

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
KHOA CÔNG NGHỆ SỐ

ĐỒ ÁN TỐT NGHIỆP
ĐẠI HỌC

NGÀNH: Công nghệ thông tin
CHUYÊN NGÀNH: Công nghệ thông tin

ĐỀ TÀI:

XÂY DỰNG ỨNG DỤNG KHÁM
CHỮA BỆNH ONLINE TẠI BỆNH VIỆN
ĐA KHOA ĐÀ NẴNG

Người hướng dẫn: ThS. Nguyễn Văn Phát
Sinh viên thực hiện: Đặng Minh Hiếu
Mã sinh viên: 1811505310109
Lớp: 18T1

Đà Nẵng, 01/2022

**ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
KHOA CÔNG NGHỆ SỐ**

**ĐỒ ÁN TỐT NGHIỆP
ĐẠI HỌC**

**NGÀNH: Công nghệ thông tin
CHUYÊN NGÀNH: Công nghệ thông tin**

ĐỀ TÀI:

**XÂY DỰNG ỨNG DỤNG KHÁM
CHỮA BỆNH ONLINE TẠI BỆNH VIỆN
ĐA KHOA ĐÀ NẴNG**

Người hướng dẫn: ThS. Nguyễn Văn Phát
Sinh viên thực hiện: Đặng Minh Hiếu
Mã sinh viên: 1811505310109
Lớp: 18T1

Đà Nẵng, 01/2022

TÓM TẮT

1. Tên đề tài: XÂY DỰNG ỨNG DỤNG KHÁM CHỮA BỆNH ONLINE TẠI BỆNH VIỆN ĐA KHOA ĐÀ NẴNG.
2. Sinh viên thực hiện: ĐẶNG MINH HIẾU
3. Mã SV: 1811505310109
4. Lớp: 18T1

Qui trình xây dựng ứng dụng khám chữa bệnh Online tại Bệnh viện Đa khoa Đà Nẵng bao gồm:

- Khảo sát thực tế nhu cầu khám chữa bệnh của người dùng.
- Tập hợp tài liệu các phần mềm liên quan.
- Phân tích chức năng hệ thống
- Phân tích cơ sở dữ liệu
- Xây dựng ứng dụng
- Xây dựng website
- Chỉnh sửa và hoàn thiện
- Kiểm tra và vận hành
- Đánh giá

LỜI CẢM ƠN

Ngày nay, công nghệ thông tin đã có những bước phát triển mạnh mẽ theo cả chiều rộng và sâu. Máy tính điện tử không còn là một thứ phương tiện quý hiếm mà đang ngày càng trở thành một công cụ làm việc và giải trí thông dụng của con người, không chỉ ở công sở mà còn ngay cả trong gia đình.

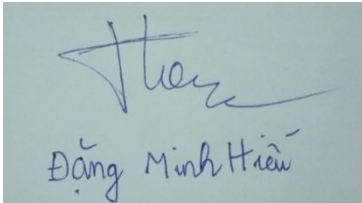
Công nghệ thông tin đang dần chứng tỏ tầm ảnh hưởng rất lớn đến mọi mặt của đời sống xã hội. Đối với hoạt động của ngành y tế, có thể thấy rằng, CNTT ngày càng đóng vai trò quan trọng, không chỉ là mũi nhọn cho quá trình cải cách hành chính trong công tác quản lý, điều hành của cơ quan quản lý mà còn “đỡ đầu” cho việc triển khai và ứng dụng thành công các kỹ thuật cao trong KCB như chụp cắt lớp, mổ nội soi... thăm khám cho bệnh nhân qua hệ thống điện tử..

Đứng trước đại dịch Covid-19 nhu cầu khám bệnh đặt lịch trước và tư vấn bệnh từ xa tăng cao. Để giải quyết vấn đề trên em đã tìm hiểu và xây dựng “Ứng dụng khám chữa bệnh online tại bệnh viện Đa khoa Đà Nẵng “ nhằm giải quyết vấn đề tụ tập nơi đông người và giãn cách xã hội, tránh ra đường khi không cần thiết. Vai trò của hệ thống giúp người dùng đặt lịch và khám bệnh online, theo dõi quá trình điều trị bệnh, thuốc điều trị, ngày tái khám, vv... Giúp bệnh viện không quá tải và đảm bảo được giãn cách xã mà vẫn đảm bảo có thể khám bệnh cho người dân.

Với sự hướng dẫn tận tình của thầy Nguyễn Văn Phát , em đã hoàn thành đồ án tốt nghiệp này. Tuy đã cố gắng hết sức tìm hiểu, phân tích thiết kế và cài đặt hệ thống nhưng chắc rằng không tránh khỏi những thiếu sót. Em rất mong nhận được sự thông cảm và góp ý của quý Thầy cô.

Em xin chân thành cảm ơn !

Sinh viên thực hiện



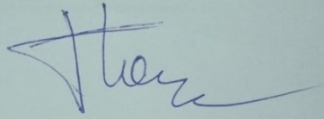
Đặng Minh Hiền

CAM ĐOAN

Chúng tôi xin cam đoan :

- 1 Những nội dung trong luận văn này do chúng em thực hiện dưới sự hướng dẫn của thầy Nguyễn Văn Phát
- 2 Mọi tham khảo trong luận văn đều được trích dẫn rõ ràng tên tác giả, tên công trình ,thời gian, địa điểm công bố.
- 3 Mọi sao chép không hợp lệ, vi phạm quy chế đào tạo, hay gian trá, chúng em xin chịu hoàn toàn trách nhiệm

Sinh viên thực hiện



Đặng Minh Hiền

MỤC LỤC

Nhận xét của người hướng dẫn	
Nhận xét của người phản biện	
Tóm tắt	
Nhiệm vụ đồ án	
Lời cảm ơn.....	i
Lời cam đoan	ii
Mục lục	iii
Danh sách các bảng, hình vẽ	v
Danh sách các ký hiệu, chữ viết tắt	v
	Trang

CHƯƠNG 1: TỔNG QUAN.....1

1.1. PHÂN TÍCH HIỆN TRẠNG.....	1
1.2. LÍ DO CHỌN ĐỀ TÀI	1
1.3. MỤC ĐÍCH ĐỀ TÀI.....	1
1.4. PHẠM VI ĐỀ TÀI	1
1.5. ĐỐI TƯỢNG NGHIÊN CỨU	1
1.6. PHƯƠNG PHÁP NGHIÊN CỨU	1
1.7. CÔNG NGHỆ SỬ DỤNG	2
1.8. CƠ CẤU TỔ CHỨC	2
1.9. HOẠT ĐỘNG NGHIỆP VỤ THỰC TẾ	2
1.10. QUI TRÌNH NGHIỆP VỤ CỦA ĐỀ TÀI.....	5

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT6

2.1. PHẦN MỀM.....	6
2.1.1. <i>Visual Studio Code</i>	6
2.1.2. <i>phpMyAdmin</i>	7
2.1.3. <i>Công cụ Postman</i>	7
2.2. CÔNG NGHỆ	8

2.2.1. Công nghệ Bootstrap	8
2.2.2. Công nghệ NodeJS.....	9
2.2.3. Công nghệ WebRTC.....	11
2.2.4. PeerJs.....	12
CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG.....	14
3.1. ĐẶC TẢ YÊU CẦU PHẦN MỀM (SOFTWARE REQUIREMENTS)	14
3.1.1. Nhân viên bộ phận đón tiếp	14
3.1.2. Bác sĩ/Trợ lý bác sĩ.....	14
3.1.3. Bệnh nhân.....	15
3.1.4. Phòng IT.....	15
3.2. SƠ ĐỒ USECASE.....	16
3.3. KỊCH BẢN HOẠT ĐỘNG.....	16
3.3.1. Nhân viên bộ phận đón tiếp	16
3.3.2. Bác sĩ/Trợ lý bác sĩ.....	17
3.3.3. Bệnh nhân.....	19
3.4. SƠ ĐỒ HOẠT ĐỘNG.....	19
3.4.1. Nhân viên bộ phận đón tiếp	20
3.4.2. Bác sĩ/Trợ lý bác sĩ.....	21
3.4.3. Bệnh nhân.....	23
3.5. MÔ HÌNH DỮ LIỆU	23
3.5.1. Sơ đồ ERD.....	24
CHƯƠNG 4: TRIỂN KHAI CHƯƠNG TRÌNH.....	25
4.1. NHÂN VIÊN BỘ PHẬN ĐÓN TIẾP	25
4.1.1. Xác nhận phiếu hẹn.....	25
4.1.2. Thống kê tình hình khám bệnh của bệnh viện.....	25
4.2. BÁC SĨ/TRỢ LÝ BÁC SĨ.....	26

4.2.1. Tìm kiếm bệnh nhân	26
4.2.2. Phê duyệt lịch hẹn	26
4.2.3. Xây dựng lộ trình điều trị.....	27
4.3. BỆNH NHÂN	29
4.3.1. Đăng ký tài khoản khám bệnh.....	29
4.3.2. Đặt lịch khám bệnh	31
KẾT LUẬN & HƯỚNG PHÁT TRIỂN.....	33
TÀI LIỆU THAM KHẢO.....	34

DANH SÁCH CÁC BẢNG, HÌNH VẼ

HÌNH 1.1: CƠ CẤU TỔ CHỨC CỦA BỆNH VIỆN ĐA KHOA ĐÀ NẴNG	2
HÌNH 1.2: QUI TRÌNH KHÁM CHỮA BỆNH TẠI BỆNH VIỆN	3
HÌNH 1.3: QUI TRÌNH KHÁM CHỮA BỆNH ĐẶT LỊCH TRƯỚC NGOÀI THỰC TẾ.....	3
HÌNH 1.4: QUI TRÌNH CÁC BƯỚC NHẬP VIỆN	4
HÌNH 1.5: QUI TRÌNH TƯ VẤN KHÁM BỆNH TỪ XA CỦA ỨNG DỤNG BOOKINGCARE.....	4
HÌNH 1.6: QUI TRÌNH NGHIỆP VỤ CỦA ĐỀ TÀI.....	5
HÌNH 3.1: SƠ ĐỒ USECASE	16
HÌNH 3.9: SƠ ĐỒ HOẠT ĐỘNG “XÁC NHẬN PHIẾU HẸN” CỦA NHÂN VIÊN ĐÓN TIẾP.....	20
HÌNH 3.10: SƠ ĐỒ HOẠT ĐỘNG “PHÊ DUYỆT LỊCH HẸN” CỦA BÁC SĨ.....	21
HÌNH 3.11: SƠ ĐỒ HOẠT ĐỘNG “XÂY DỰNG LỘ TRÌNH ĐIỀU TRỊ” CỦA BÁC SĨ.....	22
HÌNH 3.12: SƠ ĐỒ HOẠT ĐỘNG “ĐẶT LỊCH KHÁM BỆNH” CỦA BỆNH NHÂN.....	23
HÌNH 3.13: SƠ ĐỒ ERD	24
HÌNH 4.1: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “XÁC NHẬN PHIẾU HẸN”	25
HÌNH 4.2: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “THỐNG KÊ TÌNH HÌNH KHÁM BỆNH” 1	25
HÌNH 4.3: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “THỐNG KÊ TÌNH HÌNH KHÁM BỆNH” 2	26
HÌNH 4.4: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “TÌM KIẾM BỆNH NHÂN”	26
HÌNH 4.5: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “PHÊ DUYỆT LỊCH HẸN” 1	27
HÌNH 4.6: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “PHÊ DUYỆT LỊCH HẸN” 2	27
HÌNH 4.7: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “XÂY DỰNG LỘ TRÌNH ĐIỀU TRỊ” 1	28
HÌNH 4.8: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “XÂY DỰNG LỘ TRÌNH ĐIỀU TRỊ” 2.....	28
HÌNH 4.9: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “XÂY DỰNG LỘ TRÌNH ĐIỀU TRỊ” 3.....	29
HÌNH 4.10: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “XÂY DỰNG LỘ TRÌNH ĐIỀU TRỊ” 4.....	29
HÌNH 4.11: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “XÂY DỰNG LỘ TRÌNH ĐIỀU TRỊ” 5.....	29
HÌNH 4.12: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “ĐĂNG KÝ TÀI KHOẢN KHÁM BỆNH” 1	30
HÌNH 4.13: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “ĐĂNG KÝ TÀI KHOẢN KHÁM BỆNH” 2	30
HÌNH 4.14: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “ĐĂNG KÝ TÀI KHOẢN KHÁM BỆNH” 3	31

HÌNH 4.15: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “ĐẶT LỊCH KHÁM BỆNH” 1	31
HÌNH 4.16: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “ĐẶT LỊCH KHÁM BỆNH” 2	32
HÌNH 4.17: GIAO DIỆN TRIỂN KHAI CHỨC NĂNG “ĐẶT LỊCH KHÁM BỆNH” 3	32
.....	
BẢNG 3.1:...KỊCH BẢN NHÂN VIÊN ĐÓN TIẾP XÁC NHẬN PHIẾU HẸN KHI BỆNH NHÂN ĐẾN KHÁM	16
BẢNG 3.2: KỊCH BẢN BÁC SĨ PHÊ DUYỆT LỊCH HẸN KHÁM DO BỆNH NHÂN ĐẶT	17
BẢNG 3.3: KỊCH BẢN BÁC SĨ LÊN KẾ HOẠCH ĐIỀU TRỊ CHO BỆNH NHÂN	18
BẢNG 3.4: KỊCH BẢN BỆNH NHÂN ĐẶT LỊCH KHÁM BỆNH.....	19

CHƯƠNG 1: TỔNG QUAN

1.1. Phân tích hiện trạng

- Ngày nay, nhu cầu khám chữa bệnh từ xa, tại nhà đang có nhu cầu cao và thật sự cần thiết trong mùa dịch covid-19.
- Việc phải ngồi đợi cho đến lượt mỗi lần đi khám bệnh mất rất nhiều thời gian, nhiều người vì điều này mà đã lười đi khám bệnh, đến khi bệnh nặng rồi mới đi khám.
- Không chỉ có vậy, việc đi khám bệnh chờ đợi quá đông sẽ dẫn đến nguy cơ lây nhiễm covid-19 rất cao.
- Bên cạnh đó là cuộc cách mạng công nghiệp 4.0, hướng đến thành phố thông minh.

1.2. Lí do chọn đề tài

- Do nhu cầu khám chữa bệnh tại nhà đang rất cấp thiết mà tại Đà Nẵng chưa có ứng dụng nào có thể hoạt động tốt.
- Do thực trạng việc khám bệnh tại bệnh viện mất quá nhiều thời gian chờ đợi.

1.3. Mục đích đề tài

Xây dựng ứng dụng đặt lịch và khám bệnh online dành cho bệnh viện Đa khoa Đà Nẵng giúp:

- Đặt lịch khám chữa bệnh tại bệnh viện.
- Khám bệnh trực tiếp bằng Video Call.
- Liên hệ với bác sĩ bằng cách nhắn tin, gọi video trực tiếp trên ứng dụng.
- Bệnh nhân có thể xem kết quả khám bệnh trực tiếp trên ứng dụng (sau mỗi lần khám).
- Tạo tài khoản người bệnh, sổ theo dõi khám bệnh.
- Theo dõi được lịch trình khám bệnh, điều trị bệnh.

1.4. Phạm vi đề tài

Áp dụng tại bệnh viện Đa khoa Đà Nẵng

1.5. Đối tượng nghiên cứu

- Người khám bệnh
- Bác sĩ

1.6. Phương pháp nghiên cứu

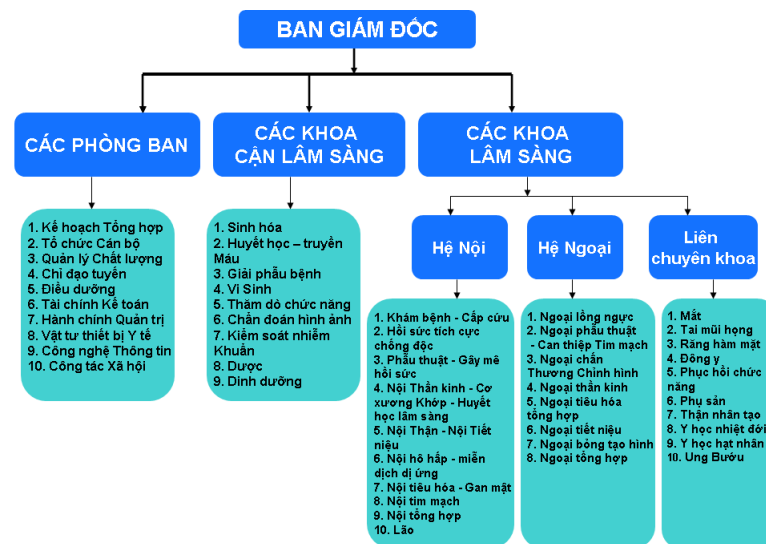
Đề tài được nghiên cứu thông qua thứ tự các phương pháp sau:

- Khảo sát và Phân tích bài toán
- Lên ý tưởng giải quyết bài toán
- Nghiên cứu các công nghệ sẽ sử dụng
- Phân tích thiết kế hệ thống
- Xây dựng hệ thống
- Thử nghiệm và Ứng dụng

1.7. Công nghệ sử dụng

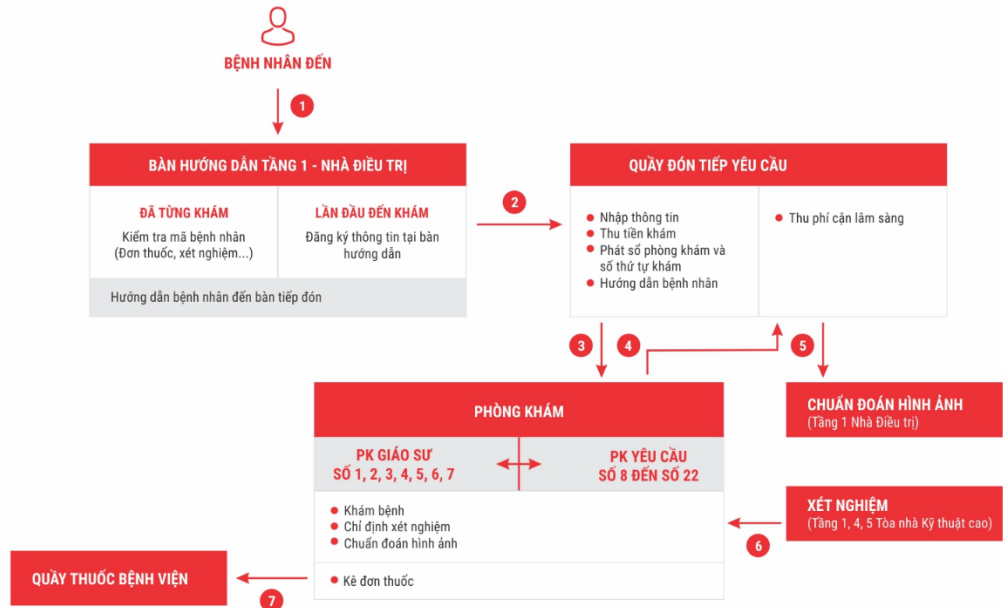
- Hệ quản trị CSDL mySql
- Nền tảng, ngôn ngữ lập trình: Bootstrap, Nodejs, WebRTC, PeerJs, SocketIO

1.8. Cơ cấu tổ chức

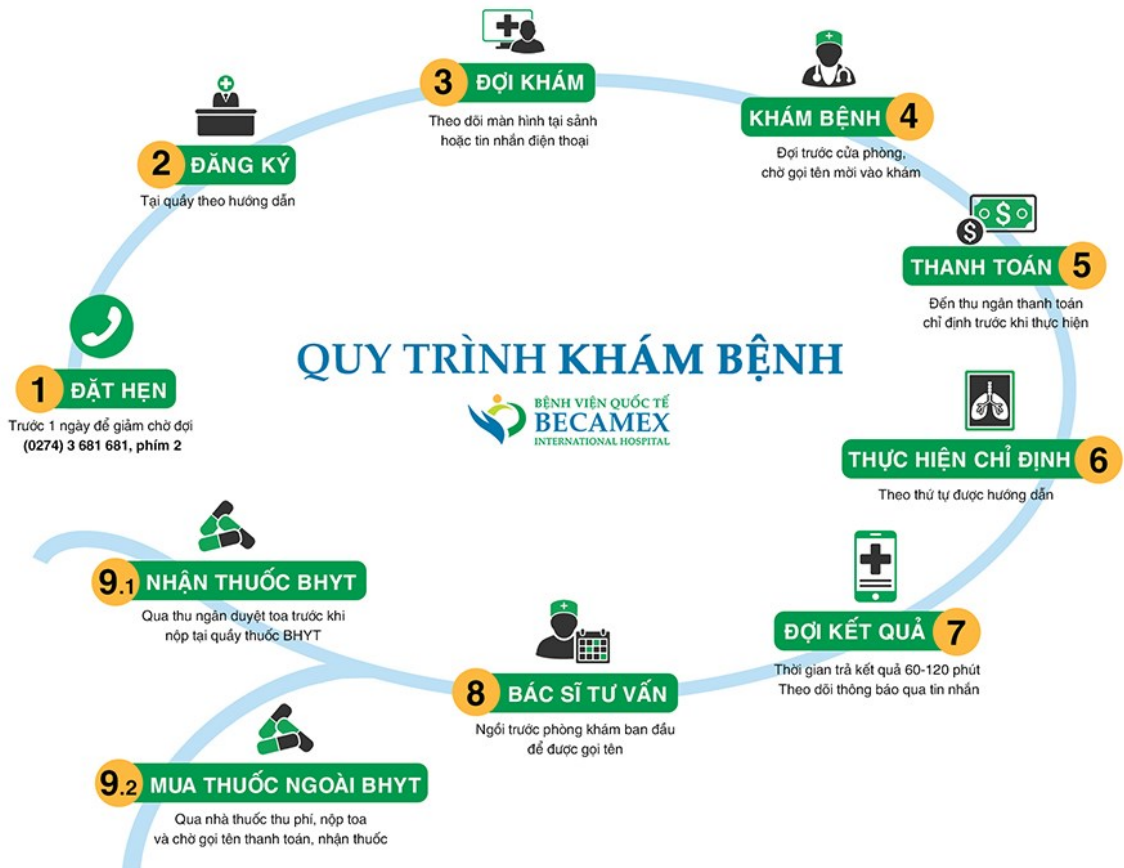


Hình 1.1: Cơ cấu tổ chức của bệnh viện Đa khoa Đà Nẵng

1.9. Hoạt động nghiệp vụ thực tế



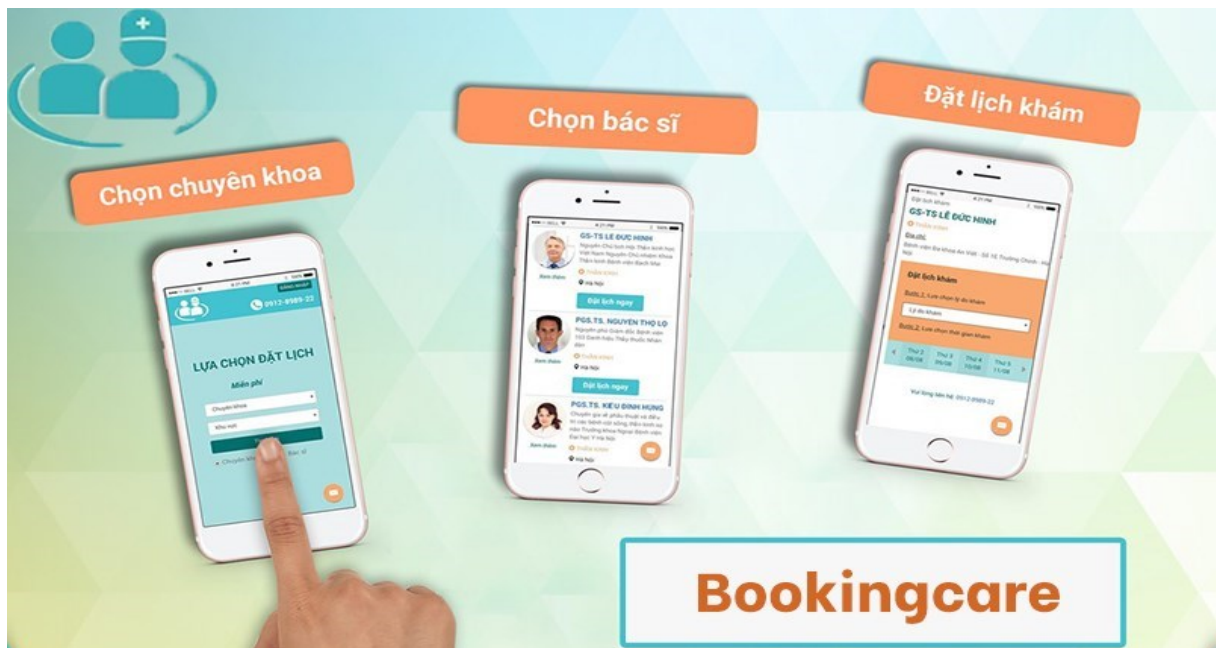
Hình 1.2: Quy trình khám chữa bệnh tại bệnh viện



Hình 1.3: Quy trình khám chữa bệnh đặt lịch trước ngoài thực tế

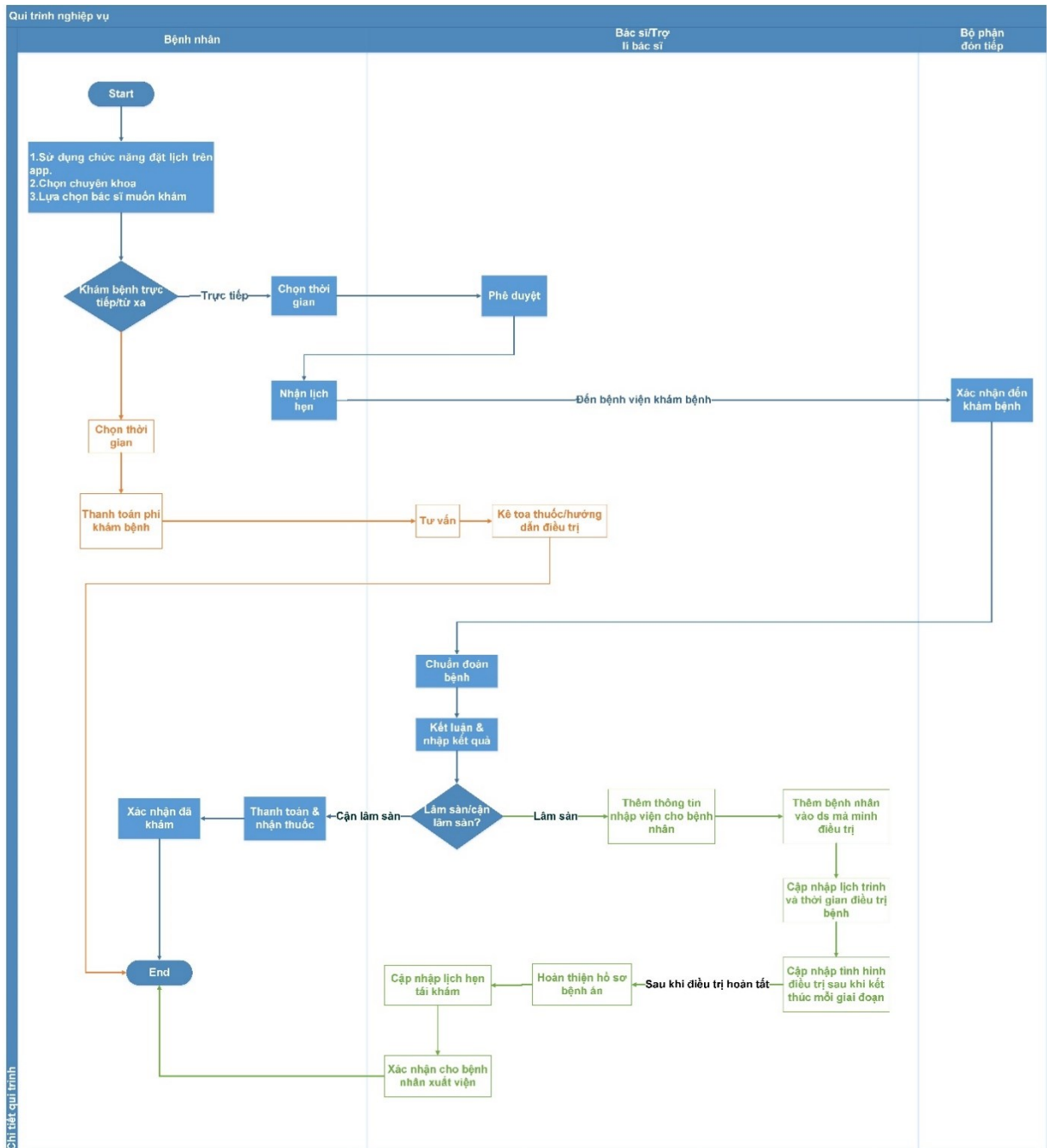


Hình 1.4: Quy trình các bước nhập viện



Hình 1.5: Quy trình tư vấn khám bệnh từ xa của ứng dụng Bookingcare

1.10. Quy trình nghiệp vụ của đề tài



Hình 1.6: Quy trình nghiệp vụ của đề tài

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Phần mềm

2.1.1. Visual Studio Code

Là một trình biên tập lập trình code miễn phí dành cho Windows, Linux và macOS, Visual Studio Code được phát triển bởi Microsoft. Nó được xem là một sự kết hợp hoàn hảo giữa IDE và Code Editor.

Visual Studio Code hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép người dùng thay đổi theme, phím tắt, và các tùy chọn khác.

Một số tính năng nổi bật:

- Hỗ trợ nhiều ngôn ngữ lập trình
- Hỗ trợ đa nền tảng
- Ít dung lượng, nhẹ
- Tính năng mạnh mẽ
- Cung cấp kho tiện ích mở rộng
- Kho lưu trữ an toàn
- Visual Studio Code hỗ trợ nhiều ứng dụng web. Ngoài ra, nó cũng có một trình soạn thảo và thiết kế website.
- Lưu trữ dữ liệu dạng phân cấp
- Hỗ trợ viết Code
- Visual Studio Code có tích hợp thiết bị đầu cuối, giúp người dùng khỏi phải chuyển đổi giữa hai màn hình hoặc trở về thư mục gốc khi thực hiện các thao tác
- Người dùng Visual Studio Code có thể mở cùng lúc nhiều tệp tin và thư mục – mặc dù chúng không hề liên quan với nhau.
- Intellisense: có thể phát hiện nếu bất kỳ đoạn mã nào không đầy đủ. Thậm chí, khi lập trình viên quên không khai báo biến, Intellisense sẽ tự động giúp họ bổ sung các cú pháp còn thiếu.
- Hỗ trợ Git

2.1.2. phpMyAdmin

PhpMyAdmin là phần mềm mã nguồn mở được viết bằng ngôn ngữ PHP giúp quản trị cơ sở dữ liệu MySQL thông qua giao diện web. Tính đến nay, phpMyAdmin đã có đến hàng triệu lượt sử dụng và vẫn không ngừng tăng. Vậy tính năng hữu ích mà phpMyAdmin mang lại là gì:

- Quản lý user(người dùng): thêm, xóa, sửa(phân quyền).
- Quản lý cơ sở dữ liệu: tạo mới, xóa, sửa, thêm bảng, hàng, trường, tìm kiếm đối tượng.
- Nhập xuất dữ liệu(Import/Export): hỗ trợ các định dạng SQL, XML và CSV.
- Thực hiện các truy vấn MySQL, giám sát quá trình và theo dõi.
- Sao lưu và khôi phục(Backup/Restore): Thao tác thủ công.

Bên cạnh việc cung cấp nhiều tính năng cần thiết như đã đề cập, phpMyAdmin còn có thể vừa làm việc với một đối tượng vừa xử lý các tình huống bất ngờ. Một vài ví dụ kể đến như SQL injection, các vấn đề phát sinh, lỗi database...

Dù có nhiều ưu điểm song phpMyAdmin vẫn khó tránh khỏi một vài điểm yếu cố hữu. Đặc biệt, trong việc sao lưu dữ liệu thủ công sẽ không có một vài tính năng cần thiết như : Scheduling, Storage Media Support.

2.1.3. Công cụ Postman

Postman là một công cụ cho phép chúng ta thao tác với API, phổ biến nhất là REST. Postman hiện là một trong những công cụ phổ biến nhất được sử dụng trong thử nghiệm các API. Với Postman, ta có thể gọi Rest API mà không cần viết dòng code nào.

Postman hỗ trợ tất cả các phương thức HTTP (GET, POST, PUT, PATCH, DELETE, ...). Bên cạnh đó, nó còn cho phép lưu lại lịch sử các lần request, rất tiện cho việc sử dụng lại khi cần.

Những lợi ích khi sử dụng Postman:

- Sử dụng Collections (Bộ sưu tập) – Postman cho phép người dùng tạo bộ sưu tập cho các lệnh gọi API của họ. Mỗi bộ sưu tập có thể tạo các thư mục con và nhiều yêu cầu (request). Điều này giúp việc tổ chức các bộ thử nghiệm.
- Collaboration – Collections và environment có thể được import hoặc export giúp chia sẻ tệp dễ dàng.
- API Testing – Test trạng thái phản hồi HTTP.

- Gỡ lỗi – Bảng điều khiển Postman giúp kiểm tra dữ liệu nào đã được truy xuất giúp dễ dàng gỡ lỗi kiểm tra.
- Các chức năng cơ bản
- Cho phép gửi HTTP Request với các method GET, POST, PUT, DELETE.
- Cho phép post dữ liệu dưới dạng form (key-value), text, json.
- Hiện kết quả trả về dạng text, hình ảnh, XML, JSON.
- Hỗ trợ authorization (Oauth1, 2).
- Cho phép thay đổi header của các request.

2.2. Công nghệ

2.2.1. Công nghệ Bootstrap

Thiết kế một giao diện web đẹp, trực quan không phải là điều dễ dàng với hầu hết lập trình viên. Tuy giao diện không phải là thành phần quan trọng nhất của một ứng dụng, nhưng nếu giao diện quá khó coi thì sẽ gây mất cảm tình với người dùng. Lập trình viên nào cũng muốn ứng dụng trông thật bắt mắt, nhưng họ lại không muốn tốn nhiều thời gian và công sức để thiết kế giao diện. Do vậy, Bootstrap được xem như là vị cứu tinh trong tình huống khó xử này.

Bootstrap là một framework nổi tiếng trong cộng đồng lập trình web, và cũng là dự án được theo dõi nhiều nhất trên Github. Bootstrap không đòi hỏi người dùng phải có kiến thức về thiết kế hoặc mỹ thuật để tạo ra một giao diện đẹp. Mọi thứ trong Bootstrap đều đã được xây dựng sẵn. Chỉ cần dùng những thành phần (component) có sẵn này cho mục đích riêng của mình. Bootstrap khởi đầu là một dự án nội bộ của Twitter. Sau khi nhận ra được tính hữu ích của nó đối với cộng đồng, nhóm lập trình viên của Twitter quyết định mở nguồn cho Bootstrap và đăng tải nó lên Github vào năm 2011. Đến nay, Bootstrap đã được 5 tuổi và hiện đang ở phiên bản 3, và sẽ ra mắt phiên bản 4 trong thời gian sắp tới.

Bất kỳ ai cũng có thể dùng Bootstrap, kể cả khi người không có khiếu thẩm mỹ hoặc không rành về CSS. Nếu có tay nghề cao, có thể tùy biến Bootstrap theo ý thích riêng, hoặc tự tạo theme dựa trên Bootstrap. Nếu không chuyên về công nghệ front-end, hãy dùng kỹ thuật copy và paste đoạn code ví dụ có sẵn trên documentation của Bootstrap. Tuy nhiên, đây chỉ là phương thức chữa cháy lúc mới bắt đầu học mà thôi. Sau này, nên nâng cao hiểu biết về Bootstrap để không phải phụ thuộc code mẫu nữa.

Bootstrap được cấu tạo từ các file CSS và JavaScript. Để dùng Bootstrap, bạn chỉ cần thêm link đến những file này trong HTML. Sau đó, hãy xem qua các component

có trong documentation, rồi chọn ra component ưng ý. Kế đến, copy những đoạn code minh họa và paste vào file HTML.

Mặc định, các thành phần trong Bootstrap đều responsive, nghĩa là chúng tự co giãn cho phù hợp với kích thước màn hình. Trước đây, khi thiết kế một trang web, người ta phải tạo ra nhiều phiên bản khác nhau. Sau đó, lập trình viên dùng một đoạn code để phát hiện thiết bị của người dùng. Đoạn code sẽ chuyển hướng sang trang web thiết kế riêng cho thiết bị đó. Phương pháp này có nhiều điểm bất lợi, như khi cần cập nhật web thì phải sửa tất cả các phiên bản. Do vậy, responsive đang trở thành xu thế chủ đạo trong thiết kế web, và Bootstrap giúp thực hiện điều này một cách tự động.

Giống như Windows Forms giúp thiết kế giao diện ứng dụng desktop, Bootstrap cung cấp nhiều component để thiết kế giao diện web. Đây là các component HTML đã được Bootstrap định kiểu (style) sẵn. Nhờ vậy, lập trình viên có thể dùng ngay mà không phải viết code phức tạp. Ngoài ra, có thể tùy biến các style một cách dễ dàng. Nếu không có khả năng thiết kế, nên dùng theme nguồn mở của Bootstrap để thay đổi kiểu dáng các component.

Trước đây, khi thiết kế web, người ta thường dùng `<table>` để dàn trang. Cách này có khuyết điểm là khiến trang web tải chậm, đặc biệt khi `<table>` lồng vào nhau. Ngoài ra, `<table>` không phải là công cụ dùng để dàn trang. Nó chỉ dùng để hiển thị thông tin dạng bảng. Do đó, dùng `<table>` để dàn trang là sai phương pháp. Bootstrap sử dụng các thẻ `<div>` để dàn trang, cho nên khả năng responsive của nó rất mạnh mẽ. Mặc định, Bootstrap sử dụng một khung lưới (grid) gồm 12 cột. Khi thêm component, bạn sẽ cung cấp số cột mà component chiếm. Từ đó, Bootstrap sẽ tự động canh chỉnh cho phù hợp.

Vì Bootstrap được dùng rộng rãi, cộng đồng nguồn mở cung cấp một loạt các theme miễn phí nếu không muốn chỉnh style cho từng component. Để thay đổi theme, vào trang bootswatch.com. Ở đây, có thể chọn theme theo từng chủ đề với nhiều phong cách khác nhau. Để thay đổi theme, chỉ cần đổi file CSS của Bootstrap thành file CSS lấy từ trang bootswatch.com. Sau đó refresh trình duyệt để xem kết quả.

2.2.2. Công nghệ NodeJS

✓ Giới thiệu

Nodejs là một nền tảng (Platform) phát triển độc lập được xây dựng ở trên Javascript Runtime của Chrome mà chúng ta có thể xây dựng được các ứng dụng mạng một cách nhanh chóng và dễ dàng mở rộng.

Nodejs được xây dựng và phát triển từ năm 2009, bảo trợ bởi công ty Joyent, trụ sở tại California, Hoa Kỳ.

Phần Core bên dưới của Nodejs được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao. Nodejs tạo ra được các ứng dụng có tốc độ xử lý nhanh, realtime thời gian thực.

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

Lí do sử dụng NodeJs:

- Các ứng dụng Nodejs được viết bằng javascript, ngôn ngữ này là một ngôn ngữ khá thông dụng.
- Nodejs chạy đa nền tảng phía Server, sử dụng kiến trúc hướng sự kiện Event-driven, cơ chế non-blocking I/O làm cho nó nhẹ và hiệu quả.
- Có thể chạy ứng dụng Nodejs ở bất kỳ đâu trên máy Mac – Window – Linux, hơn nữa cộng đồng Nodejs rất lớn và hoàn toàn miễn phí. Các bạn có thể thấy cộng đồng Nodejs lớn như thế nào tại đây, các package đều hoàn toàn free: <https://www.npmjs.com/>
- Các ứng dụng NodeJS đáp ứng tốt thời gian thực và chạy đa nền tảng, đa thiết bị.
- Có Framework hỗ trợ tạo web server cực nhanh: Express.js
- Với thư viện Socket.IO cực kỳ nổi tiếng trong việc tạo các ứng dụng Web cho phép chạy Runtime như: Stream Video,....

✓ ExpressJS Framework

Expressjs là một framework được xây dựng trên nền tảng của Nodejs. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile. Expressjs hỗ trợ các method HTTP và middleware tạo ra API vô cùng mạnh mẽ và dễ sử dụng. Dưới đây là các tính năng cơ bản của Express framework:

- Cho phép thiết lập các lớp trung gian để trả về các HTTP request.
- Định nghĩa bảng routing có thể được sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL.
- Cho phép trả về các trang HTML dựa vào các tham số truyền vào đến template.

Các khái niệm cơ bản:

- Định tuyến Routing: Định tuyến liên quan đến cách xác định một ứng dụng trả về cho một Client Request đến một Endpoint cụ thể, đó là một đường dẫn URI

và trả về một phương thức HTTP request (GET, POST và các phương thức khác).

- Đường dẫn tĩnh (Static): Express cung cấp các tiện ích lớp trung gian `express.static` để phục vụ cho các file tĩnh như hình ảnh, CSS, Javascript,... Để các template sử dụng chúng một cách trực tiếp.
- `body-parser` là một module để xử lý JSON, dữ liệu thô, text và mã hóa URL, truyền dữ liệu giữa giao diện và Controller.
- `multer` là một module dùng để upload hình ảnh lên hệ thống.
- Đối tượng Request: biểu diễn một HTTP request và có các thuộc tính cho các request như các chuỗi truy vấn, tham số, body, HTTP header và những phần khác. Hay hiểu một cách đơn giản là những yêu cầu từ phía người dùng.
- Đối tượng Response: biểu diễn HTTP response được ứng dụng Express gửi đi khi nó nhận về một HTTP request. Hay hiểu đơn giản là gửi dữ liệu từ server đến người dùng, trả lời Request của người dùng.

✓ SocketIO

Để xây dựng một ứng dụng realtime cần sử dụng socketio. Socketio sẽ giúp các bên ở những địa điểm khác nhau kết nối với nhau, truyền dữ liệu ngay lập tức thông qua server trung gian. Socketio có thể được sử dụng trong nhiều ứng dụng như chat, game online, cập nhật kết quả của một trận đấu đang xảy ra,...

Socketio không phải là một ngôn ngữ, mà chỉ là 1 công cụ giúp thực hiện những ứng dụng realtime. Vì thế, không thể sử dụng socketio để thay thế hoàn toàn cho một ngôn ngữ, mà phải sử dụng kết hợp với một ngôn ngữ khác. Ngôn ngữ đó có thể là php, asp.net, nodejs,...

2.2.3. Công nghệ WebRTC

WebRTC được viết tắt từ cụm “Web realtime communication” và được hiểu là giao tiếp với website theo thời gian thực. Nó bao gồm các hàm lập trình (web API) và được viết bằng JavaScript cho phép mọi người trên website trò chuyện trực tiếp thông qua video trên tất cả các trình duyệt.

Các ứng dụng học trực tuyến Zoom hoặc tính năng phát video trực tiếp trên Facebook hoặc Youtube chẳng hạn. Đó chính là những trang web và ứng dụng tích hợp công nghệ WebRTC.

WebRTC là một mã nguồn mở được duy trì bởi nhóm Google WebRTC dưới sự hỗ trợ của Apple, Google, Microsoft và Mozilla cùng những ông lớn trong lĩnh vực công nghệ khác. Nó mang đến rất nhiều lợi ích như sau:

- **Thứ nhất**, WebRTC cho phép người dùng gọi điện video và trò chuyện trực tiếp với nhau thông qua trình duyệt web
- **Thứ hai**, WebRTC cho phép người dùng chia sẻ các file dữ liệu, hình ảnh, video trong thời gian giao tiếp thực. Ví dụ như chức năng share màn hình hoặc gửi file của Zoom
- **Thứ ba**, WebRTC cho phép bạn có thể tổ chức các buổi hội thảo trực tuyến với đồng thành viên cùng giao tiếp với nhau. Để dễ hiểu thì bạn có thể hình dung nó giống như chức năng họp online của ứng dụng Google meet vậy.
- Bên cạnh đó, WebRTC còn cho phép người dùng tạo ra và chơi các game trực tuyến ứng với thời gian thực.

Ưu điểm:

- WebRTC là thân thiện với tất cả các hệ điều hành và ngôn ngữ lập trình.
- WebRTC miễn phí
- WebRTC thân thiện với tất cả các thiết bị
- Dùng WebRTC không cần các ứng dụng hoặc Plugin hỗ trợ
- WebRTC có tính bảo mật cao WebRTC giúp tiết kiệm chi phí đi lại, giao tiếp do người dùng có thể kết nối trao đổi trực tiếp qua internet

Nhược điểm:

- Khi cố gắng thực hiện kết nối P2P (Peer to peer - hình thức giao tiếp giữa các máy tính với nhau không thông qua máy chủ) WebRTC có thể bị cản bởi tường lửa (firewall) và NAT.
- Với mỗi trình duyệt WebRTC hỗ trợ các hàm API khác nhau nên có khả năng phát sinh lỗi khi kết nối giữa 2 máy khác trình duyệt.
- Video truyền giữa 2 máy chưa được các hàng trình duyệt thống nhất nên video trên WebRTC có thể có chất lượng khác nhau

2.2.4. PeerJs

PeerJS là một thư viện JavaScript hoạt động như một trình bao bọc cho WebRTC và cho phép bạn tạo các kết nối ngang hàng, các cuộc gọi rất dễ dàng. Với PeerJS, chúng ta có thể tạo kết nối như vậy chỉ trong 3 dòng mã.

PeerJS giao dịch với bắt tay WebRTC và cho phép kết nối bằng ID ngang hàng. Tuy nhiên, để thiết lập kết nối, thì cần có PeerServer. Điều này được viết bằng Node.js và cần thiết lập nó.

CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

3.1. Đặc tả yêu cầu phần mềm (Software Requirements)

3.1.1. Nhân viên bộ phận đón tiếp

- ✓ Đăng nhập
- ✓ Cập nhập
 - Xác nhận phiếu hẹn (khi bệnh nhân đến khám)
- ✓ Xem
 - Lịch đặt khám bệnh của bệnh viện
 - Hồ sơ sức khỏe bệnh nhân (Hồ sơ, tiền sử bệnh, bệnh đang điều trị, lịch sử sức khỏe, lịch sử khám bệnh)
- ✓ Tìm kiếm
 - Lịch khám của bệnh nhân theo: CMND/CCID
 - Bệnh nhân: CMND/CCID
- ✓ Lọc
 - Lịch khám bệnh (Theo ngày cụ thể, hôm nay, ngày mai, tuần này, tháng này)
- ✓ Thống kê
 - Tình hình khám bệnh

3.1.2. Bác sĩ/Trợ lý bác sĩ

Kế thừa Nhân viên bộ phận đón tiếp các tác nhân: Đăng nhập, xem hồ sơ sức khỏe của bệnh nhân, tìm kiếm bệnh nhân

- ✓ Cập nhập
 - Phê duyệt lịch hẹn
 - Danh sách bệnh nhân đang được mình điều trị
 - Chat và video call với bệnh nhân
 - Kết quả khám bệnh
 - Thuốc điều trị, ghi chú liều lượng sử dụng thuốc và những điều cần lưu ý
 - Kế hoạch và quá trình điều trị bệnh
 - Hồ sơ sức khỏe cho bệnh nhân
 - Thông tin cá nhân

- Hoàn thành kế hoạch điều trị
- ✓ Xem
 - Lịch điều trị bệnh
- ✓ Lọc
 - Lịch điều trị bệnh

3.1.3. Bệnh nhân

Kế thừa Bác sĩ/Trợ lý bác sĩ các tác nhân: Đăng nhập, cập nhập thông tin cá nhân, lọc lịch khám bệnh

- ✓ Cập nhập
 - Thành viên trong gia đình
 - Đặt lịch khám bệnh (theo chuyên khoa)
 - Hủy hẹn (khi chưa được xác nhận)
 - Chat và video call với bác sĩ
- ✓ Xem
 - Lịch khám bệnh
 - Kế hoạch điều trị
 - Kết quả khám bệnh
 - Lịch sử khám bệnh
 - Thông tin giới thiệu của bác sĩ

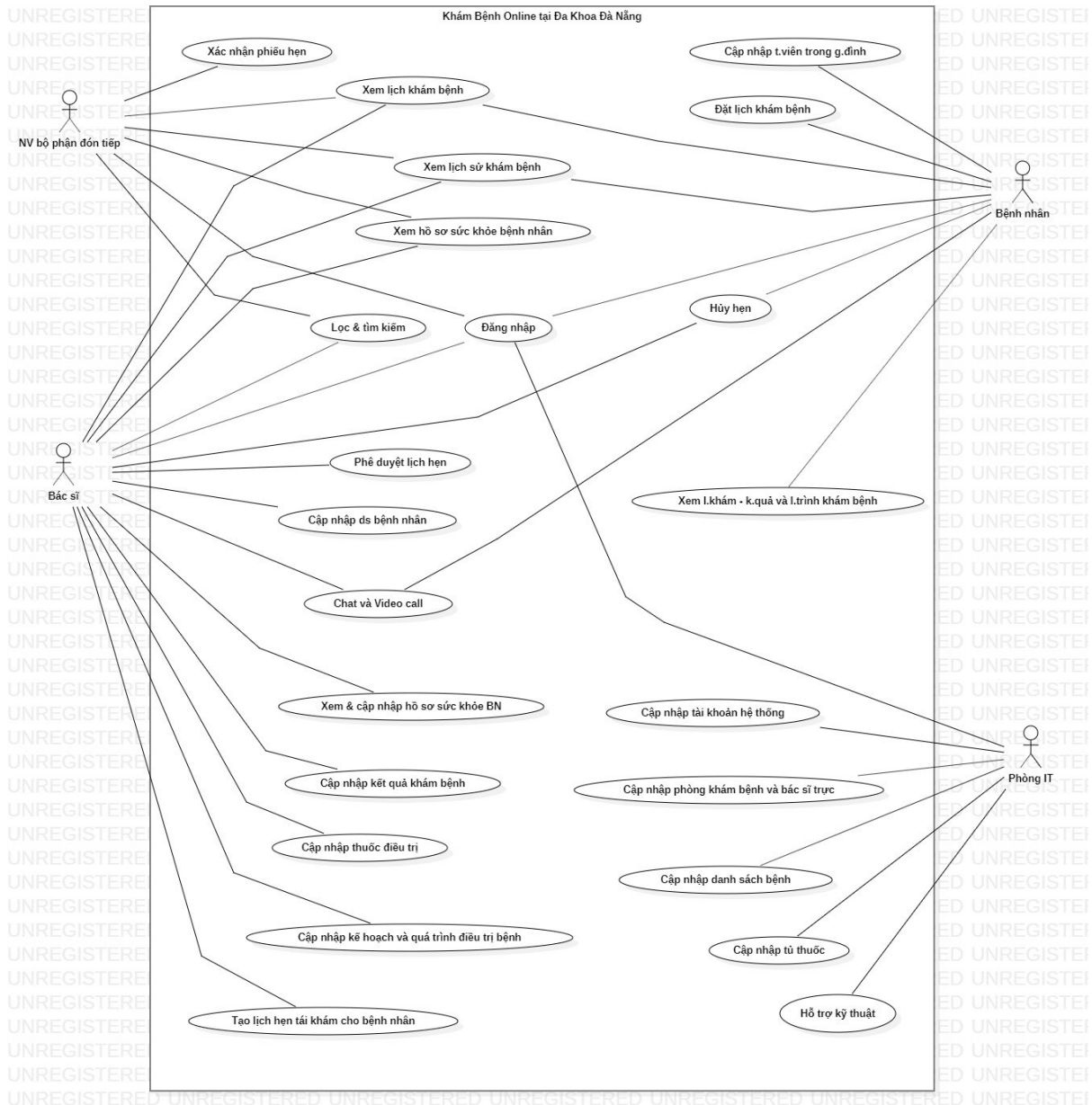
3.1.4. Phòng IT

Kế thừa Nhân viên đón tiếp các tác nhân: Đăng nhập, cập nhập thông tin cá nhân, lọc lịch khám bệnh

- ✓ Cập nhập
 - Tài khoản hệ thống
 - Phòng khám bệnh và bác sĩ trực
 - Danh sách bệnh
 - Tủ thuốc
 - Thông tin bệnh viện
 - Hỗ trợ kỹ thuật qua cửa sổ chat
- ✓ Xem
 - Thông tin tài khoản hệ thống
- ✓ Tìm kiếm

- Tài khoản (theo tên, cmnd/ccid)
- ✓ **Lọc**
- Theo chức vụ (Bác sĩ/nhân viên/bệnh nhân)

3.2. Sơ đồ Usecase



Hình 3.1: Sơ đồ Usecase

3.3. Kịch bản hoạt động

3.3.1. Nhân viên bộ phận đón tiếp

- ✓ **Xác nhận phiếu hẹn**

Bảng 3.1: Kịch bản nhân viên đón tiếp xác nhận phiếu hẹn khi bệnh nhân đến khám

STT	Usecase name	Order
1.	Description	Nhân viên bộ phận đón tiếp xác nhận là bệnh nhân có đến khám, lưu lại thời gian đến khám thực tế
2.	Actors	Nhân viên bộ phận đón tiếp
3.	Input	Nhân viên đón tiếp và bệnh nhân đăng nhập thành công Bệnh nhân cung cấp QR code cho nhân viên quét để xác nhận
4.	Output	Hệ thống lưu lại thời gian đến khám bệnh thực tế Thông báo đã xác nhận lịch hẹn.
5.	Basic flow	<ol style="list-style-type: none"> Actors truy cập quét QR trên thanh menu (giao diện 1) – Bắt đầu use case Actors quét QR lịch hẹn của bệnh nhân đưa cho (giao diện 2) Hệ thống lưu lại thời gian đến khám và lịch hẹn chuyển sang trạng thái “Chờ khám” – Kết thúc use case
6.	Alternative flow (nêu các bước phát sinh trong Basic flow)	<ol style="list-style-type: none"> 2.1. Nếu mã QR không chính xác, thì yêu cầu bệnh nhân kiểm tra lại. 2.2. Tiến hành quét lại lần nữa
7.	Exception flow (Nêu các trường hợp ngoại lệ của Basic flow)	<ol style="list-style-type: none"> 2.2.1. Nếu vẫn không chính xác, thì bệnh nhân đặt lại lịch hẹn khác

3.3.2. Bác sĩ/Trợ lý bác sĩ

✓ Phê duyệt lịch hẹn

Bảng 3.2: Kịch bản bác sĩ phê duyệt lịch hẹn khám do bệnh nhân đặt

STT	Usecase name	Order
1.	Description	Bệnh nhân đặt lịch khám bệnh, bác sĩ là người phê duyệt hoặc từ chối lịch hẹn khám bệnh đó
2.	Actors	Bác sĩ/Trợ lý bác sĩ
3.	Input	Bệnh nhân tiến hành đặt lịch hẹn khám bệnh
4.	Output	Lịch hẹn được bác sĩ phê duyệt

5.	Basic flow	1. Actors truy cập chức năng quản lý lịch hẹn (giao diện 1) – Bắt đầu use case 2. Actors xem lịch hẹn chưa được phê duyệt (giao diện 2) 3. Actors nhấn phê duyệt lịch hẹn (giao diện 3) 4. Hệ thống lưu lại trạng thái của lịch hẹn và thông báo ra màn hình lịch hẹn khám bệnh đã được phê duyệt – Kết thúc use case
6.	Alternative flow (nêu các bước phát sinh trong Basic flow)	3.1. Actors từ chối lịch hẹn, do bận hoặc lí do nào đó không thể khám bệnh cho bệnh nhân
7.	Exception flow (Nêu các trường hợp ngoại lệ của Basic flow)	

✓ Xây dựng lộ trình điều trị

Bảng 3.3: Kịch bản bác sĩ lên kế hoạch điều trị cho bệnh nhân

STT	Usecase name	Order
1.	Description	Bác sĩ lên kế hoạch điều trị một bệnh cụ thể nào đó cho bệnh nhân
2.	Actors	Bác sĩ/Trợ lí bác sĩ
3.	Input	Mã bệnh nhân/Tên bệnh nhân
4.	Output	Danh sách chi tiết kế hoạch điều trị cho bệnh nhân
5.	Basic flow	1. Actors tìm bệnh nhân trên hệ thống thông qua tên hoặc mã bệnh nhân (giao diện 1) – Bắt đầu use case 2. Actors lựa chọn thêm bệnh sẽ điều trị(giao diện 2) 3. Actors nhập thông tin điều trị bệnh(giao diện 3) 4. Actors tiến hành lên các kế hoạch điều trị cho bệnh vừa thêm(giao diện 4) 5. Hệ thống lưu lại kế hoạch điều trị bệnh và bệnh nhân có thể xem kế hoạch điều trị của mình và lịch tái khám ở từng giai đoạn – Kết thúc use case
6.	Alternative flow	

	(nêu các bước phát sinh trong Basic flow)	
7.	Exception flow (Nêu các trường hợp ngoại lệ của Basic flow)	

3.3.3. Bệnh nhân

✓ Đặt lịch khám bệnh

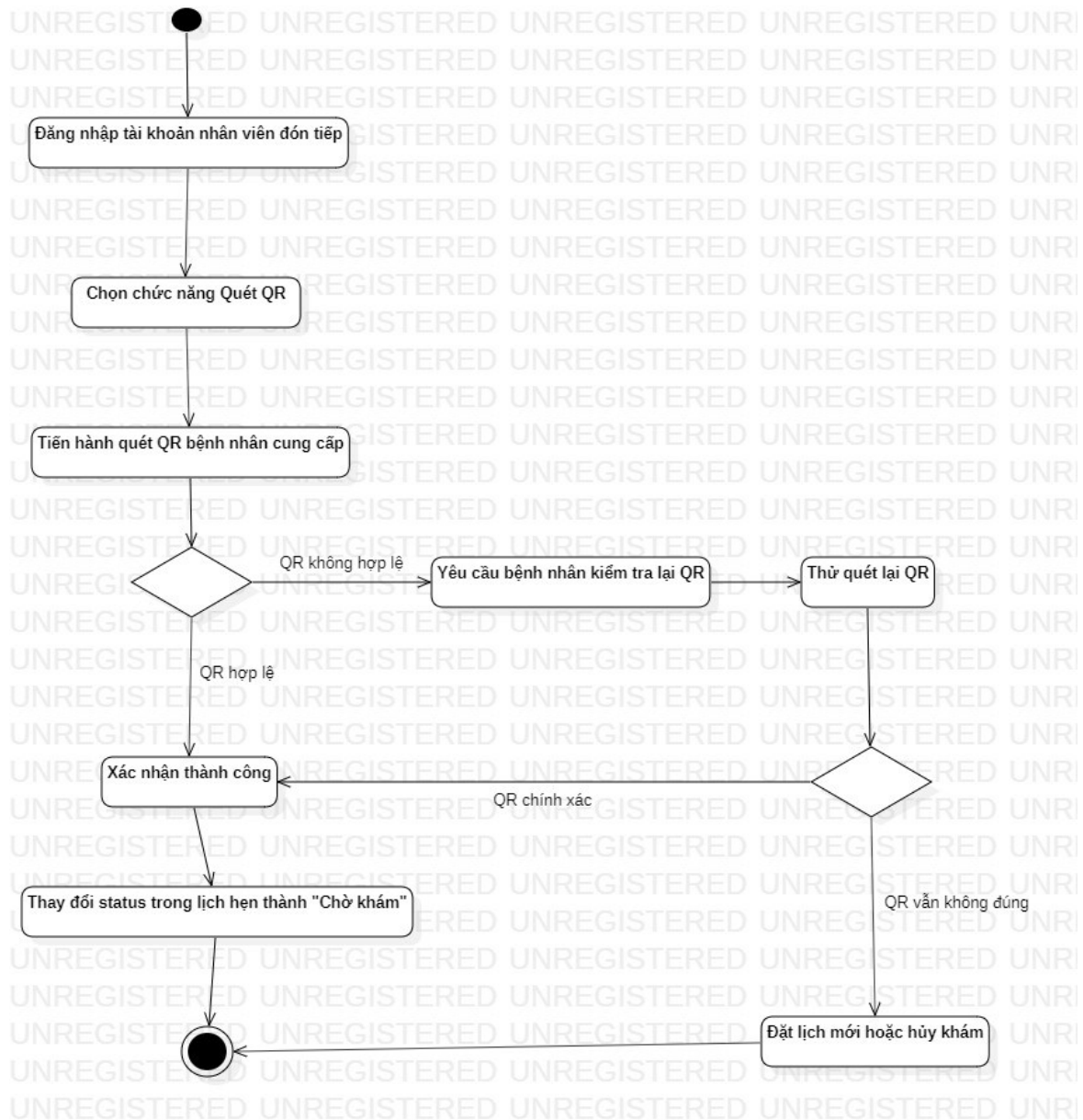
Bảng 3.4: Kịch bản bệnh nhân đặt lịch khám bệnh

STT	Usecase name	Order
1.	Description	Bệnh nhân đặt lịch khám bệnh
2.	Actors	Bệnh nhân
3.	Input	Đăng nhập bằng tài khoản Bệnh nhân
4.	Output	Lịch hẹn khám bệnh
5.	Basic flow	1. Actors chọn chức năng “Phòng khám” (giao diện 1) – Bắt đầu use case 2. Actors lựa chọn khoa muốn khám (giao diện 2) 3. Actors lựa chọn bác sĩ muốn khám (giao diện 3) 4. Actors tiến hành đặt lịch hẹn theo khung giờ (giao diện 4) 5. Hệ thống lưu lại lịch hẹn chờ bác sĩ – Kết thúc use case
6.	Alternative flow (nêu các bước phát sinh trong Basic flow)	
7.	Exception flow (Nêu các trường hợp ngoại lệ của Basic flow)	

3.4. Sơ đồ hoạt động

3.4.1. Nhân viên bộ phận đón tiếp

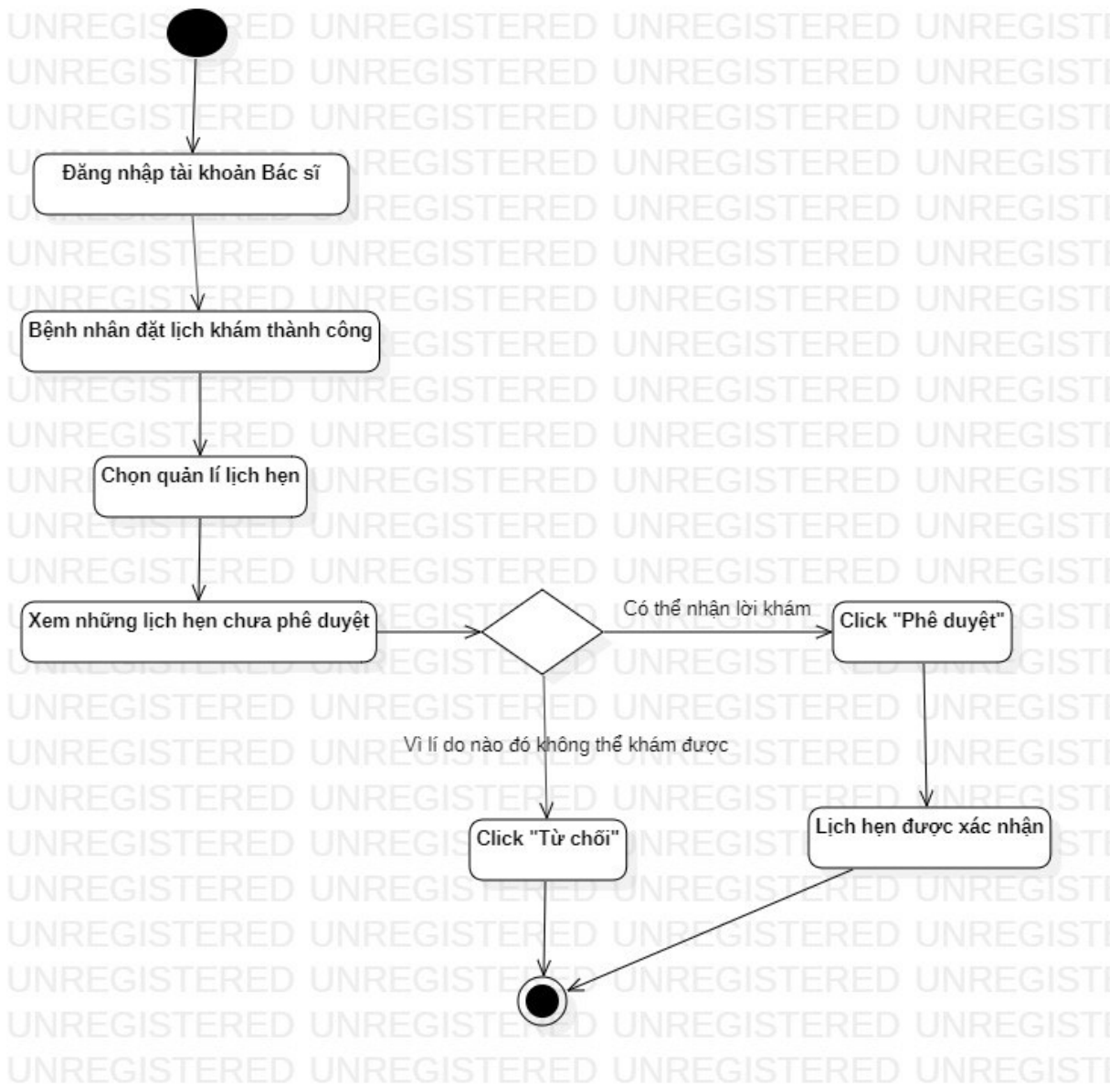
- ✓ Xác nhận phiếu hẹn



Hình 3.9: Sơ đồ hoạt động “Xác nhận phiếu hẹn” của Nhân viên đón tiếp

3.4.2. Bác sĩ/Trợ lý bác sĩ

- ✓ Phê duyệt lịch hẹn



Hình 3.10: Sơ đồ hoạt động “Phê duyệt lịch hẹn” của Bác sĩ

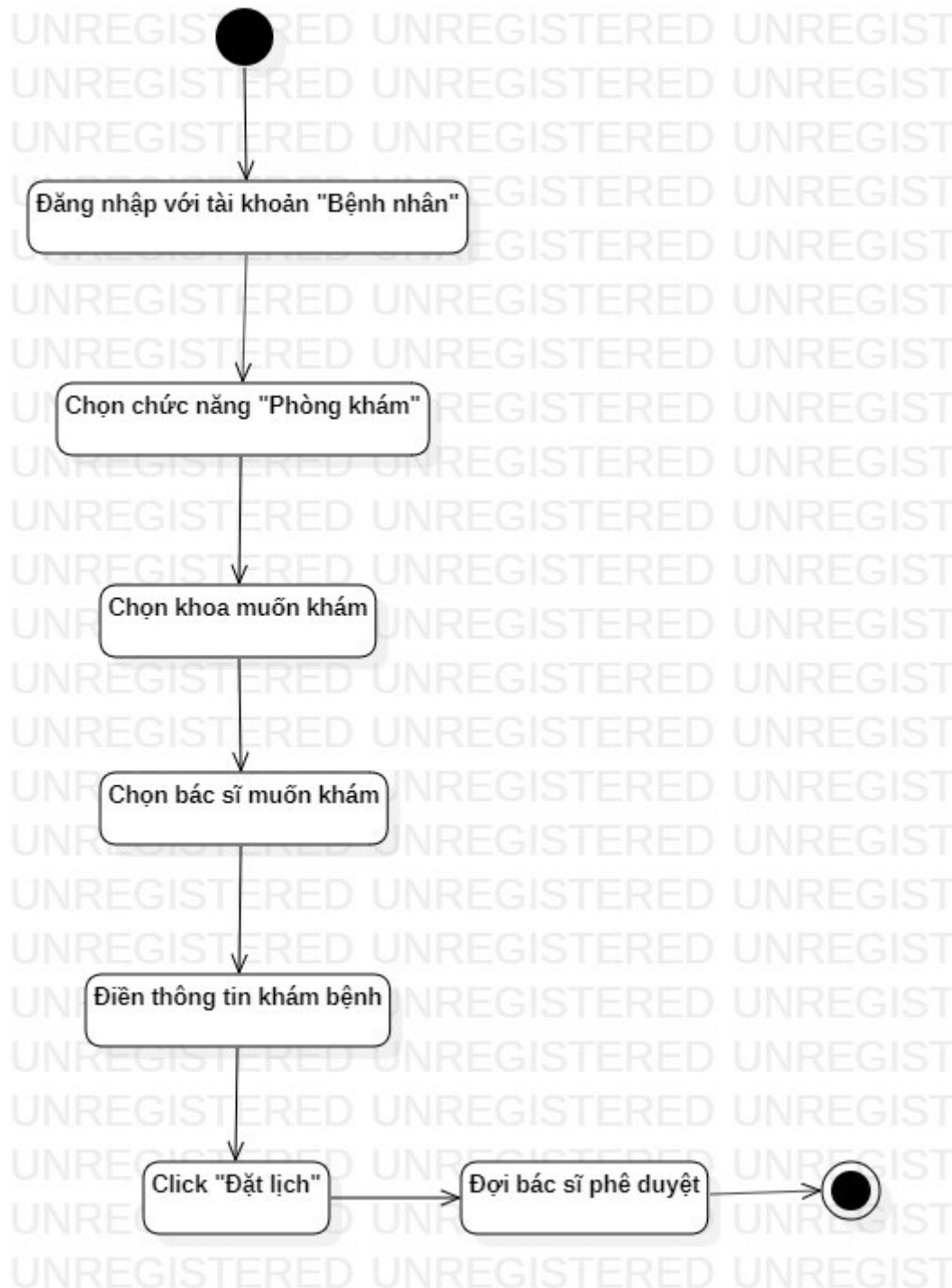
- ✓ Xây dựng lộ trình điều trị



Hình 3.11: Sơ đồ hoạt động “Xây dựng lộ trình điều trị” của Bác sĩ

3.4.3. Bệnh nhân

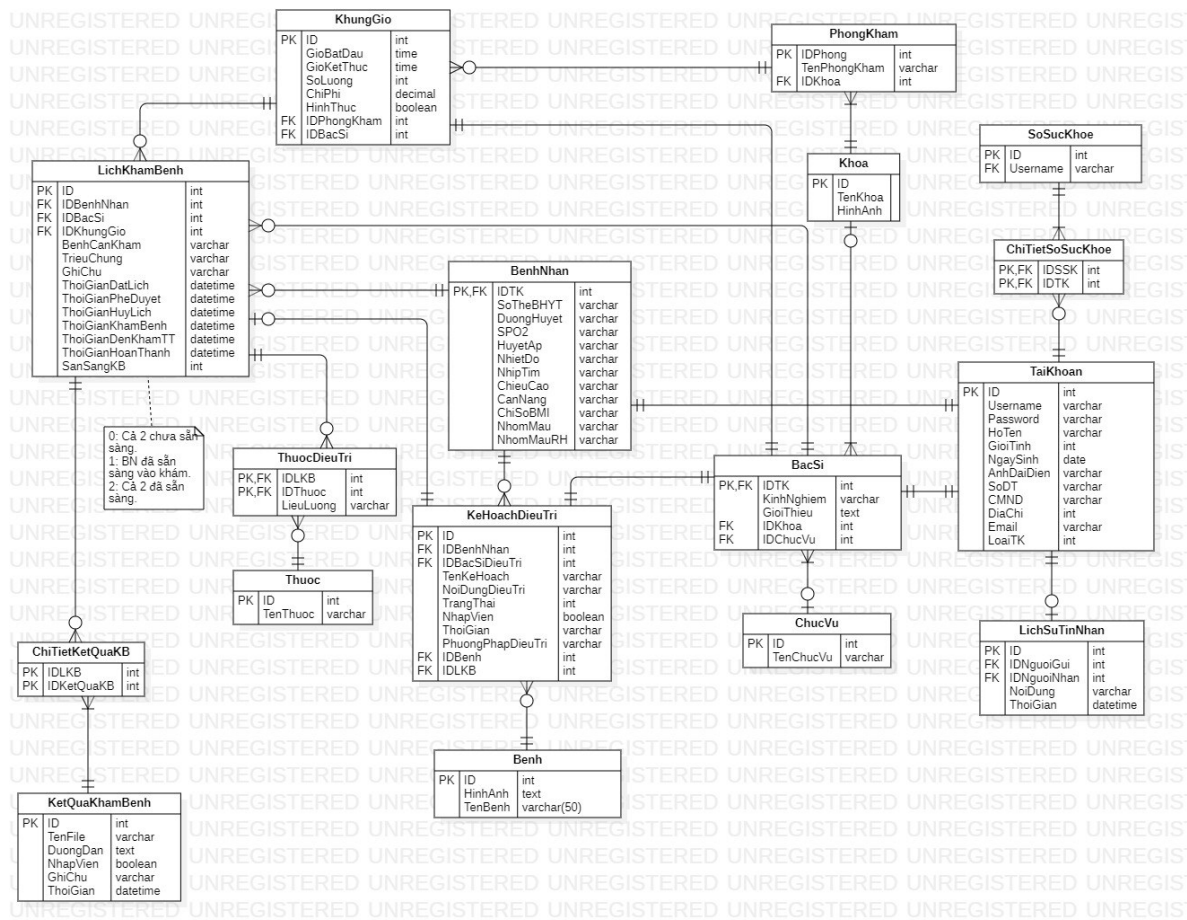
- ✓ Đặt lịch khám bệnh



Hình 3.12: Sơ đồ hoạt động “Đặt lịch khám bệnh” của Bệnh nhân

3.5. Mô hình dữ liệu

3.5.1. Sơ đồ ERD

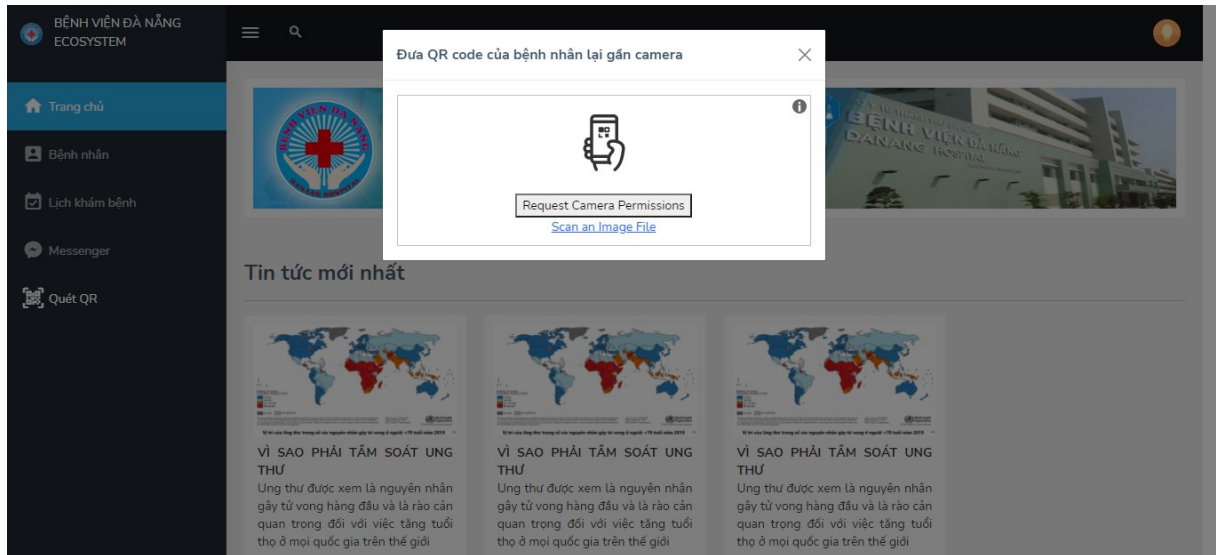


Hình 3.13: Sơ đồ ERD

CHƯƠNG 4: TRIỂN KHAI CHƯƠNG TRÌNH

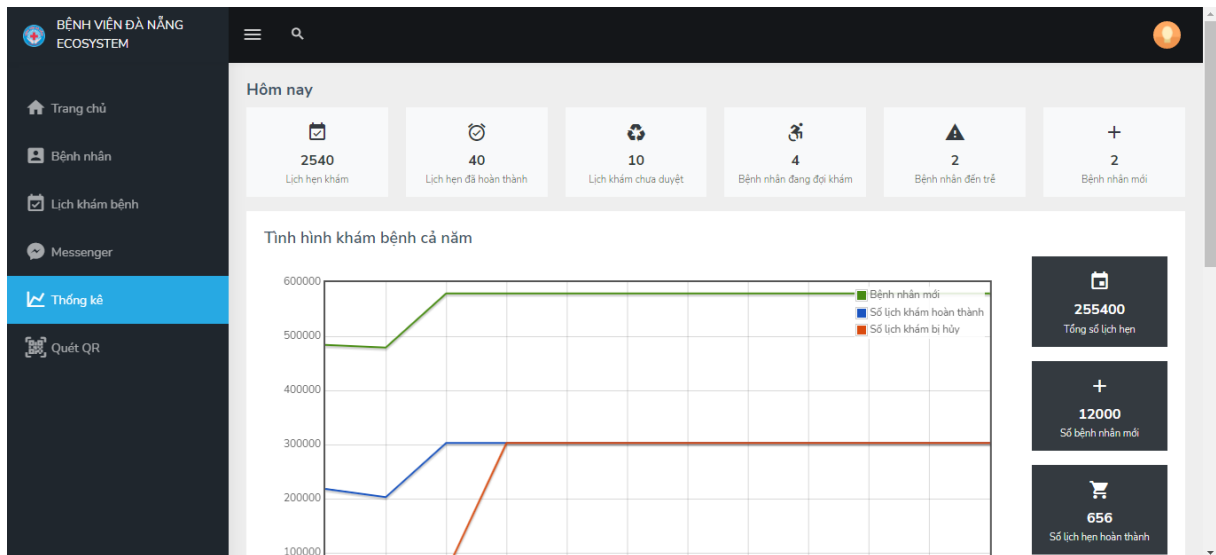
4.1. Nhân viên bộ phận đón tiếp

4.1.1. Xác nhận phiếu hẹn

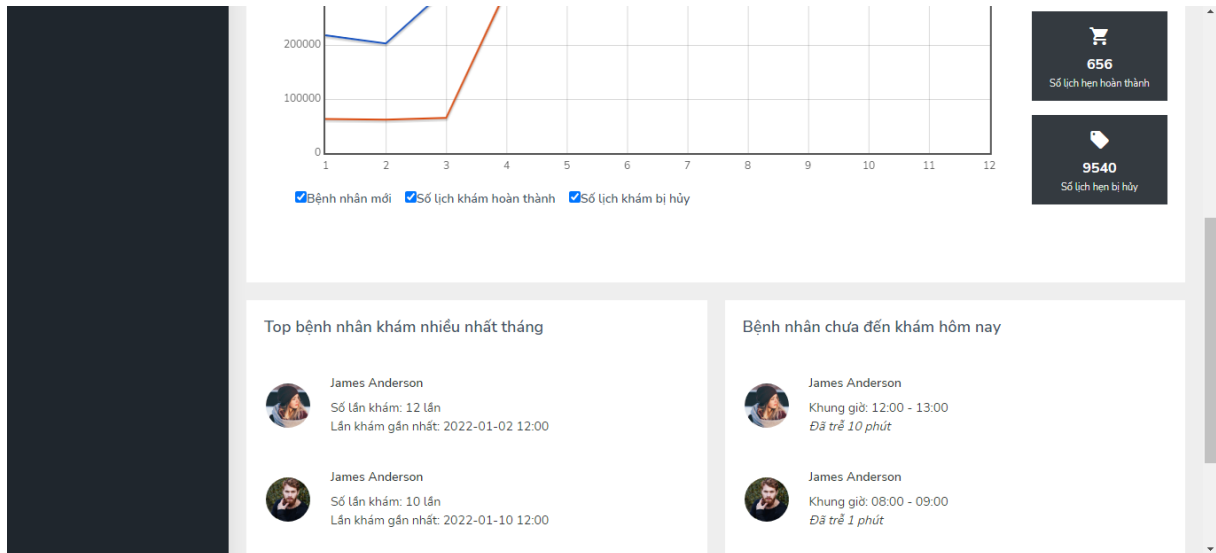


Hình 4.1: Giao diện triển khai chức năng “Xác nhận phiếu hẹn”

4.1.2. Thống kê tình hình khám bệnh của bệnh viện



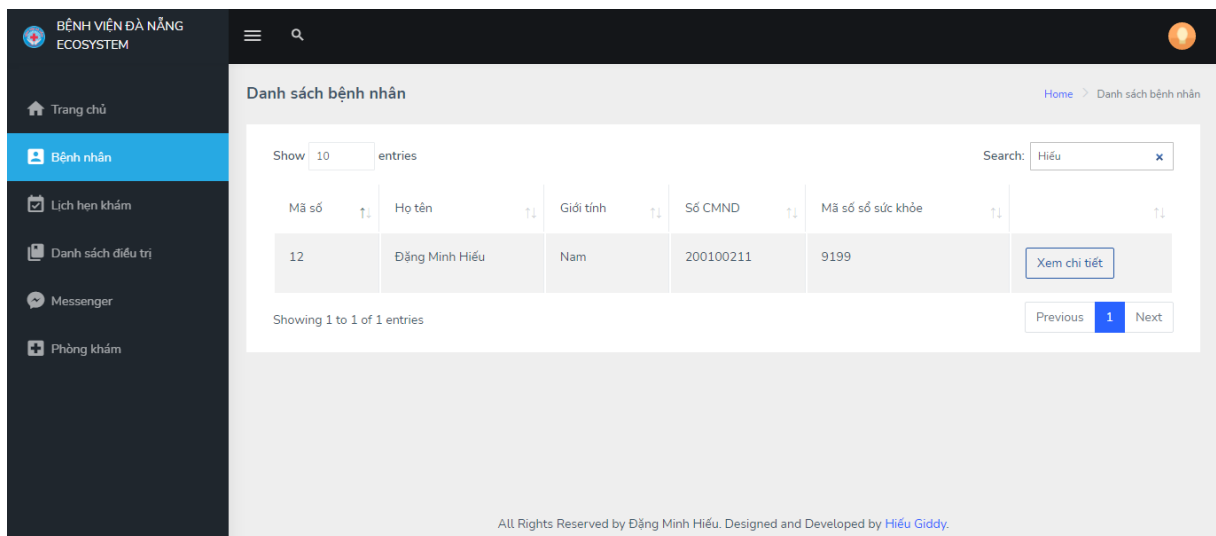
Hình 4.2: Giao diện triển khai chức năng “Thống kê tình hình khám bệnh” 1



Hình 4.3: Giao diện triển khai chức năng “Thống kê tình hình khám bệnh” 2

4.2. Bác sĩ/Trợ lý bác sĩ

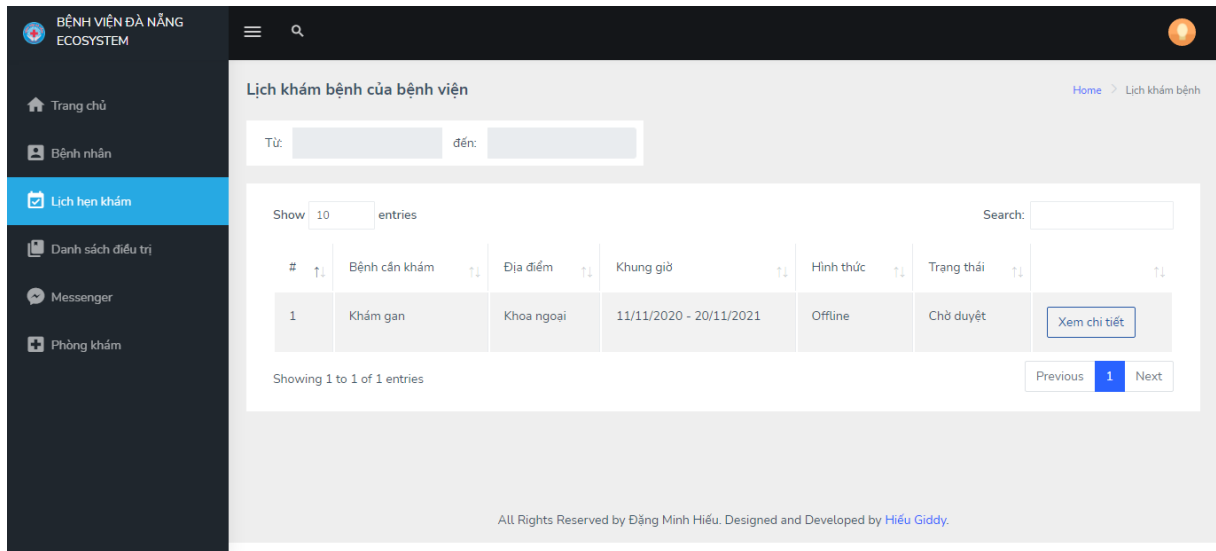
4.2.1. Tìm kiếm bệnh nhân



Hình 4.4: Giao diện triển khai chức năng “Tìm kiếm bệnh nhân”

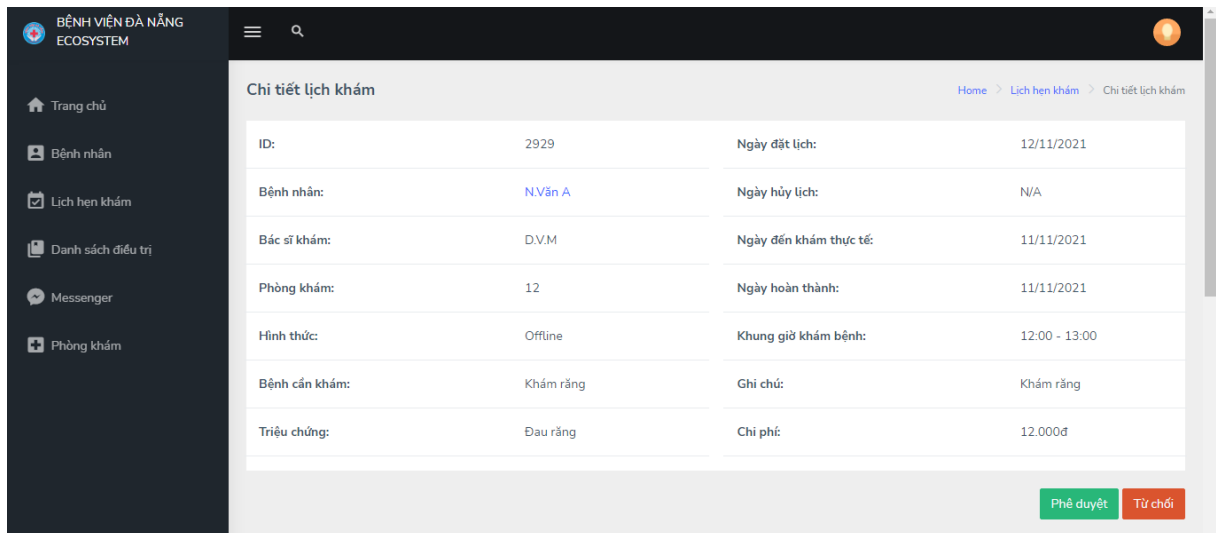
4.2.2. Phê duyệt lịch hẹn

- Đầu tiên, vào trang quản lý lịch hẹn khám bệnh



Hình 4.5: Giao diện triển khai chức năng “Phê duyệt lịch hẹn” 1

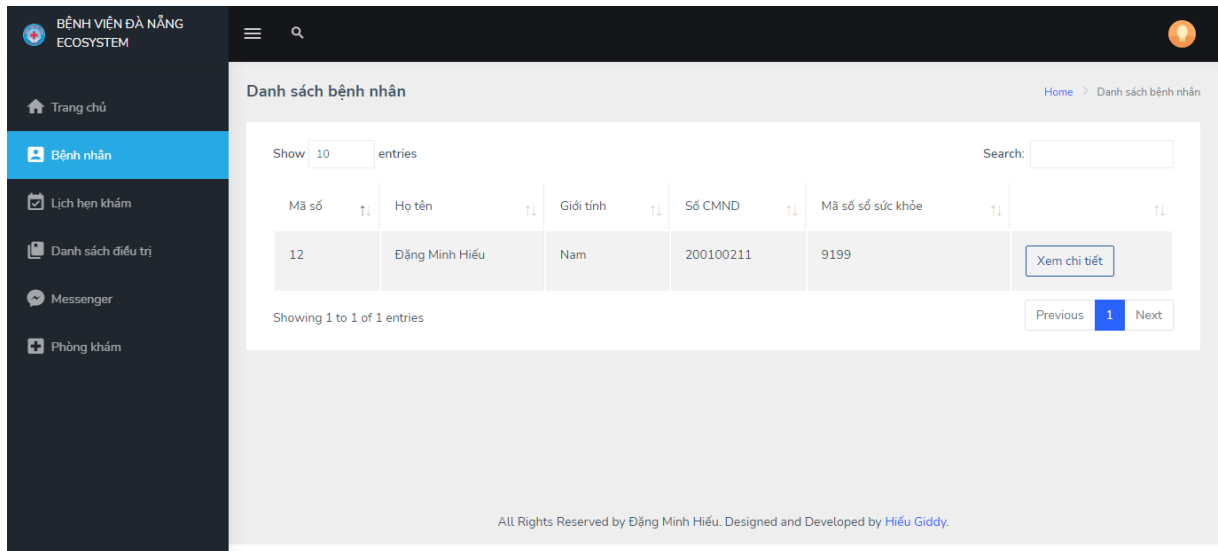
- Sau đó, là xem chi tiết lịch hẹn



Hình 4.6: Giao diện triển khai chức năng “Phê duyệt lịch hẹn” 2

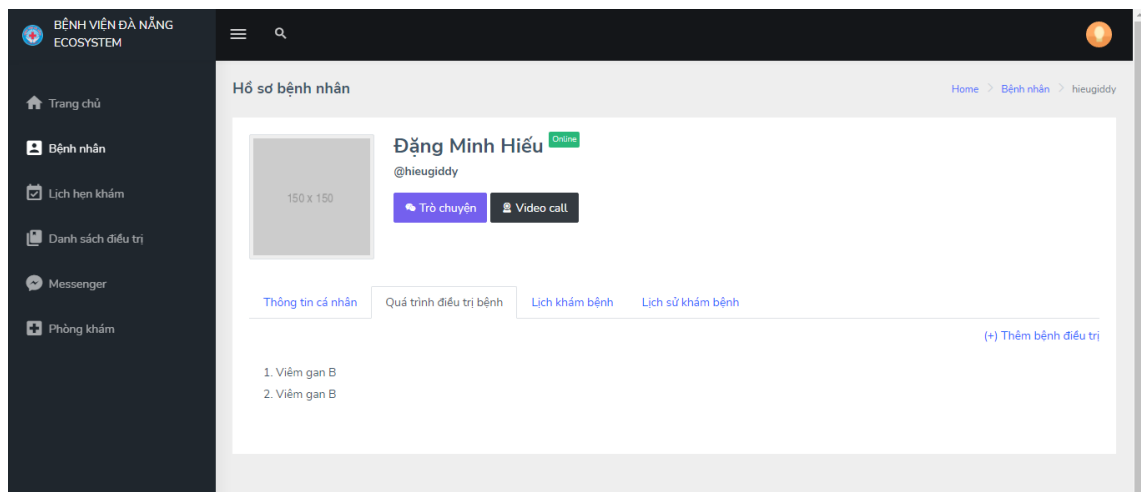
4.2.3. Xây dựng lộ trình điều trị

- Đầu tiên, vào trang Quản lý bệnh nhân



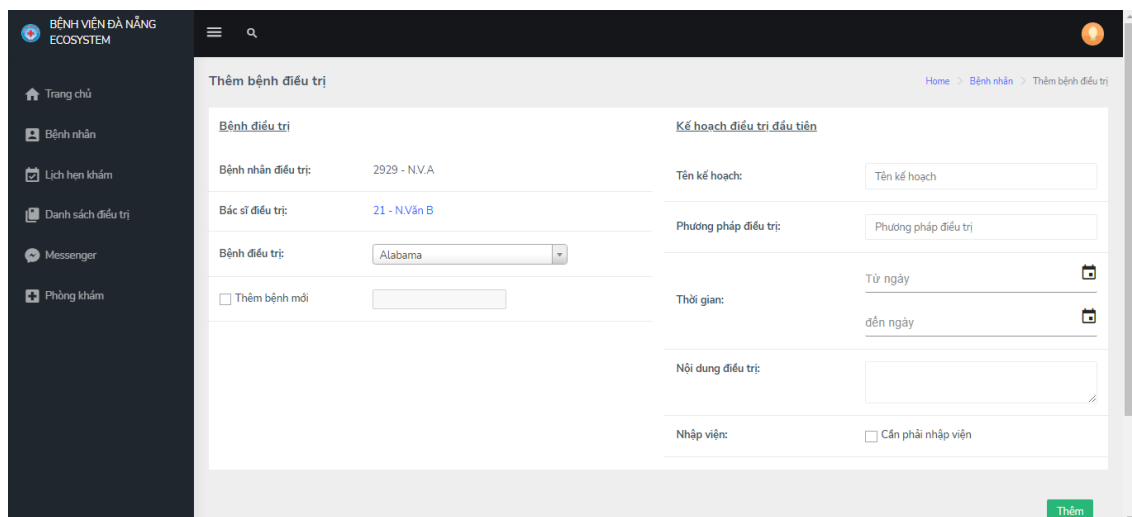
Hình 4.7: Giao diện triển khai chức năng “Xây dựng lộ trình điều trị” 1

- Tiếp theo, chọn “Thêm bệnh điều trị” trong tab “Quá trình điều trị” tại trang “Xem hồ sơ sức khỏe bệnh nhân”



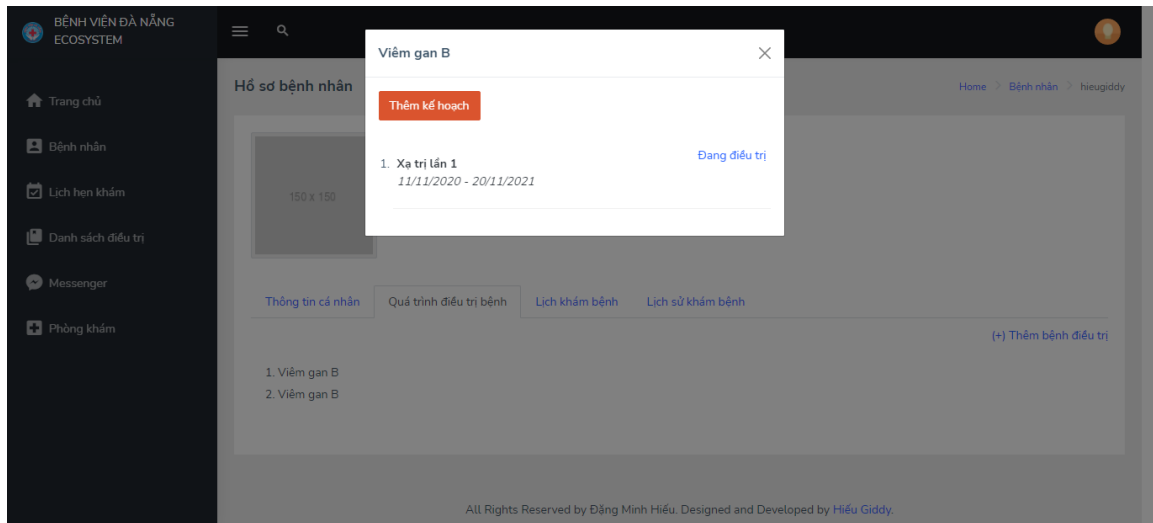
Hình 4.8: Giao diện triển khai chức năng “Xây dựng lộ trình điều trị” 2

- Sau đó, thêm bệnh điều trị



Hình 4.9: Giao diện triển khai chức năng “Xây dựng lộ trình điều trị” 3

- Sau khi thêm bệnh điều trị xong, tiếp theo thêm các kế hoạch điều trị



Hình 4.10: Giao diện triển khai chức năng “Xây dựng lộ trình điều trị” 4

Bệnh nhân điều trị:	2929 - N.V.A	Tên kế hoạch:	Tên kế hoạch
Bác sĩ điều trị:	21 - N.Vân B	Phương pháp điều trị:	Phương pháp điều trị
Bệnh điều trị:	Viêm gan B	Nội dung điều trị:	[Text area]
Thời gian:	Từ ngày đến ngày	Nhập viện:	<input type="checkbox"/> Cần phải nhập viện

Hình 4.11: Giao diện triển khai chức năng “Xây dựng lộ trình điều trị” 5

4.3. Bệnh nhân

4.3.1. Đăng ký tài khoản khám bệnh

- Điền thông tin tài khoản

The screenshot shows the 'Đăng ký tài khoản khám bệnh' (Register medical account) form, Step 1: Account creation. The form is titled 'Đăng ký tài khoản khám bệnh' and has a progress bar with three steps: 1. Tài khoản (Account), 2. Hồ sơ (Profile), and 3. Hoàn thành (Complete). The first step, '1. Tài khoản', is currently active and highlighted in blue. The form contains the following fields: 'Tên đăng nhập *' (Username), 'Mật khẩu *' (Password), and 'Nhập lại mật khẩu *' (Repeat password). Below these fields is a note '(*) Bắt buộc' (Required). At the bottom right, there are two buttons: 'Trước đó' (Previous) and 'Tiếp tục' (Next).

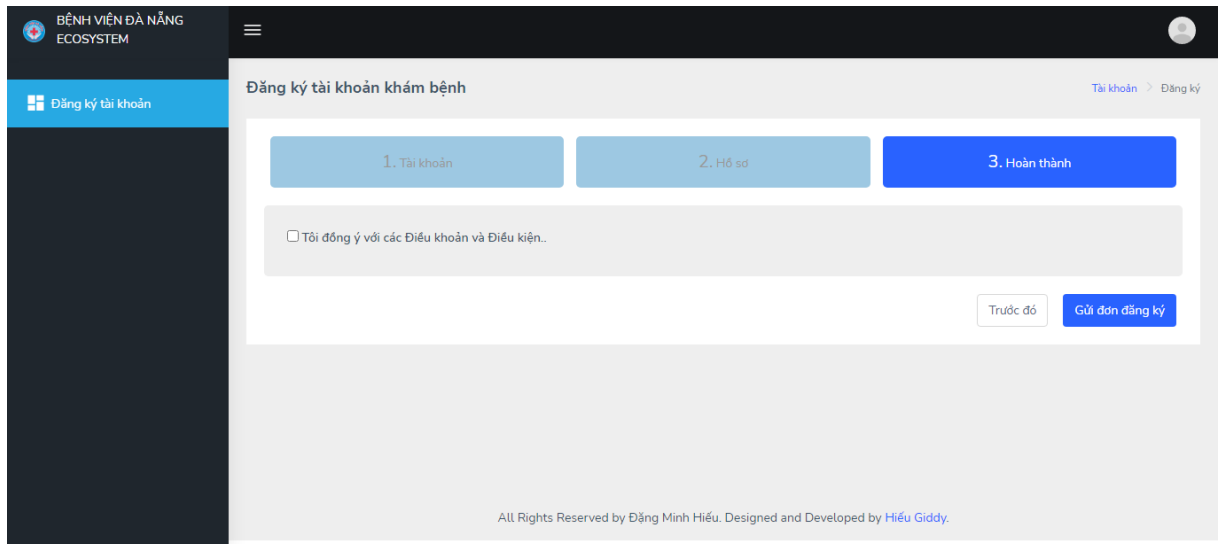
Hình 4.12: Giao diện triển khai chức năng “Đăng ký tài khoản khám bệnh” 1

- Điền các thông tin cơ bản

The screenshot shows the 'Đăng ký tài khoản khám bệnh' (Register medical account) form, Step 2: Basic information. The form is titled 'Đăng ký tài khoản khám bệnh' and has a progress bar with three steps: 1. Tài khoản (Account), 2. Hồ sơ (Profile), and 3. Hoàn thành (Complete). The second step, '2. Hồ sơ', is currently active and highlighted in blue. The form contains the following fields: 'Họ tên *' (Full name), 'Giới tính *' (Gender), 'Ngày sinh *' (Date of birth), 'Số ĐT *' (Phone number), 'Email', and 'CMND/CCID *' (ID card/Passport). The 'Giới tính' field has a dropdown menu with 'Nam' (Male) selected. The 'Ngày sinh' field has a date picker. At the bottom right, there are two buttons: 'Trước đó' (Previous) and 'Tiếp tục' (Next).

Hình 4.13: Giao diện triển khai chức năng “Đăng ký tài khoản khám bệnh” 2

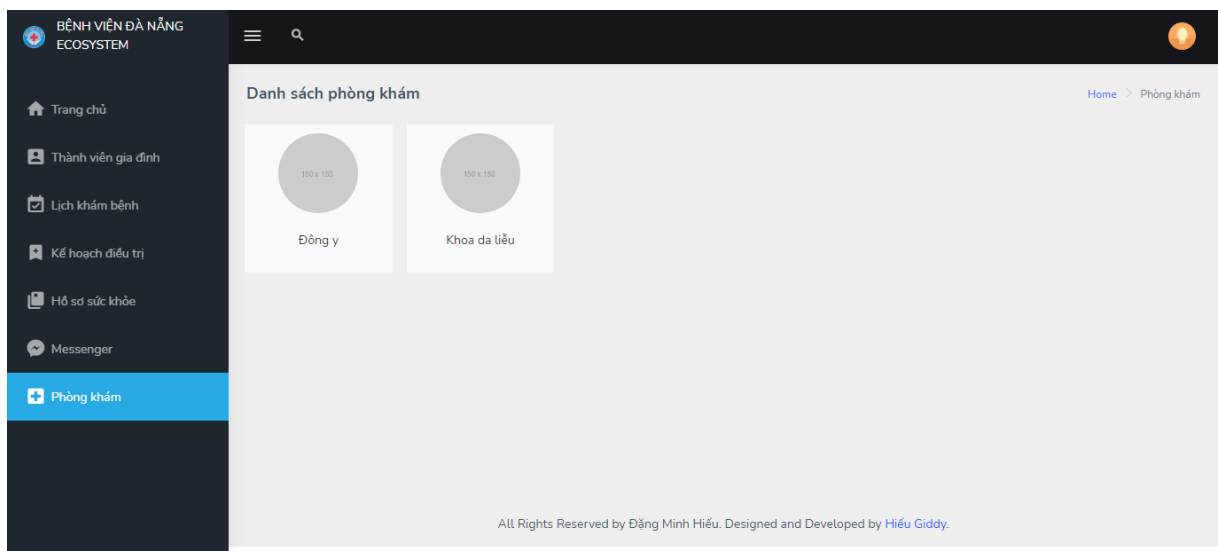
- Đồng ý với các điều khoản và đăng ký



Hình 4.14: Giao diện triển khai chức năng “Đăng ký tài khoản khám bệnh” 3

4.3.2. Đặt lịch khám bệnh

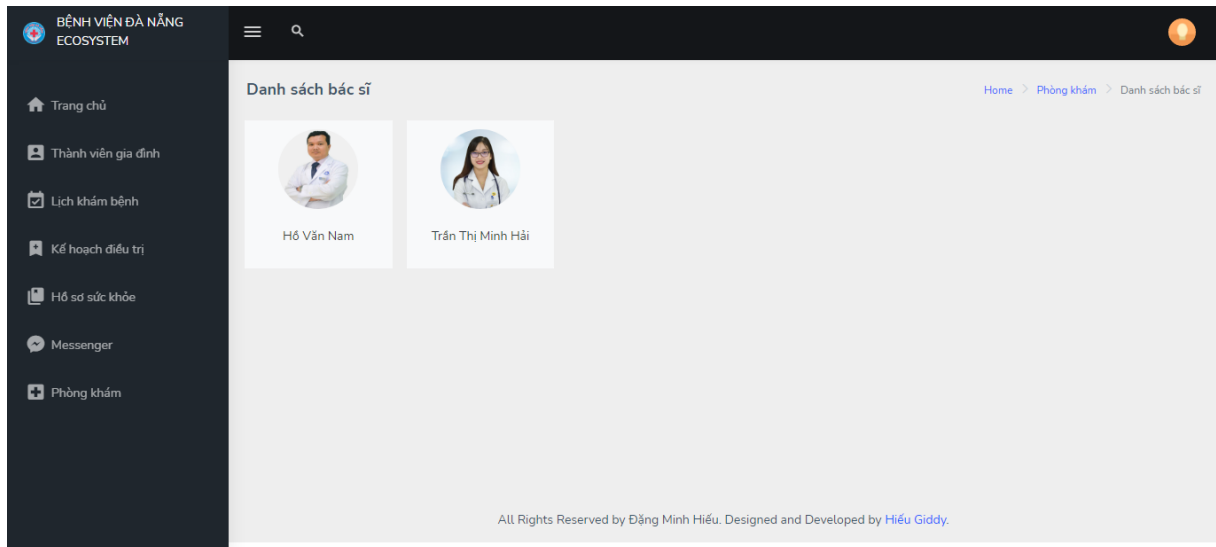
- Đầu tiên, vào trang “Phòng khám”



Hình 4.15: Giao diện triển khai chức năng “Đặt lịch khám bệnh” 1

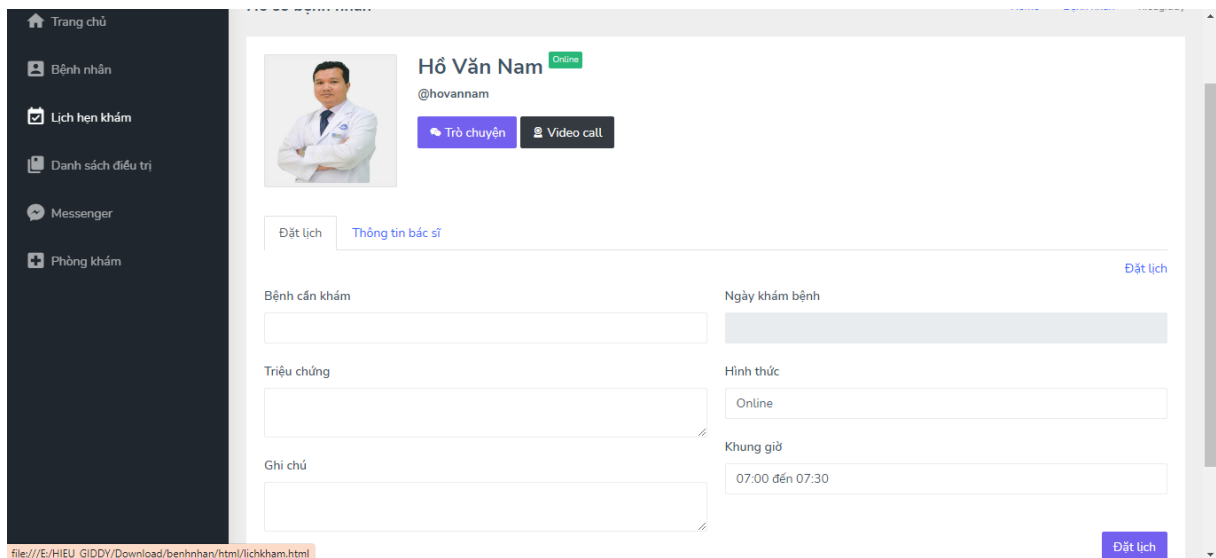
- Lựa chọn bác sĩ muốn khám

ỨNG DỤNG KHÁM CHỮA BỆNH ONLINE TẠI BỆNH VIỆN ĐA KHOA ĐÀ NẴNG



Hình 4.16: Giao diện triển khai chức năng “Đặt lịch khám bệnh” 2

- Đặt lịch khám



Hình 4.17: Giao diện triển khai chức năng “Đặt lịch khám bệnh” 3

KẾT LUẬN & HƯỚNG PHÁT TRIỂN

1. Kết quả đạt được

- Về mặt lý thuyết

- Giải quyết được bài toán đặt lịch khám bệnh
- Giải quyết được bài toán quản lý quá trình khám chữa bệnh của bệnh nhân
- Giúp giảm tải nghẽn, quá tải khi đi khám bệnh
- Tiết kiệm được nhiều thời gian chờ đợi cho bệnh nhân

- Về mặt ứng dụng

Ứng dụng khám chữa bệnh Online tại bệnh viện Đa khoa Đà Nẵng hoàn thành các chức năng liên quan đến cứu hộ xe như sau:

- Đăng ký tài khoản khám bệnh
- Đặt lịch khám bệnh
- Quản lý quá trình khám chữa bệnh của bệnh nhân
- Bác sĩ có thể quản lý danh sách bệnh nhân mà mình đang điều trị

2. Hướng phát triển

- Xây dựng ứng dụng dành cho thiết bị di động
- Thêm chức năng quản lý tài khoản hệ thống cho Actors Nhân viên Phòng IT
- Thêm chức năng quản lý phòng khám bệnh và phân công bác sĩ trực cho Actors Nhân viên Phòng IT
- Thêm chức năng quản lý danh sách bệnh cho Actors Nhân viên Phòng IT
- Thêm chức năng quản lý tủ thuốc cho Actors Nhân viên Phòng IT
- Thêm chức năng Chat thời gian thực

TÀI LIỆU THAM KHẢO

- [1] Bệnh viện Đà Nẵng - Da Nang General Hospital,
<http://dananghospital.org.vn/>
- [2] PeerJS - Simple peer-to-peer with WebRTC,
<https://peerjs.com/>
- [3] Hướng dẫn WebRTC: Xây dựng web stream real time,
<https://www.youtube.com/playlist?list=PLzrVYRai0riQIDNESC4ZBWCJrfFBb3kF>
- [4] Nodejs documentation,
<https://nodejs.org/en/docs/>