**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**ĐẠI HỌC DUY TÂN**



**Tên đề tài**

**NGHIÊN CỨU VÀ XÂY DỰNG WEBSITE Y TẾ VÀ TƯ VẤN SỨC KHỎE SỬ DỤNG AI CHATBOT**

**TÓM TẮT BÁO CÁO**

**THÀNH VIÊN NHÓM**

**Nguyễn Thành Trung -3427**

**Nguyễn Tấn Tân -1237**

**Từ Minh Hưng -0778**

**Trần Quang -4597**

**Trần Hải Đăng -1285**

**GVHD : ThS. Phạm Khánh Linh**

**Đà Nẵng, Tháng 05 / 2025**

# **MỞ ĐẦU**

**1. Lý do chọn đề tài**

Từ trước tới nay, sức khỏe con người luôn là vấn đề được quan tâm hàng đầu. Các bệnh viện và phòng khám mọc lên khắp mọi nơi. Đời sống của người dân được nâng cao, dẫn đến nhu cầu được tư vấn sức khỏe và khám bệnh cũng được chú trọng hơn. Vì quá nhiều bệnh viện và phòng khám mọc lên, nên việc sàng lọc và lựa chọn nơi khám bệnh phù hợp là một điều hết sức khó khăn.

Với thời đại công nghệ hiện nay, công nghệ thông tin đang được ứng dụng trong nhiều ngành nghề, lĩnh vực. Đối với y tế cũng vậy, việc ứng dụng công nghệ thông tin vào trong y tế sẽ giúp bệnh viện và bác sĩ tiếp cận được nhiều bệnh nhân hơn, bệnh nhân sẽ dễ dàng hơn trong việc chọn nơi để khám hoặc tư vấn sức khỏe.

Với những lí do trên, sinh viên chọn đề tài “**Nghiên cứu và xây dựng website y tế và tư vấn sức khỏe sử dụng ai chatbot**” làm đề tài khóa luận tốt nghiệp. Đề tài nhằm làm cho sự tiếp cận y tế của bệnh nhân trở nên dễ dàng hơn, nâng cao sức khỏe cộng đồng.

**2. Mục đích và ý nghĩa của đề tài**

1. **Mục đích**

* Website này giúp cho bệnh nhân sẽ tiếp cận được thông tin của bác sĩ và các bệnh viện, phòng khám, bệnh nhân có thể đặt lịch khám online, trò chuyện trực tiếp với đội ngũ hỗ trợ để được tư vấn sức khỏe.
* Website này cũng giúp các phòng khám, bệnh viện, bác sĩ tiếp cận được nhiều bệnh nhân hơn, dễ dàng quản lí thông tin của bác sĩ, bệnh viện và phòng khám.

1. **Ý nghĩa**

* Giúp cho người bệnh có thể dễ dàng nhận được sự tư vấn và chăm sóc sức khỏe.
* Giúp cho phòng khám và bệnh viện tiếp cận được nhiều bệnh nhân hơn.

**3. Đối tượng và phạm vi nghiên cứu**

1. **Đối tượng**

Người dùng có nhu cầu cần được tư vấn sức khỏe hoặc có nhu cầu được khám bệnh.

1. **Phạm vi**

Đáp ứng theo yêu cầu của khách hàng để phát triển web phải:

* Xác định yêu cầu của khách hàng.
* Phân tích , đặc tả yêu cầu chức năng của hệ thống.
* Thiết kế giao diện cho hệ thống.
* Phát triển hệ thống bằng ReactJS.
* Kiểm thử một số chức năng của người dùng.

**4. Phương pháp nghiên cứu**

Đề tài sử dụng các phương pháp nghiên cứu sau:

* Tổng hợp các kết quả nghiên cứu từ các tư liệu liên quan.
* Sử dụng các kiến thức đã có và các công cụ để thiết kế website.
* Quan sát hoạt động và quy trình các website tương tự.
* Góp ý của các chuyên gia.

**5. Bố cục đề tài**

Nội dung của đề tài gồm các chương sau:

Chương 1: Tổng quan về công cụ và môi trường phát triển. Chương này giới thiệu về các ngôn ngữ và công cụ được dùng để xây dựng website.

Chương 2: Phân tích và thiết kế hệ thống. Chương này đi sâu vào phân tích các chức năng của hệ thống

Chương 3: Demo chương trình. Chương này trình bày các kết quả đạt được.

# **CHƯƠNG I**

# **TỔNG QUAN VỀ CÔNG CỤ VÀ MÔI TRƯỜNG PHÁT TRIỂN**

## **GIỚI THIỆU NGÔN NGỮ LẬP TRÌNH JAVASCRIPT**

### **Lịch sử phát triển của ngôn ngữ lập trình Javascript**

Lịch sử của JavaScript từ lúc bắt đầu tới nay đã trải qua rất nhiều giai đoạn. Đầu tiên nó được đặt tên là Mocha, được giới thiệu với thế giới là LiveScript và được đặt tên mới sau đó, nó được thiết kế như một nỗ lực để làm cho các trang web trở nên dễ tương tác hơn. Tên này đã được thay đổi chỉ vài tháng sau khi ra mắt thành JavaScript và giờ đây nó đã trở nên cần thiết cho các trang web và được cho là ngôn ngữ lập trình hàng đầu trên thế giới.

Brandon Eich, người tạo ra JavaScript, đã thiết kế nó thành một ngôn ngữ mà các lập trình viên và nhà thiết kế có thể sử dụng để lắp ráp các thành phần và tự động hóa công việc của họ.

“**Always bet on JavaScript**”

- *Brandon Eich*

### **Javascript là gì?**

Javascript là ngôn ngữ lập trình được các nhà phát triển web sử dụng để tạo trang web tương tác. Từ làm mới bảng tin trên trang mạng xã hội đến hiển thị hình ảnh động và bản đồ tương tác, các chức năng của JavaScript có thể cải thiện trải nghiệm người dùng của trang web. Là ngôn ngữ kịch bản phía máy khách, JavaScript là một trong những công nghệ cốt lõi của World Wide Web.

### **Một số đặc điểm nổi bậc của ngôn ngữ lập trình Javascript**

**Ngôn ngữ lập trình kịch bản**

**Kiểu dữ liệu động**

**Nền tảng độc lập**

**Xử lý không đồng bộ**

**Viết tắt**

**Module**

**Hỗ trợ mạnh cho việc phát triển web**

Các  ứng dụng Javascript

**Javascript và websites**

Tất nhiên rồi, JavaScript sinh ra để bổ sung tính tương tác và hành vi cho các trang web tĩnh vào năm 1995.

**Javascript và ứng dụng web (web applications)**

Các trình duyệt và máy tính được cải thiện theo thời gian, do đó khả năng của chúng trong việc hỗ trợ các ứng dụng web mạnh mẽ.

**Javascript và Server applications**

Sự nổi lên của Node.js vài năm trước cho phép các nhà phát triển mở rộng JavaScript ra ngoài trình duyệt.

**Javascript và lập trình game**

Bằng cách kết hợp HTML5 với JavaScript, bạn có thể thêm nhiều tương tác hơn để làm cho nội dung năng động hơn – điều này chỉ hoàn hảo cho mục đích chơi game.

## **1.2 GIỚI THIỆU REACTJS**

### **1.2.1 ReactJS là gì?**

* ReactJS là một framework Javascript mã nguồn mở do Meta phát triển. Nó được sử dụng để xây dựng các giao diện người dùng và ứng dụng web tương tác một cách nhanh chóng và hiệu quả với ít mã hơn đáng kể so với JavaScript gốc.

**1.2.2 Ưu, nhược điểm của ReactJS**

**Ưu điểm:**

* ReactJS cực kì hiệu quả: ReactJS tạo ra các DOM ảo – nơi mà các components thật sự tồn tại ở đó. ReactJS sẽ tính toán và chỉ cập nhật các component có sự thay đổi vào DOM thật, điều này giúp cải thiện đáng kể hiệu suất của ứng dụng.
* ReactJS giúp viết code Javascript đơn giản hơn: ReactJS cung cấp cú pháp rất đặc biệt là JSX, cho phép ta trộn lẫn code Javascript và HTML.
* ReatJS sở hữu cộng đồng sử dụng và hỗ trợ cực kì lớn.
* ReactJS có rất nhiều tiện ích nguồn mở.
* ReactJS cung cấp khả năng tái sử dụng các components, giúp cho việc viết code tốn ít thời gian hơn.
* Với ReactJS, lập trình viên sẽ viết ít code hơn, với nhiều chức năng hơn.
* Tính bảo mật cao.

**Nhược điểm:** Mặc dù ReactJS có thể kể đến rất nhiều các ưu điểm, nhưng đến thời điểm này, nó vẫn có một số điểm yếu sau:

* **Dung lượng** : ReactJS khá nặng nếu so với các framework khác.
* Khó tiếp cận với những người mới sử dụng.

**1.2.3 Các tính năng cốt lõi của ReactJS**

Dưới đây là các tính năng cốt lõi của ReactJS:

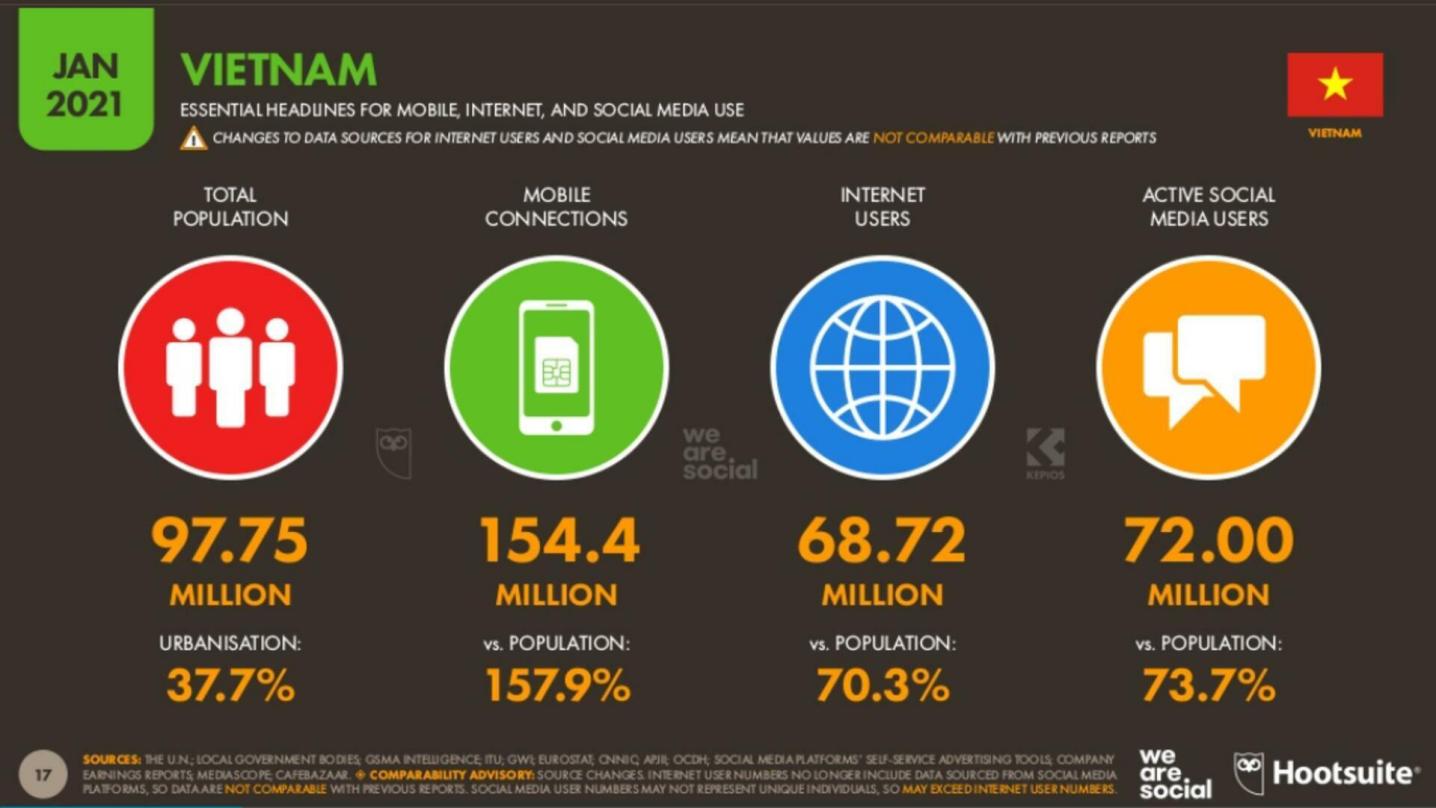
* **JSX - Javascript Syntax Extension**
* **Unidirectional Data Flow**
* **Virtual DOM**
* **Extentions**
* **Debugging**
* **Components**
* **ReactJS State**
* **ReactJS Props**

# **CHƯƠNG II**

# **PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

## **KHẢO SÁT THỰC TRẠNG**

Hiện nay, xã hội ngày càng phát triển, con người càng quan tâm đến sức khỏe nhiều hơn. Nhu cầu được khám bệnh và chăm sóc sức khỏe ngày càng được nâng cao. Tuy vậy, không phải ai cũng có thể dễ dàng tiếp cận được với các dịch vụ chăm sóc sức khỏe.

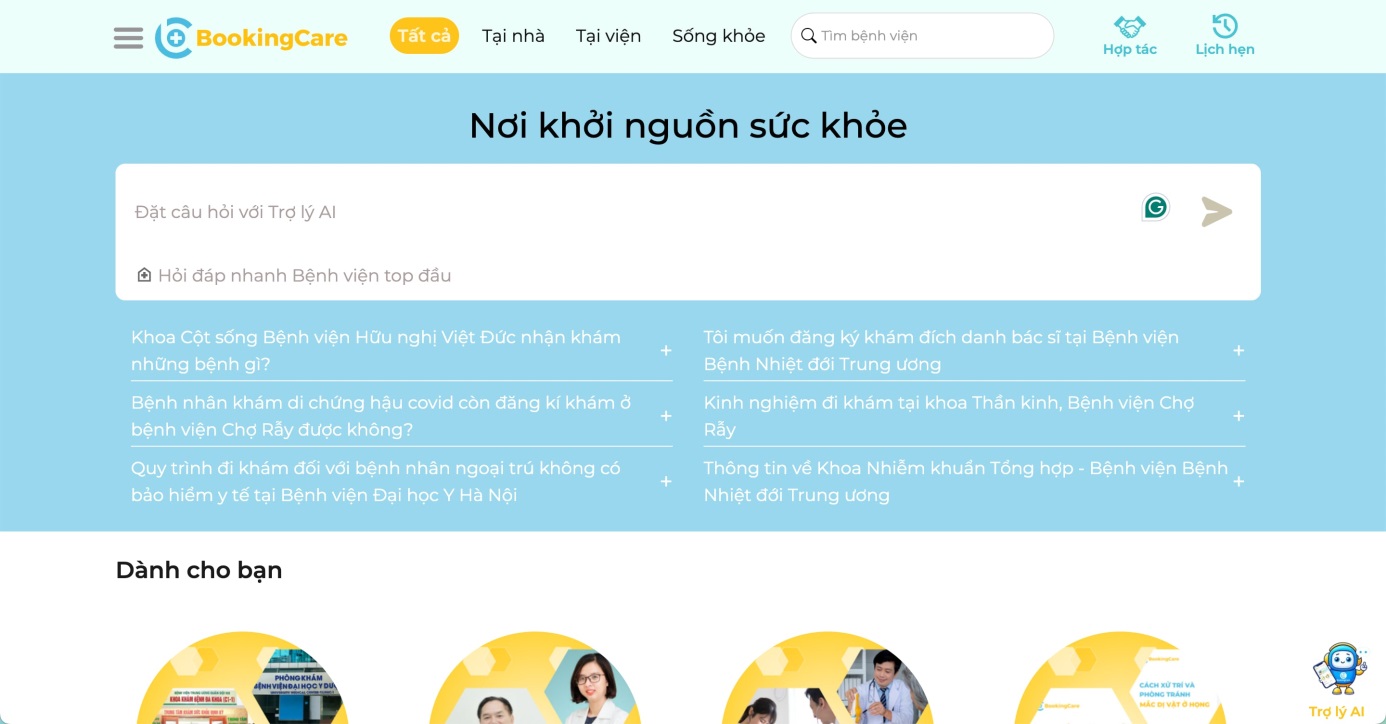


*Hình 1. 1 Các chỉ số thống kê kỹ thuật số ở Việt Nam năm 2021*

## **MỘT SỐ WEB SITE ĐẶT LỊCH KHÁM BỆNH, TƯ VẤN SỨC KHỎE**

* **Bookingcare**

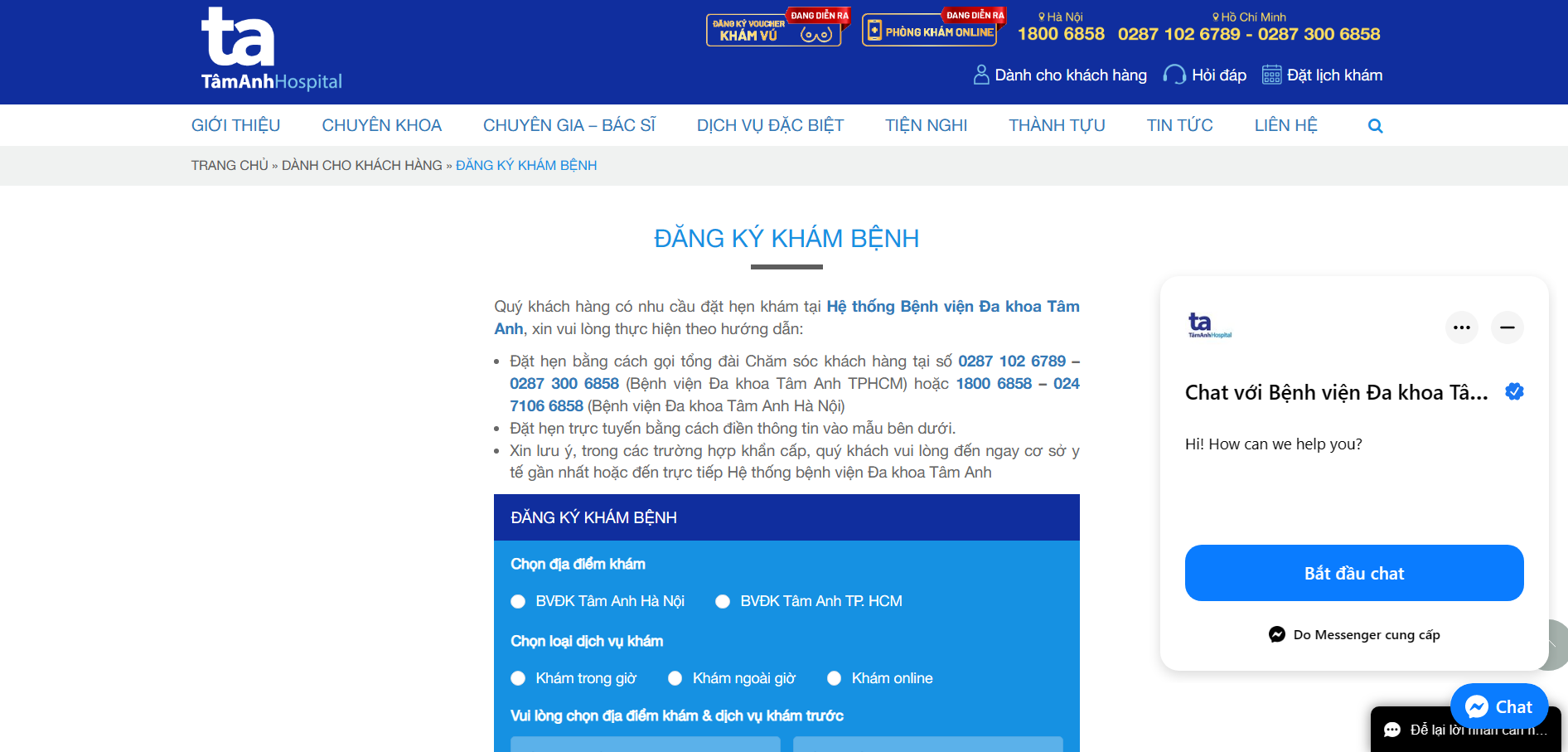
Link: [BookingCare - Nền tảng y tế sức khỏe toàn diện](https://bookingcare.vn/)



*Hình 1. 2 Giao diện trang web Bookingcare*

* **Trang Tamanhhospital**

Link: [Đăng ký khám bệnh | Bệnh viện Đa khoa Tâm Anh, Hà Nội (tamanhhospital.vn)](https://tamanhhospital.vn/danh-cho-khach-hang/dat-lich-kham/)



*Hình 1. 2 Giao diện trang web tamanhhospital*

Sau khi tìm hiểu, phân tích các Website này chúng tôi nhận thấy được những ưu nhược điểm và sẽ xây dựng công cụ hỗ trợ học có các đối tượng và chức năng cơ bản như mô tả trong phần sau.

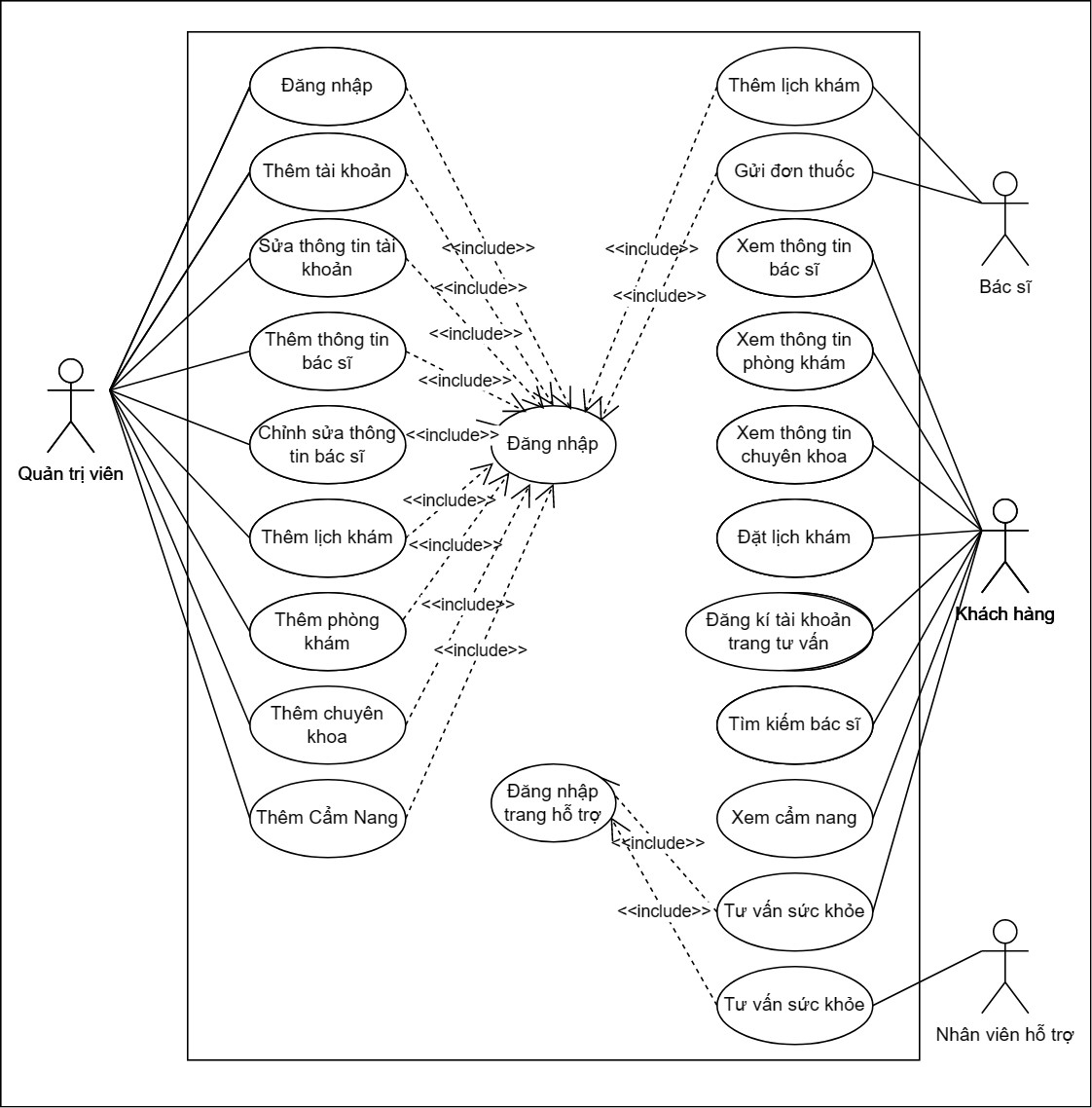
## **CÁC ĐỐI TƯỢNG TƯƠNG TÁC VỚI HỆ THỐNG**

Hệ thống có các tác nhân: quản trị viên, khách hàng, bác sĩ, đội ngũ tư vấn. Vai trò của các tác nhân được thể hiện trong bảng 2.1 như sau:

*Bảng 2. 1 Các tác nhân của hệ thống*

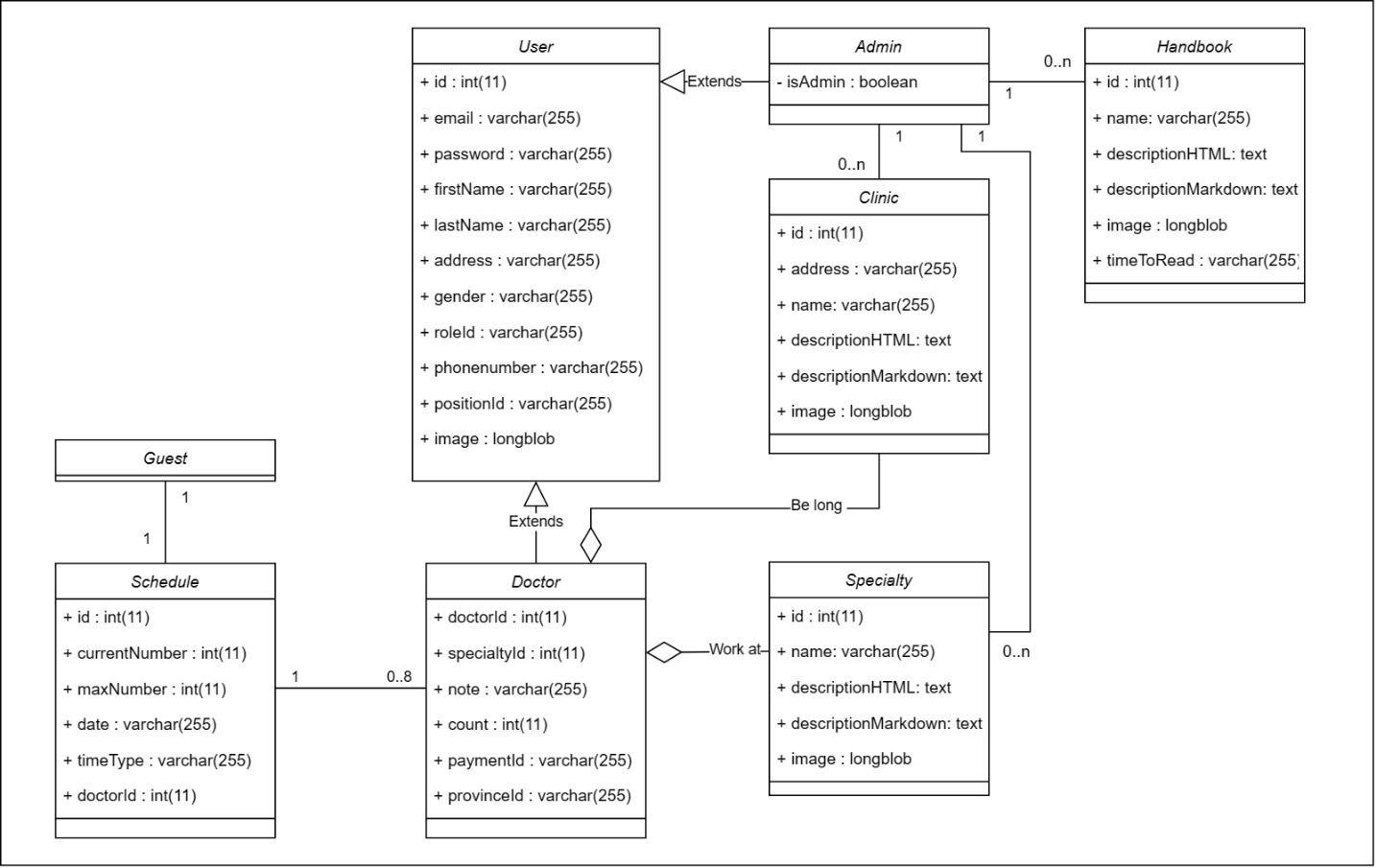
|  |  |  |
| --- | --- | --- |
| STT | Tên tác nhân | Mô tả |
| 1 | Quản trị viên | Quản trị viên có thể thêm người dùng, thêm, chỉnh sửa thông tin bác sĩ, thêm lịch khám cho bác sĩ, thêm phòng khám, thêm chuyên khoa, thêm cẩm nang. |
| 2 | Bác sĩ | Bác sĩ có thể đăng nhập, thêm lịch khám, gửi đơn thuốc thông qua email. |
| 3 | Khách hàng | Khách hàng có thể xem thông tin bác sĩ, phòng khám, chuyên khoa, nhận tư vấn sức khỏe và đăng nhập vào trang tư vấn sức khỏe, tìm kiếm bác sĩ, xem cẩm nang. |
| 4 | Hỗ trợ viên | Tư vấn sức khỏe và đăng nhập vào trang tư vấn. |

## **SƠ ĐỒ USECASE SỬ DỤNG (USECASE DIAGRAM)**

1. **Use case tổng quát**

*Hình 1. 3 Usecase tổng quát*

## **2.5. SƠ ĐỒ LỚP (CLASS DIAGRAM)**



*Hình 2. 52 Sơ đồ lớp*

## **2.6. THIẾT KẾ CƠ SỞ DỮ LIỆU**

Để xây dựng website tôi đã thiết kế những table sau đây đễ lưu trữ dữ liệu.

### **2.6.1. Các Tables**

#### **2.6.1.1. Table Users**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **Int(11)** | **No** | **auto\_increment** |  |
| email | varchar(255) | Yes |  |  |
| password | varchar(255) | Yes |  |  |
| firstName | varchar(255) | Yes |  |  |
| lastName | varchar(255) | Yes |  |  |
| address | varchar(255) | Yes |  |  |
| gender | varchar(255) | Yes |  | allcodes.keyMap |
| roleId | varchar(255) | Yes |  |  |
| phonenumber | varchar(255) | Yes |  |  |
| positionId | varchar(255) | Yes |  | allcodes.keyMap |
| image | longblob | Yes |  |  |

*Bảng 2. 15 Bảng users*

#### **2.6.1.2 Table Allcodes**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| key | varchar(255) | Yes |  |  |
| type | varchar(255) | Yes |  |  |
| valueEn | varchar(255) | Yes |  |  |
| valueVi | varchar(255) | Yes |  |  |

*Bảng 2. 16 Bảng allcodes*

#### **2.6.1.3. Table Markdowns**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| contentHTML | longtext | No |  |  |
| contentMarkdown | longtext | No |  |  |
| description | longtext | Yes |  |  |
| doctorId | int(11) | Yes |  |  |
| specialtyId | int(11) | Yes |  |  |
| clinicId | int(11) | Yes |  |  |

*Bảng 2. 17 Bảng markdowns*

#### **2.6.1.4. Table Doctor\_Infor**

*Bảng 2. 18 Bảng doctor\_infor*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| doctorId | int(11) | No |  |  |
| specialtyId | int(11) | Yes |  |  |
| clinicId | int(11) | Yes |  |  |
| priceId | varchar(255) | No |  | allcodes.keyMap |
| provinceId | varchar(255) | No |  | allcodes.keyMap |
| paymentId | varchar(255) | No |  | allcodes.keyMap |
| addressClinic | varchar(255) | No |  |  |
| nameClinic | varchar(255) | No |  |  |
| note | varchar(255) | Yes |  |  |
| count | int(11) | No |  |  |

#### **2.6.1.5. Table Schedules**

*Bảng 2. 19 Bảng schedules*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| currentNumber | int(11) | No |  |  |
| maxNumber | int(11) | Yes |  |  |
| date | varchar(255) | Yes |  |  |
| timeType | varchar(255) | Yes |  | allcodes.keyMap |
| doctorId | int(11) | Yes |  | users.id |

#### **2.6.1.6. Table Bookings**

*Bảng 2. 20 Bảng bookings*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| statusId | varchar(255) | Yes |  |  |
| doctorId | int(11) | Yes |  |  |
| patientId | int(11) | Yes |  | users.id |
| date | varchar(255) | Yes |  |  |
| timeType | varchar(255) | Yes |  | allcodes.keyMap |
| token | varchar(255) | Yes |  |  |

#### **2.6.1.7. Table Histories**

Bảng 2. 21 Bảng histories

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| patientId | int(11) | Yes |  |  |
| doctorId | int(11) | Yes |  |  |
| description | text | Yes |  |  |
| files | text | Yes |  |  |

#### **2.6.1.8. Table Clinics**

*Bảng 2. 22 Bảng clinics*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| address | varchar(255) | Yes |  |  |
| name | varchar(255) | Yes |  |  |
| descriptionHTML | text | Yes |  |  |
| descriptionMarkdown | text | Yes |  |  |
| image | longblob | Yes |  |  |

#### **2.6.1.9. Table Doctor\_clinic\_specialty**

*Bảng 2. 23 Bảng doctor\_clinic\_specialty*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| doctorId | int(11) | Yes |  |  |
| clinicId | int(11) | Yes |  |  |
| specialtyId | int(11) | Yes |  |  |

#### **2.6.1.10. Table Specialties**

*Bảng 2. 24 Bảng specialties*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| image | longblob | Yes |  |  |
| name | varchar(255) | Yes |  |  |
| descriptionHTML | text | Yes |  |  |
| descriptionMarkdown | text | Yes |  |  |

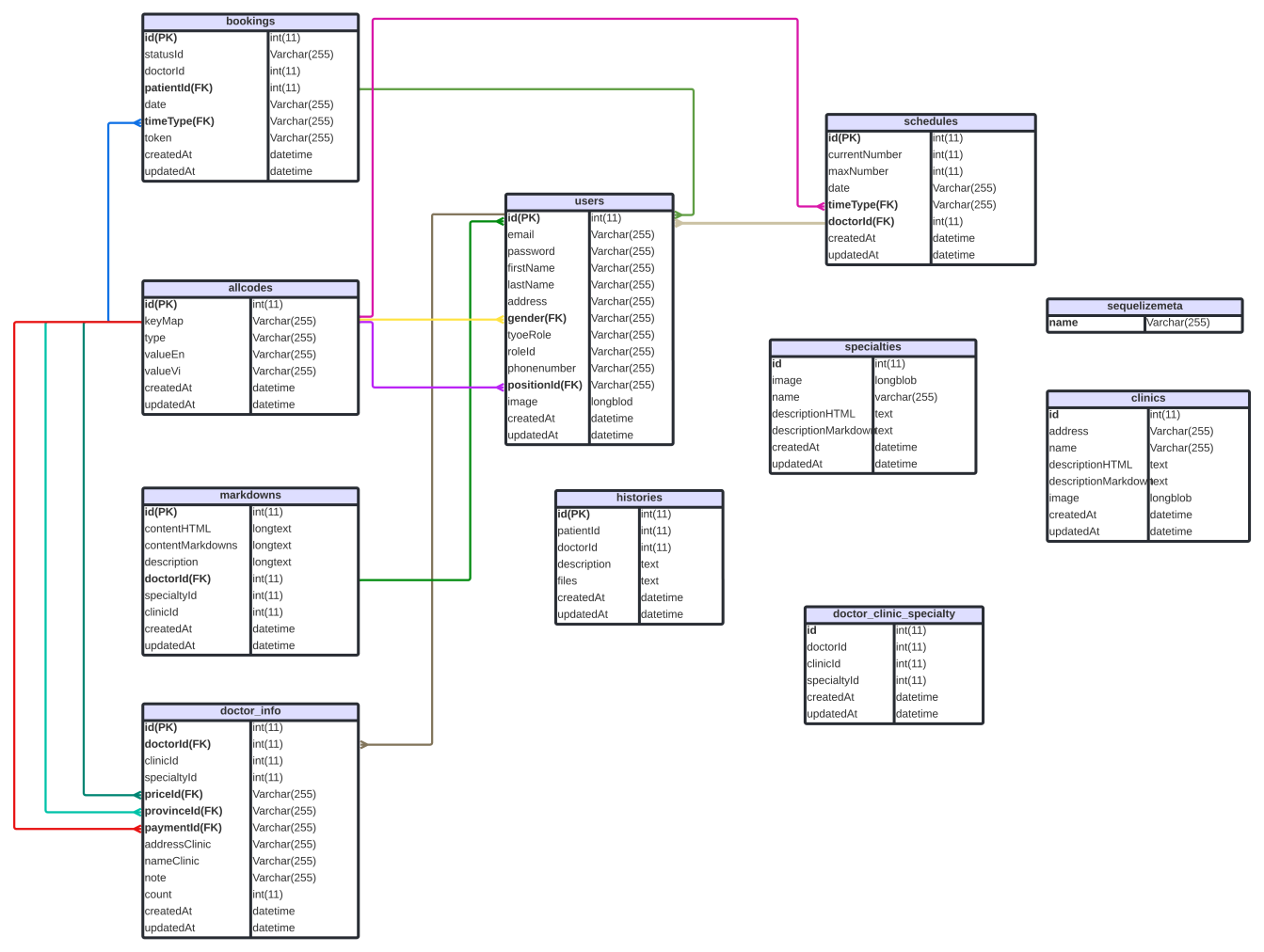
#### **2.6.1.11. Table Handbooks**

*Bảng 2. 25 Bảng handbooks*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Extra** | **Link to** |
| **id** | **int(11)** | **No** | **auto\_increment** |  |
| image | longblob | Yes |  |  |
| name | varchar(255) | Yes |  |  |
| descriptionHTML | text | Yes |  |  |
| descriptionMarkdown | text | Yes |  |  |
| timeToRead | varchar(255) | Yes |  |  |

### **2.6.2. Mô hình quan hệ**

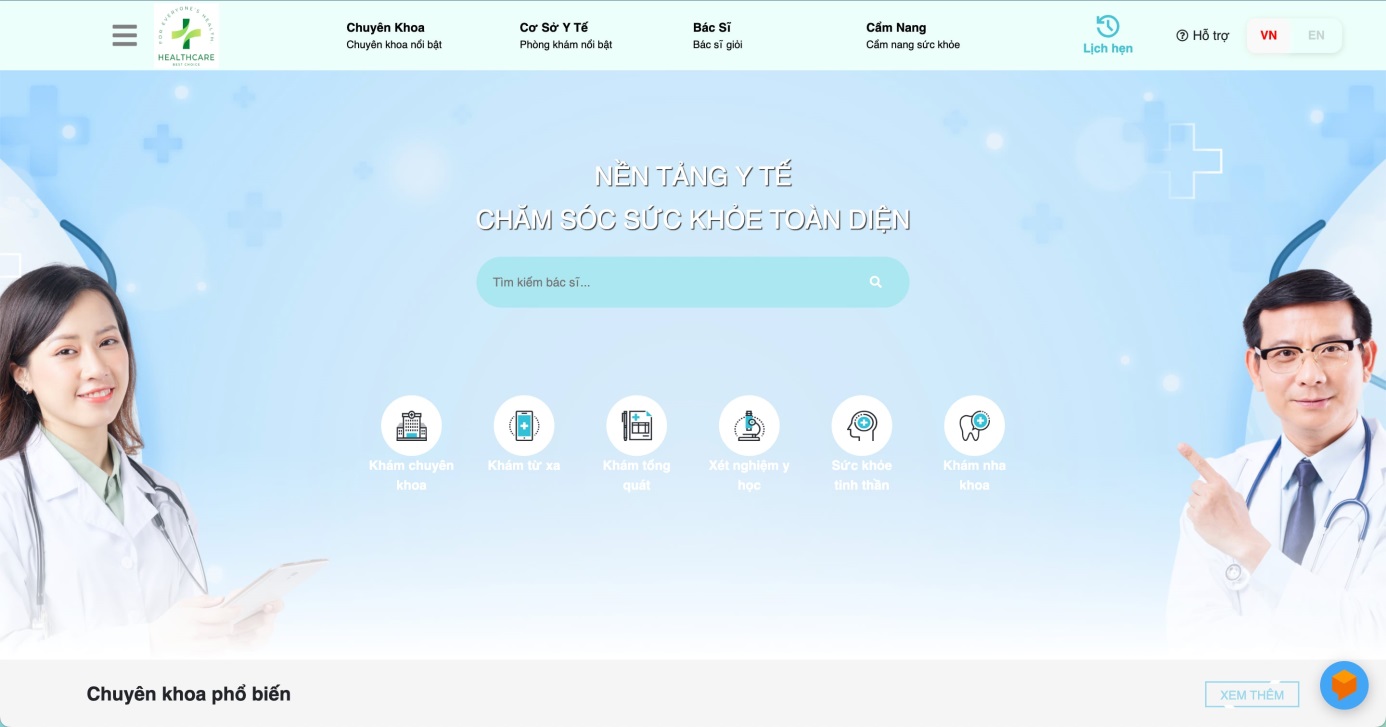
*Hình 2. 53 Mô hình quan hệ*



# **CHƯƠNG III**

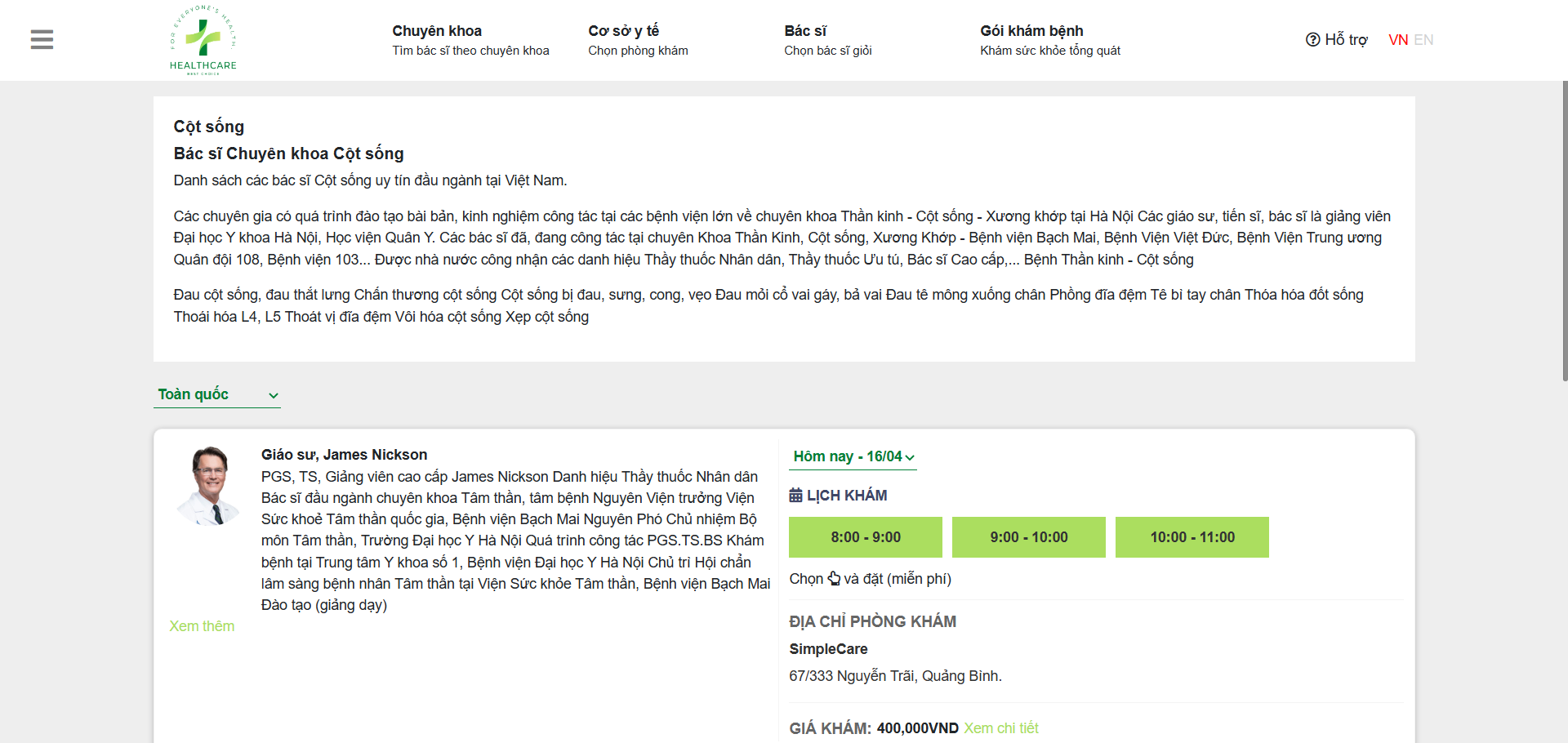
# **DEMO CHƯƠNG TRÌNH**

## **TRANG CHỦ**



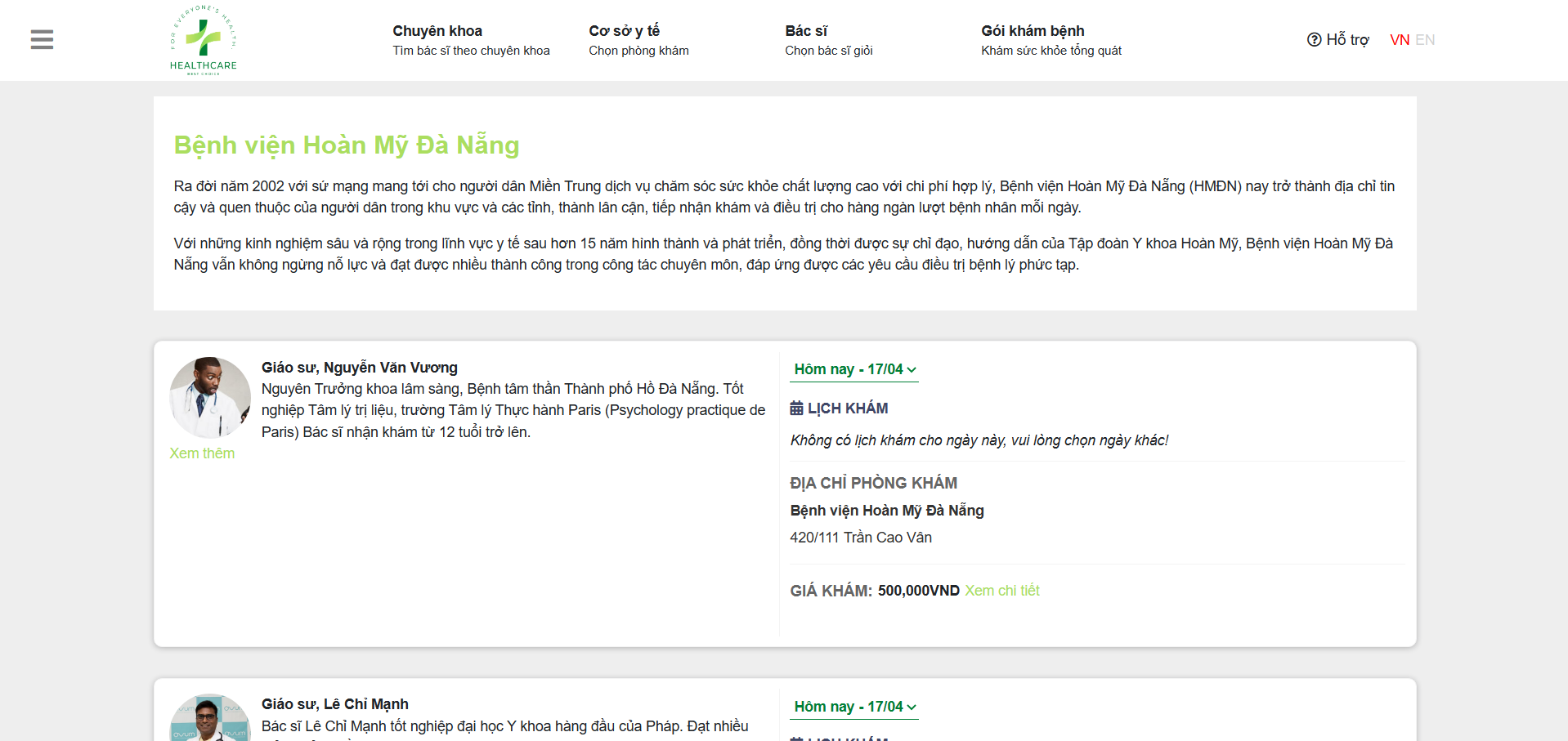
*Hình 3. 1 Giao* diện Trang chủ

## **3.2. THÔNG TIN CHI TIẾT CHUYÊN KHOA**



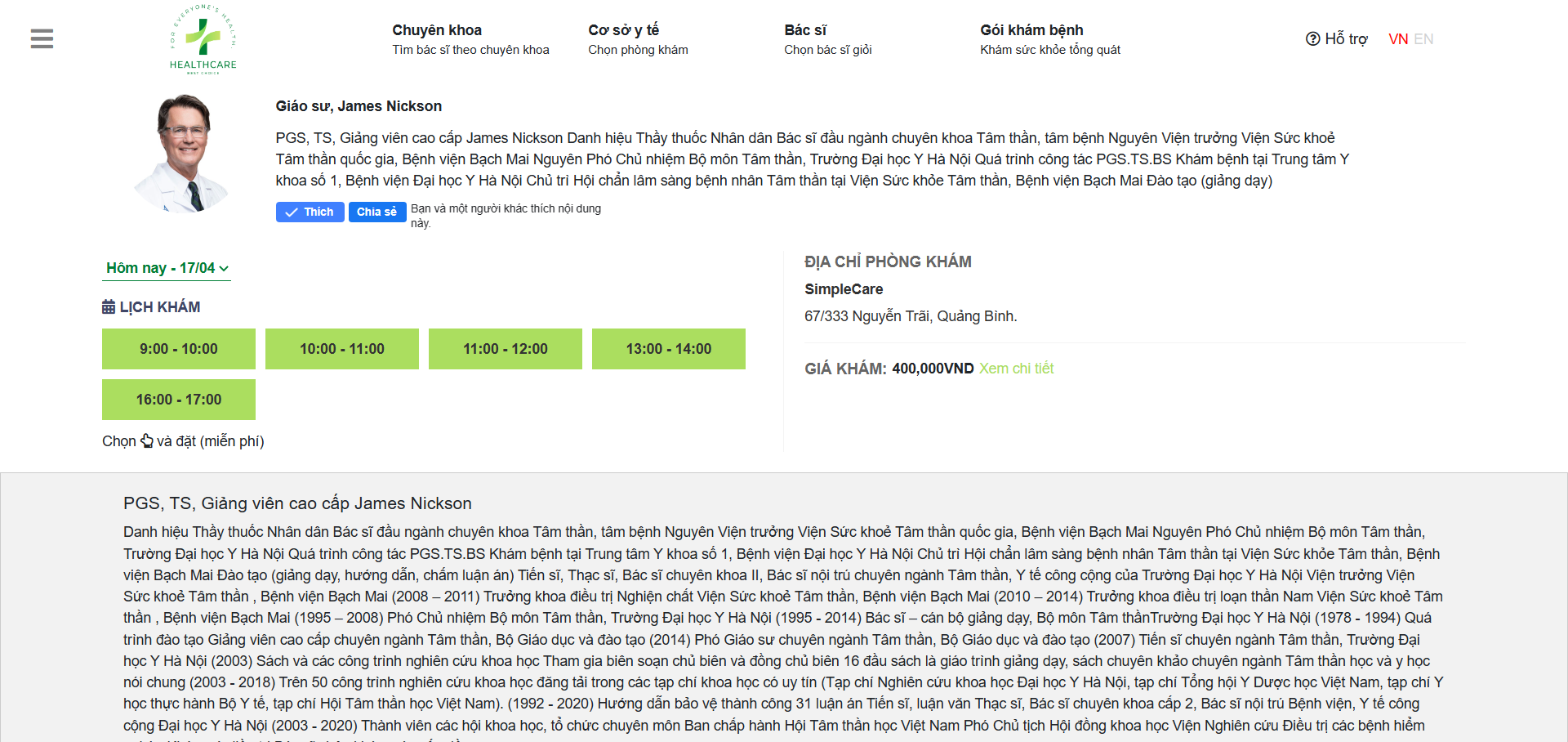
*Hình 3. 2 Giao diện trang chi tiết chuyên khoa*

## **3.3. THÔNG TIN CHI TIẾT PHÒNG KHÁM**



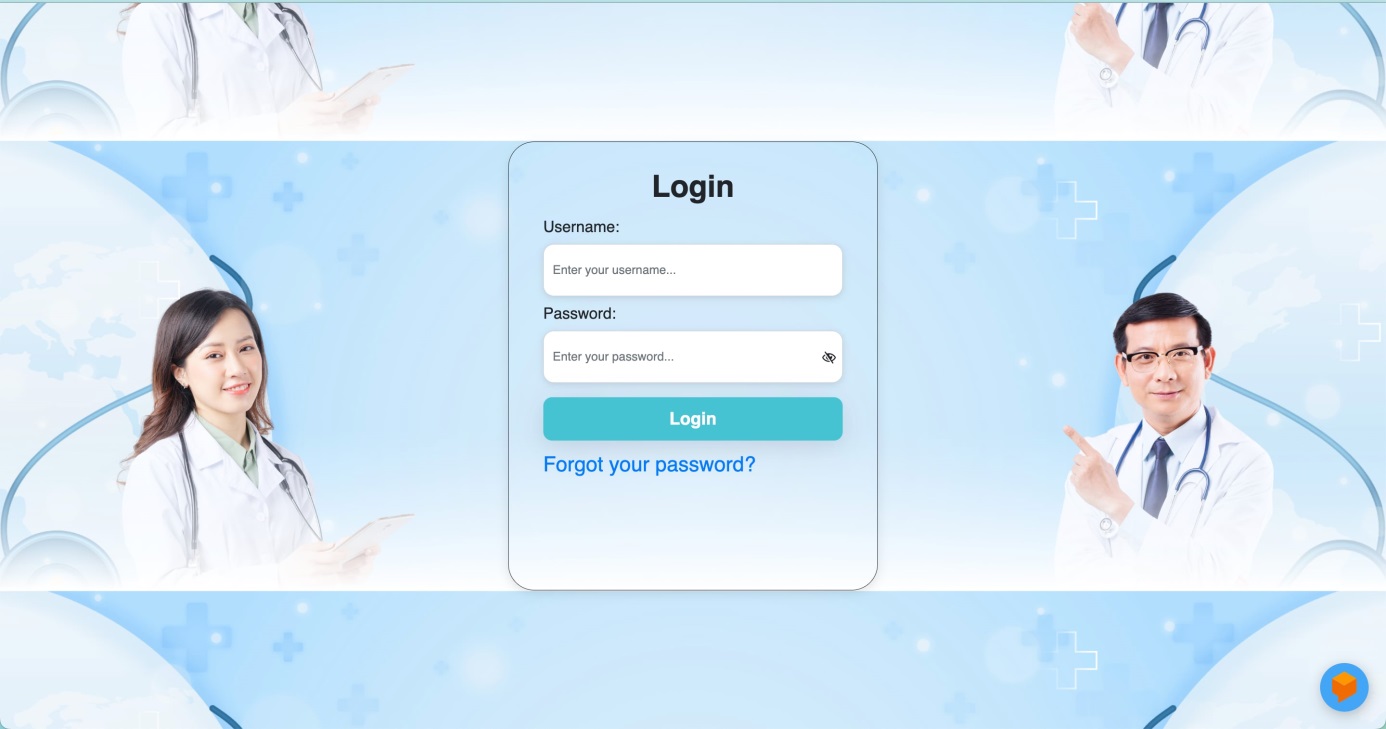
*Hình 3. 3 Giao diện trang Phòng khám*

## **3.4. THÔNG TIN CHI TIẾT BÁC SĨ**



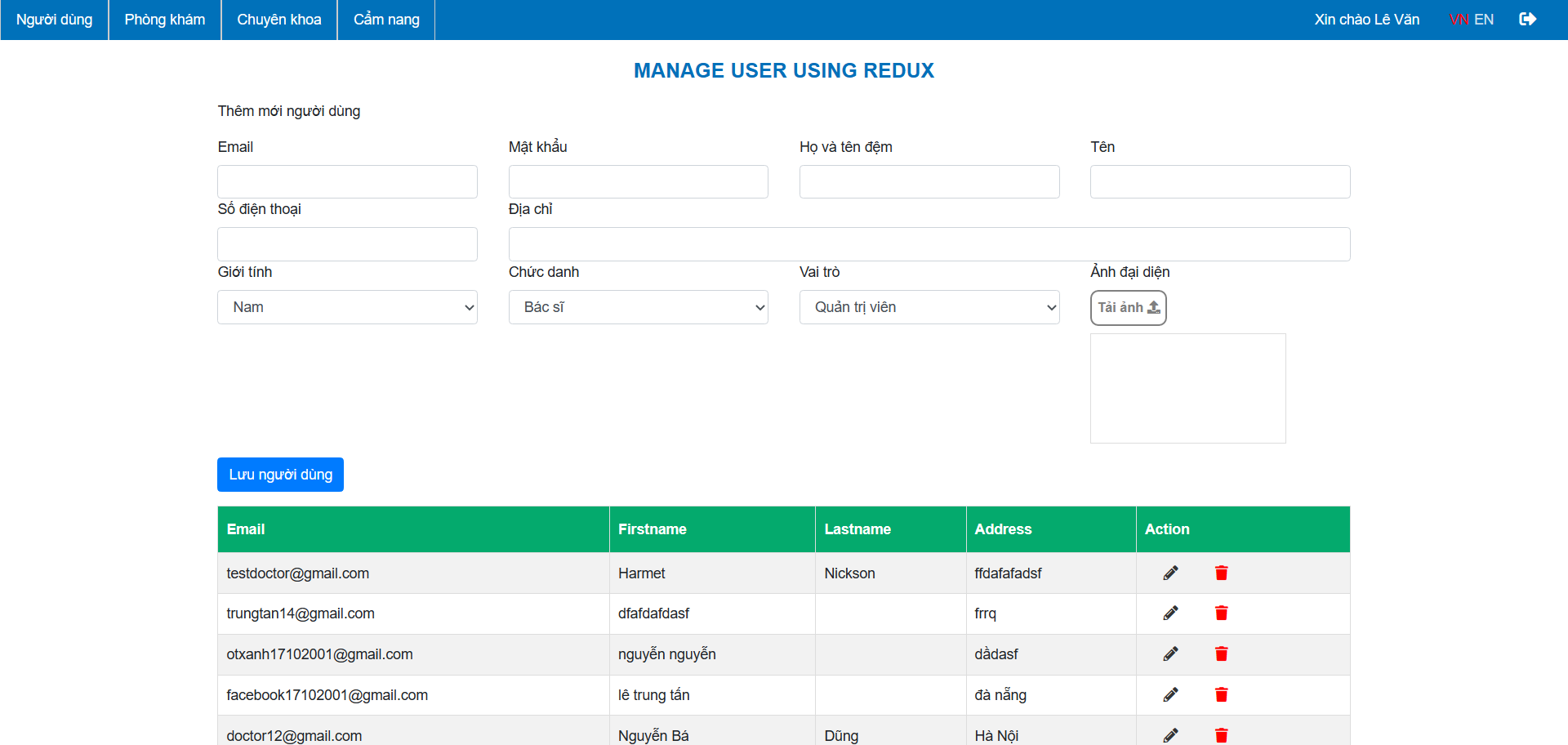
*Hình 3. 4 Giao diện trang thông tin chi tiết bác sĩ*

## **3.5. ĐĂNG NHẬP**

**TÓM TẮT BÁO CÁO**

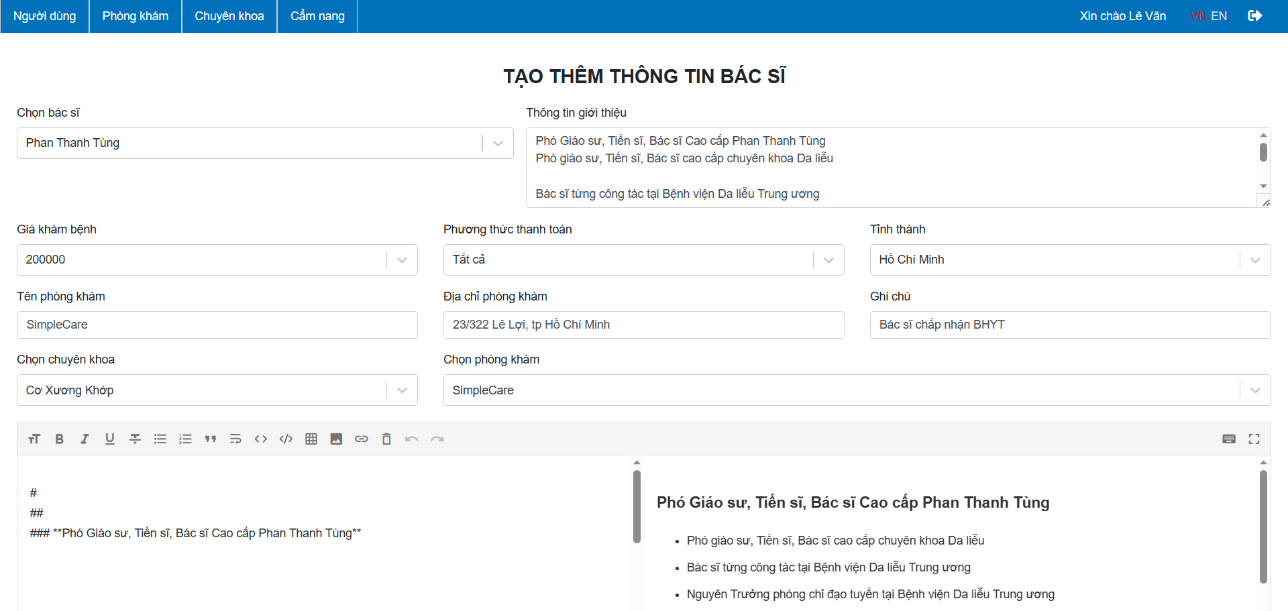
*Hình 3. 5 Giao diện trang Đăng nhập*

## **3.6. QUẢN LÝ TÀI KHOẢN**



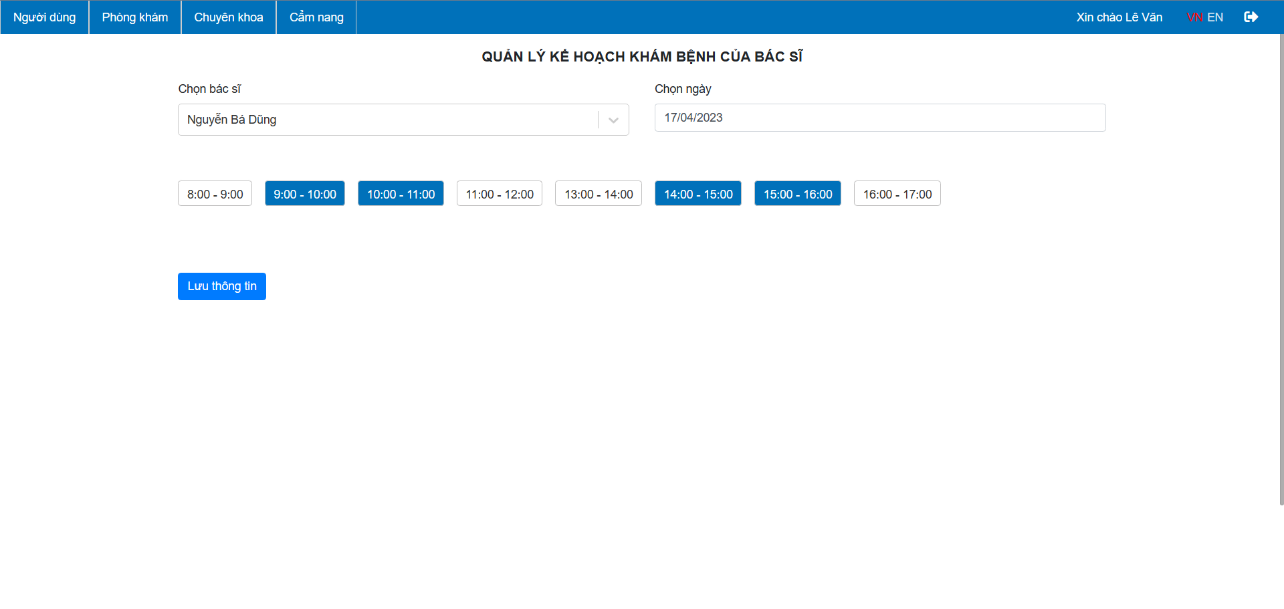
*Hình 3. 6 Giao diện trang Quản lí tài khoản*

## **3.7. QUẢN LÝ THÔNG TIN BÁC SĨ**



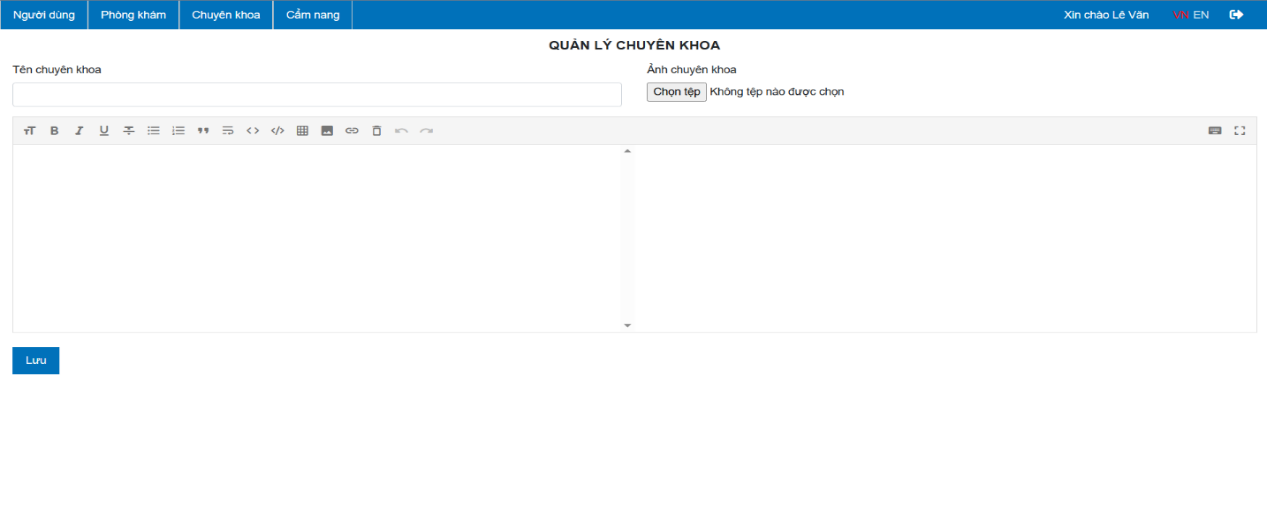
*Hình 3. 7 Giao diện trang Quản lí thông tin bác sĩ*

## **3.8. THÊM LỊCH KHÁM BÁC SĨ**



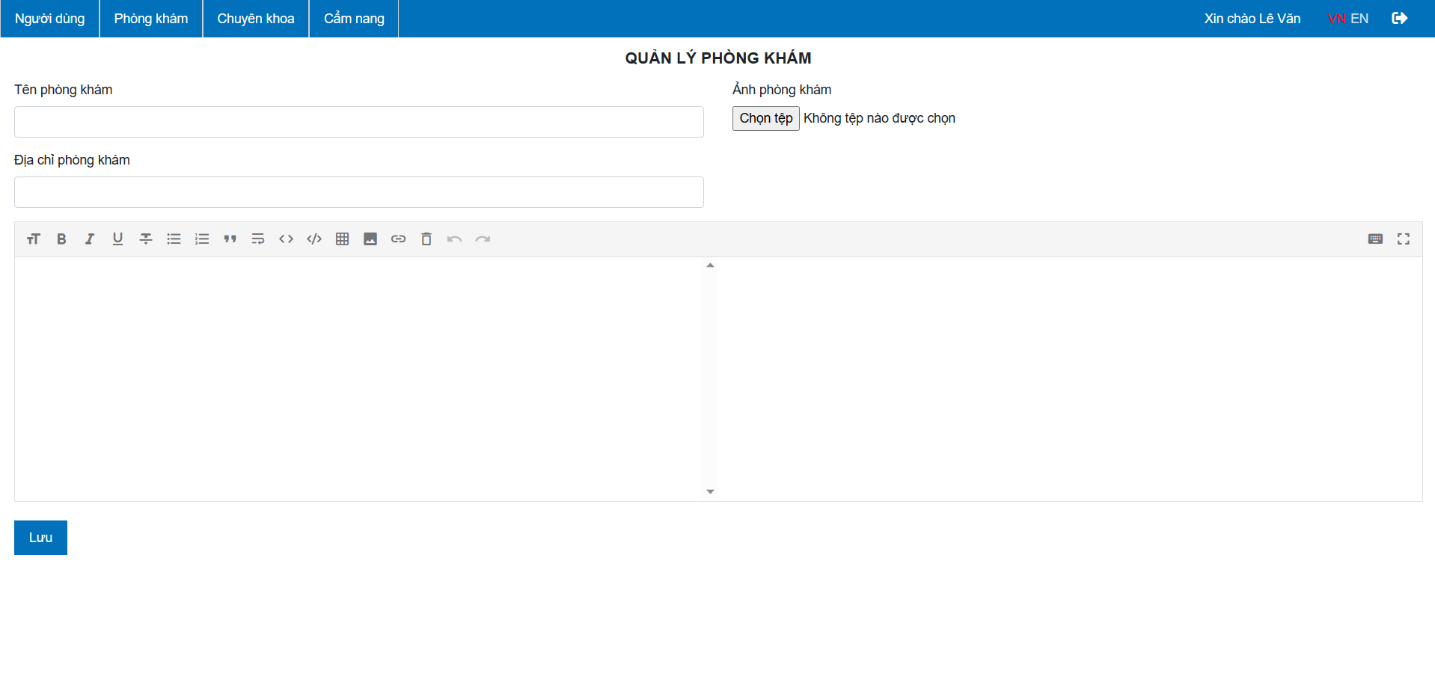
*Hình 3. 8 Giao diện trang Thêm lịch khám bác sĩ*

## **3.9. THÊM CHUYÊN KHOA**



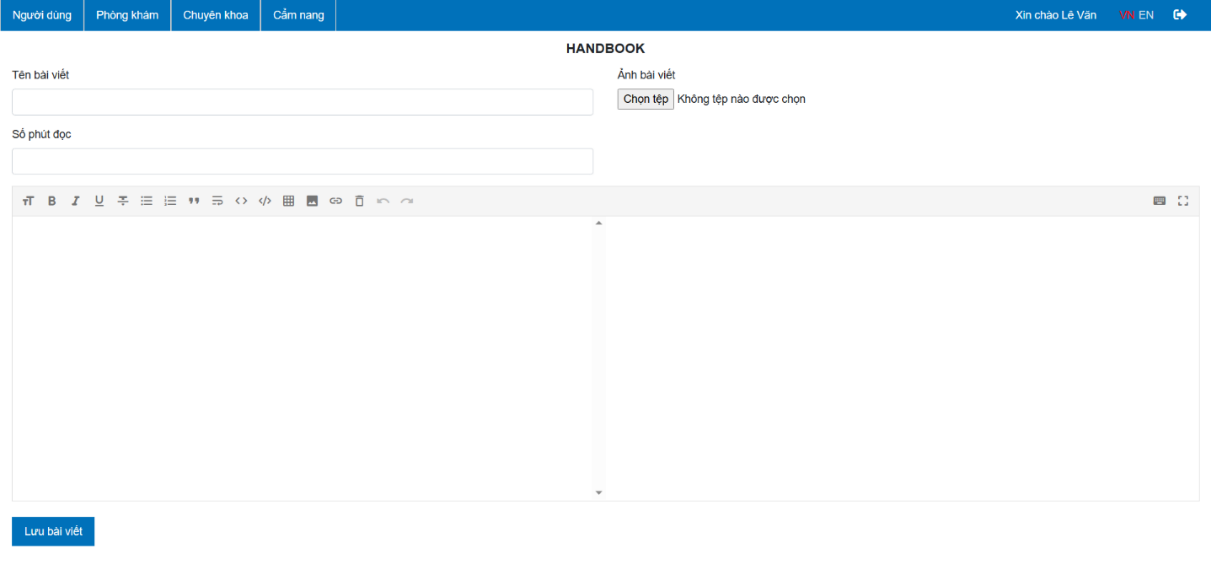
*Hình 3. 9 Giao diện trang Thêm chuyên khoa*

## **3.10. THÊM PHÒNG KHÁM**



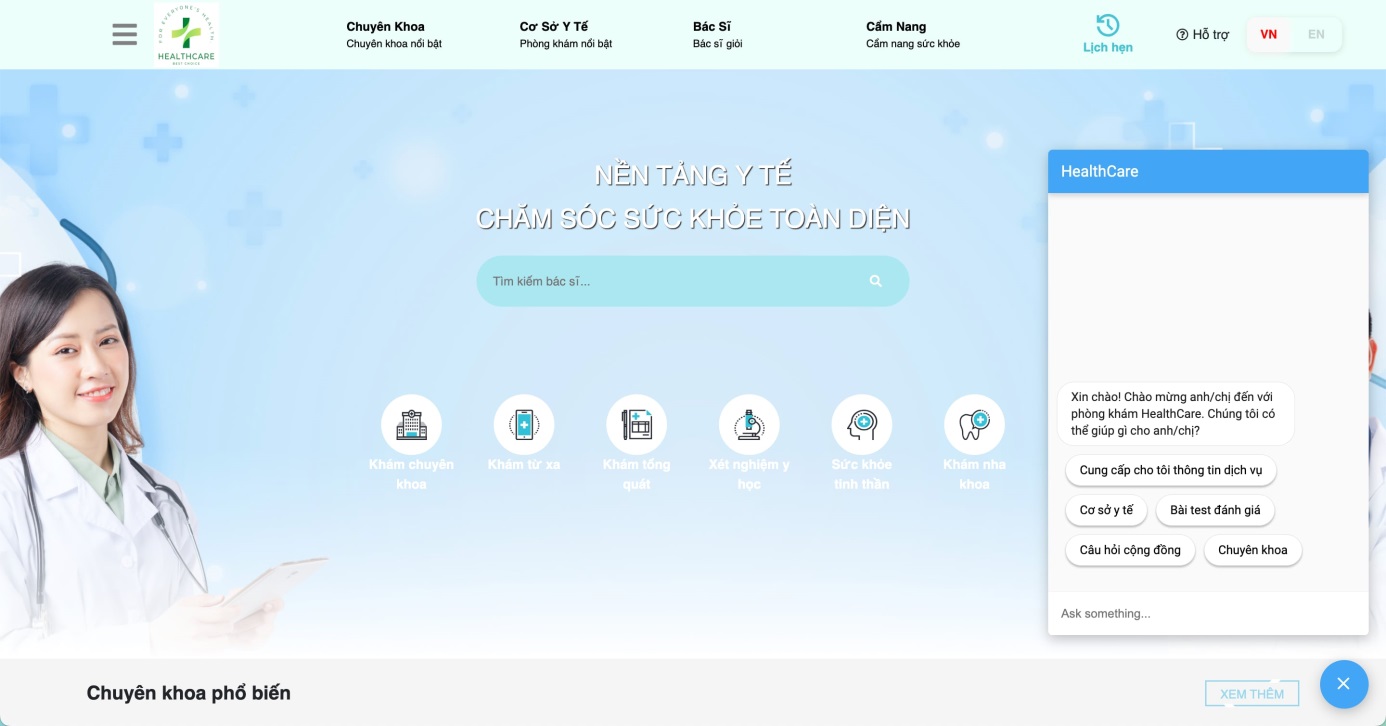
*Hình 3. 10 Giao diện trang Thêm phòng khám*

## **3.11. THÊM CẨM NANG**



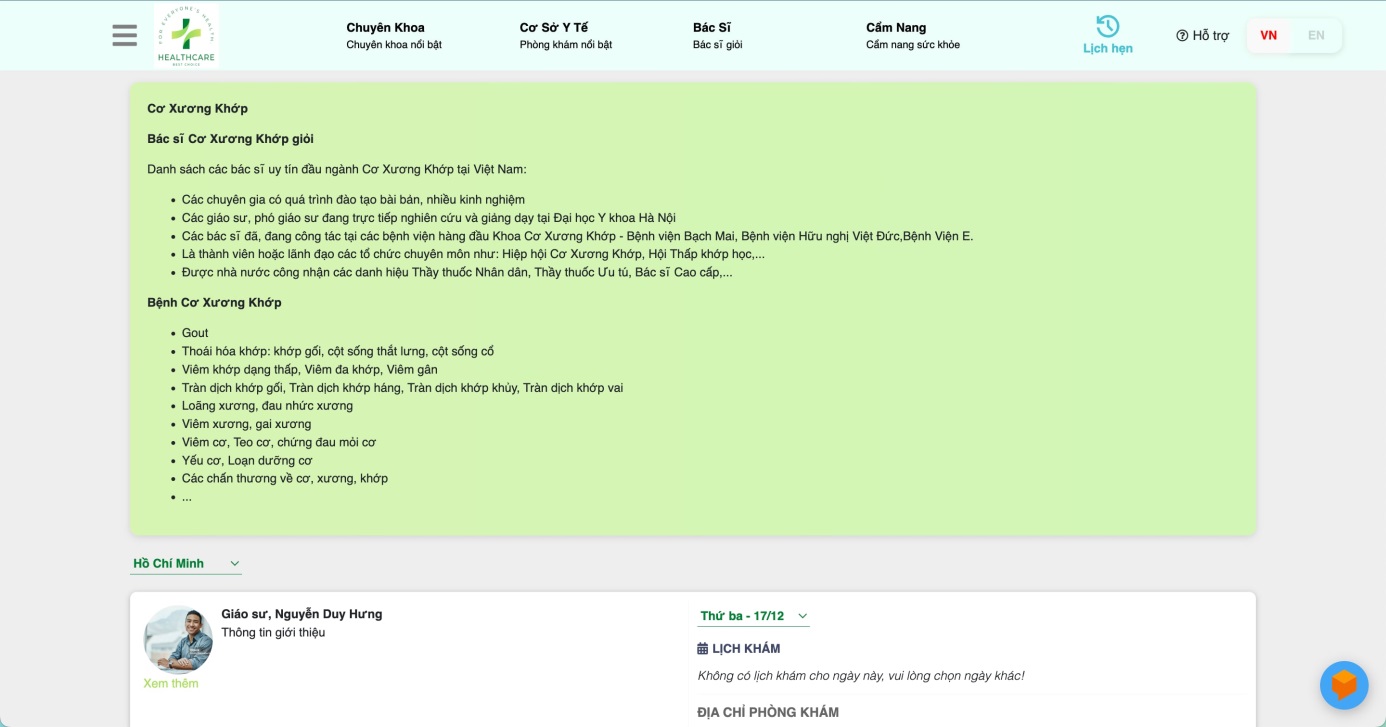
*Hình 3. 11 Giao diện trang Thêm cẩm nang*

## **3.12. Tư vấn sức khỏe trực tuyến**



*Hình 4. 12 Giao diện trang Thêm cẩm nang*

## **3.12. THÔNG TIN CHI TIẾT CẨM NANG**



*Hình 3. 12 Giao diện trang Thông tin chi tiết cẩm nang*

# **KẾT LUẬN**

Qua quá trình thực hiện khóa luận này, đã đạt được những kết quả sau:

**Ưu điểm:**

* Phân tích được các chức năng của toàn bộ hệ thống một cách tổng quát.
* Xây dựng được cơ sở dữ liệu cho website y tế.
* Xây dựng, hoàn thiện được các chức năng cơ bản cho website có thể hỗ trợ cho việc đặt lịch khám:

- Xem thông tin bác sĩ, thông tin phòng khám, thông tin chuyên khoa, xem cẩm nang.

- Quản lý tài khoản, quản lý thông tin bác sĩ, quản lý phòng khám, cẩm nang, chuyên khoa.

- Đặt lịch khám bệnh.

- Tìm kiếm bác sĩ theo tên.

- Tư vấn sức khỏe.

**Điểm hạn chế:**

* Chưa xây dựng được các chức năng gói khám bệnh.
* Giao diện chưa được hoàn thiện.
* Nội dung website chưa đa dạng.

**Hướng phát triển**

* Xây dựng chức năng gói khám bệnh.
* Tích hợp bảo hiểm y tế vào hệ thống.
* Phát triển nội dung website đa dạng .
* Thiết kế cơ sở dữ liệu tối ưu hơn.

# **TÀI LIỆU THAM KHẢO**

**Tài liệu Internet**

[1]. [Quick Start – React](https://react.dev/learn)

[2]. [Model Basics | Sequelize](https://sequelize.org/docs/v6/core-concepts/model-basics/)

[3]. [MySQL :: MySQL Documentation](https://dev.mysql.com/doc/)

[4]. [Mối quan hệ giữa Actor và Use Case - Cộng đồng công nghệ - Khoa Công nghệ thông tin - Đại học Duy Tân (duytan.edu.vn)](https://kcntt.duytan.edu.vn/Home/ArticleDetail/vn/161/2531/moi-quan-he-giua-actor-va-use-case)

[5]. [Axios API | Axios Docs (axios-http.com)](https://axios-http.com/docs/api_intro)

[6]. [React Hooks (w3schools.com)](https://www.w3schools.com/react/react_hooks.asp)

[7]. [UML Class Diagram Tutorial | Lucidchart](https://www.lucidchart.com/pages/uml-class-diagram" \l ":~:text=Class%20diagrams%20are%20one%20of,overwhelming%20as%20it%20might%20appear.)

[8]. [Introduction · Bootstrap v5.0 (getbootstrap.com)](https://getbootstrap.com/docs/5.0/getting-started/introduction/)

[9]. [What is MVC Architecture? (w3schools.in)](https://www.w3schools.in/mvc-architecture)

[10]. [Docs | Node.js (nodejs.org)](https://nodejs.org/ro/docs)