ĐẠI HỌC HUẾ

TRƯỜNG ĐẠI HỌC KHOA HỌC

**KHOA CÔNG NGHỆ THÔNG TIN**

**🙞 🕮 🙜**

**KHOÁ LUẬN**

**TỐT NGHIỆP ĐẠI HỌC**

**Đề tài:**

**XÂY DỰNG ỨNG DỤNG ĐA NỀN TẢNG CHO PHÒNG KHÁM NHA KHOA**

Sinh viên thực hiện: **LÊ NGUYỄN TRUNG PHƯỚC**

Khóa: K43 - Hệ chính quy

**Huế, 05 – 2023**

ĐẠI HỌC HUẾ

TRƯỜNG ĐẠI HỌC KHOA HỌC

**KHOA CÔNG NGHỆ THÔNG TIN**

**🙞 🕮 🙜**

**KHOÁ LUẬN**

**TỐT NGHIỆP ĐẠI HỌC**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài:**

**XÂY DỰNG ỨNG DỤNG ĐA NỀN TẢNG CHO PHÒNG KHÁM NHA KHOA**

Sinh viên thực hiện: **LÊ NGUYỄN TRUNG PHƯỚC**

Khóa: K43 - Hệ chính quy

Giáo viên hướng dẫn: ThS. NGUYỄN DŨNG

**Huế, 05 – 2023**

# 

# LỜI CẢM ƠN

Lời đầu tiên, cho em được gửi lời cảm ơn đến tất cả những người đã đồng hành cùng em trong suốt quãng đường học tập và rèn luyện cho đến ngày hôm nay, đặc biệt là bố mẹ đã sinh ra, nuôi nấng và dạy dỗ em.

Tiếp theo là lời cảm ơn sâu sắc đến quý thầy cô giáo khoa Công nghệ thông tin – Trường Đại học Khoa Học Huế đã tận tình dạy bảo, truyền đạt kiến thức, cho em một hành trang tri thức để sau này vững bước trên con đường đã chọn.

Em cũng xin chân thành cảm ơn thầy giáo – Thạc sĩ Nguyễn Dũng đã hướng dẫn, hỗ trợ em hoàn thành khóa luận tốt nghiệp.

Cuối cùng em muốn gửi lời cảm ơn đến bạn bè, các anh chị cùng khoa và công ty 3S HUE Intersoft đã giúp đỡ em trong suốt quá trình học tập và rèn luyện trong 4 năm học ở trường.

Em xin chân thành cảm ơn.

# DANH MỤC HÌNH ẢNH, ĐỒ THỊ, SƠ ĐỒ

[Hình 1.1 Biên dịch và thực thi trong .NET 8](#_Toc135039057)

[Hình 2.1 Biểu đồ Use Case các chức năng trong hệ thống quản lý phòng khám nha khoa 17](#_Toc135039058)

[Hình 2.2 Biểu đồ Use case các chức năng trong hệ thống quản lý phòng khám nha khoa 20](#_Toc135039059)

[Hình 2.3 Biểu đồ hoạt động chức năng đăng nhập của tài khoản quản lý hệ thống 23](#_Toc135039060)

[Hình 2.4 Biểu đồ hoạt động chức năng đăng xuất của tài khoản quản lý hệ thống 23](#_Toc135039061)

[Hình 2.5 Biểu đồ hoạt động chức năng thêm lịch hẹn của tài khoản quản lý hệ thống 24](#_Toc135039062)

[Hình 2.6 Biểu đồ hoạt động chức năng chỉnh sửa lịch hẹn của tài khoản quản lý hệ thống 25](#_Toc135039063)

[Hình 2.7 Biểu đồ hoạt động chức năng xóa lịch hẹn của tài khoản quản lý hệ thống 25](#_Toc135039064)

[Hình 2.8 Biểu đồ hoạt động chức năng xác nhận lịch hẹn của tài khoản quản lý hệ thống 26](#_Toc135039065)

[Hình 2.9 Biểu đồ hoạt động chức năng thêm khách hàng của tài khoản quản lý hệ thống 27](#_Toc135039066)

[Hình 2.10 Biểu đồ hoạt động chức năng chỉnh sửa khách hàng của tài khoản quản lý hệ thống 28](#_Toc135039067)

[Hình 2.11 Biểu đồ hoạt động chức năng thêm sổ sức khỏe của tài khoản quản lý hệ thống 29](#_Toc135039068)

[Hình 2.12 Biểu đồ hoạt động chức năng chỉnh sửa sổ sức khỏe của tài khoản quản lý hệ thống 30](#_Toc135039069)

[Hình 2.13 Biểu đồ hoạt động chức năng xóa sổ sức khỏe của tài khoản quản lý hệ thống 30](#_Toc135039070)

[Hình 2.14 Biểu đồ hoạt động chức năng in hóa đơn của tài khoản quản lý hệ thống 31](#_Toc135039071)

[Hình 2.15 Biểu đồ hoạt động chức năng xem chi tiết sổ sức khỏe của tài khoản quản lý hệ thống 31](#_Toc135039072)

[Hình 2.16 Biểu đồ hoạt động chức năng thêm chi tiết sổ sức khỏe của tài khoản quản lý hệ thống 32](#_Toc135039073)

[Hình 2.17 Biểu đồ hoạt động chức năng chỉnh sửa chi tiết sổ sức khỏe của tài khoản quản lý hệ thống 33](#_Toc135039074)

[Hình 2.18 Biểu đồ hoạt động chức năng xóa chi tiết sổ sức khỏe của tài khoản quản lý hệ thống 33](#_Toc135039075)

[Hình 2.19 Biểu đồ hoạt động chức năng xem thông tin phòng khám của khách 34](#_Toc135039076)

[Hình 2.20 Biểu đồ hoạt động chức năng đặt lịch hẹn của khách 35](#_Toc135039077)

[Hình 2.21 Biểu đồ hoạt động chức năng đăng nhập của khách hàng 36](#_Toc135039078)

[Hình 2.22 Biểu đồ hoạt động chức năng đăng xuất của khách hàng 36](#_Toc135039079)

[Hình 2.23 Biểu đồ hoạt động chức năng đổi mật khẩu của khách hàng 37](#_Toc135039080)

[Hình 2.24 Biểu đồ hoạt động chức năng xem danh sách sổ sức khỏe của khách hàng 38](#_Toc135039081)

[Hình 2.25 Biểu đồ hoạt động chức năng xem chi tiết sổ sức khỏe của khách hàng 39](#_Toc135039082)

[Hình 2.26 Biểu đồ hoạt động chức nhắc lịch tái khám của khách hàng 39](#_Toc135039083)

[Hình 2.27 Lược đồ cơ sở dữ liệu 40](#_Toc135039084)

[Hình 3.1 Giao diện đăng nhập 52](#_Toc135039085)

[Hình 3.2 Giao diện cho chức năng đăng xuất 52](#_Toc135039086)

[Hình 3.3 Giao diện danh sách lịch hẹn 53](#_Toc135039087)

[Hình 3.4 Giao diện thêm lịch hẹn 54](#_Toc135039088)

[Hình 3.5 Giao diện sửa lịch hẹn 55](#_Toc135039089)

[Hình 3.6 Giao diện xóa lịch hẹn 56](#_Toc135039090)

[Hình 3.7 Giao diện xác nhận lịch hẹn 56](#_Toc135039091)

[Hình 3.8 Giao diện sau khi xác nhận lịch hẹn thành công 57](#_Toc135039092)

[Hình 3.9 Giao diện danh sách khách hàng 57](#_Toc135039093)

[Hình 3.10 Giao diện thêm khách hàng 58](#_Toc135039094)

[Hình 3.11 Thông báo thêm thất bại khi trùng số điện thoại 58](#_Toc135039095)

[Hình 3.12 Giao diện sửa thông tin khách hàng 59](#_Toc135039096)

[Hình 3.13 Giao diện danh sách sổ sức khỏe 60](#_Toc135039097)

[Hình 3.14 Giao diện thêm sổ sức khỏe 60](#_Toc135039098)

[Hình 3.15 Giao diện sửa thông tin sổ sức khỏe 61](#_Toc135039099)

[Hình 3.16 Giao diện xóa sổ sức khỏe 62](#_Toc135039100)

[Hình 3.17 Giao diện hóa đơn 62](#_Toc135039101)

[Hình 3.18 Giao diện danh sách chi tiết sổ sức khỏe 63](#_Toc135039102)

[Hình 3.19 Giao diện thêm chi tiết sổ sức khỏe 63](#_Toc135039103)

[Hình 3.20 Giao diện sửa thông tin chi tiết sổ sức khỏe 64](#_Toc135039104)

[Hình 3.21 Giao diện xóa chi tiết sổ sức khỏe 65](#_Toc135039105)

[Hình 3.22 Giao diện xem, tải tài liệu đính kèm 65](#_Toc135039106)

[Hình 3.23 Giao diện thông tin về dịch vụ trên trang web 66](#_Toc135039107)

[Hình 3.24 Giao diện bảng giá trên trang web 66](#_Toc135039108)

[Hình 3.25 Giao diện danh sách bác sĩ trên trang web 67](#_Toc135039109)

[Hình 3.26 Giao diện danh sách tin tức trên trang web 67](#_Toc135039110)

[Hình 3.27 Giao diện thông tin về dịch vụ trên điện thoại 68](#_Toc135039111)

[Hình 3.28 Giao diện bảng giá trên điện thoại 68](#_Toc135039112)

[Hình 3.29 Giao diện thông tin về bác sĩ trên điện thoại 69](#_Toc135039113)

[Hình 3.30 Giao diện tin tức trên điện thoại 69](#_Toc135039114)

[Hình 3.31 Giao diện đặt lịch hẹn trên trang web 70](#_Toc135039115)

[Hình 3.32 Giao diện đặt lịch hẹn trên điện thoại 70](#_Toc135039116)

[Hình 3.33 Giao diện đăng nhập trên trang web 71](#_Toc135039117)

[Hình 3.34 Giao diện đăng nhập trên trang web 71](#_Toc135039118)

[Hình 3.35 Giao diện đăng xuất trên trang web 72](#_Toc135039119)

[Hình 3.36 Giao diện đăng xuất trên trang web 72](#_Toc135039120)

[Hình 3.37 Giao diện sổ sức khỏe của khách hàng trên trang web 73](#_Toc135039121)

[Hình 3.38 Giao diện sổ sức khỏe của khách hàng trên điện thoại 73](#_Toc135039122)

[Hình 3.39 Giao diện chi tiết sổ sức khỏe của khách hàng trên trang web 74](#_Toc135039123)

[Hình 3.40 Giao diện chi tiết sổ sức khỏe của khách hàng trên trang điện thoại 74](#_Toc135039124)

[Hình 3.41 Giao diện chi tiết sổ sức khỏe của khách hàng trên trang web 75](#_Toc135039125)

[Hình 3.42 Giao diện chi tiết sổ sức khỏe của khách hàng trên điện thoại 75](#_Toc135039126)

[Hình 3.43 Giao diện đổi mật khẩu trên trang web 76](#_Toc135039127)

[Hình 3.44 Giao diện đặt thông báo nhắc lịch tái khám 76](#_Toc135039128)

# MỤC LỤC

[LỜI CẢM ƠN i](#_Toc135037800)

[DANH MỤC HÌNH ẢNH, ĐỒ THỊ, SƠ ĐỒ ii](#_Toc135037801)

[MỤC LỤC v](#_Toc135037802)

[MỞ ĐẦU 1](#_Toc135037803)

[CHƯƠNG 1. GIỚI THIỆU CÔNG NGHỆ 2](#_Toc135037804)

[1.1. Angular 2](#_Toc135037805)

[1.1.1. Giới thiệu chung về Angular 2](#_Toc135037806)

[1.1.2. Các tính năng nổi bật của Angular 2](#_Toc135037807)

[1.1.3. Ưu, nhược điểm của Angular 3](#_Toc135037808)

[1.2. Flutter 4](#_Toc135037809)

[1.2.1. Giới thiệu chung về Flutter 4](#_Toc135037810)

[1.2.2. Kiến trúc trong Flutter 5](#_Toc135037811)

[1.2.3. Các tính năng nổi bật của Flutter 6](#_Toc135037812)

[1.2.4. Ưu, nhược điểm của Flutter 6](#_Toc135037813)

[1.3.1. Giới thiệu chung về .NET 8](#_Toc135037814)

[1.3.2. Biên dịch và thực thi trong .NET 8](#_Toc135037815)

[1.3.3. Các tính năng nổi bật của .NET 9](#_Toc135037816)

[1.3.4. Ưu nhược điểm của .NET 9](#_Toc135037817)

[CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 10](#_Toc135037818)

[2.1. Phân tích hệ thống 10](#_Toc135037819)

[2.1.1. Tổng quan 10](#_Toc135037820)

[2.1.2. Quy trình nghiệp vụ 10](#_Toc135037821)

[2.1.3. Phân tích chức năng 11](#_Toc135037822)

[2.2. Thiết kế hệ thống 16](#_Toc135037823)

[2.2.1. Thiết kế chức năng 16](#_Toc135037824)

[2.2.2. Thiết kế cơ sở dữ liệu 40](#_Toc135037825)

[CHƯƠNG 3. CÀI ĐẶT HỆ THỐNG 50](#_Toc135037826)

[3.1. Mô hình tổ chức 50](#_Toc135037827)

[3.1.1. Mô hình thiết kế 50](#_Toc135037828)

[3.1.2. Hệ quản trị cơ sở dữ liệu 51](#_Toc135037829)

[3.2. Kết quả cài đặt 51](#_Toc135037830)

[3.2.1. Các chức năng của quản trị hệ thống 51](#_Toc135037831)

[3.2.2. Các chức năng của khách 65](#_Toc135037832)

[KẾT LUẬN 78](#_Toc135037833)

[1. Kết quả đạt được 78](#_Toc135037834)

[2. Vấn đề hạn chế 78](#_Toc135037835)

[3. Hướng phát triển 78](#_Toc135037836)

[TÀI LIỆU THAM KHẢO 80](#_Toc135037837)

# MỞ ĐẦU

Răng - bộ phận đầu tiên của hệ thống tiêu hóa - có vai trò đặc biệt quan trọng đối với sức khỏe của chúng ta. Sở hữu hàm răng chắc khỏe giúp quá trình hấp thụ chất dinh dưỡng dễ dàng hơn và giảm đi gánh nặng cho các bộ phận khác trong hệ thống tiêu hóa. Ngược lại, một hàm răng bị bệnh sẽ gây ra các triệu chứng như ăn không ngon, đau nhức kéo dài,… gây ảnh hưởng rất lớn đến cơ thể con người. Ngoài vai trò đối với sức khỏe, một hàm răng khỏe đẹp sẽ tăng phần tự tin khi giao tiếp với người khác. Ngày nay, nhiều người vẫn thường chủ quan đối với sức khỏe hàm răng của mình, cho rằng răng còn khỏe thì không cần phải đi đến nha sĩ để kiểm tra hoặc sợ đau, sợ chảy máu,… và họ chỉ tới các phòng khám khi răng trong tình trạng xấu. Điều đó lý giải cho việc “Việt Nam có tới 90% người dân mắc các bệnh về răng miệng” (Nguồn: Hội Răng Hàm Mặt Việt Nam). Phòng bệnh hơn chữa bệnh, việc đi đến phòng khám nha khoa thường xuyên có thể giúp phát hiện sớm các bệnh, cho phép nha sĩ có những biện pháp điều trị đơn giản, tiết kiệm tiền chữa trị và giảm thiểu tối đa ảnh hưởng xấu đến sức khỏe của chúng ta.

Trong bối cảnh ngày nay, có rất nhiều phòng khám nha khoa lớn nhỏ chất lượng trên cả nước, có thể đáp ứng được các yêu cầu khám chữa bệnh và làm đẹp cho khách hàng. Cùng với sự bùng nổ của khoa học công nghệ, việc ứng dụng Internet vào hoạt động kinh doanh ngày càng trở nên phổ biến và rộng rãi. Các phòng khám có nhiều cơ hội để đến gần hơn với khách hàng thông qua các diễn đàn, nhiều kênh mạng xã hội và việc sở hữu một website cũng sẽ giúp nâng cao uy tín trong mắt khách hàng lên nhiều lần. Bên cạnh đó, quy trình khám chữa bệnh và làm đẹp bao gồm nhiều công đoạn, với nhiều quy trình nghiệp vụ phức tạp. Tuy nhiên, hiện nay nhiều phòng khám vẫn còn sử dụng phương pháp thủ công để lưu trữ dữ liệu, quản lý việc khám, chữa bệnh và tương tác với khách hàng. Điều này không tránh khỏi sự sai sót trong các quy trình nghiệp vụ, có thể dẫn đến các rủi ro như mất dữ liệu, giấy tờ quan trọng, bảo mật không chặt chẽ, báo cáo thống kê không chính xác và gây thất thoát về mặt kinh tế xuyên suốt quá trình hoạt động, làm suy giảm uy tín, chất lượng của phòng khám. Khách hàng cũng gặp khó khăn trong việc lưu trữ nhiều loại giấy tờ liên quan. Vì vậy, cần thiết phải có một giải pháp để thuận tiện cho việc quản lý phòng khám và giúp khách hàng dễ dàng theo dõi những thông tin, lịch sử khám, chữa bệnh của mình. Vì lý do đó, tôi đã xây dựng ứng dụng đa nền tảng cho phòng khám nha khoa để giải quyết các vấn đề cấp thiết ở trên.

# GIỚI THIỆU CÔNG NGHỆ

## 1.1. Angular

### 1.1.1. Giới thiệu chung về Angular

Angular là một JavaScript Framework dùng để xây dựng các Single Page Application bằng JavaScript, HTML và TypeScript. Code được viết bằng TypeScript sẽ được biên dịch thành JavaScript và hiển thị trong trình duyệt. Được viết bởi Misko Hevery và Adam Abrons và sau đó được Google duy trì và phát triển từ năm 2009. Angular luôn được xem là một trong những Framework Front-end mạnh mẽ nhất và được số lượng lớn developer tin dùng.

Angular có hai phiên bản hoàn toàn khác nhau là AngularJS và Angular. AngularJS dùng để chỉ Angular 1, được ra đời năm 2009 và được viết bằng JavaScript còn Angular là tên gọi chung từ Angular 2 trở lên và nó được viết bằng TypeScript.

### 1.1.2. Các tính năng nổi bật của Angular

**1.1.2.1. Component**

Angular là một tập hợp những component (là một khối code trong Angular Application) và các services cung cấp chức năng trên những component đó, chúng ta tạo các component và sắp xếp chúng lại để tạo ra ứng dụng. Nó là sự kết hợp của template HTML, kèm theo code TypeScript và các Metadata giúp Angular xử lý các lớp. Các components độc lập với nhau và độc lập với hệ thống. Có thể xem component như một control trên màn hình hiển thị, gồm giao diện HTML và code logic xử lý đi kèm control đó. Một component có thể chứa và gọi nhiều component khác nối vào. Chúng ta có thể truyền dữ liệu giữa các component thông qua 2 decorator là @Input và @Output.

**1.1.2.2. Data Binding**

Data Binding là kỹ thuật nơi dữ liệu được đồng bộ giữa component và view. Data Binding có thể chia làm 2 nhóm là One-way Binding và Two-way Binding.

* **One-way binding** là dữ liệu sẽ được truyền theo 1 chiều. Có thể từ view sang component bằng cách sử dụng Interpolation & Property Binding hoặc từ component sang view bằng Property Binding. Chúng ta cũng có thể sử dụng Event Binding để bind các sự kiện như click chuột, sự kiện bàn phím,… để xử lý bên class component.
* **Two-way binding** là cơ chế được developer đánh giá là ấn tượng nhất của Angular. Bất cứ thay đổi nào trên view, dù là nhỏ nhất, cũng đều được tự động cập nhật thuộc tính vào component class và ngược lại.

**1.1.2.3. Directive**

Directive là đối tượng giúp chúng ta thao tác, thêm, sửa, xóa các thành phần giao diện trên website. Hay nói cách khác chúng ta thao tác với các element của DOM. Angular Directive chia làm 3 loại: Component Directives, Structural Directives, Attribute Directives.

* **Component Directives:** Component cho phép định nghĩa selector và gọi ra như một thẻ html tag (<component-name></component-name>)
* **Structural Directives:** Là directive cấu trúc dùng để thay đổi layout của trang web bằng cách thêm hoặc xóa các thành phần trên trang web (DOM Elements). Một số Structural Directives phổ biến có thể kể tới như:
  + **ngFor:** Dùng để lặp các phần tử HTML cho mỗi item, nó giống như vòng lặp for.
  + **ngIf:** Cho phép chúng ta thêm và xóa các thành phần HTML dựa vào điều kiện có được thỏa mãn hay không.
* **Attribute Directives:** Được dùng để thay đổi sự hiển thị hoặc hành vi của một thành phần trên trang web:
  + **ngModel:** Được dùng cho two-way binding.
  + **ngClass:** Được dùng để thêm hoặc xóa một class của thành phần web.
  + **ngStyle:** Được dùng để thêm nhiều thuộc tính css cho một thành phần trang web.

Ngoài các Directive có sẵn trong Angular, chúng ta hoàn toàn có thể tự tạo một directive cho riêng mình.

**1.1.2.4. Routing**

Routing trong Angular là chức năng giúp nạp component vào trang web theo từng route (đường dẫn đã khai báo), đây là chức năng nền tảng để tạo nên Single Page Application. Routing có nhiệm vụ chính là chuyển trang, thay đổi một số thành phần mà không cần phải tải lại trang.

Angular Router là module được tích hợp sau vào Angular, giúp bạn có thể dễ dàng tạo các route cho ứng dụng.

**1.1.2.5. Các tính năng nổi bật khác**

Thiết kế module hóa**:** Người dùng phải tạo các Angular Module để tổ chức tốt hơn và quản lý source code.

Hỗ trợ làm việc với hệ thống Backend:Thực thi bất cứ logic nào và nhận dữ liệu về.

### 1.1.3. Ưu, nhược điểm của Angular

**1.1.3.1 Ưu điểm:**

* Angular cho phép người dùng xây dựng các component của riêng họ và có thể đóng gói chức năng cùng với logic kết xuất thành các thành phần có thể tái sử dụng.
* Mở rộng, phát triển tính năng của các file HTML nhờ sự hiện diện của các Directives.
* Cho phép người dùng viết các service module và đưa chúng vào bất cứ nơi nào họ cần. Điều này cải thiện khả năng kiểm tra và tái sử dụng của các service giống nhau.
* Có thể dễ dàng Unit test mọi phần trong ứng dụng của mình.
* Angular là một full-fledged Framework và có thể cung cấp các giải pháp tiện ích cho giao tiếp máy chủ, định tuyến trong ứng dụng của bạn.
* Cung cấp đa nền tảng và tương thức với nhiều trình duyệt.
* Là Framework phát triển liên tục, đi kèm với đó là sự cập nhật không ngừng của tài liệu được nhiều người đóng góp.

**1.1.3.2. Nhược điểm**

Bên cạnh nhiều ưu điểm đã nêu trên, Angular vẫn còn một số hạn chế như sau:

* Không an toàn: Bản chất của Angular là một Framework Front-end, thường vốn không thể bảo mật bằng Back-end. Vì vậy, nên có phương pháp bảo mật tốt và xác thực từ phía server để giúp ứng dụng an toàn.
* Với một số trình duyệt có tính năng Disable JavaScript nghĩa là trình duyệt đó đã vô hiệu hóa JavaScript thì bạn chỉ có thể nhìn thấy trang web cơ bản.

## 1.2. Flutter

### 1.2.1. Giới thiệu chung về Flutter

Flutter là một bộ công cụ phát triển phần mềm giao diện người dùng với mã nguồn mở do Google tạo ra. Flutter được sử dụng để phát triển các ứng dụng đa nền tảng cho nhiều hệ điều hành như: iOS, Android, Linux,… từ một cơ sở mã nguồn duy nhất.

Tuy Flutter được Google công bố vào đầu năm 2015 nhưng đến tháng 5 năm 2017 nền tảng này mới chính thức ra mắt thị trường. Với mục đích xây dựng và phát triển các ứng dụng chất lượng, mang đến cho người dùng những trải nghiệm tuyệt vời nhất trên hệ điều hành iOS và Android.

Flutter có hai thành phần chính đó là:

* Software Development Kit (SDK): Ở đây tích hợp các công cụ có khả năng phát triển các ứng dụng của bạn, trong đó có các công cụ giúp bạn biên dịch mã của mình thành mã máy gốc (mã cho Android và iOS).
* UI Library based on widgets (Framework): Tổng hợp các thành phần giao diện người dùng (UI) có thể tái sử dụng như: text input, button, slider,… hỗ trợ bạn có thể cá nhân hóa theo nhu cầu sử dụng.

Để phát triển với Flutter, bạn sẽ sử dụng một ngôn ngữ lập trình có tên là Dart. Ngôn ngữ này được tạo ra bởi Google vào tháng 10 năm 2011 nhắm đến việc tạo ra các ứng dụng đa nền tảng – Web, Mobile, Desktop, IoT và được cải tiến rất nhiều trong những năm qua.

### 1.2.2. Kiến trúc trong Flutter

**1.2.2.1. Widgets**

* Khái niệm cốt lõi nhất trong Flutter Framework đó là “Trong Flutter, mọi thứ đều là widget”. Widget là thành phần giao diện cơ bản nhất tạo nên toàn bộ giao diện người dùng của ứng dụng.
* Trong Flutter, bản thân chính ứng dụng đã là một widget. Mỗi ứng dụng chính là một top-level widget và nó bao gồm một hoặc nhiều các widget con, mỗi widget này lại có thể bao gồm một hoặc nhiều widget con khác. Nhờ sự kết hợp linh hoạt này chúng ta có thể tạo ra bất kì ứng dụng phức tạp nào.

**1.2.2.2. Gesture**

* Widget trong Flutter hỗ trợ tương tác thông qua 1 widget đặc biệt, đó là GestureDetector. GestureDetector là 1 widget vô hình có khả năng bắt được tương tác người dùng như chạm, kéo,… của widget con của nó.
* Rất nhiều widget native trong Flutter hỗ trợ tương tác thông qua việc sử dụng GestureDetector. Chúng ta còn có thể kết hợp chức năng tương tác vào trong widget đang tồn tại bằng cách sử dụng widget GestureDetector.

**1.2.2.3. State**

* Widget trong Flutter hỗ trợ bảo trì State bằng 1 widget đặc biệt đó là StatefulWidget. Widget cần phải lấy gốc từ StatefulWidget để hỗ trợ kiểm soát State.
* Điểm này trong Flutter rất giống với ReactJS và StatefulWidget sẽ tự động render lại bất cứ khi nào trạng thái bên trong hệ thống thay đổi.
* Khi render lại thì nó sẽ giúp tối ưu hóa bằng cách tìm kiếm điểm khác biệt giữa các widget UI cũ và mới để render những thứ thực sự cần thiết.

**1.2.2.4. Layers**

* Khái niệm quan trọng nhất của Framework Flutter là Framework được lồng ghép vào trong nhiều loại mức độ phức tạp và được sắp xếp rất rõ ràng các lớp giảm dần độ phức tạp.
* Một layer được tạo ra bằng cách sử dụng trực tiếp layer kế tiếp của nó. Các layer trên cùng là 1 widget chi tiết với Android và iOS. Các layer tiếp theo có sử dụng các widget native của Flutter. Layer càng xuống sâu thì càng chạm tới code cụ thể của nền tảng đó.

### 1.2.3. Các tính năng nổi bật của Flutter

* Flutter sử dụng ngôn ngữ lập trình Dart rất đơn giản và dễ dàng sử dụng. Hiện Flutter đã và đang là một react Framework.
* Tính năng hot reload tiện lợi, dễ trải nghiệm, dễ dàng xây dựng giao diện. Hot reload nâng cao năng suất của lập trình viên, giúp họ tiết kiệm thời gian phát triển sản phẩm, kiểm thử mã nguồn và tìm lỗi nhanh hơn đáng kể.
* Các widget built-in đẹp mắt giúp giao diện người dùng đẹp, hoạt động phong phú, scroll mượt mà, linh hoạt và tự nhiên.
* Framework giải quyết những thách thức khó khăn trong giao diện người dùng bằng việc tập hợp các layout, platform và widget phong phú.
* Khi tiến hành biên dịch để đóng gói sản phẩm, Dart sẽ sử dụng trình biên dịch AOT để có được file thực thi hiệu năng cao. Đây là một đặc tính cho phép DART vừa có khả năng hot reload trong lúc phát triển vừa có tính chất như một ngôn ngữ static typed.
* Flutter thực sự là một single-code base đa nền tảng. Điều này có nghĩa là bạn chỉ cần viết code một lần và nó sẽ hoạt động được trên iOS, Android và thậm chí cả Web.
* Ứng dụng Flutter có thể được viết bằng bất kỳ trình soạn thảo mã nào, nhưng hai ứng dụng được ưu tiên là Android Studio và Visual Studio Code (VS Code). Các lệnh tích hợp trong Flutter (command line tools) cũng cho phép bạn chạy các ứng dụng Flutter trên nhiều thiết bị cùng một lúc. Đây là một tính năng cực kỳ tiện dụng, vì bạn có thể thấy ứng dụng của mình sẽ trông như thế nào và hoạt động như thế nào trên nền tảng iOS và Android, thậm chí cả nền tảng Web.

### 1.2.4. Ưu, nhược điểm của Flutter

**1.2.4.1. Ưu điểm:**

* **Thời gian phát triển nhanh hơn:** Flutter giúp phát triển các ứng dụng nhanh hơn. Bạn có thể khởi chạy ứng dụng của mình trên nhiều nền tảng thông qua việc sử dụng single code-base của Flutter. Bạn chỉ cần viết code một lần thay vì phải viết code riêng cho từng hệ điều hành. Do đó, giúp tiết kiệm về chi phí mà lại cực kỳ hiệu quả.
* **Tự do tùy chỉnh và kết xuất nhanh hơn:** Flutter có kiến trúc phân lớp sẽ giúp bạn tự do tùy chỉnh các thành phần giao diện người dùng. Đặc biệt, UI của Flutter không cần các thành phần giao diện người dùng dành riêng cho từng nền tảng. Flutter tiếp cận rất đơn giản và hiệu quả, giúp nhất quán giao diện người dùng trên nhiều nền tảng.
* **Tính năng hot reload:** Tính năng hot-reload được xem là tính năng vượt trội, mang tính cách mạng của Flutter giúp nó hoàn toàn khác biệt với các đối thủ cạnh tranh. Khi sử dụng hot reload, có thể thay đổi mã và nhìn thấy chúng ở thời gian chạy trong bản xem trước ứng dụng. Không cần phải biên dịch lại code mỗi lần, thêm tính năng và sửa lỗi nhanh hơn, giúp bạn tự do thử nghiệm với các giao diện người dùng khác nhau khi xây dựng. Đồng thời, không làm mất trạng thái, trên emulator, simulator và device cho iOS và Android khi tải lại lần thứ hai.
* **Tương đương với hiệu suất ứng dụng gốc:** UX tốt sẽ mang đến thành công cho ứng dụng của bạn, ngược lại hiệu suất kém sẽ ảnh hưởng nghiêm trọng đến trải nghiệm ứng dụng của người dùng. Ứng dụng di động được phát triển bằng Flutter sẽ không phân biệt được với các ứng dụng tạo bằng mã gốc. Thực tế, ứng dụng của Flutter có thể hoạt động tốt hơn trong các tình huống giao diện người dùng phức tạp. Bởi Flutter không dựa trên cầu nối JavaScript hay bất kỳ code trung gian nào mà được tích hợp trực tiếp vào mã máy. Từ đó giúp loại bỏ lỗi hiệu suất có thể xảy ra.
* **Tài liệu tốt và một cộng đồng mạnh:** Flutter sở hữu nhiều ưu điểm vượt trội với nhiều Framework khác nên thu hút nhiều lập trình viên, từ đó tạo nên một cộng đồng tích cực có thể cung cấp hay hỗ trợ cho nhau.

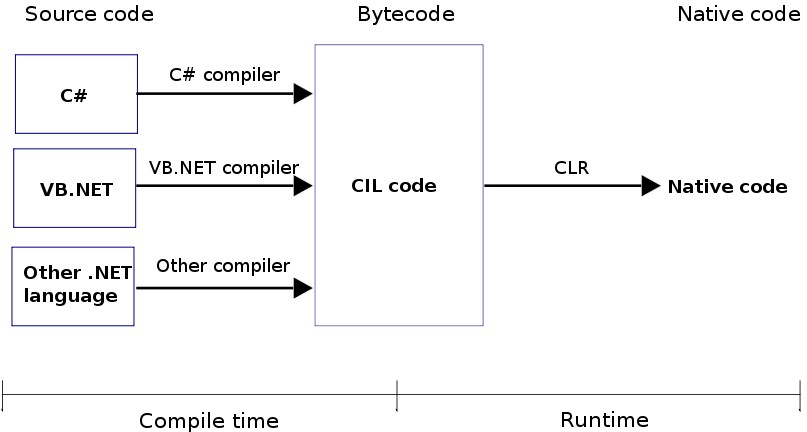
**1.2.4.2. Nhược điểm:**

* **Chưa ổn định:** Flutter là một trong những Framework mới và đang tiếp tục phát triển. Nhiều thư viện của nó cần phải được kiểm tra trước khi ổn định. Vì vậy, Flutter cần có thời gian để khai thác tiềm năng của một hệ điều hành.
* **Kích thước tệp lớn:** Kích thước tệp lớn là một nhược điểm khi sử dụng Flutter. Bởi các ứng dụng được tạo bằng Flutter rất nặng, mất rất nhiều thời gian để khởi động hoặc tải. Điều này có thể làm hiệu suất kém, trải nghiệm từ người dùng không tốt.
* **Các thay đổi và cập nhật trong Framework:** Do Flutter là một Framework khá mới mẻ nên thường có những thay đổi hay cập nhật nên có thể ảnh hưởng đến việc phát triển. Vì vậy, duy trì code có lẽ là thách thức khi thay đổi nhanh như vậy.
* **Thiếu hướng dẫn phát triển tiêu chuẩn:** Các Framework ổn định có các hướng dẫn được soạn thảo chuẩn hóa nên dễ phát triển hơn. Flutter mới ra đời, còn thiếu các nguyên tắc như vậy, sẽ gặp khó khăn khi xây dựng các ứng dụng phức tạp. 1.3. .NET

### 1.3.1. Giới thiệu chung về .NET

.NET là một nền tảng mã nguồn mở được phát triển bởi Microsoft nhằm mục đích xây dựng các ứng dụng dành cho Desktop, Web, Mobile, IoT ,…có thể chạy trên nhiều hệ điều hành. Hệ thống .NET bao gồm các công cụ, thư viện và ngôn ngữ hỗ trợ phát triển phần mềm hiện đại, linh hoạt và hiệu suất cao. Để phát triển .NET, bạn có thể sử dụng nhiều ngôn ngữ như C#, F# hoặc Visual Basic.

### 1.3.2. Biên dịch và thực thi trong .NET



Hình . Biên dịch và thực thi trong .NET

Trong hình trên, phần Compile time là quá trình build, còn Runtime là quá trình thực thi chương trình.

Sau khi viết code xong, ta có thể dùng tool Visual Studio để build hoặc dùng lệnh ‘dotnet build’, lúc này source code sẽ được biên dịch sang một ngôn ngữ trung gian được gọi là MSIL (Microsoft Intermediate Language). Quá trình này gọi là Compile time.

Khi ứng dụng được chạy, một thành phần trong .NET có tên là CLR (Common Language Runtime) sẽ tiến hành dịch mã MSIL thành mã máy (Native Code) để máy tính có thể hiểu và thực thi. Quá trình này gọi là JIT (Just-In-Time) hay Runtime.

Một số ứng dụng .NET thường chạy lần đầu lâu hơn những lần sau đó rất nhiều là vì lần tiên được chạy CLR tiến hành dịch mã MSIL thành mã máy (quá trình JIT), ở những lần chạy tiếp theo, không cần phải JIT nữa vì nó đã được biên dịch trước đó, nên thời gian thực thi nhanh hơn nhiều.

### 1.3.3. Các tính năng nổi bật của .NET

* .NET có khả năng tạo ra sự nhất quán cho mọi trải nghiệm của nhà phát triển với những ứng dụng khác nhau ví dụ như: App windows, App web,…
* .NET sở hữu những thư viện lập trình rất lớn, có khả năng hỗ trợ tối đa cho việc tạo lập, xây dựng các ứng dụng web; Truy cập, kết nối các CSDL, cấu trúc dữ liệu; Lập trình giao diện,…
* Cung cấp môi trường thực thi cùng mục đích giải quyết rắc rối, xung đột có mối liên quan đến hiệu suất. Thúc đẩy quá trình phát triển, triển khai mã an toàn.
* .NET có phần kiến trúc bảo mật được thiết kế theo dạng từ dưới lên. Điều này giúp bảo vệ dữ liệu cũng như các ứng dụng khỏi sự đe dọa của hacker thông qua mô hình bảo mật tinh vi là evidence-based.
* Lập trình, thiết kế ứng dụng với .NET tiết kiệm rất nhiều thời gian bởi nó cung cấp sẵn khá nhiều thành phần dùng trong thiết kế. Chỉ cần bạn biết cách sử dụng cũng như tùy biến những đoạn code có sẵn này cho phù hợp với dự án của mình, việc lập trình sẽ trở nên đơn giản hơn bao giờ hết.
* .NET là nền tảng hỗ trợ cho đa ngôn ngữ. Lập trình viên có thể sử dụng nền tảng này để tạo lập các ứng dụng web bằng nhiều ngôn ngữ lập trình khác nhau mà vẫn đảm bảo khả năng tích hợp của nó.

### 1.3.4. Ưu nhược điểm của .NET

**1.3.4.1. Ưu điểm:**

* Nhiều ngôn ngữ hỗ trợ: .NET hỗ trợ nhiều ngôn ngữ lập trình như C#, F#, VB.NET và nhiều ngôn ngữ khác, cho phép lập trình viên chọn ngôn ngữ phù hợp với yêu cầu của dự án.
* Dễ dàng tích hợp với các hệ thống khác: .NET có thể dễ dàng tích hợp với các hệ thống khác như các ứng dụng Web, Mobile, IoT và nhiều hệ thống khác.
* Cung cấp các thư viện phổ biến: .NET cung cấp rất nhiều thư viện phổ biến cho phép lập trình viên dễ dàng tích hợp và sử dụng các tính năng mạnh mẽ.
* An toàn và bảo mật cao: .NET cung cấp các công cụ và tính năng an toàn và bảo mật cao để giữ cho dữ liệu và hệ thống an toàn.
* Cộng đồng lập trình viên lớn: .NET có một cộng đồng lập trình viên sử dụng rộng rãi, dễ dàng trao đổi kiến thức chuyên môn.

**1.3.4.2. Nhược điểm:**

* Tốc độ chạy chậm hơn so với các ngôn ngữ lập trình khác như C++ và Assembly.
* Cấu hình máy tính cần cao để chạy .NET một cách hiệu quả.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## 2.1. Phân tích hệ thống

### 2.1.1. Tổng quan

**Tên khóa luận:** Xây dựng ứng dụng đa nền tảng cho phòng khám nha khoa.

**Mục tiêu:**

* Là kênh quảng bá hình ảnh, giới thiệu thông tin dịch vụ phòng khám nha khoa tới khách hàng.
* Tăng độ uy tín, chuyên nghiệp và tin tưởng trong mắt khách hàng.
* Giảm bớt khối lượng công việc mà nhân viên phòng khám phải làm từ đó -nâng cao chất lượng phục vụ.
* Thu hút khách hàng mới tìm hiểu, thuận tiện cho khách hàng cũ theo dõi sổ sức khỏe điện tử, lịch tái khám và cập nhật các dịch vụ, tin tức của phòng khám.
* Quảng cáo phòng khám với chi phí thấp nhất, đem lại hiệu quả tốt nhất trong thời gian dài.
* Thu hút khách hàng bằng thiết kế giao diện của ứng dụng nha khoa.
* Ứng dụng được coi là bộ mặt của phòng khám trên thị trường online. Để đưa được nhiều khách hàng hơn nữa đến phòng khám, ngoài việc đầu tư vào chất lượng dịch vụ khám chữa bệnh, phải thu hút được khách hàng truy cập vào ứng dụng, từ đó đưa ra bằng chứng thuyết phục họ có thể tin tưởng phòng khám.

### 2.1.2. Quy trình nghiệp vụ

Một số quy trình chính trong việc tương tác giữa phòng khám và khách hàng được mô tả chi tiết dưới đây.

**2.1.2.1. Quy trình đặt lịch khám**

* Khách hàng có thể tham khảo trước thông tin chi tiết và chi phí của các dịch vụ mà phòng khám cung cấp trên ứng dụng của phòng khám. Ngoài ra, khách hàng có thể xem thông tin các bác sĩ, ý tá và đọc tin tức trên ứng dụng.
* Để đặt lịch thì khách hàng phải điền đủ thông tin cá nhân vào biểu mẫu mà phòng khám đã cung cấp trên ứng dụng.
* Sau khi gửi biểu mẫu thành công thì nhân viên phòng khám sẽ nhận được thông tin đặt lịch của khách hàng. Sau đó, nhân viên sẽ liên lạc với khách hàng qua số điện thoại đã cung cấp trong biểu mẫu đó. Nếu xác nhận thành công thì sẽ chỉnh sửa trạng thái xác nhận của lịch hẹn đó, ngược lại sẽ xóa.

**2.1.2.2. Quy trình nghiệp vụ của sổ sức khỏe điện tử**

* Khi khách hàng tới sử dụng dịch vụ tại phòng khám, nếu là lần đầu tiên thì nhân viên sẽ thêm thông tin khách hàng vào hệ thống. Sau đó, nhân viên lập sổ sức khỏe điện tử lưu thông tin các dịch vụ mà khách hàng chỉ định.
* Sau khi tạo sổ sức khỏe điện tử thành công thì nhân viên sẽ đưa đến các phòng chức năng tương ứng với các dịch vụ đã chỉ định. Sau khi sử dụng dịch vụ trong phòng đó thì bác sĩ hoặc nhân viên có thể điền thông tin của lần khám đó vào trong sổ sức khỏe điện tử.
* Khách hàng có thể theo dõi thông tin khám bằng cách đăng nhập vào website hoặc sử dụng ứng dụng điện thoại của khách hàng.

### 2.1.3. Phân tích chức năng

**2.1.3.1. Khách hàng**

**2.1.3.1.1. Xem thông tin phòng khám**

* Mục đích: Cho phép khách hàng đọc tin tức, xem thông tin bác sĩ, địa chỉ các phòng khám, giờ làm việc, tham khảo các dịch vụ, bảng giá,… của phòng khám.
* Mô tả:
  + Khách hàng có thể truy cập trực tiếp lên trang web của phòng khám để xem những thông tin nêu trên hoặc tải ứng dụng trên điện thoại của phòng khám về máy.
  + Nếu khách hàng sử dụng ứng dụng điện thoại, khách hàng có thể sử dụng thêm các chức năng như đặt lịch nhắc ngày tái khám, đọc một số tin tức khuyến mãi chỉ có trên điện thoại.

**2.1.3.1.2. Đặt lịch khám**

* Mục đích: Khách hàng nếu đặt lịch khám thành công sẽ được ưu tiên so với khách hàng chưa đặt lịch trước, vậy nên không cần mất nhiều thời gian để chờ đợi để sử dụng dịch vụ.
* Mô tả:
  + Khách hàng có thể truy cập trực tiếp trên website hoặc tải ứng dụng điện thoại của phòng khám.
  + Chọn vào phần đặt lịch và điền đầy đủ thông tin được yêu cầu trong biểu mẫu.
  + Khách hàng sẽ nhận được cuộc gọi để xác nhận của nhân viên phòng khám.

**2.1.3.1.3. Đăng nhập, đăng xuất**

**a. Đăng nhập**

* Mục đích: Cho phép khách hàng đăng nhập vào để xem thông tin sổ sức khỏe điện tử.
* Mô tả:
  + Khi đăng nhập, khách hàng phải điền đầy đủ 2 trường là số điện thoại và mật khẩu.
  + Số điện thoại là số đã đăng ký với phòng khám khi khách tới sử dụng dịch vụ, mật khẩu nếu đăng nhập lần đầu sẽ là số điện thoại, khách hàng có thể đổi được mật khẩu sau khi đăng nhập vào trang web.
  + Hệ thống sẽ tiến hành kiểm tra thông tin khách hàng đã nhập vào, nếu hợp lệ sẽ cho phép đăng nhập.

**b. Đăng xuất**

* Mục đích: Cho phép khách hàng đăng xuất khỏi hệ thống để đảm bảo an toàn khi không sử dụng nữa.
* Mô tả: Khách hàng bấm vào nút đăng xuất thì hệ thống sẽ xóa thông tin đăng nhập khách hàng trên trang web.

**2.1.3.1.4. Đổi mật khẩu**

* Mục đích: Sau khi khách hàng được thêm vào hệ thống thì mật khẩu sẽ được lưu tự động là số điện thoại của khách hàng. Vì lý do bảo mật, khách hàng nên đổi mật khẩu sau khi đăng nhập vào trang web của phòng khám.
* Mô tả: Sau khi đăng nhập thành công, khách hàng vào bấm vào phần đổi mật khẩu. Nhập mật khẩu mới khách hàng muốn đổi và xác nhận.

**2.1.3.1.5. Theo dõi sổ sức khỏe điện tử**

* Mục đích: Cho phép khách hàng xem lại thông tin tổng quát về các lần sử dụng dịch vụ của phòng khám trước đây mọi lúc mọi nơi.
* Mô tả:
  + Khách hàng truy cập trang web hoặc tải ứng dụng trên điện thoại của phòng khám.
  + Sau khi đăng nhập thành công thì khách hàng có thể vào phần sổ sức khỏe để xem thông tin tổng quát về các lần sử dụng dịch vụ tại phòng khám.

**2.1.3.1.6. Xem thông tin chi tiết của sổ sức khỏe điện tử**

* Mục đích: Xem thêm thông tin chi tiết về sổ sức khỏe.
* Mô tả: Sổ sức khỏe chỉ hiển thị những thông tin tổng quát, bấm vào nút xem chi tiết của sổ sức khỏe cần xem để hiển thị thông tin chi tiết kèm theo đó là xem hoặc tải tài liệu được đính kèm.

**2.1.3.1.7. Đặt lịch thông báo cho ngày tái khám**

* Mục đích: Gửi thông báo để nhắc khách hàng trước ngày tái khám. Đây là chức năng chỉ dành cho khách hàng tải ứng dụng trên điện thoại của phòng khám.
* Mô tả:
  + Trong giao diện của sổ sức khỏe điện tử, khách hàng có thể bấm vào “chuông thông báo” để hệ thống gửi thông báo nhắc nhở.
  + Thông báo sẽ được gửi trước 24h trước ngày tái khám. Ví dụ thời gian tái khám là 8h ngày 30/4/2025 thì điện thoại của bạn sẽ được thông báo vào lúc 8h ngày 29/4/2025.

**2.1.3.2. Hệ thống quản lý**

**2.1.3.2.1. Đăng nhập, đăng xuất**

**a. Đăng nhập**

* Mục đích: Cho phép người dùng đăng nhập vào hệ thống quản lý
* Mô tả:
  + Mỗi người dùng được cung cấp tài khoản để đăng nhập vào hệ thống.
  + Khi đăng nhập phải nhập thông tin vào 2 trường tài khoản và mật khẩu.
  + Hệ thống sẽ xác thực của thông tin và cho đăng nhập nếu hợp lệ.

**b. Đăng xuất**

* Mục đích: Cho phép người dùng đăng xuất khỏi hệ thống để đảm bảo an toàn khi không sử dụng nữa.
* Mô tả: Người dùng bấm vào nút đăng xuất thì hệ thống sẽ xóa phiên làm việc của người dùng đó.

**2.1.3.2.2. Quản lý lịch hẹn**

**Yêu cầu:** Sau khi người dùng đăng nhập vào hệ thống, chỉ những người dùng nào được cấp vai trò quản lý lịch hẹn mới được thực hiện chức năng này.

**a. Theo dõi danh sách lịch hẹn**

* Mục đích: Theo dõi nhằm phân bố lịch hẹn của khách hàng hợp lý theo từng ngày làm việc.
* Mô tả: Sau khi bấm vào phần lịch hẹn thì hệ thống sẽ hiển thị danh sách lịch hẹn được sắp xếp theo ngày.

**b. Xác nhận lịch hẹn**

* Mục đích: Cho phép người dùng thay đổi trạng thái của lịch hẹn thành đã xác nhận sau khi xác minh tính hợp lệ của lịch hẹn.
* Mô tả:
  + Sau khi khách hàng gửi lịch hẹn thì người dùng có thể xem được thông tin của lịch hẹn mới.
  + Tiến hành xác minh lịch hẹn bằng cách gọi điện cho khách hàng theo số điện thoại có trong lịch hẹn đó.
  + Nếu xác minh thành công thì bấm nút xác nhận để chuyển trạng thái còn ngược lại thì xóa lịch hẹn.

**c. Sửa lịch hẹn**

* Mục đích: Sửa thông tin lịch hẹn của khách hàng trong trường hợp cần thiết.
* Mô tả:
  + Bấm vào nút chỉnh sửa và tiến hành chỉnh sửa những thông tin chưa chính xác.
  + Hệ thống sẽ thay đổi thông tin lịch hẹn sau khi lưu thành công.

**d. Xóa lịch hẹn**

* Mục đích: Trong trường hợp khách hàng muốn hủy lịch hẹn hoặc không thể liên lạc với khách hàng để xác nhận lịch hẹn thì người dùng có thể xóa lịch hẹn.
* Mô tả: Bấm vào nút xóa ứng với lịch hẹn cần xóa. Hệ thống sẽ xóa lịch hẹn đó.

**2.1.3.2.3. Quản lý khách hàng**

**Yêu cầu:** Sau khi người dùng đăng nhập vào hệ thống, chỉ những người dùng nào được cấp vai trò quản lý khách hàng mới được thực hiện chức năng này.

**a. Thêm mới khách hàng**

* Mục đích: Lưu thông tin khách hàng vào hệ thống vào lần đầu tiên khách hàng tới phòng khám.
* Mô tả:
  + Khách hàng cung cấp những thông tin cần thiết và người dùng thêm các thông tin đó vào biểu mẫu thêm mới khách hàng.
  + Nếu một số thông tin như số điện thoại hoặc bảo hiểm y tế bị trùng thì hệ thống sẽ thông báo lỗi.
  + Khi tạo thành công thì mật khẩu sẽ trùng với số điện thoại của khách hàng.

**b. Sửa thông tin khách hàng**

* Mục đích: Sửa thông tin khách hàng khi có yêu cầu.
* Mô tả:
  + Khách hàng sẽ cung cấp những thông tin cần chỉnh sửa.
  + Nếu một số thông tin như số điện thoại hoặc bảo hiểm y tế bị trùng thì hệ thống sẽ thông báo lỗi.

**2.1.3.2.4. Quản lý sổ sức khỏe điện tử**

**Yêu cầu:** Sau khi người dùng đăng nhập vào hệ thống, chỉ những người dùng nào được cấp vai trò quản lý sổ sức khỏe điện tử mới được thực hiện chức năng này.

**a. Thêm mới sổ sức khỏe**

* Mục đích: Tạo mới sổ sức khỏe ứng cho khách hàng.
* Mô tả:
  + Khi khách hàng tới phòng khám và chỉ định các dịch vụ muốn sử dụng. Nhân viên sẽ tiến hành tạo sổ sức khỏe cho khách hàng.
  + Nhân viên thêm các dịch vụ mà khách hàng yêu cầu, hệ thống sẽ tự động tính toán chi phí.
  + Sau khi tạo xong, nhân viên sẽ đưa khách hàng tới các phòng chức năng để sử dụng dịch vụ.

**b. Chỉnh sửa sổ sức khỏe**

* Mục đích: Chỉnh sửa một số thông tin trong sổ sức khỏe.
* Mô tả:
  + Chỉnh sửa danh sách dịch vụ khách hàng đưa ra khi có yêu cầu thay đổi hoặc phát sinh thêm một số dịch vụ mới.
  + Sau khi sử dụng dịch vụ xong thì có thể thêm ngày tái khám.

**c. In hóa đơn**

* Mục đích: In hóa đơn theo các thông tin trong sổ sức khỏe.
* Mô tả: Bấm vào nút in hóa đơn tương ứng sổ sức khỏe cần in. Hệ thống sẽ lấy thông tin các dịch vụ có trong sổ sức khỏe để in hóa đơn.

**d. Xóa sổ sức khỏe**

* Mục đích: Xóa sổ sức khỏe
* Mô tả: Khi có nhu cầu cần xóa sổ sức khỏe thì bấm vào nút xóa đối với sổ sức khỏe tương ứng. Hệ thống sẽ xóa sổ sức khỏe đó.

**2.1.3.2.5. Quản lý chi tiết sổ sức khỏe điện tử**

**Yêu cầu:** Sau khi người dùng đăng nhập vào hệ thống, chỉ những người dùng nào được cấp vai trò quản lý sổ sức khỏe điện tử mới được thực hiện chức năng này.

**a. Xem danh sách chi tiết theo sổ sức khỏe điện tử**

* Mục đích: Xem danh sách chi tiết của sổ sức khỏe điện tử
* Mô tả: Sổ sức khỏe chỉ đưa ra những thông tin cơ bản của ngày khám hôm đó. Nên muốn xem thông tin chi tiết cần bấm vào sổ sức khỏe tương ứng muốn xem. Hệ thống sẽ đưa ra toàn bộ thông tin chi tiết của sổ sức khỏe đó.

**b. Thêm chi tiết sổ sức khỏe**

* Mục đích: Thêm chi tiết sổ sức khỏe cho sổ sức khỏe
* Mô tả: Khi khách hàng khám xong ở mỗi phòng chức năng, bác sĩ hoặc nhân viên có thể thêm mới chi tiết sổ sức khỏe cho khách hàng ứng với sổ sức khỏe đã được lập trước đó.

**c. Chỉnh sửa thông tin chi tiết sổ sức khỏe**

* Mục đích: Chỉnh sửa những thông tin cần thiết trong chi tiết sổ sức khỏe
* Mô tả: Khi cần chỉnh sửa các thông tin trong chi tiết sổ sức khỏe, người dùng sẽ bấm vào nút sửa của sổ sức khỏe cần chỉnh sửa, hệ thống sẽ hiển thị một biểu mẫu chứa thông tin của chi tiết đó. Người dùng tiến hành chỉnh sửa và lưu lại.

**d. Xóa chi tiết sổ sức khỏe**

* Mục đích: Dùng để xóa thông tin chi tiết của sổ sức khỏe.
* Mô tả: Khi người dùng cần xóa đi bất kỳ chi tiết nào, bấm vào nút xóa của sổ sức khỏe cần xóa. Hệ thống sẽ xóa đi chi tiết đó.

## 2.2. Thiết kế hệ thống

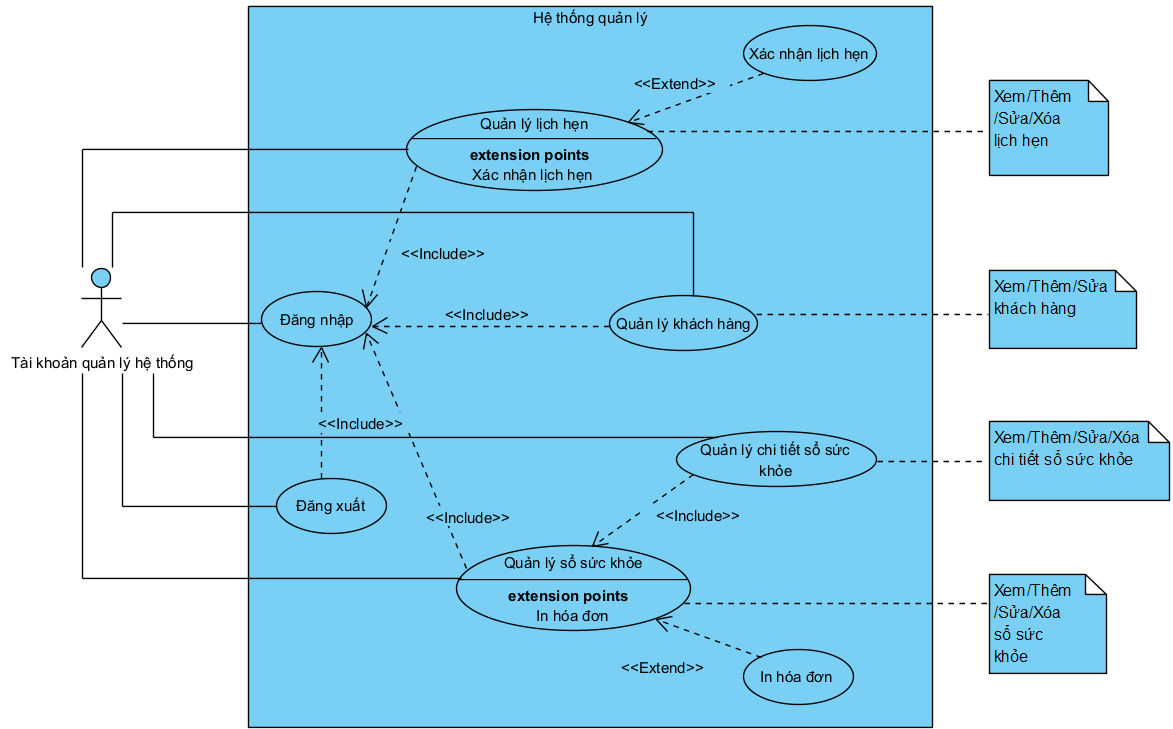
### 2.2.1. Thiết kế chức năng

**2.2.1.1. Biểu đồ Use Case**

Các biểu đồ Use Case dưới đây cung cấp cái nhìn tổng quát về các chức năng cho 3 đối tượng sử dụng hệ thống như sau:

* Tài khoản quản lý hệ thống: Dùng để quản lý, thao tác với dữ liệu trong hệ thống.
* Khách vãng lai: Dùng cho người dùng trang web hoặc ứng dụng để đặt lịch, theo dõi các thông tin của phòng khám mà không cần đăng nhập.
* Tài khoản khách hàng: Dùng cho khách hàng đã sử dụng dịch vụ của phòng khám để theo dõi sổ sức khỏe. Ngoài ra cũng có thể sử dụng những chức năng như khách vãng lai.

**2.2.1.1.1. Biểu đồ Use Case các chức năng quản lý hệ thống**

****

Hình . Biểu đồ Use Case các chức năng trong hệ thống quản lý phòng khám nha khoa

Actor Tài khoản quản lý hệ thống là tài khoản được cấp cho nhân viên, bác sĩ,… của phòng khám để thao tác với dữ liệu của hệ thống.

**2.2.1.1.1.1. Đặc tả Use Case Đăng nhập**

* Actor: Tài khoản quản lý hệ thống.
* Tóm tắt: Cho phép actor đăng nhập vào hệ thống.
* Tiền điều kiện: Actor truy cập vào hệ thống quản lý của phòng khám.
* Kịch bản chính:
  + B1: Actor nhập tên đăng nhập và mật khẩu.
  + B2: Hệ thống kiểm tra thông tin Actor vừa nhập.
  + B3: Nếu kiểm tra thành công thì sẽ cho khách hàng đăng nhập vào hệ thống, ngược lại sẽ báo lỗi.
  + B4: Kết thúc Use Case.

**2.2.1.1.1.2. Đặc tả Use Case Đăng xuất**

* Actor: Tài khoản quản lý hệ thống.
* Tóm tắt: Cho phép actor đăng xuất khỏi hệ thống.
* Tiền điều kiện: Actor đăng nhập vào hệ thống.
* Kịch bản chính:
  + B1: Actor bấm vào nút đăng xuất.
  + B2: Hệ thống xóa phiên làm việc của khách hỏi trang web.
  + B3: Kết thúc Use Case.

**2.2.1.1.1.3. Đặc tả Use Case Quản lý lịch hẹn**

* Actor: Tài khoản quản lý hệ thống.
* Tóm tắt: Cho phép actor thực hiện các thao tác với lịch hẹn như thêm, sửa, xóa.
* Tiền điều kiện: Actor đăng nhập vào hệ thống.
* Kịch bản chính:
  + B1: Actor vào phần quản lý lịch hẹn.
  + B2: Chọn và thực hiện chức năng mà Actor muốn thao tác.
  + B3: Kết thúc Use Case.

**2.2.1.1.1.4. Đặc tả Use Case Xác nhận lịch hẹn**

* Actor: Tài khoản quản lý hệ thống.
* Tóm tắt: Cho phép actor xác nhận lịch hẹn.
* Tiền điều kiện: Actor vào Quản lý lịch hẹn.
* Kịch bản chính:
  + B1: Actor xem danh sách những lịch hẹn chưa được xác nhận.
  + B2: Actor gọi điện cho số điện thoại có trên lịch hẹn để xác minh.
  + B3: Nếu lịch hẹn được xác minh thì bấm vào nút xác nhận lịch hẹn có trên lịch hẹn đó.
  + B4: Kết thúc Use Case.

**2.2.1.1.1.5. Đặc tả Use Case Quản lý khách hàng**

* Actor: Tài khoản quản lý hệ thống.
* Tóm tắt: Cho phép actor thực hiện các thao tác với khách hàng như thêm, sửa.
* Tiền điều kiện: Actor đăng nhập vào hệ thống.
* Kịch bản chính:
  + B1: Actor vào phần quản lý khách hàng.
  + B2: Chọn và thực hiện chức năng mà Actor muốn thao tác.
  + B3: Kết thúc Use Case.

**2.2.1.1.1.6. Đặc tả Use Case Quản lý sổ sức khỏe**

* Actor: Tài khoản quản lý hệ thống.
* Tóm tắt: Cho phép actor thực hiện các thao tác với sổ sức khỏe như thêm, sửa, xóa.
* Tiền điều kiện: Actor đăng nhập vào hệ thống.
* Kịch bản chính:
  + B1: Actor vào phần quản lý sổ sức khỏe.
  + B2: Chọn và thực hiện chức năng mà Actor muốn thao tác.
  + B3: Kết thúc Use Case.

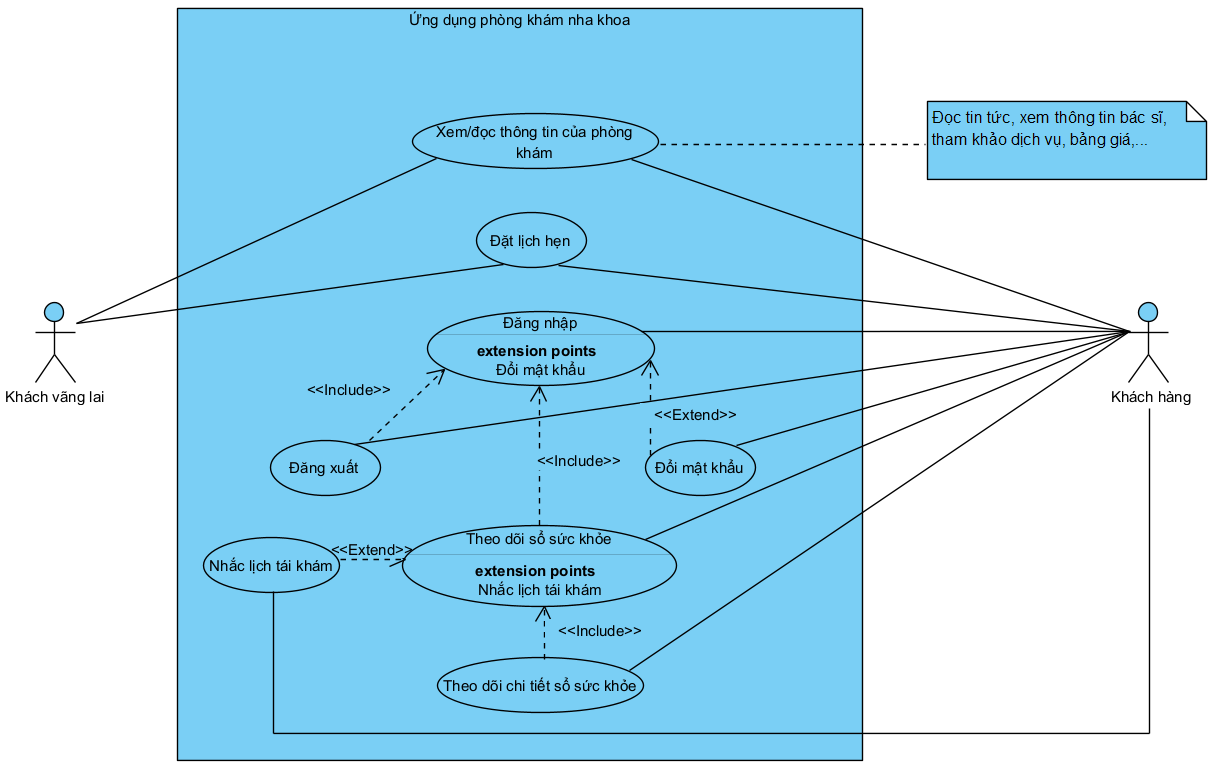
**2.2.1.1.1.7. Đặc tả Use Case In hóa đơn**

* Actor: Tài khoản quản lý hệ thống.
* Tóm tắt: Cho phép actor in hóa đơn khi khách hàng thanh toán.
* Tiền điều kiện: Actor vào phần Quản lý sổ sức khỏe.
* Kịch bản chính:
  + B1: Actor vào phần Quản lý sổ sức khỏe và bấm vào nút in hóa đơn đúng với sổ sức khỏe muốn in.
  + B2: Hệ thống sẽ hiển thị hóa đơn mẫu trước khi in.
  + B3: Actor bấm vào nút in để in hóa đơn ra giấy.
  + B4: Kết thúc Use Case.

**2.2.1.1.1.8. Đặc tả Use Case Quản lý chi tiết sổ sức khỏe**

* Actor: Tài khoản quản lý hệ thống.
* Tóm tắt: Cho phép actor thực hiện các thao tác với chi tiết sổ sức khỏe như thêm, sửa, xóa.
* Tiền điều kiện: Actor đăng nhập vào hệ thống.
* Kịch bản chính:
  + B1: Actor vào phần quản lý sổ sức khỏe.
  + B2: Actor chọn vào nút chi tiết tương ứng với sổ sức khỏe cần quản lý chi tiết.
  + B3: Hệ thống sẽ chuyển tới trang quản lý chi tiết sổ sức khỏe.
  + B4: Chọn và thực hiện chức năng mà Actor muốn thao tác trong trang chi tiết sổ sức khỏe.
  + B5: Kết thúc Use Case.

**2.2.1.1.2. Biểu đồ Use Case các chức năng của ứng dụng phòng khám nha khoa**

****

Hình . Biểu đồ Use case các chức năng trong hệ thống quản lý phòng khám nha khoa

Actor Khách vãng lai là những người sử dụng website hoặc ứng dụng điện thoại của phòng khám. Actor Khách hàng là những người đã sử dụng dịch vụ của phòng khám và được cấp tài khoản.

**2.2.1.1.2.1. Đặc tả Use Case Xem/đọc thông tin phòng khám**

* Actor: Khách vãng lai hoặc khách hàng.
* Tóm tắt: Cho phép actor đọc tin tức, xem thông tin bác sĩ, tham khảo các dịch vụ và bảng giá mà phòng khám cung cấp.
* Tiền điều kiện: Actor truy cập vào đúng trang web hoặc tải ứng dụng.
* Kịch bản chính:
  + B1: Actor truy cập vào trang web hoặc ứng dụng trên điện thoại của phòng khám.
  + B2: Actor chọn vào thông tin muốn tham khảo.
  + B3: Hệ thống sẽ hiển thị thông tin tương ứng.
  + B4: Kết thúc Use Case.

**2.2.1.1.2.2. Đặc tả Use Case Đặt lịch hẹn**

* Actor: Khách vãng lai hoặc khách hàng.
* Tóm tắt: Cho phép actor đặt lịch khám tại phòng khám.
* Tiền điều kiện: Actor truy cập vào đúng trang web hoặc vào ứng dụng.
* Kịch bản chính:
  + B1: Actor bấm vào phần đặt lịch hẹn.
  + B2: Actor điền đầy đủ thông tin vào biểu mẫu hệ thống đưa ra và gửi biểu mẫu.
  + B3: Kết thúc Use Case.

**2.2.1.1.2.3. Đặc tả Use Case Đăng nhập**

* Actor: Khách hàng.
* Tóm tắt: Cho phép actor đăng nhập vào ứng dụng.
* Tiền điều kiện: Actor đã sử dụng dịch vụ trước đó để được cấp tài khoản.
* Kịch bản chính:
  + B1: Actor điền số điện thoại, mật khẩu và bấm đăng nhập.
  + B2: Hệ thống xác nhận thông tin mà actor vừa nhập. Nếu đúng thì cho phép đăng nhập, nếu sai thì hiển thị lỗi.
  + B3: Kết thúc Use Case.

**2.2.1.1.2.4. Đặc tả Use Case Đăng xuất**

* Actor: Khách hàng.
* Tóm tắt: Cho phép actor đăng xuất khỏi ứng dụng.
* Tiền điều kiện: Actor đã đăng nhập thành công.
* Kịch bản chính:
  + B1: Actor bấm vào nút đăng xuất.
  + B2: Hệ thống xóa thông tin đăng nhập cho actor.
  + B3: Kết thúc Use Case.

**2.2.1.1.2.5. Đặc tả Use Case Đổi mật khẩu**

* Actor: Khách hàng.
* Tóm tắt: Cho phép actor đổi mật khẩu.
* Tiền điều kiện: Actor đã đăng nhập thành công.
* Kịch bản chính:
  + B1: Actor bấm vào nút đổi mật khẩu.
  + B2: Actor điền mật khẩu cũ, mật khẩu mới muốn đổi.
  + B3: Hệ thống sẽ đổi mật khẩu cho Actor.
  + B4: Kết thúc Use Case.

**2.2.1.1.2.6. Đặc tả Use Case Theo dõi sổ sức khỏe**

* Actor: Khách hàng.
* Tóm tắt: Cho phép actor theo dõi lịch sử sử dụng dịch vụ thông qua sổ sức khỏe.
* Tiền điều kiện: Actor đã đăng nhập thành công.
* Kịch bản chính:
  + B1: Actor bấm vào sổ sức khỏe.
  + B2: Ứng dụng sẽ hiển thị danh sách sổ sức khỏe ứng với tài khoản đã đăng nhập.
  + B3: Kết thúc Use Case.

**2.2.1.1.2.7. Đặc tả Use Case Theo dõi chi tiết sổ sức khỏe**

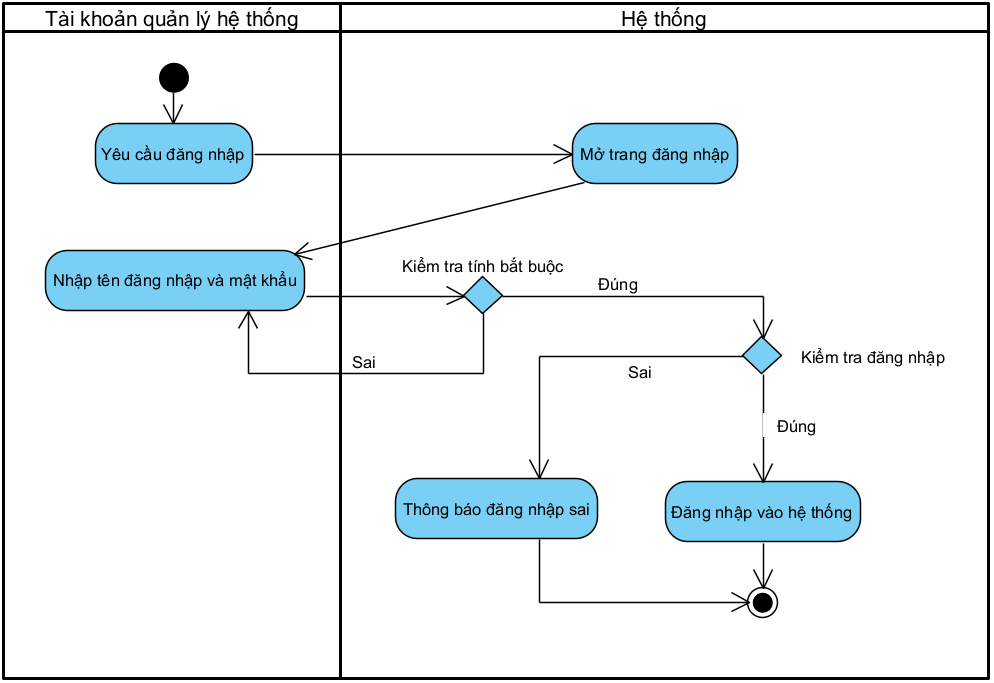
* Actor: Khách hàng.
* Tóm tắt: Cho phép actor theo dõi thông tin chi tiết của sổ sức khỏe.
* Tiền điều kiện: Actor đã đăng nhập thành công.
* Kịch bản chính:
  + B1: Actor bấm vào nút chi tiết của sổ sức khỏe cần theo dõi.
  + B2: Ứng dụng sẽ hiển thị danh sách chi tiết sổ sức khỏe ứng với sổ sức khỏe đã chọn.
  + B3: Kết thúc Use Case.

**2.2.1.1.2.8. Đặc tả Use Case Nhắc lịch tái khám**

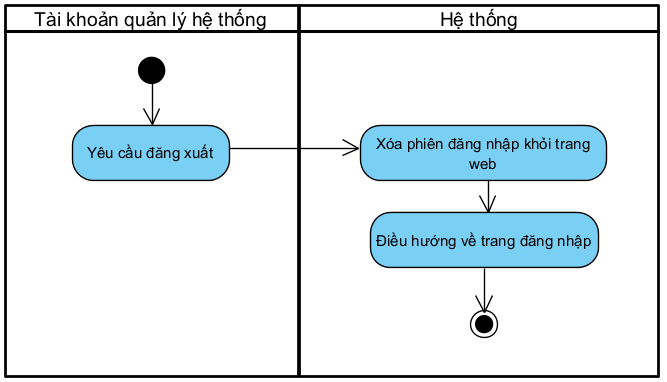
* Actor: Khách hàng.
* Tóm tắt: Cho phép actor bật thông báo cho lịch tái khám trước một ngày.
* Tiền điều kiện: Actor sử dụng ứng dụng điện thoại của phòng khám.
* Kịch bản chính:
  + B1: Actor vào danh sách sổ sức khỏe.
  + B2: Bấm vào nút chuông thông báo bên cạnh lịch tái khám muốn nhắc.
  + B3: Kết thúc Use Case.

**2.2.1.2. Biểu đồ hoạt động**

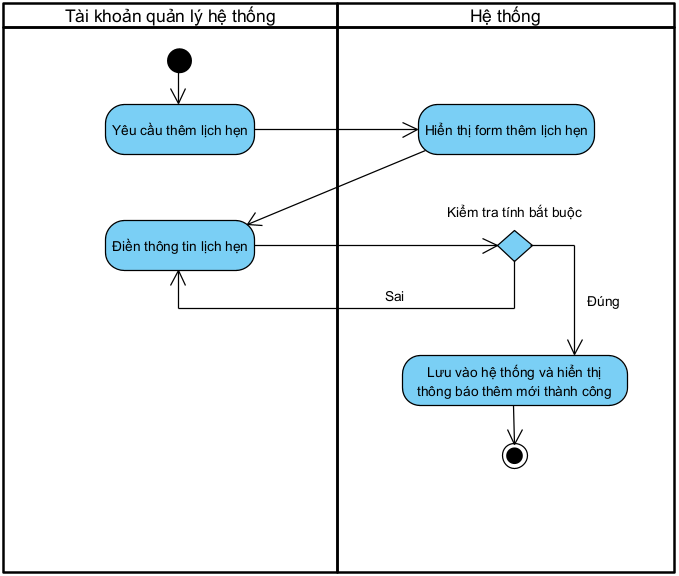
**2.2.1.2.1. Biểu đồ hoạt động cho các chức năng quản lý hệ thống**



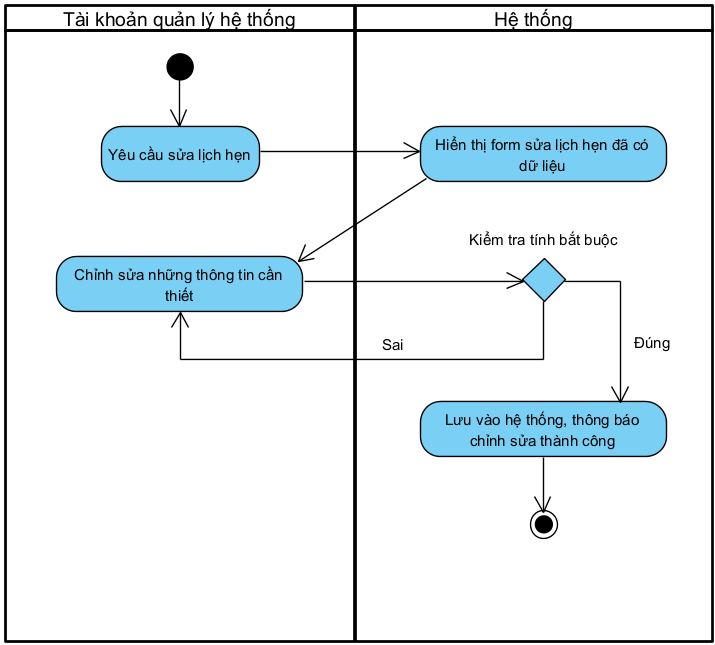
Hình . Biểu đồ hoạt động chức năng đăng nhập của tài khoản quản lý hệ thống

**

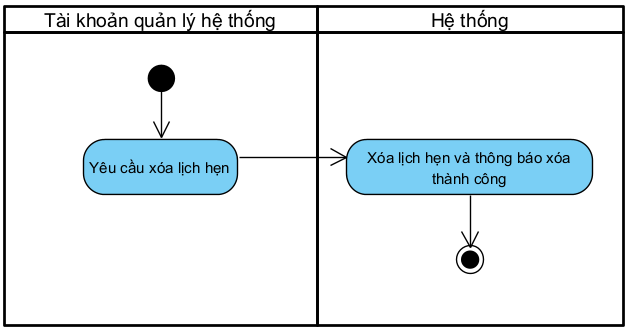
Hình . Biểu đồ hoạt động chức năng đăng xuất của tài khoản quản lý hệ thống

**

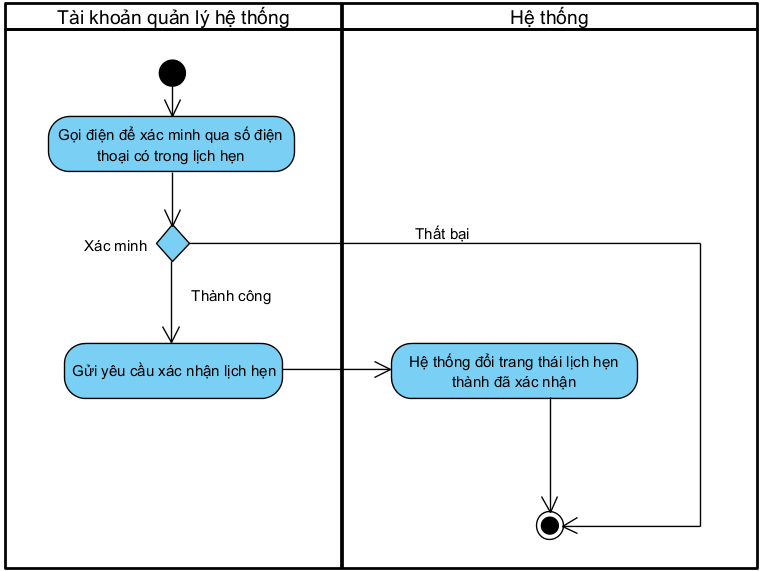
Hình . Biểu đồ hoạt động chức năng thêm lịch hẹn của tài khoản quản lý hệ thống

**

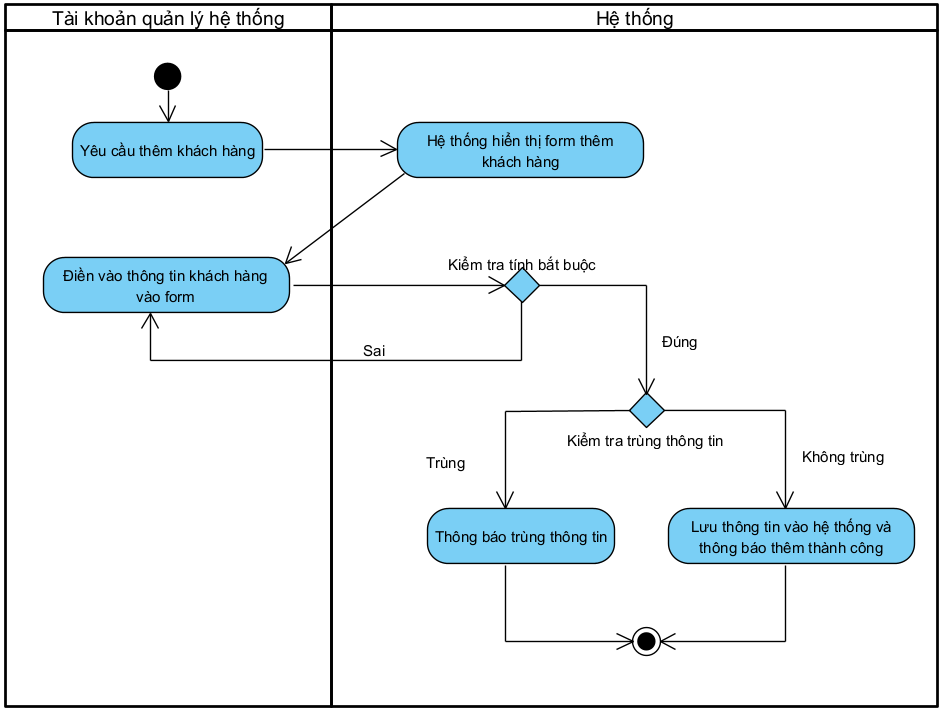
Hình . Biểu đồ hoạt động chức năng chỉnh sửa lịch hẹn của tài khoản quản lý hệ thống

**

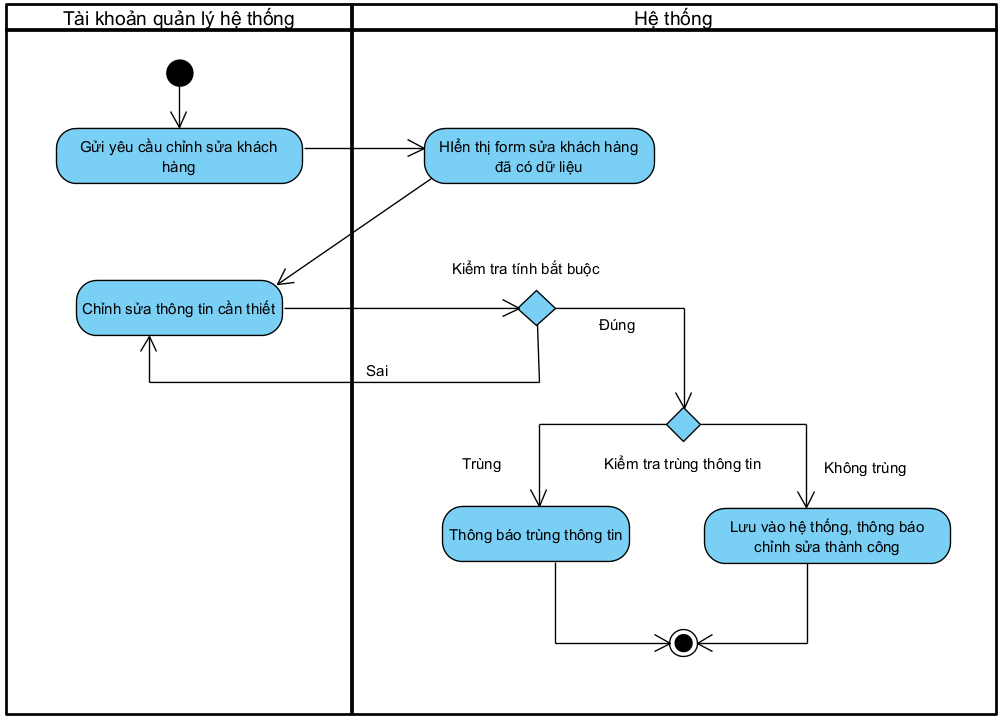
Hình . Biểu đồ hoạt động chức năng xóa lịch hẹn của tài khoản quản lý hệ thống

**

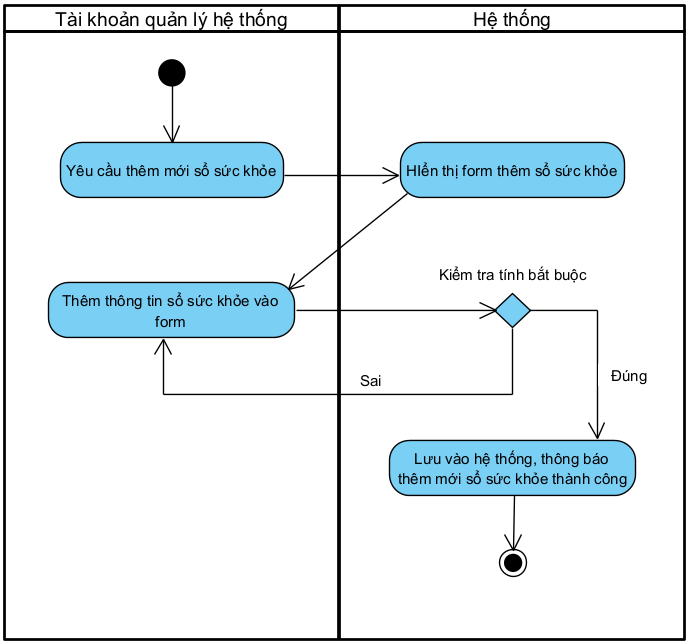
Hình . Biểu đồ hoạt động chức năng xác nhận lịch hẹn của tài khoản quản lý hệ thống

**

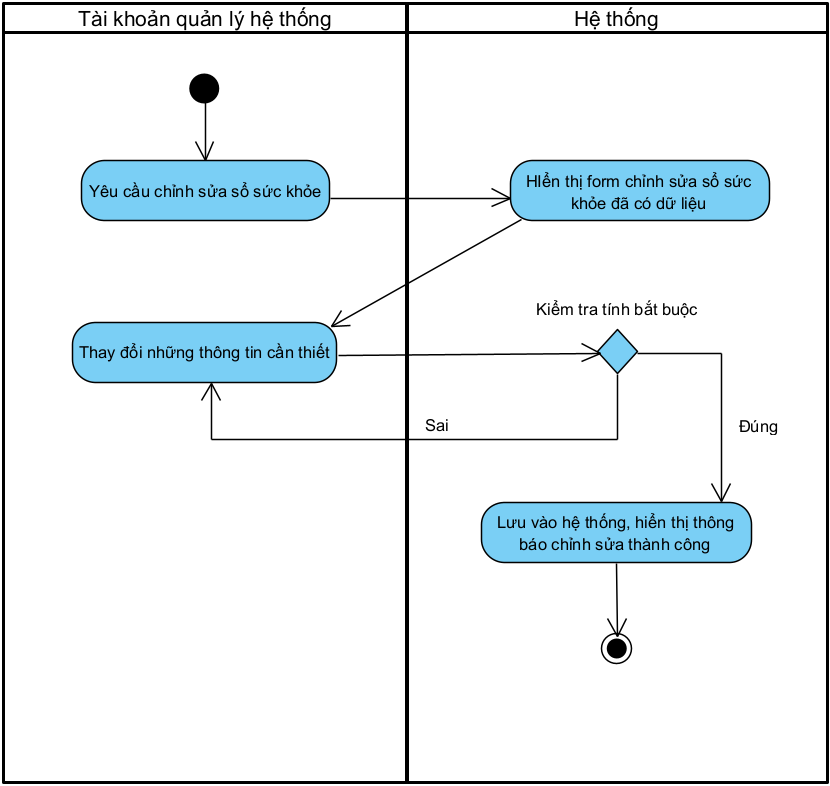
Hình . Biểu đồ hoạt động chức năng thêm khách hàng của tài khoản quản lý hệ thống

**

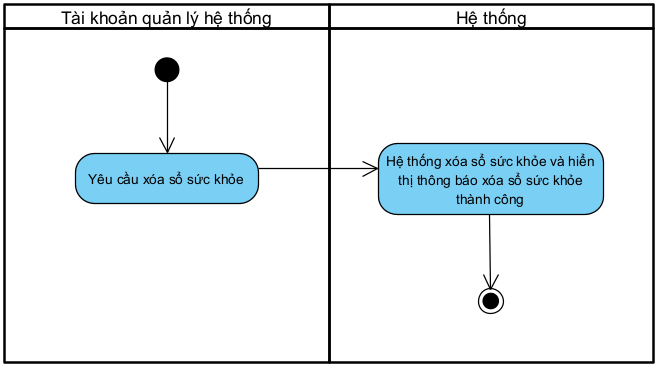
Hình . Biểu đồ hoạt động chức năng chỉnh sửa khách hàng của tài khoản quản lý hệ thống

**

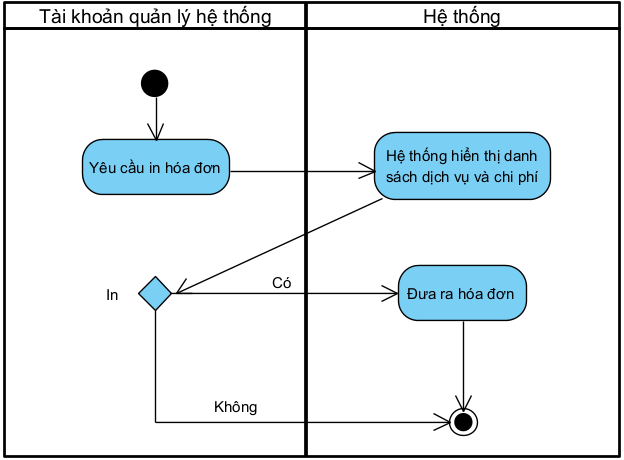
Hình . Biểu đồ hoạt động chức năng thêm sổ sức khỏe của tài khoản quản lý hệ thống

**

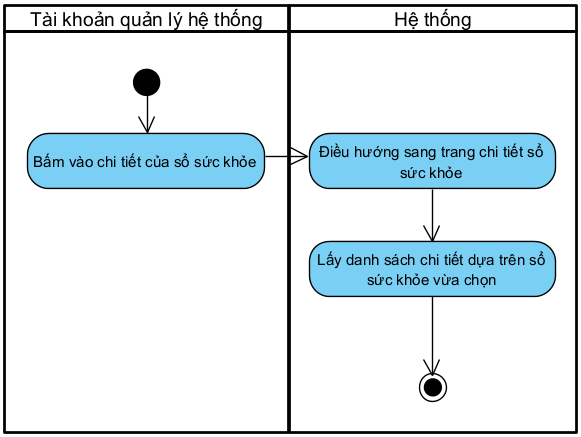
Hình . Biểu đồ hoạt động chức năng chỉnh sửa sổ sức khỏe của tài khoản quản lý hệ thống

**

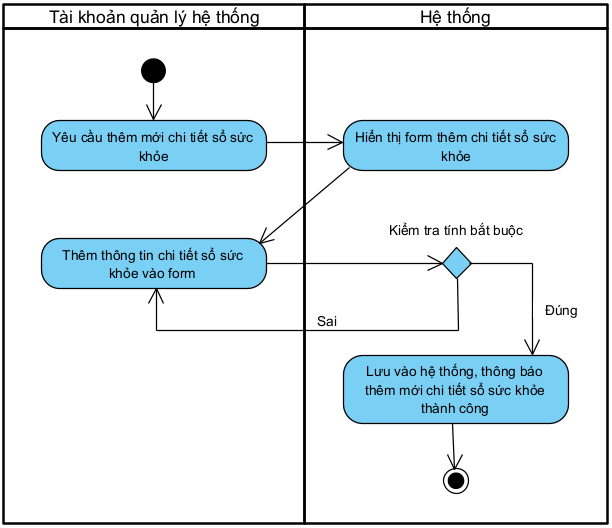
Hình . Biểu đồ hoạt động chức năng xóa sổ sức khỏe của tài khoản quản lý hệ thống



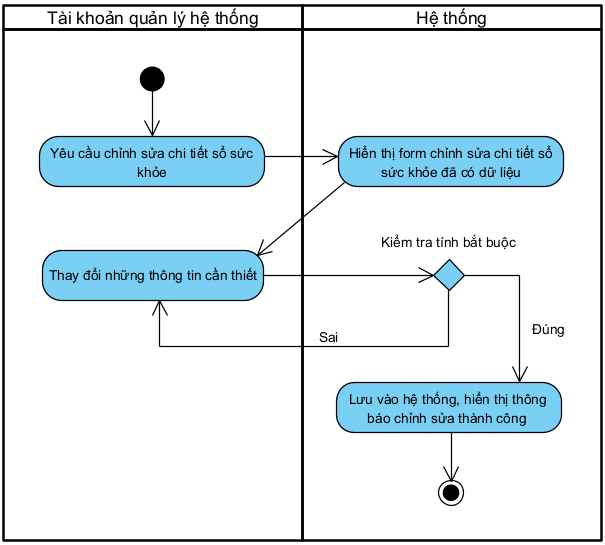
Hình . Biểu đồ hoạt động chức năng in hóa đơn của tài khoản quản lý hệ thống

**

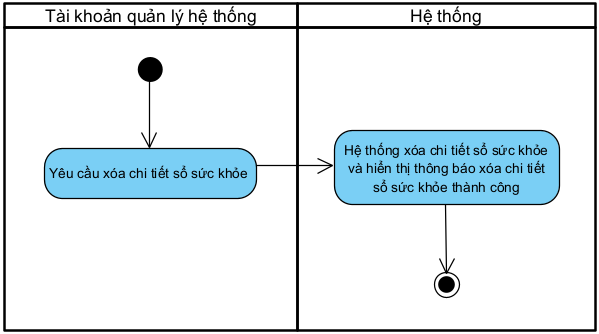
Hình . Biểu đồ hoạt động chức năng xem chi tiết sổ sức khỏe của tài khoản quản lý hệ thống

**

Hình . Biểu đồ hoạt động chức năng thêm chi tiết sổ sức khỏe của tài khoản quản lý hệ thống

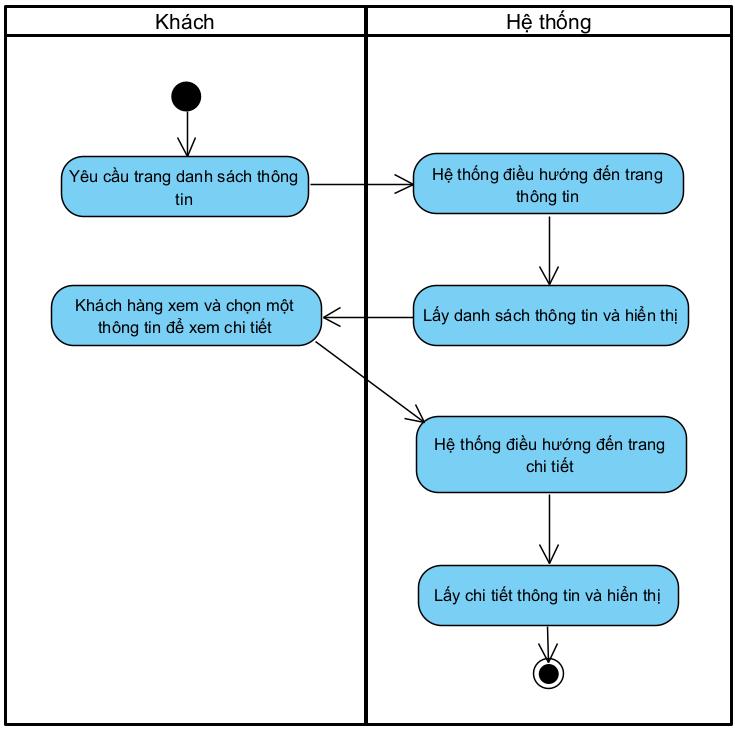
**

Hình . Biểu đồ hoạt động chức năng chỉnh sửa chi tiết sổ sức khỏe của tài khoản quản lý hệ thống

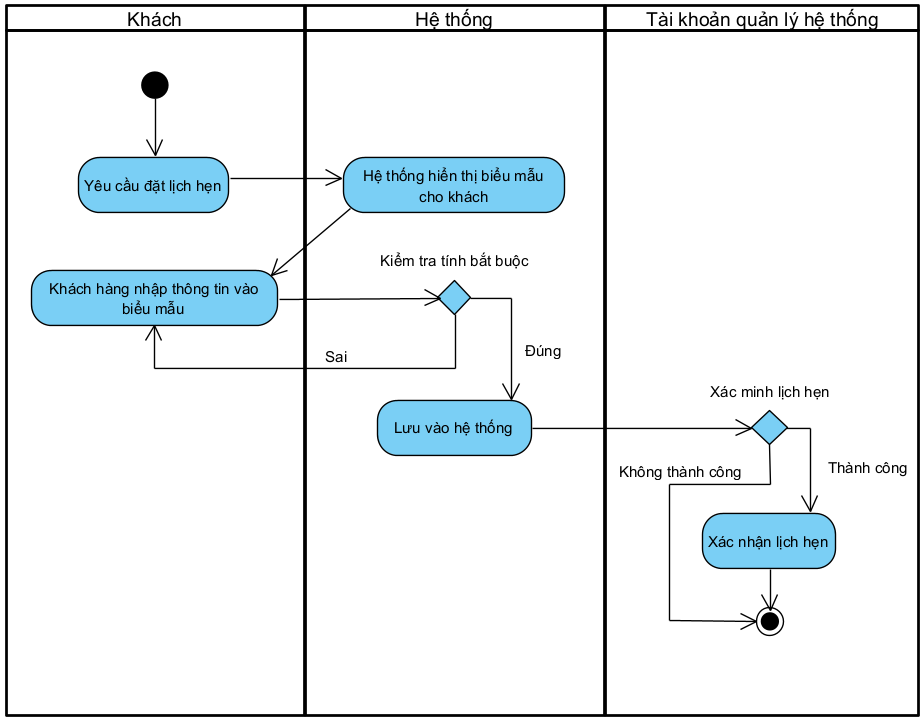
**

Hình . Biểu đồ hoạt động chức năng xóa chi tiết sổ sức khỏe của tài khoản quản lý hệ thống

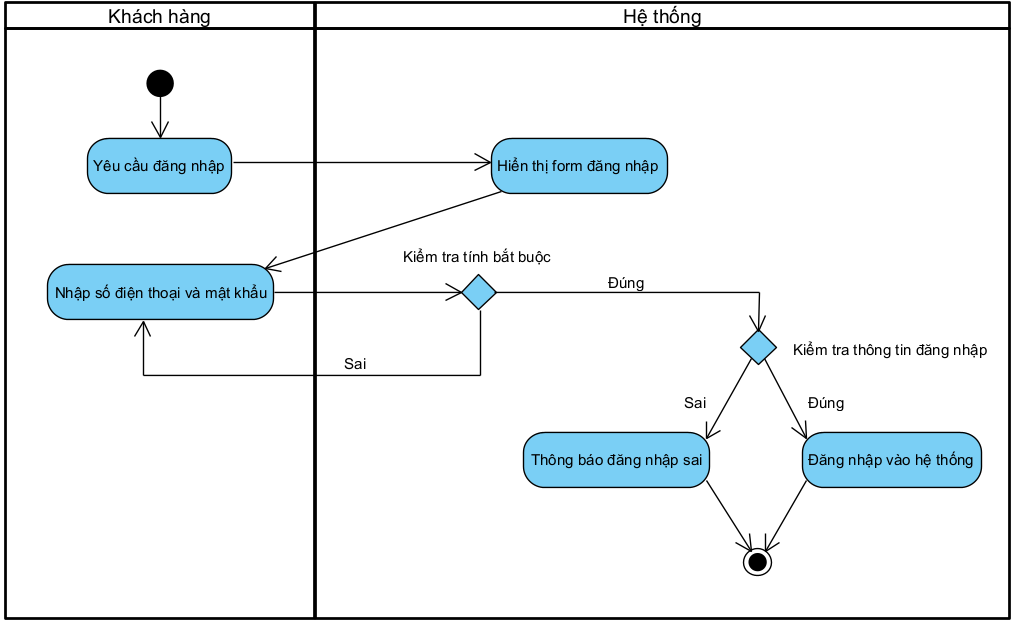
**2.2.1.2.2. Biểu đồ hoạt động cho các chức năng của khách**

**

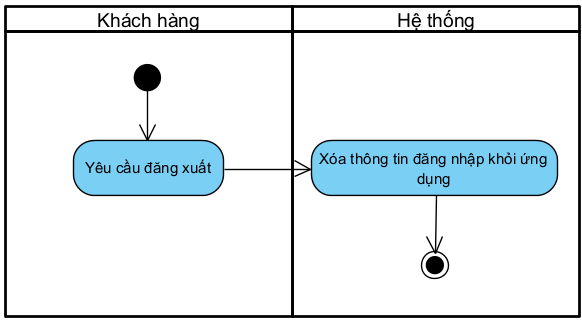
Hình . Biểu đồ hoạt động chức năng xem thông tin phòng khám của khách

**

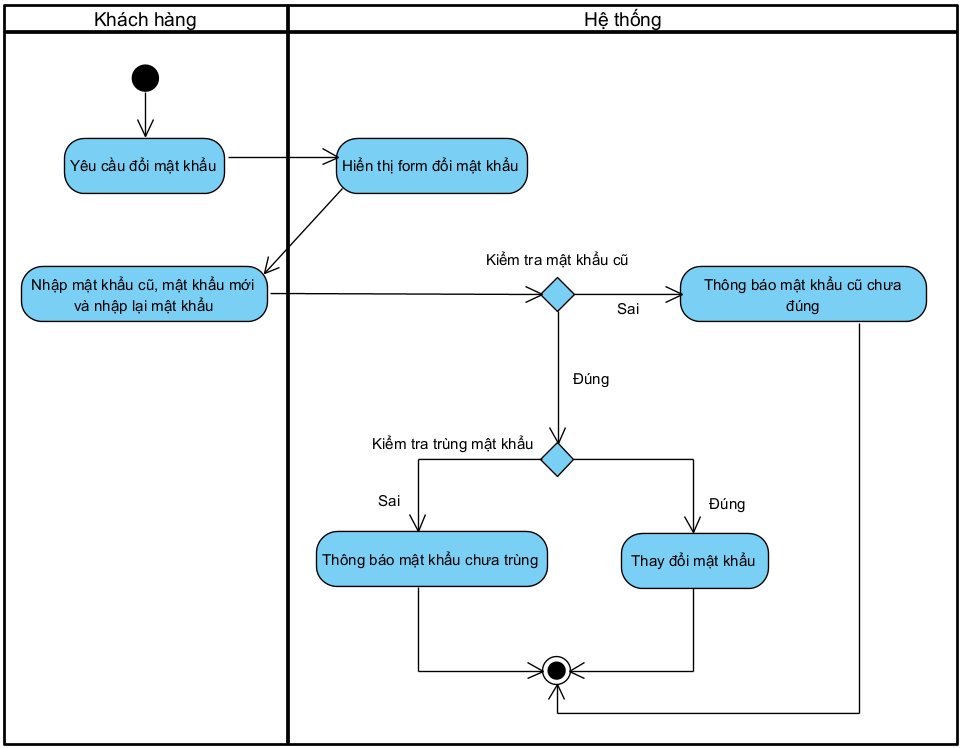
Hình . Biểu đồ hoạt động chức năng đặt lịch hẹn của khách

**

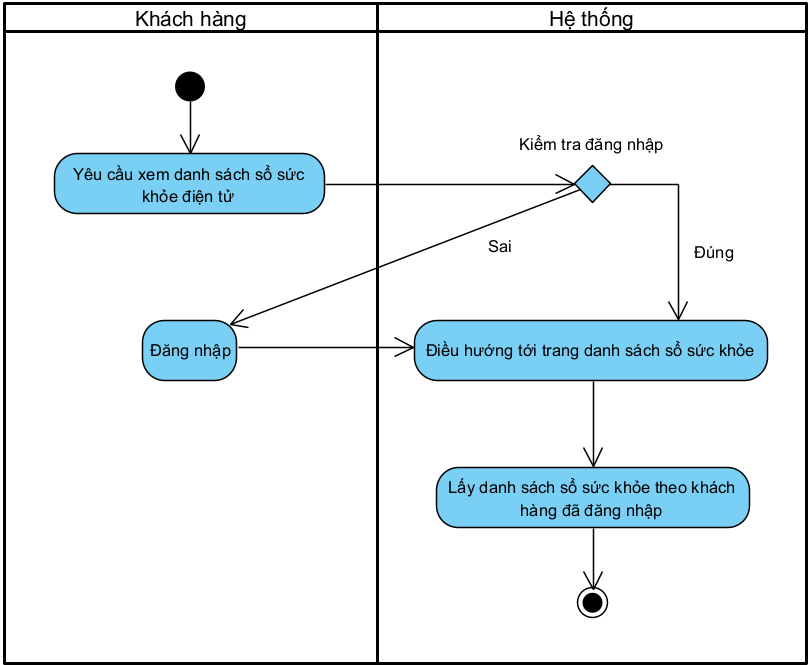
Hình . Biểu đồ hoạt động chức năng đăng nhập của khách hàng

**

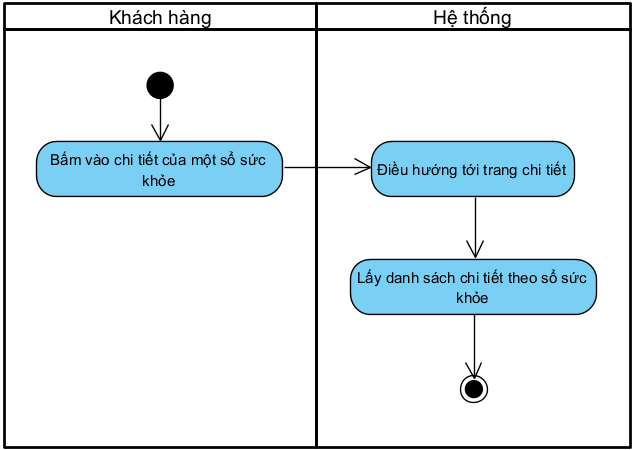
Hình . Biểu đồ hoạt động chức năng đăng xuất của khách hàng

**

Hình . Biểu đồ hoạt động chức năng đổi mật khẩu của khách hàng

**

Hình . Biểu đồ hoạt động chức năng xem danh **sách** sổ sức khỏe của khách hàng



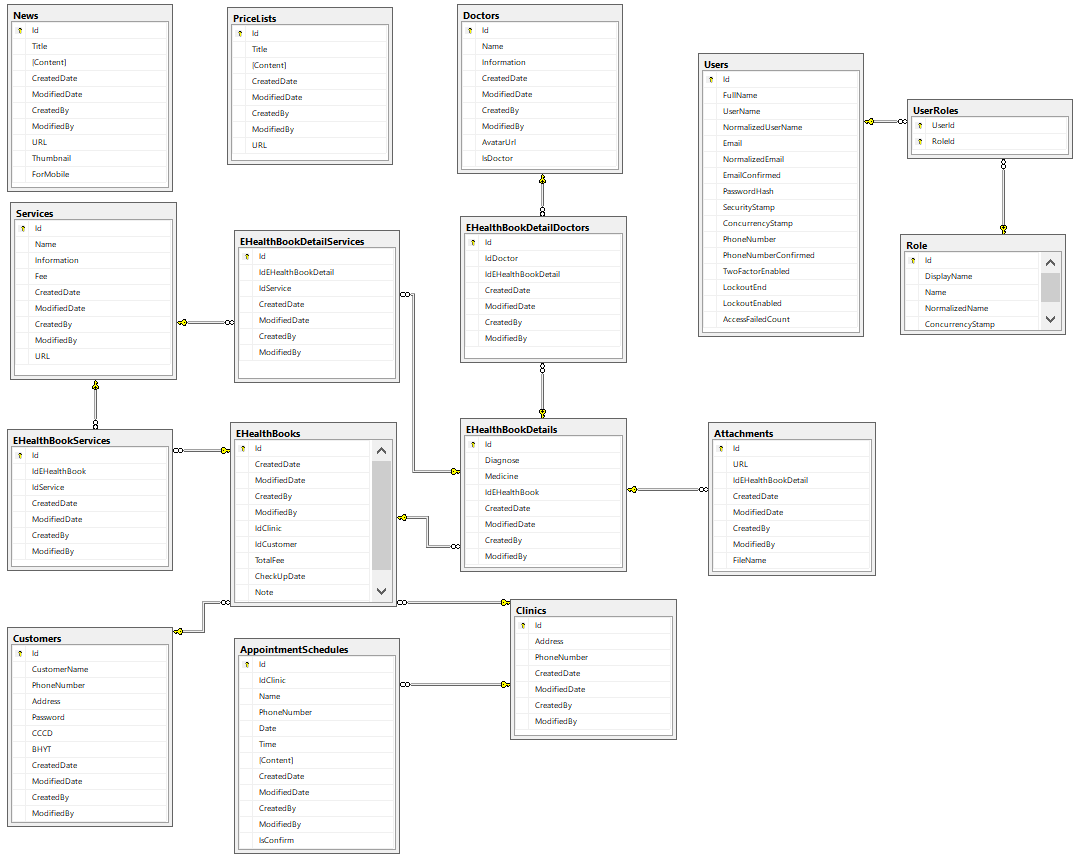
Hình . Biểu đồ hoạt động chức năng xem chi tiết sổ sức khỏe của khách hàng



Hình . Biểu đồ hoạt động chức nhắc lịch tái khám của khách hàng

### 2.2.2. Thiết kế cơ sở dữ liệu

**2.2.2.1. Thiết kế lược đồ**

****

Hình . Lược đồ cơ sở dữ liệu

**2.2.2.2. Chi tiết các bảng**

**2.2.2.2.1. News**

Mô tả: Lưu thông tin các bài viết, tin tức của phòng khám.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt các bài viết trong hệ thống |  |
| Title | nvarchar(100) | Tiêu đề bài viết |  |
| Content | nvarchar(MAX) | Nội dung bài viết |  |
| CreatedDate | datetime2(7) | Ngày tạo bài viết |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa bài viết |  |
| CreatedBy | int | Người tạo bài viết |  |
| ModifiedBy | int | Người chỉnh sửa bài viết |  |
| URL | nvarchar(MAX) | Đường dẫn bài viết | Mỗi bài viết có một đường dẫn khác nhau |
| Thumbnail | nvarchar(MAX) | Lưu ảnh bìa bài viết |  |
| ForMobile | bit | Xác định bài viết có dành cho ứng dụng điện thoại hay không | 1: Có  0: Không |

**2.2.2.2.2. PriceLists**

Mô tả: Lưu thông tin bảng giá có trong phòng khám.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt các bảng giá trong hệ thống |  |
| Title | nvarchar(MAX) | Tiêu đề bảng giá |  |
| Content | nvarchar(MAX) | Nội dung bảng giá |  |
| CreatedDate | datetime2(7) | Ngày tạo bảng giá |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa bảng giá |  |
| CreatedBy | int | Người tạo bảng giá |  |
| ModifiedBy | int | Người chỉnh sửa bảng giá |  |
| URL | nvarchar(MAX) | Đường dẫn bảng giá |  |

**2.2.2.2.3. Users**

Mô tả: Lưu thông tin tài khoản quản lý hệ thống.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt các tài khoản trong hệ thống |  |
| FullName | nvarchar(MAX) | Tên đầy đủ người dùng |  |
| UserName | nvarchar(256) | Tên đăng nhập |  |
| NormalizedUserName | nvarchar(256) | Tên đăng nhập viết thường |  |
| Email | nvarchar(256) | Email người dùng |  |
| NormalizedEmail | nvarchar(256) | Email viết thường |  |
| EmailConfirmed | bit | Lưu thông tin việc xác nhận email | 0: Chưa xác nhận  1: Đã xác nhận |
| PasswordHash | nvarchar(MAX) | Mật khẩu được mã hóa |  |
| SecurityStamp | nvarchar(MAX) | Giá trị ngẫu nhiên khi người dùng thay đổi thông tin đăng nhập |  |
| PhoneNumber | nvarchar(MAX) | Số điện thoại người dùng |  |
| PhoneNumberConfirmed | bit | Lưu thông tin số điện thoại đã được xác nhận chưa | 0: Chưa xác nhận  1: Đã xác nhận |
| TwoFactorEnabled | bit | Lưu thông tin xác thực bằng 2 yếu tố | 0: Chưa xác nhận  1: Đã xác nhận |
| LockoutEnd | datetimeoffset(7) | Thời gian kết thúc khóa tài khoản |  |
| LockoutEnabled | bit | Tài khoản bị khóa | 0: Không bị khóa  1: Bị khóa |
| AccessFailedCount | int | Số lần truy cập lỗi |  |

**2.2.2.2.4. Role**

Mô tả: Lưu thông tin các vai trò trong hệ thống.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt các vai trò trong hệ thống |  |
| DisplayName | nvarchar(MAX) | Tên hiển thị của vai trò |  |
| Name | nvarchar(256) | Tên vai trò |  |
| NormalizedName | nvarchar(256) | Tên viết thường của vai trò |  |

**2.2.2.2.5. UserRoles**

Mô tả: Lưu vai trò của tài khoản quản lý hệ thống.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| UserId | int | Lưu mã tài khoản quản lý hệ thống |  |
| RoleId | int | Lưu mã vai trò |  |

**2.2.2.2.6. Doctors**

Mô tả: Lưu thông tin bác sĩ trong phòng khám.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt các bác sĩ trong hệ thống |  |
| Name | nvarchar(50) | Họ và tên bác sĩ |  |
| Information | nvarchar(MAX) | Thông tin bác sĩ |  |
| CreatedDate | datetime2(7) | Ngày tạo bác sĩ |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin bác sĩ |  |
| CreatedBy | int | Người tạo bác sĩ |  |
| ModifiedBy | int | Người chỉnh sửa thông tin bác sĩ |  |
| AvatarUrl | nvarchar(MAX) | Đường dẫn ảnh đại diện bác sĩ |  |
| IsDoctor | bit | Phân biệt bác sĩ hay y tá | 0: Y tá  1: Bác sĩ |

**2.2.2.2.7. Services**

Mô tả: Lưu thông tin các dịch vụ phòng khám cung cấp.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt các dịch vụ trong hệ thống |  |
| Name | nvarchar(50) | Tên dịch vụ |  |
| Information | nvarchar(MAX) | Thông tin dịch vụ |  |
| Fee | real | Chi phí dịch vụ |  |
| CreatedDate | datetime2(7) | Ngày tạo dịch vụ |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin dịch vụ |  |
| CreatedBy | int | Người tạo dịch vụ |  |
| ModifiedBy | int | Người chỉnh sửa thông tin dịch vụ |  |
| URL | nvarchar(MAX) | Đường dẫn của dịch vụ |  |

**2.2.2.2.8. Clinics**

Mô tả: Lưu thông tin các phòng khám đang hoạt động.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt các phòng khám trong hệ thống |  |
| Address | nvarchar(100) | Địa chỉ phòng khám |  |
| PhoneNumber | nvarchar(15) | Số điện thoại phòng khám |  |
| CreatedDate | datetime2(7) | Ngày tạo phòng khám |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin phòng khám |  |
| CreatedBy | int | Người tạo phòng khám |  |
| ModifiedBy | int | Người chỉnh sửa thông tin phòng khám |  |

**2.2.2.2.9. AppointmentSchedules**

Mô tả: Lưu thông tin danh sách lịch hẹn của phòng khám.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt các lịch hẹn trong hệ thống |  |
| IdClinic | int | Lưu mã phòng khám mà khách hàng muốn đến |  |
| Name | nvarchar(50) | Tên khách hàng |  |
| PhoneNumber | nvarchar(15) | Số điện thoại khách hàng |  |
| Date | datetime2(7) | Ngày hẹn |  |
| Time | datetime2(7) | Giờ hẹn |  |
| Content | nvarchar(MAX) | Nội dung cuộc hẹn |  |
| CreatedDate | datetime2(7) | Ngày tạo lịch hẹn |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin lịch hẹn |  |
| CreatedBy | int | Người tạo lịch hẹn |  |
| ModifiedBy | int | Người chỉnh sửa thông tin lịch hẹn |  |
| IsConfirm | bit | Lưu thông tin đã xác nhận hay chưa | 0: Chưa xác nhận  1: Đã xác nhận |

**2.2.2.2.10. Customers**

Mô tả: Lưu thông tin khách hàng trong hệ thống.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt khách hàng trong hệ thống |  |
| CustomerName | nvarchar(MAX) | Họ tên khách hàng |  |
| PhoneNumber | nvarchar(MAX) | Số điện thoại khách hàng |  |
| Address | nvarchar(MAX) | Địa chỉ khách hàng |  |
| Password | nvarchar(MAX) | Mật khẩu |  |
| CCCD | nvarchar(MAX) | Số Chứng minh nhân dân/Căn cước công dân của khách hàng |  |
| BHYT | nvarchar(MAX) | Số bảo hiểm y tế của khách hàng |  |
| CreatedDate | datetime2(7) | Ngày tạo khách hàng |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin khách hàng |  |
| CreatedBy | int | Người tạo khách hàng |  |
| ModifiedBy | int | Người chỉnh sửa thông tin khách hàng |  |

**2.2.2.2.11. EHealthBooks**

Mô tả: Lưu thông tin sổ sức khỏe của khách hàng.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt sổ sức khỏe trong hệ thống |  |
| CreatedDate | datetime2(7) | Ngày tạo sổ sức khỏe |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin sổ sức khỏe |  |
| CreatedBy | int | Người tạo sổ sức khỏe |  |
| ModifiedBy | int | Người chỉnh sửa thông tin sổ sức khỏe |  |
| IdClinic | int | Mã phòng khám khách hàng sử dụng dịch vụ |  |
| IdCustomer | int | Mã khách hàng |  |
| TotalFee | float | Tổng chi phí dịch vụ |  |
| CheckUpDate | datetime2(7) | Ngày sử dụng dịch vụ |  |
| Note | nvarchar(MAX) | Ghi chú |  |
| ReExaminationDate | datetime2(7) | Ngày tái khám |  |

**2.2.2.2.12. EHealthBookServices**

Mô tả: Lưu thông tin các dịch vụ sử dụng trong sổ sức khỏe.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt trong hệ thống |  |
| IdEHealthBook | int | Mã sổ sức khỏe |  |
| IdService | int | Mã dịch vụ |  |
| CreatedDate | datetime2(7) | Ngày tạo |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin |  |
| CreatedBy | int | Người tạo |  |
| ModifiedBy | int | Người chỉnh sửa thông tin |  |

**2.2.2.2.13. EHealthBookDetails**

Mô tả: Lưu thông tin chi tiết của sổ sức khỏe trong hệ thống.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã chi tiết sổ sức khỏe phân biệt trong hệ thống |  |
| Diagnose | nvarchar(MAX) | Chẩn đoán |  |
| Medicine | nvarchar(MAX) | Thuốc |  |
| IdEHealthBook | int | Mã sổ sức khỏe |  |
| CreatedDate | datetime2(7) | Ngày tạo chi tiết sổ sức khỏe |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin chi tiết sổ sức khỏe |  |
| CreatedBy | int | Người tạo chi tiết sổ sức khỏe |  |
| ModifiedBy | int | Người chỉnh sửa thông tin chi tiết sổ sức khỏe |  |

**2.2.2.2.14. Attachments**

Mô tả: Lưu thông tin tài liệu của chi tiết sổ sức khỏe.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã tài liệu phân biệt trong hệ thống |  |
| URL | nvarchar(MAX) | Đường dẫn tài liệu |  |
| IdEHealthBookDetail | int | Mã chi tiết sổ sức khỏe |  |
| CreatedDate | datetime2(7) | Ngày tạo tài liệu |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa tài liệu |  |
| CreatedBy | int | Người tạo tài liệu |  |
| ModifiedBy | int | Người chỉnh sửa thông tin tài liệu |  |
| FileName | nvarchar(MAX) | Tên tài liệu |  |

**2.2.2.2.15. EhealthBookDetailDoctors**

Mô tả: Lưu thông tin danh sách bác sĩ trong chi tiết sổ sức khỏe.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt trong hệ thống |  |
| IdDoctor | int | Mã bác sĩ |  |
| IdEHealthBookDetail | int | Mã chi tiết sổ sức khỏe |  |
| CreatedDate | datetime2(7) | Ngày tạo |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin |  |
| CreatedBy | int | Người tạo |  |
| ModifiedBy | int | Người chỉnh sửa thông tin |  |

**2.2.2.2.16. EHealthBookDetailServices**

Mô tả: Lưu thông tin danh sách dịch vụ trong chi tiết sổ sức khỏe.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** | **Chú thích** |
| Id | int | Lưu mã phân biệt trong hệ thống |  |
| IdService | int | Mã dịch vụ |  |
| IdEHealthBookDetail | int | Mã chi tiết sổ sức khỏe |  |
| CreatedDate | datetime2(7) | Ngày tạo |  |
| ModifiedDate | datetime2(7) | Ngày chỉnh sửa thông tin |  |
| CreatedBy | int | Người tạo |  |
| ModifiedBy | int | Người chỉnh sửa thông tin |  |

# CÀI ĐẶT HỆ THỐNG

## 3.1. Mô hình tổ chức

### 3.1.1. Mô hình thiết kế

**Tổng quan về mô hình Client – Server**

Client Server là mô hình mạng máy tính gồm có 2 thành phần chính đó là máy khách (client) và máy chủ (server). Server chính là nơi giúp lưu trữ tài nguyên cũng như cài đặt các chương trình dịch vụ theo đúng như yêu cầu của client. Ngược lại, Client bao gồm máy tính cũng như các loại thiết bị điện tử nói chung sẽ tiến hành gửi yêu cầu đến server.

**Nguyên tắc hoạt động**

Trong mô hình Client Server, server chấp nhận tất cả các yêu cầu hợp lệ từ mọi nơi khác nhau trên Internet, sau đó trả kết quả về máy tính đã gửi yêu cầu đó

Máy tính được coi là máy khách khi chúng làm nhiệm vụ gửi yêu cầu đến các máy chủ và đợi câu trả lời được gửi về.

Để máy khách và máy chủ có thể giao tiếp được với nhau thì giữa chúng phải có một chuẩn nhất định, và chuẩn đó được gọi là giao thức. Một số giao thức được sử dụng phổ biến hiện nay như: HTTPS, TCP/IP, FTP,...

Nếu máy khách muốn lấy được thông tin từ máy chủ, chúng phải tuân theo một giao thức mà máy chủ đó đưa ra. Nếu yêu cầu đó được chấp nhận thì máy chủ sẽ thu thập thông tin và trả về kết quả cho máy khách yêu cầu. Bởi vì Server - máy chủ luôn luôn trong trạng thái sẵn sàng để nhận request từ client nên chỉ cần client gửi yêu cầu tín hiệu và chấp nhận yêu cầu đó thì server sẽ trả kết quả về phía client trong thời gian ngắn nhất.

**Ưu điểm**

* Giúp chúng ta có thể làm việc trên bất kì một máy tính nào có hỗ trợ giao thức truyền thông. Giao thức chuẩn này cũng giúp các nhà sản xuất tích hợp lên nhiều sản phẩm khác nhau mà không gặp phải khó khăn gì.
* Có thể có nhiều server cùng làm một dịch vụ, chúng có thể nằm trên nhiều máy tính hoặc một máy tính.
* Chỉ mang đặc điểm của phần mềm mà không hề liên quan đến phần cứng, ngoài yêu cầu duy nhất là server phải có cấu hình cao hơn các client.
* Hỗ trợ người dùng nhiều dịch vụ đa dạng và sự tiện dụng bởi khả năng truy cập từ xa.
* Cung cấp một nền tảng lý tưởng, cho phép cung cấp tích hợp các kỹ thuật hiện đại như mô hình thiết kế hướng đối tượng, hệ chuyên gia, hệ thông tin địa lý (GIS).

**Nhược điểm**

* Trong trường hợp có quá nhiều Client tạo request từ cùng một Server thì nó có thể sẽ làm cho kết nối chậm hơn. Trong trường hợp xấu nhất còn có thể xuất hiện hiện tượng crash.
* Client Server là mạng tập trung chính vì thế, khi Server chính xảy ra sự cố hoặc bị nhiễu thì cũng đồng nghĩa với việc toàn bộ hệ thống mạng sẽ bị gián đoạn.
* Chi phí được sử dụng để thiết lập và bảo trì Server trong Client Server thường sẽ khá cao.

### 3.1.2. Hệ quản trị cơ sở dữ liệu

SQL (Structured Query Language) được thiết kế để quản lý dữ liệu trong một hệ thống quản lý cơ sở dữ liệu quan hệ, cho phép thực hiện đa dạng các chức năng, tiêu biểu là:

* Mô tả, truy cập dữ liệu trong các hệ thống quản lý cơ sở dữ liệu quan hệ.
* Chèn, xóa và cập nhật các hàng trong 1 quan hệ.
* Cho phép tạo các cơ sở dữ liệu và bảng, chế độ view, thủ tục lưu trữ, chức năng trong cơ sở dữ liệu.
* Thiết lập quyền trên các bảng, thủ tục và view.

Microsoft SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ sử dụng câu lệnh SQL để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. SQL Server được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn lên đến Terabyte và có thể phục vụ cùng lúc cho hàng ngàn người sử dụng.

## 3.2. Kết quả cài đặt

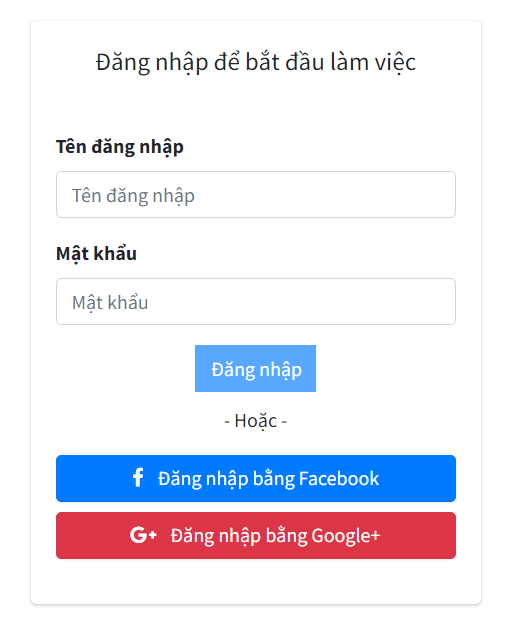
Sau quá trình phân tích và thiết kế hệ thống, cùng với việc áp dụng các mô hình và các công nghệ phù hợp, ứng dụng đa nền tảng cho phòng khám nha khoa đã được hoàn thành với các chức năng dành cho 3 đối tượng sử dụng của hệ thống như sau:

* Quản trị hệ thống: Chức năng chính là thao tác lịch hẹn, khách hàng, sổ sức khỏe thông qua trang Admin.
* Khách vãng lai: Xem thông tin của phòng khám, đọc tin tức, đặt lịch hẹn.
* Khách hàng: Ngoài những chức năng giống khách vãng lai, khách hàng có thể theo dõi sổ sức khỏe trong ứng dụng của phòng khám.

Các chức năng đã được cài đặt trong hệ thống được mô tả chi tiết dưới đây.

### 3.2.1. Các chức năng của quản trị hệ thống

**3.2.1.1. Đăng nhập**

****

Hình . Giao diện đăng nhập

Truy cập: Người có tài khoản quản lý hệ thống.

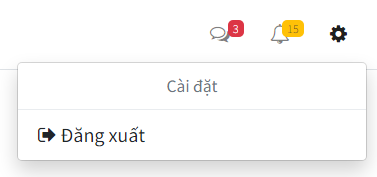
Tên hành động: Đăng nhập.

Mô tả: Sau khi nhập tên đăng nhập và mật khẩu thì ấn nút đăng nhập, hệ thống sẽ kiểm tra giá trị vừa nhập vào có khớp với cơ sở dữ liệu hay không.

Thành công: Cho phép đăng nhập và điều hướng tới trang chính.

Thất bại: Nếu kiểm tra thông tin đăng nhập thất bại thì sẽ thông báo “Sai tên đăng nhập hoặc mật khẩu”.

**3.2.1.2. Đăng xuất**



Hình . Giao diện cho chức năng đăng xuất

Truy cập: Người có tài khoản quản lý hệ thống.

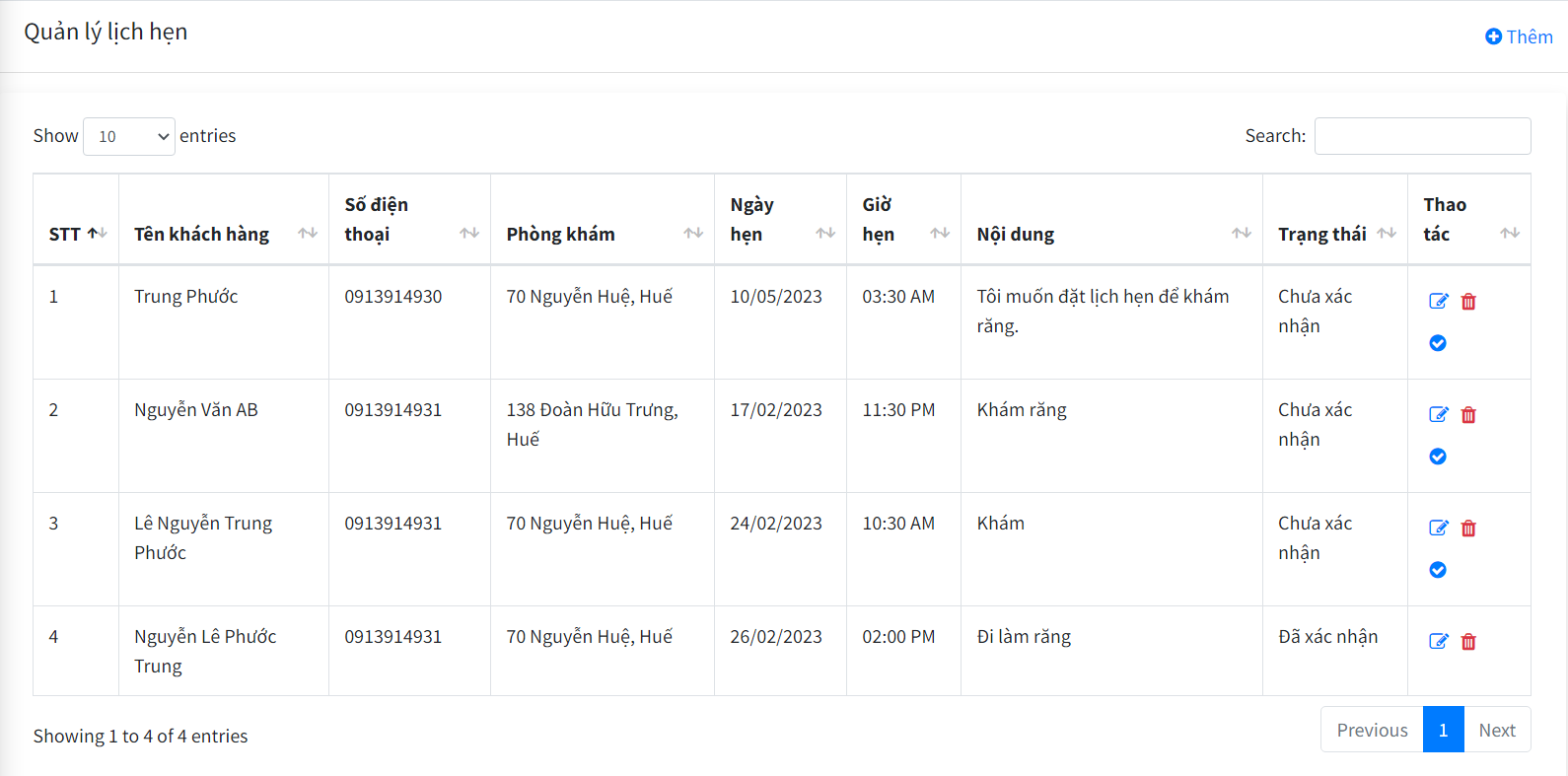
Tên hành động: Đăng xuất.

Mô tả: Sau khi người dùng bấm đăng xuất, hệ thống sẽ xóa phiên làm việc của người dùng đó.

Thành công: Xóa thông tin đăng nhập và điều hướng về trang đăng nhập.

**3.2.1.3. Quản lý lịch hẹn**

**3.2.1.3.1. Xem danh sách lịch hẹn**

****

Hình . Giao diện danh sách lịch hẹn

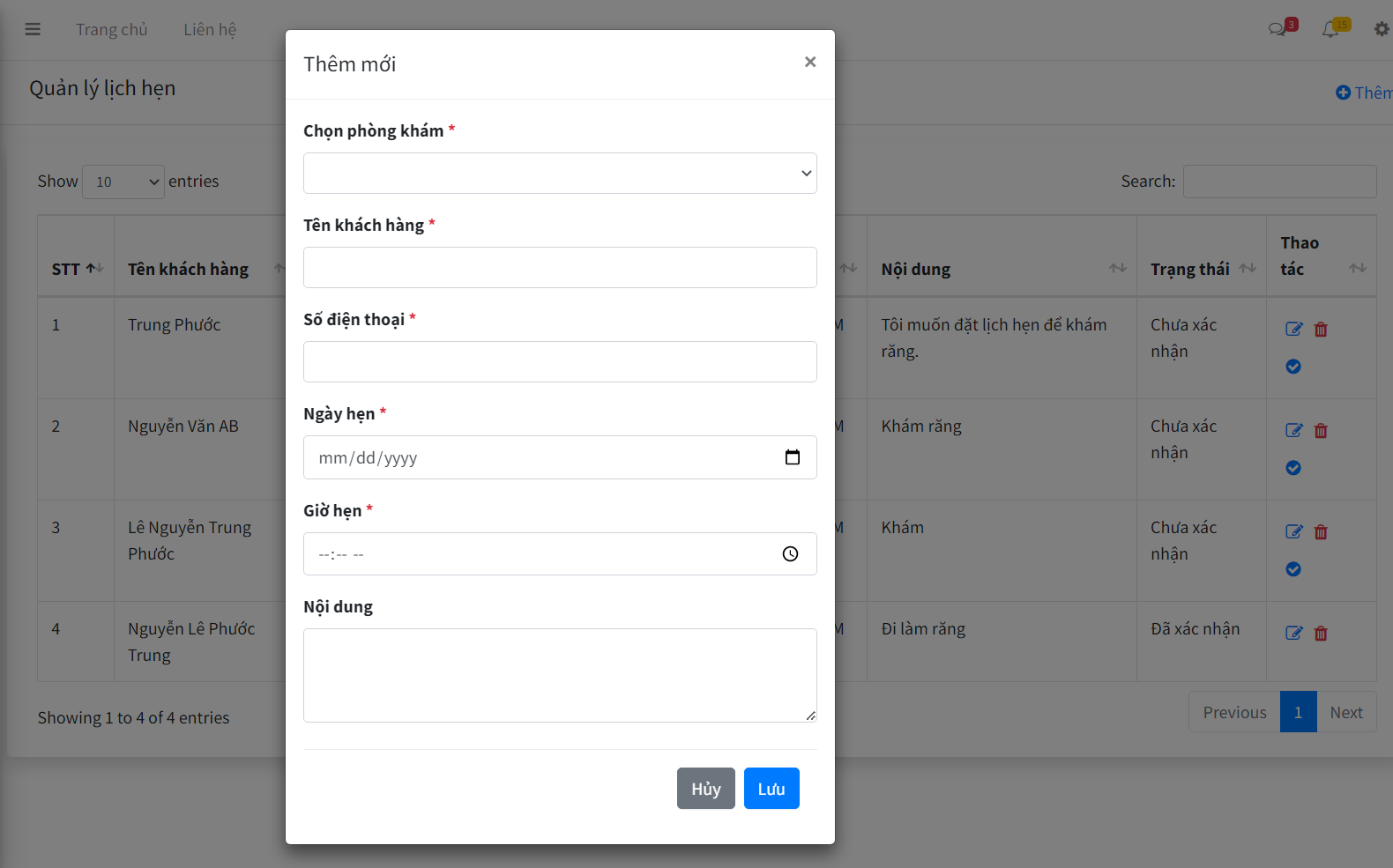
Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Xem danh sách lịch hẹn.

Mô tả: Sau khi người dùng bấm vào phần lịch hẹn, hệ thống sẽ hiển thị giao diện danh sách lịch hẹn cho người dùng.

Thành công: Hệ thống điều hướng đến trang lịch hẹn, lấy toàn bộ danh sách và phân trang, sau đó trả về danh sách như hình trên.

**3.2.1.3.2. Thêm lịch hẹn**

**

Hình . Giao diện thêm lịch hẹn

Truy cập: Người có tài khoản quản lý hệ thống.

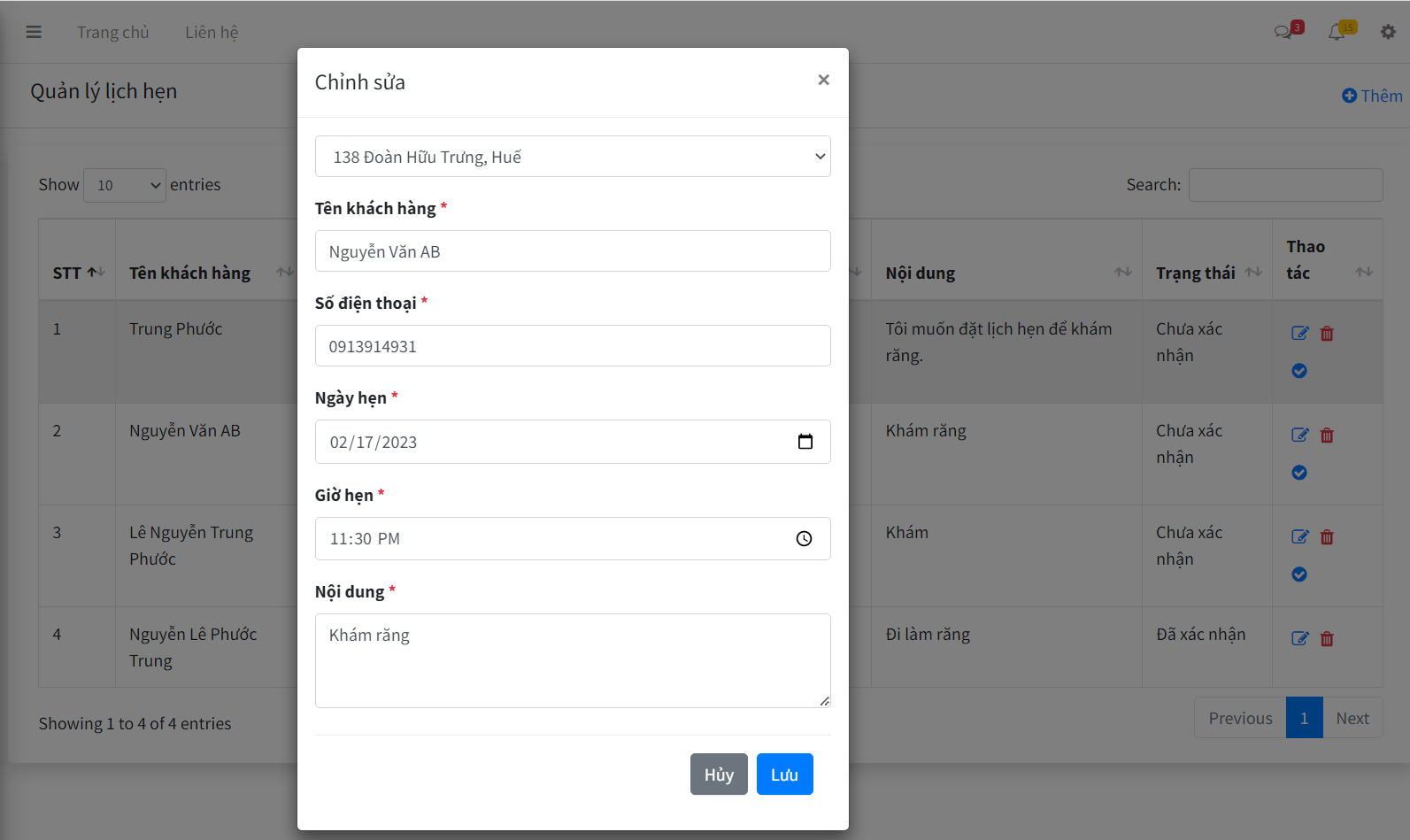
Tên hành động: Thêm lịch hẹn.

Mô tả: Sau khi người dùng bấm vào nút thêm ở góc phải trên màn hình, hệ thống sẽ hiển thị giao diện thêm mới cho người dùng. Nhập đầy đủ thông tin và bấm lưu để thêm.

Thành công: Hệ thống sẽ lưu lại lịch hẹn và load lại danh sách lịch hẹn.

Thất bại: Nếu điền thiếu thông tin, khi bấm lưu thì hệ thống sẽ nhắc nhở điền đầy đủ thông tin cần thiết.

**3.2.1.3.3. Sửa lịch hẹn**

**

Hình . Giao diện sửa lịch hẹn

Truy cập: Người có tài khoản quản lý hệ thống.

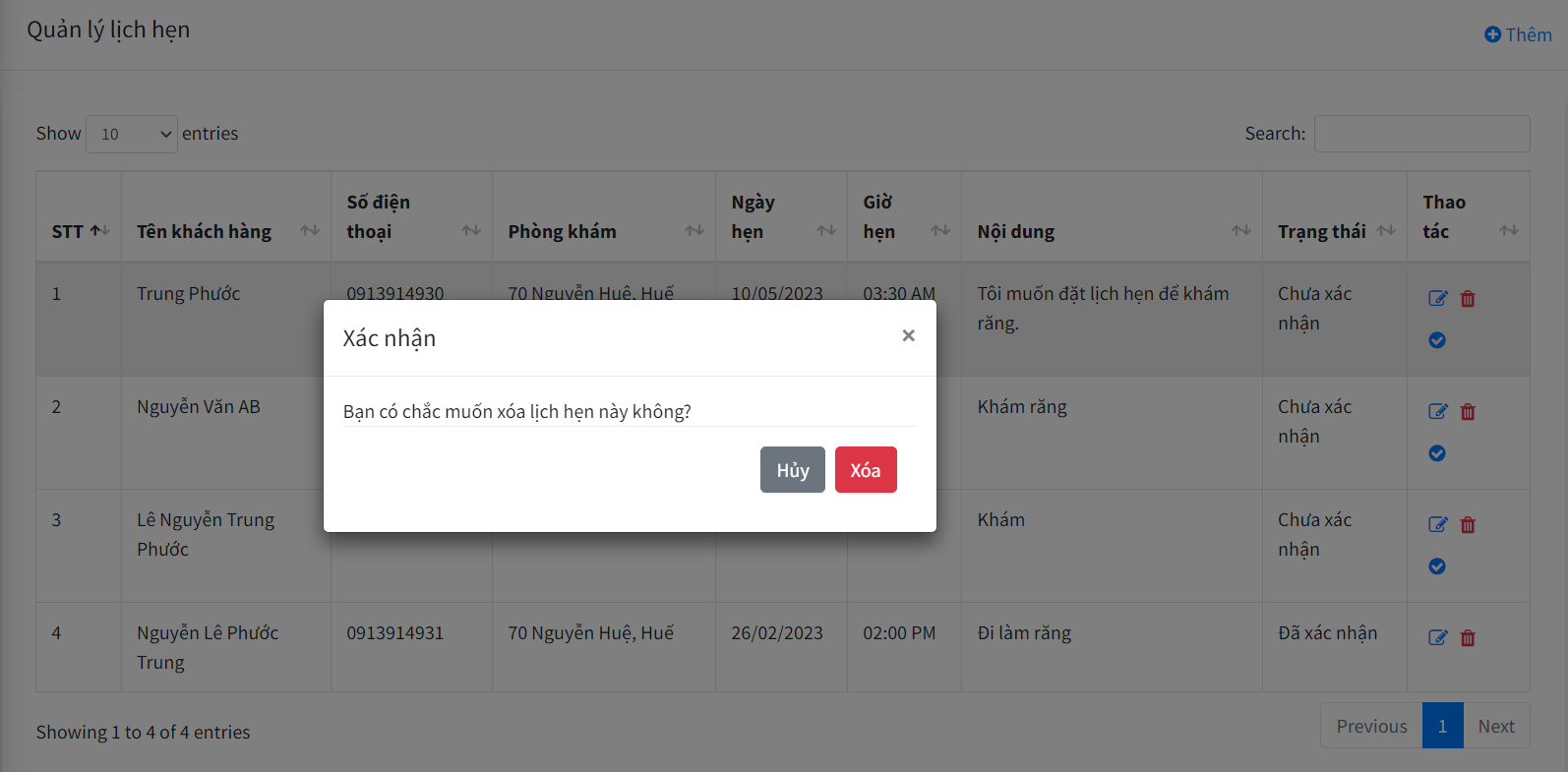
Tên hành động: Sửa lịch hẹn.

Mô tả: Người dùng bấm nút sửa tương ứng với lịch hẹn muốn chỉnh sửa, hệ thống sẽ hiển thị giao diện sửa lịch hẹn. Người dùng chỉnh sửa những thông tin cần thiết và bấm lưu để hoàn tất.

Thành công: Hệ thống sẽ lưu lại thông tin chỉnh sửa và load lại danh sách lịch hẹn.

Thất bại: Nếu điền thiếu thông tin, khi bấm lưu thì hệ thống sẽ nhắc nhở điền đầy đủ thông tin cần thiết.

**3.2.1.3.4. Xóa lịch hẹn**



Hình . Giao diện xóa lịch hẹn

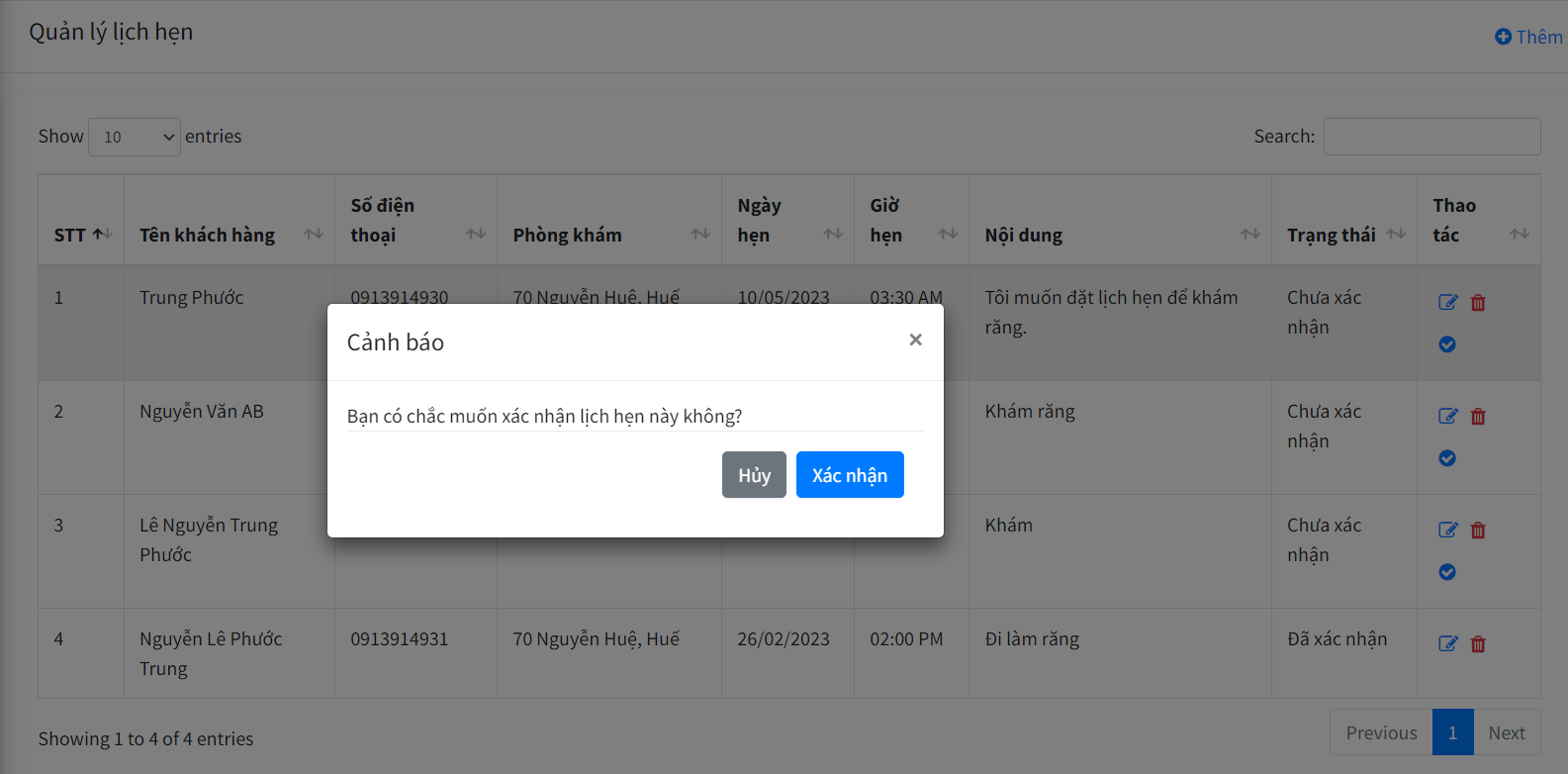
Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Xóa lịch hẹn.

Mô tả: Người dùng bấm nút xóa tương ứng với lịch hẹn muốn xóa, hệ thống sẽ hỏi lại trước khi xóa. Bấm nút xóa để xác nhận muốn xóa lịch hẹn đó.

Thành công: Hệ thống sẽ lưu lại xóa lịch hẹn và load lại danh sách lịch hẹn.

**3.2.1.3.5. Xác nhận lịch hẹn**



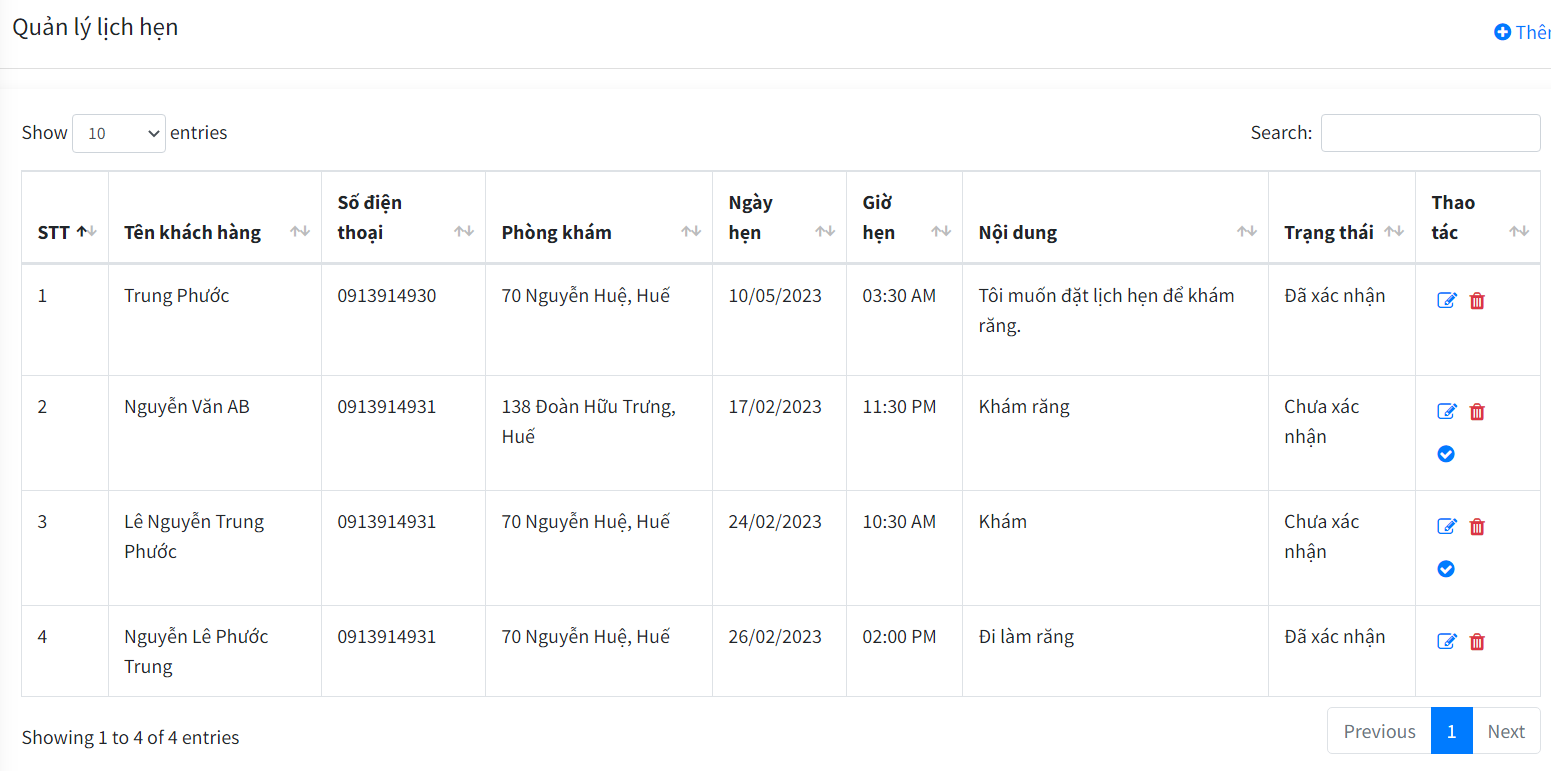
Hình . Giao diện xác nhận lịch hẹn

Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Xóa lịch hẹn.

Mô tả: Khi người dùng đã xác minh lịch hẹn với khách hàng. Bấm nút xác nhận tương ứng với lịch hẹn muốn đổi trạng thái, hệ thống sẽ hỏi lại trước khi xác nhận. Bấm nút xác nhận để thay đổi trạng thái lịch hẹn.

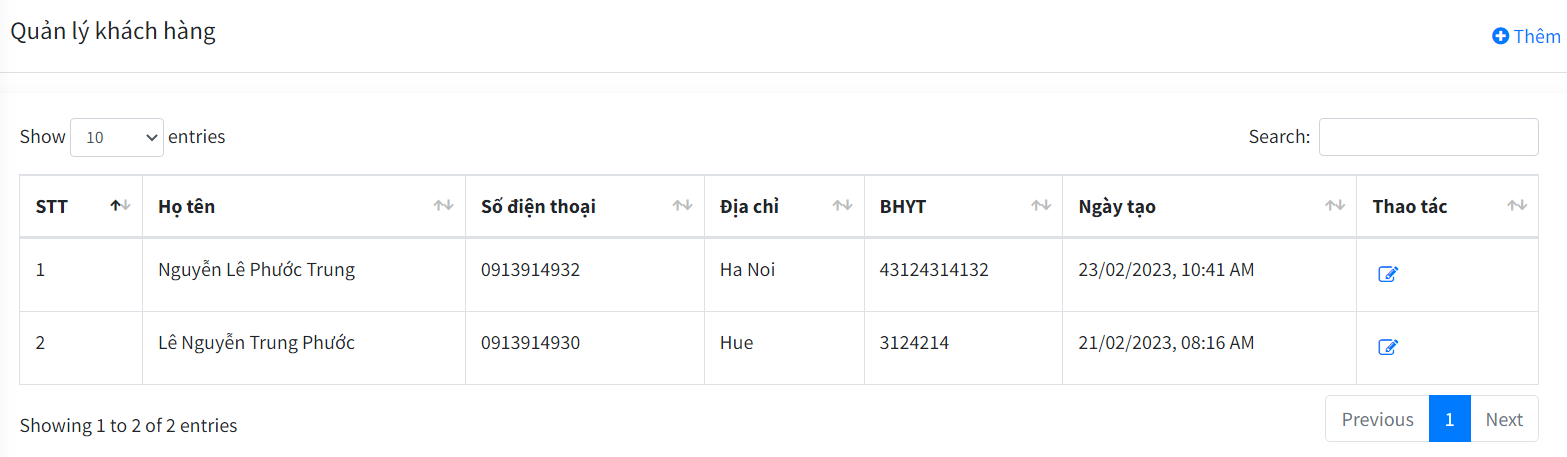
Thành công: Hệ thống sẽ đổi trạng thái lịch hẹn và load lại danh sách lịch hẹn.



Hình . Giao diện sau khi xác nhận lịch hẹn thành công

**3.2.1.4. Quản lý khách hàng**

**3.2.1.4.1. Xem danh sách khách hàng**

**

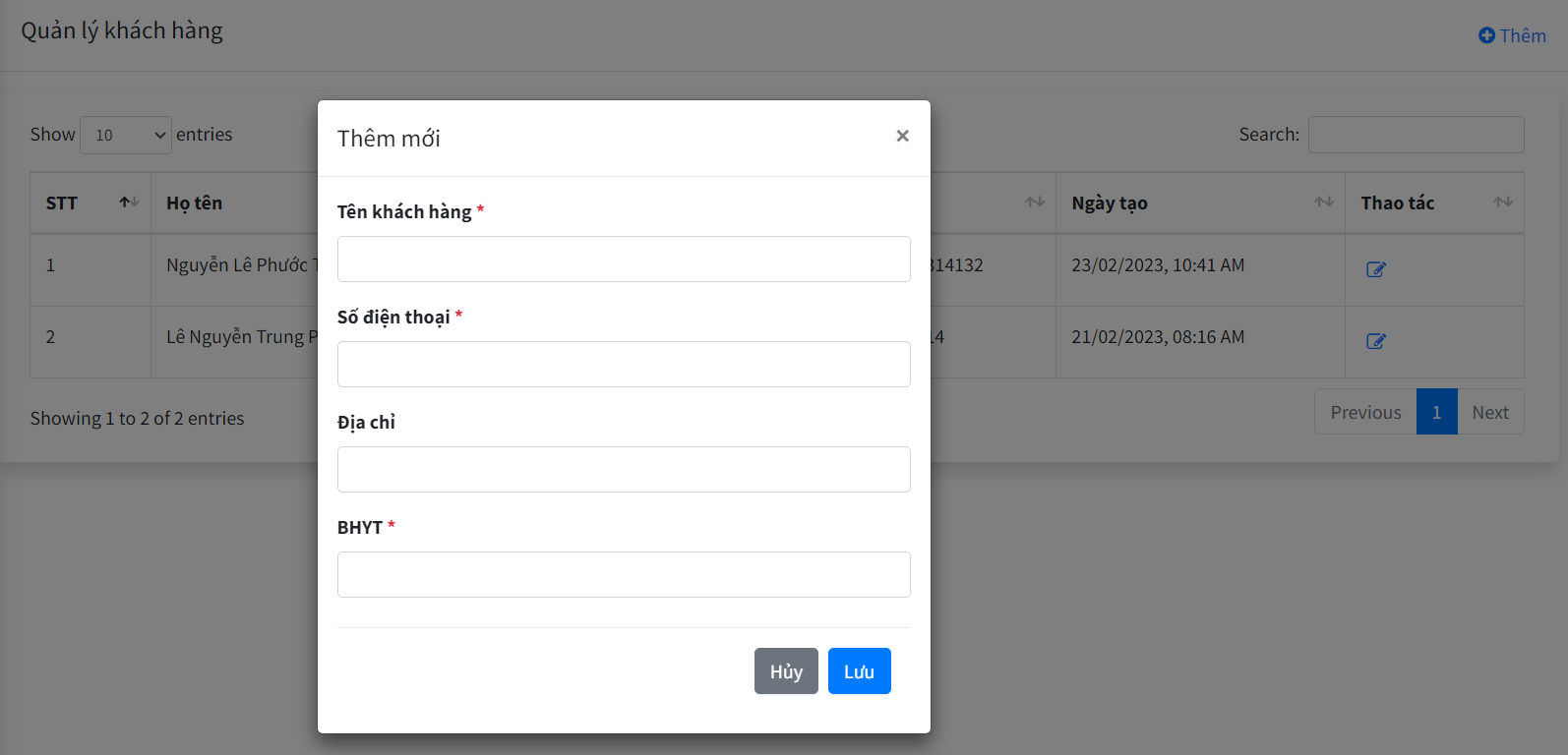
Hình . Giao diện danh sách khách hàng

Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Xem danh sách khách hàng.

Mô tả: Sau khi người dùng bấm vào khách hàng, hệ thống sẽ hiển thị giao diện danh sách khách hàng cho người dùng.

Thành công: Hệ thống điều hướng đến trang khách hàng, lấy toàn bộ danh sách và phân trang, sau đó trả về danh sách như hình trên.

**3.2.1.4.2. Thêm khách hàng**

Hình . Giao diện thêm khách hàng

Truy cập: Người có tài khoản quản lý hệ thống.

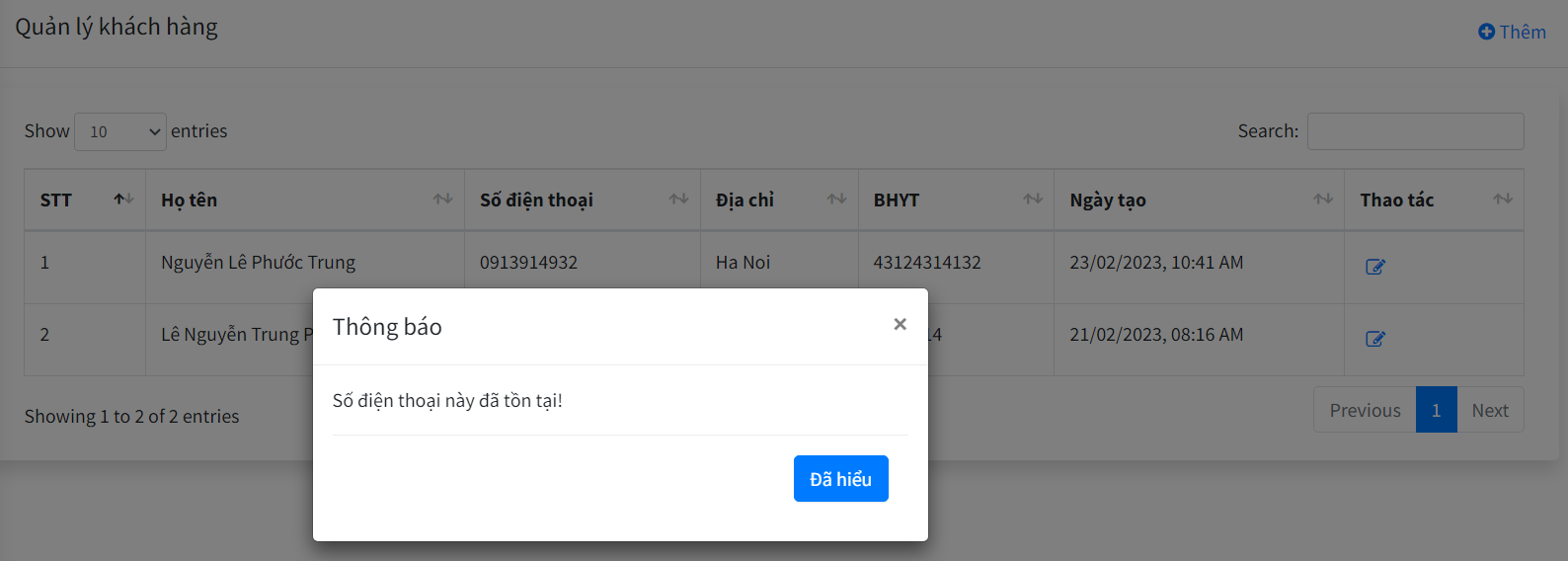
Tên hành động: Thêm khách hàng.

Mô tả: Sau khi người dùng bấm vào nút thêm ở góc phải trên màn hình, hệ thống sẽ hiển thị giao diện thêm mới cho người dùng. Nhập đầy đủ thông tin và bấm lưu để thêm.

Thành công: Hệ thống sẽ lưu lại khách hàng và load lại danh sách lịch hẹn.

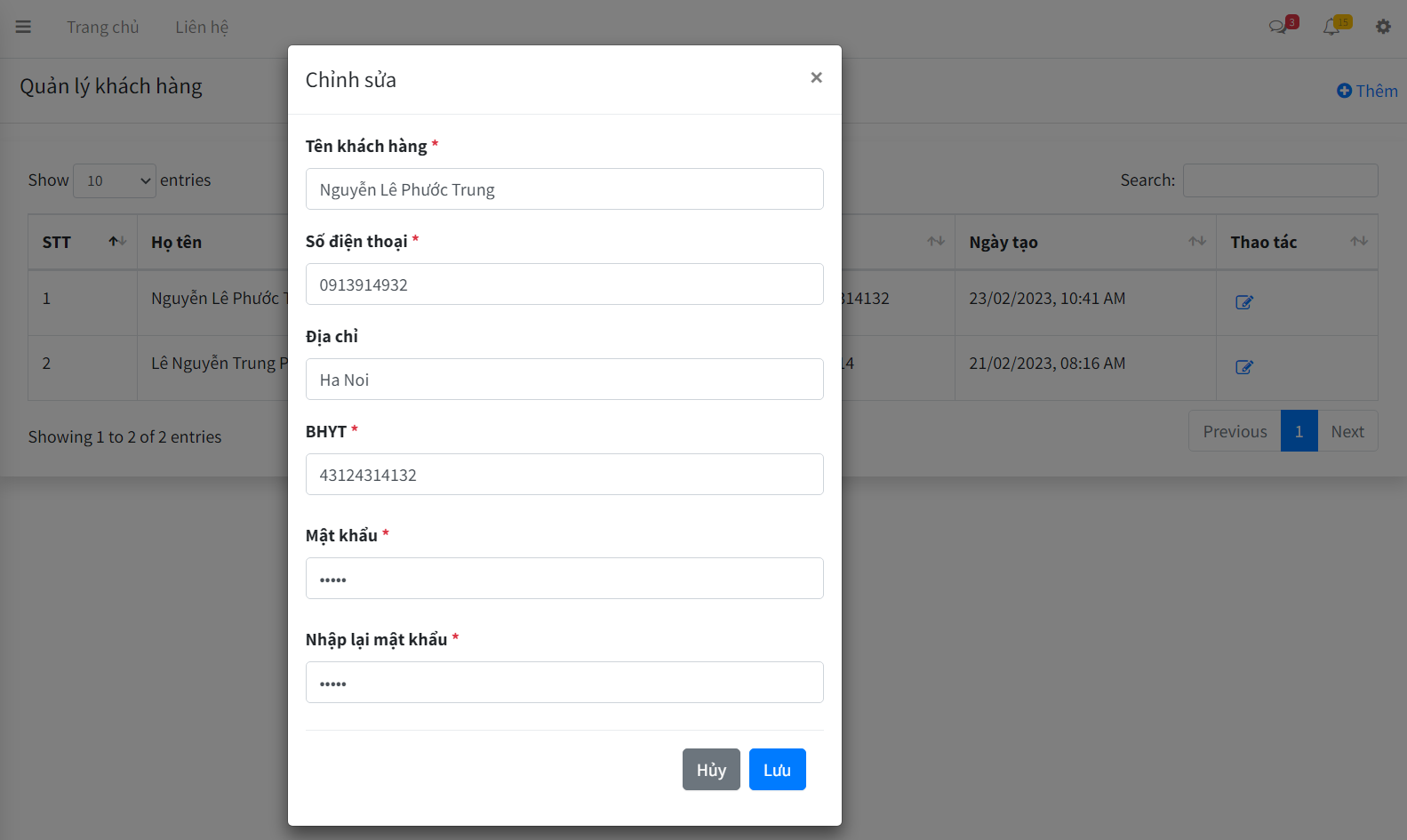
Thất bại:

* Nếu điền thiếu thông tin, khi bấm lưu thì hệ thống sẽ nhắc nhở điền đầy đủ thông tin cần thiết.
* Nếu số điện thoại hoặc bảo hiểm y tế đã trùng hệ thống sẽ thông báo thêm thất bại và nêu lý do.



Hình . Thông báo thêm thất bại khi trùng số điện thoại

**3.2.1.4.3. Sửa thông tin khách hàng**

**

Hình . Giao diện sửa thông tin khách hàng

Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Sửa thông tin khách hàng.

Mô tả: Người dùng bấm nút sửa tương ứng với khách hàng muốn chỉnh sửa, hệ thống sẽ hiển thị giao diện sửa thông tin khách hàng. Người dùng chỉnh sửa những thông tin cần thiết và bấm lưu để hoàn tất.

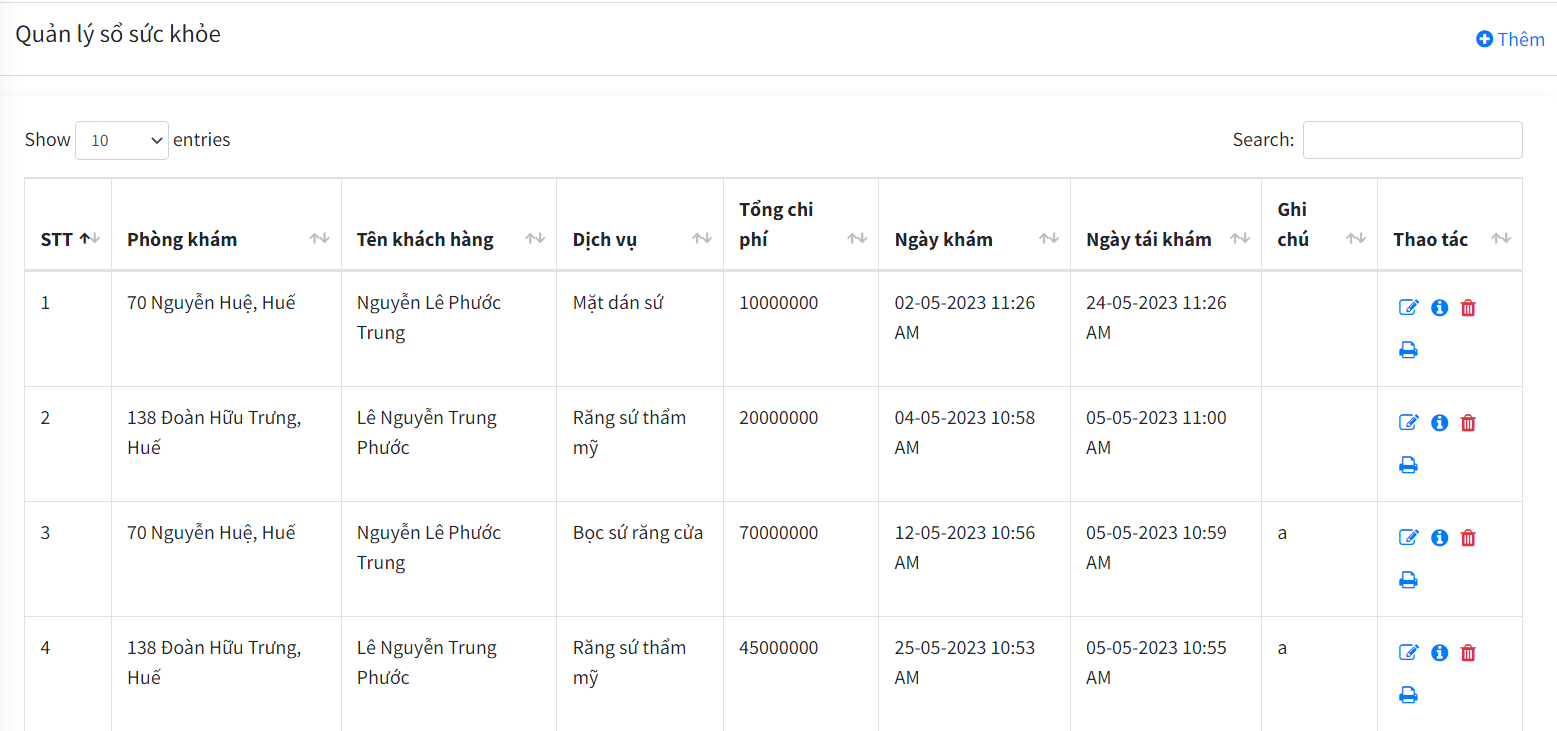
Thành công: Hệ thống sẽ lưu lại thông tin chỉnh sửa và load lại danh sách khách hàng.

Thất bại:

* Nếu điền thiếu thông tin, khi bấm lưu thì hệ thống sẽ nhắc nhở điền đầy đủ thông tin cần thiết.
* Nếu số điện thoại hoặc bảo hiểm y tế đã trùng hệ thống sẽ thông báo thêm thất bại và nêu lý do (Hình 3.11).

**3.2.1.5. Quản lý sổ sức khỏe**

**3.2.1.5.1. Xem danh sách sổ sức khỏe**

****

Hình . Giao diện danh sách sổ sức khỏe

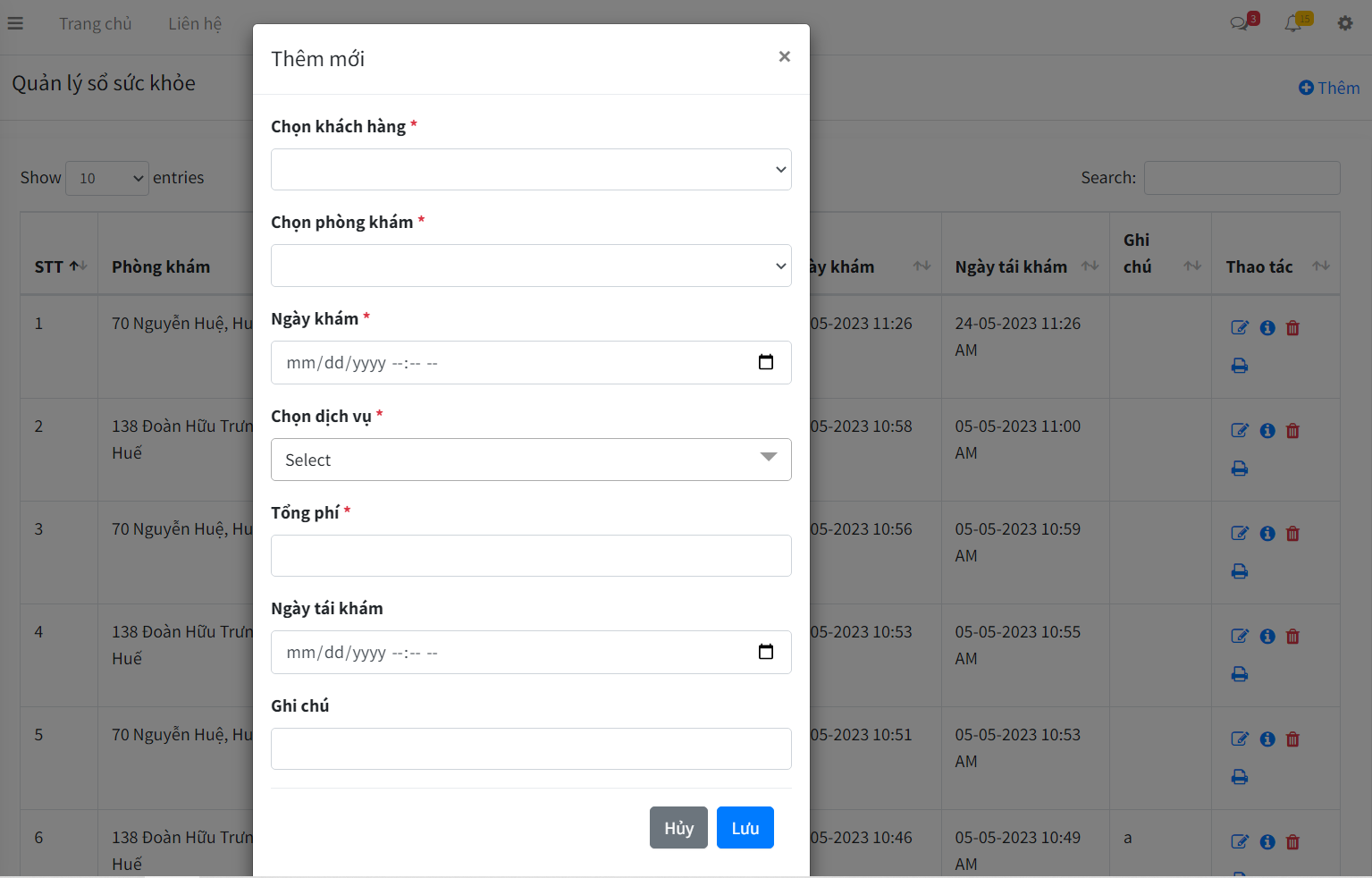
Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Xem danh sách sổ sức khỏe.

Mô tả: Sau khi người dùng bấm vào sổ sức khỏe, hệ thống sẽ hiển thị giao diện danh sách sổ sức khỏe cho người dùng.

Thành công: Hệ thống điều hướng đến trang sổ sức khỏe, lấy toàn bộ danh sách và phân trang, sau đó trả về danh sách như hình trên.

**3.2.1.5.2. Thêm sổ sức khỏe**

****

Hình . Giao diện thêm sổ sức khỏe

Truy cập: Người có tài khoản quản lý hệ thống.

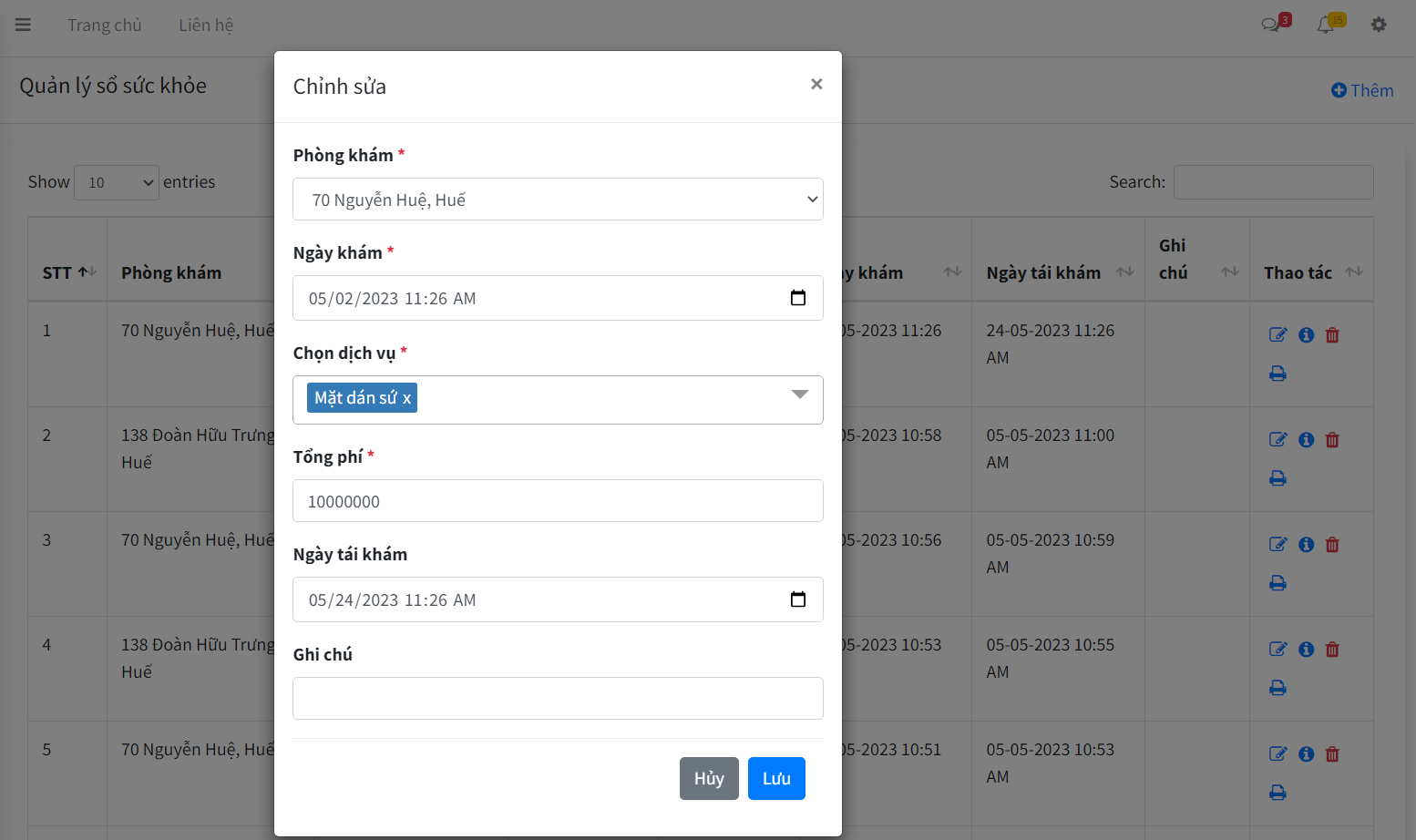
Tên hành động: Thêm sổ sức khỏe.

Mô tả: Sau khi người dùng bấm vào nút thêm ở góc phải trên màn hình, hệ thống sẽ hiển thị giao diện thêm mới cho người dùng. Nhập đầy đủ thông tin và bấm lưu để thêm.

Thành công: Hệ thống sẽ lưu lại lịch hẹn và load lại danh sách sổ sức khỏe.

Thất bại: Nếu điền thiếu thông tin, khi bấm lưu thì hệ thống sẽ nhắc nhở điền đầy đủ thông tin cần thiết.

**3.2.1.5.3. Sửa thông tin sổ sức khỏe**



Hình . Giao diện sửa thông tin sổ sức khỏe

Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Sửa thông tin sổ sức khỏe.

Mô tả: Người dùng bấm nút sửa tương ứng với sổ sức khỏe muốn chỉnh sửa, hệ thống sẽ hiển thị giao diện sửa sổ sức khỏe. Người dùng chỉnh sửa những thông tin cần thiết và bấm lưu để hoàn tất.

Thành công: Hệ thống sẽ lưu lại thông tin chỉnh sửa và load lại danh sách sổ sức khỏe.

Thất bại: Nếu điền thiếu thông tin, khi bấm lưu thì hệ thống sẽ nhắc nhở điền đầy đủ thông tin cần thiết.

**3.2.1.5.4. Xóa sổ sức khỏe**

**

Hình . Giao diện xóa sổ sức khỏe

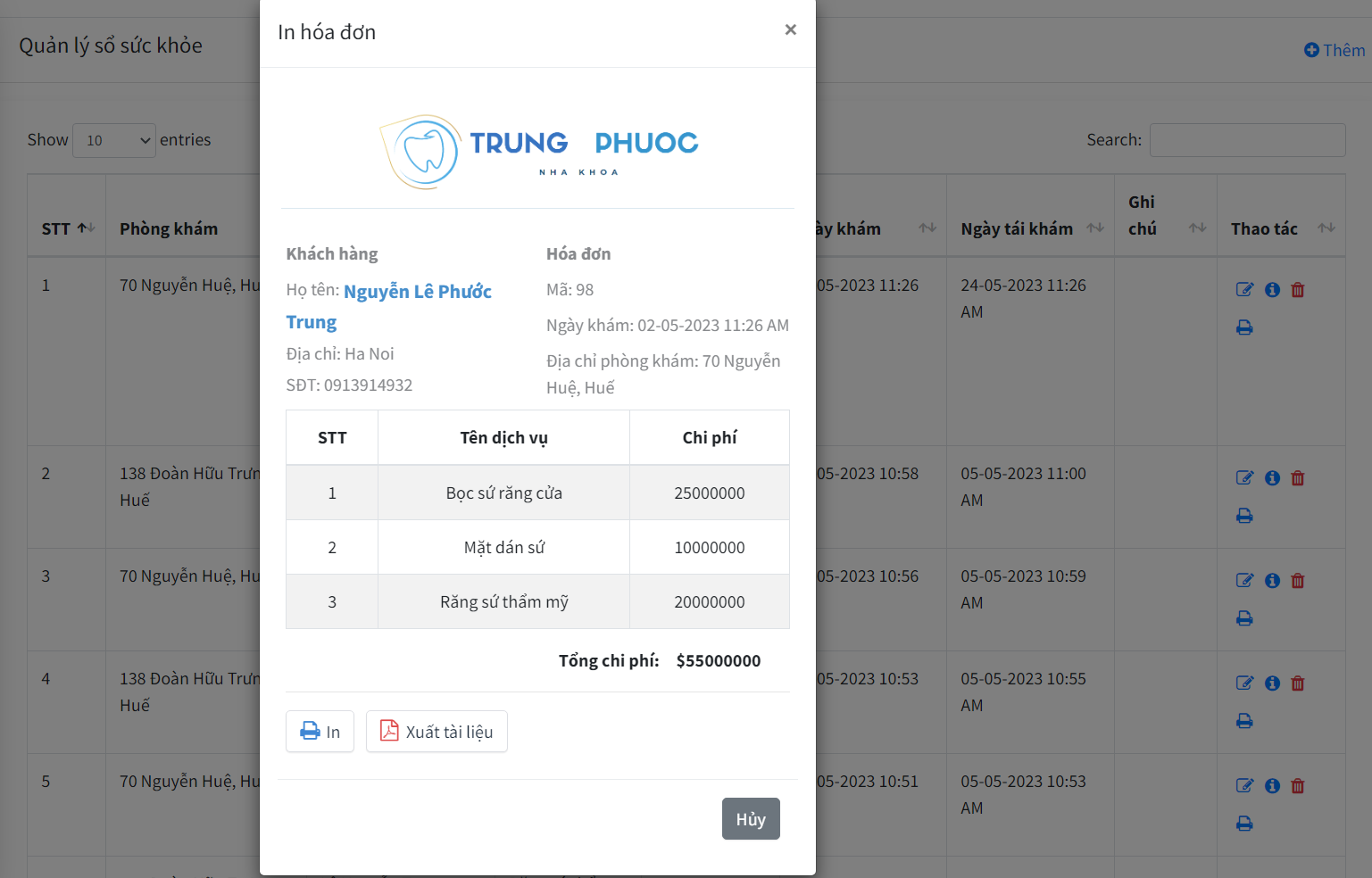
Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Xóa sổ sức khỏe.

Mô tả: Người dùng bấm nút sửa tương ứng với sổ sức khỏe muốn xóa, hệ thống sẽ hỏi lại trước khi xóa. Bấm nút xóa để xác nhận muốn xóa sổ sức khỏe đó.

Thành công: Hệ thống sẽ lưu lại xóa sổ sức khỏe và load lại danh sách sổ sức khỏe.

**3.2.1.5.5. In hóa đơn**

****

Hình . Giao diện hóa đơn

Truy cập: Người có tài khoản quản lý hệ thống.

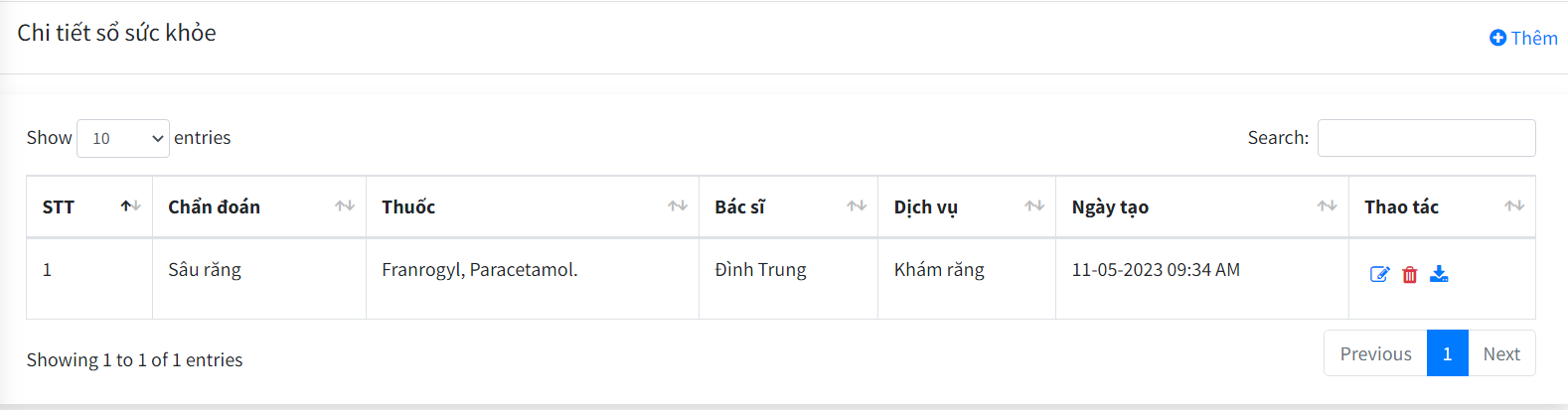
Tên hành động: In hóa đơn.

Mô tả: Người dùng bấm nút in hóa đơn tương ứng với sổ sức khỏe muốn in, hệ thống sẽ đưa ra giao diện hóa đơn trước khi in.

Thành công: Hệ thống sẽ hiển thị thông tin của khách hàng, hóa đơn và các dịch vụ đã sử dụng.

**3.2.1.6. Quản lý chi tiết sổ sức khỏe**

**3.2.1.6.1. Xem danh sách chi tiết sổ sức khỏe**

****

Hình . Giao diện danh sách chi tiết sổ sức khỏe

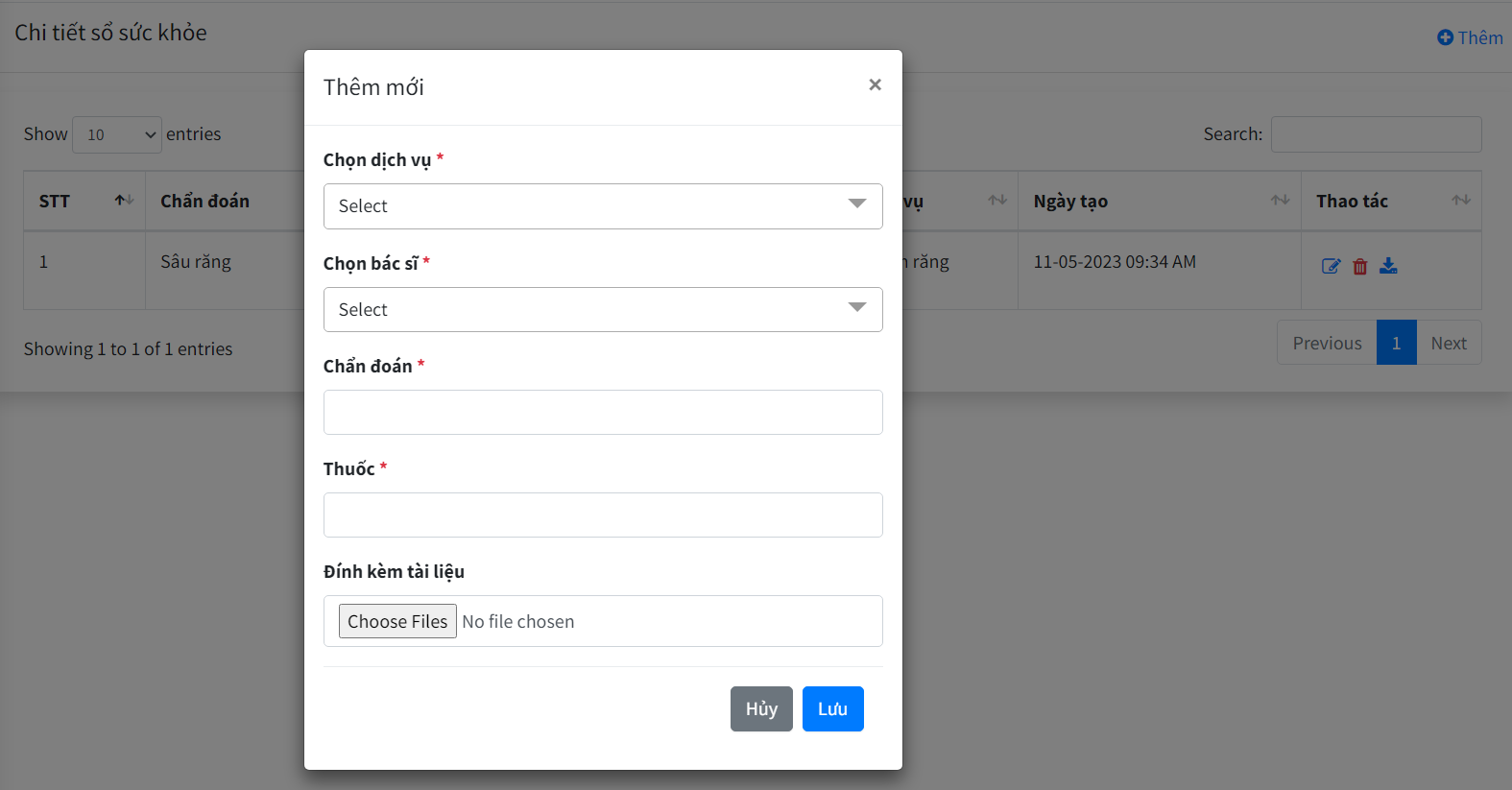
Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Xem danh sách chi tiết sổ sức khỏe.

Mô tả: Sau khi người dùng bấm nút chi tiết có trong từng ô thao tác của mỗi sổ sức khỏe, hệ thống sẽ hiển thị giao diện danh sách chi tiết sổ sức khỏe cho người dùng.

Thành công: Hệ thống điều hướng đến trang chi tiết sổ sức khỏe, lấy toàn bộ danh sách và phân trang, sau đó trả về danh sách như hình trên.

**3.2.1.6.2. Thêm chi tiết sổ sức khỏe**

****

Hình . Giao diện thêm chi tiết sổ sức khỏe

Truy cập: Người có tài khoản quản lý hệ thống.

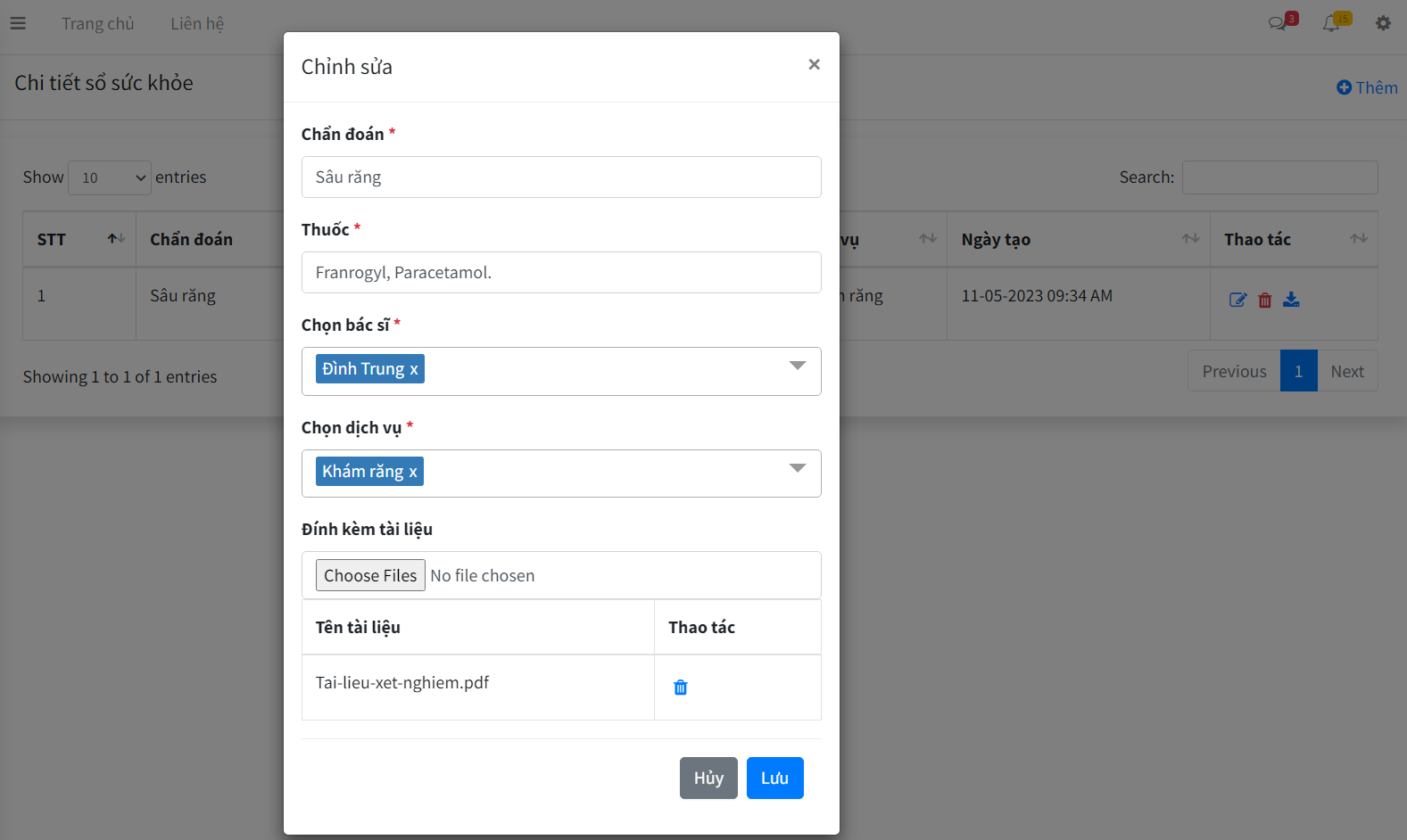
Tên hành động: Thêm chi tiết sổ sức khỏe.

Mô tả: Sau khi người dùng bấm vào nút thêm ở góc phải trên màn hình, hệ thống sẽ hiển thị giao diện thêm mới cho người dùng. Nhập đầy đủ thông tin và bấm lưu để thêm.

Thành công: Hệ thống sẽ lưu lại chi tiết sổ sức khỏe và load lại danh sách.

Thất bại: Nếu điền thiếu thông tin, khi bấm lưu thì hệ thống sẽ nhắc nhở điền đầy đủ thông tin cần thiết.

**3.2.1.6.3. Sửa thông tin chi tiết sổ sức khỏe**

**

Hình . Giao diện sửa thông tin chi tiết sổ sức khỏe

Truy cập: Người có tài khoản quản lý hệ thống.

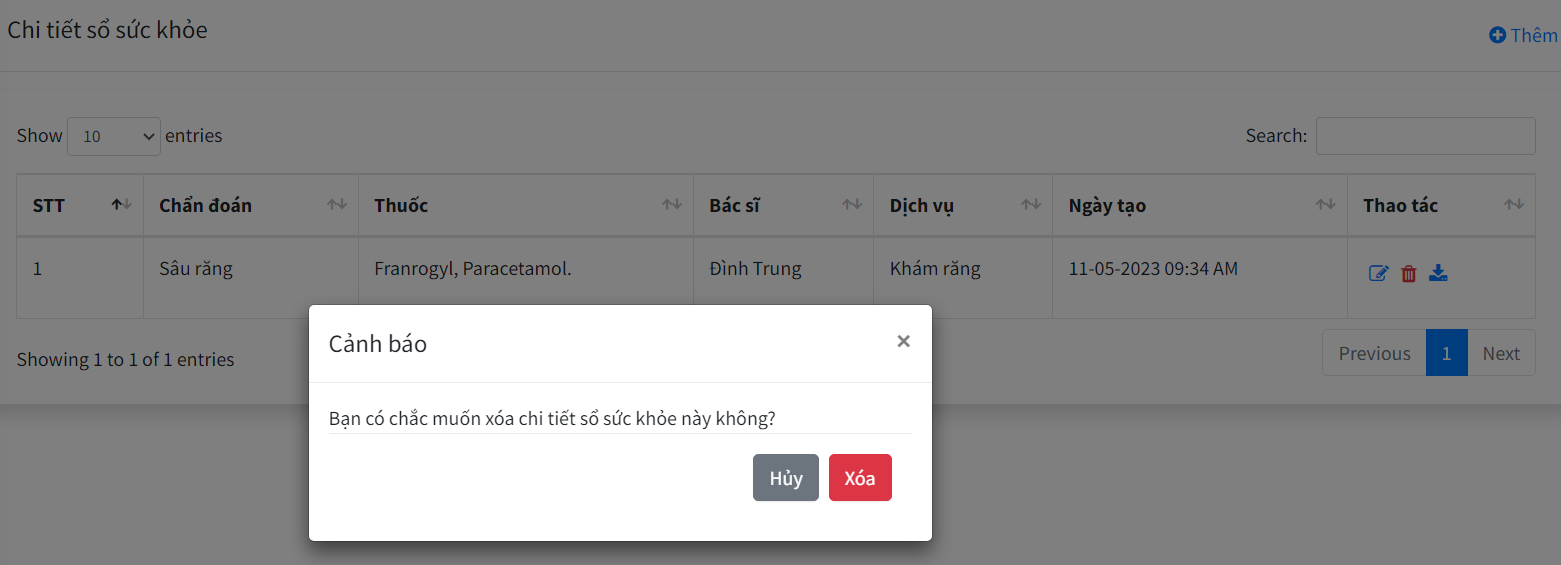
Tên hành động: Sửa thông tin chi tiết sổ sức khỏe.

Mô tả: Người dùng bấm nút sửa tương ứng với chi tiết sổ sức khỏe muốn chỉnh sửa, hệ thống sẽ hiển thị giao diện sửa chi tiết sổ sức khỏe. Người dùng chỉnh sửa những thông tin cần thiết và bấm lưu để hoàn tất.

Thành công: Hệ thống sẽ lưu lại thông tin chỉnh sửa và load lại danh sách chi tiết sổ sức khỏe.

Thất bại: Nếu điền thiếu thông tin, khi bấm lưu thì hệ thống sẽ nhắc nhở điền đầy đủ thông tin cần thiết.

**3.2.1.6.4. Xóa chi tiết sổ sức khỏe**

****

Hình . Giao diện xóa chi tiết sổ sức khỏe

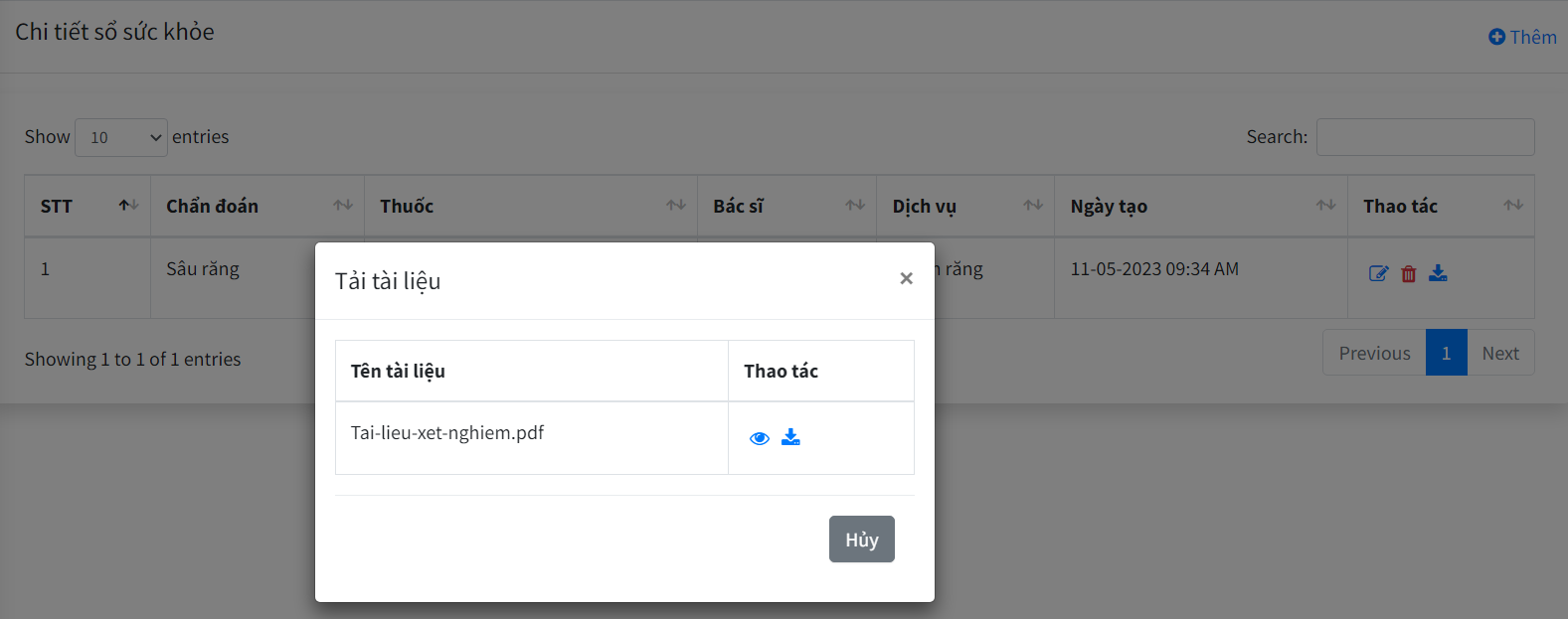
Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Xóa chi tiết sổ sức khỏe.

Mô tả: Người dùng bấm nút xóa tương ứng với chi tiết sổ sức khỏe muốn xóa, hệ thống sẽ hỏi lại trước khi xóa. Bấm nút xóa để xác nhận muốn xóa chi tiết sổ sức khỏe đó.

Thành công: Hệ thống sẽ lưu lại xóa chi tiết sổ sức khỏe và load lại danh sách.

**3.2.1.6.5. Xem, tải tài liệu đính kèm**

****

Hình . Giao diện xem, tải tài liệu đính kèm

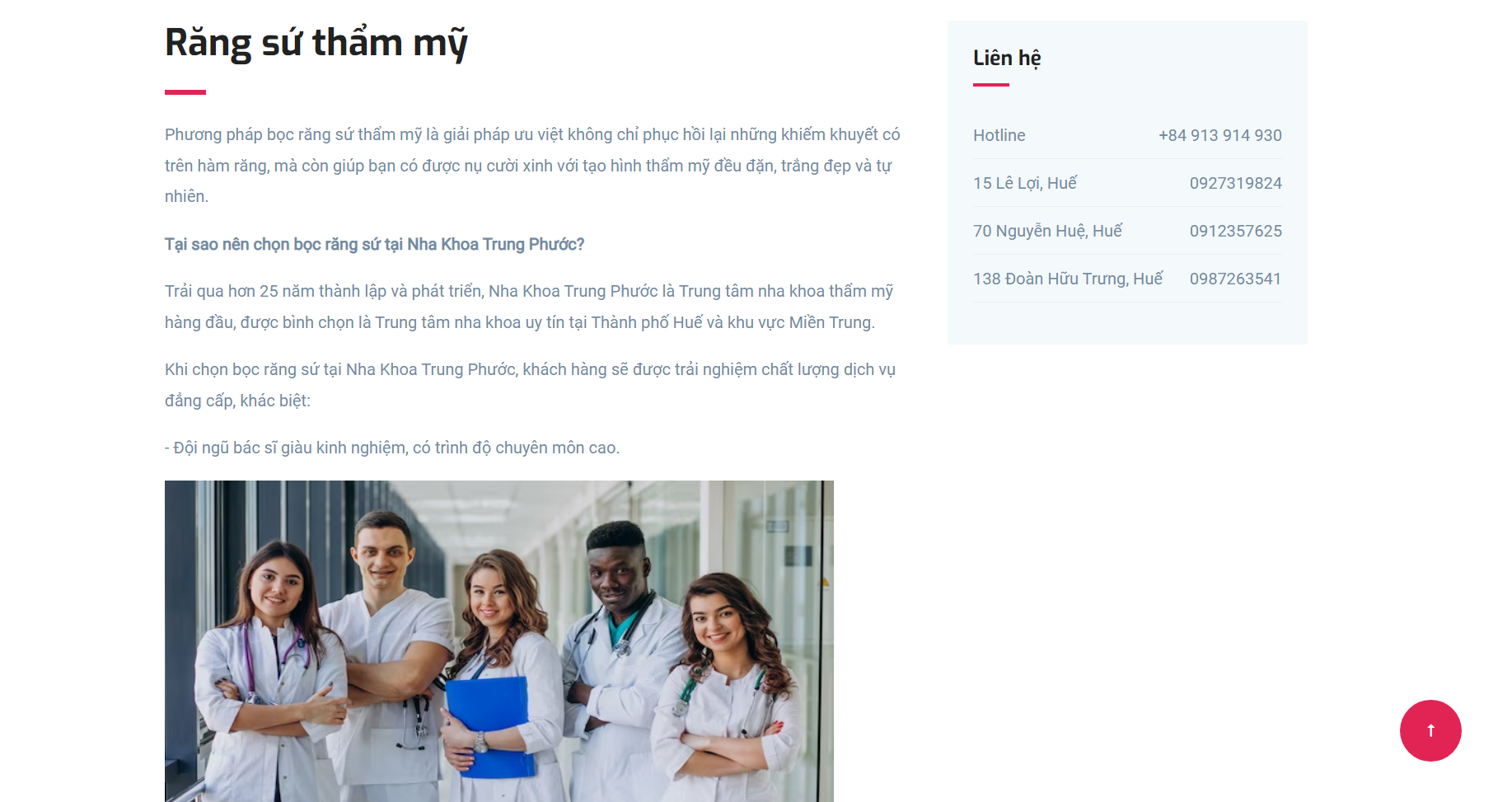
Truy cập: Người có tài khoản quản lý hệ thống.

Tên hành động: Xem, tải tài liệu đính kèm.

Mô tả: Người dùng bấm nút xem để mở tài liệu sang trang mới. Bấm tải để tải tài liệu về máy.

### 3.2.2. Các chức năng của khách

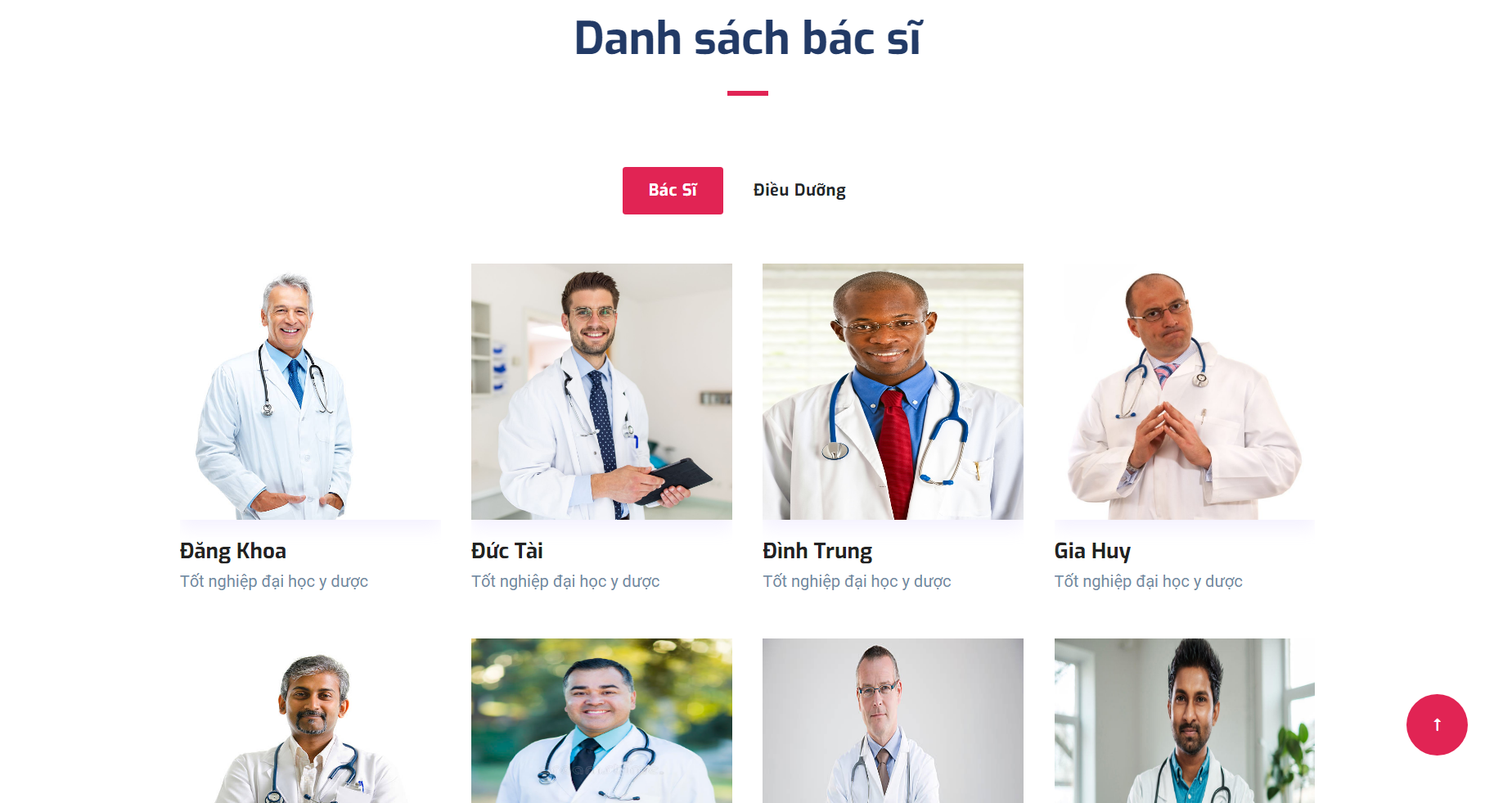
**3.2.2.1. Xem thông tin của phòng khám, đọc tin tức**

******

Hình . Giao diện thông tin về dịch vụ trên trang web

******

Hình . Giao diện bảng giá trên trang web

**

Hình . Giao diện danh sách bác sĩ trên trang web

**

Hình . Giao diện danh sách tin tức trên trang web

|  |  |
| --- | --- |
| Hình . Giao diện thông tin về dịch vụ trên điện thoại | Hình . Giao diện bảng giá trên điện thoại |

|  |  |
| --- | --- |
| Hình . Giao diện thông tin về bác sĩ trên điện thoại | Hình . Giao diện tin tức trên điện thoại |

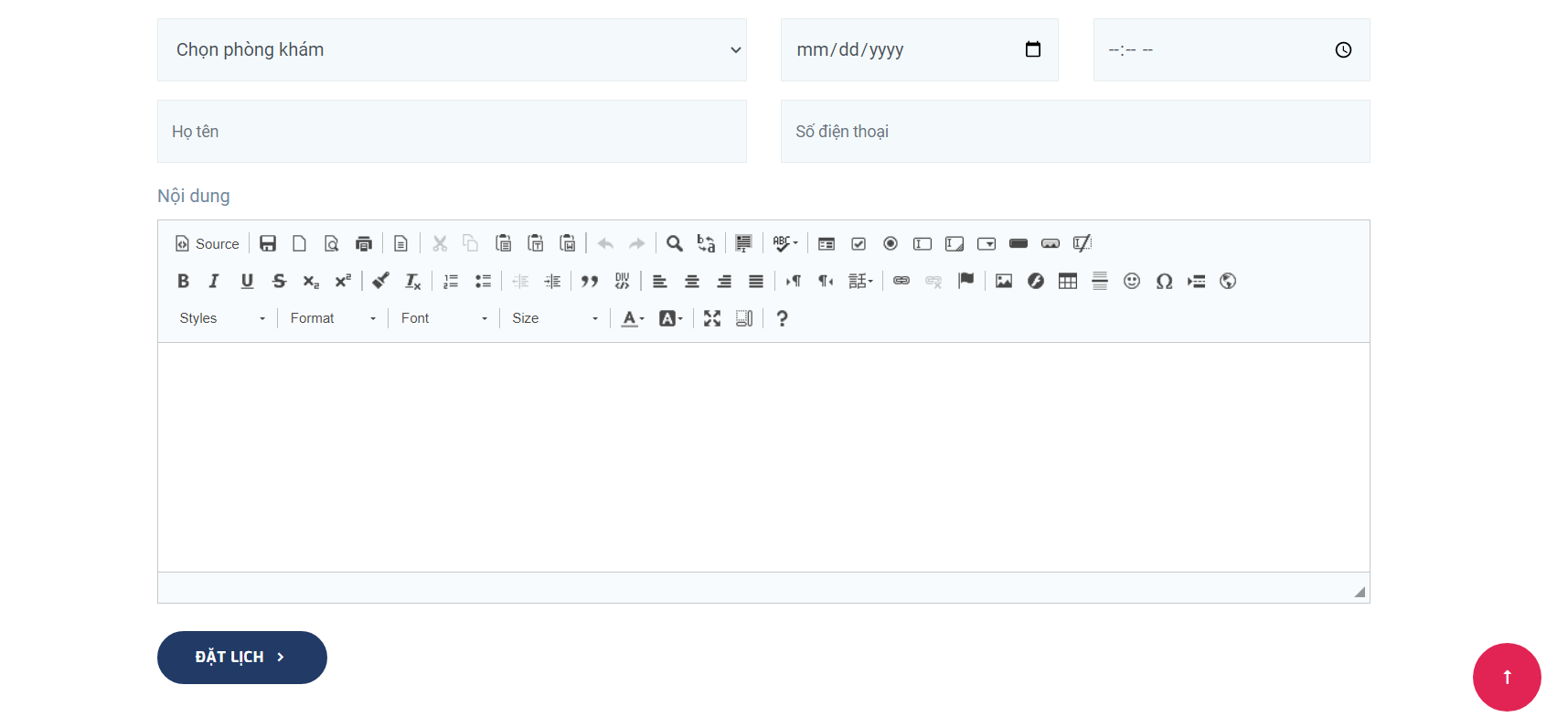
Truy cập: Khách vãng lai, khách hàng của phòng khám.

Tên hành động: Xem thông tin của phòng khám, đọc tin tức.

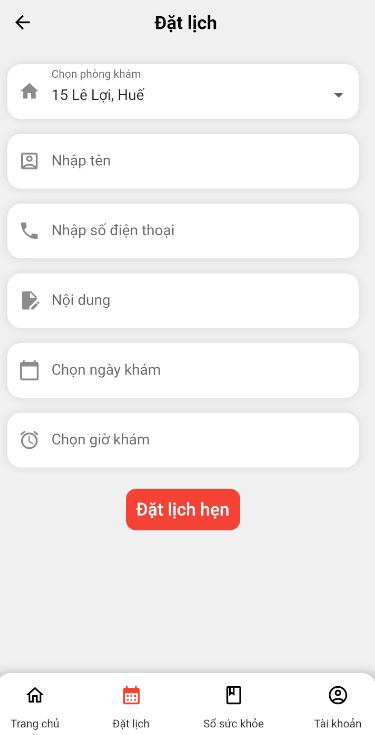
Mô tả: Khách bấm vào thông tin bất kỳ, hệ thống sẽ đưa ra danh sách hoặc thông tin chi tiết.

Thành công: Hệ thống sẽ điều hướng tới trang thông tin và lấy thông tin theo yêu cầu của khách hàng.

**3.2.2.2. Đặt lịch hẹn**

****

Hình . Giao diện đặt lịch hẹn trên trang web

**

Hình . Giao diện đặt lịch hẹn trên điện thoại

Truy cập: Khách vãng lai, khách hàng của phòng khám.

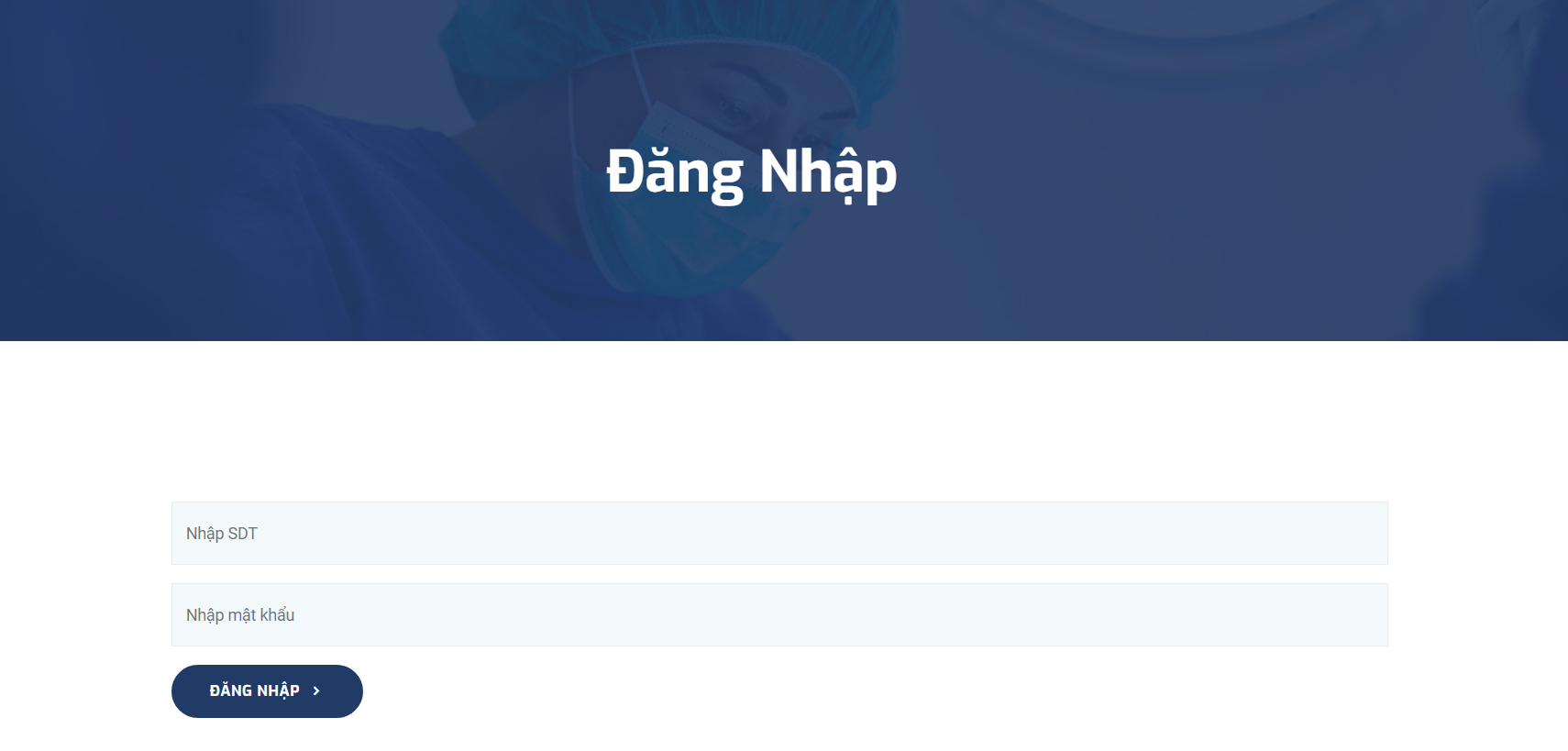
Tên hành động: Đặt lịch hẹn.

Mô tả: Khách vào trang đặt lịch, điền đầy đủ thông tin và bấm nút đặt lịch.

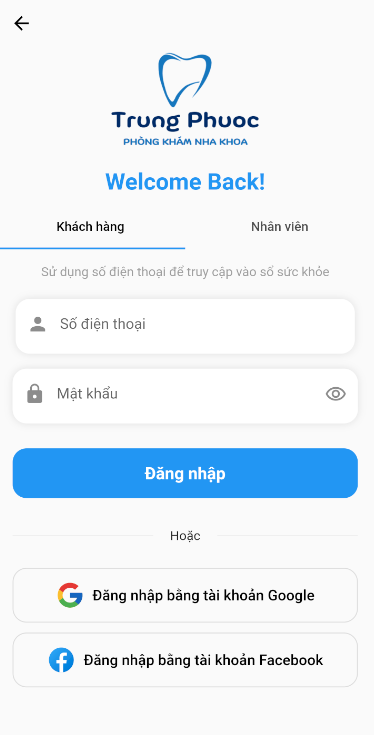
Thành công: Hệ thống lưu lại thông tin lịch hẹn và thông báo thành công.

Thất bại: Nếu điền thiếu thông tin, khi bấm lưu thì hệ thống sẽ nhắc nhở điền đầy đủ thông tin cần thiết.

**3.2.2.3. Đăng nhập**

****

Hình . Giao diện đăng nhập trên trang web

****

Hình . Giao diện đăng nhập trên trang web

Truy cập: Khách.

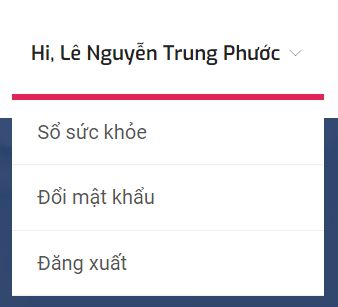
Tên hành động: Đăng nhập.

Mô tả: Khi khách hàng bấm vào sổ sức khỏe mà chưa đăng nhập, hệ thống sẽ điều hướng tới trang đăng nhập. Sau khi nhập tên số điện thoại và mật khẩu thì ấn nút đăng nhập, hệ thống sẽ kiểm tra giá trị vừa nhập vào có khớp với cơ sở dữ liệu hay không.

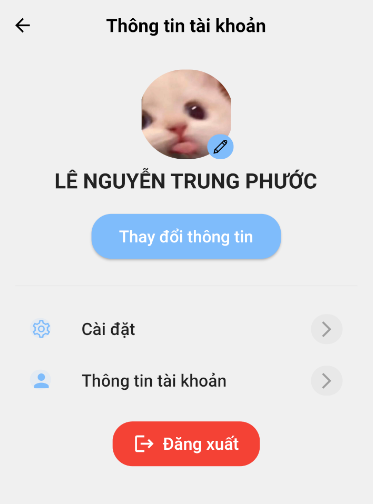
Thành công: Cho phép đăng nhập và điều hướng tới trang sổ sức khỏe.

Thất bại: Nếu kiểm tra thông tin đăng nhập thất bại thì sẽ thông báo “Sai tên đăng nhập hoặc mật khẩu”.

**3.2.2.4. Đăng xuất**

****

Hình . Giao diện đăng xuất trên trang web

**

Hình . Giao diện đăng xuất trên trang web

Truy cập: Khách hàng.

Tên hành động: Đăng xuất.

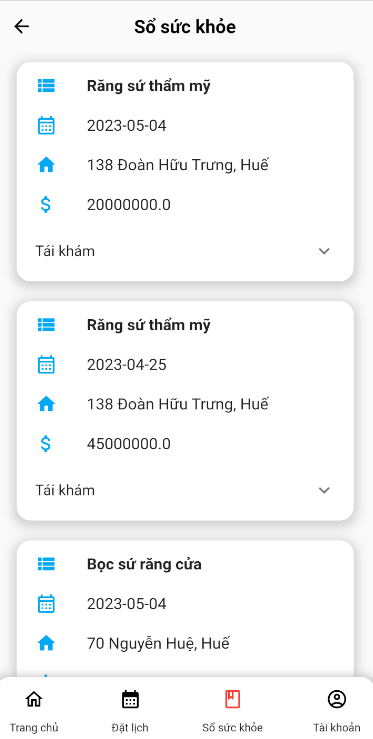
Mô tả: Sau khi Khách hàng bấm đăng xuất, hệ thống sẽ thông tin đăng nhập của khách hàng đó.

Thành công: Xóa thông tin đăng nhập và điều hướng về trang chủ.

**3.2.2.5.** **Theo dõi sổ sức khỏe**

****

Hình . Giao diện sổ sức khỏe của khách hàng trên trang web

**

Hình . Giao diện sổ sức khỏe của khách hàng trên điện thoại

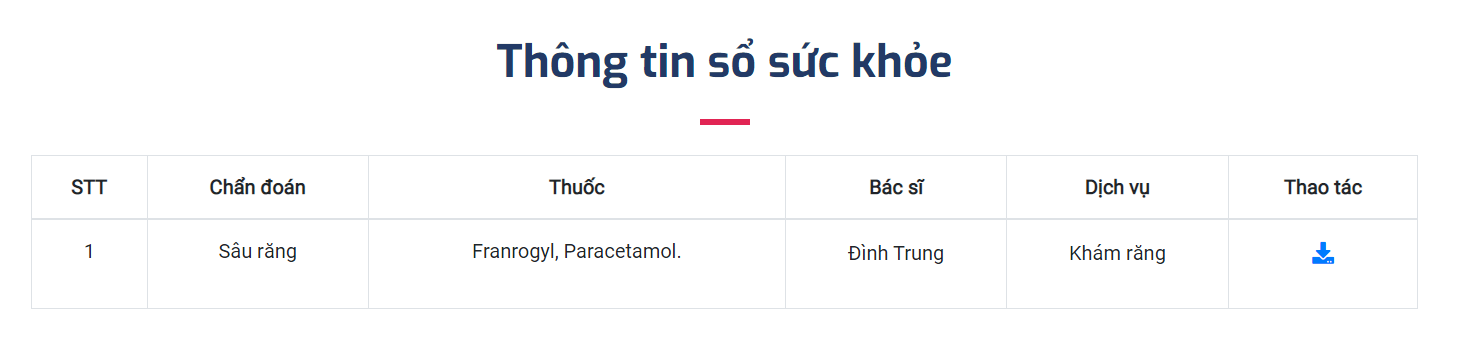
Truy cập: Khách hàng.

Tên hành động: Theo dõi sổ sức khỏe.

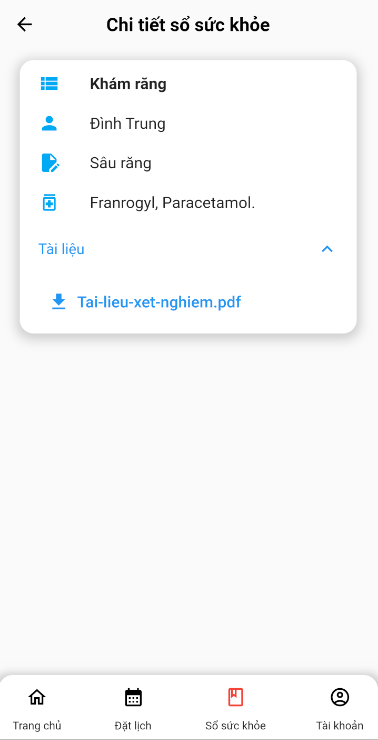
Mô tả: Khi bấm vào nút sổ sức khỏe và đã đăng nhập, khách hàng có thể theo dõi sổ sức khỏe của mình.

Thành công: Điều hướng tới trang sổ sức khỏe và lấy danh sách sổ sức khỏe của khách hàng đã đăng nhập.

**3.2.2.6.** **Theo dõi chi tiết sổ sức khỏe**

****

Hình . Giao diện chi tiết sổ sức khỏe của khách hàng trên trang web

**

Hình . Giao diện chi tiết sổ sức khỏe của khách hàng trên trang điện thoại

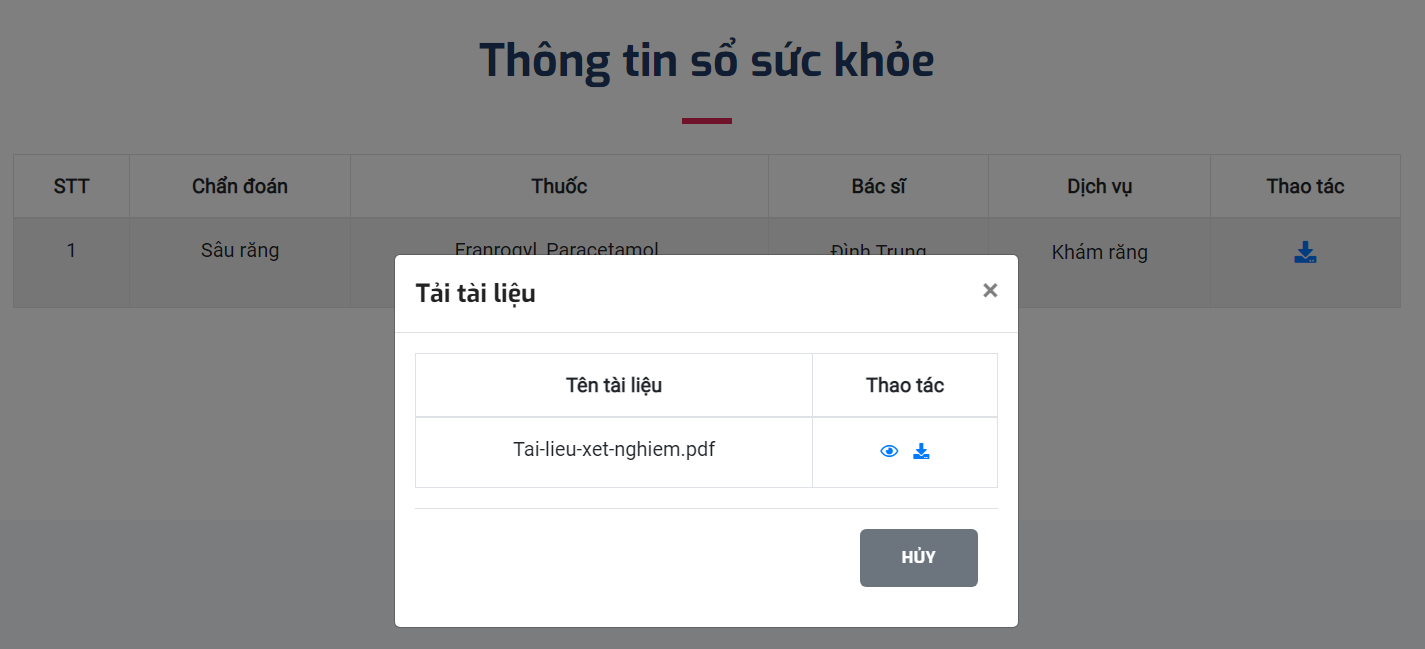
Truy cập: Khách hàng.

Tên hành động: Theo dõi chi tiết sổ sức khỏe.

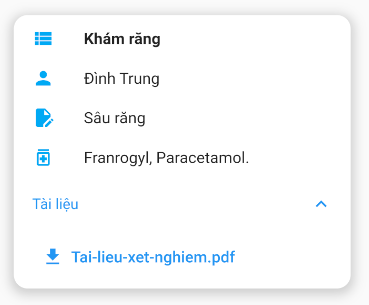
Mô tả: Khi bấm vào nút chi tiết trong ô thao tác của mỗi sổ sức khỏe, khách hàng có thể theo dõi danh sách chi tiết của sổ sức khỏe đó.

Thành công: Điều hướng tới trang chi tiết sổ sức khỏe và lấy danh sách chi tiết của sổ sức khỏe mà khách hàng yêu cầu.

**3.2.2.7.** **Xem, tải tài liệu đính kèm**

****

Hình . Giao diện chi tiết sổ sức khỏe của khách hàng trên trang web

****

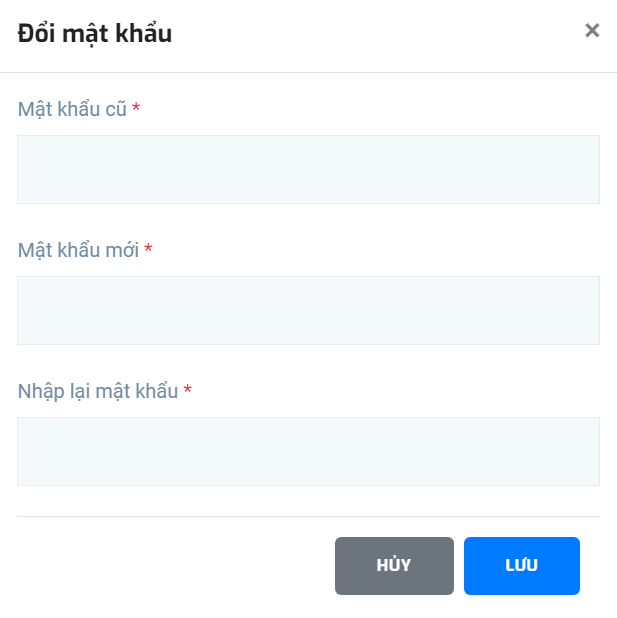
Hình . Giao diện chi tiết sổ sức khỏe của khách hàng trên điện thoại

Truy cập: Khách hàng.

Tên hành động: Xem, tải tài liệu đính kèm.

Mô tả: Người dùng bấm nút xem để mở tài liệu sang trang mới. Bấm tải để tải tài liệu về máy.

**3.2.2.8.** **Đổi mật khẩu**



Hình . Giao diện đổi mật khẩu trên trang web

Truy cập: Khách hàng.

Tên hành động: Đổi mật khẩu.

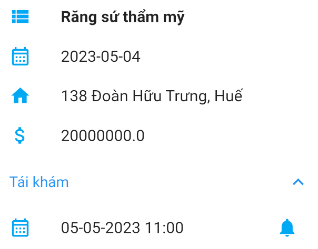
Mô tả: Khách hàng bấm vào nút đổi mật khẩu, nhập mật khẩu cũ, mật khẩu mới và nhập lại mật khẩu. Sau đó bấm lưu để thay đổi.

Thành công: Hệ thống sẽ thay đổi mật khẩu cho khách hàng.

Thất bại:

* Nếu mật khẩu cũ nhập sai thì sẽ báo “Mật khẩu cũ chưa chính xác”.
* Nếu mật khẩu mới và nhập lại mật khẩu không trùng nhau thì thông báo “Mật khẩu mới và nhập lại mật khẩu chưa trùng”.

**3.2.2.9.** **Đặt thông báo nhắc lịch tái khám**

******

Hình . Giao diện đặt thông báo nhắc lịch tái khám

Truy cập: Khách hàng.

Tên hành động: Đặt thông báo nhắc lịch tái khám.

Mô tả: Khách hàng bấm vào chuông bên cạnh lịch tái khám, hệ thống sẽ thông báo trước 24h ngày tái khám.

Thành công: Hệ thống sẽ ghi nhận thời điểm gửi thông báo và gửi thông báo đúng thời gian.

# KẾT LUẬN

## 1. Kết quả đạt được

Sau thời gian tìm hiểu và thực hiện khóa luận, em đã học hỏi được nhiều kiến thức mới cũng như hoàn thiện tương đối hệ thống ứng dụng, cụ thể như sau:

* Áp dụng được những kiến thức đã học ở trường trong việc phân tích và thiết kế hệ thống, cũng như sử dụng các kiến thức nền tảng về mô hình thiết kế Client – Server hay quản trị cơ sở dữ liệu bằng SQL Server để đưa vào bài toán thực tế.
* Tìm hiểu thêm nhiều kiến thức mới về .NET, Angular và Flutter cũng như ưu nhược điểm của nó để từ đó lựa chọn xây dựng hệ thống, đồng thời học hỏi được cách cài đặt một hệ thống đa nền tảng hoàn chỉnh bằng các công nghệ trên. Cùng với đó, rèn luyện được thêm các kỹ năng sử dụng các công nghệ thiết kế front-end như Bootstrap, jQuery.
* Sau một thời gian học tập công nghệ mới và bắt tay xây dựng, ứng dụng đa nền tảng cho phòng khám nha khoa đã hoàn thiện được các chức năng đề ra ban đầu, sát với quy trình nghiệp vụ thực tế, cùng với giao diện thân thiện.

## 2. Vấn đề hạn chế

Cùng với các kết quả đã đạt được như trên thì trong thời gian ngắn vừa tìm hiểu vừa thực hiện khóa luận, hệ thống không tránh khỏi tồn tại một số hạn chế:

* Sử dụng mô hình Client – Server nên có thể tồn tại vấn đề về bảo mật dữ liệu thông tin.
* Chưa vận dụng tốt sự hỗ trợ của các Framework vào hệ thống.
* Cần thêm hoạt động tương tác của phòng khám với khách hàng thông qua ứng dụng.
* Chưa có điều kiện đưa hệ thống vào hoạt động ở thực tế.

## 3. Hướng phát triển

Đối với những hạn chế vừa nêu, trong khoảng thời gian sắp tới sẽ được nghiên cứu khắc phục và đưa vào sử dụng, trong đó các giải pháp để giải quyết các nhược điểm đó có thể kể đến như sau:

* Tăng cường bảo mật cho hệ thống.
* Tìm hiểu sâu về các Framework, tối ưu hóa hệ thống, nâng cao khả năng lập trình để cải thiện hiệu suất vận hành của hệ thống.
* Xây dựng thêm các chức năng như tư vấn, giải đáp thắc mắc trực tuyến. Đăng nhập, bình luận thông qua Facebook để ứng dụng tiếp cận được nhiều đối tượng hơn và các hoạt động trên hệ thống được diễn ra suôn sẻ, khả năng tương tác cao hơn. Thông báo ngay lập tức khi có khách hàng đặt lịch.
* Sau khi đưa vào vận hành, có thể thu thập ý kiến của người dùng để cải thiện chất lượng và tính năng của website cho phù hợp hơn với phần đông người sử dụng.

# TÀI LIỆU THAM KHẢO

**Tiếng Anh**

[1] Angular documentation - https://angular.io/docs

[2] Angular Feature - <https://www.syncfusion.com/blogs/post/top-10-features-in-angular-13-every-developer-should-know.aspx>

[3] <https://aws.amazon.com/vi/what-is/net/>

**Tiếng Việt**

[4] <https://itnavi.com.vn/blog/angular-la-gi>

[5] <https://viblo.asia/p/gioi-thieu-tong-quan-ve-angular-07LKX9j2ZV4>

[6] <https://wiki.tino.org/angular-la-gi/>

[7] https://levunguyen.com/laptrinhweb/2021/06/13/su-dung-directive-trong-angular/

[8] <https://hocjavascript.net/angular/angular-la-gi-uu-diem-va-nhuoc-diem/>

<https://vietnix.vn/flutter-la-gi/>

[9] https://itnavi.com.vn/blog/flutter-la-gi

[10] <https://websitehcm.com/cau-truc-ung-dung-flutter/>

[11] <https://200lab.io/blog/flutter-la-gi/>

[12] <https://glints.com/vn/blog/so-sanh-flutter-vs-react-native/#.ZFcp_Hb99EY>

[13] <https://topdev.vn/blog/flutter-vs-react-native-so-sanh-chi-tiet-ve-nhung-diem-tuong-dong-va-uu-viet/>

[14] [https://viblo.asia/p/gioi-thieu-ve-net-net-Framework-aspnet-oop-gGJ59NDaKX2](https://viblo.asia/p/gioi-thieu-ve-net-net-framework-aspnet-oop-gGJ59NDaKX2)

[15] https://aws.amazon.com/vi/what-is/net/

[16] https://topdev.vn/blog/anh-em-nha-net-phan-1/

[17] <https://mona.media/net-la-gi/>

[18] <https://fptshop.com.vn/tin-tuc/danh-gia/dotnet-la-gi-154186>

[19] https://codelearn.io/sharing/tim-hieu-ve-mo-hinh-client-server