặt tả Use Case quản lý báo cáo thống kê

Nội dung	Mô tả
Tên chức năng	Quản lý báo cáo thống kê
Luồng xử lý chính	1. Admin mở mục “Thống kê – báo cáo”. 2. Chọn loại báo cáo: lịch hẹn, phòng khám, dịch vụ... 4. Admin xuất báo cáo (PDF/Excel).
Luồng thay thế	- Không có dữ liệu → hiển thị thông báo rỗng. - Lỗi khi xuất file.
Kết quả	Admin có dữ liệu phân tích để ra quyết định.

2.2.2 Use Case bác sĩ



Hình 2.2 Biểu đồ Use Case bác sĩ

Mô tả Use Case Bác sĩ

Bảng 2.6 Đặt tả Use Case xem lịch làm việc

Nội dung	Mô tả
Tên chức năng	Xem lịch làm việc
Luồng xử lý chính	1. Bác sĩ đăng nhập hệ thống. 2. Truy cập “Lịch làm việc”.

	3. Hệ thống hiển thị lịch khám theo ngày/tuần/tháng. 4. Bác sĩ xem chi tiết từng lịch
Luồng thay thế	- Lỗi tải dữ liệu lịch làm việc. - Không có lịch hẹn nào trong ngày → hiển thị thông báo.
Kết quả	Bác sĩ nắm rõ lịch làm việc và các cuộc hẹn sắp tới.

Bảng 2.7 Đặt tả Use Case xem hồ sơ bệnh nhân

Nội dung	Mô tả
Tên chức năng	Xem hồ sơ bệnh nhân
Luồng xử lý chính	1. Bác sĩ chọn một lịch hẹn. 2. Hệ thống hiển thị thông tin bệnh nhân (thông tin cá nhân, tiền sử răng miệng, toa thuốc, kết quả điều trị trước...).
Luồng thay thế	- Hồ sơ bệnh nhân thiếu dữ liệu. - Lỗi tải hồ sơ.
Kết quả	Bác sĩ có đủ thông tin để chuẩn bị khám và điều trị.

Bảng 2.8 Đặt tả Use Case xác nhận lịch hẹn

Nội dung	Mô tả
Tên chức năng	Xác nhận lịch hẹn
Luồng xử lý chính	1. Bác sĩ chọn lịch hẹn mới. 2. Kiểm tra thông tin bệnh nhân và thời gian. 3. Nhấn “Xác nhận”. 4. Hệ thống gửi thông báo cho bệnh nhân.

Luồng thay thế	- Thời gian trùng hoặc bác sĩ bận → hệ thống báo lỗi. - Kết nối mạng lỗi → không gửi được xác nhận.
Kết quả	Lịch hẹn được xác nhận và đồng bộ với bệnh nhân.

Bảng 2.9 Đặt tả Use Case cập nhật tình trạng khám

Nội dung	Mô tả
Tên chức năng	Cập nhật tình trạng khám
Luồng xử lý chính	1. Sau khi khám, bác sĩ mở lịch hẹn tương ứng. 2. Chọn tình trạng: đã khám, đang khám, hủy... 3. Lưu cập nhật.
Luồng thay thế	- Thông tin không hợp lệ. - Lỗi hệ thống khi lưu.
Kết quả	Trạng thái lịch hẹn phản ánh đúng tiến trình khám.

Bảng 2.10 Đặt tả Use Case ghi chú, chuẩn đoán, lưu kết quả khám

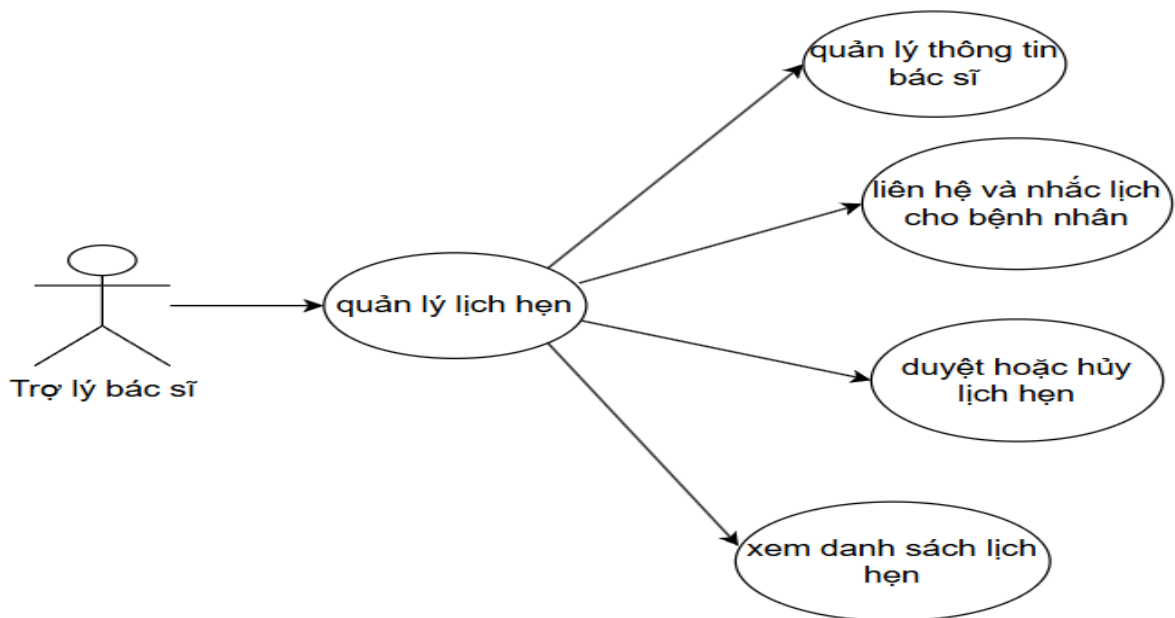
Nội dung	Mô tả
Tên chức năng	Ghi chú – chuẩn đoán – lưu kết quả khám
Luồng xử lý chính	1. Bác sĩ mở mục ghi chú cho lịch hẹn. 2. Nhập các thông tin: chuẩn đoán, kết quả khám, hình ảnh X-quang, đề xuất điều trị... 3. Lưu hồ sơ khám.
Luồng thay thế	- Thiếu thông tin bắt buộc (ví dụ chuẩn đoán). - File đính kèm lỗi hoặc không tải được.

Kết quả	Hồ sơ điều trị được lưu vào hệ thống, bệnh nhân có thể xem sau.
---------	---

Bảng 2.11 Đặt tả Use Case nhận thông báo khi lịch thay đổi

Nội dung	Mô tả
Tên chức năng	Nhận thông báo khi có lịch mới hoặc thay đổi
Luồng xử lý chính	1. Bác sĩ mở mục ghi chú cho lịch hẹn. 2. Nhập các thông tin: chuẩn đoán, kết quả khám, hình ảnh X-quang, đề xuất điều trị... 3. Lưu hồ sơ khám.
Luồng thay thế	- Thiếu thông tin bắt buộc (ví dụ chuẩn đoán). - File đính kèm lỗi hoặc không tải được.
Kết quả	Hồ sơ điều trị được lưu vào hệ thống, bệnh nhân có thể xem sau.

2.2.3 Use Case trợ lý bác sĩ



Hình 2.4 Biểu đồ Use Case trợ lý bác sĩ

Mô tả Use Case trợ lý bác sĩ

Bảng 2.12 Đặt tả Use Case quản lý thông tin bác sĩ

Nội dung	Mô tả
Tên chức năng	Quản lý thông tin bác sĩ
Luồng xử lý chính	1. Trợ lý đăng nhập hệ thống. 2. Chọn mục “Thông tin bác sĩ”. 3. Xem lịch làm việc, chuyên môn, lịch hẹn của bác sĩ. 4. Chỉnh sửa thông tin cần thiết (nếu được phân quyền). 5. Lưu thay đổi.
Luồng thay thế	- Nhập thiếu thông tin → hệ thống báo lỗi. - Không đủ quyền chỉnh sửa → hệ thống chặn thao tác. - Lỗi kết nối dữ liệu.
Kết quả	Thông tin bác sĩ được cập nhật đầy đủ, hỗ trợ bố trí lịch làm việc.

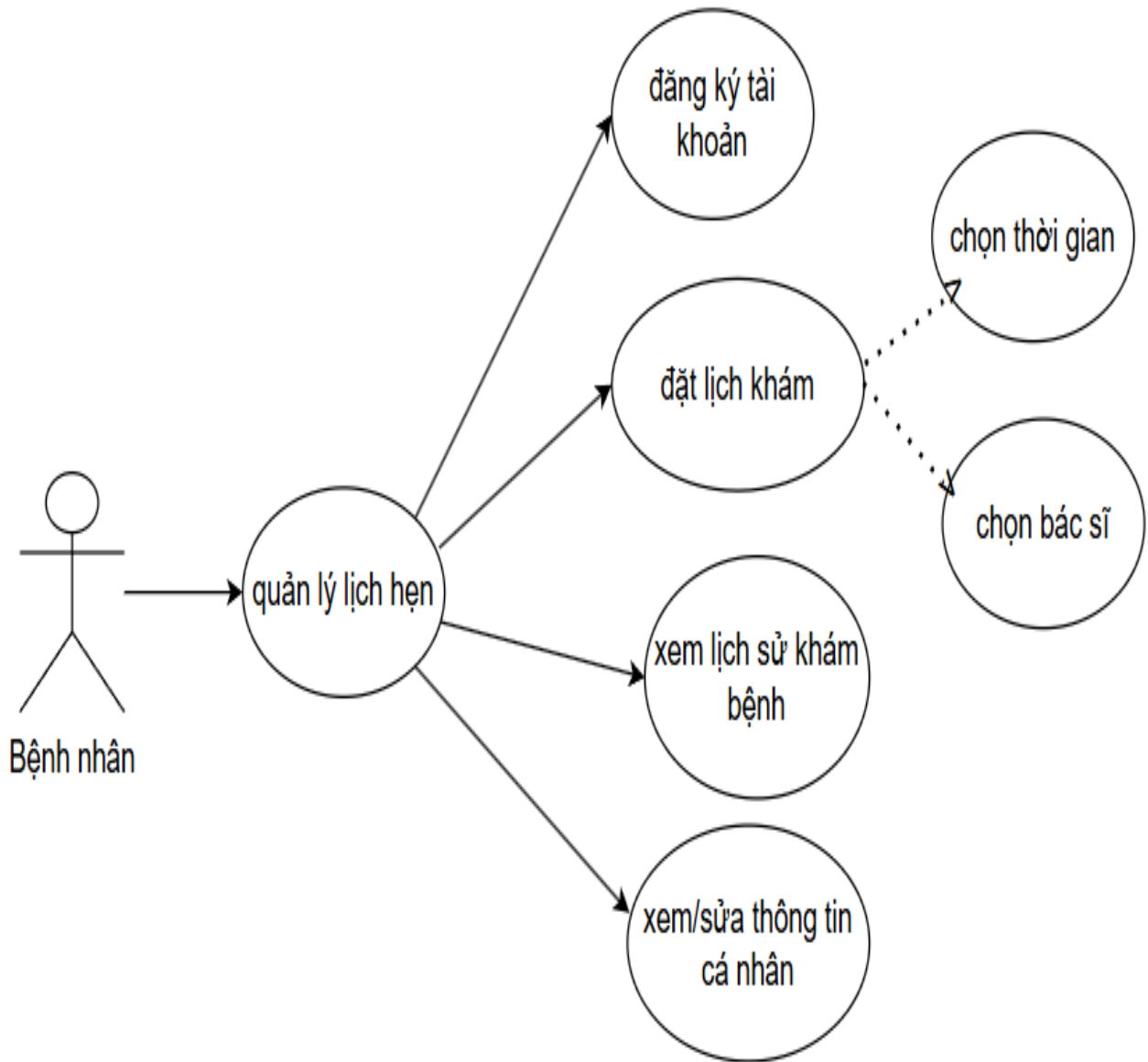
Bảng 2.13 Đặt tả Use Case liên hệ và nhắc lịch cho bệnh nhân

Nội dung	Mô tả
Tên chức năng	Liên hệ và nhắc lịch cho bệnh nhân
Luồng xử lý chính	1. Trợ lý truy cập danh sách lịch hẹn. 2. Chọn bệnh nhân cần nhắc.
Luồng thay thế	3. Gửi SMS/Email/cuộc gọi nhắc lịch. 4. Ghi lại trạng thái đã liên h
Kết quả	- Không liên lạc được bệnh nhân. - Lỗi hệ thống gửi thông báo. - Số điện thoại/email sai.

Bảng 2.14 Đặt tả Use Case duyệt hoặc hủy lịch hẹn

Nội dung	Mô tả
Tên chức năng	Duyệt hoặc hủy lịch hẹn
Luồng xử lý chính	1. Tr� lý xem danh sách lịch mới đăng ký. 2. Kiểm tra thông tin, lịch bác sĩ. 3. Nhấn “Duyệt” hoặc “Hủy”. 4. Hệ thống gửi thông báo cho bệnh nhân.
Luồng thay thế	- Lịch trùng giờ → báo lỗi. - Bác sĩ bận hoặc nghỉ → yêu cầu dời lịch. - Hủy nhưng không gửi được thông báo.
Kết quả	Lịch hẹn được xác nhận hoặc hủy đúng quy trình

2.2.4 Use Case bệnh nhân



Hình 2. 5 Biểu đồ Use Case bệnh nhân

Mô tả Use Case bệnh nhân

Bảng 2.15 Đặt tả Use Case đăng ký tài khoản

Nội dung	Mô tả
Tên chức năng	Đăng ký tài khoản

Luồng xử lý chính	1. Bệnh nhân mở trang đăng ký. 2. Nhập thông tin cá nhân (tên, SĐT, email...). 3. Nhập mật khẩu và xác nhận..
Luồng thay thế	- Email/SĐT đã tồn tại. - Nhập thiếu thông tin → báo lỗi.
Kết quả	Tài khoản mới được tạo và kích hoạt để sử dụng hệ thống.

Bảng 2.16 Đặt tả Use Case đặt lịch khám

Nội dung	Mô tả
Tên chức năng	Đặt lịch khám
Luồng xử lý chính	1. Bệnh nhân đăng nhập hệ thống. 2. Chọn chức năng “Đặt lịch khám”. 3. Chọn thời gian khám. 4. Chọn bác sĩ hoặc dịch vụ. 5. Xác nhận thông tin đặt lịch. 6. Hệ thống tạo lịch hẹn và gửi thông báo.
Luồng thay thế	- Trùng lịch hoặc thời gian không khả dụng. - Bác sĩ không làm việc trong khung giờ đó. - Thông tin không hợp lệ → báo lỗi. - Mạng yếu → đặt lịch thất bại.
Kết quả	Lịch hẹn được tạo thành công và sẽ chờ duyệt từ trợ lý/bác sĩ.

Bảng 2.17 Đặt tả Use Case xem lịch sử khám

Nội dung	Mô tả
Tên chức năng	Xem lịch sử khám bệnh

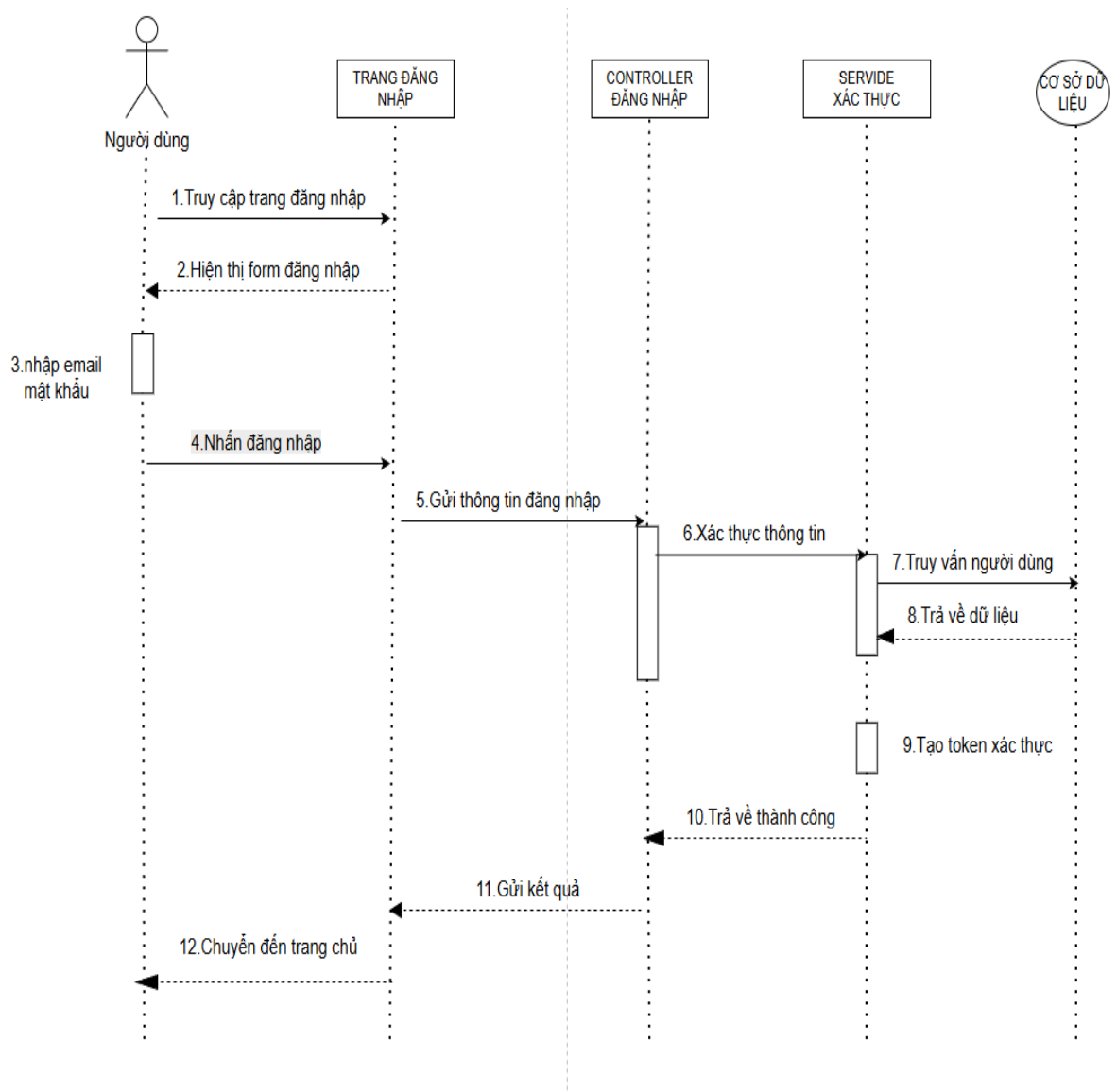
Luồng xử lý chính	1. Bệnh nhân mở mục “Lịch sử khám”. 2. Hệ thống hiển thị danh sách các lần khám trước đây. 3. Bệnh nhân có thể xem chi tiết: chẩn đoán, toa thuốc, hình ảnh...
Luồng thay thế	- Không có lịch sử khám (bệnh nhân mới). - Lỗi tải dữ liệu.
Kết quả	Bệnh nhân xem được toàn bộ lịch sử điều trị của mình.

Bảng 2.18 Đặt tả Use Case xem/sửa thông tin cá nhân

Nội dung	Mô tả
Tên chức năng	Xem/Sửa thông tin cá nhân
Luồng xử lý chính	1. Bệnh nhân truy cập “Thông tin cá nhân”. 2. Hệ thống hiển thị thông tin hiện có. 3. Bệnh nhân chỉnh sửa dữ liệu (địa chỉ, SĐT, ảnh...). 4. Nhấn lưu thay đổi.
Luồng thay thế	- Nhập sai định dạng (email, SĐT). - Thiếu thông tin bắt buộc. - Lỗi hệ thống khi lưu.
Kết quả	Thông tin cá nhân được cập nhật chính xác.

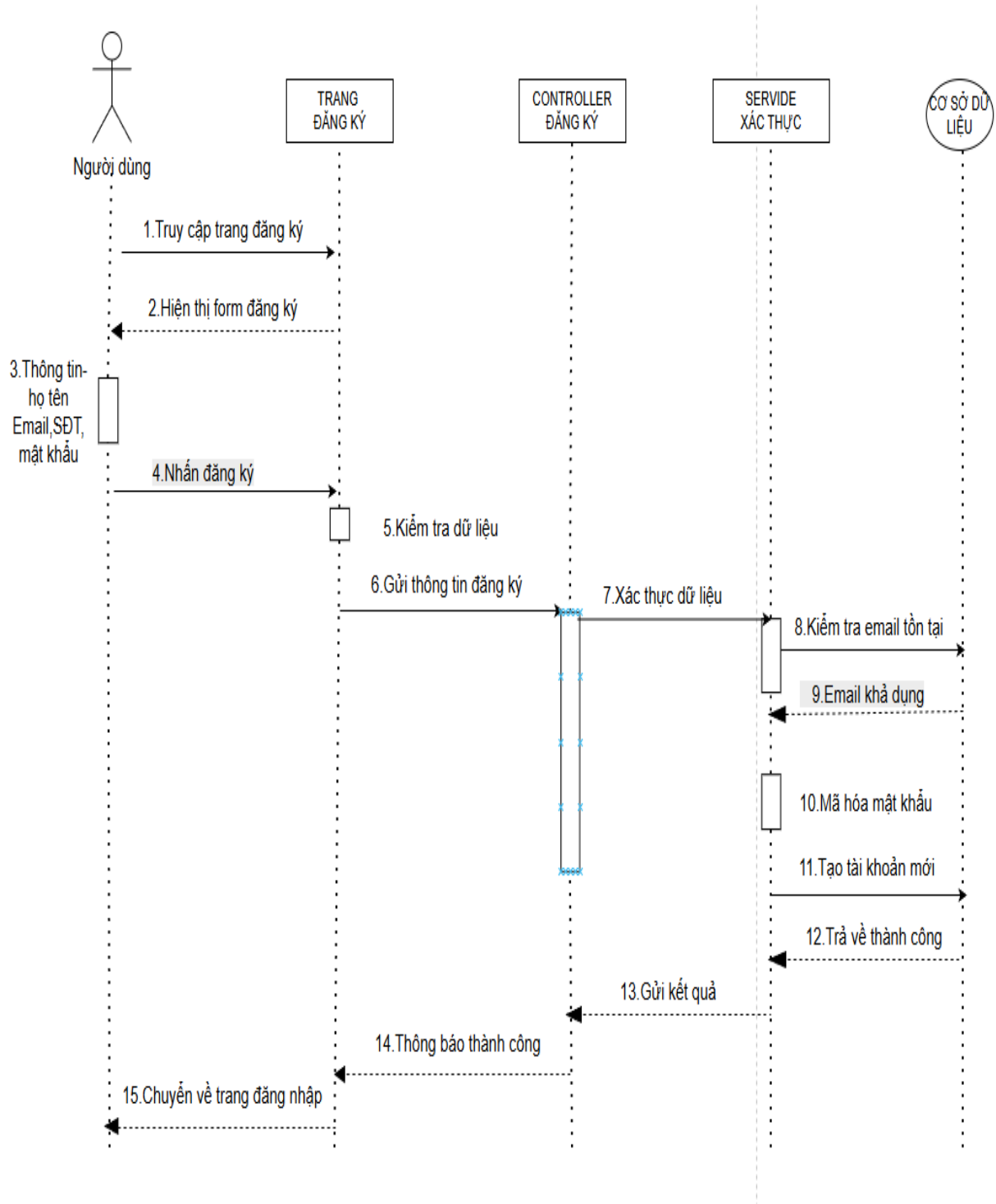
2.3 Biểu đồ tuần tự

2.3.1 Biểu đồ tuần tự chức năng “ Đăng nhập ”



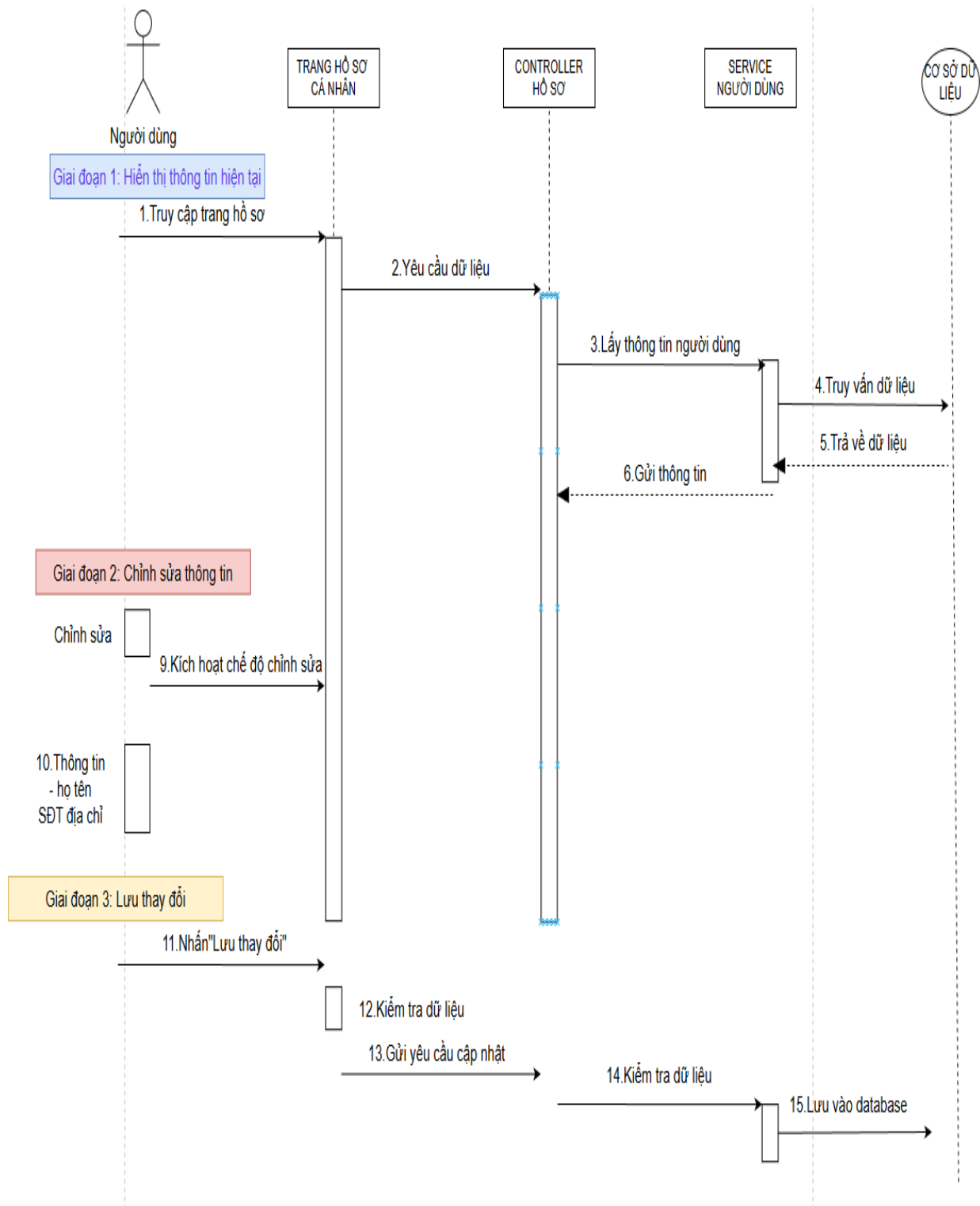
Hình 2.1 Biểu đồ tuần tự chức năng đăng nhập

2.3.2 Biểu đồ tuần tự chức năng “Đăng ký”



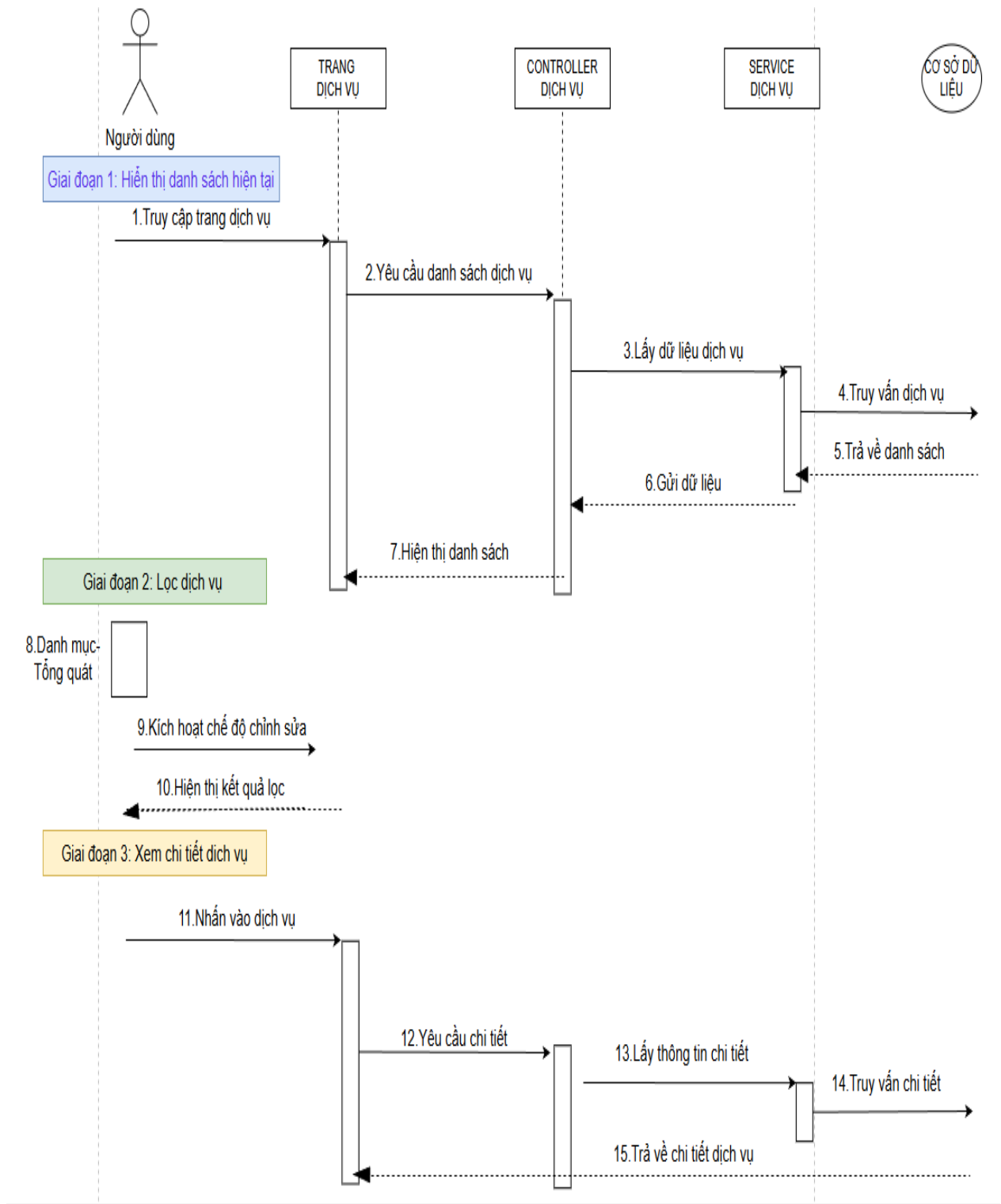
Hình 2.2 Biểu đồ tuần tự chức năng đăng ký

2.3.2 Biểu đồ tuần tự chức năng “Chỉnh sửa thông tin cá nhân”



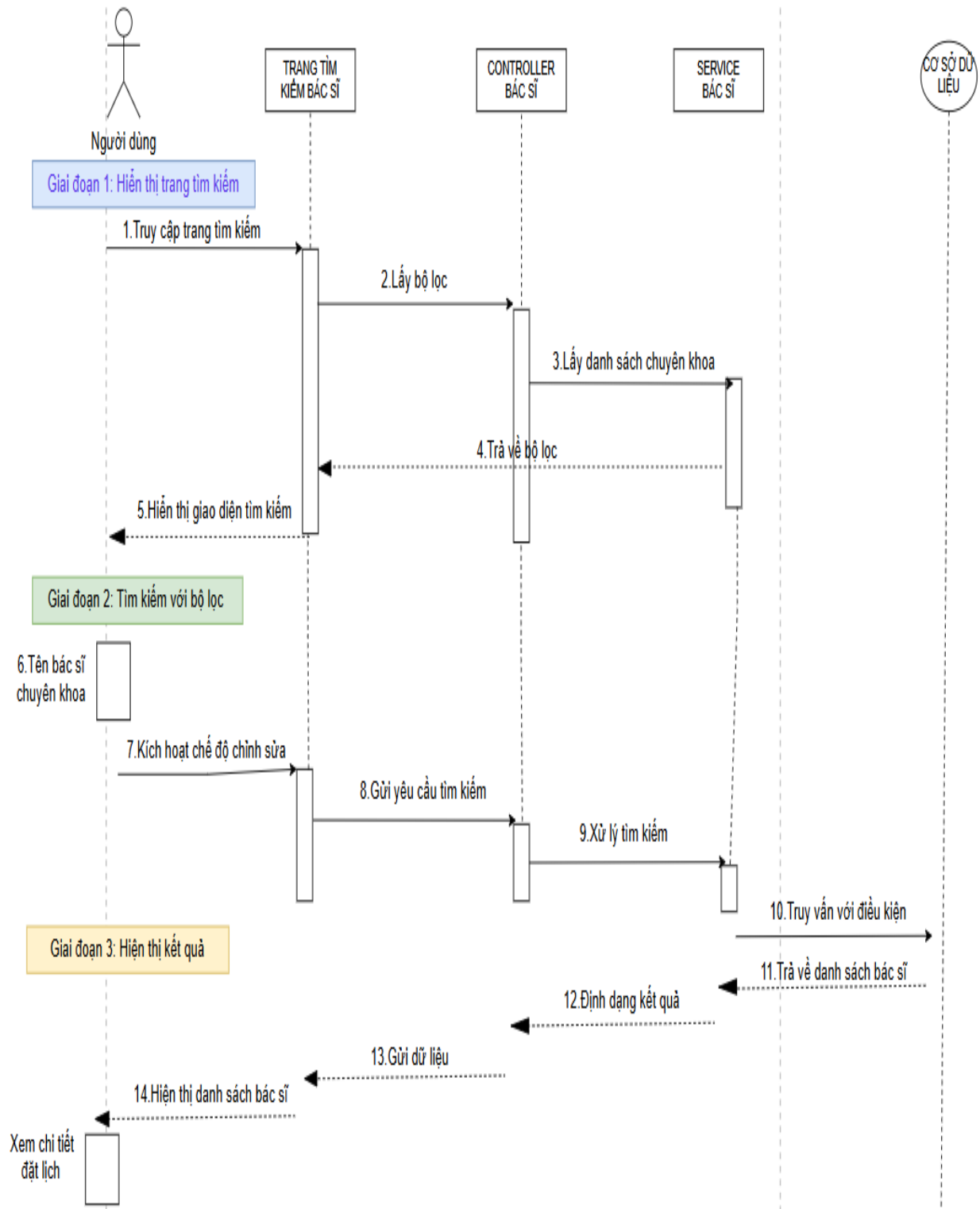
Hình 2.3 Biểu đồ tuần tự chức chỉnh sửa thông tin cá nhân

2.3.3 Biểu đồ tuần tự chức năng “ Xem thông tin dịch vụ khám ”



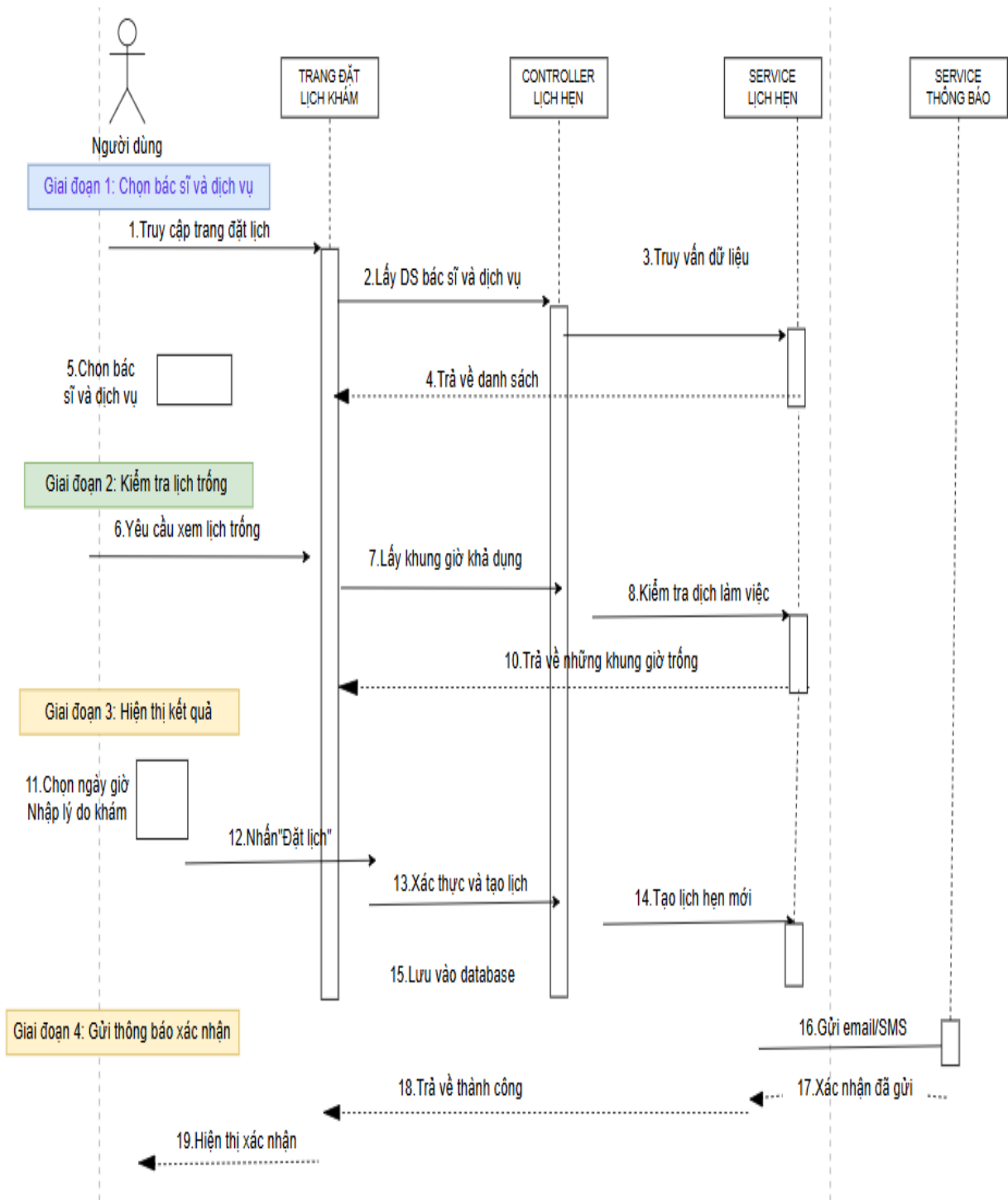
Hình 2. 4 Biểu đồ tuần tự chức xem thông tin dịch vụ khám

2.3.4 Biểu đồ tuần tự chức năng “Tìm kiếm bác sĩ”



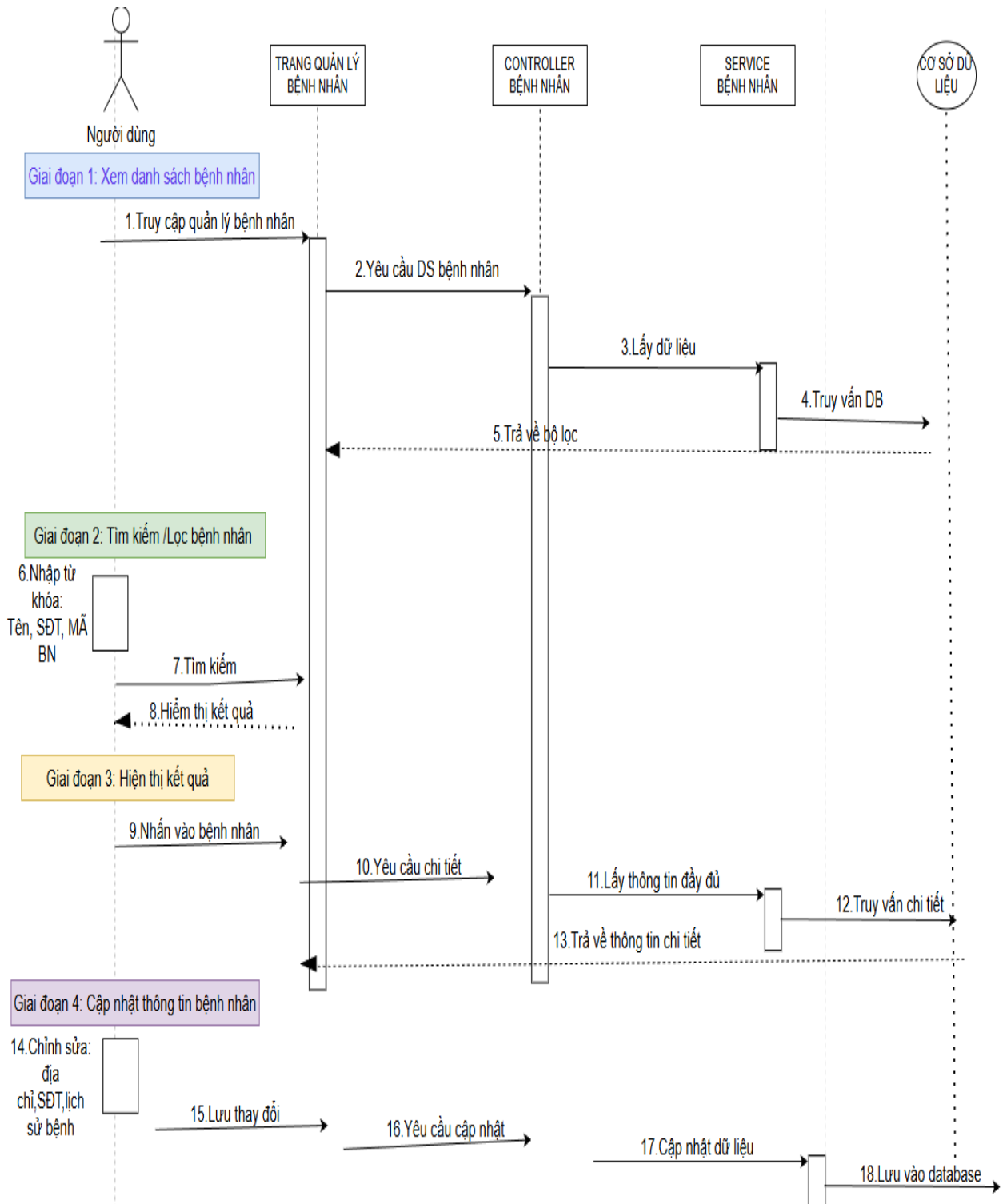
Hình 2. 5 Biểu đồ tuần tự chức năng tìm kiếm bác sĩ

2.3.5 Biểu đồ tuần tự chức năng “Đặt lịch khám”



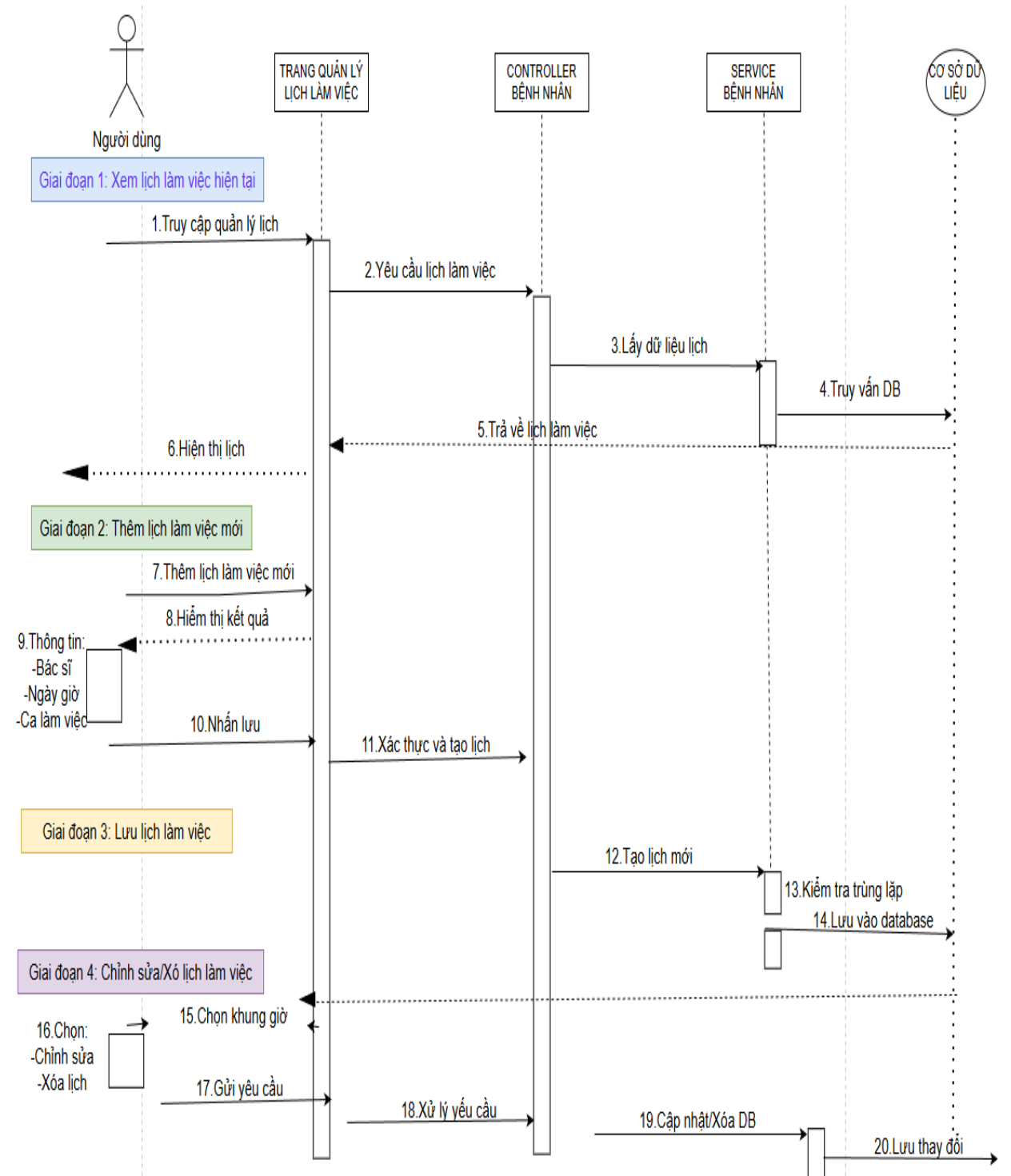
Hình 2. 6 Biểu đồ tuần tự chức năng đặt lịch khám

2.3.6 Biểu đồ tuần tự chức năng “Quản lý thông tin bệnh nhân”



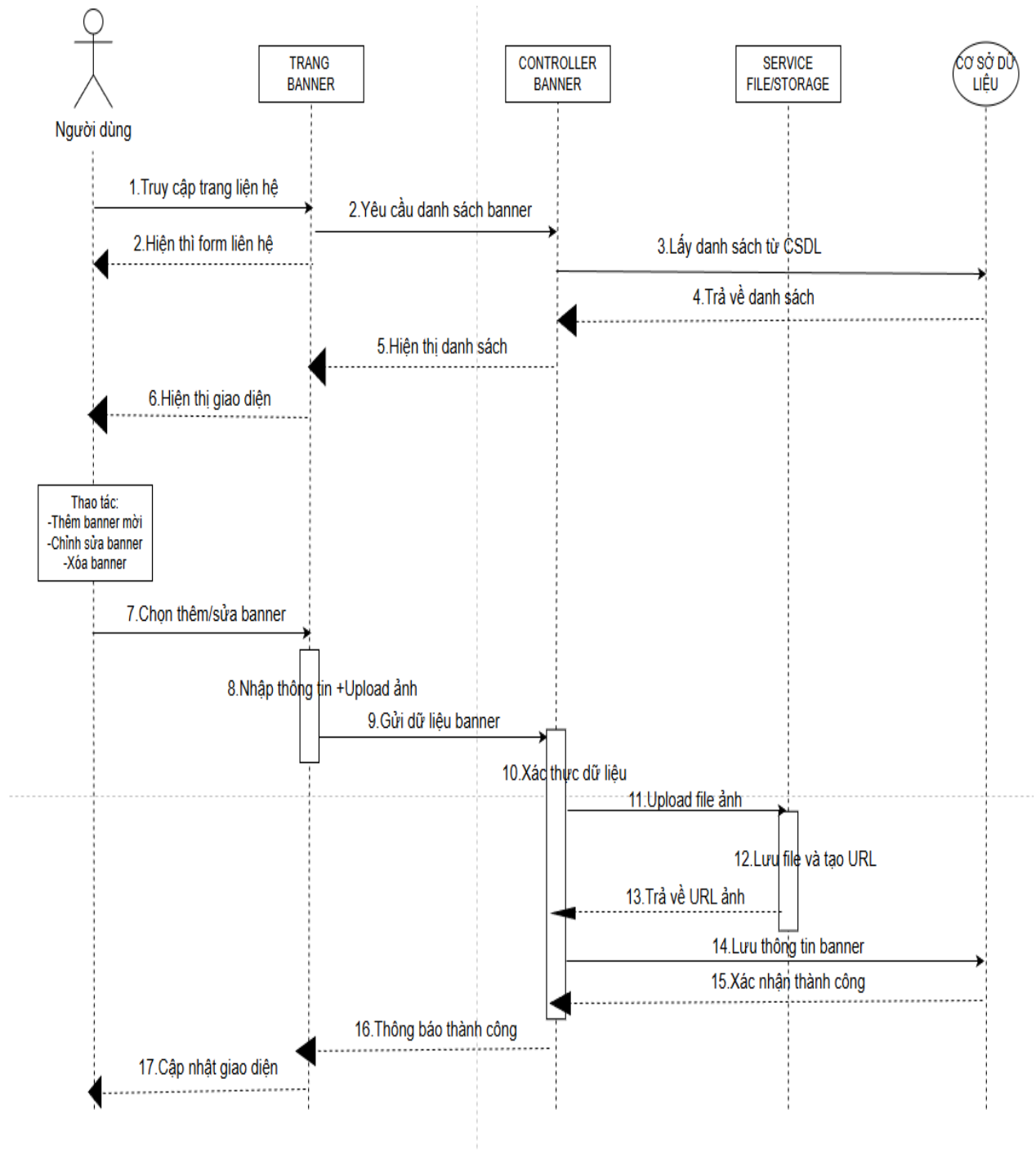
Hình 2.7 Biểu đồ tuần tự chức năng quản lý thông tin bệnh nhân

2.3.7 Quản lý lịch làm việc bác sĩ



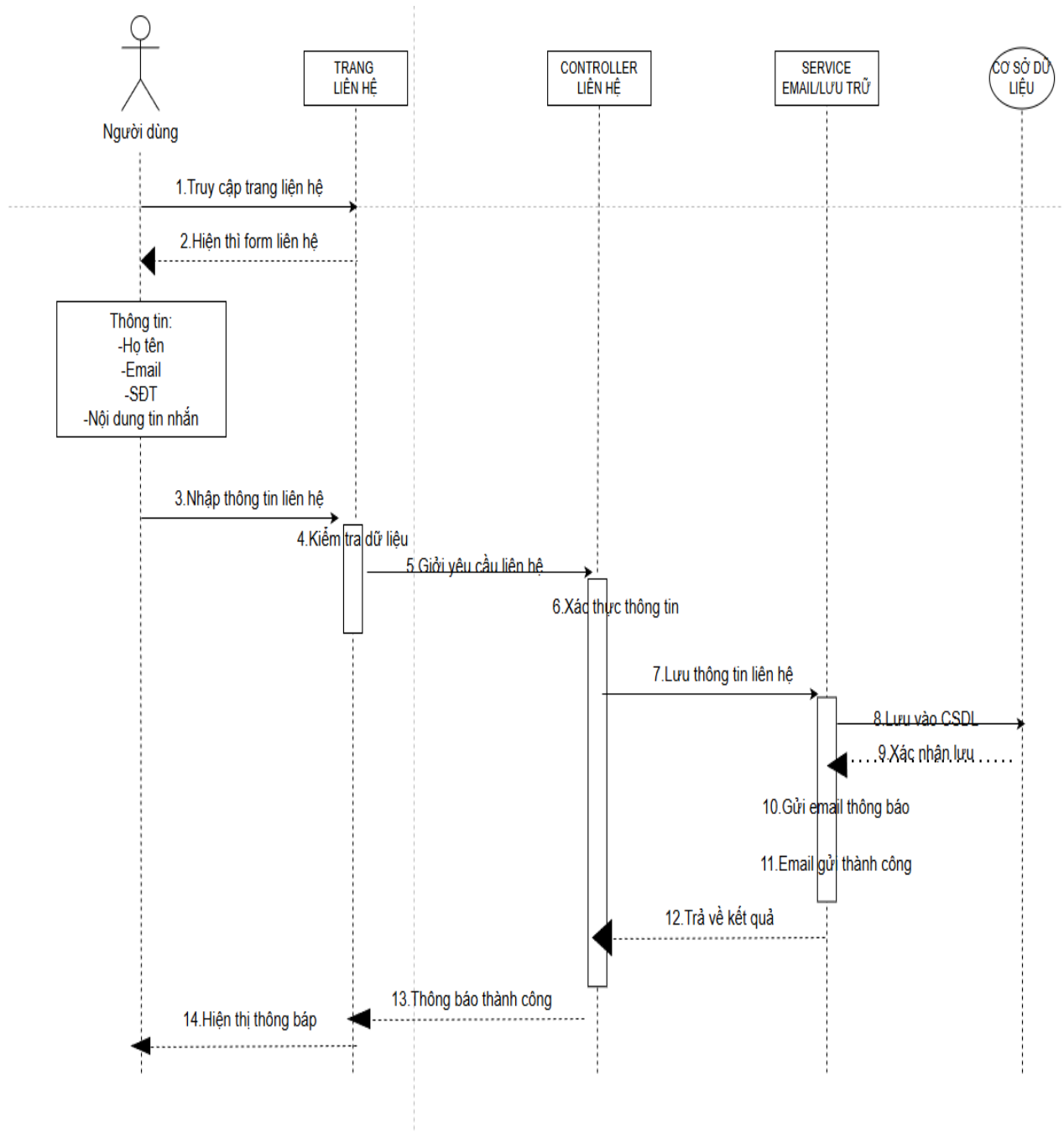
Hình 2. 8 Biểu đồ tuần tự chức năng quản lý việc làm việc bác sĩ

2.3.9 Quản lý banner



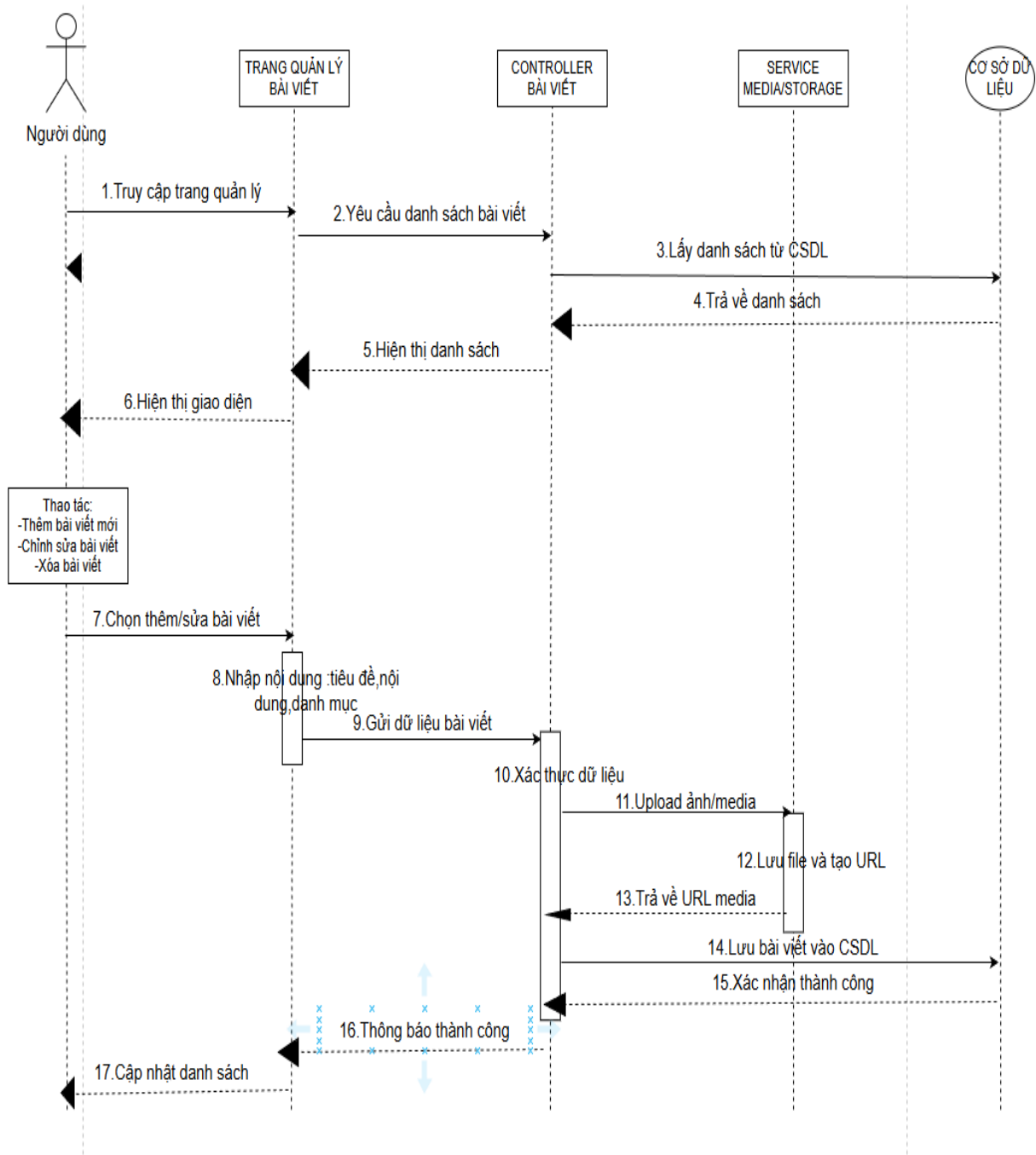
Hình 2. 9 Biểu đồ tuần tự chức năng quản lý banner

2.3.8 Quản lý liên hệ



Hình 2. 10 Biểu đồ tuần tự chức năng quản lý liên hệ

2.3.10 Quản lý bài viết



Hình 2. 11 Biểu đồ tuần tự chức năng quản lý bài viết

2.4 Cơ sở dữ liệu

2.4.1 Các thực thể có trong hệ thống

Bảng 2. 19 Bảng thực thể có trong hệ thống

	Mô tả
articles	Lưu trữ các bài viết, tin tức về nha khoa
bookings	Quản lý lịch hẹn khám của bệnh nhân
bot_knowledge	Lưu trữ kiến thức cho chatbot hỗ trợ
patients	Lưu trữ thông tin bệnh nhân (hồ sơ cá nhân, liên hệ, lịch sử khám)
doctors	Lưu thông tin bác sĩ (chuyên khoa, lịch làm việc, hồ sơ nghề nghiệp)
categories	Phân loại các bài viết

Mô tả về giá trị của các trường

Trường **categories** - Bảng phân loại các bài viết

Các trường dữ liệu thường có:

- **id**: Mã danh mục duy nhất
- **name**: Tên danh mục
- **slug**: URL thân thiện
- **description**: Mô tả danh mục
- **parent_id**: ID danh mục cha (cho danh mục con)
- **order**: Thứ tự hiển thị
- **status**: Trạng thái (active, inactive)
- **image**: Hình ảnh đại diện
- **created_at**: Thời gian tạo
- **updated_at**: Thời gian cập nhật

2.4.2 Chi tiết cho từng thực thể

Bảng 2.20 Thực thể articles

Tên cột	Kiểu dữ liệu	Ràng buộc	Mô tả
id	BIGINT UNSIGNED	PRIMARY KEY, AUTO_INCREMENT	ID bài viết
name	VARCHAR(255)	NOT NULL	Tiêu đề bài viết
slug	VARCHAR(255)	NOT NULL	URL thân thiện
view	INT	DEFAULT 0	Lượt xem
description	MEDIUMTEXT	NULL	Mô tả ngắn
image	VARCHAR(255)	NULL	Hình ảnh minh họa
content	TEXT	NULL	Nội dung bài viết
category_id	BIGINT UNSIGNED	FOREIGN KEY	ID danh mục
created_at	TIMESTAMP	NULL	Thời gian tạo
updated_at	TIMESTAMP	NULL	Thời gian cập nhật

Mô tả về giá trị của trường view

Trường **category_id** - Mã danh mục phân loại

Không có trạng thái cố định, là số nguyên dương tham chiếu đến bảng danh mục/categories

Ví dụ:

- 1, 2, 3, 4... (ID của danh mục cụ thể)
- Liên kết với bảng categories để phân loại bài viết/dịch vụ/sản phẩm

Bảng 2. 21 Thực thể bookings

Tên cột	Kiểu dữ liệu	Ràng buộc	Mô tả
id	BIGINT UNSIGNED	PRIMARY KEY, AUTO_INCREMENT	ID lịch hẹn
doctor_id	BIGINT UNSIGNED	FOREIGN KEY	ID bác sĩ
patient_id	BIGINT UNSIGNED	FOREIGN KEY	ID bệnh nhân
date_booking	DATE	NULL	Ngày hẹn
time_booking	VARCHAR(255)	NULL	Thời gian hẹn
status	TINYINT	DEFAULT 1	Trạng thái lịch hẹn
reason_other	TEXT	NULL	Lý do khám

Mô tả về giá trị của các trường

Trường **patient_id** - Mã định danh của bệnh nhân đặt lịch

Không có trạng thái cố định, là số nguyên dương tham chiếu đến bảng bệnh nhân/người dùng

Ví dụ:

- 1, 2, 3, 4... (ID của bệnh nhân cụ thể)
- Liên kết với bảng users/patients để lấy thông tin bệnh nhân

Bảng 2. 22 Thực thể bot_knowledge

Tên cột	Kiểu dữ liệu	Ràng buộc	Mô tả
id	BIGINT UNSIGNED	PRIMARY KEY, AUTO_INCREMENT	ID kiến thức

Tên cột	Kiểu dữ liệu	Ràng buộc	Mô tả
question	TEXT	NOT NULL	Câu hỏi
answer	TEXT	NOT NULL	Câu trả lời
confidence	INT	DEFAULT 0	Độ tin cậy

Mô tả về giá trị của các trường

Trường **confidence** - Độ tin cậy của câu trả lời

Không có trạng thái cố định, là số nguyên từ 0-100 thể hiện mức độ chắc chắn của câu trả lời

Ví dụ:

- 0-30: Độ tin cậy thấp
- 31-70: Độ tin cậy trung bình
- 71-100: Độ tin cậy cao
- Mặc định: 0

Bảng 2. 23 Thực thể chat_messages

Tên cột	Kiểu dữ liệu	Ràng buộc	Mô tả
id	BIGINT UNSIGNED	PRIMARY KEY, AUTO_INCREMENT	ID tin nhắn
user_id	VARCHAR(255)	NOT NULL	ID người dùng
message	TEXT	NOT NULL	Nội dung tin nhắn gửi đi
response	TEXT	NULL	Phản hồi từ bot/hệ thống
type	VARCHAR(255)	DEFAULT	Loại tin nhắn

Tên cột	Kiểu dữ liệu	Ràng buộc	Mô tả
is_bot	TINYINT(1)	DEFAULT	Nguồn tin nhắn

Mô tả về giá trị các của trường

Trường id - Mã định danh duy nhất của tin nhắn, tự động tăng

Không có trạng thái cố định, là số nguyên dương tự động sinh ra cho mỗi tin nhắn mới

Ví dụ: 1, 2, 3, 4...

Trường response - Phản hồi từ bot hoặc hệ thống

Không có trạng thái cố định, là văn bản tự do chứa câu trả lời tự động từ chatbot/hệ thống (không giới hạn độ dài)

Ví dụ:

- "Cảm ơn bạn đã liên hệ, chúng tôi sẽ phản hồi sớm"
- "Phòng khám mở cửa từ 8h-20h hàng ngày"
- NULL nếu chưa có phản hồi

Bảng 2. 24 Thực thể clinics

Tên cột	Kiểu dữ liệu	Ràng buộc	Mô tả
id	BIGINT UNSIGNED	PRIMARY KEY	ID phòng khám
name	VARCHAR(255)	NOT NULL	Tên phòng khám
email	VARCHAR(255)	NULL	Email liên hệ
phone	VARCHAR(255)	NULL	Số điện thoại
address	VARCHAR(255)	NULL	Địa chỉ đầy đủ
logo	VARCHAR(255)	NULL	Đường dẫn logo
link_website	VARCHAR(255)	NULL	URL website

Tên cột	Kiểu dữ liệu	Ràng buộc	Mô tả
description	VARCHAR(255)	NULL	Mô tả ngắn
contents	TEXT	NULL	Nội dung chi tiết

Mô tả về giá trị của các trường

Trường logo - Đường dẫn file logo của phòng khám

Không có trạng thái cố định, là chuỗi URL hoặc đường dẫn file ảnh logo (tối đa 255 ký tự)

Ví dụ:

- "/uploads/clinics/logo_123.png"
- "https://example.com/images/clinic-logo.jpg"
- NULL nếu chưa có logo

Trường contents - Nội dung chi tiết về phòng khám

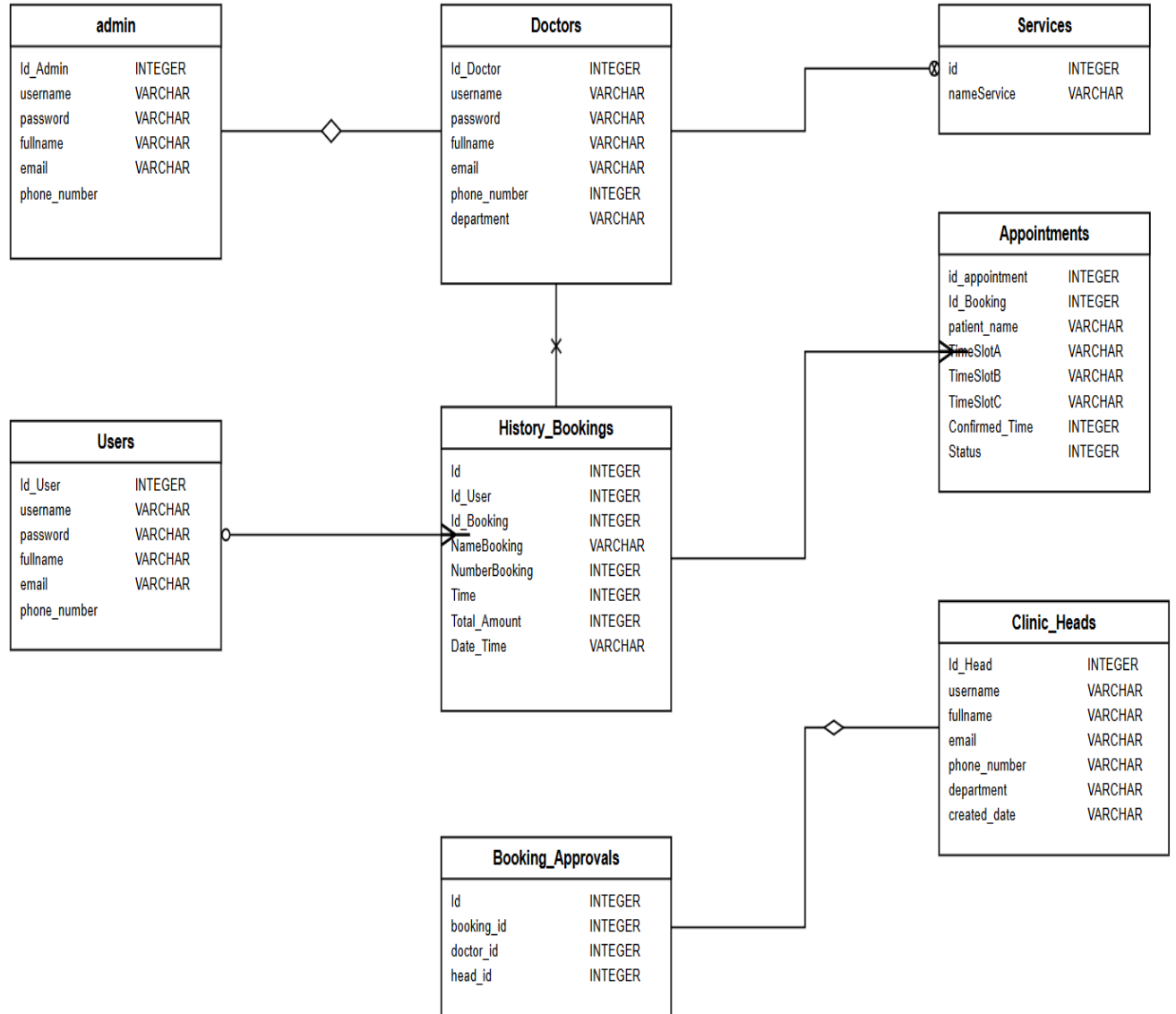
- Không có trạng thái cố định, là văn bản tự do chứa thông tin giới thiệu đầy đủ về phòng khám (không giới hạn độ dài)
- Ví dụ: mô tả lịch sử, đội ngũ bác sĩ, trang thiết bị, dịch vụ nổi bật, chứng nhận...

Bảng 2. 25 Thực thể categories

Tên cột	Kiểu dữ liệu	Ràng buộc	Mô tả
id	BIGINT UNSIGNED	PRIMARY KEY	ID danh mục
name	VARCHAR(255)	NOT NULL	Tên danh mục
parent_id	INT	DEFAULT 0	ID danh mục cha (nếu có)
slug	VARCHAR(255)	NOT NULL	URL thân thiện
description	VARCHAR(255)	NULL	Mô tả ngắn

2.4.3 Sơ đồ ERD tổng thể

Hình 2. 12 Sơ đồ ERD tổng thể



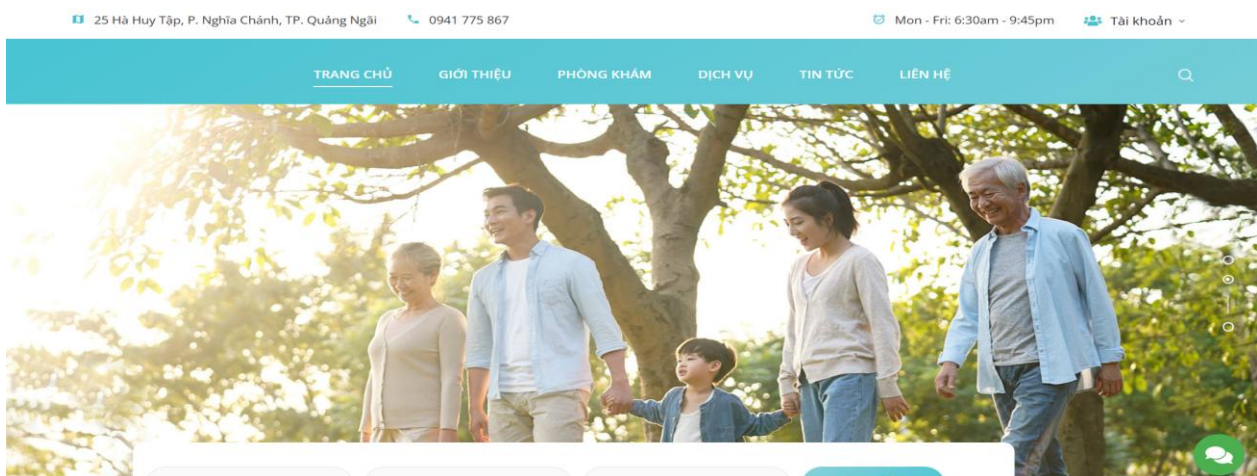
CHƯƠNG 3 . KẾT QUẢ THỰC NGHIỆM ĐỀ TÀI

3.1 Môi trường

- ❖ Đề tài được phát triển trên môi trường phát triển:
 - Vi xử lý (processor): Processor 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz
 - Ram: 16,0 GB
 - Storage 238 GB SSD Colorful SL500 256GB, 238 GB SSD NVMe Lexar SSD NM620 256GB
 - Graphics Card NVIDIA GeForce MX330 (2 GB), Intel(R) Iris(R) Xe Graphics (128 MB)
- ❖ Công cụ lập trình
 - Phần mềm viết code: Visual Studio
 - Cơ sở dữ liệu: PhpMysql
 - Ngôn ngữ lập trình: Laravel

3.2 Giao diện kết quả

3.2.1 Giao diện trang chủ



Hình 3.1 Giao diện trang chủ

3.2.2 Giao diện đăng nhập

TRANG CHỦ GIỚI THIỆU PHÒNG KHÁM DỊCH VỤ TIN TỨC LIÊN HỆ

ĐĂNG NHẬP

Đăng nhập để nhận tư vấn và hưởng nhiều ưu đãi hơn

Email đăng nhập: dunghuynh@gmail.com

Mật khẩu đăng nhập: *****

ĐĂNG NHẬP

Hình 3.2 Giao diện đăng nhập

3.2.3 Giao diện đăng ký

ĐĂNG KÝ

Đăng ký tài khoản để nhận tư vấn và hưởng nhiều ưu đãi hơn. Chúng tôi có thể tư vấn chi tiết hơn cho bạn

Họ và tên (*):

Mật khẩu đăng nhập (*):

Ngày sinh (*):

Số điện thoại:

Cần cước công dân:

Email đăng nhập (*):

Nhập lại mật khẩu đăng nhập (*):

Giới tính (*):

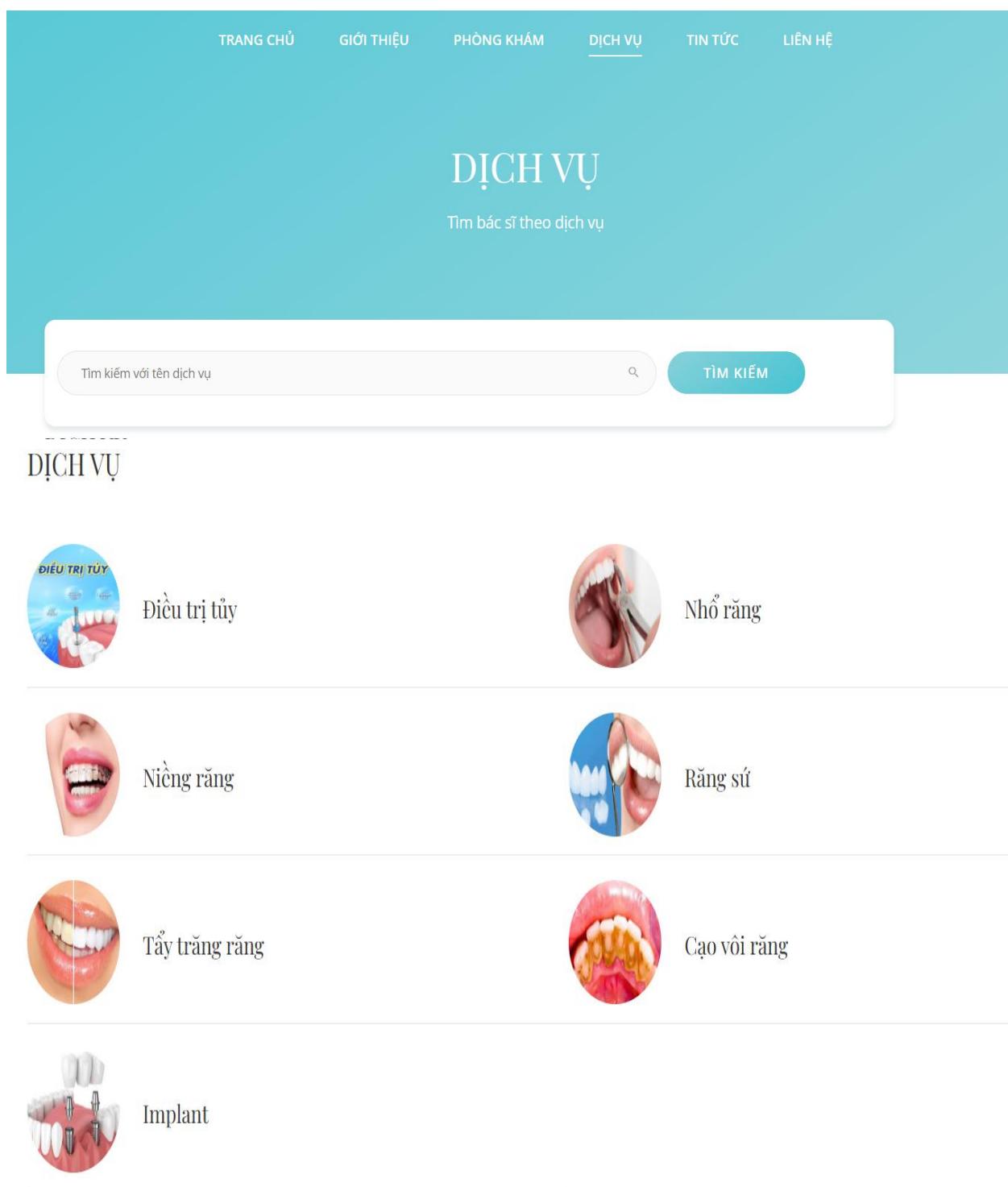
Địa chỉ:

Số thẻ bảo hiểm:

ĐĂNG KÝ

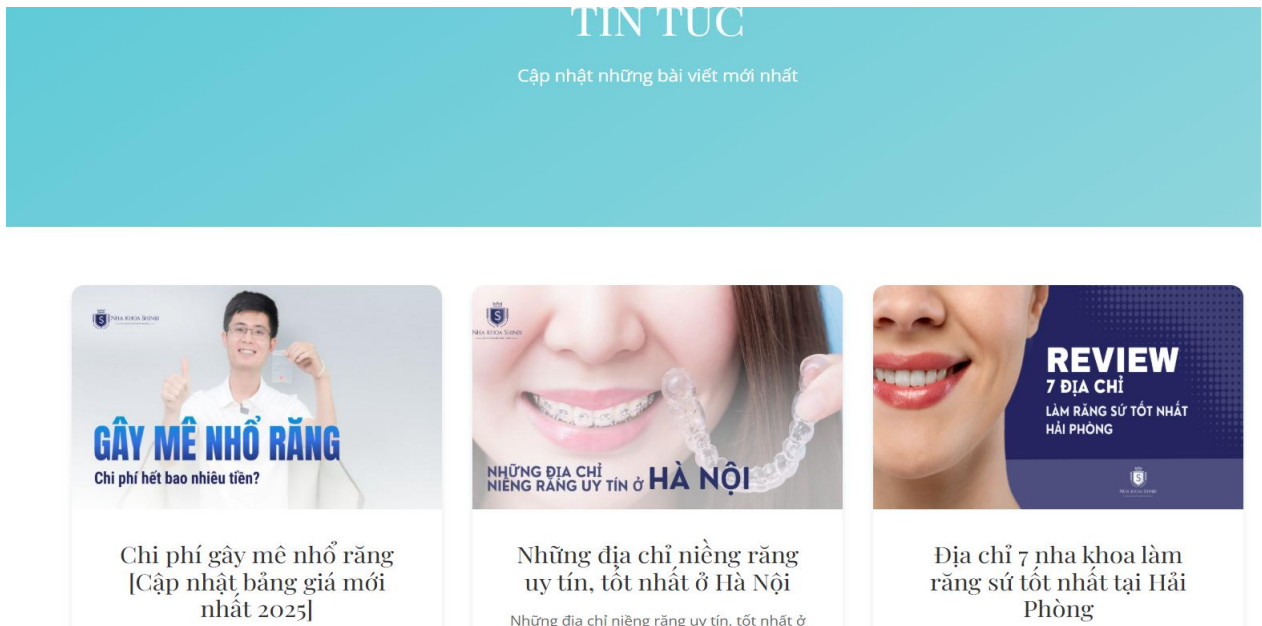
Hình 3.3 Giao diện đăng ký

3.2.4 Giao diện các dịch vụ phòng khám



Hình 3.4 Giao diện dịch vụ

3.2.5 Giao diện trang tin tức



Hình 3.5 Giao diện trang tin tức

3.2.6 Giao diện trang liên hệ

The screenshot displays the 'Liên hệ' (Contact) page. It includes a map of Hanoi showing the location of 'P. Hào Nam' and a contact form below it.

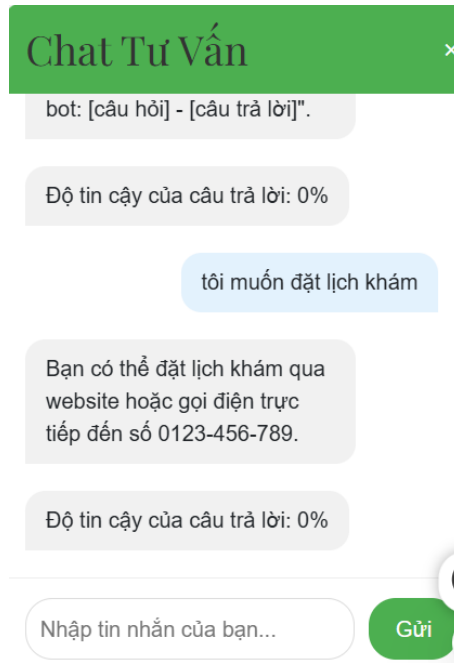
Map: Shows the location of 'P. Hào Nam' in Hanoi, with various landmarks and streets labeled.

Contact Form:

- Trụ sở tại Hà Nội:** P. Hào Nam, Đống Đa, Hà Nội.
- Thời gian làm việc:** Mon - Fri: 6:30am - 9:45pm; Mon - Fri: 8:30am - 7:45pm.
- Form fields:**
 - Họ và tên (Last name and first name)
 - Email
 - Số điện thoại (Phone number)
 - Nội dung liên hệ (Contact content)
- Buttons:** GỬI LIÊN HỆ (Send contact).

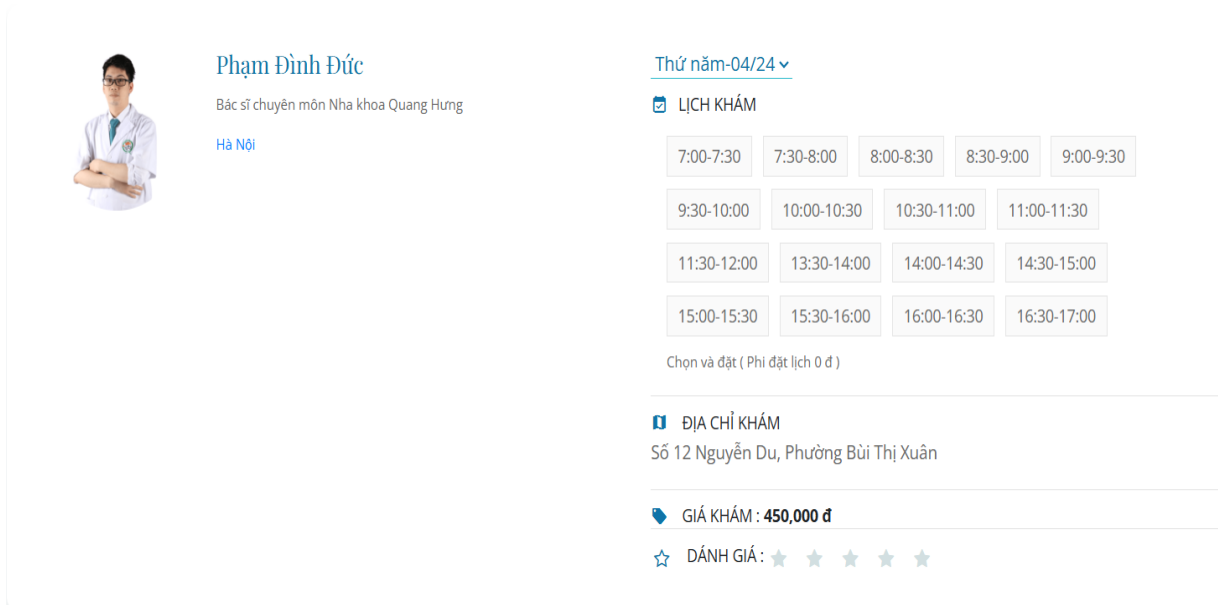
Hình 3.6 Giao diện liên hệ

3.2.7 Giao diện chatbox



Hình 3.7 Giao diện chatbox

3.2.8 Giao diện đặt lịch khám



Hình 3.8 Giao diện đặt lịch khám

3.2.11 Giao diện trang quản lý bác sĩ

Trang chủ / Người dùng / Danh sách

Form tìm kiếm

Mã bác sĩ Tên bác sĩ Chọn Phòng khám

Trạng thái

STT	Mã bác sĩ	Họ tên	Phòng khám	Dịch vụ	Giới tính	Chức vụ	Chức danh	Vai trò	Trạng thái	Hành động
1	BS000077	Đặng Vũ Hải	Phòng điều trị tùy	Điều trị tùy	Nam	Tiến sĩ	Bác sĩ chuyên khoa	Bác sĩ	Hoạt động	Sửa Xóa
2	BS000076	Nguyễn Hoàng Vũ	Phòng Implant	Implant	Nam	Bác sĩ	Bác sĩ chuyên khoa	Bác sĩ	Hoạt động	Sửa Xóa
3	QT000075	Trần Thị Dung	Phòng cạo vôi răng	Cạo vôi răng	Nữ	Bác sĩ	Bác sĩ chuyên khoa	Bác sĩ	Hoạt động	Sửa Xóa
4	BS000074	Nguyễn Trọng Hiếu	Phòng tẩy trắng răng	Tẩy trắng răng	Nam	Tiến sĩ	Bác sĩ chuyên khoa	Bác sĩ	Hoạt động	Sửa Xóa
5	BS000073	Lê Thành Hải	Phòng răng sứ	Răng sứ	Nam	Tiến sĩ	Bác sĩ chuyên khoa	Bác sĩ	Hoạt động	Sửa Xóa

Hình 3.11 Giao diện trang quản lý bác sĩ

3.2.12 Giao diện trang quản lý lịch làm việc

Trang chủ / Lịch làm việc / Danh sách

Form tìm kiếm

Mã bác sĩ Tên bác sĩ mm/dd/yyyy mm/dd/yyyy

Trạng thái

STT	Mã bác sĩ	Tên bác sĩ	Ngày đăng ký	Lịch làm việc	Trạng thái	Hành động
1	BS000014	Phạm Đình Đức	2025-05-01	7:00-7:30 7:30-8:00 8:00-8:30 8:30-9:00 9:00-9:30 9:30-10:00 10:00-10:30 10:30-11:00 11:00-11:30 11:30-12:00 13:30-14:00 14:00-14:30 14:30-15:00 15:00-15:30 15:30-16:00 16:00-16:30 16:30-17:00	Xuất bản	Sửa Xóa
2	BS000014	Phạm Đình Đức	2025-04-28	7:00-7:30 7:30-8:00 8:00-8:30 8:30-9:00 9:00-9:30 9:30-10:00 10:00-10:30 10:30-11:00 11:00-11:30 11:30-12:00 13:30-14:00 14:00-14:30 14:30-15:00 15:00-15:30 15:30-16:00 16:00-16:30 16:30-17:00	Xuất bản	Sửa Xóa

Hình 3.12 Giao diện trang quản lý lịch làm việc

3.2.13 Giao diện trang quản lý đặt lịch khám

[Trang chủ](#) / [Danh sách đặt lịch khám](#) / [Danh sách](#)

From tìm kiếm

Mã bệnh nhân

Tên bệnh nhân

mm/dd/yyyy

mm/dd/yyyy

Trạng thái

Tìm kiếm

STT	Thông tin bác sĩ	Thông tin bệnh nhân	Ngày đặt	Thời gian	Giá	Kết quả	Trạng thái
1	Mã bác sĩ: BS000014 Bác sĩ: Phạm Đình Đức Phòng khám: Nha khoa quốc tế Việt Sing dịch vụ: Niềng răng Địa chỉ khám: Số 12 Nguyễn Du, Phường Bùi Thị Xuân	Bệnh nhân: DƯƠNG THỊ KIM CÚC Giới tính: Nam Phone: 0344200402 Ngày sinh: 2002-02-22 Đặt cho: Đặt cho mình	2025-04-24	10:00-10:30	450,000 đ		Tiếp nhận
	Mã bác sĩ: BS000014 Bác sĩ: Phạm Đình Đức	Bệnh nhân: DƯƠNG THỊ KIM CÚC Giới tính: Nữ					

Hình 3.13 Giao diện trang quản lý đặt lịch khám

3.2.14 Giao diện trang quản lý tin tức

[Trang chủ](#) / [Bài viết](#) / [Danh sách](#)

Tạo mới

STT	Tiêu đề	Hình ảnh	Ngày đăng	Hành động
1	Chi phí gây mê nhổ răng [Cập nhật bảng giá mới nhất 2025]		2025-04-20 16:09:44	
2	Những địa chỉ niềng răng uy tín, tốt nhất ở Hà Nội		2025-04-20 16:07:59	
3	Địa chỉ 7 nha khoa làm răng sứ tốt nhất tại Hải Phòng		2025-04-20 16:06:56	

Hình 3.14 Giao diện trang quản lý tin tức

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết quả đạt được

Sau một thời gian tập trung triển khai đề tài, em đã hoàn thành được “Xây dựng Website đặt lịch khám cho phòng khám nha khoa” với giao diện đẹp, dễ sử dụng và thân thiện với người dùng. Các chức năng hoạt động tốt, đáp ứng đúng quy trình nghiệp vụ đặt lịch của phòng khám. Giao diện gần gũi, thao tác thuận lợi và trải nghiệm người dùng được đảm bảo. Hệ thống cũng đạt mức độ bảo mật dữ liệu tương đối an toàn.

Sau khi hoàn thành được đề tài “Xây dựng Website đặt lịch khám cho phòng khám nha khoa”, em đã củng cố thêm các kỹ năng nền tảng trong việc xây dựng một website bằng PHP và MySQL, đồng thời áp dụng hiệu quả các kiến thức HTML, CSS và JavaScript vào phát triển ứng dụng web. Bên cạnh đó, em cũng học hỏi thêm nhiều kiến thức chuyên ngành mới thông qua quá trình tìm hiểu tài liệu, lập trình và hoàn thiện các chức năng của hệ thống. Nhờ vậy, kỹ năng tự học và khả năng vận dụng công nghệ mới của em được nâng cao đáng kể.

Trong quá trình thực hiện và hoàn thành đồ án, em đã tích lũy thêm nhiều kinh nghiệm quý báu trong việc xây dựng website nói riêng và phát triển phần mềm nói chung. Đây sẽ là nền tảng quan trọng giúp em nâng cao trình độ chuyên môn và tự tin hơn khi làm việc thực tế tại các doanh nghiệp trong tương lai.

Những mặt còn hạn chế

- Các chức năng còn đơn giản, chưa có chức năng để người dùng chia sẻ sản phẩm.
- Web chưa hỗ trợ thanh toán trực tuyến cho người dùng.
- Tốc độ xử lý chưa cao, vẫn còn một số lỗi phát sinh khi vận hành hệ thống.

Đề xuất giải pháp khắc phục

Nghiên cứu thiết kế và xây dựng các chức năng: chia sẻ sản phẩm, thanh toán trực tuyến cho người dùng trong tương lai.

TÀI LIỆU THAM KHẢO

- [1] Thạc Bình Cường, *Phân tích và thiết kế hệ thống thông tin*, nhà xuất bản Khoa học và Kỹ thuật, 2002.
- [2] Phạm Hữu Khang, Hoàng Đức Hải, *Giáo trình nhập môn PHP và MySQL – Xây dựng ứng dụng web (tủ sách để học)*, nhà xuất bản Phương Đông, 2005.
- [3] <https://en.wikipedia.org/wiki/PHP>
- [4] <https://en.wikipedia.org/wiki/Laravel>
- [5] <https://en.wikipedia.org/wiki/MySQL>
- [6] <https://vi.wikipedia.org/wiki/CSS>
- [7] <https://vi.wikipedia.org/wiki/HTML>
- [8] https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm