

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**  
**KHOA CÔNG NGHỆ PHẦN MỀM**



**BÁO CÁO ĐỒ ÁN 2**  
**XÂY DỰNG WEBSITE ĐẶT LỊCH KHÁM BỆNH ONLINE**

**Giảng viên hướng dẫn: ThS. Trần Thị Hồng Yến**

**Sinh viên thực hiện:**

**Nguyễn Nam Hùng - 17520544**

**TP. Hồ Chí Minh, tháng 12 năm 2022**

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**  
**KHOA CÔNG NGHỆ PHẦN MỀM**



**BÁO CÁO ĐỒ ÁN 2**  
**XÂY DỰNG WEBSITE ĐẶT LỊCH KHÁM BỆNH ONLINE**

**Giảng viên hướng dẫn: ThS. Trần Thị Hồng Yến**

**Sinh viên thực hiện:**

**Nguyễn Nam Hùng - 17520544**

**TP. Hồ Chí Minh, tháng 12 năm 2022**

**TRƯỜNG ĐẠI HỌC  
CÔNG NGHỆ THÔNG TIN****Độc Lập – Tự Do – Hạnh Phúc****ĐỀ CƯƠNG CHI TIẾT**

<b>Tên đề tài: Website đặt lịch khám bệnh online</b>
<b>Cán bộ hướng dẫn: ThS. Trần Thị Hồng Yến</b>
<b>Thời gian thực hiện: Từ ngày 2/9/2022 đến ngày 31/12/2022</b>
<b>Sinh viên thực hiện:</b>  Nguyễn Nam Hùng - 17520544

**Nội dung đề tài:**

- **Mục tiêu:**
- Xây dựng hệ thống bao gồm những chức năng cơ bản của một trang web đặt lịch khám bệnh online như: Đặt lịch khám bệnh, tra cứu thông tin bác sĩ, chuyên khoa, cơ sở y tế, quản lý bác sĩ, quản lý kế hoạch khám bệnh cũng như bệnh nhân
- Xây dựng hệ thống cho phép người dùng có thể tìm kiếm, tra cứu thông tin một cách nhanh chóng và trực quan
- Xây dựng trang web thông tin có các tin tức về cơ sở y tế, bác sĩ, chuyên khoa

- Xây dựng hệ thống với các công nghệ mới như NodeJS, ReactJS, MySQL. Em chọn những công nghệ này vì chúng có tính bao quát rộng hơn, NodeJS hỗ trợ bất đồng bộ giúp nhiều phương thức chạy đồng thời, MySQL là cơ sở dữ liệu tốc độ cao, ổn định hoạt động trên nhiều hệ điều hành, cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Đặc biệt, hệ quản trị cơ sở dữ liệu MySQL hoàn toàn miễn phí nên người dùng có thể thoải mái tải về từ trang chủ
- Đối tượng: Những người có nhu cầu khám bệnh online
- Tính mới/ khác biệt về chức năng:
- Thêm ý tưởng:

Các trang web sẽ hỗ trợ thêm bác sĩ, giúp bác sĩ dễ dàng quản lý lịch làm việc cũng như trao đổi với người dùng dễ dàng hơn

Chúng em sẽ thêm tính năng quản lý thông tin các phòng khám, cơ sở y tế, chuyên khoa giúp cho người dùng dễ dàng tra cứu.

- Chọn lọc các chức năng:

Khảo sát qua các trang khám bệnh online của Việt Nam hiện nay như [umc.medpro.com.vn](http://umc.medpro.com.vn), [bookingcare](http://bookingcare.com),... nhóm nhận thấy:

Các trang web khám bệnh online đều trình bày theo hình thức đơn giản, khiến người dùng dễ dàng sử dụng khi lần đầu vào trang web. Cũng vì thế mà nhóm cũng đã thiết kế giao diện gọn gàng không rối mắt.

Em sẽ hoàn thành các chức năng cần thiết như hỗ trợ tư vấn, đặt lịch, trao đổi trực tiếp với bác sĩ, quản lý lịch khám bệnh

Ngoài ra còn các chức năng cơ bản của các trang web như tìm kiếm, xem thông tin chi tiết

- Phạm vi:
- Phạm vi địa lý: Ứng dụng sử dụng tại Việt Nam.
- Phạm vi chức năng:

- Quản lý tài khoản
- Quản lý bác sĩ, chuyên khoa, cơ sở y tế
- Tra cứu các thông tin bác sĩ, chuyên khoa, cơ sở y tế
- Quản lý lịch khám bệnh của bác sĩ và bệnh nhân
- Quản lý tìm kiếm thông tin theo bác sĩ, bệnh nhân
- Phương pháp thực hiện:
- Làm việc cá nhân, sử dụng request để lấy thông tin từ những trang web cụ thể, lưu xuống database và xuất lên cho người dùng thấy.
- Xây dựng database với MySQL
- Xây dựng hệ thống Restful API với NodeJS và express.
- Công nghệ sử dụng:
  - ReactJS.
  - MySQL
  - NodeJS
  - ExpressJS
- **Lí do sử dụng các công nghệ trên:**
  - NodeJS là một backend bất đồng bộ, khiến các chức năng chạy xong xong với nhau, điều này giúp ứng dụng chạy mượt hơn, không có những khoảng khắc ứng dụng gần như không làm gì.
  - MySQL là cơ sở dữ liệu tốc độ cao, ổn định hoạt động trên nhiều hệ điều hành, cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Đặc biệt, hệ quản trị cơ sở dữ liệu MySQL hoàn toàn miễn phí nên người dùng có thể thoải mái tải về từ trang chủ.
  - ReactJS là một frontend được nhiều phần mềm sử dụng vì tính phong phú và đa dạng, được nhiều thư viện hỗ trợ, nên có thể tùy biến phần frontend trở nên đẹp hơn mà không thô cứng.
  - ExpressJS là công nghệ sử dụng ở dưới phần backend NodeJS, giúp xây dựng các API để gọi tới MySQL

- Postman là một chương trình ngoài để kiểm tra các API một cách độc lập mà không cần phải thao tác trên ứng dụng.
- Visual Code là phần mềm để viết và thực thi các dòng lệnh.

**Kết quả mong đợi:**

- Hoàn thiện ứng dụng web đặt lịch khám bệnh online đáp ứng đủ các nhu cầu cơ bản.
- Giao diện dễ thao tác, đăng kí lịch khám bệnh, hiển thị thông tin cho người dùng.

**Định hướng phát triển:**

Mở rộng thêm các trang web khác, thêm các tính năng mới tiện lợi cho người dùng

### Kế hoạch thực hiện

STT	Công việc	Thời gian thực hiện	Phân công
<b>Phase 1: Tìm hiểu và xây dựng ứng dụng web</b>		<b>4 tuần</b>	
<b>1</b>	<b>Sprint 1: Tìm hiểu các công nghệ sử dụng</b>		
1.1	<ul style="list-style-type: none"> <li>- NodeJS</li> <li>- ReactJS</li> <li>- Express</li> <li>- MySQL</li> <li>- Cách xây dựng các tính năng của một trang web</li> </ul>	Tuần 1 – 2 (21/9/2022 – 06/10/2022)	
<b>2</b>	<b>Sprint 2: Xây dựng ứng dụng cơ bản</b>		
2.1	- Xây dựng ứng dụng cơ bản để hiểu về luồng chạy của các ngôn ngữ, kết hợp NodeJS và ReactJS	Tuần 3 – 4 (07/10/2022 – 20/10/2022)	
<b>Phase 2: Xây dựng ứng dụng thu nhập và chia sẻ thông tin của các trang web</b>		<b>8 tuần</b>	
<b>3</b>	<b>Sprint 3: Xây dựng RESTful API</b>		

3.1	<ul style="list-style-type: none"> <li>- Phân tích hệ thống báo cáo và soạn báo cáo</li> <li>- Thiết kế Database</li> </ul>	Tuần 5 – 10 (20/10/2022 – 29/11/2022)	
<b>4</b>	<b>Sprint 4: Xây dựng Frontend với ReactJS</b>		
4.1	<ul style="list-style-type: none"> <li>- Xây dựng giao diện với ReactJS</li> <li>- Gọi API từ Backend lên Frontend để sử dụng</li> <li>- Xây dựng trang chi tiết quản lý</li> <li>- Hoàn thiện giao diện đặt lịch và điền thông tin Xác nhận</li> </ul>	Tuần 5 – 10 (29/11/2022 – 24/12/2022)	
<b>Phase 3: Kiểm thử và đánh giá</b>		<b>1 tuần</b>	
<b>5</b>	<b>Sprint 5: Kiểm thử</b>		
5.1	<ul style="list-style-type: none"> <li>- Kiểm thử</li> <li>- Đánh giá</li> <li>- Hoàn thiện báo cáo</li> </ul>	Tuần 11 - 12 (24/12/2022 – 31/12/2022)	
<b>Xác nhận của CBHD</b> (Ký tên và ghi rõ họ tên)   <b>ThS. Trần Thị Hồng Yến</b>		<b>TP. HCM, ngày 05 tháng 09 năm 2022</b>  <b>Sinh viên</b> (Ký tên và ghi rõ họ tên)   <b>Nguyễn Nam Hùng</b>	



## **LỜI CẢM ƠN**

Đề tài “Website đặt lịch khám bệnh ” là nội dung em chọn để thực hiện Đồ Án 2 theo học bậc đại học chuyên ngành Công nghệ Phần mềm tại Trường Đại học Công nghệ thông tin – Đại học Quốc gia TP.HCM.

Để hoàn thành tốt đồ án này, lời đầu tiên em xin chân thành gửi lời cảm ơn sâu sắc đến quý Thầy, Cô Khoa Công nghệ Phần mềm – Trường Đại học Công nghệ thông tin – Đại học Quốc gia TP.HCM đã tận tình giảng dạy, chỉ bảo em trong suốt quá trình học tập.

Em xin chân thành biết ơn sâu sắc đến cô Trần Thị Hồng Yến, người đã dành nhiều thời gian hướng dẫn, giúp em hoàn thành đồ án này.

Mặc dù em đã cố gắng, nỗ lực hết mình để hoàn thành tốt đồ án tốt nghiệp, song do thời gian có hạn cùng với những hạn chế về kinh nghiệm cũng như kỹ năng của bản thân nên không tránh khỏi những sai sót trong quá trình thực hiện đồ án. Vì vậy, em rất mong nhận được sự thông cảm và những ý kiến đóng góp, chỉ bảo từ quý thầy cô, bạn bè để em có thể khắc phục và hoàn thiện bản thân mình hơn trong quá trình làm việc sau này.

Em xin chân thành cảm ơn!

**Sinh viên thực hiện**

**Nguyễn Nam Hùng**

## NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

**Giảng viên hướng dẫn**

**Trần Thị Hồng Yến**

## MỤC LỤC

<b>1</b>	<b>Chương 1: GIỚI THIỆU TỔNG QUAN ĐỀ TÀI.....</b>	<b>17</b>
1.1.	Lý do chọn đề tài.....	17
1.2.	Mục tiêu đề tài.....	18
1.3.	Đối tượng và phạm vi nghiên cứu.....	18
1.3.1.	Đối tượng nghiên cứu.....	18
1.3.2.	Phạm vi nghiên cứu.....	18
1.4.	Phương pháp nghiên cứu.....	19
1.5.	Tiêu chí và chức năng của đề tài.....	19
1.5.1.	Bệnh nhân. ....	19
1.5.2.	Bác sĩ.....	19
1.5.3.	Quản trị viên.....	20
<b>2.</b>	<b>CHƯƠNG 2: TỔNG QUAN VỀ CÔNG NGHỆ SỬ DỤNG.....</b>	<b>21</b>
2.1.	NodeJS.....	21
2.1.1.	Giới thiệu về NodeJS.....	21
2.1.2.	Tại sao nên sử dụng NodeJS.....	21
2.1.3.	Áp dụng NodeJS vào đề tài.....	22
2.2.	ExpressJS.....	23
2.2.1.	Giới thiệu về ExpressJS .....	23
2.2.2.	Tại sao nên sử dụng ExpressJS .....	24
2.2.3.	Áp dụng ExpressJS vào đề tài.....	24
2.3.	MySQL.....	25
2.3.1.	Giới thiệu về MySQL .....	25
2.3.2.	Tại sao nên sử dụng hệ quản trị cơ sở dữ liệu MySQL.....	25
2.3.3.	Áp dụng MySQL vào đề tài .....	26
2.4.	ReactJS.....	26
2.4.1.	Giới thiệu về ReactJS .....	26
2.4.2.	Tại sao nên sử dụng ReactJS.....	27
<b>3.</b>	<b>Chương 3. PHÁT BIỂU BÀI TOÁN .....</b>	<b>29</b>
3.1.	Giới thiệu hệ thống website “Đặt lịch khám bệnh online ” .....	29
<b>4.</b>	<b>CHƯƠNG 4 : PHÂN TÍCH HỆ THỐNG ĐẶT LỊCH KHÁM CHỮA BỆNH DOCTOR CARE .....</b>	<b>30</b>
4.1.	Khảo sát .....	30
4.2.	Phân tích hệ thống.....	31
4.2.1.	Đặc tả hệ thống .....	31
4.2.2.	Yêu cầu chức năng .....	31
4.2.3.	Yêu cầu phi chức năng.....	32
4.3.	User case.....	34

4.3.1.	Danh sách tác nhân .....	34
4.3.2.	Danh sách Use Case.....	34
4.3.3.	Biểu đồ UC tổng quát.....	36
4.3.4.	Đặc tả Use Case .....	42
4.4.	Biểu đồ tuần tự .....	56
4.5.	Biểu đồ hoạt động.....	63
4.6.	Mô hình E-R .....	71
4.7.	Mô hình quan hệ dữ liệu.....	72
4.8.	Thiết kế hệ cơ sở dữ liệu vật lý.....	73
5.	<b>CHƯƠNG 5: XÂY DỰNG VÀ THIẾT KẾ ỨNG DỤNG ĐẶT LỊCH KHÁM CHỮA BỆNH DOCTOR CARE.....</b>	<b>78</b>
5.1.	Giao diện trang web và chức năng dành cho bệnh nhân .....	78
5.2.	Giao diện trang web và chức năng dành cho bác sĩ.....	82
5.3.	Giao diện trang web và chức năng dành cho admin.....	84
6.	<b>KẾT LUẬN.....</b>	<b>87</b>
6.1.	Kết quả đạt được.....	87
6.2.	Hạn chế của đề tài .....	87
6.3.	Hướng phát triển đề tài .....	87
	<b>TÀI LIỆU THAM KHẢO.....</b>	<b>88</b>

## DANH MỤC HÌNH ẢNH

<b>Số hiệu hình ảnh</b>	<b>Tên hình ảnh</b>	<b>Trang</b>
1	UC tổng quát bác sĩ	36
2	UC tổng quát bệnh nhân	37
3	UC tổng quát Admin	38
4	UC đăng nhập	39
6	UC quản lý lịch khám	40
7	UC đặt lịch	40
8	UC quản lý bác sĩ	41
9	UC quản lý bệnh nhân	41
10	UC quản lý khoa	42
11	Biểu đồ tuần tự đăng nhập	56
12	Biểu đồ tuần tự đăng nhập	56
13	Biểu đồ tuần tự sửa thông tin cá nhân	57
14	Biểu đồ tuần tự đặt lịch	48
15	Biểu đồ tuần tự sửa thông tin khoa	48
16	Biểu đồ tuần tự thêm thông tin khoa	49
17	Biểu đồ tuần tự tạo lịch nghỉ bác sĩ	50
18	Biểu đồ tuần tự tìm kiếm bác sĩ	51
19	Biểu đồ tuần tự tìm kiếm khoa	52
20	Biểu đồ hoạt động đăng nhập	53
21	Biểu đồ hoạt động đăng ký	54
22	Biểu đồ hoạt động sửa thông tin cá nhân	55
23	Biểu đồ hoạt động đặt lịch	56
24	Biểu đồ hoạt động sửa thông tin khoa	57
25	Biểu đồ hoạt động thêm thông tin khoa	58
26	Biểu đồ hoạt động tạo lịch nghỉ	59

27	Biểu đồ hoạt động tìm kiếm bác sĩ	60
28	Biểu đồ hoạt động tìm kiếm khoa	61
29	Mô hình E-R	62
30	Mô hình quan hệ dữ liệu	63
31	Giao diện trang chủ	78
32	Giao diện tìm kiếm bác sĩ	78
33	Giao diện tìm kiếm chuyên khoa	79
34	Giao diện chức năng đặt lịch	79
35	Giao diện thông tin bác sĩ	80
36	Giao diện thông báo đặt lịch khám thành công	81
37	Giao diện thông báo tái khám từ bác sĩ	81
38	Giao diện xem thông tin tài khoản bệnh nhân	82
39	Giao diện đăng ký thông tin bệnh nhân, bác sĩ	83
40	Giao diện thông tin cá nhân bác sĩ	75
41	Giao diện danh sách bệnh nhân đặt lịch	76
42	Giao diện lịch khám của bác sĩ	76
43	Giao diện đặt lịch của bác sĩ	77
44	Giao diện quản lý bác sĩ	77
45	Giao diện quản lý chi tiết bác sĩ	78
46	Giao diện quản lý bệnh nhân	78
47	Giao diện quản lý khoa	79
48	Giao diện tài khoản admin	79

## DANH MỤC BẢNG

<b>Số hiệu bảng</b>	<b>Tên bảng</b>	<b>Trang</b>
1	Đặc tả UC đăng nhập	42
2	Đặc tả UC xem thông tin cá nhân	44
3	Đặc tả UC sửa thông tin cá nhân	44
4	Đặc tả UC thiết lập thời gian làm việc	45
5	Đặc tả UC tạo lịch nghỉ	46
6	Đặc tả UC tạo lịch tái khám cho bệnh nhân	47
7	Đặc tả UC xóa lịch đặt	48
8	Đặc tả UC đặt lịch khám	49
9	Đặc tả UC xác nhận thông tin đơn đặt khám	50
10	Đặc tả UC hủy lịch đặt	50
11	Đặc tả UC nhận đơn tái khám	51
12	Đặc tả UC Xem thông tin bệnh nhân	51
13	Đặc tả UC xóa thông tin bệnh nhân	52
14	Đặc tả UC xóa bác sĩ	53
15	Đặc tả UC quản lý đánh giá của bệnh nhân với bác sĩ	53
16	Đặc tả UC quản lý lịch đặt khám của bác sĩ	54
17	Đặc tả UC xem lịch nghỉ của bác sĩ	54
18	Đặc tả UC xem thông tin bác sĩ	55
19	Đặc tả UC sửa thông tin khoa	56
20	Đặc tả UC thêm mới khoa	56
21	Bảng Doctor	65
22	Bảng Patient	66
23	Bảng Notify_patient	66
24	Bảng Comment	67

25	Bảng Specialist	67
26	Bảng Day_off	67
27	Bảng Admin	68



# **1 Chương 1:**

## **GIỚI THIỆU TỔNG QUAN ĐỀ TÀI**

### **1.1. Lý do chọn đề tài**

Thiết kế website phòng khám, bệnh viện hiện là tiêu chí hàng đầu cho nhu cầu truyền thông dịch vụ của nhiều trung tâm chăm sóc và điều trị sức khỏe cả công và tư.

Một trong số những ngành dịch vụ quan trọng nhất hiện nay chính là dịch vụ chăm sóc sức khỏe. Trong chăm sóc sức khỏe lại được chia ra thành nhiều chuyên khoa như khoa nội, khoa ngoại, khoa mắt, thần kinh, khoa răng miệng,... Trước đây, khi những dịch vụ này chưa được chú ý phát triển ở nước ta, hầu như ngành dịch vụ này không cần được truyền thông nhắc đến vẫn thu hút “khách hàng”. Người dân khi có nhu cầu sẽ trực tiếp đến các trung tâm chăm sóc sức khỏe, bệnh viện công để khám và chữa bệnh. Nhưng hiện nay, vì nhiều nguyên nhân mà mọi người có nhu cầu chăm sóc sức khỏe nhiều hơn, không cần phải mắc bệnh rồi mới đến kiểm tra và chữa trị. Nhu cầu tăng đột ngột dẫn đến sự ra đời của rất nhiều phòng khám, bệnh viện tư nhân. Khi ấy, ngoài chất lượng dịch vụ luôn được đảm bảo, phòng khám nào có danh tiếng hơn, phòng khám ấy chắc chắn được nhiều bệnh nhân trao gửi niềm tin. Hiện nay, công cụ truyền thông quan trọng nhất đối với phòng khám chính là website giới thiệu chuyên nghiệp và chất lượng.

Bây giờ là thời đại công nghệ, đã có khoảng ít nhất 70% dân số sử dụng điện thoại thông minh. Trong đó khoảng 80% những người đó rất thông thạo việc tìm kiếm trên mạng. Việc tìm kiếm thông tin về phòng khám bây giờ có thể hoàn tất trong vòng vài giây. Như vậy, để phòng khám, bệnh viện tiếp cận với khách hàng của mình chỉ có cách duy nhất là thiết kế và phát triển website chuyên nghiệp.

Sự gắn kết chặt chẽ giữa phòng khám với người dân được thể hiện hoàn toàn rõ ràng trên website của mình thông qua bình luận, những câu chuyện chia sẻ, những câu hỏi và tư vấn từ bác sĩ chuyên môn.

Một trong những khó khăn trong ngành chăm sóc sức khỏe hiện nay là việc người dân có quá ít kiến thức về bệnh tật cũng như cách tự chăm sóc và bảo vệ sức khỏe của mình. Trước hiện trạng nhiều loại bệnh tật xuất hiện và gây nguy hiểm đến tính mạng hiện nay, nhiều người dần hình thành cảm giác lo lắng và mong muốn được bác sĩ giúp

đỡ, tư vấn. Khi nhu cầu quá lớn thì quỹ thời gian của bác sĩ không thể nào đáp ứng được cho từng người cụ thể. Do đó, website chính là nơi để bác sĩ giải quyết vấn đề này. Giải đáp thắc mắc một lần cho tất cả mọi người. Hiệu năng của công việc cũng nhờ đó mà tăng lên gấp bội.

## **1.2. Mục tiêu đề tài**

- Nghiên cứu về cách tổ chức và hoạt động của framework ReactJS và áp dụng xây dựng giao diện ứng dụng.
- Nghiên cứu về phương thức nhận và trả dữ liệu của framework loopback và áp dụng xây dựng API cho ứng dụng.
- Xây dựng Website đáp ứng nhu cầu đặt lịch qua mạng, tra cứu thông tin bệnh tình, tư vấn trực tuyến các vấn đề về sức khỏe, dễ dàng thao tác với nhiều đối tượng, tiết kiệm thời gian.
- Xây dựng được hệ thống quản lý thông tin bác sĩ, bệnh nhân.
- Xây dựng được chức năng tạo lịch khám bệnh giúp bác sĩ chủ động được thời gian làm việc.
- Xây dựng được chức năng thống kê, báo cáo số bệnh nhân khám bệnh để đưa ra những định hướng phát triển cho bệnh viện.

## **1.3. Đối tượng và phạm vi nghiên cứu**

### **1.3.1. Đối tượng nghiên cứu**

- Quy trình đặt lịch khám bệnh.
- Quy trình tạo lịch khám bệnh cho bác sĩ.
- Cách thức tổ chức và quản lý dữ liệu về thông tin bệnh nhân và bác sĩ.

### **1.3.2. Phạm vi nghiên cứu**

- Nghiên cứu cách viết API trên framework.
- Nghiên cứu về kiến trúc của ReactJS và cách xây dựng các Single Page Application.
- Nghiên cứu cách kết nối dữ liệu gọi API từ phía backend cho phía Frontend.

#### **1.4. Phương pháp nghiên cứu**

- Phương pháp quan sát .
- Phương pháp điều tra, khảo sát nhu cầu thực tế, sử dụng các phương pháp phân tích, điều tra thống kê từ đó xác định các yêu cầu chức năng và yêu cầu phi chức năng cần thiết.
- Tổng hợp các tài liệu liên quan đến xây dựng và phát triển phần mềm.

#### **1.5. Tiêu chí và chức năng của đề tài.**

- Website “Đặt lịch khám chữa bệnh online” với các chức năng dành cho ba đối tượng chính là bệnh nhân, bác sĩ, quản trị viên.
- Một số tiêu chí chức năng.

##### **1.5.1. Bệnh nhân.**

- Giao diện dễ dùng, đơn giản trong sử dụng.
- Giao diện web hiển thị đầy đủ và chính xác thông tin và đáp ứng nhu cầu sử dụng của bệnh nhân.
- Hệ thống cho phép bệnh nhân được quyền đăng ký tài khoản, thông tin đăng ký được bảo mật, an toàn, chính xác.
- Dễ thuận tiện cho việc chỉnh sửa thông tin bệnh nhân, hệ thống cho phép bệnh nhân có thể xem, cập nhật lại thông tin cá nhân của mình.
- Hệ thống cho phép bệnh nhân tìm kiếm bác sĩ để đặt lịch khám bệnh theo từng khoa với từng bác sĩ trong chuyên khoa.
- Hệ thống cho phép bệnh nhân tìm kiếm bác sĩ theo tên bác sĩ, theo ngày đặt lịch.
- Dễ thuận tiện cho quá trình đặt lịch hệ thống cho phép bệnh nhân đặt lịch và hủy lịch đặt trong danh sách đặt lịch.
- Dễ thuận tiện cho bệnh nhân trong quá trình lựa chọn khám chữa bệnh, hệ thống hiển thị thông tin chi tiết về bác sĩ, giá khám bệnh...

##### **1.5.2. Bác sĩ.**

- Cung cấp cho bác sĩ giao diện rõ ràng, dễ dùng.
- Hệ thống cho phép bác sĩ đăng nhập và cập nhật thông tin.

- Hệ thống cho phép bác sĩ xem được số bệnh nhân đặt lịch trong ngày.
- Hệ thống cho phép bác sĩ tạo lịch tái khám cho bệnh nhân.
- Cung cấp chức năng quản lý danh sách lịch đặt.
- Cung cấp chức năng tạo lịch nghỉ cho bác sĩ.

### **1.5.3. Quản trị viên.**

- Cung cấp cho người quản trị giao diện rõ ràng, thuận tiện, dễ dùng.
- Hệ thống cho phép người quản trị xem thông tin bác sĩ.
- Cung cấp chức năng xem lịch đặt khám của từng bác sĩ.
- Hệ thống cho phép người quản trị xem thông tin bệnh nhân.
- Cung cấp chức năng quản lý danh sách bác sĩ.
- Hệ thống cho phép người quản trị cập nhật Chuyên khoa, xem thông tin về số bác sĩ trong chuyên khoa.
- Hệ thống cho phép tạo người quản trị tạo mới thêm tài khoản.

## CHƯƠNG 2: TỔNG QUAN VỀ CÔNG NGHỆ SỬ DỤNG

### 2.1. NodeJS

#### 2.1.1. Giới thiệu về NodeJS

NodeJS là một nền tảng được xây dựng trên V8 JavaScript Engine – trình thông dịch thực thi mã JavaScript, giúp xây dựng các ứng dụng web một cách đơn giản và dễ dàng mở rộng.

NodeJS được phát triển bởi Ryan Dahl vào năm 2009 và có thể chạy trên nhiều hệ điều hành khác nhau: OS X, Microsoft Windows, Linux.

Phần Core bên dưới của NodeJS được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao.

NodeJS tạo ra được các ứng dụng có tốc độ xử lý nhanh, realtime thời gian thực.

NodeJS áp dụng cho các sản phẩm có lượng truy cập lớn, cần mở rộng nhanh, cần đổi mới công nghệ, hoặc tạo ra các dự án Startup nhanh nhất có thể.

#### 2.1.2. Tại sao nên sử dụng NodeJS

Các ứng dụng NodeJS được viết bằng javascript, ngôn ngữ này là một ngôn ngữ khá thông dụng. Theo tác giả của ngôn ngữ Javascript, Ryan Dahl: “Javascript có những đặc tính mà làm cho nó rất khác biệt so với các ngôn ngữ lập trình động còn lại, cụ thể là nó không có khái niệm về đa luồng, tất cả là đơn luồng và hướng sự kiện.”

Tốc độ xử lý nhanh. Nhờ cơ chế xử lý bất đồng bộ (non-blocking), NodeJS có thể xử lý hàng ngàn kết nối cùng lúc mà không gặp bất cứ khó khăn nào.

Dễ dàng mở rộng. Nếu bạn có nhu cầu phát triển website thì tính năng dễ dàng mở rộng của NodeJS là một lợi thế cực kỳ quan trọng.

Các ứng dụng NodeJS đáp ứng tốt thời gian thực và chạy đa nền tảng, đa thiết bị. Node.js sử dụng một mô hình luồng duy nhất với sự kiện lặp. cơ chế tổ chức sự kiện giúp các máy chủ để đáp ứng một cách không ngăn chặn và làm cho máy chủ cao khả năng mở rộng như trái ngược với các máy chủ truyền thống mà tạo ra hạn chế để xử lý yêu cầu. Node.js sử dụng một chương trình đơn luồng và các chương trình tương tự có thể cung cấp dịch vụ cho một số lượng lớn hơn nhiều so với yêu cầu máy chủ truyền thống như Apache HTTP Server.



Hình 2.1. Blocking I/O (trái) và Non-Blocking I/O (phải)

### 2.1.3. Áp dụng NodeJS vào đề tài

Trong hệ thống ứng dụng của nhóm, NodeJS được áp dụng để làm phần backend cho hệ thống, vì những ưu điểm của NodeJS dành cho một ứng dụng nhỏ rất là tiện lợi đặc biệt là tốc độ xử lý.

## 2.2. ExpressJS

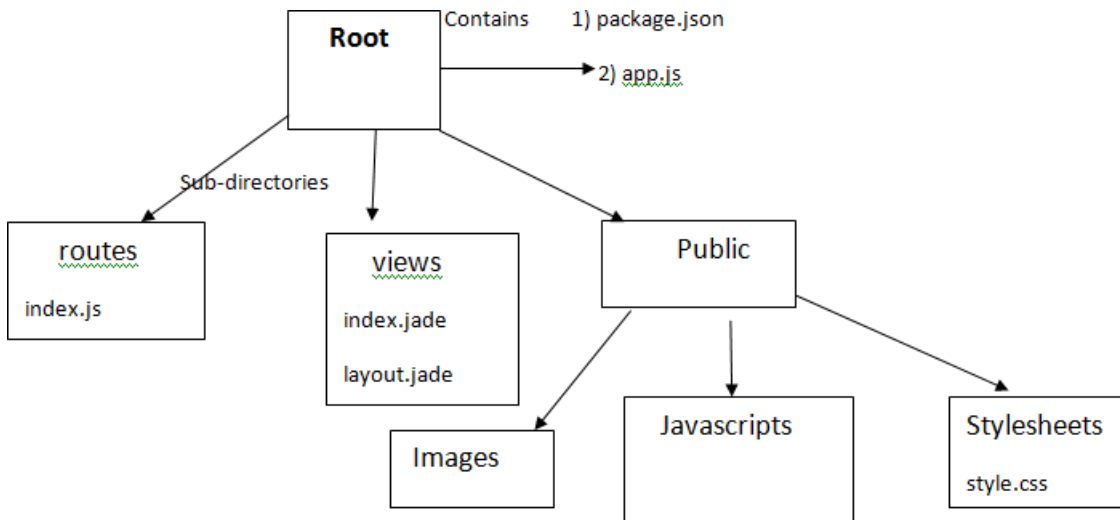
### 2.2.1. Giới thiệu về ExpressJS

Express js là một Framework nhỏ, nhưng linh hoạt được xây dựng trên nền tảng của NodeJS. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile

Về các package hỗ trợ: ExpressJS có vô số các package hỗ trợ nên không phải lo lắng khi làm việc với Framework này.

Về performance: Express cung cấp thêm về các tính năng (feature) để dev lập trình tốt hơn. Chứ không làm giảm tốc độ của NodeJS.

Và hơn hết, các Framework nổi tiếng của NodeJS hiện nay đều sử dụng ExpressJS như một core function, chẳng hạn: SailsJS, MEAN,....



Hình 2.2. Cấu trúc của ExpressJs

- App.js chứa các thông tin về cấu hình, khai báo, các định nghĩa,... để ứng dụng của chúng ta chạy trơn tru.

- Package.json chứa các package cho ứng dụng chạy, có chức năng tương tự như composer.json hoặc Gemfile trong PHP hoặc RoR.
- Folder routes: chứa các route có trong ứng dụng
- Folder view: chứa view/template cho ứng dụng
- Folder public chứa các file css, js, images, ... cho ứng dụng

### **2.2.2. Tại sao nên sử dụng ExpressJS**

Framework giúp cho việc phát triển ứng dụng được rút ngắn đi rất nhiều. Cũng như các framework dựa trên những ngôn ngữ khác như Rails (Ruby); Django (Python); Laravel, CakePHP (PHP)... Express được xây dựng dựa trên NodeJS. Vậy nó có ưu điểm gì để ta lựa chọn cho việc phát triển ứng dụng.

Express hỗ trợ việc phát triển ứng dụng theo mô hình MVC, mô hình phổ biến cho việc lập trình web hiện nay.

Cho phép định nghĩa Middleware hỗ trợ cho việc tổ chức và tái sử dụng code. Định nghĩa routes và các request method đến server một cách dễ dàng.

Hỗ trợ REST API.

Và còn rất nhiều thứ mà framework này hỗ trợ.

### **2.2.3. Áp dụng ExpressJS vào đề tài**

Trong hệ thống ứng dụng của nhóm, ExpressJS được áp dụng để làm phần framework backend cho hệ thống, vì những ưu điểm của ExpressJS có dành cho một ứng dụng nhỏ rất là tiện lợi đặc biệt là tốc độ xử lý, hỗ trợ Rest API, cộng đồng lớn mạnh.



## 2.3. MySQL

### 2.3.1. Giới thiệu về MySQL

Hệ quản trị cơ sở dữ liệu MySQL được hiểu như là chương trình dùng để quản lý hệ thống cơ sở dữ liệu, trong đó, cơ sở dữ liệu là một hệ thống lưu trữ thông tin được sắp xếp rõ ràng, phân lớp ngăn nắp. Nó giúp bạn có thể truy cập dữ liệu một cách thuận lợi và nhanh chóng nhất. Vì hỗ trợ đa số các ngôn ngữ lập trình nên MySQL chính là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất trên thế giới. Hiện MySQL đang được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng.

MySQL là cơ sở dữ liệu tốc độ cao, ổn định hoạt động trên nhiều hệ điều hành, cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Đặc biệt, hệ quản trị cơ sở dữ liệu MySQL hoàn toàn miễn phí nên người dùng có thể thoải mái tải về từ trang chủ. Nó có rất nhiều những phiên bản cho các hệ điều hành khác nhau. MySQL được sử dụng cho việc hỗ trợ PHP, Perl và nhiều ngôn ngữ khác. Là nơi lưu trữ những thông tin trên các trang web viết bằng framework PHP hay Perl...

### 2.3.2. Tại sao nên sử dụng hệ quản trị cơ sở dữ liệu MySQL

*Ưu điểm:*

- Dễ sử dụng: MySQL là cơ sở dữ liệu tốc độ cao, ổn định, dễ sử dụng và hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh.
- Độ bảo mật cao: MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên Internet khi sở hữu nhiều nhiều tính năng bảo mật thậm chí là ở cấp cao.
- Đa tính năng: MySQL hỗ trợ rất nhiều chức năng SQL được mong chờ từ một hệ quản trị cơ sở dữ liệu quan hệ cả trực tiếp lẫn gián tiếp.
- Khả năng mở rộng và mạnh mẽ: MySQL có thể xử lý rất nhiều dữ liệu và hơn thế nữa nó có thể được mở rộng nếu cần thiết.
- Nhanh chóng: Việc đưa ra một số tiêu chuẩn cho phép MySQL để làm việc rất hiệu quả và tiết kiệm chi phí, do đó nó làm tăng tốc độ thực thi.

*Nhược điểm:*

- Giới hạn: Theo thiết kế, MySQL không có ý định làm tất cả và nó đi kèm với các hạn chế về chức năng mà một vào ứng dụng có thể cần.
- Độ tin cậy: Cách các chức năng cụ thể được xử lý với MySQL (ví dụ tài liệu tham khảo, các giao dịch, kiểm toán,...) làm cho nó kém tin cậy hơn so với một số hệ quản trị cơ sở dữ liệu quan hệ khác.
- Dung lượng hạn chế: Nếu số bản ghi của bạn lớn dần lên thì việc truy xuất dữ liệu của bạn là khá khó khăn, khi đó chúng ta sẽ phải áp dụng nhiều biện pháp để tăng tốc độ truy xuất dữ liệu như là chia tải database này ra nhiều server, hoặc tạo cache MySQL

### **2.3.3. Áp dụng MySQL vào đề tài**

Nhờ những tính năng nổi trội của MySQL, nhóm đã sử dụng MySQL dùng làm hệ quản trị cơ sở dữ liệu chính. Server sử dụng MySQL để lưu trữ toàn bộ dữ liệu, dữ liệu này sẽ được server sử dụng REST API để gọi lên.

## **2.4. ReactJS**

### **2.4.1. Giới thiệu về ReactJS**

React.js là một thư viện Javascript đang nổi lên trong những năm gần đây với xu hướng Single Page Application. Trong khi những framework khác cố gắng hướng đến một mô hình MVC hoàn thiện thì React nổi bật với sự đơn giản và dễ dàng phối hợp với những thư viện Javascript khác. Nếu như AngularJS là một Framework cho phép nhúng code javascript trong code html thông qua các attribute như ng-model, ng-repeat...thì với react là một library cho phép nhúng code html trong code javascript nhờ vào JSX, có thể dễ dàng lồng các đoạn HTML vào trong JS. Tích hợp giữa javascript và HTML vào trong JSX làm cho các component dễ hiểu hơn

React là một thư viện UI phát triển tại Facebook để hỗ trợ việc xây dựng những thành phần (components) UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được. React được sử dụng tại Facebook trong production, và Instagram được viết hoàn toàn trên React.

Một trong những điểm hấp dẫn của React là thư viện này không chỉ hoạt động trên phía client, mà còn được render trên server và có thể kết nối với nhau. React so sánh sự thay đổi giữa các giá trị của lần render này với lần render trước và cập nhật ít thay đổi nhất trên DOM.

#### **2.4.2. Tại sao nên sử dụng ReactJS**

- Dễ sử dụng:
- React là một thư viện GUI nguồn mở JavaScript tập trung vào một điều cụ thể; hoàn thành nhiệm vụ UI hiệu quả. Nó được phân loại thành kiểu “V” trong mô hình MVC (Model-View-Controller).
- Là lập trình viên JavaScript, bạn sẽ dễ dàng hiểu được những điều cơ bản về React. Bạn thậm chí có thể bắt đầu phát triển các ứng dụng dựa trên web bằng cách sử dụng react chỉ trong vài ngày.
- Để củng cố hiểu biết của mình, bạn hãy thử khám phá thêm nhiều hướng dẫn về React. Chúng mang đến nhiều thông tin về cách sử dụng công cụ: videos, hướng dẫn và dữ liệu làm phong phú góc nhìn của bạn.
- Viết component dễ dàng hơn:
- React component dễ viết hơn vì nó sử dụng JSX, mở rộng cú pháp tùy chọn cho JavaScript cho phép kết hợp HTML với JavaScript.
- JSX là một sự pha trộn tuyệt vời của JavaScript và HTML. Nó làm rõ toàn bộ quá trình viết cấu trúc trang web. Ngoài ra, phần mở rộng cũng giúp render nhiều lựa chọn dễ dàng hơn.
- JSX có thể không là phần mở rộng cú pháp phổ biến nhất, nhưng nó được chứng minh là hiệu quả trong việc phát triển components đặc biệt hoặc các ứng dụng có khối lượng lớn.
- Hiệu suất tốt hơn với Virtual DOM:
- React sẽ cập nhật hiệu quả quá trình DOM (Document Object Model – Mô hình đối tượng tài liệu). Như đã biết, quá trình này có thể gây ra nhiều thất vọng trong các dự án ứng dụng dựa trên web. May mắn là React sử dụng virtual DOMs, vì vậy có thể tránh được vấn đề này.

- Công cụ cho phép xây dựng các virtual DOMs và host chúng trong bộ nhớ. Nhờ vậy, mỗi khi có sự thay đổi trong DOM thực tế, thì virtual sẽ thay đổi ngay lập tức.
- Hệ thống này sẽ ngăn DOM thực tế để buộc các bản cập nhật được liên tục. Do đó, tốc độ của ứng dụng sẽ không bị gián đoạn.
- Thân thiện với SEO:
- React cho phép tạo giao diện người dùng có thể được truy cập trên các công cụ tìm kiếm khác nhau. Tính năng này là một lợi thế rất lớn vì không phải tất cả các khung JavaScript đều thân thiện với SEO.
- Ngoài ra, vì React có thể tăng tốc quá trình của ứng dụng nên có thể cải thiện kết quả SEO. Cuối cùng tốc độ web đóng một vai trò quan trọng trong tối ưu hóa SEO.

## **Chương 3. PHÁT BIỂU BÀI TOÁN**

### **3.1. Giới thiệu hệ thống website “Đặt lịch khám bệnh online ”**

- Là website tổng thể được thiết kế hiện đại nổi bật giới thiệu tổng quan về bệnh viện, khoa phòng chức năng.
- Trang web giúp người dùng có thể tra cứu các thông tin cần thiết trước khi đặt lịch khám với bác sĩ như chuyên khoa, cơ sở y tế, bác sĩ..
- Trang web có công cụ sắp xếp một cách hiệu quả giúp người dùng cần tìm một số thông tin một cách nhanh nhất có thể, ví dụ tìm theo: bác sĩ, chuyên khoa, cơ sở y tế
- Trang web hỗ trợ các bác sĩ có thể quản lý lịch làm việc, thông tin bệnh nhân một cách hiệu quả và chính xác
- Trang web giúp người dùng đăng ký đặt lịch khám bệnh online dễ dàng và nhanh chóng trong mùa dịch covid hạn chế di chuyển giữa các vùng
- Trang web giúp người dùng xác nhận thông tin và phản hồi kết quả qua gmail nhanh chóng và hiệu quả
- Trang web giúp quản lý các thông tin bác sĩ, chuyên khoa và phòng khám một cách chi tiết và chính xác giúp người dùng dễ dàng tra cứu thông tin cần thiết
- Chức năng cho phép người dân có thể đăng ký khám bệnh trực tuyến thông qua website của khoa khám bệnh. Các bước đăng ký khám được quy định chặt chẽ đồng bộ với lịch làm việc của các bác sĩ - khoa khám bệnh

## CHƯƠNG 4 : PHÂN TÍCH HỆ THỐNG ĐẶT LỊCH KHÁM CHỮA BỆNH DOCTOR CARE

### 4.1. Khảo sát

Để có được một trang web đảm bảo chất lượng và hoàn thiện em đã thực hiện khảo sát và tham khảo một số trang web đặt lịch khám bệnh khác để từ đó giúp cho em trong việc hoàn thiện sản phẩm của mình tốt hơn.

Các trang web tham khảo: umc.medpro.com.vn, BookingCare.vn, youmed.vn...

Về mặt tổng thể một trang đặt lịch khám bệnh cần đảm bảo những chức năng cơ bản sau:

- Giới thiệu về bệnh viện
- Giới thiệu thông tin về website.
- Các thông tin cơ bản về nội dung website và các điều lệ.
- Thông tin bác sĩ.
- Thông tin chi tiết về các bác sĩ.
- Giá bán khám bệnh theo từng bác sĩ, thông tin chi tiết.
- Thông tin chuyên khoa.
- Thông tin chi tiết về các chuyên khoa.
- Bác sĩ trong chuyên khoa.
- Trợ giúp.
- Cách đặt lịch.
- Tìm kiếm bác sĩ.
- Lịch đã đặt.

➤ **Kết luận:** Từ việc tham khảo các trang web trên đã giúp cho em định hình được cấu trúc cơ bản của một trang web đặt lịch online. Những chức năng về bác sĩ, phương thức đặt lịch..., ngoài ra một trang đặt lịch phải có giao diện dễ nhìn, thân thiện với người dùng, đơn giản nhưng đầy đủ tính năng để tiết kiệm chi phí về thời gian và kinh phí đi lại.

## **4.2. Phân tích hệ thống**

### **4.2.1. Đặc tả hệ thống**

- DoctorCare là nền tảng đặt lịch khám giúp bệnh nhân dễ dàng lựa chọn đúng bác sĩ từ mạng lưới bác sĩ chuyên khoa giỏi, với thông tin đã xác thực và đặt lịch nhanh chóng.
- Cung cấp nội dung hướng dẫn giúp bệnh nhân dễ dàng lựa chọn bác sĩ phù hợp với vấn đề của mình để đạt hiệu quả cao trong quá trình khám và chữa bệnh.
- Đặc biệt hệ thống còn tiếp nhận và phản hồi các ý kiến đóng góp của bệnh nhân đã từng tham gia khám chữa bệnh trực tuyến để đóng góp và tăng thêm thông tin tham khảo dành cho các bệnh nhân khác để lựa chọn bác sĩ phù hợp với mình. Việc lựa chọn bác sĩ đúng với tình trạng bệnh của mỗi người là cực kỳ quan trọng giúp bệnh nhân an tâm trong quá trình chữa trị bệnh.
- Hệ thống đặt lịch khám chữa bệnh là hệ thống đặt lịch trực tuyến giúp người bệnh dễ dàng lựa chọn bác sĩ chuyên khoa giỏi phù hợp. Với thông tin chính xác người bệnh dễ dàng tìm hiểu được thông tin tình trạng bệnh của mình. Hơn nữa, ứng dụng còn cung cấp cho người dùng giá cả dịch vụ, thông tin bác sĩ rõ ràng, tiết kiệm thời gian và chi phí đi lại. Không chỉ tập trung vào tiện ích hay giảm thiểu thời gian chờ khám, hệ thống áp dụng công nghệ kỹ thuật cao tiên tiến giúp bệnh nhân dễ dàng lựa chọn bác sĩ, cơ sở y tế phù hợp với vấn đề bệnh tật của mình để đi khám chữa bệnh nâng cao hiệu quả khám chữa bệnh, tiết kiệm thời gian chi phí khám chữa bệnh.

### **4.2.2. Yêu cầu chức năng**

- Website phải có đầy đủ các chức năng dành cho bệnh nhân, bác sĩ, admin.
- Các chức năng dành cho bệnh nhân: Đăng nhập, đăng ký, xem thông tin bệnh nhân, xem lịch sử đặt lịch, đặt lịch, tìm kiếm bác sĩ, tìm kiếm theo chuyên khoa, tìm kiếm bác sĩ theo chuyên khoa, đánh giá bác sĩ, xem thông tin bác sĩ, cập nhật thông tin cá nhân.
- Các chức năng dành cho bác sĩ: Đăng nhập, đăng ký, tạo lịch tái khám, xem thông tin đơn đặt lịch, tạo lịch nghỉ, thiết lập giờ làm việc, xem thông tin bệnh nhân đặt lịch theo ngày, cập nhật thông tin cá nhân. Tìm kiếm bệnh nhân.

- Các chức năng dành cho admin: Đăng nhập, đăng ký, quản lý danh sách bác sĩ, xem thông tin từng bác sĩ, quản lý thông tin lịch đặt của bác sĩ, quản lý các phản hồi bình luận của bệnh nhân, quản lý xác thực thông tin tin đăng ký bác sĩ, quản lý lịch nghỉ của bác sĩ, quản lý thông tin bệnh nhân.

#### **4.2.3. Yêu cầu phi chức năng**

- Giao diện đẹp mắt, thân thiện với người dùng.
- Hệ thống xử lý nhanh, ít bị sự cố.
- An toàn và bảo mật thông tin
- Có tính tương thích cao
- Mô tả chức năng

#### **Chức năng cho bệnh nhân**

- **Đăng nhập:** Đăng nhập vào website. Bệnh nhân đăng nhập tài khoản đã đăng ký với hệ thống trước khi thực hiện đặt lịch, nếu chưa có tài khoản thì phải đăng ký tài khoản.
- **Đăng ký:** Đăng ký tài khoản vào website. Bệnh nhân điền các thông tin cá nhân, thông tin về tài khoản vào chức năng đăng ký.
- **Xem thông tin bác sĩ:** Bệnh nhân có thể xem thông tin bác sĩ và xem đánh giá về bác sĩ.
- **Đặt lịch:** Bệnh nhân thực hiện chức năng đặt lịch, thì lịch sẽ được chuyển vào lịch chờ của bệnh nhân.
- **Lịch chờ:** Bệnh nhân xem thông tin lịch đã đặt tại lịch chờ, chứa các thông tin chi tiết về lịch đã đặt bao gồm tên bác sĩ, giờ đặt, ngày đặt, giá khám. Bệnh nhân có thể thực hiện được các chức năng hủy lịch đặt để hủy lịch đặt theo ngày đó.
- **Đánh giá bác sĩ:** Bệnh nhân có thể đánh giá về bác sĩ sau khi khám.
- **Tìm kiếm:** Bệnh nhân có thể tìm kiếm thông tin lịch đặt theo chuyên khoa hay bác sĩ bằng cách nhập thông tin cần tìm vào ô tìm kiếm.

#### **Chức năng bác sĩ**



- **Đăng nhập:** Bác sĩ đăng nhập tài khoản đã được cấp để thực hiện đăng nhập vào hệ thống.
- **Đăng ký:** Đăng ký tài khoản vào website. Bác sĩ điền các thông tin cá nhân, thông tin về tài khoản vào chức năng đăng ký.
- **Xem thông tin bác sĩ:** Bác sĩ có thể xem thông tin cá nhân của mình ở menubar thông tin tài khoản.
- **Cập nhật thông tin tài khoản:** Bác sĩ có thể sửa là lưu thay đổi về thông tin cá nhân của mình tại mục thông tin tài khoản.
- **Quản lý lịch:** Bác sĩ có thể tạo lịch đặt theo từng ngày, thêm lịch đặt cho ngày, xóa giờ khám, chỉnh sửa giờ khám, xem thông tin bệnh nhân theo giờ đặt.
- **Tạo lịch nghỉ:** Bác sĩ có thể tạo lịch nghỉ theo ngày và thời gian.
- **Cài đặt lịch:** bác sĩ cài đặt lịch khi mới tạo tài khoản, cài đặt về thời gian khám và thời gian bắt đầu làm việc và thời gian kết thúc làm việc.

### **Chức năng cho người quản trị**

- **Quản lý bác sĩ:** Người quản trị có thể thêm mới, xóa thông tin bác sĩ trên hệ thống. Bác sĩ được hệ thống quản lý với các thông tin: Mã số, Họ tên, Ngày sinh, Chuyên khoa, Hình ảnh, Chi tiết thông tin bác sĩ. Đặc biệt hệ thống có hỗ trợ cho người dùng quản lý lịch làm việc và các phản hồi của bệnh nhân. Có thể tìm kiếm thông tin bác sĩ theo tên bác sĩ.
- **Quản lý lịch khám của bác sĩ:** Người quản trị có thể xem thông tin về bệnh nhân khám và danh sách lịch khám của bác sĩ, có khả năng xóa lịch của bác sĩ.
- **Quản lý lịch nghỉ bác sĩ:** Người quản trị có thể xem thông tin về lịch nghỉ của bác sĩ.
- **Quản lý đánh giá về bác sĩ:** Người quản trị có khả năng quản lý về đánh giá của bệnh nhân về bác sĩ như xem và xóa đánh giá.
- **Quản lý khoa:** Người quản trị có thể thêm mới, cập nhật thông tin khoa của bệnh viện trên hệ thống. Khoa sẽ được hệ thống quản lý với các thông tin: Mã số, Tên khoa, chi tiết thông tin khoa. Người quản trị có thể được tìm kiếm theo tên khoa.

- **Quản lý bệnh nhân:** Người quản trị có thể xoá thông tin bệnh nhân trên hệ thống. Tài khoản của bệnh nhân được hệ thống quản lý với các thông tin: Mã số, Họ tên, Ngày sinh, và các thông tin chi tiết về bệnh nhân. Người quản trị có thể tìm kiếm thông tin bệnh nhân theo tên của bệnh nhân.

#### 4.3. User case

##### 4.3.1. Danh sách tác nhân

- **BỆNH NHÂN:** là người sử dụng hệ thống thông qua các tác vụ như xem thông tin bác sĩ, đặt lịch khám chữa bệnh, xem lịch sử khám chữa bệnh, đánh giá bác sĩ, tìm kiếm bác sĩ, tìm kiếm chuyên khoa.
- **BÁC SĨ:** là người sử dụng hệ thống và thực hiện các tác vụ như xem lịch làm việc (lịch bệnh nhân đã đặt), xem thông tin thông tin đơn đặt khám theo ngày hoặc giờ, tạo lịch nghỉ, thiết lập thời gian làm việc, tạo lịch tái khám cho bệnh nhân.
- **NGƯỜI QUẢN LÝ:** là người trực tiếp điều hành hệ thống, quản lý và theo dõi các hoạt động, cập nhật thông tin.

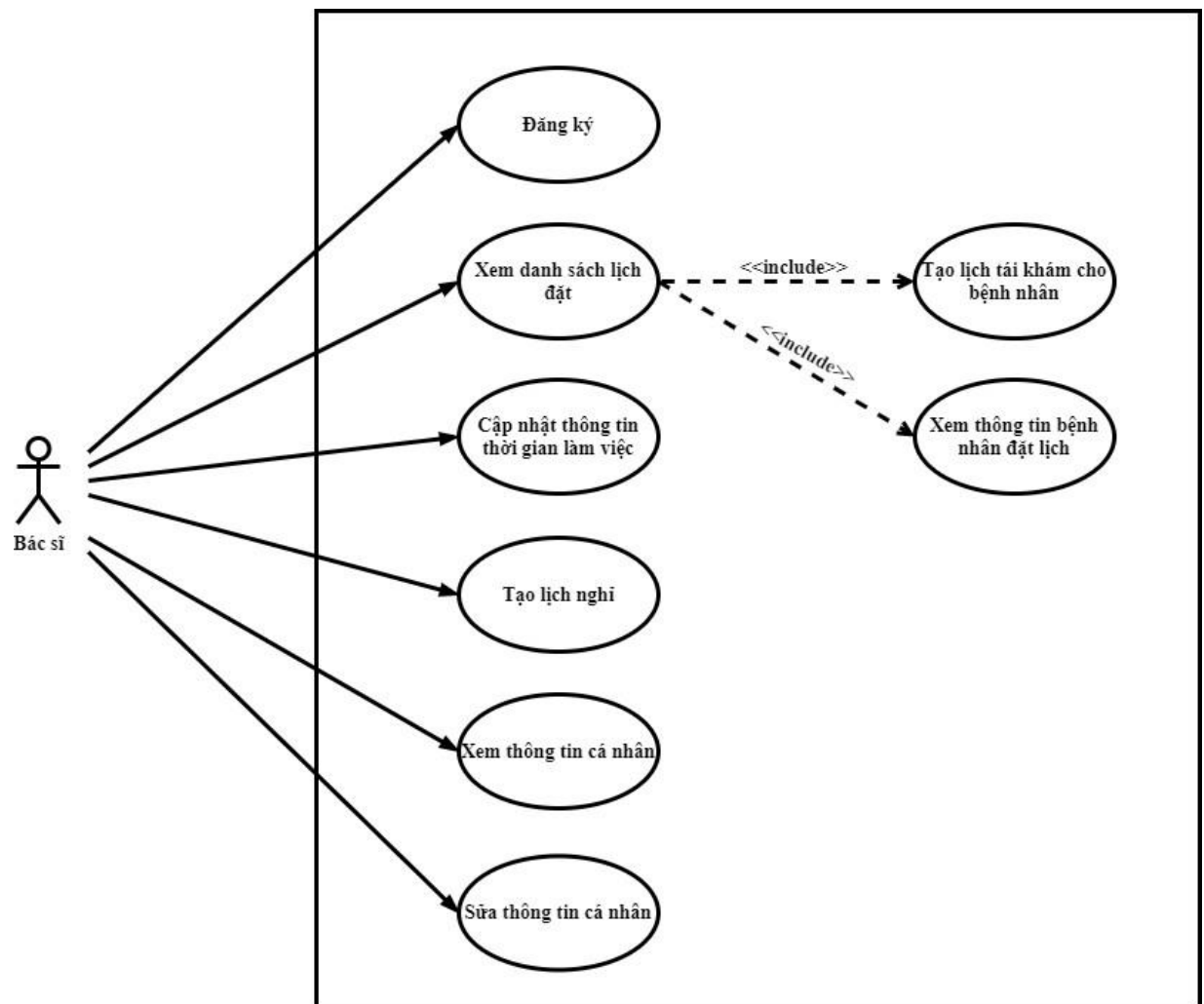
##### 4.3.2. Danh sách Use Case

- Tác nhân BỆNH NHÂN có các UC sau:
  - Đăng ký.
  - Xem thông tin bác sĩ.
  - Xem thông tin các khoa.
  - Đặt lịch.
  - Xem danh sách lịch đã đặt.
  - Đánh giá bác sĩ.
- Tác nhân BÁC SĨ có các UC sau:
  - Đăng ký.
  - Xem danh sách đặt lịch
  - Cập nhật thông tin thời gian làm việc.
  - Tạo lịch nghỉ
  - Tạo lịch tái khám cho bệnh nhân
  - Xem thông tin bệnh nhân đặt lịch

- Tác nhân NGƯỜI QUẢN LÝ có các UC sau:
  - Quản lý danh sách bác sĩ
  - Quản lý danh sách bệnh nhân.
  - Quản lý lịch đặt khám của bác sĩ
  - Quản lý lịch nghỉ của bác sĩ.
  - Quản lý đánh giá của bệnh nhân với bác sĩ
  - Quản lý thông tin khoa
- Ngoài ra, các thành viên của hệ thống bao gồm người quản lý, bác sĩ và các bệnh nhân đã đăng ký làm thành viên còn có các UC sau:
  - Đăng nhập/Đăng xuất
  - Xem thông tin cá nhân
  - Sửa đổi thông tin cá nhân

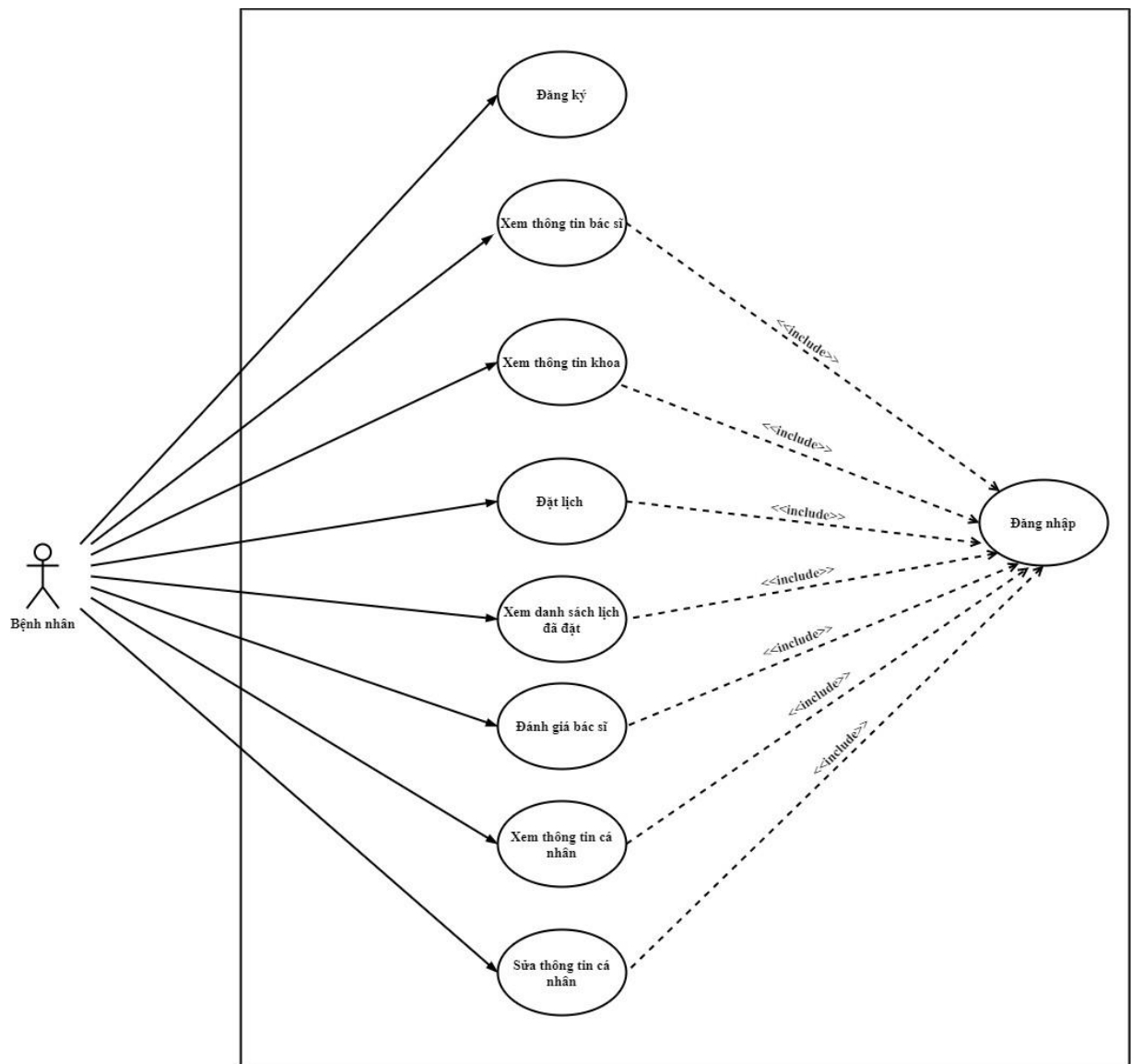
### 4.3.3. Biểu đồ UC tổng quát

- UC tổng quát bác sĩ.



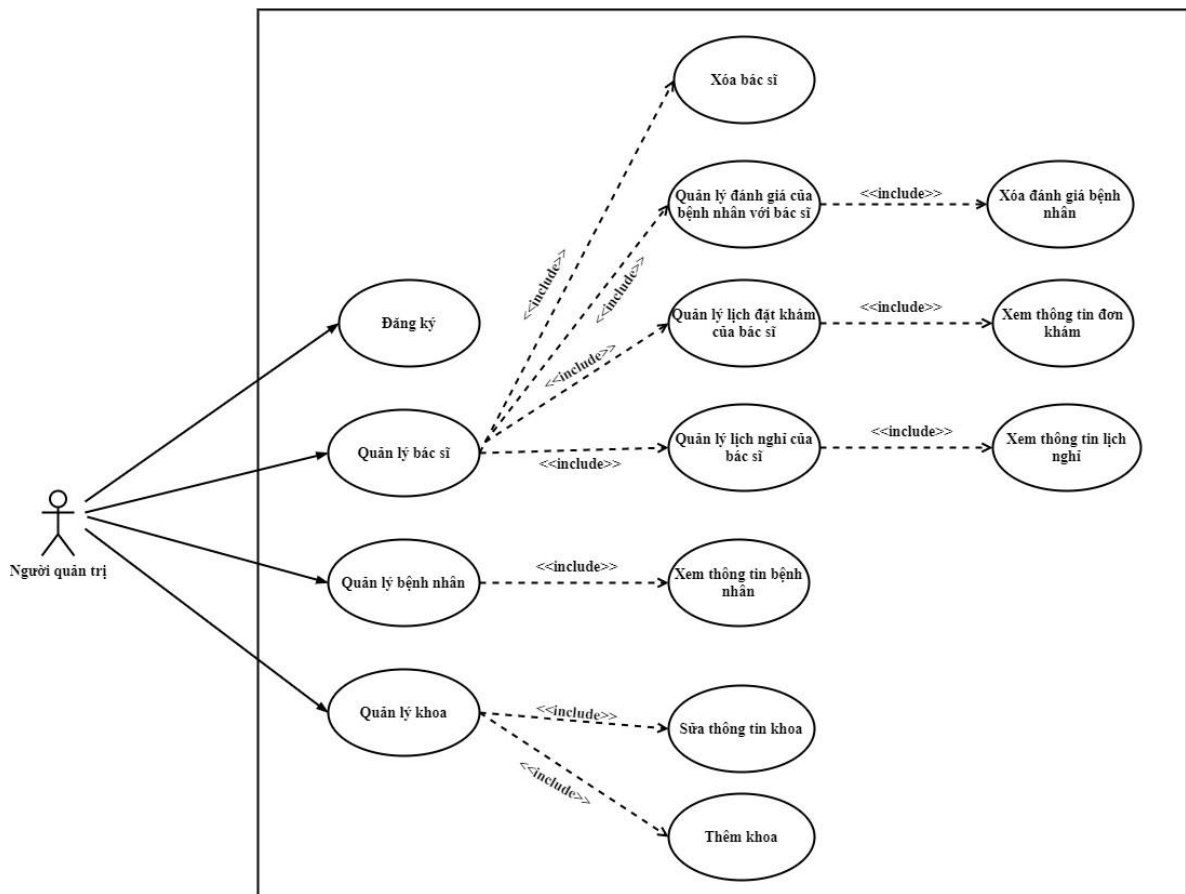
**Hình 1.** UC tổng quát bác sĩ.

- UC tổng quát bệnh nhân.



**Hình 2.** UC tổng quát bệnh nhân.

- UC tổng quát Admin.



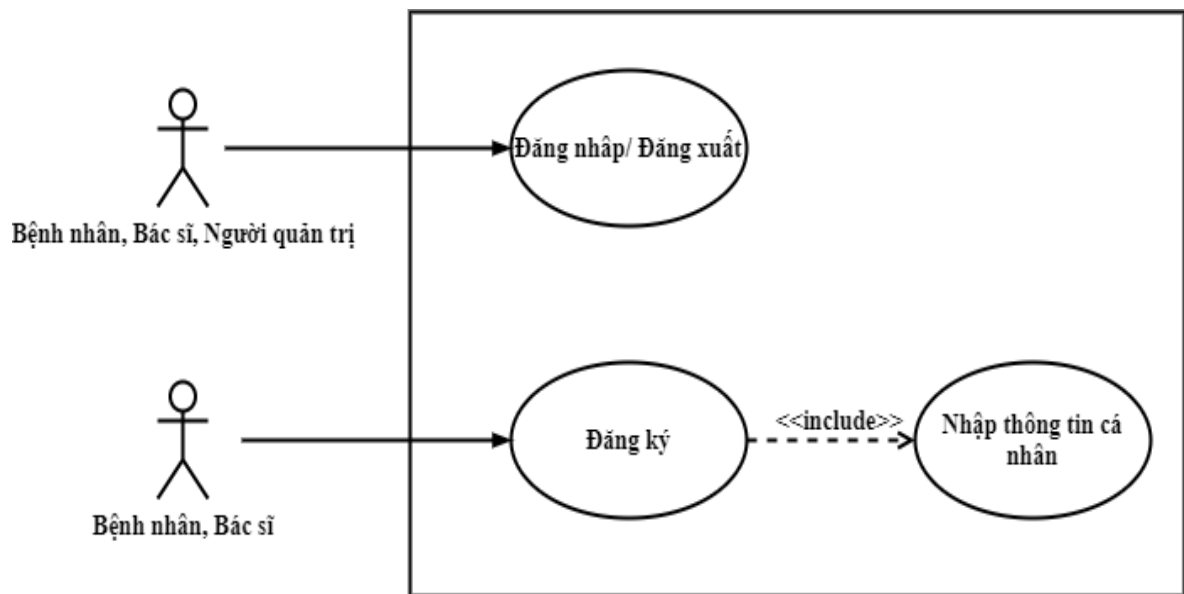
**Hình 3.** UC tổng quát admin.

- Gói UC, biểu đồ UC chi tiết.

Từ việc phân tích các UC của từng tác nhân, em xây dựng được các gói UC như sau:

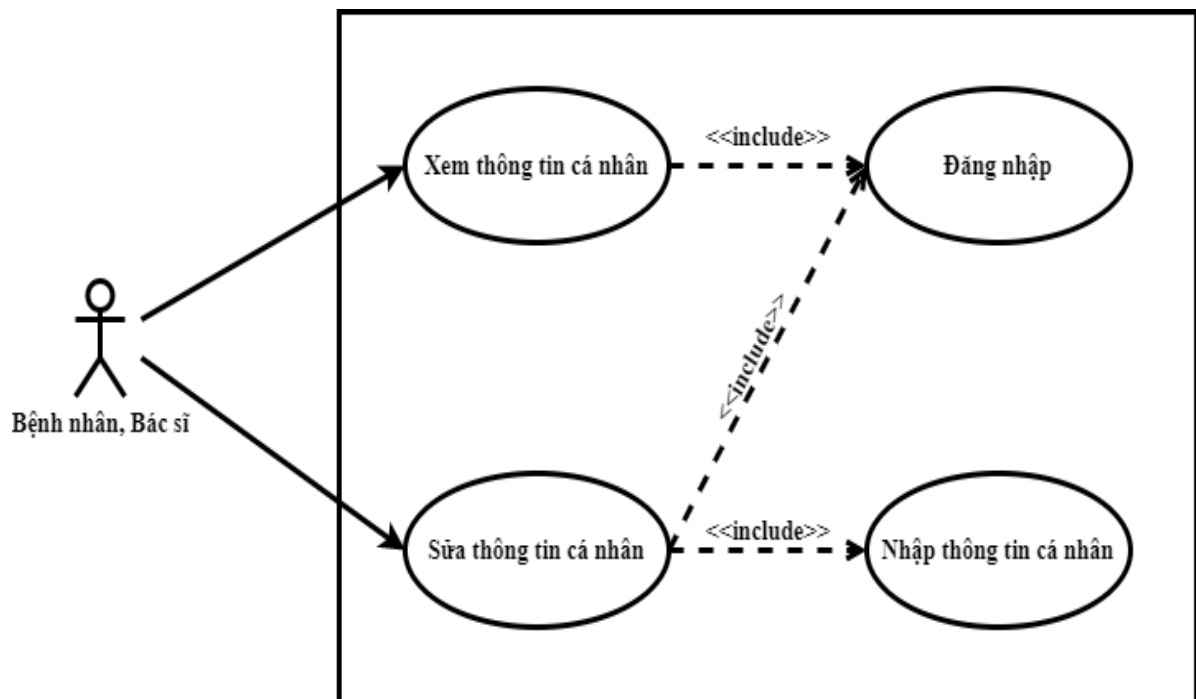
- Đăng nhập/ Đăng ký.
- Quản lý thông tin cá nhân.
- Quản lý lịch khám.
- Đặt lịch
- Quản lý bệnh nhân.
- Quản lý bác sĩ.
- Quản lý khoa.

- Gói UC đăng nhập/ đăng ký.



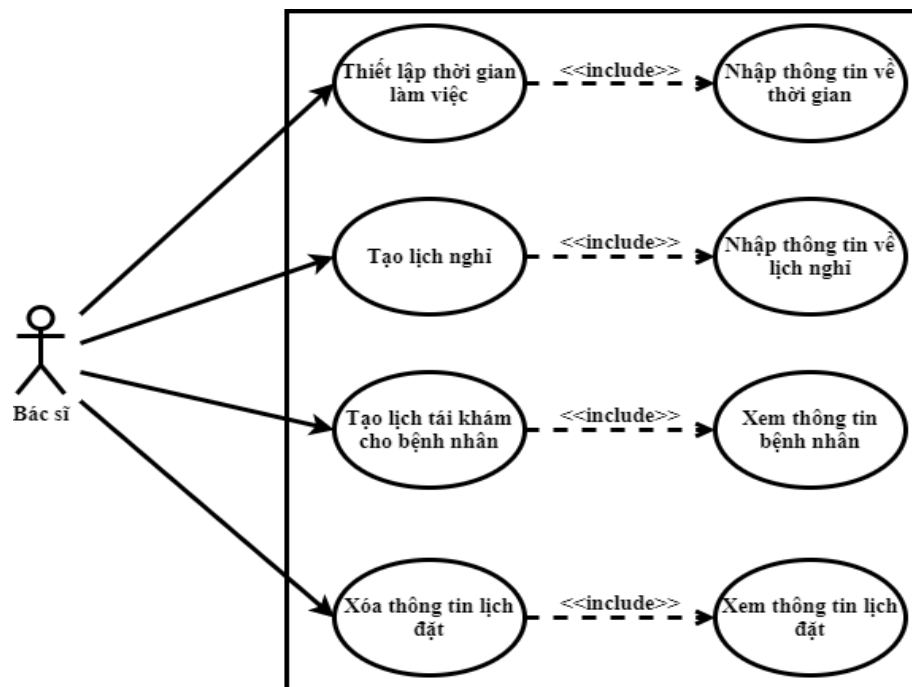
**Hình 4.** UC đăng nhập/ đăng ký.

- Gói UC quản lý thông tin cá nhân.



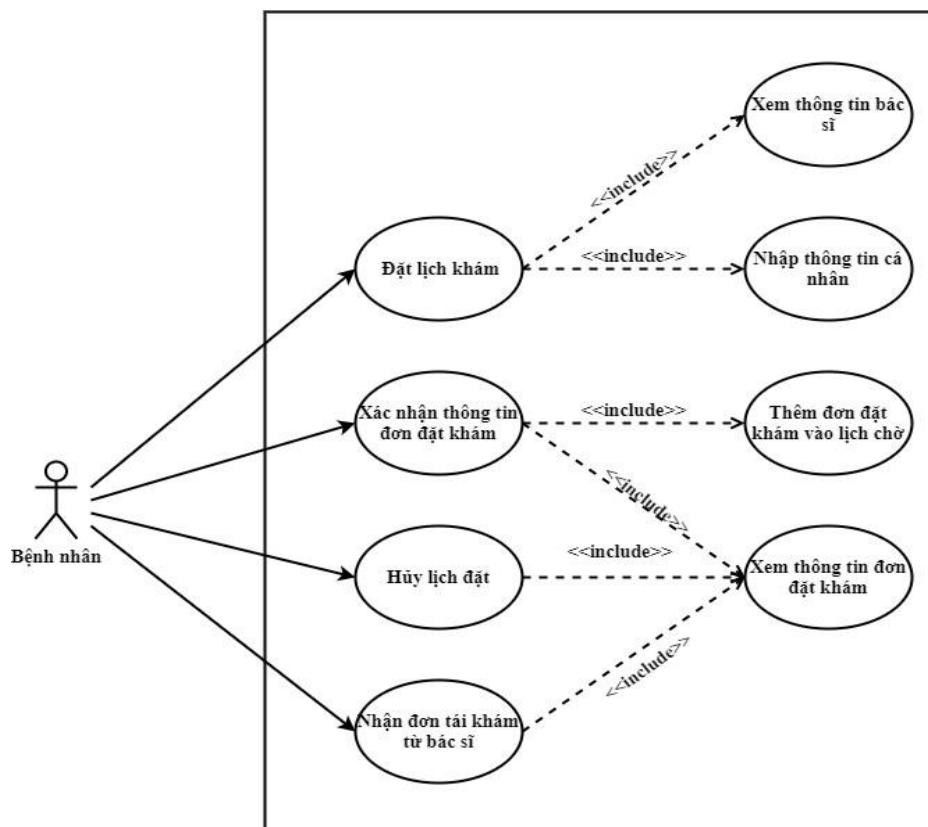
**Hình 5.** UC quản lý thông tin cá nhân.

- Gói UC quản lý lịch khám.



**Hình 6.** UC quản lý lịch khám.

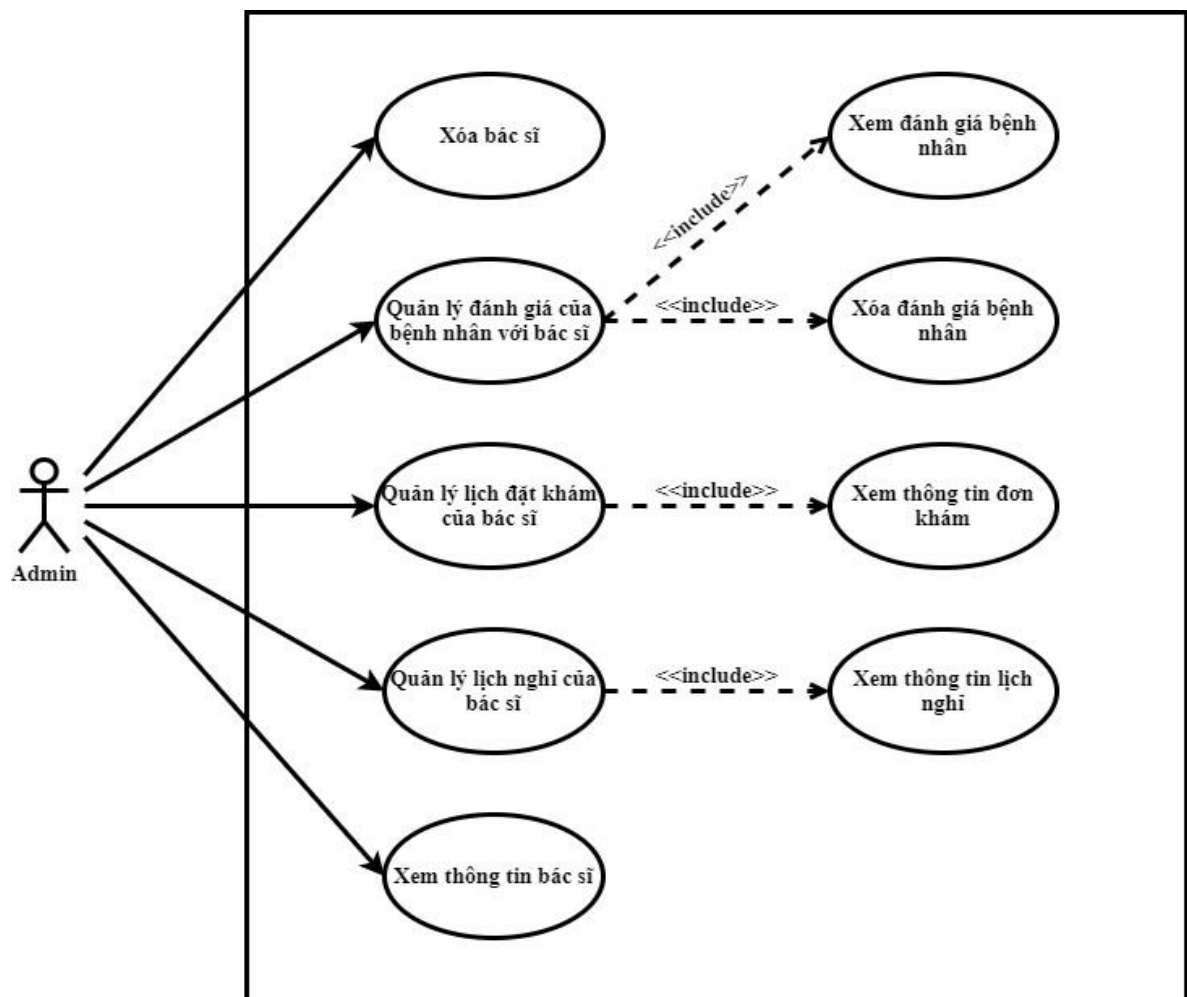
- Gói UC đặt lịch.



**Hình 7.** UC đặt lịch.

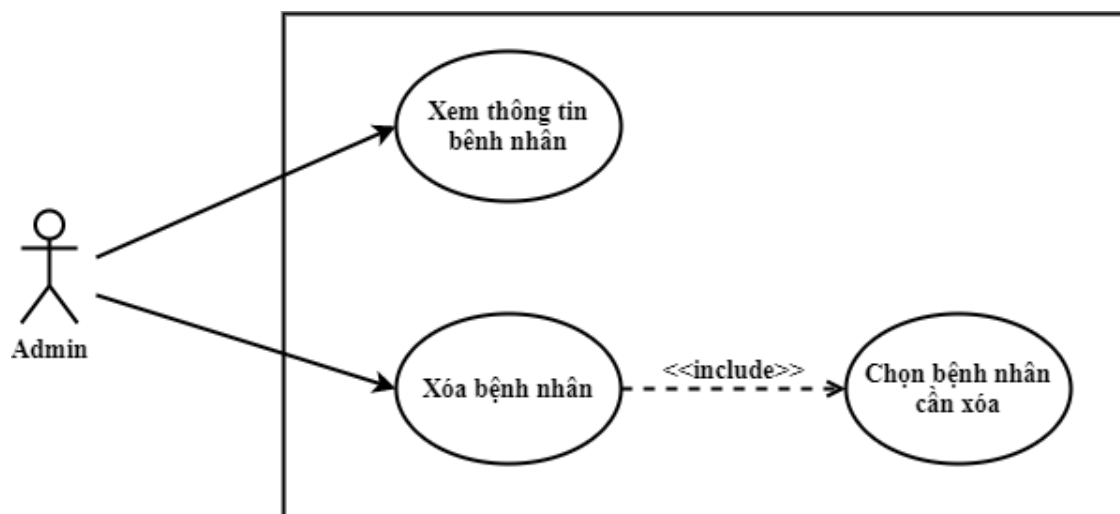


- Gói UC quản lý bác sĩ.



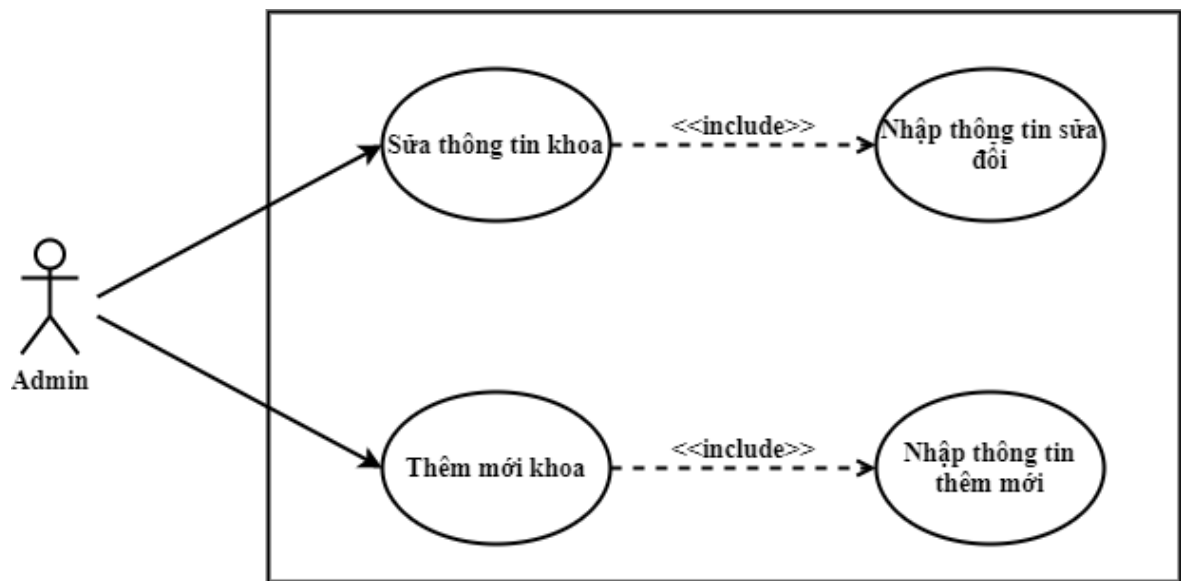
**Hình 8.** UC quản lý bác sĩ.

- Gói UC quản lý bệnh nhân.



**Hình 9.** UC quản lý bệnh nhân.

- Gói UC quản lý khoa.



**Hình 10.** UC quản lý khoa.

#### 4.3.4. Đặc tả Use Case

- Đăng nhập
  - Đặc tả UC đăng nhập

Tên Use Case	Đăng nhập
Actor	Bệnh nhân, Bác sĩ, Admin
Mô tả	UC cho phép thành viên đăng nhập vào hệ thống.
Tiền điều kiện	Thành viên chưa đăng nhập vào hệ thống
Hậu điều kiện	Người dùng đăng nhập thành công và có thể sử dụng các chức năng của hệ thống.
Luồng sự kiện chính	<p>Người dùng chọn chức năng đăng nhập.</p> <ol style="list-style-type: none"> <li>1. Form đăng nhập hiển thị.</li> <li>2. Nhập tên, mật khẩu vào form đăng nhập.</li> <li>3. Hệ thống kiểm tra tên, mật khẩu của người dùng.</li> <li>4. Nếu việc đăng nhập thành công thì người dùng có thể vào sử dụng các chức năng của hệ thống. Nếu nhập sai tên</li> </ol>

	<p>hoặc mật khẩu thì chuyển sang luồng nhánh A1.</p> <p>5. UC kết thúc</p>
Luồng sự kiện rẽ nhánh.	Luồng nhánh A1: Người dùng đăng nhập không thành công
	<ol style="list-style-type: none"> <li>1. Hệ thống thông báo quá trình đăng nhập không thành công.</li> <li>2. Chọn đăng ký hoặc nhập lại. Nếu chọn đăng ký thì chuyển sang luồng đăng ký.</li> <li>3. Hệ thống yêu cầu người dùng nhập lại tên và mật khẩu.</li> <li>4. Nếu người dùng đồng ý thì quay về bước 2 của luồng sự kiện chính, nếu không đồng ý thì UC kết thúc.</li> </ol>

**Bảng 1.** Đặc tả UC đăng nhập

- Quản lý thông tin cá nhân.
  - Đặc tả UC xem thông tin cá nhân

Tên Use Case	Xem thông tin cá nhân
Actor	Bệnh nhân, Bác sĩ
Mô tả	Cho phép bác sĩ, bệnh nhân xem thông tin cá nhân trên hệ thống
Tiền điều kiện	Thành viên đã đăng nhập vào hệ thống
Hậu điều kiện	Xem thông tin cá nhân trên hệ thống
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Thành viên chọn xem thông tin cá nhân trên hệ thống</li> <li>2. Form xem thông tin thành viên xuất hiện.</li> <li>3. Hệ thống hiển thị thông tin cá nhân của thành viên. Nếu thông tin cá nhân chưa được nhập đầy đủ thì thực hiện luồng rẽ nhánh A1.</li> <li>4. Hệ thống cung cấp liên kết để thành viên có thể sửa đổi thông tin cá nhân.</li> <li>5. UC kết thúc</li> </ol>
Luồng sự kiện rẽ nhánh.	Luồng nhánh A1: Thiếu thông tin cá nhân. <ol style="list-style-type: none"> <li>1. Hệ thống cho phép người dùng nhập thêm thông tin cá nhân</li> <li>2. Hệ thống yêu cầu lưu lại thông tin.</li> <li>3. Nếu thông tin nhập thêm phù hợp thì hệ thống lưu lại thông tin, nếu không phù hợp thì hệ thống thông báo lỗi.</li> <li>4. UC kết thúc</li> </ol>

**Bảng 3.** Đặc tả UC xem thông tin cá nhân

- Đặc tả UC sửa thông tin cá nhân.

Tên Use Case	Sửa thông tin cá nhân
Actor	Bệnh nhân, Bác sĩ
Mô tả	Cho phép bác sĩ, bệnh nhân sửa thông tin cá nhân trên hệ thống

Tiền điều kiện	Thành viên đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin thành viên thay đổi được lưu vào hệ thống
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Thành viên chọn chức năng sửa thông tin cá nhân trên hệ thống.</li> <li>2. Form thông tin thành viên xuất hiện.</li> <li>3. Thành viên nhập các thông tin mới.</li> <li>4. Chọn chức năng lưu thông tin.</li> </ol>
	<ol style="list-style-type: none"> <li>5. Nếu việc thực hiện thành công thì thực hiện bước 6. Nếu báo lỗi thực hiện luồng rẽ nhánh A1.</li> <li>6. Lưu thông tin.</li> <li>7. UC kết thúc.</li> </ol>
Luồng sự kiện rẽ nhánh.	<p>Luồng nhánh A1: Thông tin nhập không hợp lệ.</p> <ol style="list-style-type: none"> <li>1. Hệ thống thông báo việc nhập thông tin không hợp lệ.</li> <li>2. Thành viên nhập lại thông tin.</li> <li>3. Quay lại bước 4 của luồng sự kiện chính.</li> <li>4. UC kết thúc</li> </ol>

**Bảng 4.** *Đặc tả UC sửa thông tin cá nhân*

- Quản lý lịch khám.
  - Đặc tả UC thiết lập thời gian làm việc

Tên Use Case	Thiết lập thời gian làm việc
Actor	Bác sĩ
Mô tả	Cho phép bác sĩ cài đặt thời gian làm việc trên hệ thống.
Tiền điều kiện	Bác sĩ đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin thời gian được lưu vào hệ thống

Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Bác sĩ chọn chức năng cài đặt thời gian trên hệ thống.</li> <li>2. Form cài đặt thời gian xuất hiện.</li> <li>3. Bác sĩ thay đổi các thông tin về thời gian mặc định của hệ thống.</li> <li>4. Chọn chức năng lưu.</li> <li>5. Nếu việc thực hiện thành công thì thực hiện bước 6. Nếu báo lỗi thực hiện luồng rẽ nhánh A1.</li> <li>6. Lưu thông tin.</li> <li>7. UC kết thúc.</li> </ol>
Luồng sự kiện rẽ nhánh.	<p>Luồng nhánh A1: Thông tin thời gian nhập không hợp lệ.</p> <ol style="list-style-type: none"> <li>1. Hệ thống thông báo việc nhập thông tin không hợp lệ.</li> <li>2. Bác sĩ nhập lại thông tin.</li> <li>3. Quay lại bước 4 của luồng sự kiện chính.</li> <li>4. UC kết thúc</li> </ol>

**Bảng 5.** *Đặc tả UC thiết lập thời gian làm việc*

Đặc tả UC tạo lịch nghỉ

Tên Use Case	Tạo lịch nghỉ
Actor	Bác sĩ
Mô tả	Cho phép bác sĩ thêm lịch nghỉ trên hệ thống.
Tiền điều kiện	Bác sĩ đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin lịch nghỉ được lưu vào hệ thống
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Bác sĩ chọn chức năng thêm thời gian nghỉ trên hệ thống.</li> <li>2. Form thêm thời gian nghỉ xuất hiện.</li> <li>3. Bác sĩ thay đổi các thông tin về thời gian mặc định của hệ thống.</li> <li>4. Chọn chức năng lưu.</li> <li>5. Nếu việc thực hiện thành công thì thực hiện bước 6. Nếu báo lỗi thực hiện luồng rẽ nhánh A1.</li> <li>6. Lưu thông tin.</li> </ol>

	7. UC kết thúc.
Luồng sự kiện rẽ nhánh.	<p>Luồng nhánh A1: Thông tin thời gian thay đổi không hợp lệ.</p> <ol style="list-style-type: none"> <li>1. Hệ thống thông báo việc nhập thông tin không hợp lệ.</li> <li>2. Bác sĩ nhập lại thông tin thời gian nghỉ.</li> <li>3. Quay lại bước 4 của luồng sự kiện chính.</li> <li>4. UC kết thúc</li> </ol>

**Bảng 6.** *Đặc tả UC tạo lịch nghỉ*

- Đặc tả UC tạo lịch tái khám cho bệnh nhân

Tên Use Case	Tạo lịch tái khám cho bệnh nhân
Actor	Bác sĩ
Mô tả	Cho phép bác sĩ tạo lịch tái khám cho bệnh nhân trên hệ thống.
Tiền điều kiện	Bác sĩ đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin lịch tái khám được lưu vào hệ thống và gửi thông báo đến cho bệnh nhân.
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Bác sĩ chọn bệnh nhân trên danh sách và chọn chức năng tạo lịch tái khám.</li> <li>2. Form tạo lịch tái khám bệnh nhân xuất hiện.</li> <li>3. Bác sĩ nhập các thông tin về thời gian tái khám và số lần tái khám của bệnh nhân.</li> <li>4. Chọn chức năng tạo lịch tái khám.</li> <li>5. Nếu việc thực hiện thành công thì thực hiện bước 6. Nếu báo lỗi thực hiện luồng rẽ nhánh A1.</li> </ol>

	6. Tạo lịch tái khám. 7. UC kết thúc.
Luồng sự kiện rẽ nhánh.	Luồng nhánh A1: Thông tin tạo lịch không hợp lệ.  1. Hệ thống thông báo việc nhập thông tin không hợp lệ. 2. Bác sĩ nhập lại thông tin tạo lịch tái khám. 3. Quay lại bước 4 của luồng sự kiện chính. 4. UC kết thúc

**Bảng 7.** Đặc tả UC tạo lịch tái khám cho bệnh nhân

- Đặc tả UC xóa lịch đặt

Tên Use Case	Xóa lịch đặt
Actor	Bác sĩ
Mô tả	Cho phép bác sĩ xóa lịch đặt khám của bệnh nhân trên hệ thống.
Tiền điều kiện	Bác sĩ đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin lịch đặt khám được xóa khỏi hệ thống và thông báo cho bệnh nhân lịch đã bị xóa.
Luồng sự kiện chính	1. Bác sĩ chọn lịch đặt của bệnh nhân cần xóa. 2. Chọn chức năng xóa lịch đặt. 3. Xóa lịch đặt khám. 4. UC kết thúc.
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 8.** Đặc tả UC xóa lịch đặt



- Đặt lịch
  - Đặc tả UC đặt lịch khám

Tên Use Case	Đặt lịch khám
Actor	Bệnh nhân
Mô tả	Cho phép bệnh nhân chọn lịch đặt khám trên hệ thống.
Tiền điều kiện	Bệnh nhân đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin lịch đặt được chuyển đến form xác nhận.
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Bệnh nhân chọn bác sĩ cần đặt lịch và thực hiện chức năng đặt lịch.</li> <li>2. Form đặt lịch bệnh nhân xuất hiện.</li> <li>3. Bệnh nhân nhập các thông tin cần để đặt lịch khám.</li> <li>4. Chọn chức năng đặt lịch khám.</li> <li>5. Nếu việc thực hiện thành công thì thực hiện bước 6. Nếu báo lỗi thực hiện luồng rẽ nhánh A1.</li> <li>6. Tạo lịch đặt khám.</li> <li>7. UC kết thúc.</li> </ol>
Luồng sự kiện rẽ nhánh.	Luồng nhánh A1: Thông tin đặt lịch không hợp lệ. <ol style="list-style-type: none"> <li>1. Hệ thống thông báo việc nhập thông tin không hợp lệ.</li> <li>2. Bệnh nhân nhập lại thông tin đặt lịch khám.</li> <li>3. Quay lại bước 4 của luồng sự kiện chính.</li> <li>4. UC kết thúc</li> </ol>

**Bảng 9.** Đặc tả UC đặt lịch khám

- Đặc tả UC xác nhận thông tin đơn đặt khám

Tên Use Case	Xác nhận thông tin đơn đặt khám
Actor	Bệnh nhân
Mô tả	Cho phép bệnh nhân xác nhận thông tin lịch đặt khám trên hệ thống.

Tiền điều kiện	Bệnh nhân đã đặt lịch khám trên hệ thống.
Hậu điều kiện	Thông tin lịch đặt được lưu vào hệ thống và thông báo thông tin lịch đặt cho bệnh nhân.
Luồng sự kiện chính	1. Bệnh nhân kiểm tra lại thông tin đặt lịch. 2. Thực xác nhận 3. UC kết thúc.
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 10.** Đặc tả UC xác nhận thông tin đơn đặt khám

- Đặc tả UC hủy lịch đặt

Tên Use Case	Hủy lịch đặt
Actor	Bệnh nhân
Mô tả	Cho phép bệnh nhân hủy lịch đặt khám trên hệ thống.
Tiền điều kiện	Bệnh nhân đã đặt lịch trên hệ thống
Hậu điều kiện	Thông tin lịch đặt sẽ được loại bỏ khỏi hệ thống.
Luồng sự kiện chính	1. Bệnh nhân chọn lịch đã đặt trong danh sách lịch sử đặt lịch 2. Thực hiện chức năng hủy lịch đặt. 3. Thông báo xác nhận hủy lịch thành công. 4. UC kết thúc.
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 11.** Đặc tả UC hủy lịch đặt

- Đặc tả UC nhận đơn tái khám

Tên Use Case	Nhận đơn tái khám
Actor	Bệnh nhân
Mô tả	Cho phép bệnh nhân xem thông tin lịch tái khám từ bác sĩ.
Tiền điều kiện	Bệnh nhân đã đặt lịch khám trên hệ thống
Hậu điều kiện	Thông tin lịch đặt được chuyển đến form xác nhận.

Luồng sự kiện chính	1. Bệnh nhân nhận thông báo về lịch tái khám. 2. Thực hiện chức năng xem lịch tái khám. 3. Xem thông tin lịch tái khám 4. UC kết thúc
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 12.** Đặt tả UC nhận đơn tái khám

- Quản lý bệnh nhân.
  - Đặc tả UC xem thông tin bệnh nhân

Tên Use Case	Xem thông tin bệnh nhân
Actor	Admin
Mô tả	Cho phép admin xem thông tin bệnh nhân trên hệ thống.
Tiền điều kiện	Admin đã đăng nhập vào hệ thống
Hậu điều kiện	Xem thông tin bệnh nhân trên hệ thống.
Luồng sự kiện chính	1. Admin chọn xem thông tin bệnh nhân trên hệ thống 2. Form xem thông tin bệnh nhân xuất hiện. 3. UC kết thúc
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 13.** Đặt tả UC Xem thông tin bệnh nhân

- Đặc tả UC xóa thông tin bệnh nhân

Tên Use Case	Xóa thông tin bệnh nhân
Actor	Admin
Mô tả	Cho phép admin xóa thông tin bệnh nhân trên hệ thống.
Tiền điều kiện	Admin đã đăng nhập vào hệ thống
Hậu điều kiện	Xóa thông tin bệnh nhân trên hệ thống.

Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Admin chọn bệnh nhân cần xóa trên hệ thống.</li> <li>2. Thực hiện chức năng xóa bệnh nhân.</li> <li>3. Thông tin bệnh nhân đã được xóa.</li> <li>4. UC kết thúc.</li> </ol>
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 14.** Đặc tả UC xóa thông tin bệnh nhân

- Quản lý bác sĩ.
  - Đặc tả UC xóa bác sĩ

Tên Use Case	Xóa thông tin bác sĩ
Actor	Admin
Mô tả	Cho phép admin xóa thông tin bác sĩ trên hệ thống.
Tiền điều kiện	Admin đã đăng nhập vào hệ thống
Hậu điều kiện	Xóa thông tin bác sĩ trên hệ thống.
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Admin chọn bác sĩ cần xóa trên hệ thống.</li> <li>2. Thực hiện chức năng xóa bác sĩ.</li> <li>3. Thông tin bác sĩ đã được xóa.</li> <li>4. UC kết thúc.</li> </ol>
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 15.** Đặc tả UC xóa bác sĩ

- Đặc tả UC quản lý đánh giá của bệnh nhân với bác sĩ

Tên Use Case	Quản lý đánh giá của bệnh nhân với bác sĩ
Actor	Admin
Mô tả	Cho phép bác sĩ xem, xóa đánh giá của bệnh nhân.
Tiền điều kiện	Admin đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin đánh giá thay đổi được lưu vào cơ sở dữ liệu

Luồng sự kiện chính	1. người quản lý chọn kiểu tác động lên đánh giá bệnh nhân. A. Xem thông tin đánh giá. 1.1 Admin chọn xem thông tin bình luận trên hệ thống 1.2 Form xem thông tin bình luận xuất hiện. B. Xóa thông tin đánh giá. 1.1 Admin chọn đánh giá cần xóa. 1.2 Nhấn nút xóa để thực hiện xóa đánh giá. 1.3 Thông báo bình luận đã được xóa. 1.4 hệ thống hiển thị lại danh sách đánh giá. 2. UC kết thúc
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 16.** *Đặc tả UC quản lý đánh giá của bệnh nhân với bác sĩ*

- Đặc tả UC quản lý lịch đặt khám của bác sĩ

Tên Use Case	Quản lý lịch đặt khám của bác sĩ
Actor	Admin
Mô tả	Cho phép admin xem, xóa lịch đặt khám của bác sĩ.
Tiền điều kiện	Admin đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin thay đổi của lịch đặt được lưu vào cơ sở dữ liệu
Luồng sự kiện chính	1. Người quản lý chọn kiểu tác động lên lịch đặt bệnh nhân. A. Xem thông tin lịch đặt khám. 1.1 Admin chọn xem thông tin lịch đặt khám trên hệ thống 1.2 Form xem thông tin lịch đặt khám xuất hiện. B. Xóa thông tin lịch đặt khám. 1.1 Admin chọn lịch đặt khám cần xóa. 1.2 Nhấn nút xóa để thực hiện xóa đánh giá. 1.3 Thông báo lịch đặt khám đã được xóa. 1.4 Hệ thống hiển thị lại danh sách lịch đặt khám. 2. UC kết thúc

Luồng sự kiện rẽ nhánh.	Không có
-------------------------	----------

**Bảng 17.** Đặc tả UC quản lý lịch đặt khám của bác sĩ

- Đặc tả UC xem lịch nghỉ của bác sĩ

Tên Use Case	Xem lịch nghỉ của bác sĩ
Actor	Admin
Mô tả	Cho phép admin xem thông tin lịch nghỉ bác sĩ trên hệ thống.
Tiền điều kiện	Admin đã đăng nhập vào hệ thống
Hậu điều kiện	Xem thông tin lịch nghỉ của bác sĩ trên hệ thống.
Luồng sự kiện chính	1. Admin chọn xem thông tin lịch nghỉ của bác sĩ trên hệ thống 2. Form xem thông tin lịch nghỉ của bác sĩ xuất hiện. 3. UC kết thúc
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 18.** Đặc tả UC xem lịch nghỉ của bác sĩ

- Đặc tả UC xem thông tin bác sĩ

Tên Use Case	Xem thông tin bác sĩ
Actor	Admin
Mô tả	Cho phép admin xem thông tin bác sĩ trên hệ thống.
Tiền điều kiện	Admin đã đăng nhập vào hệ thống
Hậu điều kiện	Xem thông tin bác sĩ trên hệ thống.
Luồng sự kiện chính	1. Admin chọn xem thông tin bác sĩ trên hệ thống 2. Form xem thông tin bác sĩ xuất hiện. 3. UC kết thúc
Luồng sự kiện rẽ nhánh.	Không có

**Bảng 19.** Đặt tả UC xem thông tin bác sĩ

- Quản lý khoa.
  - Đặc tả UC sửa thông tin khoa

Tên Use Case	Sửa thông tin khoa
Actor	Admin
Mô tả	Cho phép admin sửa thông tin khoa trên hệ thống
Tiền điều kiện	Admin đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin khoa thay đổi được lưu vào hệ thống
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>Admin chọn chức năng sửa thông tin khoa trên hệ thống.</li> <li>Form thông tin khoa xuất hiện.</li> <li>Admin nhập các thông tin mới.</li> <li>Chọn chức năng lưu thông tin.</li> <li>Nếu việc thực hiện thành công thì thực hiện bước 6. Nếu báo lỗi thực hiện luồng rẽ nhánh A1.</li> <li>Lưu thông tin.</li> <li>UC kết thúc.</li> </ol>
Luồng sự kiện rẽ nhánh.	Luồng nhánh A1: Thông tin nhập không hợp lệ. <ol style="list-style-type: none"> <li>Hệ thống thông báo việc nhập thông tin không hợp lệ.</li> <li>Thành viên nhập lại thông tin.</li> <li>Quay lại bước 4 của luồng sự kiện chính.</li> <li>UC kết thúc</li> </ol>

**Bảng 20.** Đặc tả UC sửa thông tin khoa

- Đặc tả UC thêm mới khoa

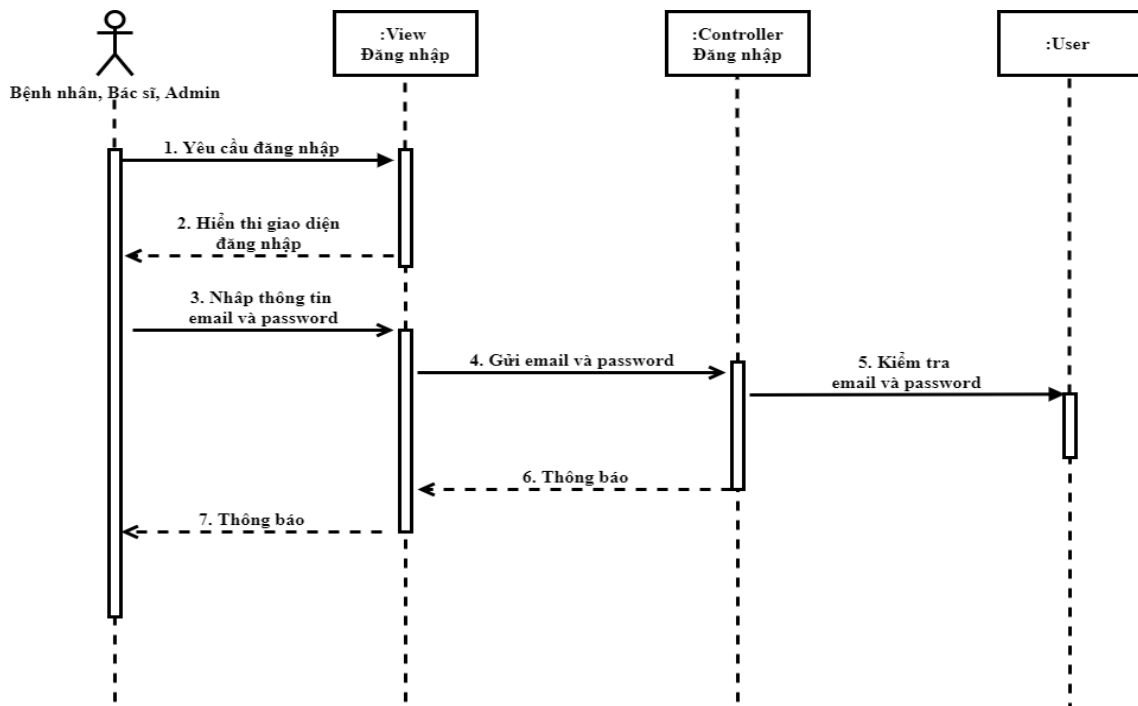
Tên Use Case	Thêm mới thông tin khoa
Actor	Admin
Mô tả	Cho phép admin thêm mới thông tin khoa trên hệ thống
Tiền điều kiện	Admin đã đăng nhập vào hệ thống
Hậu điều kiện	Thông tin khoa thay đổi được lưu vào hệ thống

Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Admin chọn chức năng thêm thông tin khoa trên hệ thống.</li> <li>2. Nhập thông tin thêm mới vào form thêm mới khoa.</li> <li>3. Chọn chức năng thêm thông tin.</li> <li>4. Nếu việc thực hiện thành công thì thực hiện bước 6. Nếu báo lỗi thực hiện luồng rẽ nhánh A1.</li> <li>5. Thêm thông tin.</li> <li>6. UC kết thúc.</li> </ol>
Luồng sự kiện rẽ nhánh.	<p>Luồng nhánh A1: Thông tin nhập không hợp lệ.</p> <ol style="list-style-type: none"> <li>1. Hệ thống thông báo việc nhập thông tin không hợp lệ.</li> <li>2. Admin nhập lại thông tin.</li> <li>3. Quay lại bước 4 của luồng sự kiện chính.</li> <li>4. UC kết thúc</li> </ol>

**Bảng 21.** Đặc tả UC thêm mới khoa

#### 4.4. Biểu đồ tuần tự

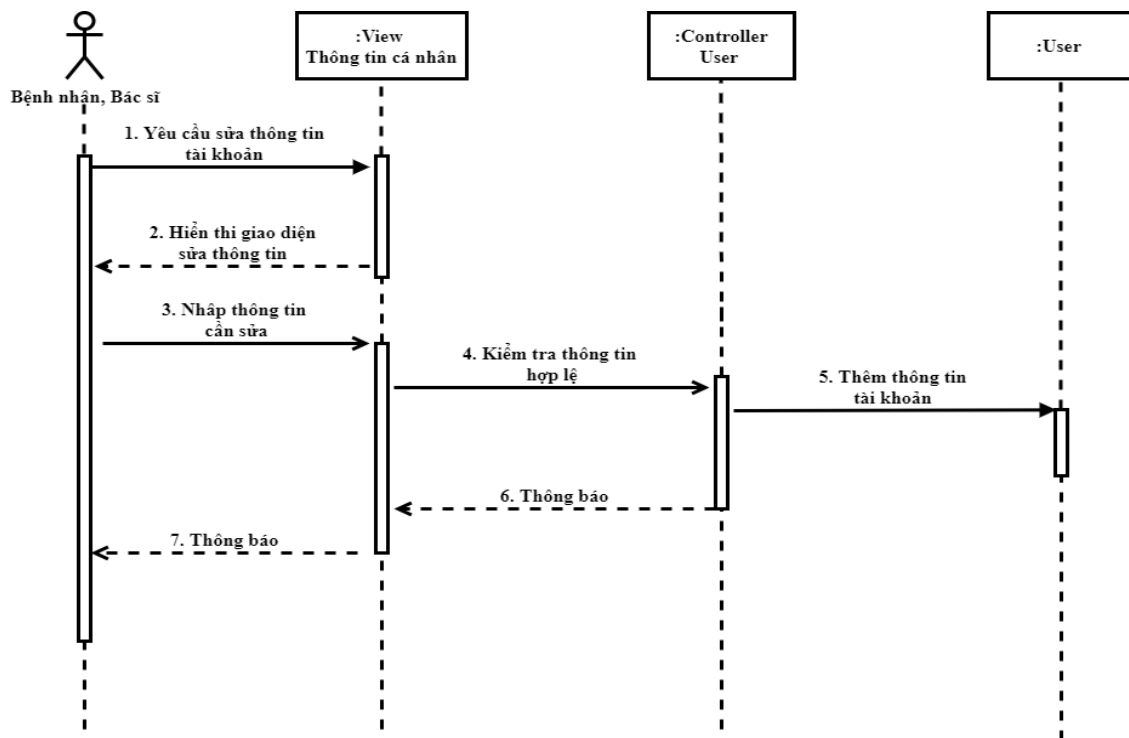
- Biểu đồ tuần tự đăng nhập



**Hình 12.** Biểu đồ tuần tự đăng nhập

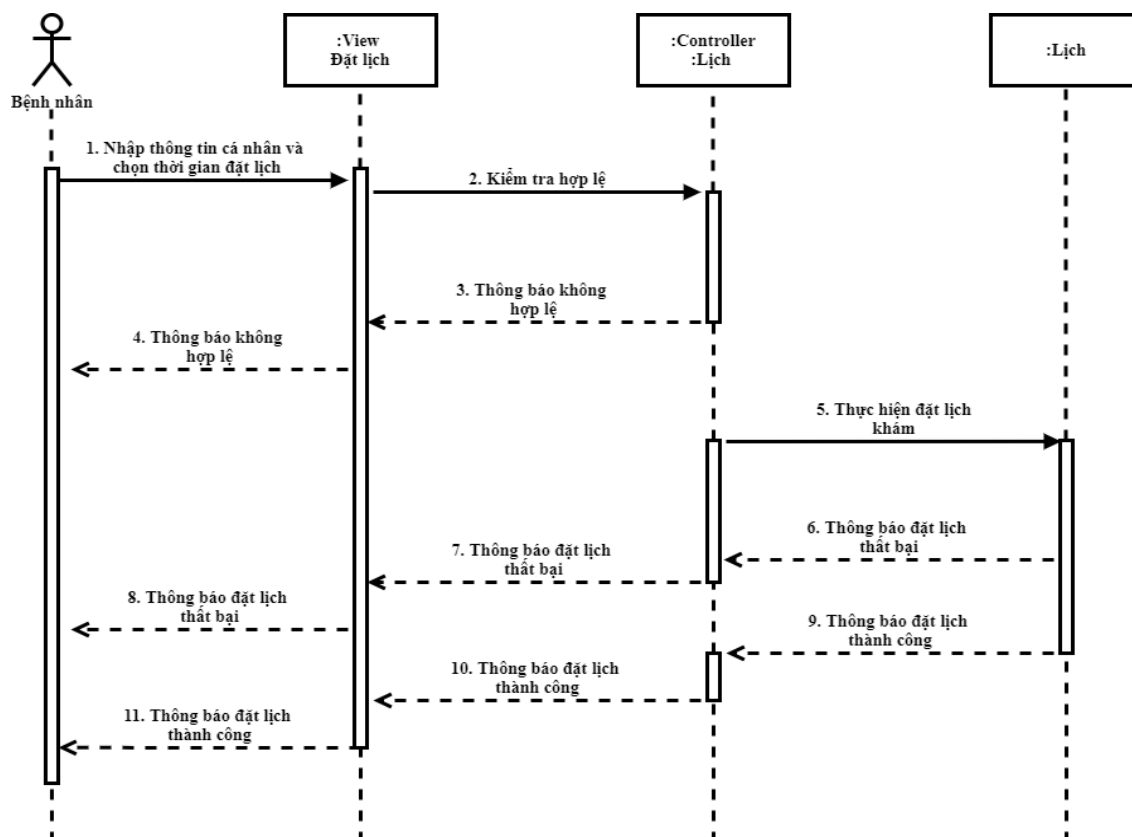


- Biểu đồ tuần tự sửa thông tin cá nhân



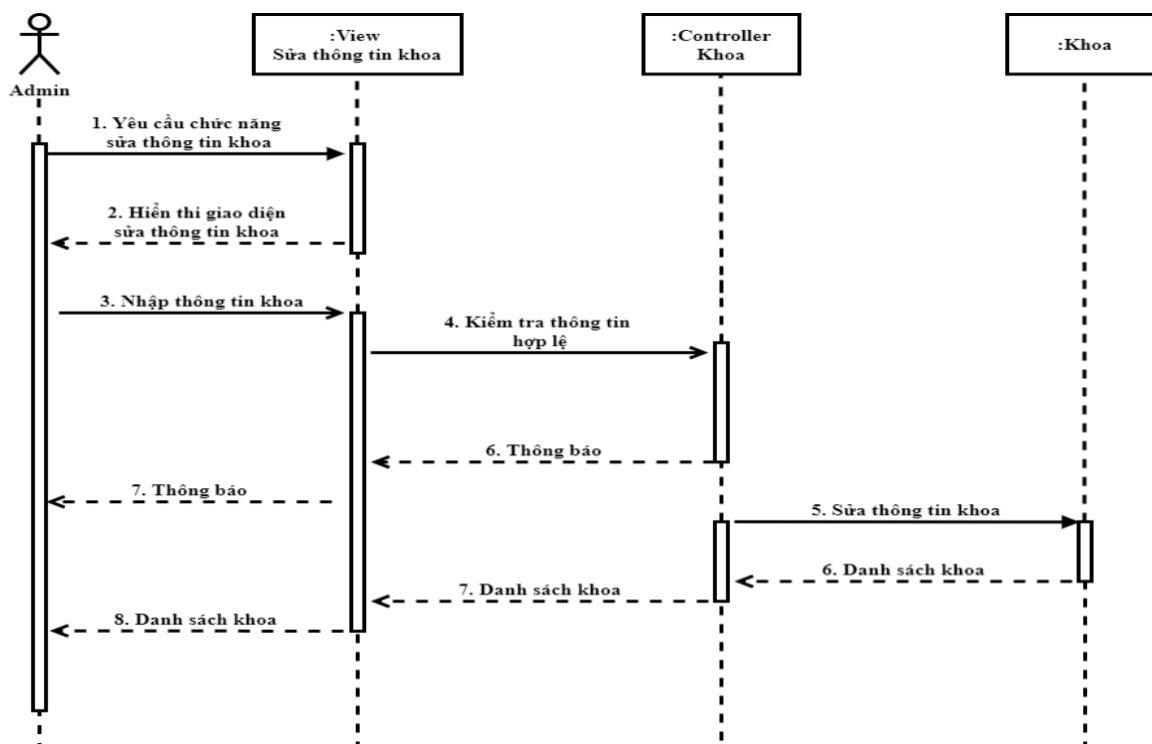
**Hình 13.** Biểu đồ tuần tự sửa thông tin cá nhân

- Biểu đồ tuần tự đặt lịch



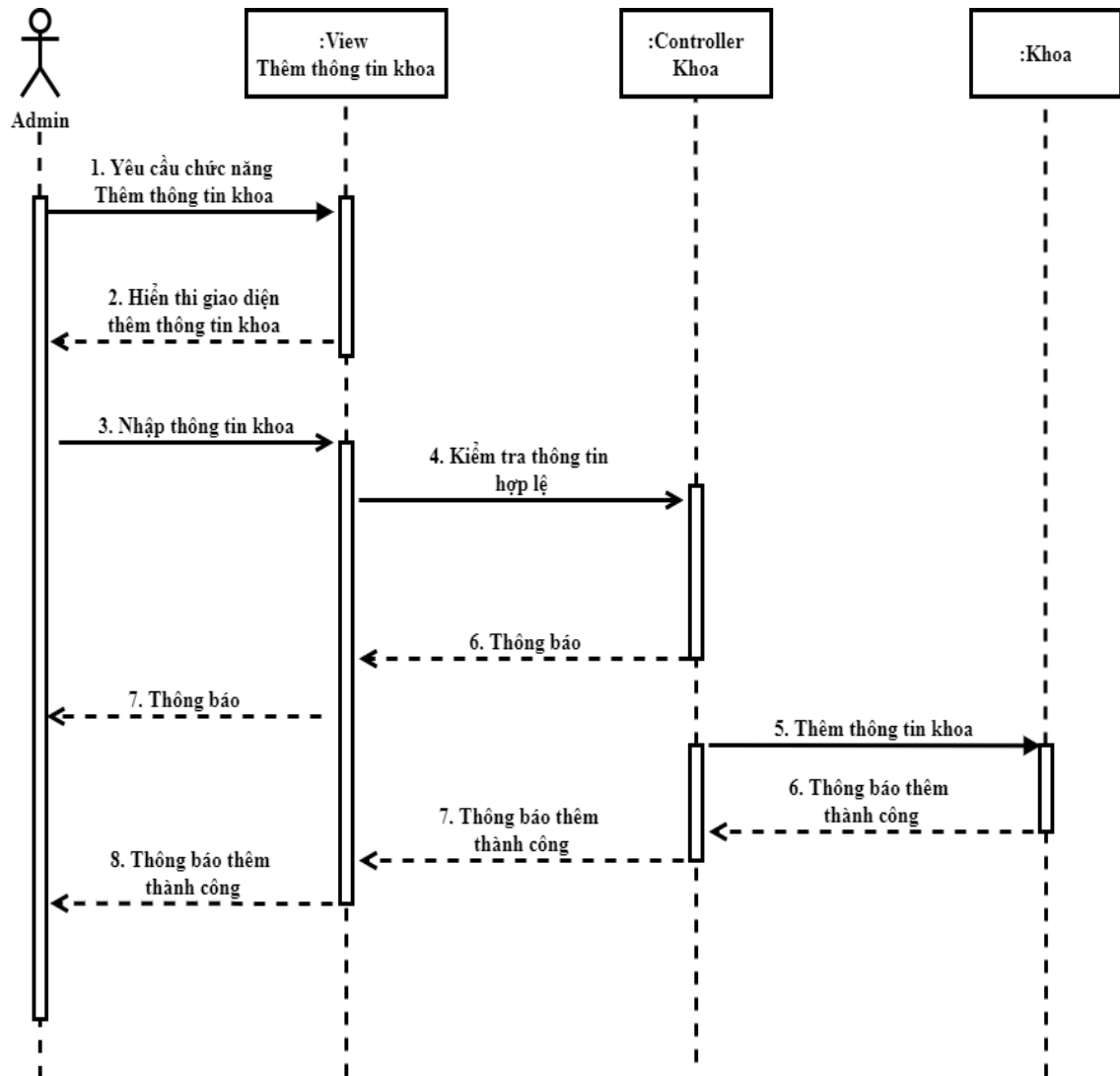
**Hình 14.** Biểu đồ tuần tự đặt lịch.

- Biểu đồ tuần tự sửa thông tin khoa.



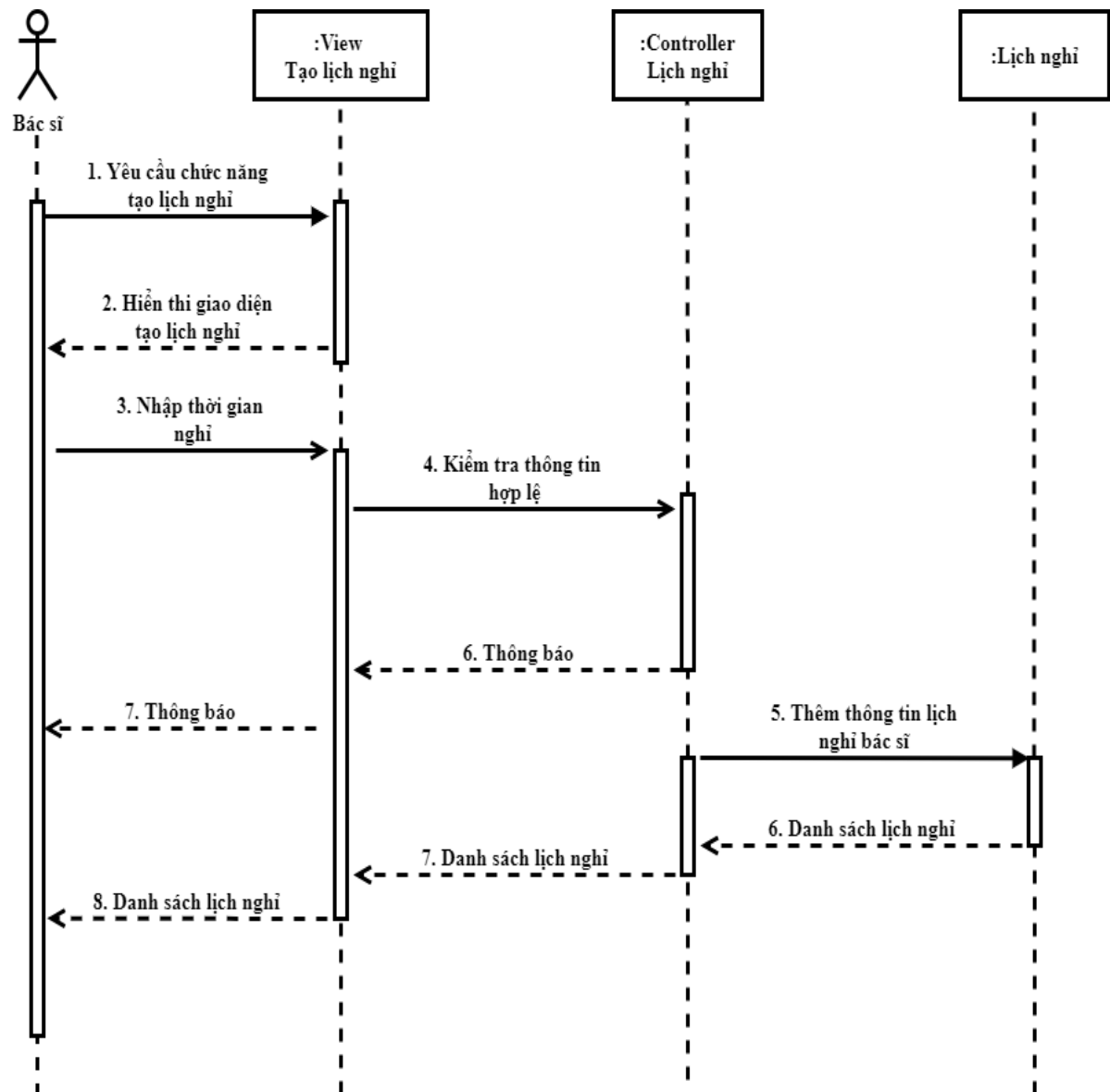
**Hình 15.** Biểu đồ tuần tự sửa thông tin khoa.

- Biểu đồ tuần tự thêm thông tin khoa



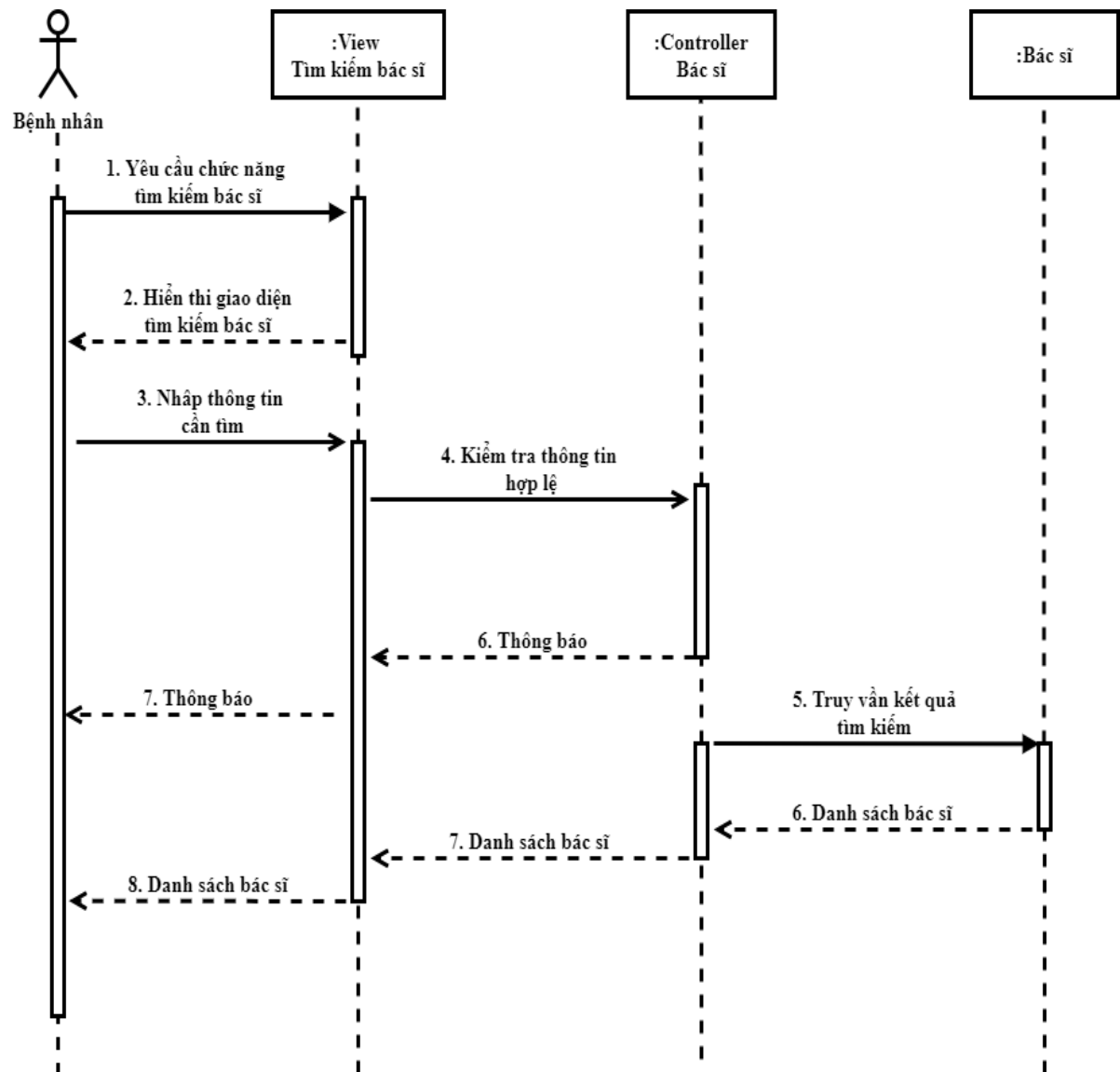
**Hình 16.** Biểu đồ tuần tự thêm thông tin khoa

- Biểu đồ tuần tự tạo lịch nghỉ bác sĩ.



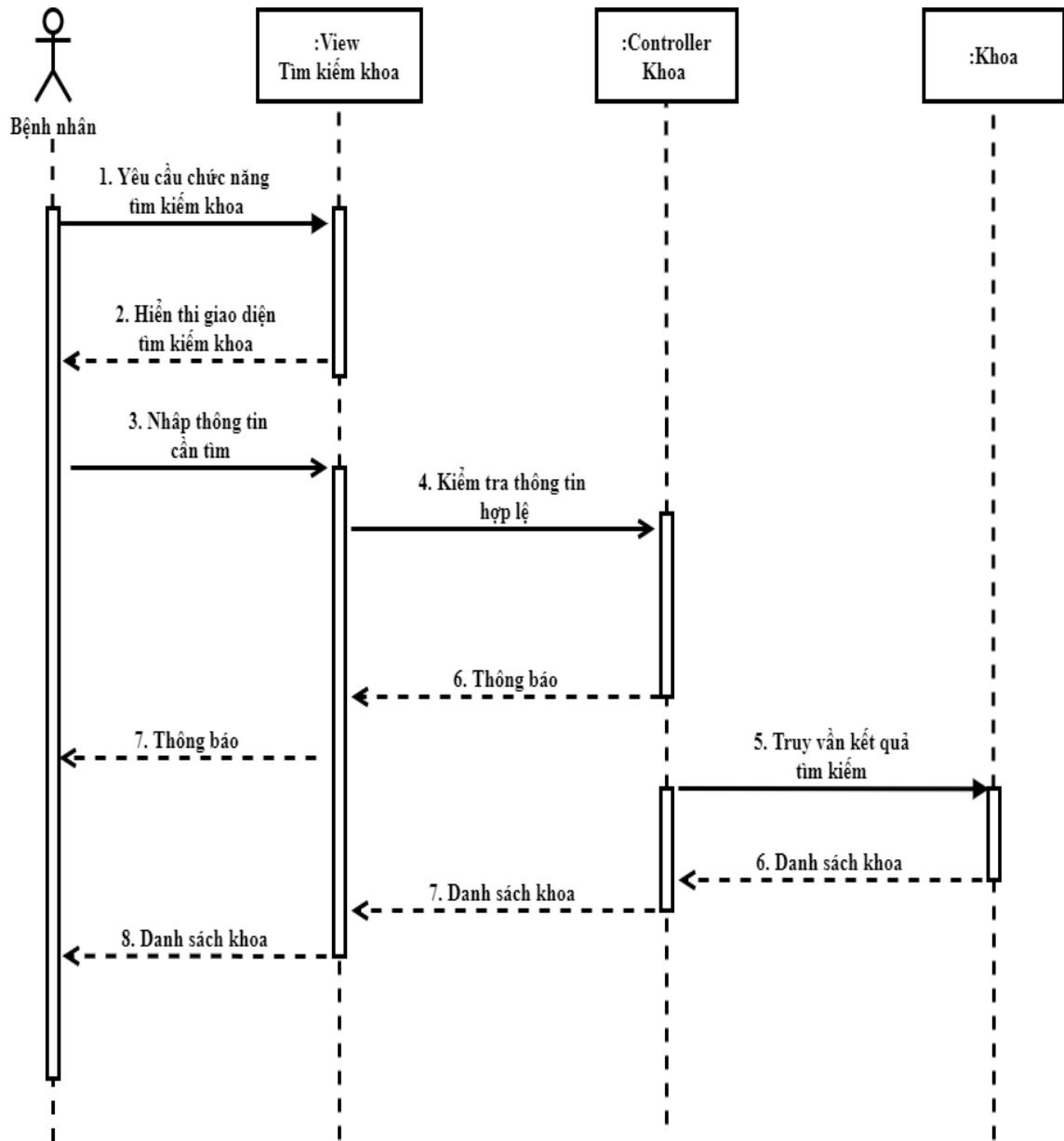
**Hình 17.** Biểu đồ tuần tự tạo lịch nghỉ bác sĩ.

- Biểu đồ tuần tự tìm kiếm bác sĩ



**Hình 18.** Biểu đồ tuần tự tìm kiếm bác sĩ.

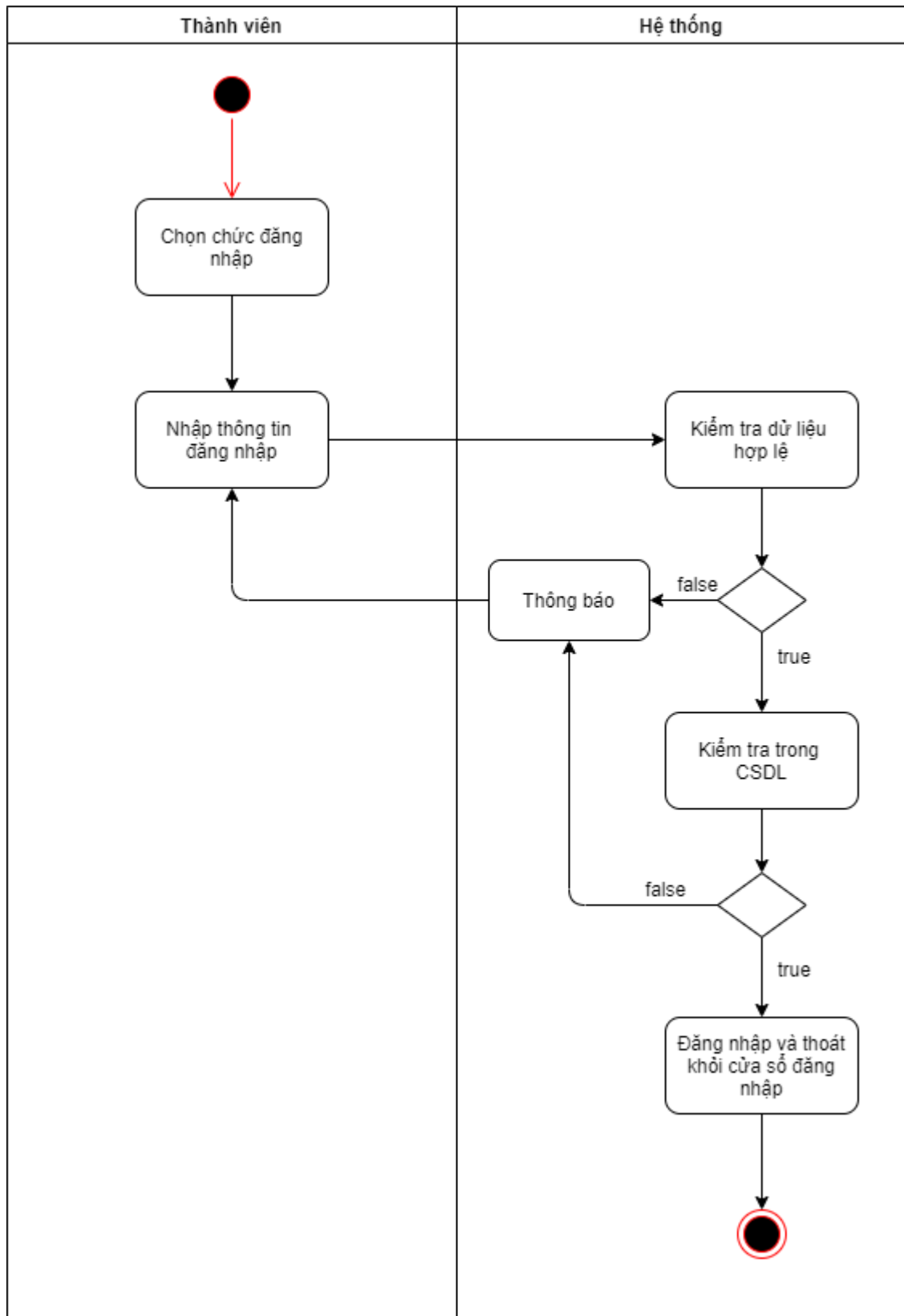
- Biểu đồ tuần tự tìm kiếm khoa



**Hình 19.** Biểu đồ tuần tự tìm kiếm khoa.

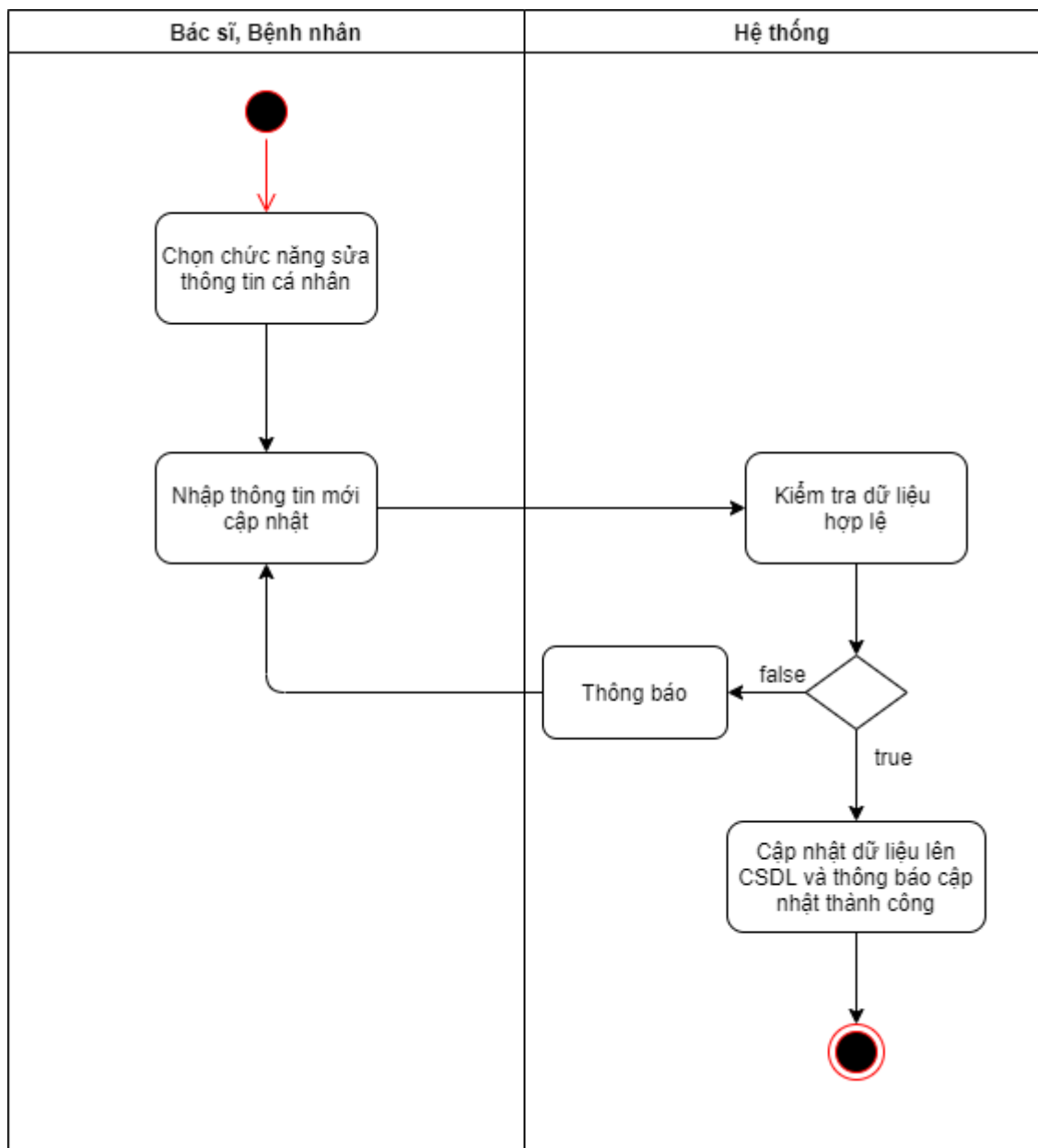
#### 4.5. Biểu đồ hoạt động

- Biểu đồ hoạt động đăng nhập.



**Hình 20.** Biểu đồ hoạt động đăng nhập

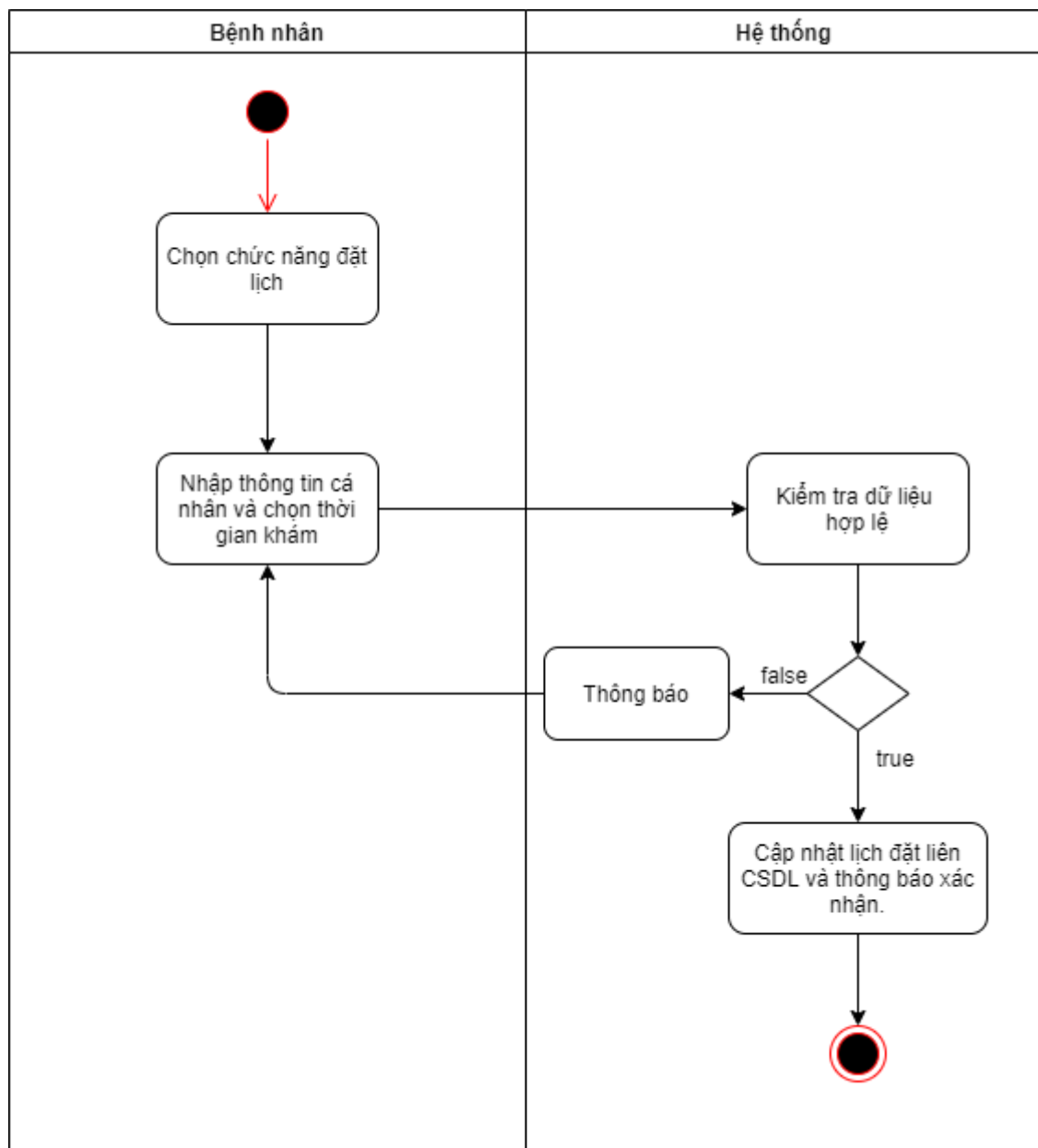
- Biểu đồ hoạt động sửa thông tin cá nhân



**Hình 22.** Biểu đồ hoạt động sửa thông tin cá nhân

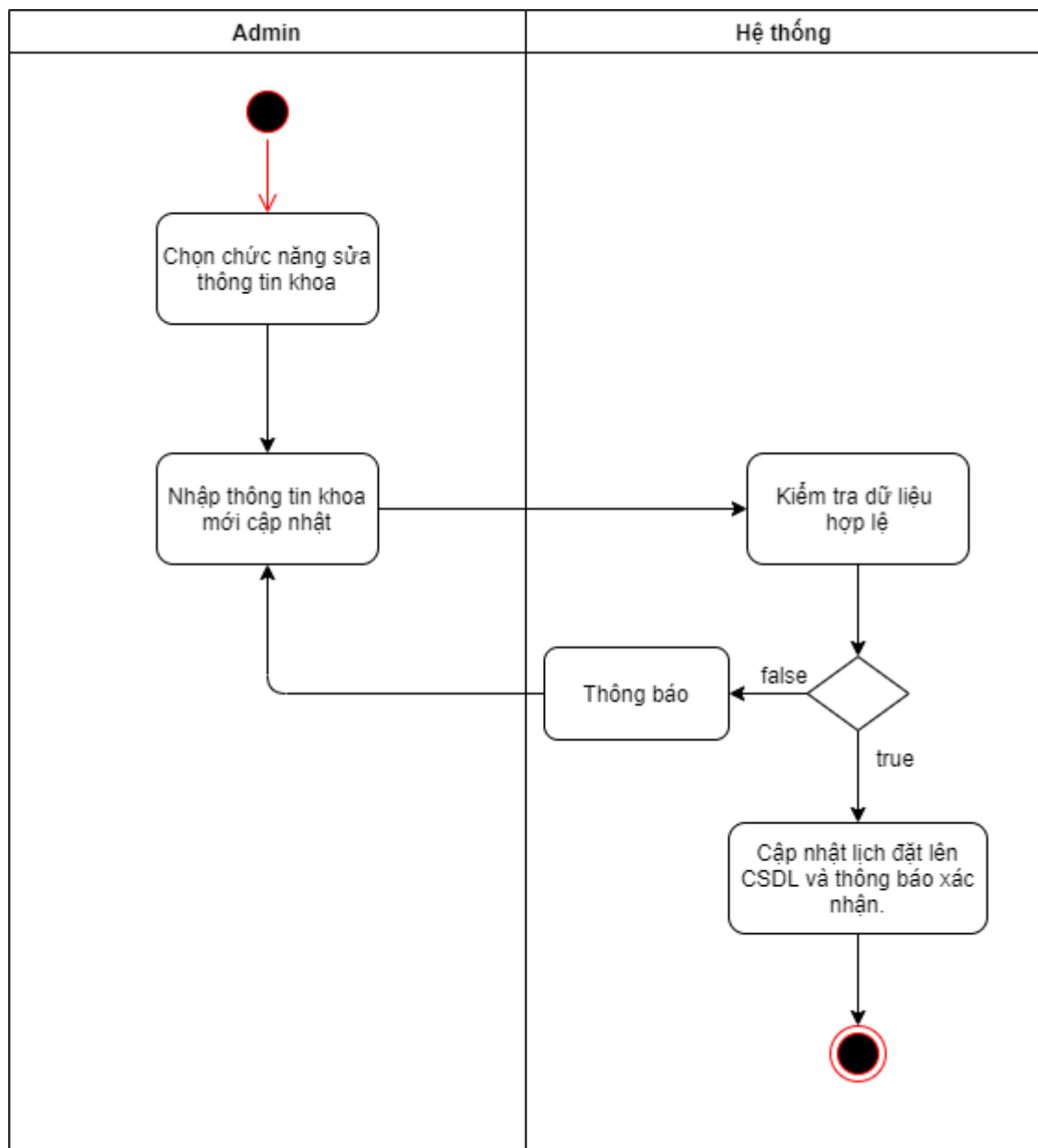


- Biểu đồ hoạt động đặt lịch



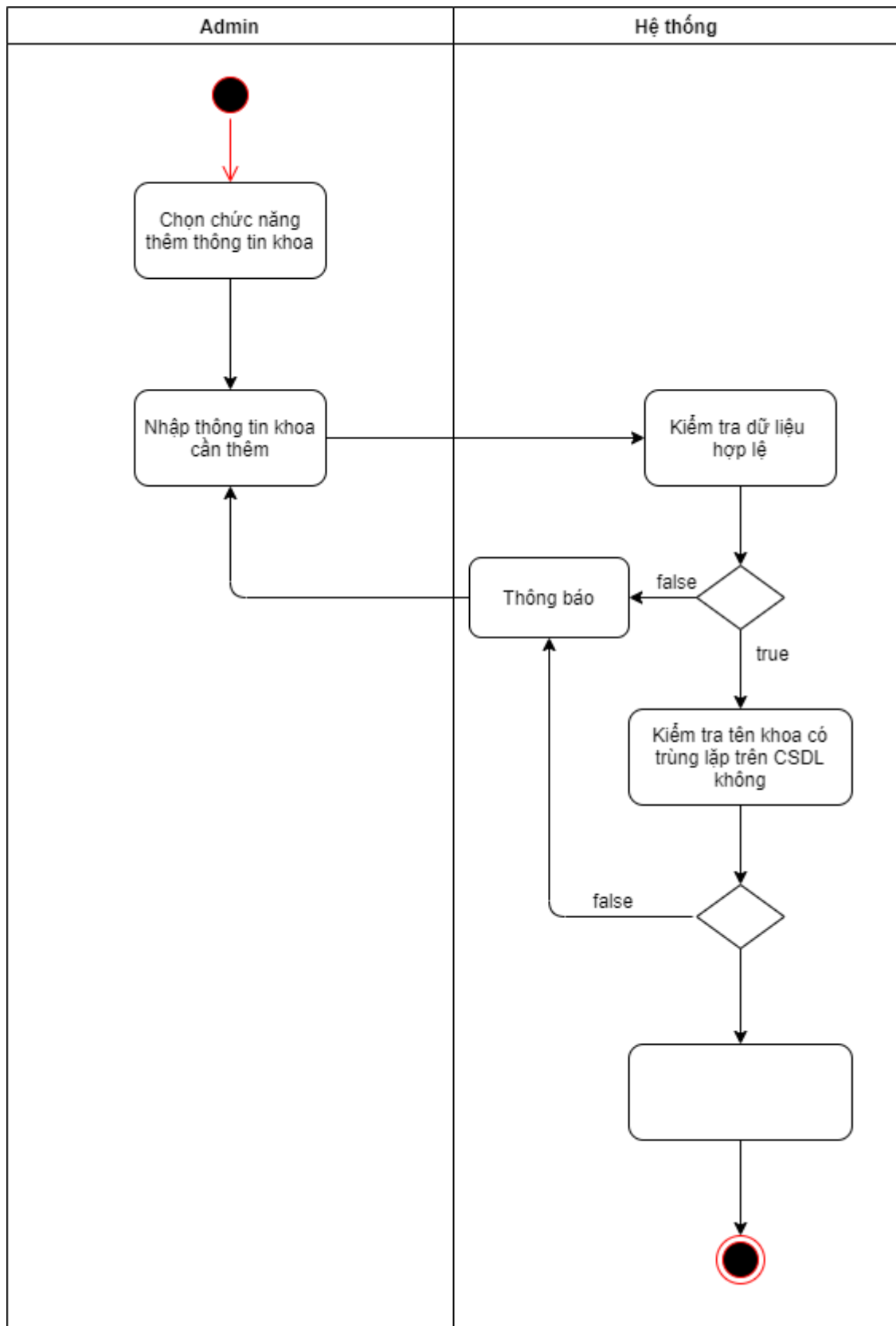
**Hình 23.** Biểu đồ hoạt động đặt lịch

- Biểu đồ hoạt động sửa thông tin khoa.



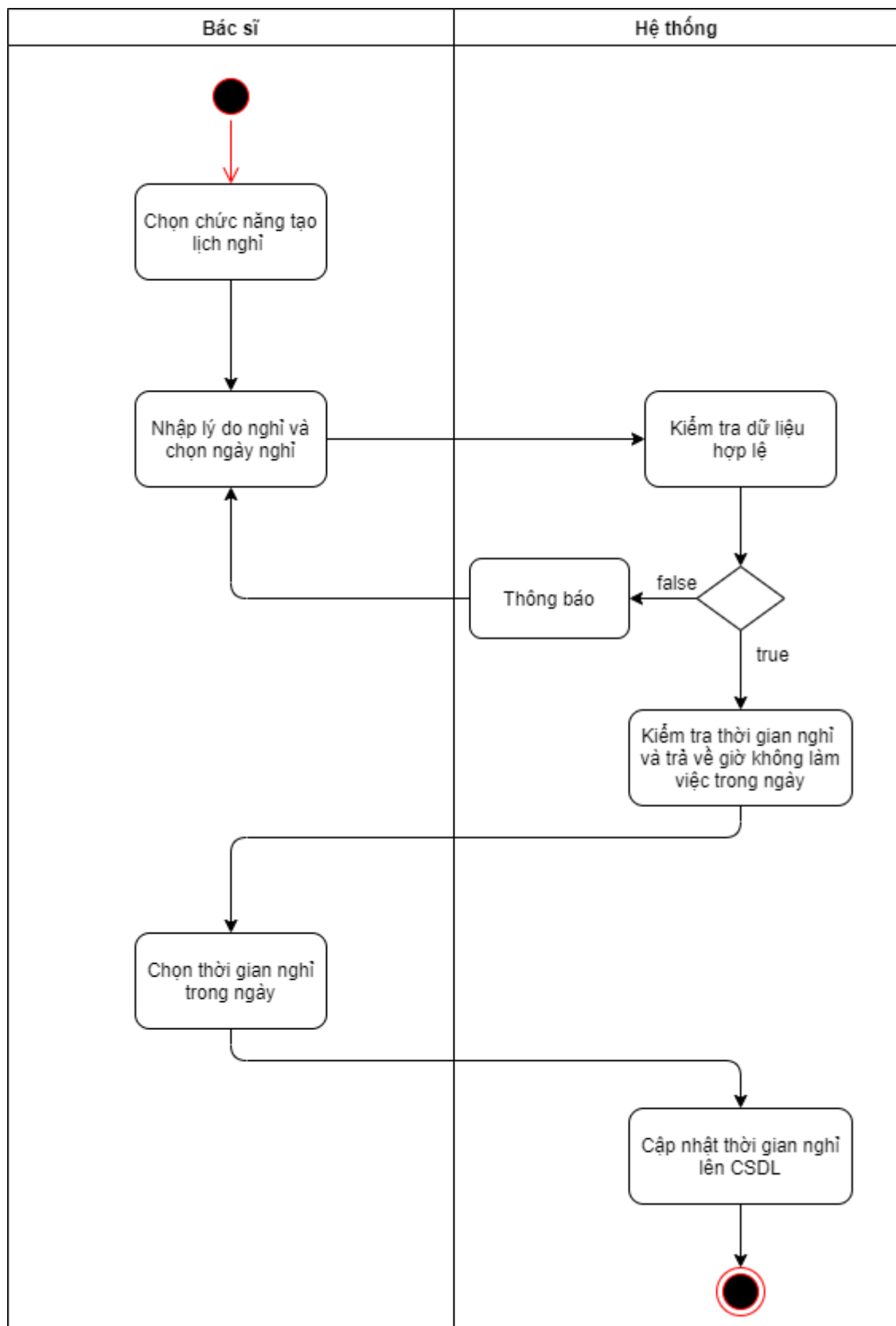
**Hình 24.** Biểu đồ hoạt động sửa thông tin khoa.

- Biểu đồ hoạt động thêm thông tin khoa.



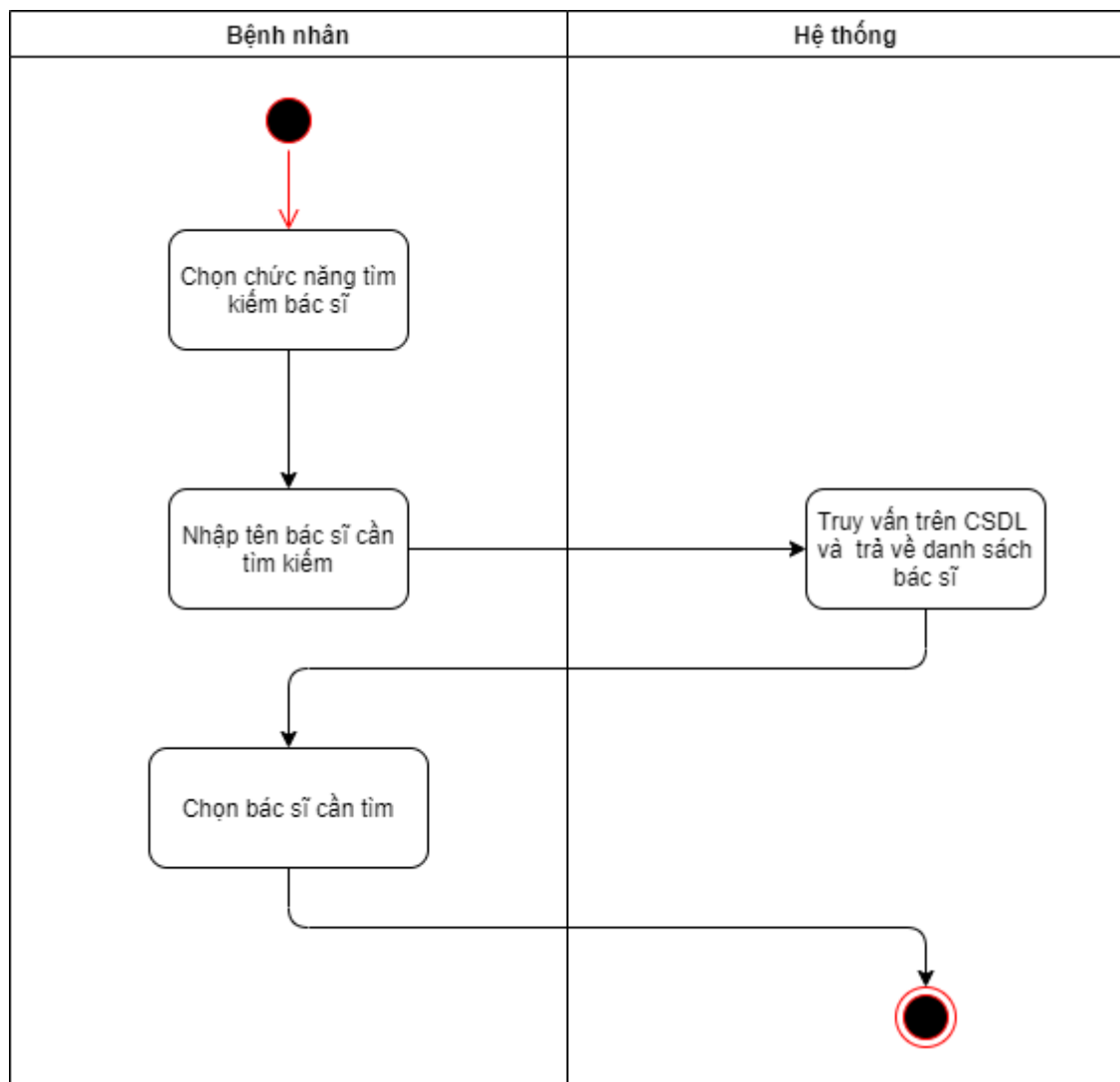
**Hình 25.** Biểu đồ hoạt động thêm thông tin khoa.

- Biểu đồ hoạt động tạo lịch nghỉ



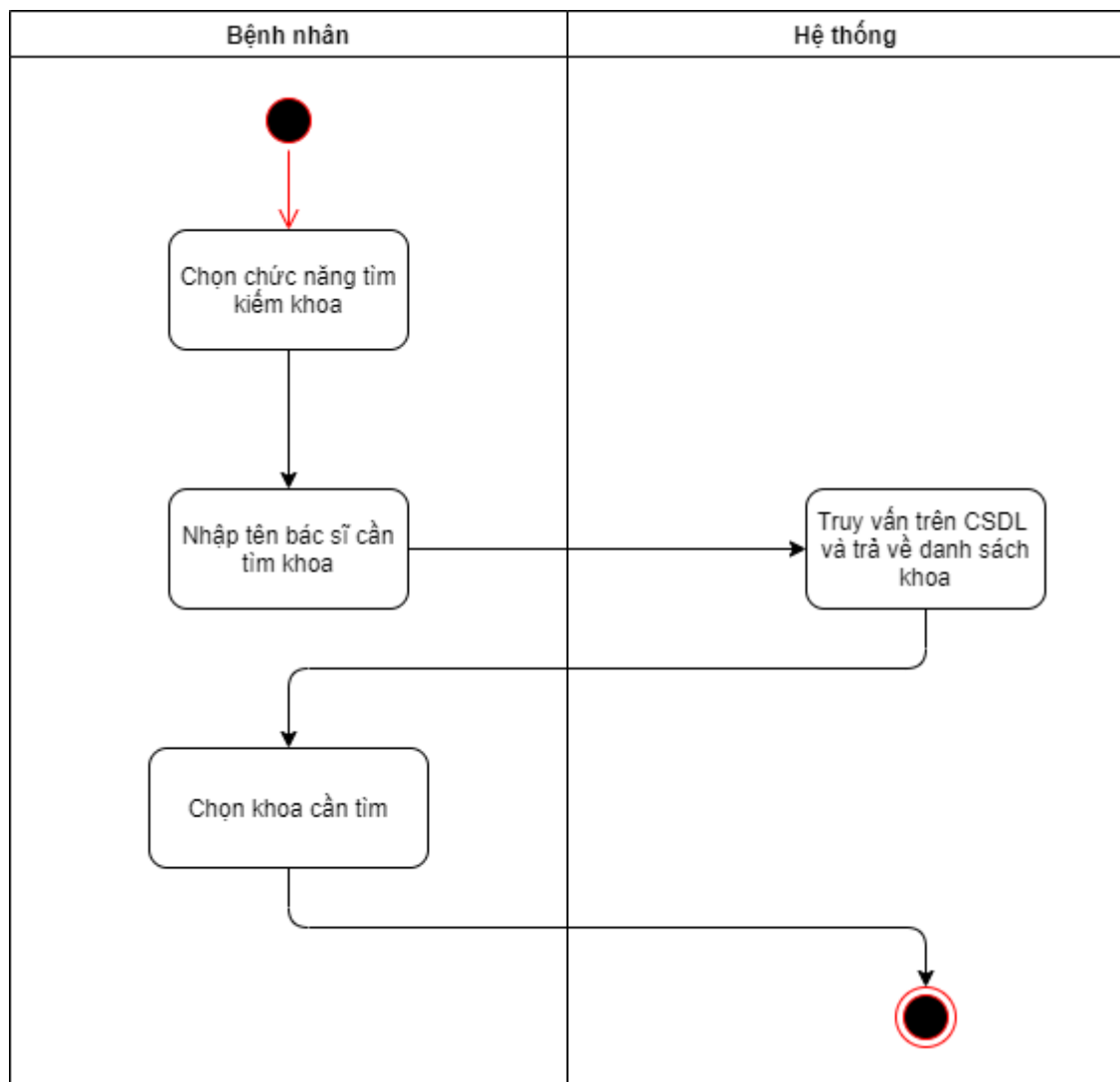
***Hình 26.*** Biểu đồ hoạt động tạo lịch nghỉ

- Biểu đồ hoạt động tìm kiếm bác sĩ



**Hình 27.** Biểu đồ hoạt động tìm kiếm bác sĩ

- Biểu đồ hoạt động tìm kiếm khoa.

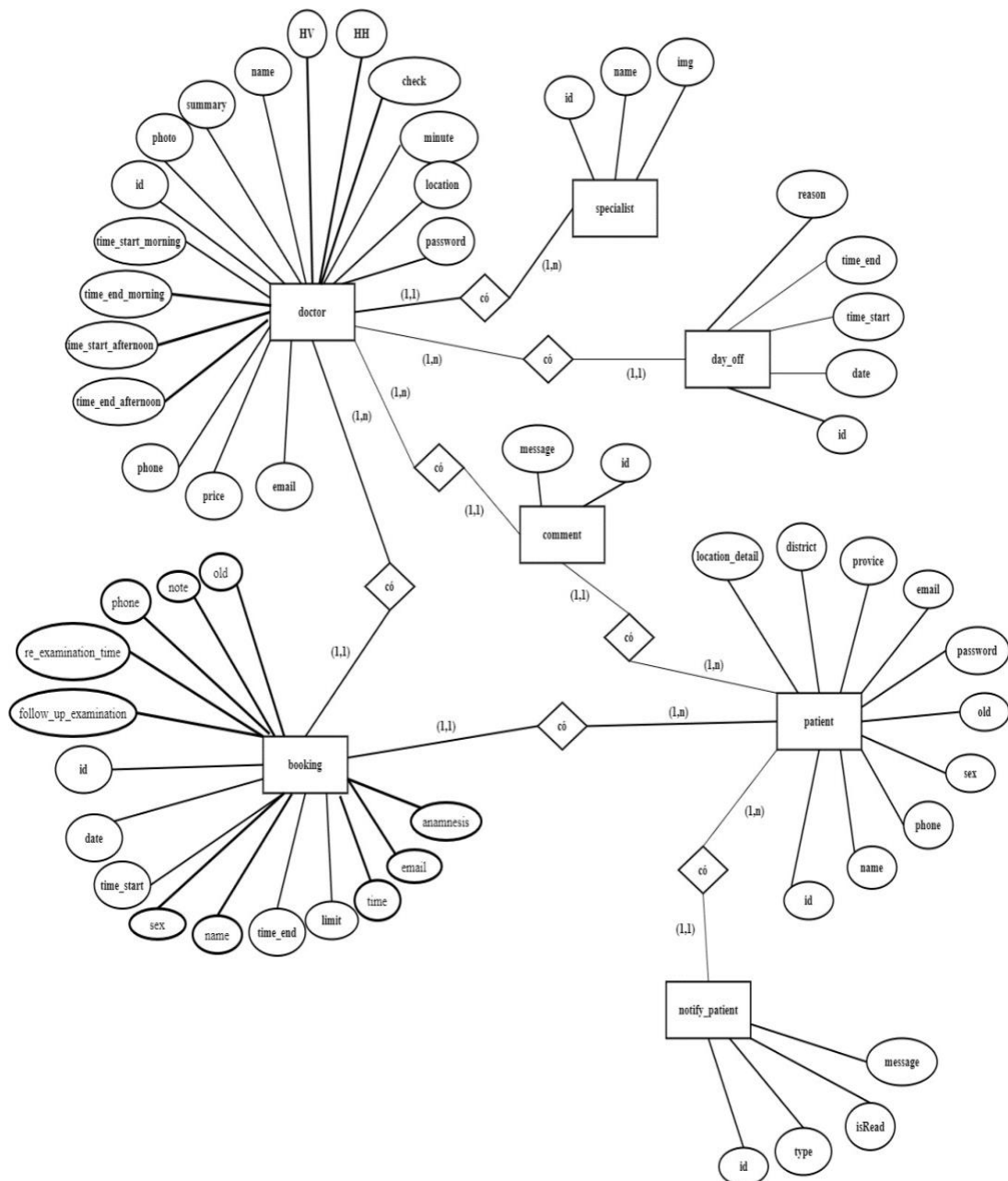


**Hình 28.** Biểu đồ hoạt động tìm kiếm khoa.

#### 4.6. Mô hình E-R

**Mô hình E – R mô tả các mối quan hệ sau:**

- Quan hệ giữa thực thể doctor với thực thể specialist.
- Quan hệ giữa thực thể Booking với doctor.
- Quan hệ giữa thực thể patient với Booking.
- Quan hệ giữa thực thể patient, doctor với comment.
- Quan hệ giữa thực thể day\_off với doctor.

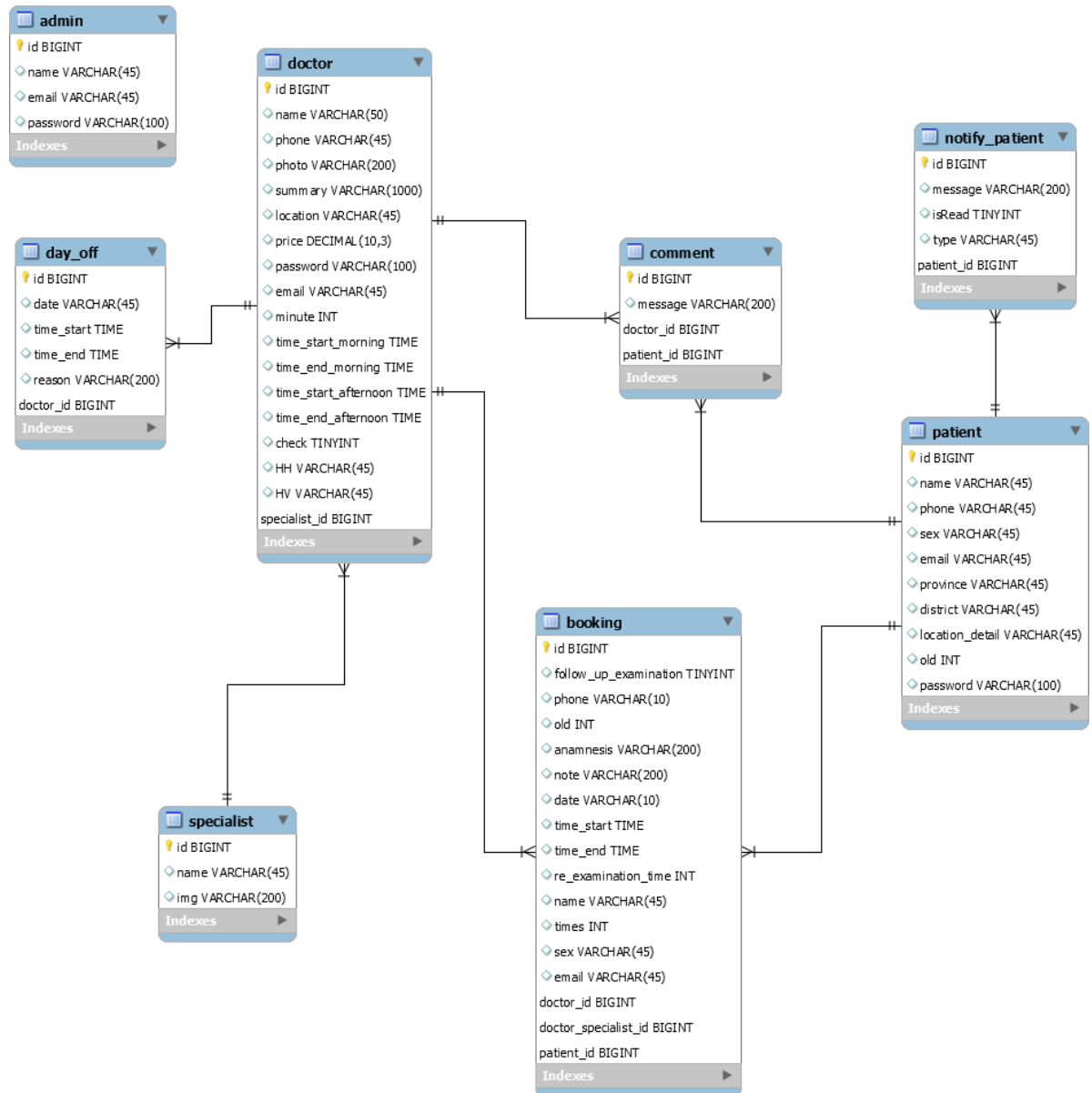


- Quan hệ giữa thực thể patient với notify\_patient.

**Hình 29.** Mô hình E-R.

#### 4.7. Mô hình quan hệ dữ liệu

Mô hình quan hệ dữ liệu thể hiện mối quan hệ của các bảng trong cơ sở dữ liệu. Kết quả cuối cùng của quá trình phân tích hệ thống thông tin.



**Hình 30.** Mô hình quan hệ dữ liệu.



#### 4.8. Thiết kế hệ cơ sở dữ liệu vật lý

**Doctor:** id, name, phone, photo, summary, location, price, password, email, minute, time\_start\_morning, time\_end\_morning, time\_start\_afternoon, time\_end\_afternoon, check, HH, HV, specialist\_id.

Tên cột	Kiểu dữ liệu	Khóa	Mô tả
id	BIGINT(20)	PK	Mã lịch
name	VARCHAR(50)		Tên bác sĩ
phone	VARCHAR(45)		Số điện thoại bác sĩ
photo	VARCHAR(200)		Hình đại diện cho bác sĩ
summary	VARCHAR(100)		Tóm tắt thông tin bác sĩ
location	VARCHAR(45)		Địa chỉ bác sĩ
password	VARCHAR(100)		Mật khẩu
price	DECIMAL(10,3)		Giá khám
email	VARCHAR(45)		Email đăng nhập bác sĩ
minute	INT		Phút khám trong một lượt
time_start_morning	TIME		Thời gian bắt đầu làm việc vào buổi sáng của bác sĩ
time_end_morning	TIME		Thời gian kết thúc làm việc vào buổi sáng của bác sĩ
time_start_afternoon	TIME		Thời gian bắt đầu làm việc vào buổi chiều của bác sĩ
time_end_afternoon	TIME		Thời gian kết thúc làm việc vào buổi chiều của bác sĩ
check	TINYINT		Xác thực bác sĩ

specialist_id	BIGINT(20)	FK	Mã về thông tin chuyên khoa của bác sĩ
---------------	------------	----	---

**Bảng 22.** Bảng Doctor

**DOCTOR:** id, follow\_up\_examination, phone, old, anamnesis, note, date, time\_start, time\_end, re\_examination, name, times, sex, email, doctor\_id, patient\_id.

Tên cột	Kiểu dữ liệu	Khóa	Mô tả
id	BIGINT(20)	PK	Mã lịch đặt
follow_up_examination	TINYINT		Trạng thái lịch
phone	VARCHAR(10)		Số điện thoại bệnh nhân
old	INT		Tuổi bệnh nhân
anamnesis	VARCHAR(200)		Tiền sử về bệnh của bệnh nhân
note	VARCHAR(200)		Ghi chú về bệnh nhân
date	VARCHAR(10)		Ngày khám
time_start	TIME		Thời gian bắt đầu khám
time_end	TIME		Thời gian kết thúc khám
re_examination	INT		Ngày tái khám lại
name	VARCHAR(45)		Tên bệnh nhân
times	INT		Số lần tái khám
sex	VARCHAR(45)		Giới tính bệnh nhân
email	VARCHAR(45)		Email bệnh nhân
doctor_id	BIGINT(20)	FK	Mã bác sĩ
patient_id	BIGINT(20)	FK	Mã bệnh nhân

**Bảng 23.** Bảng DOCTOR

**Patient:** id, name, phone, sex, email, province, district, location, old, password.

Tên cột	Kiểu dữ liệu	Khóa	Mô tả
id	BIGINT(20)	PK	Mã bệnh nhân
name	VARCHAR(45)		Tên bệnh nhân
phone	VARCHAR(45)		Số điện thoại bệnh nhân
sex	VARCHAR(45)		Giới tính bệnh nhân
email	VARCHAR(45)		Email bệnh nhân
province	VARCHAR(45)		Thông tin về địa chỉ tỉnh của bệnh nhân
district	VARCHAR(45)		Thông tin về địa chỉ huyện của bệnh nhân
location	VARCHAR(45)		Địa chỉ nhà chi tiết của bệnh nhân
old	INT		Tuổi bệnh nhân
password	VARCHAR(100)		Mật khẩu bệnh nhân

**Bảng 24.** Bảng Patient

**Notify\_patient:** id, message, isReal, type, patient\_id

Tên cột	Kiểu dữ liệu	Khóa	Mô tả
id	BIGINT(20)	PK	Mã thông báo
message	VARCHAR(200)		Thông điệp của thông báo
isReal	TINYINT		Trạng thái kiểm tra đọc thông báo
type	VARCHAR(45)		Kiểu thông báo
patient_id	BIGINT(20)	FK	Mã bệnh nhân

**Bảng 25.** Bảng Notify\_patient

**Comment:** id, message, doctor\_id, patient\_id.

Tên cột	Kiểu dữ liệu	Khóa	Mô tả
id	BIGINT(20)	PK	Mã comment
message	VARCHAR(200)		Thông điệp bình luận
doctor_id	BIGINT(20)		Mã bác sĩ
patient_id	BIGINT(20)		Mã bệnh nhân

**Bảng 26.** Bảng Comment

**Specialist:** id, name, img.

Tên cột	Kiểu dữ liệu	Khóa	Mô tả
id	BIGINT(20)	PK	Mã khoa
name	VARCHAR(45)		Tên khoa
img	VARCHAR(200)		Địa chỉ hình ảnh khoa

**Bảng 27.** Bảng Specialist

**Day\_off:** id, date, time\_start, time\_end, reason, doctor\_id.

Tên cột	Kiểu dữ liệu	Khóa	Mô tả
id	BIGINT(20)	PK	Mã ngày nghỉ
date	VARCHAR(45)		Ngày nghỉ
time_start	TIME		Thời gian bắt đầu nghỉ
time_end	TIME		Thời gian kết thúc nghỉ
reason	VARCHAR(200)		Lý do nghỉ
doctor_id	BIGINT(20)		Mã bác sĩ

**Bảng 28.** Bảng Day\_off

**Admin:** id, name, email, password.

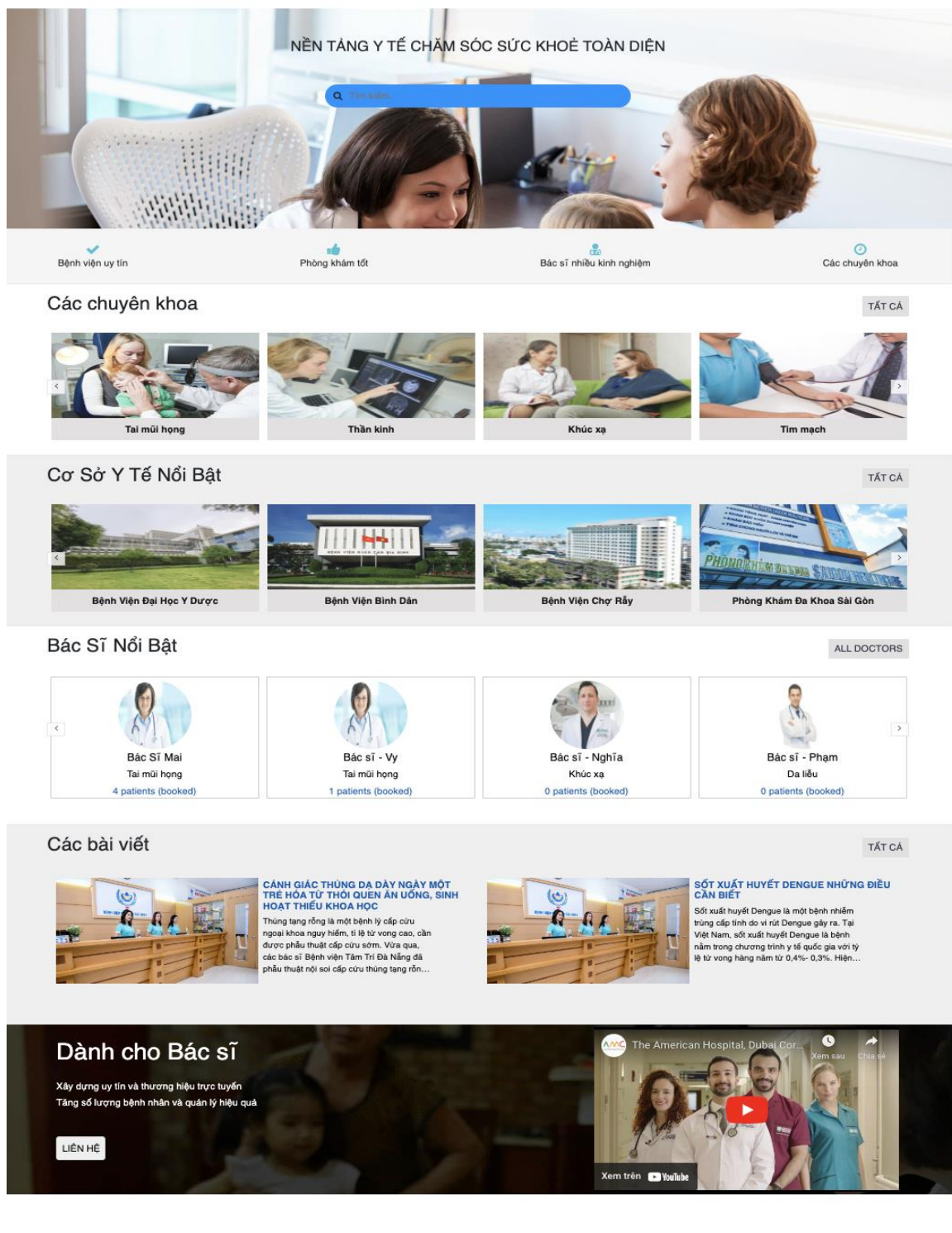
Tên cột	Kiểu dữ liệu	Khóa	Mô tả
id	BIGINT(20)	PK	Mã admin
name	VARCHAR(45)		Tên admin
email	VARCHAR(45)		Email admin
password	VARCHAR(100)		Mật khẩu admin

**Bảng 29.** *Bảng Admin*

## CHƯƠNG 5: XÂY DỰNG VÀ THIẾT KẾ ỨNG DỤNG ĐẶT LỊCH KHÁM CHỮA BỆNH DOCTOR CARE

### 5.1. Giao diện trang web và chức năng dành cho bệnh nhân

- Trang chủ: Hiển thị thông tin website, giới thiệu về trang web, danh sách lựa chọn chức năng



**Hình 31.** Giao diện trang chủ

- Giao diện chức năng tìm kiếm chuyên khoa.


## Các chuyên khoa



**Hình 33.** *Giao diện tìm kiếm chuyên khoa.*

- Giao diện chức năng đặt lịch

Tạo cuộc hẹn



Thông tin bác sĩ

Bác Sĩ Mai

11:00 - 12:00

09/01/2023

201B Đ. Nguyễn Chí Thanh, Phường 12, Quận 5, Thành phố Hồ Chí Minh

Chi phí: 200.000 VND

Thông tin bệnh nhân

Full name

☒ Male ☐ Female

Phone number (\*)

Email (\*)

Year

Address

Biểu hiện:

Khai báo Y tế (không bắt buộc) [\(View\)](#):

Cancel

Xác nhận cuộc hẹn

**Hình 34.** *Giao diện chức năng đặt lịch*

- *Ý nghĩa:* Cửa sổ cho phép bệnh nhân điền đầy đủ các thông tin cần thiết như Họ tên, Giới tính, SĐT, Địa chỉ cũng như biểu hiện của bệnh nhân trước khi xác nhận lịch hẹn,.

- Giao diện Thông tin bác sĩ



### Bác sĩ - Vy

Chuyên khoa: Tai mũi họng

Gửi cảm nhận sau khi đi khám

Bác sĩ với nhiều năm kinh nghiệm khám chữa bệnh. Có nhiều công trình nghiên cứu, bài báo khoa học về bệnh đái tháo đường, tuyến giáp, tuyến yên, rối loạn lipid máu... Bác sĩ cũng đang là 1 trong những báo cáo viên được giới chuyên môn quan tâm đánh giá cao tại các hội thảo khoa học, tọa đàm và đào tạo chia sẻ kinh nghiệm tại các bệnh viện tuyến quận, huyện, tỉnh.

QUÁ TRÌNH ĐÀO TẠO ✓2002 – 2008 : Bác sĩ đa khoa – Đại Học Y Dược TP.HCM

✓2008 – 2011: Bác sĩ nội trú chuyên ngành Nội tiết – Đại Học Y Dược TP.HCM

✓Tháng 04/2016: Tham gia khóa huấn luyện tại trung tâm đái tháo đường Steno – Đan Mạch

✓2021 – nay: Nghiên cứu sinh chuyên ngành Nội tim mạch trường đại Học Y Dược TP. HCM. Tham gia nhiều hội nghị, khóa huấn luyện về đái tháo đường và các bệnh lý nội tiết trong và ngoài nước.

✓Bác sĩ Lê Hoàng Bảo trực thuộc khoa Nội tiết – Bệnh viện Đại Học Y Dược TPHCM

✓Tham gia công tác đào tạo:

– Tham gia các chương trình đào tạo, nâng cao kiến về thực hành chăm sóc bệnh nhân đái tháo đường tại các bệnh viện tuyến quận, huyện và các tỉnh.

– Đào tạo trực tuyến (Insuwebinar) cho các BS tại các tỉnh Đồng Nai, Bến Tre, Vĩnh Long...

#### Chọn lịch khám:

Monday - 16/01/2023

08:00 - 09:00

09:00 - 10:00

10:00 - 11:00

11:00 - 12:00

13:00 - 14:00

14:00 - 15:00

15:00 - 16:00

16:00 - 17:00

Chọn

ĐỊA CHỈ KHÁM BỆNH:

HCM

...

Địa chỉ khám bệnh:



### **Hình 35.** *Giao diện thông tin bác sĩ*

- Ý nghĩa: Cửa sổ cho phép bệnh nhân xem thông tin về bác sĩ, về chuyên môn của Bác sĩ. Đồng thời có thể xem cảm nhận của bệnh nhân khác về bác sĩ. Bên phải hiển thị thông tin về lịch khám của bác sĩ mà bệnh nhân có thể lựa chọn để đặt lịch hẹn.



- Giao diện đăng ký thông tin bệnh nhân, bác sĩ.

## Làm việc với DoctorsCare

DoctorCare hân hạnh được hợp tác với các Bác sĩ và cơ sở y tế. Rất hân hạnh được hợp tác với Bác sĩ và các cơ sở y tế. Vui lòng gửi thông tin, chúng tôi sẽ liên hệ với bạn trong thời gian sớm nhất.

Tên liên hệ	Email
<input type="text" value="Contact"/>	<input type="text" value="Email"/>
SDT	Bệnh viện hiện đang công tác
<input type="text" value="SDT"/>	<input type="text" value="Hospitals, clinics, organizations, companies"/>
Địa chỉ	
<input type="text" value="Địa chỉ"/>	
Nội dung tin nhắn	
<input type="text"/>	
<input type="button" value="Gửi thông tin"/>	

**Hình 39.** *Giao diện đăng ký thông tin bệnh nhân, bác sĩ*

- *Ý nghĩa:* Cửa sổ cho phép các cá nhân, tổ chức đang hoạt động trong lĩnh vực y khoa có thể kết nối với Doctor Care thông qua việc cung cấp thông tin liên hệ cũng như nội dung muốn hợp tác..

## 5.2. Giao diện trang web và chức năng dành cho bác sĩ

- Giao diện thông tin cá nhân bác sĩ.

Doctor information

Name

Bác sĩ - Vy

SDT

088456735

Clinic

Bệnh Viện Bình Dân


Chuyên khoa

Tai mũi họng

Địa chỉ

HCM

Avatar:



OK

**Hình 40.** *Giao diện thông tin cá nhân bác sĩ.*



- Giao diện danh sách bệnh nhân đặt lịch.

Quản lý Bệnh nhân				
<a href="#">Thêm bệnh nhân(0)</a> <a href="#">Cần xác nhận(0)</a> <a href="#">Đã xác nhận (5)</a> <a href="#">Đã huỷ (1)</a>				
ID - Name	SDT	Email	Update	Actions
9 - Nguyễn Văn A	0941663665	gavangvip@gmail.com	14/01/2023, 15:14 PM	<a href="#">Chi tiết</a>
8 - Đặng Thị B	0941663670	gavangvip@gmail.com	01/01/2023, 12:39 PM	<a href="#">Chi tiết</a>
7 - Hoàng Tuấn Tú	0941662773	conmeomeo11@gmail.com	07/01/2023, 18:37 PM	<a href="#">Chi tiết</a>
5 - Võ Khang	0321848484	kane@gmail.com	15/11/2022, 08:27 AM	<a href="#">Chi tiết</a>
4 - Lê Đan	0321456789	alex@gmail.com	14/11/2022, 20:22 PM	<a href="#">Chi tiết</a>

**Hình 41.** Giao diện danh sách bệnh nhân đặt lịch

- *Ý nghĩa:* Cho phép Bác sĩ có thể xem thông tin các bệnh nhân đến khám. Có thể lựa chọn theo ngày làm việc và xem chi tiết về bệnh nhân. Đồng thời sau khi bệnh nhân đã khám, Bác sĩ có thể gửi đơn thuốc qua email

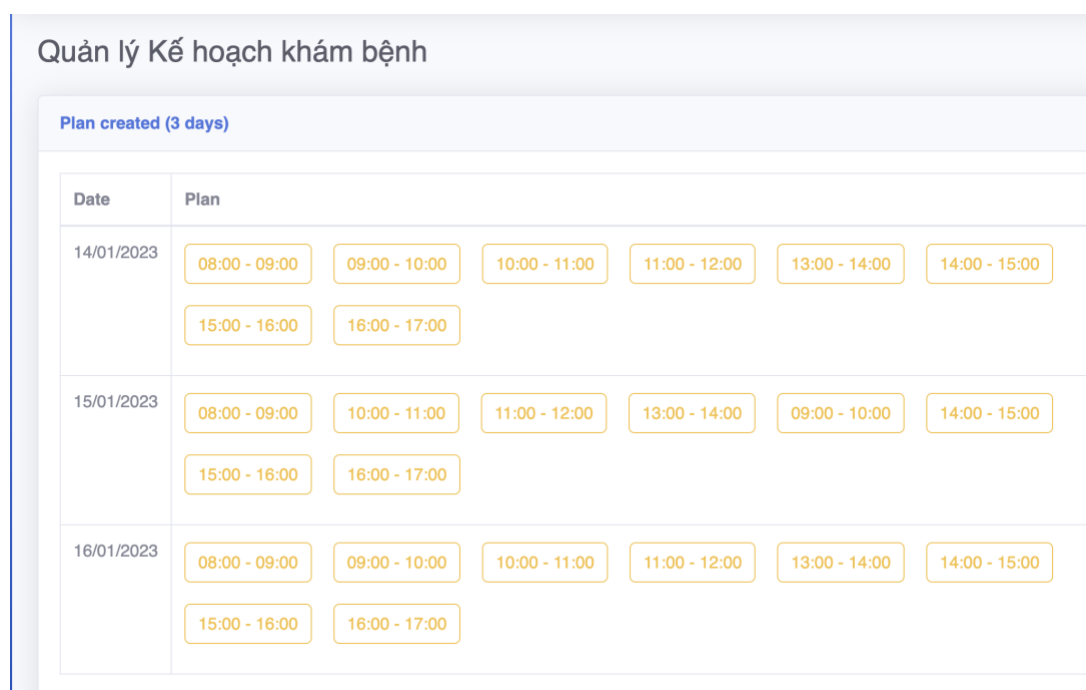
- Giao diện lịch khám của bác sĩ.

Danh sách bệnh nhân				
<div>Danh sách bệnh nhân</div> <div>Time:</div> <div> <input type="text" value="14/01/2023"/> <input type="button" value="Search"/> </div>				
Time	No	Name	Gender	Action
13h - 14h	1	Hùng	Female	<a href="#">Detail</a> <a href="#">Gửi đơn thuốc</a>  

**Hình 42.** Giao diện lịch khám của bác sĩ

- *Ý nghĩa:* Cho phép Bác sĩ có thể xem thông tin các bệnh nhân đến khám. Có thể lựa chọn theo ngày làm việc và xem chi tiết về bệnh nhân. Đồng thời sau khi bệnh nhân đã khám, Bác sĩ có thể gửi đơn thuốc qua email.

- Giao diện cài đặt lịch của bác sĩ

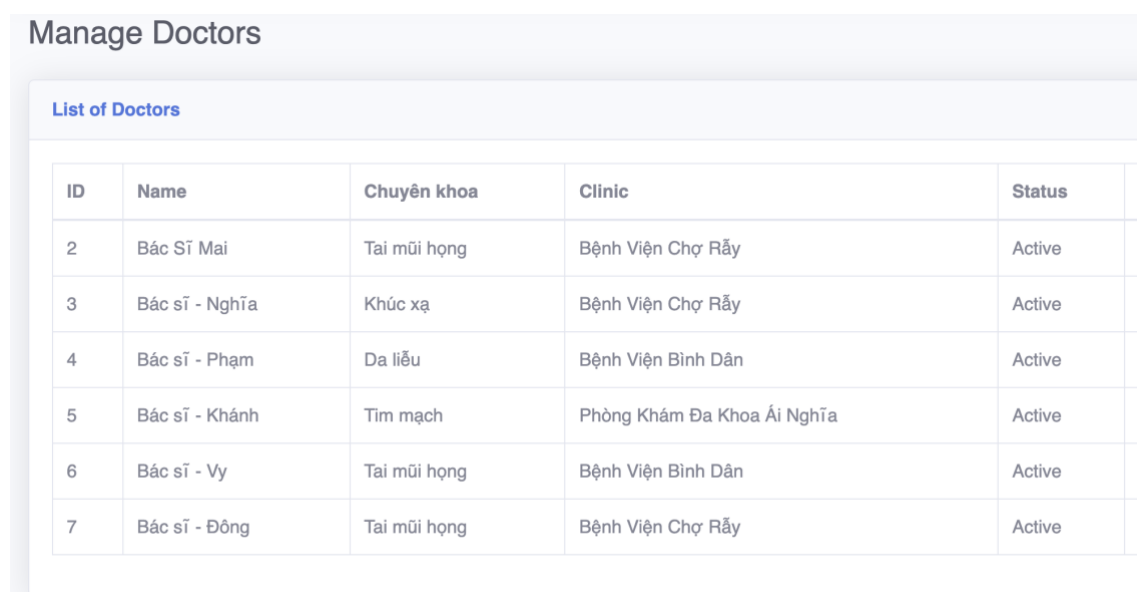


**Hình 43.** Giao diện cài đặt lịch của bác sĩ

- *Ý nghĩa:* Cho phép Bác sĩ có thể thêm các ngày và giờ làm việc cụ thể. Đây là những ngày Bác sĩ có thể tiếp nhận bệnh nhân.. Bệnh nhân chỉ có thể chọn thời gian đặt lịch hiển thị tương ứng với các ngày đã được Bác sĩ đăng ký làm việc trước đó.

### 5.3. Giao diện trang web và chức năng dành cho admin

- Giao diện quản lý bác sĩ



**Hình 44.** Giao diện quản lý bác sĩ

- Giao diện quản lý chi tiết bác sĩ.

Quản lý Bác sĩ						<a href="#">Add new</a>
Danh sách						
ID	Tên	Chuyên khoa	BV	Trạng thái	Hành động	
2	Bác Sĩ Mai	Tai mũi họng	Bệnh Viện Chợ Rẫy	Hoạt động	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>	
6	Bác sĩ - Vy	Tai mũi họng	Bệnh Viện Bình Dân	Hoạt động	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>	
3	Bác sĩ - Nghĩa	Khúc xạ	Bệnh Viện Chợ Rẫy	Hoạt động	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>	
4	Bác sĩ - Phạm	Da liễu	Bệnh Viện Bình Dân	Hoạt động	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>	
5	Bác sĩ - Khánh	Tim mạch	Phòng Khám Đa Khoa Ái Nghĩa	Hoạt động	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>	
7	Bác sĩ - Đông	Tai mũi họng	Bệnh Viện Chợ Rẫy	Hoạt động	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>	

**Hình 45.** Giao diện quản lý chi tiết bác sĩ

- *Ý nghĩa:* Cho phép admin thêm, xóa, sửa tên và thông tin của từng bác sĩ. Đồng thời ứng với từng bác sĩ, thay đổi được thông tin của bác sĩ như tên, chuyên khoa và bệnh viện đang công tác cũng như trạng thái hoạt động.

- Giao diện quản lý bệnh nhân.

Quản lý Bệnh nhân				
<a href="#">Thêm bệnh nhân(0)</a> <a href="#">Cần xác nhận(0)</a> <a href="#">Đã xác nhận (5)</a> <a href="#">Đã hủy (1)</a>				
ID - Name	SDT	Email	Update	Actions
9 - Nguyễn Văn A	0941663665	gavangvip@gmail.com	14/01/2023, 15:14 PM	<a href="#">Chi tiết</a>
8 - Đặng Thị B	0941663670	gavangvip@gmail.com	01/01/2023, 12:39 PM	<a href="#">Chi tiết</a>
7 - Hoàng Tuấn Tú	0941662773	conmeomeo11@gmail.com	07/01/2023, 18:37 PM	<a href="#">Chi tiết</a>
5 - Võ Khang	0321848484	kane@gmail.com	15/11/2022, 08:27 AM	<a href="#">Chi tiết</a>
4 - Lê Đan	0321456789	alex@gmail.com	14/11/2022, 20:22 PM	<a href="#">Chi tiết</a>

**Hình 46.** Giao diện quản lý bệnh nhân

- *Ý nghĩa:* Cho phép Admin và Nhân viên tiếp nhận bệnh nhân thông qua hành động xác nhận. Sau đó bệnh nhân sẽ chuyển sang trạng thái đã xác nhận. Trường hợp bệnh nhân không đủ điều kiện được tiếp nhận hoặc liên hệ huỷ cuộc hẹn, nhân viên sẽ thực hiện Huỷ bỏ.

- Giao diện quản lý khoa.

Quản lý chuyên khoa		
ID	Name	Actions
7	Da liễu	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>
3	Khúc xạ	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>
1	Tai mũi họng	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>
5	Tâm lý	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>
2	Thần kinh	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>
4	Tim mạch	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>
6	Xương khớp	<a href="#">i</a> <a href="#">✎</a> <a href="#">🗑</a>

**Hình 47.** Giao diện quản lý khoa

- Ý nghĩa: Cho phép admin thêm, xóa, sửa tên các Chuyên khoa y tế

- Giao diện thông tin tài khoản admin

Thông tin
Đổi mật khẩu

Email

Tên hiển thị

admin -Nam Hung

SĐT

Giới tính

☐ Nam
☐ Nữ

**Hình 48.** Giao diện thông tin tài khoản admin

- Ý nghĩa: Cho phép admin thay đổi thông tin Email, Tên hiển thị, SĐT, Giới tính và Địa chỉ. Thay đổi Mật khẩu đăng nhập.

## KẾT LUẬN

### 6.1. Kết quả đạt được

- Nắm vững được kiến thức về thiết kế và xây dựng website
- Xây dựng được hệ thống cung cấp đầy đủ các chức năng cơ bản của một website đặt lịch.
- Giao diện thân thiện, dễ nhìn giúp người dùng dễ dàng lựa chọn được những sản phẩm theo yêu cầu.
- Chức năng tạo lịch đặt khám chữa bệnh cho bác sĩ có thể xem được thông tin bệnh nhân đặt lịch giúp quản lý đầy đủ về người dùng.
- Chức năng lịch chờ giúp quan sát tốt hơn về thông tin đặt lịch của bệnh nhân.

### 6.2. Hạn chế của đề tài

Ngoài những vấn đề đạt được vẫn còn những vấn đề vẫn chưa được giải quyết:

- Chưa xuất được hồ sơ qua thông tin bệnh nhân để giảm quá trình thủ tục.
- Hệ thống trang quản trị còn nhiều thiếu sót.
- Chưa cung cấp đầy đủ chức năng cho bệnh nhân và bác sĩ.
- Giao diện còn đơn giản và thiếu thu hút.
- Cơ sở dữ liệu còn nhiều thiếu sót.
- Hệ thống bảo mật chưa cao.

### 6.3. Hướng phát triển đề tài

Xây dựng thêm các chức năng còn thiếu

- Xây dựng giao diện đẹp mắt, sinh động hơn, cơ sở dữ liệu nhất quán hơn.
- Xây dựng thêm các tính năng phân loại bệnh nhân để gợi ý các bác sĩ và chuyên khoa tốt hơn.
- Xây dựng các tính năng thống kê cho bác sĩ và người quản trị.

## **TÀI LIỆU THAM KHẢO**

- [1] Website <https://topdev.vn/blog/restful-api-la-gi/> - Lần truy cập cuối cùng: 31/12/2022
  
- [2] Website <https://ReactJS.org>- Lần truy cập cuối cùng: 31/12/2022
  
- [3] Website <https://NodeJS.org>- Lần truy cập cuối cùng: 31/12/2022
  
- [4] Website <http://www.w3schools.com>. - Lần truy cập cuối cùng: 12/12/2022
  
- [5] Website <http://bookingcare.com> - Lần truy cập cuối cùng: 20/12/2022
  
- [6] NodeJS with MySQL <https://youtu.be/dGxzQFJ0YFg> - Lần truy cập cuối cùng: 31/12/2022
  
- [7] Appointment Booking App – MERN- <https://youtu.be/7giZGFDGnkc> - Lần truy cập cuối cùng:  
31/12/2022