

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**



**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
NGÀNH KỸ THUẬT PHẦN MỀM**

**Đề tài
PHÁT TRIỂN ỨNG DỤNG HỖ TRỢ
ĐẶT LỊCH PHÒNG KHÁM TRỰC TUYẾN**

Sinh viên thực hiện: Thạch Chí Tâm

Mã số sinh viên: B18005811

Khóa: 44

Cần Thơ, 05/2022

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**



**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
NGÀNH KỸ THUẬT PHẦN MỀM**

**Đề tài
PHÁT TRIỂN ỨNG DỤNG HỖ TRỢ
ĐẶT LỊCH PHÒNG KHÁM TRỰC TUYẾN**

**Cán bộ hướng dẫn
TS. Lâm Hoài Bảo**

**Sinh viên thực hiện:
Thạch Chí Tâm
MSSV: B18005811
Khóa 44**

Cần Thơ, 05/2022

XÁC NHẬN CHỈNH SỬA LUẬN VĂN THEO YÊU CẦU CỦA HỘI ĐỒNG

Tên luận văn: Phát triển ứng dụng hỗ trợ đặt lịch phòng khám trực tuyến

Họ tên sinh viên: Thạch Chí Tâm

Mã lớp: DI1896A1

Đã báo cáo tại hội đồng ngành: Kỹ thuật phần mềm

Ngày báo cáo: 21/05/2022

Luận văn đã được chỉnh sửa theo góp ý của Hội đồng

Cần Thơ, ngày 05/06/2022

Giáo viên hướng dẫn

(Ký và ghi họ tên)

Lâm Hoài Bảo

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

LỜI CẢM ƠN

Trong suốt niên khóa bốn năm học tập và rèn luyện tại khoa Công nghệ Thông tin và Truyền thông của trường Đại học Cần thơ, tôi đã học tập được nhiều kiến thức, kinh nghiệm quý báu. Các thầy cô giáo đã trở thành những người truyền lửa cho tôi để tiếp tục cố gắng, trau dồi kinh nghiệm, là hành trang để phát triển bản thân trên con đường sự nghiệp sắp tới.

Tôi rất biết ơn với tất cả các thầy cô, đặc biệt là thầy cô khoa Công nghệ Thông tin và Truyền Thông đã tận tình giúp đỡ, hỗ trợ kiến thức nền tảng để góp phần hoàn thiện luận văn.

Tôi xin chân thành cảm ơn thầy Lâm Hoài Bảo đã dành thời gian quý báu để tận tình hỗ trợ, hướng dẫn cũng như đóng góp ý kiến để tôi có thể hoàn thành đề tài luận văn theo đúng kế hoạch.

Và đặc biệt cảm ơn sự động viên của gia đình và lời khuyên từ các anh chị đồng nghiệp tại doanh nghiệp tôi đang công tác.

Xin chân thành cảm ơn !

Cần Thơ, ngày 21 tháng 5 năm 2022

MỤC LỤC

MỤC LỤC	i
DANH MỤC HÌNH	iv
DANH MỤC BẢNG	vi
BẢNG CHÚ GIẢI THUẬT NGỮ	viii
TÓM TẮT.....	1
ABSTRACT	2
PHẦN GIỚI THIỆU	3
1. Đặt vấn đề	3
2. Lịch sử giải quyết vấn đề	3
3. Mục tiêu đề tài	5
4. Đối tượng và phạm vi nghiên cứu	5
5. Phương pháp nghiên cứu	6
6. Kết quả đạt được	7
7. Bố cục luận văn.....	7
PHẦN NỘI DUNG.....	9
CHƯƠNG 1: MÔ TẢ BÀI TOÁN	9
1. Mô tả chi tiết bài toán	9
2. Cơ sở lý thuyết.....	11
2.1. JavaScript	11
2.2. Node.js.....	12
2.3. React.js	13
2.4. MongoDB	13
2.5. Kiểm thử bằng Jest và Supertest	14
2.6. Web API Service	14
3. Mô tả giải pháp cho bài toán.....	15
CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT	16
1. Tổng quan hệ thống	16

1.1. Kiến trúc tổng thể	16
1.2. Mô hình phân rã hệ thống	17
2. Các trường hợp sử dụng.....	17
2.1. Trường hợp sử dụng của Guess (chức năng chung).....	18
2.2. Trường hợp sử dụng của bệnh nhân	25
2.3. Trường hợp sử dụng của bác sĩ	33
2.4 Trường hợp sử dụng của nhân viên phòng khám.....	37
3. Kiến trúc hệ thống	45
4. Thiết kế dữ liệu	47
4.1. Mô hình dữ liệu	47
4.2. Phân tích dữ liệu.....	48
5. Thiết kế giao diện	52
5.1. Giao diện xem lịch	52
5.2. Giao diện đặt lịch	54
5.3. Giao diện lịch hẹn cá nhân của bệnh nhân	55
5.4. Giao diện nhật ký thanh toán của bệnh nhân	56
5.5. Giao diện xem chi tiết lịch khám của bác sĩ.....	57
5.6. Giao diện khám bệnh của bác sĩ.....	58
5.7. Giao diện quản lý thông tin lịch hẹn (Trang chủ của nhân viên phòng khám)	58
5.8. Thống kê thông tin phòng khám	59
CHƯƠNG 3 KIỂM THỬ VÀ ĐÁNH GIÁ	61
1. Mục tiêu kiểm thử:.....	61
2. Trường hợp kiểm thử.....	61
2.1. Kiểm thử thủ công cho giao diện người dùng.....	61
2.2 Kiểm thử tự động dành cho server web api.....	63
3. Đánh giá	70
PHẦN KẾT LUẬN	71
1. Kết quả đạt được	71
2. Hạn chế	71

3. Hướng phát triển	71
TÀI LIỆU THAM KHẢO	72

DANH MỤC HÌNH

Hình 1 Hệ thống đặt lịch trực tuyến của Bệnh viện Đại học Y dược TP.HCM.....	4
Hình 2 Hệ thống đặt lịch của bookingcare.vn.	5
Hình 3 Mô hình tổng quát của hệ thống.	11
Hình 4 Kiến trúc tổng quát của hệ thống.....	16
Hình 5 Mô hình phân rã chức năng của hệ thống.....	17
Hình 6 Trường hợp sử dụng của Guess (chức năng chung).....	18
Hình 7. Sơ đồ tuần tự trường hợp sử dụng Đăng nhập.	19
Hình 8. Sơ đồ tuần tự trường hợp sử dụng Đăng xuất	20
Hình 9. Sơ đồ tuần tự trường hợp sử dụng Quản lý thông tin cá nhân	21
Hình 10. Sơ đồ tuần tự trường hợp sử dụng Đăng xuất	22
Hình 11. Sơ đồ tuần tự trường hợp sử dụng Quên mật khẩu	24
Hình 12. Sơ đồ tuần tự Xem lịch hẹn	25
Hình 13 Trường hợp sử dụng của bệnh nhân.	25
Hình 14. Sơ đồ tuần tự trường hợp sử dụng Đặt lịch khám	27
Hình 15. Sơ đồ tuần tự trường hợp sử dụng Xem lịch khám cá nhân.	28
Hình 16. Sơ đồ tuần tự trường hợp sử dụng Xem chi tiết thông tin lịch hẹn.	29
Hình 17. Sơ đồ tuần tự trường hợp sử dụng Lịch sử thanh toán	30
Hình 18. Sơ đồ tuần tự trường hợp sử dụng Xem chi tiết hóa đơn và đơn thuốc	31
Hình 19. Sơ đồ tuần tự trường hợp sử dụng In hóa đơn.....	32
Hình 20 Trường hợp sử dụng của bác sĩ.	33
Hình 21. Sơ đồ tuần tự trường hợp sử dụng Xem chi tiết lịch khám.	34
Hình 22. Sơ đồ tuần tự trường hợp sử dụng Khám bệnh.	35
Hình 23 Trường hợp sử dụng của nhân viên phòng khám.	37
Hình 24. Sơ đồ tuần tự trường hợp sử dụng Xem danh sách bệnh nhân.....	38
Hình 25. Sơ đồ tuần tự trường hợp sử dụng Tìm kiếm bệnh nhân.....	39
Hình 26. Sơ đồ tuần tự trường hợp sử dụng Thống kê.....	40
Hình 27. Sơ đồ tuần tự trường hợp sử dụng Đặt lịch khám	41
Hình 28. Sơ đồ tuần tự trường hợp sử dụng Xem danh sách lịch hẹn	42
Hình 29. Sơ đồ tuần tự trường hợp sử dụng Xem chi tiết lịch hẹn	43
Hình 30.Sơ đồ tuần tự trường hợp sử dụng Xuất hóa đơn.	45
Hình 31 Mô hình kiến trúc phân tầng của hệ thống.	46
Hình 32 Mô hình dữ liệu mức quan niệm của hệ thống.....	47

Hình 33 Giao diện xem lịch hẹn theo tháng	53
Hình 34 Giao diện xem lịch hẹn theo tuần	53
Hình 35 Giao diện đặt lịch của khách	54
Hình 36 Giao diện đặt lịch của bệnh nhân có tài khoản	54
Hình 37 Giao diện lịch hẹn cá nhân của bệnh nhân	55
Hình 38 Xem chi tiết khi chọn 1 lịch hẹn.....	56
Hình 39 Giao diện nhật ký/ lịch sử thanh toán	56
Hình 40 Giao diện xem chi tiết thanh toán.....	57
Hình 41 Giao diện xem chi tiết lịch khám của bác sĩ.....	57
Hình 42 Giao diện nhập thông tin khám bệnh của bác sĩ.....	58
Hình 43 Trang chủ của nhân viên phòng khám.....	59
Hình 44 Giao diện xem chi tiết lịch khám.....	59
Hình 45 Giao diện thống kê của phòng khám	60
Hình 46 Thư mục và tệp tin kiểm thử	70
Hình 47 Kết quả kiểm thử API.....	70

DANH MỤC BẢNG

Bảng 1 Trường hợp sử dụng Đăng nhập	19
Bảng 2 Trường hợp sử dụng Đăng xuất	20
Bảng 3 Trường hợp sử dụng Quản lý thông tin cá nhân	21
Bảng 4 Trường hợp sử dụng Đổi mật khẩu	22
Bảng 5 Trường hợp sử dụng Quên mật khẩu	23
Bảng 6 Trường hợp sử dụng Xem lịch hẹn	25
Bảng 7 Trường hợp sử dụng Đặt lịch khám	26
Bảng 8 Trường hợp sử dụng Xem lịch khám cá nhân	27
Bảng 9 Trường hợp sử dụng Xem chi tiết lịch hẹn	28
Bảng 10 Trường hợp sử dụng Xem lịch sử thanh toán	29
Bảng 11 Trường hợp sử dụng Xem chi tiết hóa đơn và đơn thuốc	30
Bảng 12 Trường hợp sử dụng In hóa đơn	32
Bảng 13 Trường hợp sử dụng Xem chi tiết lịch khám	34
Bảng 14 Trường hợp sử dụng Khám bệnh	35
Bảng 15 Trường hợp sử dụng Thêm toa thuốc	36
Bảng 16 Trường hợp sử dụng Xem danh sách bệnh nhân	38
Bảng 17 Trường hợp sử dụng Tìm kiếm bệnh nhân	39
Bảng 18 Trường hợp sử dụng Thống kê	39
Bảng 19 Trường hợp sử dụng Đặt lịch khám	40
Bảng 20 Trường hợp sử dụng Xem danh sách lịch hẹn	41
Bảng 21 Trường hợp sử dụng Xem chi tiết lịch hẹn	42
Bảng 22 Trường hợp sử dụng Thay đổi thông tin lịch khám	44
Bảng 23 Trường hợp sử dụng Xuất hóa đơn	45
Bảng 24 Bảng Account (Tài khoản)	48
Bảng 25 Bảng Doctor (Bác sĩ)	48
Bảng 26 Bảng Customer (Bệnh nhân / Khách hàng)	49
Bảng 27 Bảng Staff (Nhân viên phòng khám)	49
Bảng 28 Bảng Expertise (Chuyên môn)	50
Bảng 29 Bảng Doctor - Expertise	50
Bảng 30 Bảng Service (Dịch vụ)	50
Bảng 31 Bảng Appointment Schedule (Lịch hẹn)	51
Bảng 32 Bảng Receipt (Hóa đơn)	51

Bảng 33 Bảng Prescription (Toa thuốc)	51
Bảng 34 Bảng Drug (Thuốc)	51
Bảng 35 Bảng Prescription - Drug (Toa thuốc - Thuốc)	52
Bảng 36 Trường hợp kiểm thử thủ công công (các chức năng chính)	63

BẢNG CHÚ GIẢI THUẬT NGỮ

STT	Thuật ngữ	Mô tả/ ý nghĩa
1	Guess	Bao hàm tất cả tác nhân sử dụng đến hệ thống
2	Client	Trang web được tải tại phía người dùng
3	Server/ server web API	Máy chủ cung cấp tài nguyên cho các thành phần khác

TÓM TẮT

Tại các phòng khám hiện tại, việc thực hiện khám chữa bệnh giữa bác sĩ và các bệnh nhân vẫn thực hiện theo cách truyền thống, có nghĩa là khi có bệnh nhân đến thăm khám, thì lúc đó bác sĩ sẽ thực hiện khám bệnh cho bệnh nhân, bên cạnh đó trong thời gian cao điểm, việc “lấy số, xếp hàng” làm mất khá nhiều thời gian của người khám bệnh. Việc thực hiện khám chữa bệnh truyền thống như trên gây khó khăn cho bệnh nhân là không biết lúc bệnh nhân đến thăm khám thì có bác sĩ hay không và bác sĩ cũng không thể nào biết được trong tương lai sẽ có ai đến khám bệnh tại cơ sở của mình, điều này làm cho phòng khám hoạt động có lúc bị quá tải hoặc thiếu bệnh nhân bị khám chữa bệnh.

Từ yêu cầu thực tế trên, một hệ thống hỗ trợ cho bệnh nhân có thể chủ động đặt lịch hẹn với bác sĩ mong muốn vào một khoảng thời gian xác định, giúp phòng khám có thể chủ động liên lạc để thu thập thêm thông tin của bệnh nhân để đề ra phương pháp khám chữa bệnh tối ưu, bên cạnh đó tiết kiệm thời gian khám chữa bệnh vì bệnh nhân không còn phải chờ đợi, xếp hàng.

Xây dựng hệ thống đặt lịch trực tuyến dựa trên nền tảng web, giúp bệnh nhân có thể đặt lịch khám bệnh với một bác sĩ xác định, xem các hóa đơn đã thanh toán sau khi đến khám chữa bệnh cũng như xem toa thuốc và chẩn đoán của bác sĩ. Nhân viên phòng khám có thể tiếp nhận thông tin của các bệnh nhân để có thể chủ động liên lạc, thu thập thêm thông tin của bệnh nhân. Bác sĩ có thể xem được lịch làm việc của mình, danh sách lịch khám, các bệnh nhân sẽ khám trong ngày, bên cạnh đó có thể đặt toa thuốc và thông tin điều trị để bệnh nhân có thể xem lại về sau. Tất cả nhân viên, bác sĩ và bệnh nhân có định danh có thể trao đổi trực tiếp với nhau qua hệ thống tư vấn trực tuyến để mang lại kết quả điều trị tốt nhất.

ABSTRACT

In the current private clinics, medical examination and treatment between doctors and patients is still done in the traditional way, which means that when a patient comes to visit, the doctor will perform the examination. Besides, during peak times, "taking numbers, queuing up" takes a lot of time for the patients. The implementation of traditional medical examination and treatment as above makes it difficult for patients to know a reasonable time to come to the clinic and the doctor cannot know in the future who will come to visit the clinic. This makes the clinic operate at times with overload or lack of patients being examined and treated.

From the above practical requirements, a patient support system can actively book an appointment with the desired doctor at a specified time, helping the clinic to actively contact to collect more information of the patient to propose the optimal method of medical examination and treatment, besides saving time for medical examination and treatment because patients no longer have to wait and queue.

Building an online booking system, allowing patients to book an appointment with a specific doctor, view paid bills after visiting, as well as view prescriptions and diagnoses doctor's guesses. Clinic staff can receive patient information so that they can proactively contact and collect more patient information. Doctors can view their work schedule, list of appointments, patients who will be examined for the day, and can order prescriptions and treatment information for patients to review later. All staff, doctors and identified patients can communicate directly with each other through the online consultation system to bring about the best treatment results.

PHẦN GIỚI THIỆU

1. Đặt vấn đề

Hiện nay, công nghệ thông tin đã phát triển đến một tầm cao mới, được ứng dụng vào mọi mặt trong cuộc sống, giúp mọi hoạt động của con người trở nên đơn giản, tiết kiệm thời gian, từ đó tạo ra thể chủ động trong công việc. Để đáp ứng nhu cầu này, phần lớn các doanh nghiệp đã và đang tiến hành số hóa các hoạt động kinh doanh, sản xuất của mình. Để hưởng ứng theo xu thế này, ở các trung tâm y tế, bệnh lớn đã ứng dụng để tối ưu hóa quá trình khám chữa bệnh bằng cách xây dựng hệ thống quản lý doanh nghiệp cho riêng mình, đa phần tập trung phát triển hệ thống quản lý hành chính.

Nhưng bên cạnh đó, việc hiện đại hóa và ứng dụng công nghệ thông tin vào các phòng khám vừa và nhỏ dường như chưa được quan tâm và chú trọng. Một phần là vì chi phí để triển khai được một mô hình phát triển cho phòng khám là khá cao, bên cạnh đó độ phức tạp của hệ thống làm cho người sử dụng khó có thể tiếp cận.

Vì thế, các cơ sở y tế, phòng khám, thậm chí một số bệnh viện lớn vẫn sử dụng hình thức bắt số - xếp hàng để chờ đến lượt khám của bệnh nhân theo cách truyền thống một cách phổ biến, điều này làm mất thời gian của bệnh nhân vì phải chờ đợi trong thời gian dài, hơn nữa, trong tình hình dịch Covid-19 vẫn còn đang diễn biến phức tạp, vẫn tạo ra nguy cơ lây nhiễm trong cộng đồng.

Do vậy, một hệ thống đặt lịch hẹn dành cho một phòng khám là thật sự cần thiết, nó phải giải quyết được bài toán chờ đợi của bệnh nhân, hạn chế việc tiếp xúc giữa mọi người, tạo ra một môi trường an toàn cho người bệnh và phòng khám, nhưng phải làm sao tiết kiệm được chi phí, có khả năng tích hợp vào hệ thống bệnh viện sẵn có và dễ sử dụng cho mọi người.

2. Lịch sử giải quyết vấn đề

Trên thực tế, đã có rất nhiều hệ thống đặt lịch dành cho phòng khám, ví dụ: Đại học Y dược Thành phố Hồ Chí Minh có cả một hệ thống tích hợp quản lý bệnh viện và đặt lịch, nhưng đây là hệ thống đặc thù dành cho một cơ quan xác định, không thể tái sử dụng hoặc chuyển giao cho các phòng khám vừa và nhỏ, và bệnh nhân vẫn phải xếp hàng chờ đợi vì không biết mình sẽ được khám vào lúc mấy giờ.

BỆNH VIỆN ĐẠI HỌC Y DƯỢC TP.HCM
PHÒNG KHÁM BỆNH VIỆN ĐẠI HỌC Y DƯỢC 1
Điều trị chuyên nghiệp - Nâng tầm Y Việt

Trang chủ | Đăng ký khám

ĐĂNG KÝ KHÁM

Lưu ý:
Lịch hẹn có hiệu lực sau khi có xác nhận chính thức từ Phòng khám Bệnh viện Đại học Y Dược 1.
Quý khách hàng vui lòng cung cấp thông tin chính xác để được phục vụ tốt nhất. Trong trường hợp cung cấp sai thông tin email & điện thoại, việc xác nhận cuộc hẹn sẽ không hiệu lực.
Quý khách sử dụng dịch vụ đặt hẹn trực tuyến, xin vui lòng đặt trước ít nhất là 24 giờ trước khi đến khám.
Trong trường hợp khẩn cấp hoặc nghi ngờ có các triệu chứng nguy hiểm, quý khách vui lòng ĐẾN TRỰC TIẾP Phòng khám hoặc các trung tâm y tế gần nhất để kịp thời xử lý.

ĐĂNG KÝ KHÁM
vui lòng điền thông tin vào form bên dưới để đăng ký khám bệnh theo yêu cầu

Họ và tên: Email:

Ngày sinh: Số điện thoại:

Giới tính:

Địa chỉ:

Xét Nghiệm:

20/04/2022: Buổi sáng:

Triệu chứng:

GỬI

Hình 1 Hệ thống đặt lịch trực tuyến của Bệnh viện Đại học Y dược TP.HCM.

Mặc khác, có nhiều website cung cấp thông tin đặt lịch cho các phòng khám, giữa vai trò cầu nối giữa các phòng khám và bệnh nhân có nhu cầu khám bệnh, bệnh nhân có thể chủ động đăng ký chọn phòng khám và thời gian như mong muốn, nhưng người bệnh phải chấp nhận chịu một khoản phí “giới thiệu” dành cho các website này.

The image shows two identical booking cards for 'BỆNH VIỆN CHỢ RẪY' (Chợ Rẫy Hospital). Each card is for a different service: 'Khám Viêm gan tại Bệnh viện Chợ Rẫy' (Liver inflammation examination) and 'Khám Nội Tiết tại Bệnh viện Chợ Rẫy' (Endocrinology examination). Both cards include the hospital's logo, a brief description of the service, and a location pin for Hồ Chí Minh. The booking interface on the right of each card shows a date selector set to 'Thứ 3 - 19/4', a 'LỊCH KHÁM' (Examination schedule) section with a grid of time slots (06:00-07:00, 07:00-08:00, 08:00-09:00, 09:00-10:00, 10:00-11:00, 13:00-14:00, 14:00-15:00, 15:00-16:00), a 'Chọn và đặt' (Select and book) button with a note '(Phí đặt lịch 15,000đ)', and a 'GIÁ KHÁM: 38,700đ. Xem chi tiết' (Examination fee: 38,700 VND. View details) link.

Hình 2 Hệ thống đặt lịch của bookingcare.vn.

3. Mục tiêu đề tài

Mục tiêu của đề tài là xây dựng được một hệ thống mô phỏng được chức năng cốt lõi là đặt lịch phòng khám trực tuyến, mà trong đó bệnh nhân có khả năng chủ động chọn bác sĩ điều trị, thời gian gặp mặt. Còn nhân viên và bác sĩ sẽ có trước thông tin về bệnh nhân sẽ đến điều trị trong tương lai, bên cạnh đó các người dùng có thể tương tác qua lại với nhau bằng hệ thống tư vấn trực tuyến có sẵn để nâng cao chất lượng phục vụ và điều trị của phòng khám.

4. Đối tượng và phạm vi nghiên cứu

Trong phạm vi đề tài chỉ đề cập đến hệ thống đặt lịch trực tuyến nên có sẵn các tài nguyên của bệnh viện như thuốc, tài khoản dành cho bác sĩ và nhân viên.

Đề tài có 4 đối tượng sử dụng:

- Bệnh nhân không thường xuyên: là những bệnh nhân lần đầu sử dụng hệ thống hoặc tần suất sử dụng không đáng kể, nhóm người dùng này không cần phải có tài khoản và họ vẫn có thể đặt lịch hẹn. Nhóm người dùng này có thể tạo một tài khoản để sử dụng thêm các chức năng nâng cao.

- Bệnh nhân thường xuyên: là những bệnh nhân đã sử dụng hệ thống hoặc tần suất khám chữa bệnh nhiều, nhóm người dùng này đã có một tài khoản, ngoài khả

năng đặt lịch, họ còn có thể được tư vấn trực tuyến với phòng khám, xem lịch sử khám bệnh, chi phí.

- Nhân viên phòng khám: có thể xem được danh sách các bệnh nhân thuộc phòng khám của mình, lịch hẹn mà bệnh nhân đã đăng ký, từ đó có thể chủ động liên hệ tư vấn để bệnh nhân có thể đặt lịch hẹn hợp lý nhất, xác nhận các lịch hẹn mà bệnh nhân đăng ký.

- Bác sĩ: có thể xem lịch khám bệnh của mình bởi trong lịch khám bệnh, có thể tiến hành khám bệnh cho các cuộc hẹn mà nhân viên phòng khám đã xác nhận, có thể chẩn đoán tình trạng bệnh, đưa ra toa thuốc phù hợp cho bệnh nhân.

5. Phương pháp nghiên cứu

Tìm hiểu tổng quát các yêu cầu liên quan đến phòng khám, quy trình thực hiện khám chữa bệnh từ ra rút ra kinh nghiệm và đề xuất hướng giải quyết tối ưu, hiệu quả nhất.

- Hướng giải quyết: tìm kiếm thông tin thông qua internet, bên cạnh đó trau dồi thêm những kiến thức mới để giải quyết các vấn đề có thể gặp phải trong quá trình cài đặt hệ thống mô phỏng.

- Phân tích yêu cầu: hệ thống đặt lịch trực quan, dễ dàng tiếp nhận đối với người dùng lần đầu sử dụng, không xảy ra trùng lặp lịch giữa các lịch hẹn, bên cạnh đó vẫn phải đảm bảo khả năng bảo mật thông tin.

- Thiết kế: phân tích mô hình trường hợp sử dụng, kiến trúc tổng thể, kiến trúc hệ thống, mô hình kết nối giữa các thành phần, mô hình thực thể.

- Cài đặt:

- + Đề xuất kế hoạch thực hiện.

- + Ngôn ngữ lập trình chủ yếu: Javascript.

- + Framework/thư viện hỗ trợ: React.js, Node.js Express, ORM Mongoose, Jest,...

- + Hệ quản trị cơ sở dữ liệu: MongoDB.

- + Công cụ: VSCode, Microsoft Edge, MongoDB Compass, Windows Powershell

- Cơ sở lý thuyết:

- + Tìm hiểu, vận dụng các kiến thức về lập trình, phân tích, thiết kế kiến trúc hệ thống.

+ Tìm hiểu và vận dụng các kiến thức về thiết kế giao diện, tuân thủ các nguyên tắc trong tương tác người máy.

+ Vận dụng các kiến thức về React.js, Node.js, MongoDB và các thư viện liên quan dành cho việc cài đặt hệ thống.

- Môi trường vận hành

+ Máy chủ:

- Hệ điều hành: Windows 10 64bit
- RAM: 16Gb.
- CPU: intel Core i5 8300H.
- Ổ cứng: cấp phát 5Gb, mở rộng tùy thuộc vào độ mở rộng dữ liệu.
- Môi trường: Node v16.13.1.

+ Client: Máy tính cá nhân có kết nối mạng.

6. Kết quả đạt được

Xây dựng hệ thống đặt lịch trực tuyến đáp ứng được các nhu cầu như:

- Các thành phần người dùng có thể tương tác được với hệ thống.
- Hệ thống đặt lịch trực tuyến hoạt động hiệu quả, người dùng có thể đặt lịch thời gian thực, mọi người có thể theo dõi sự thay đổi mà không cần phải làm mới trang.
- Giao diện thân thiện, phản hồi nhanh, hạn chế độ trễ.

7. Bố cục luận văn

Phần giới thiệu:

Cung cấp cho người đọc có cái nhìn tổng quan nhất về đề tài, lịch sử giải quyết vấn đề, mục tiêu đề tài cần đạt được, nội dung nghiên cứu đóng góp thực tiễn của đề tài.

Phần nội dung:

Cung cấp thông tin mô tả bài toán, cách thiết kế và cài đặt hệ thống, kiểm thử và đánh giá đề tài. Đáp ứng các thông tin để người đọc hiểu rõ được các thành phần, chức năng và cách vận hành của hệ thống. Được chia thành 3 chương:

Chương 1 - Mô tả bài toán: Giúp người đọc hiểu rõ các chức năng và đặc điểm của hệ thống.

Chương 2 - Thiết kế và cài đặt giải pháp: Bao gồm thiết kế kiến trúc tổng thể của hệ thống, giải thích chức năng của từng thành phần trong hệ thống, các giải thuật xử lý của hệ thống, thiết kế cơ sở dữ liệu, từ điển dữ liệu...

Chương 3 - Kiểm thử và đánh giá: Mô tả mục tiêu kiểm thử, kịch bản kiểm thử và kết quả kiểm thử của hệ thống.

Phản kết luận:

Đánh giá được chất lượng của hệ thống, những mặt thành công và hạn chế mà hệ thống đạt được, qua đó đề xuất khả năng phát triển của hệ thống trong tương lai.

PHẦN NỘI DUNG

CHƯƠNG 1: MÔ TẢ BÀI TOÁN

1. Mô tả chi tiết bài toán

Hệ thống đặt lịch trực tuyến cung cấp chức năng cơ bản nhất là có thể đặt lịch trực tuyến nhìn mục đích đơn giản quá quy trình thực hiện khám chữa bệnh, bệnh nhân có thể tiết kiệm được thời gian đặt lịch khám bệnh, tiết kiệm chi phí hơn so với đặt lịch thông qua tổng đài bằng cách gọi đến nhân viên đặt lịch, bên cạnh đó, nhân viên viên phòng khám có thể chọn lọc danh sách khám bệnh, thu thập thêm thông tin của bệnh nhân để có một lịch hẹn phù hợp, bác sĩ có thể biết được lịch khám bệnh của các bệnh nhân, từ đó đề ra phương án làm việc hiệu quả.

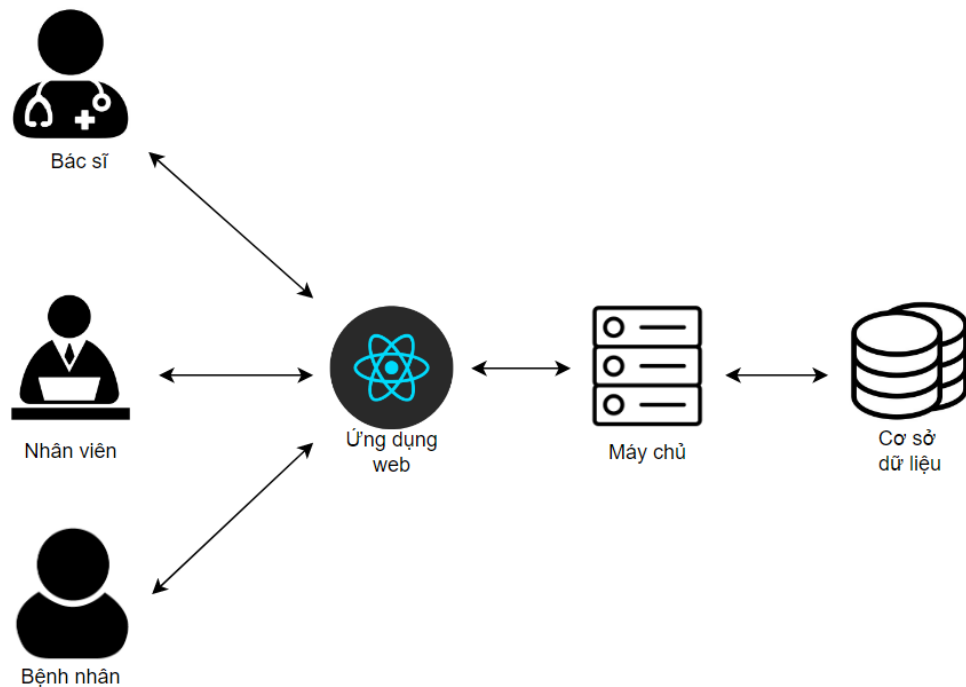
Khi bệnh nhân có nhu cầu khám chữa bệnh, bệnh nhân sẽ truy cập vào hệ thống, bệnh nhân có thể chọn đặt lịch, chọn bác sĩ mà bệnh nhân muốn gặp, bệnh nhân có thể xem được lịch làm việc của bác sĩ, lựa chọn một thời gian hợp lý, nhập các thông tin cần thiết, những thông tin này sẽ được gửi đến nhân viên phòng khám, nhân viên phòng khám dựa vào những thông tin này có thể liên lạc để tư vấn thêm cho bệnh nhân, nhân viên có thể quyết định xóa hoặc xác nhận lịch của bệnh nhân. Nếu lịch hẹn được xác nhận, bác sĩ sẽ có thể tiến hành khám chữa bệnh cho bệnh nhân. Trong quá trình khám chữa bệnh, bác sĩ sẽ cập nhật các thông tin khám chữa bệnh, thông tin toa thuốc nếu có, những thông tin này được gửi đến nhân viên phòng khám để tiến hành bốc thuốc, điều chỉnh các phụ phí. Sau đó, bệnh nhân có thể thanh toán cho phòng khám và nhận thuốc, nếu bệnh nhân là thành viên của phòng khám (có thông tin đăng nhập), bệnh nhân có thể xem lịch sử khám bệnh, toa thuốc và chi phí khám bệnh của mình.

Hệ thống đặt lịch trực tuyến cung cấp được xây dựng chủ yếu để đơn giản hóa quá trình đặt lịch của bệnh nhân. Hệ thống được thiết kế tập trung vào trọng tâm là đặt lịch trực tuyến cho ba nhóm người dùng chính là bác sĩ, bệnh nhân và nhân viên phòng khám, mỗi nhóm người dùng có các chức năng cụ thể:

- Bệnh nhân: nhóm người dùng có thể thực hiện các chức năng liên quan đến đặt lịch, khám chữa bệnh. Chi tiết bao gồm:

- Đặt lịch: bệnh nhân có thể xem các khoảng thời gian khả dụng của các bác sĩ, từ đó chọn ra khoảng thời gian thích hợp để có thể đặt lịch hẹn gặp bác sĩ. Thông tin này sẽ được gửi đến nhân viên phòng khám để xác nhận.
- Xem lịch sử khám bệnh: bệnh nhân có thể xem lịch sử khám bệnh của mình, các thông tin chẩn đoán bệnh, thông tin toa thuốc và hóa đơn đã thanh toán.
- Đăng ký tài khoản: bệnh nhân có thể tạo một tài khoản để có thể sử dụng nhiều hơn các chức năng hỗ trợ khách dành cho nhóm người dùng bệnh nhân.
- Nhân viên phòng khám: nhóm người dùng nhân viên có thể thực hiện các công việc liên quan đến quản lý phòng khám, sắp xếp các lịch hẹn cho bác sĩ:
 - Xem danh sách lịch khám bệnh: biết được các thông tin của bệnh nhân, tiến hành liên hệ xác nhận và chấp nhận lịch khám của bệnh nhân.
 - Bổ sung và xác nhận thông tin thanh toán: bổ sung các khoản phụ phí khi khám chữa bệnh và xác nhận thanh toán cho bệnh nhân.
 - Thống kê: nhân viên phòng khám có thể xem tổng quát các thông tin hoạt động của phòng khám thông qua các biểu đồ, bảng biểu.
- Bác sĩ: nhóm người dùng bác sĩ có thể thực hiện các công việc liên quan đến việc khám chữa bệnh. Các chức năng cho nhóm người dùng này bao gồm:
 - Xem lịch khám chữa bệnh: bao gồm các lịch hẹn đã được nhân viên phòng khám xác nhận
 - Khám chữa bệnh: tại chức năng này, bác sĩ sẽ ghi nhận các thông tin khám bệnh cũng như kết quả chẩn đoán, kê toa thuốc cho bệnh nhân
- Ngoài chức năng trên, hệ thống còn có các chức năng chung dành cho mọi người dùng:
 - Đăng nhập: người dùng có tài khoản có thể đăng nhập vào hệ thống để thực hiện các chức năng riêng biệt của từng nhóm người dùng
 - Quên mật khẩu: chức năng này giúp người dùng có thể đặt lại mật khẩu của tài khoản

- **Chỉnh sửa thông tin cá nhân:** người dùng có thể chỉnh sửa các thông tin được phép chỉnh sửa như họ tên, thông tin liên hệ, địa chỉ,...



Hình 3 Mô hình tổng quát của hệ thống.

2. Cơ sở lý thuyết

2.1. JavaScript

JavaScript là một ngôn ngữ lập nhẹ, được thông dịch hoặc biên dịch. Được phát triển bởi Brendan Eich tại hãng truyền thông Netscape với tên gọi đầu tiên là Mocha, sau đó được đổi tên thành LiveScript, cuối cùng là JavaScript và được sử dụng cho đến ngày nay.

Mặc dù mọi người thường biết đến JavaScript là một ngôn ngữ kịch bản cho các trang Web, nhưng JavaScript vẫn được sử dụng nhiều ở các môi trường khác không phải web như Node.js, Apache Couch và Adobe Acrobat.

JavaScript là ngôn ngữ động dựa trên nguyên mẫu, đa mô hình, đơn luồng, hỗ trợ các kiểu hướng đối tượng, mệnh lệnh và khai báo.

Phiên bản mới nhất của JavaScript là ECMAScript 12. Nhưng phiên bản được sử dụng phổ biến nhất là ECMAScript 6 (hay ECMAScript 2015) – đây là phiên bản có rất nhiều cải tiến về cú pháp, khiến cho JavaScript trở nên phổ biến và thông dụng hơn.

2.2. Node.js

Node.js - được tạo ra bởi Ryan Dahl từ năm 2009 và phát triển dưới sự bảo trợ của Joyent – là một môi trường thực thi JavaScript mã nguồn mở, đa nền tảng triển khai dựa trên Engine V8 được xây dựng bởi Google, có khả năng thực thi mã JavaScript bên ngoài trình duyệt web

Node.js cho phép các lập trình viên sử dụng JavaScript để viết các công cụ dòng lệnh và cho kịch bản phía server, bên cạnh đó còn hỗ trợ cho các thiết bị khác như mobile, desktop, thiết bị nhúng,...

Từ sự ra đời của Node.js, nhiều thư viện hỗ trợ đa nền tảng dần được hình thành như React (Facebook), Angular (Google), Express, Next.js..., lập trình viên có thể tải xuống các gói thư viện này dựa vào các trình quản lý gói thông dụng như NPM (Node Package Manager) hoặc Yarn thông qua giao diện dòng lệnh CLI (Command Line Interface)

Để cài đặt Node.js, có thể tải xuống tại trang chủ của Node.js tại <https://Node.js.org/en> hỗ trợ 3 nền tảng Windows, MacOS và Linux.

- Express Framework trong Node.js cung cấp nhiều tính năng mạnh mẽ trên nền tảng web. Express hỗ trợ các phương thức HTTP và middleware tạo ra API mạnh mẽ và dễ sử dụng. Các ưu điểm của Express như

- Thiết lập các lớp trung gian để trả về HTTP request.
- Định nghĩa router cho phép sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL.

- Để thiết lập dự án dựa trên môi trường Node.js và Express framework ta thực hiện các cú pháp sau:

`npm init` để khởi tạo dự án, thiết lập các thông tin mô tả cho dự án, các thông tin thiết lập của dự án được lưu trữ trong tệp tin package.json.

`npm install express` để cài đặt express framework vào dự án đã có. Lúc này thư mục dự án sẽ có thêm thư mục mới là `node_modules`, thư mục này sẽ lưu trữ toàn bộ các thư viện bên thứ ba.

2.3. React.js

React.js là một thư viện JavaScript mã nguồn mở được thiết kế bởi Facebook để tạo ra các trang web nhanh và hiệu quả với mã tối thiểu. Mục đích cốt lõi của React.js không chỉ là mượt và nhanh mà còn phải có khả năng mở rộng, đơn giản và khả năng tái sử dụng cao.

React tuân theo mô hình lập trình khai báo. Các nhà phát triển thiết kế các khung nhìn cho từng trạng thái của ứng dụng và React cập nhật và hiển thị các thành phần khi dữ liệu thay đổi, một thành phần của React có thể được biểu diễn bằng hàm số $y = g[f(x)]$, trong đó g là giá trị đại diện biểu diễn cho trang web, f là các thành phần của trang web và x là trạng thái của thành phần đó.

Một tính năng đáng chú ý của React là sử dụng “DOM ảo” hay Virtual DOM (Virtual Document Object Model). React tạo ra một bộ đệm cấu trúc dữ liệu trong bộ nhớ, tính toán sự khác biệt và sau đó chỉ cập nhật những khác biệt đó trên trình duyệt, điều này làm cho React trở nên nhẹ hơn thì trình duyệt chỉ cập nhật những thành phần cần thiết, không cập nhật lại toàn bộ trang web như thông thường.

Bên cạnh đó, thay vì sử dụng JavaScript để thiết kế bố cục trang web, React.js sử dụng JSX (JavaScript XML) đơn giản hơn và cho phép trích dẫn HTML cũng như cách sử dụng giống như HTML. JSX tối ưu mã khi biên soạn, vì vậy nó thực thi nhanh hơn mã JavaScript thuần.

Để thiết lập React.js vào trong dự án, có thể thông qua thiết lập thủ công bằng cách cài đặt các gói thư viện và phụ thuộc. Để thuận tiện cho việc sử dụng, thông qua CLI có thể cài đặt nhanh qua câu lệnh: `npx create-react-app <tên dự án>`.

2.4. MongoDB

MongoDB là một cơ sở dữ liệu NoSQL hàng đầu, đa nền tảng, được viết bằng C++, hoạt động trên các khái niệm Collection (Bộ sưu tập) và Document (Tài liệu) nên có hiệu suất cao và dễ mở rộng.

Các khái niệm của MongoDB:

Database: là tập chứa các collection trong MongoDB. Mỗi database sẽ có một tập dữ liệu riêng trên tệp tin hệ thống. Một server MongoDB thường chứa nhiều database.

Collection: là một tập chứa các document, các collection không có chung một schema nào – nghĩa là các document trong cùng một collection không nhất thiết phải giống nhau. Thông thường, các document tương tự nhau sẽ được tập hợp lại trong một collection.

Document: là một tập các cặp dữ liệu theo dạng khóa – giá trị, dùng để biểu diễn thông tin của một đối tượng.

2.5. Kiểm thử bằng Jest và Supertest

Jest là một framework kiểm thử bằng JavaScript tập trung vào sự đơn giản. Nó có thể hoạt động trên nhiều môi trường và framework như Babel, TypeScript, Node, React, Angular,... Do là thư viện sử dụng ngôn ngữ JavaScript nên việc tích hợp vào môi trường Node.js khá đơn giản.

Supertest là một thư viện JavaScript nhằm một đích gọi đến một địa chỉ server có sẵn hoặc một server ngoại tuyến có sẵn.

Sử dụng kết hợp Jest và Supertest để kiểm tra các điểm cuối API để đảm bảo rằng các điểm cuối của server web api hoạt động bình thường, không xảy ra lỗi.

Để cài đặt 2 framework/thư viện này, lần lượt thực hiện 2 lệnh:

```
npm i -D jest
```

```
npm i -D supertest
```

2.6. Web API Service

API là các phương thức, giao tiếp kết nối với các thư viện và ứng dụng khác. Nó là viết tắt của Application Programming Interface – giao diện lập trình ứng dụng. API cung cấp khả năng truy xuất đến một tập các hàm hay sử dụng và từ đó có thể trao đổi dữ liệu giữa các ứng dụng.

Web API là hệ thống API được sử dụng trong các hệ thống website. Các Web API hiện nay đều tuân thủ theo chuẩn REST và HTTP, tạo sự thuận tiện và dễ sử dụng với nhà phát triển. Giúp người dùng dễ dàng truy cập và dễ hiểu hơn.

Ưu điểm của việc sử dụng Web API:

- Khả năng tích hợp linh động.
- Cập nhật thông tin thời gian thực.
- Có tiêu chuẩn chung dễ sử dụng.

3. Mô tả giải pháp cho bài toán

Giải pháp xây dựng website đặt lịch trực tuyến bao gồm:

- Phân tích yêu cầu đối với website đặt lịch trực tuyến: xây dựng hệ thống dành cho ba người dùng cuối là bác sĩ, nhân viên phòng khám và bệnh nhân; đảm bảo các chức năng thiết yếu như đặt lịch, theo dõi thông tin bệnh nhân,...
- Xây dựng server Web API bằng Node.js qua framework Express:
 - Cài đặt, thiết lập các thành phần cần thiết dành cho dự án.
 - Phân tích thiết kế kiến trúc dành cho server.
 - Xây dựng các thành phần dựa trên kiến trúc đã thiết kế.
- Xây dựng trang web dành cho người dùng bằng React.js:
 - Phân tích giao diện dành cho từng đối tượng người dùng.
 - Thiết lập và cài đặt các thư viện cần thiết.
 - Dựng khung hình cho các thành phần trang web.
 - Tiến hành cài đặt các thành phần đã phân tích.
- Xây dựng thành phần thời gian thực cho dự án:
 - Cài đặt dịch vụ trên website.
 - Cài đặt dịch vụ trên server.
- Kiểm thử API của hệ thống bằng Jest và Supertest

CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT

1. Tổng quan hệ thống

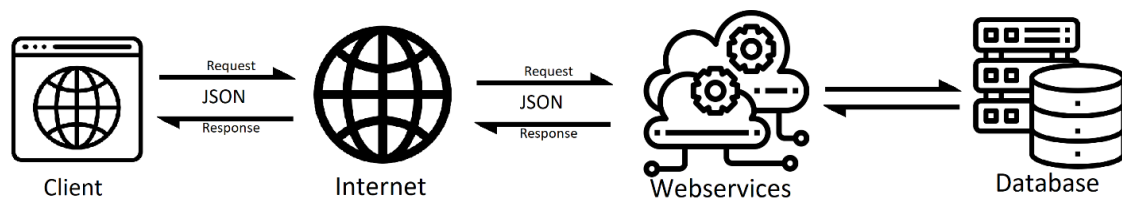
1.1. Kiến trúc tổng thể

Về tổng quát, hệ thống sử dụng mô hình Client – Server, theo cơ chế CSR (Client Side Rendering – Kết xuất phía máy khách), trong đó:

Client ở phía người dùng có chức năng tiếp nhận và phản hồi hành động của người dùng, bên cạnh đó xử lý các nguồn dữ liệu gửi và nhận từ server Node.js – Express.

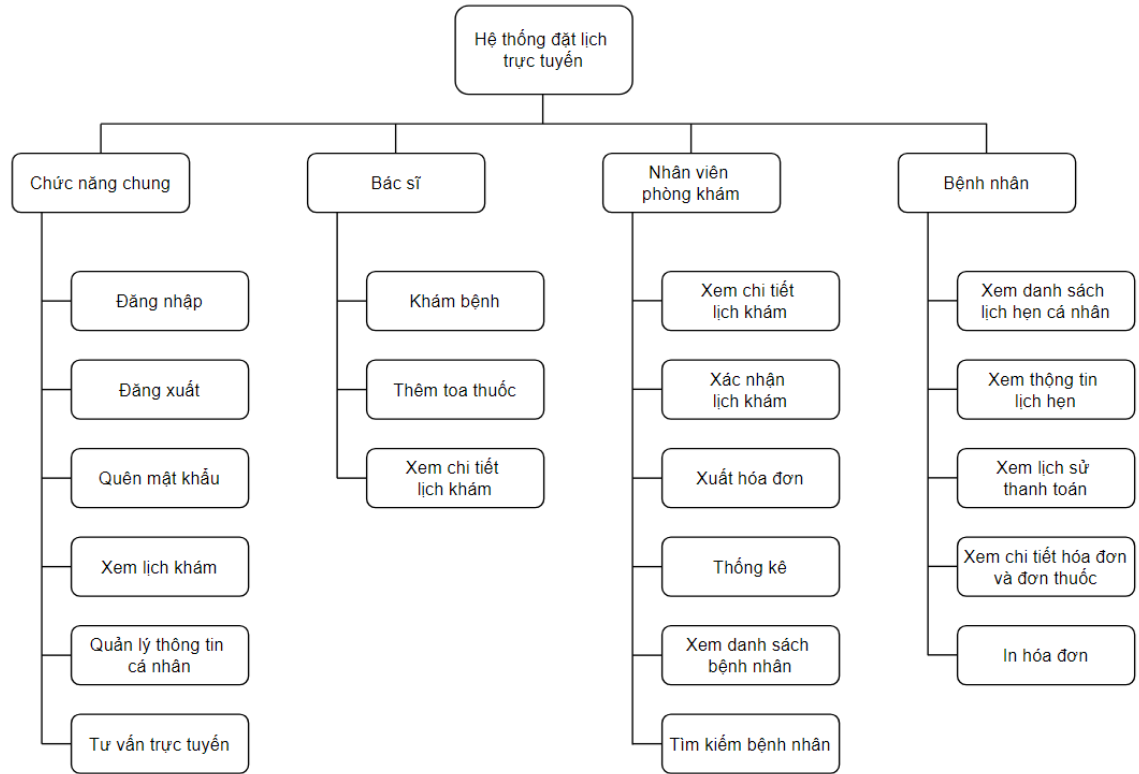
Server là một Web API service được xây dựng bằng Node.js – Express, có chức năng nhận – xử lý – giao tiếp với cơ sở dữ liệu MongoDB và trả dữ liệu cho các client.

Để đảm bảo các hoạt động xảy ra đồng thời, hệ thống thích hợp thêm WebSocket thông qua thư viện socket.io để lắng nghe các yêu cầu thay đổi dữ liệu.



Hình 4 Kiến trúc tổng quát của hệ thống.

1.2. Mô hình phân rã hệ thống



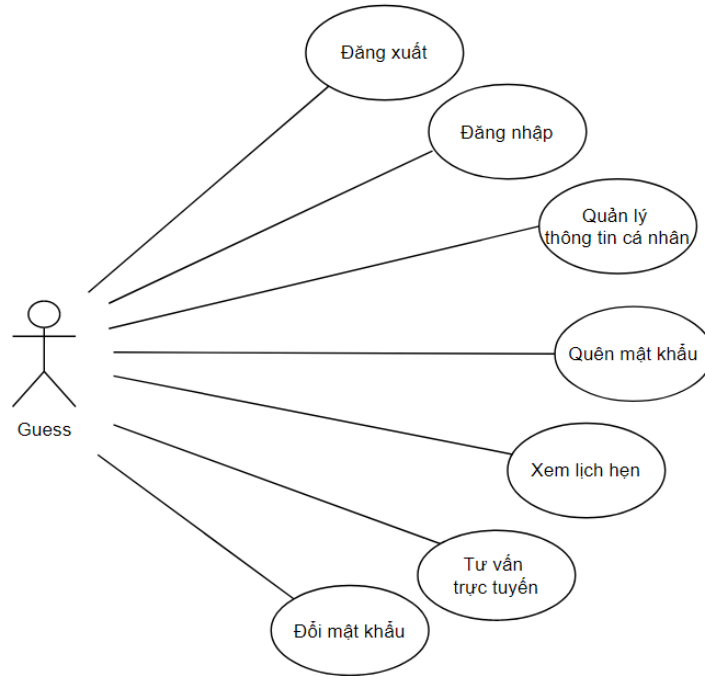
Hình 5 Mô hình phân rã chức năng của hệ thống

2. Các trường hợp sử dụng

Guess là tác nhân cốt lõi của hệ thống, tác nhân này có các trường hợp sử dụng mà các tác nhân khác đều có, mang tính chất trừu tượng để các tác nhân khác có thể kế thừa chứ không hiện thực hóa trong hệ thống.

Bệnh nhân, bác sĩ và nhân viên phòng khám là các tác nhân cụ thể hóa của Guess, tùy vào từng tác nhân mà có những trường hợp sử dụng khác nhau.

2.1. Trường hợp sử dụng của Guess (chức năng chung)



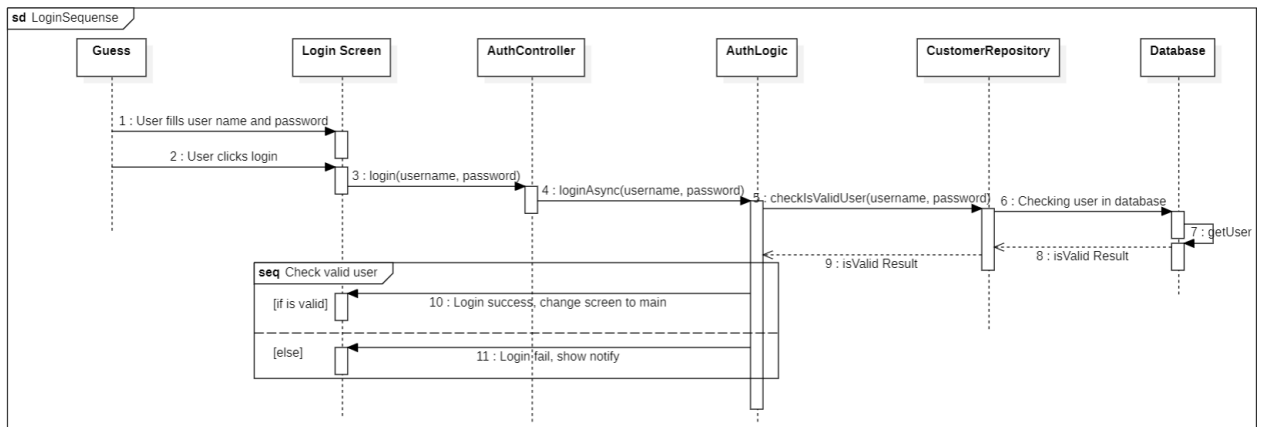
Hình 6 Trường hợp sử dụng của Guess (chức năng chung).

2.1.1. Đăng nhập

Usecase ID	REQ_01
Tên usecase	Đăng nhập
Mô tả	Cho phép xác thực bệnh nhân có tài khoản, bác sĩ và nhân viên phòng khám
Tác nhân	Người dùng
Độ ưu tiên	Cao
Kích hoạt	Khi người dùng muốn truy cập sử dụng các chức năng bên trong hệ thống
Tiền điều kiện	Người dùng chưa đăng nhập
Kết quả	Người dùng được xác thực và vào được hệ thống

Luồng xử lý bình thường	<ol style="list-style-type: none"> 1. Truy cập vào hệ thống 2. Nhập thông tin tài khoản và mật khẩu 3. Nhấp đăng nhập 4. Hệ thống kiểm tra thông tin. <ol style="list-style-type: none"> 4.1. Nếu thông tin hợp lệ. Thông báo đăng nhập thành công, chuyển đến giao diện của nhóm người dùng 4.2. Nếu thông tin không hợp lệ. Hiện cảnh báo
Luồng thay thế/ đặc biệt	

Bảng 1 Trường hợp sử dụng Đăng nhập



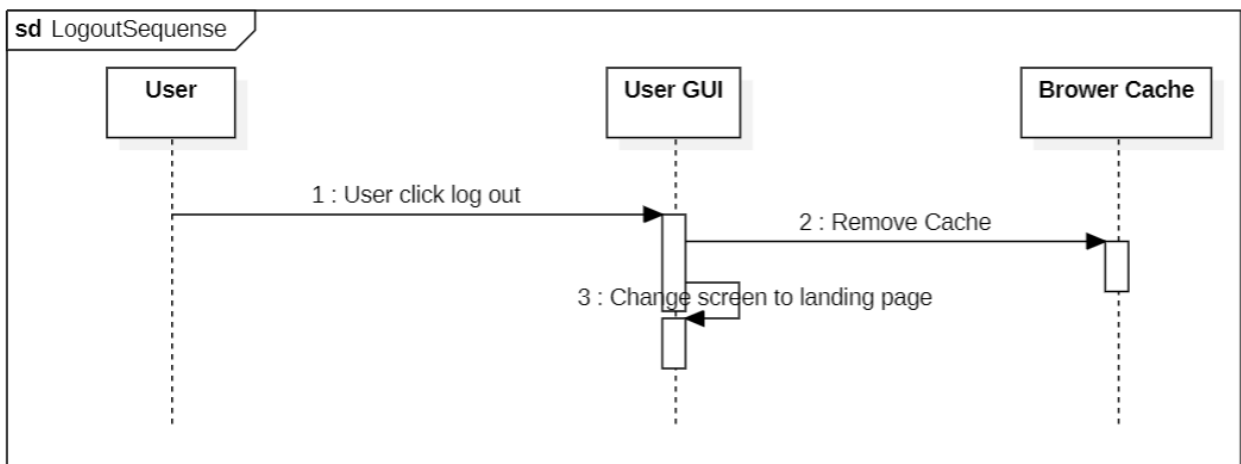
Hình 7. Sơ đồ tuần tự trường hợp sử dụng Đăng nhập.

2.1.2. Đăng xuất

Usecase ID	REQ_02
Tên usecase	Đăng xuất
Mô tả	Đăng xuất người dùng hiện tại
Tác nhân	Người dùng
Độ ưu tiên	Cao
Kích hoạt	Khi người dùng muốn đăng xuất khỏi hệ thống
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống

Kết quả	Người dùng được chuyển đến trang bắt đầu của trang web
Luồng xử lý bình thường	1. Người dùng nhấp đăng xuất 2. Hệ thống xóa các thông tin trong bộ nhớ tạm, chuyển đến trang ban đầu
Luồng thay thế/ đặc biệt	

Bảng 2 Trường hợp sử dụng Đăng xuất



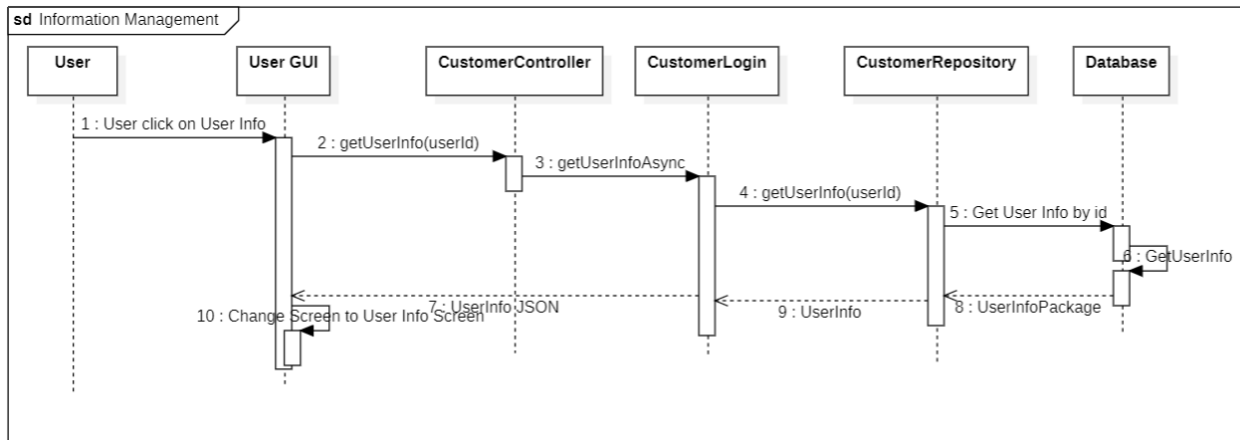
Hình 8. Sơ đồ tuần tự trường hợp sử dụng Đăng xuất

2.1.3. Quản lý thông tin cá nhân

Usecase ID	REQ_03
Tên usecase	Quản lý thông tin cá nhân
Mô tả	Người muốn xem, chỉnh sửa thông tin cá nhân
Tác nhân	Người dùng
Độ ưu tiên	Cao
Kích hoạt	Khi người dùng muốn thay đổi các thông tin được cho phép thay đổi
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Thông tin cá nhân của người dùng được hiển thị/ chỉnh sửa

Luồng xử lý bình thường	<ol style="list-style-type: none"> 1. Người dùng chọn Tiện ích → Cá nhân hoặc Cá nhân trên header của trang web 2. Người dùng có thể <i>Đổi mật khẩu</i> hoặc thay đổi thông tin đang hiển thị 3. Người dùng Nhấp cập nhật 4. Thông tin người dùng được hệ thống cập nhật 5. Hiển thị thông báo cập nhật thành công
Luồng thay thế/đặc biệt	Khi hệ thống xảy ra lỗi, hiển thị thông báo cập nhật lỗi, thử lại sau

Bảng 3 Trường hợp sử dụng Quản lý thông tin cá nhân



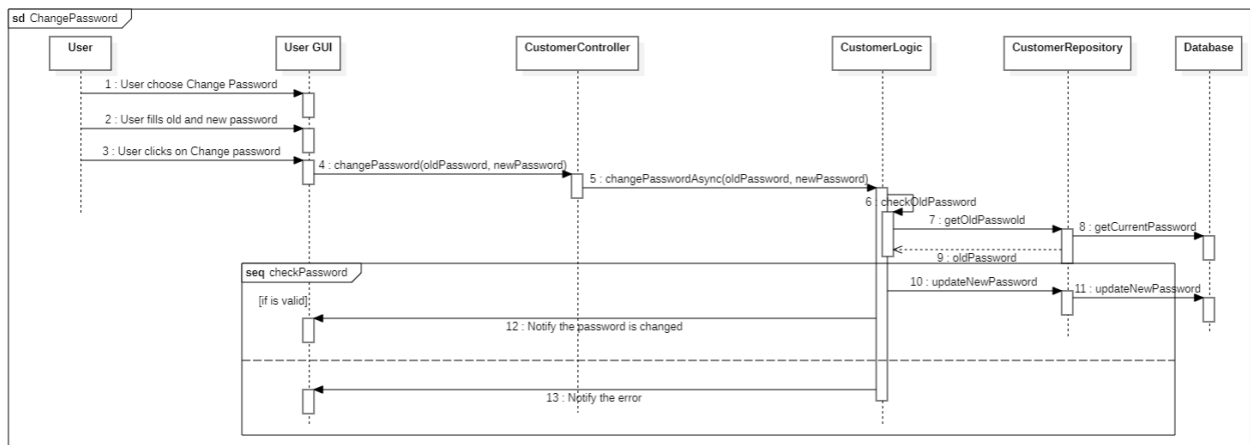
Hình 9. Sơ đồ tuần tự trường hợp sử dụng Quản lý thông tin cá nhân

2.1.4. Đổi mật khẩu

Usecase ID	REQ_04
Tên usecase	Đổi mật khẩu
Mô tả	Người dùng sử dụng chức năng để thay đổi mật khẩu hiện tại
Tác nhân	Người dùng
Độ ưu tiên	Cao
Kích hoạt	Khi người dùng muốn đổi mật khẩu
Tiền điều kiện	Người dùng đã đăng nhập

Kết quả	Mật khẩu của người dùng được thay đổi
Luồng xử lý bình thường	<ol style="list-style-type: none"> 1. Người dùng chọn Tiện ích → Cá nhân hoặc Cá nhân trên header của trang web 2. Người dùng chọn Đổi mật khẩu. 3. Người dùng xác nhận mật khẩu cũ và nhập mật khẩu mới. 4. Người dùng chọn Đổi mật khẩu 5. Hệ thống kiểm tra mật khẩu cũ và mật khẩu mới <ol style="list-style-type: none"> 5.1 Nếu mật khẩu cũ khớp với mật khẩu hiện tại, cập nhật mật khẩu mới, hiện thông báo thành công 5.2 Nếu mật khẩu cũ sai, hiện thông báo lỗi mật khẩu cũ không chính xác.
Luồng thay thế/đặc biệt	Không xử lý, gọi cập nhật mật khẩu nếu tất cả các trường chưa được điền đầy đủ

Bảng 4 Trường hợp sử dụng Đổi mật khẩu



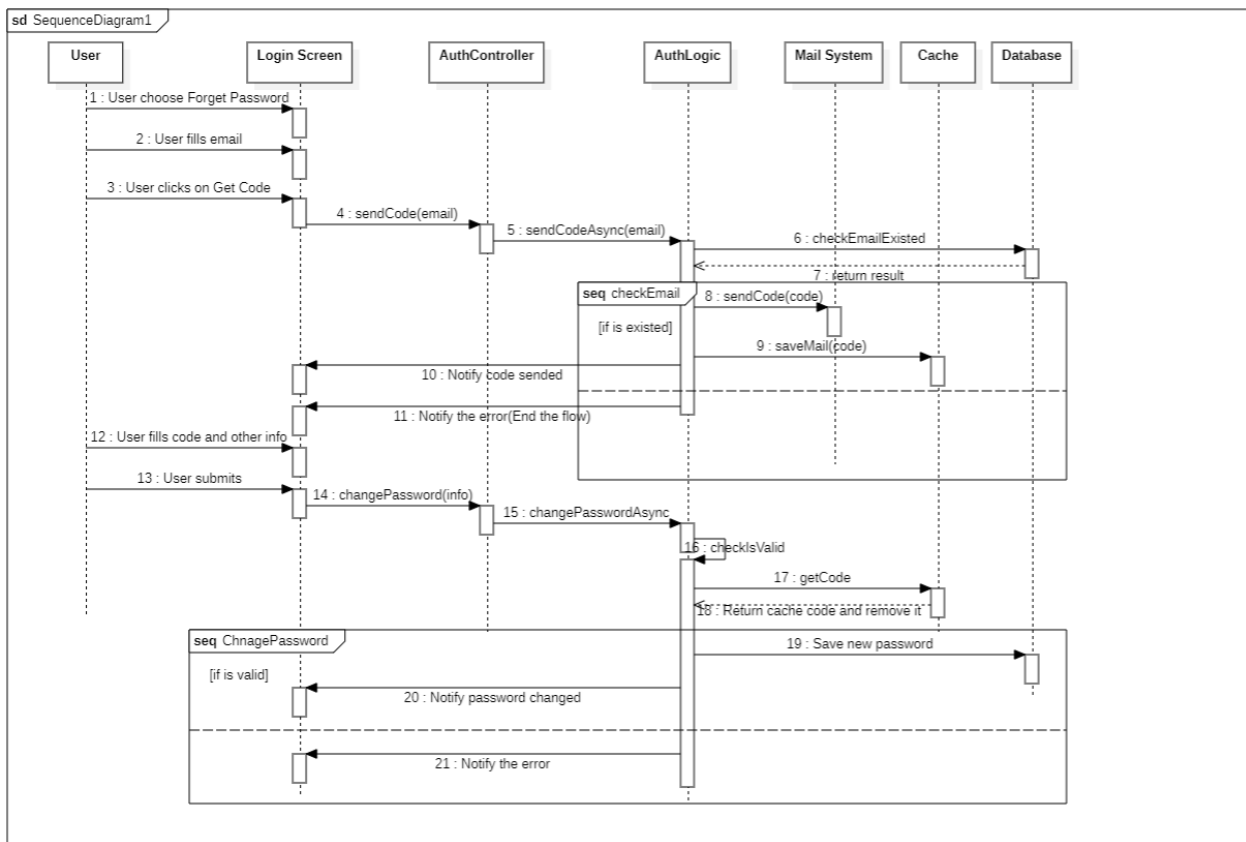
Hình 10. Sơ đồ tuần tự trường hợp sử dụng Đăng xuất

2.1.5. Quên mật khẩu

Usecase ID	REQ_05
Tên usecase	Quên mật khẩu
Mô tả	Khi người dùng quên mật khẩu, thông qua tài khoản email có thể đặt lại mật khẩu mới

Tác nhân	Người dùng
Độ ưu tiên	Cao
Kích hoạt	Khi người dùng quên mật khẩu
Tiền điều kiện	Người dùng không đăng nhập và quên mật khẩu
Kết quả	Người dùng được cấp mật khẩu mới
Luồng xử lý bình thường	<ol style="list-style-type: none"> 1. Người dùng chọn Đăng nhập 2. Người dùng nhập email của tài khoản, chọn Gửi mã. 3. Người dùng chọn Quên mật khẩu. 4. Người dùng kiểm tra mail và lấy được mã xác thực 5. Người dùng nhập các thông tin yêu cầu. 6. Người dùng chọn Đặt lại mật khẩu 7. Hệ thống kiểm tra các thông tin là hợp lệ, đóng popup quên mật khẩu
Luồng thay thế/ đặc biệt	Nếu các thông tin người dùng không hợp lệ hoặc bỏ trống, hiện thông báo lỗi

Bảng 5 Trường hợp sử dụng Quên mật khẩu



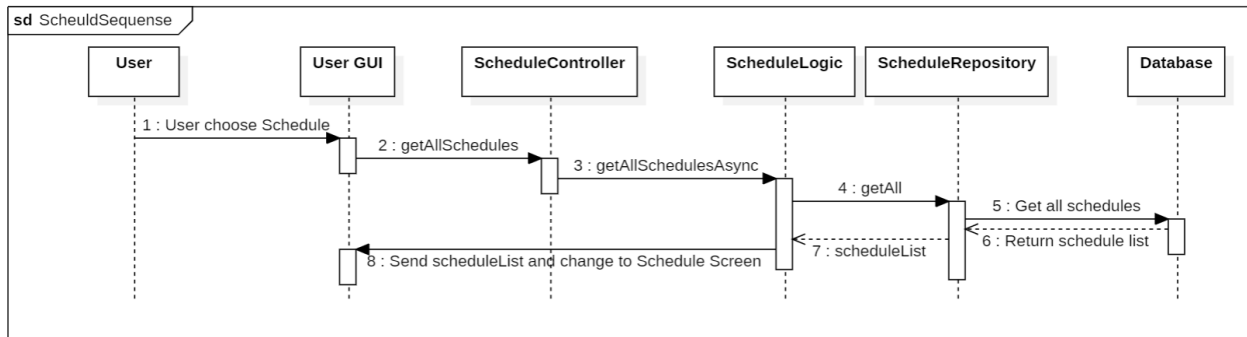
Hình 11. Sơ đồ tuần tự trường hợp sử dụng Quên mật khẩu

2.1.6. Xem lịch hẹn

Usecase ID	REQ_06
Tên usecase	Xem lịch hẹn
Mô tả	Người dùng có thể xem lịch hẹn của phòng khám
Tác nhân	Người dùng
Độ ưu tiên	Cao
Kích hoạt	Khi người dùng muốn xem lịch/ đặt lịch
Tiền điều kiện	
Kết quả	Người dùng xem được lịch hẹn
Luồng xử lý bình thường	1. Người dùng chọn Đặt lịch tại header của trang web 2. Người dùng chọn bác sĩ cần xem lịch

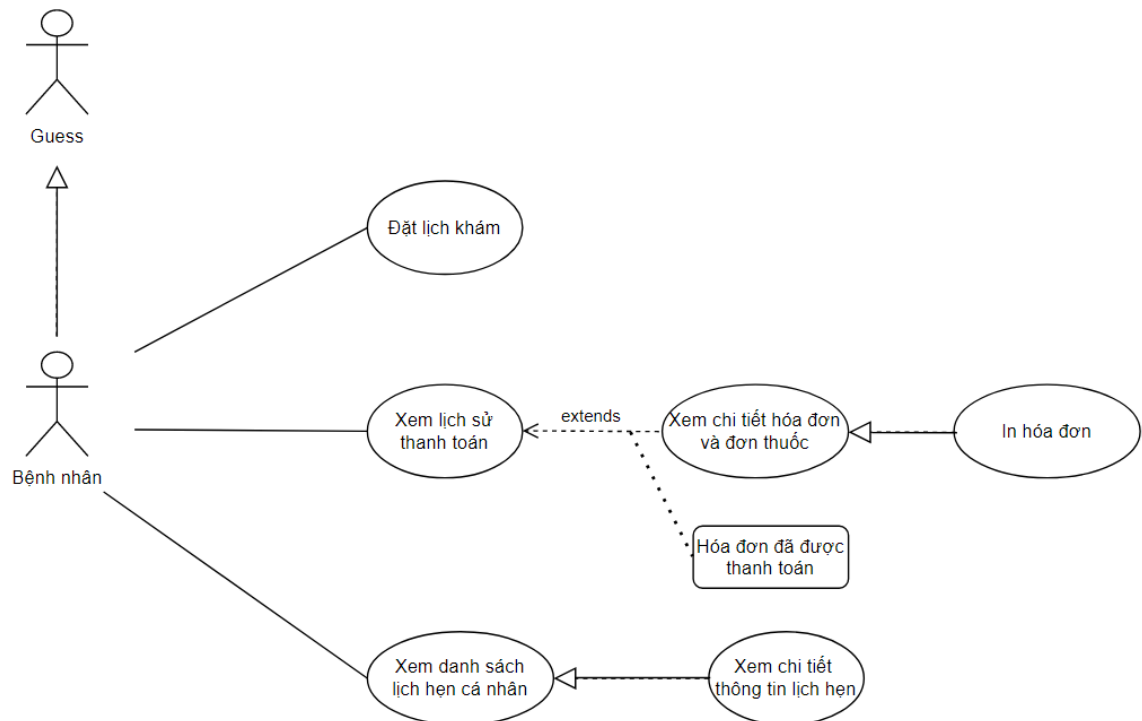
	3. Hệ thống hiển thị lịch hẹn của bác sĩ đã chọn
Luồng thay thế/ đặc biệt	Người dùng có thể thay đổi thời gian, bác sĩ, chế độ xem

Bảng 6 Trường hợp sử dụng Xem lịch hẹn



Hình 12. Sơ đồ tuần tự Xem lịch hẹn

2.2. Trường hợp sử dụng của bệnh nhân

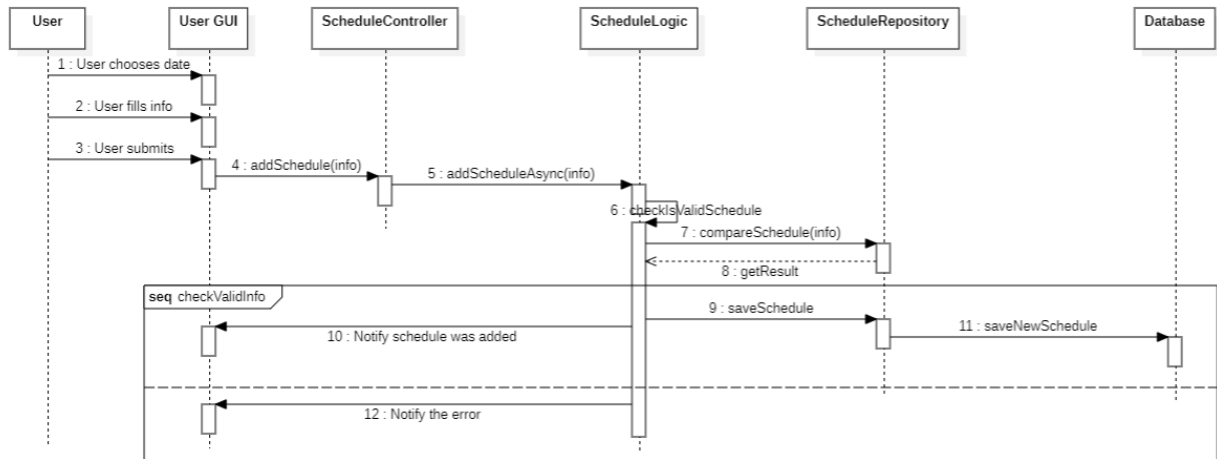


Hình 13 Trường hợp sử dụng của bệnh nhân.

2.2.1. Đặt lịch khám

Usecase ID	REQ_07
Tên usecase	Đặt lịch khám
Mô tả	Được sử dụng khi bệnh nhân có nhu cầu đặt lịch khám bệnh
Tác nhân	Bệnh nhân
Độ ưu tiên	Cao
Kích hoạt	Khi bệnh nhân có nhu cầu đặt lịch khám
Tiền điều kiện	
Kết quả	Lịch hẹn được đặt, hiển thị trong lịch và được gửi đến nhân viên phòng khám
Luồng xử lý bình thường	<ol style="list-style-type: none"> 1. Bệnh nhân chọn Đặt lịch ở header của trang web 2. Bệnh nhân chọn bác sĩ cần đặt lịch hẹn 3. Hệ thống hiển thị lịch hẹn của bác sĩ đã chọn 4. Bệnh nhân chọn một ngày còn thời gian trống để đặt lịch 4. Bệnh nhân điền đầy đủ thông tin cũng như nội dung sẽ khám. 5. Bệnh nhân chọn Đặt lịch hẹn
Luồng thay thế/ đặc biệt	<ol style="list-style-type: none"> 1. Nếu bệnh nhân chọn ngày cách giờ gian hiện tại nhỏ hơn 3 ngày, biểu mẫu đặt lịch sẽ không hiện lên. 2. Bệnh nhân không thể đặt lịch hẹn vào chủ nhật. 3. Bệnh nhân không thể đặt lịch hẹn trùng với lịch hẹn đã có.

Bảng 7 Trường hợp sử dụng Đặt lịch khám

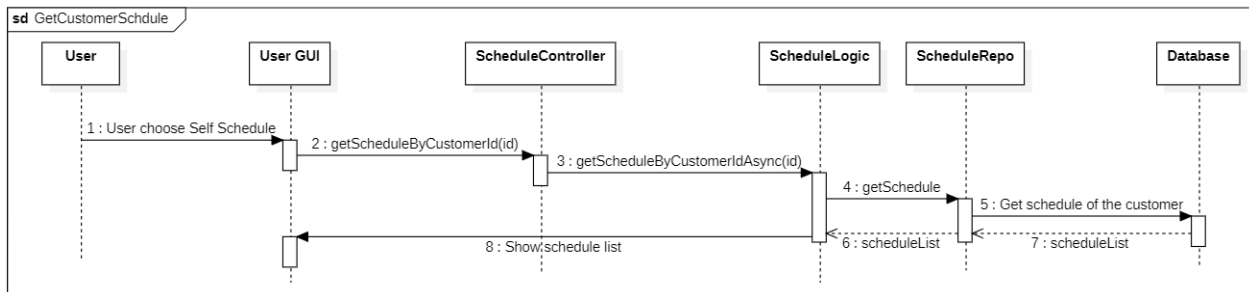


Hình 14. Sơ đồ tuần tự trường hợp sử dụng Đặt lịch khám

2.2.2. Xem lịch khám cá nhân

Usecase ID	REQ_08
Tên usecase	Xem lịch khám cá nhân
Mô tả	Bệnh nhân có thể xem danh sách lịch khám bệnh cá nhân của mình.
Tác nhân	Bệnh nhân
Độ ưu tiên	Cao
Kích hoạt	Bệnh nhân muốn xem danh sách lịch khám bệnh
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Hiển thị danh sách lịch khám bệnh
Luồng xử lý bình thường	1. Bệnh nhân chọn Tiện ích → Lịch của bạn 2. Hệ thống hiển thị danh sách lịch cá nhân
Luồng thay thế/ đặc biệt	

Bảng 8 Trường hợp sử dụng Xem lịch khám cá nhân

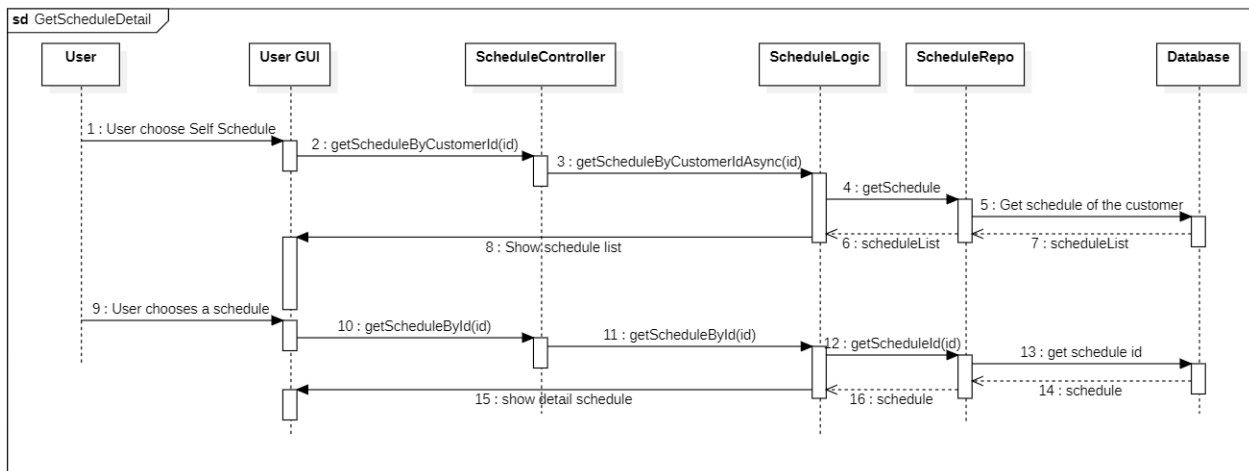


Hình 15. Sơ đồ tuần tự trường hợp sử dụng Xem lịch khám cá nhân.

2.2.3. Xem chi tiết thông tin lịch hẹn

Usecase ID	REQ_09
Tên usecase	Xem chi tiết thông tin lịch hẹn
Mô tả	Bệnh nhân có thể xem chi tiết thông tin lịch hẹn
Tác nhân	Bệnh nhân
Độ ưu tiên	Cao
Kích hoạt	Khi bệnh nhân chọn một lịch hẹn
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Hiển thị chi tiết lịch hẹn đã chọn
Luồng xử lý bình thường	1. Bệnh nhân chọn Tiện ích → Lịch của bạn 2. Hệ thống hiển thị danh sách lịch cá nhân 3. Bệnh nhân chọn một dòng lịch hẹn để xem chi tiết 4. Hệ thống hiển thị thông tin chi tiết lịch hẹn đã chọn
Luồng thay thế/ đặc biệt	Bệnh nhân có thể xem chi tiết và xóa lịch hẹn nếu lịch hẹn chưa được xác nhận

Bảng 9 Trường hợp sử dụng Xem chi tiết lịch hẹn

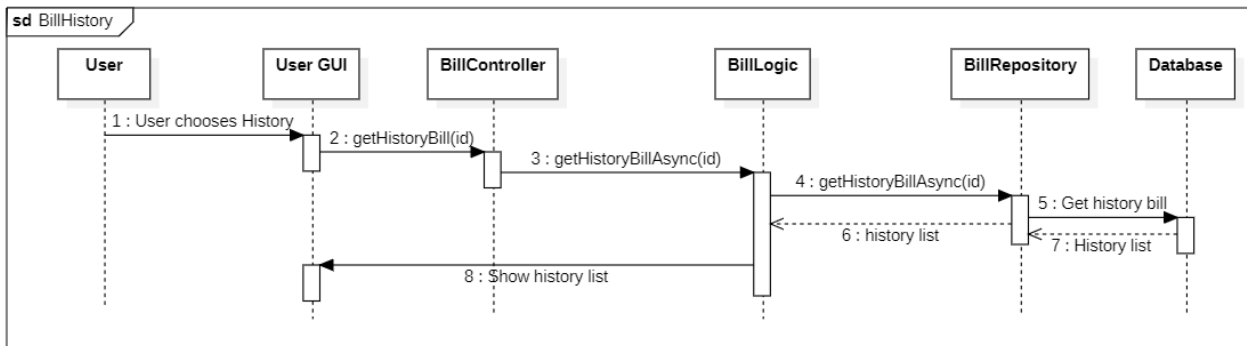


Hình 16. Sơ đồ tuần tự trường hợp sử dụng Xem chi tiết thông tin lịch hẹn.

2.2.4. Xem lịch sử thanh toán

Usecase ID	REQ_10
Tên usecase	Xem lịch sử thanh toán
Mô tả	Bệnh nhân có thể xem lại lịch sử thanh toán sau khi khám chữa bệnh
Tác nhân	Bệnh nhân
Độ ưu tiên	Cao
Kích hoạt	Khi bệnh nhân muốn xem lịch sử thanh toán
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Hiển thị danh sách đang chờ và đã thanh toán
Luồng xử lý bình thường	1. Bệnh nhân chọn Tiện ích → Hóa đơn 2. Hệ thống hiển thị danh sách hóa đơn đang chờ và đã thanh toán
Luồng thay thế/ đặc biệt	

Bảng 10 Trường hợp sử dụng Xem lịch sử thanh toán

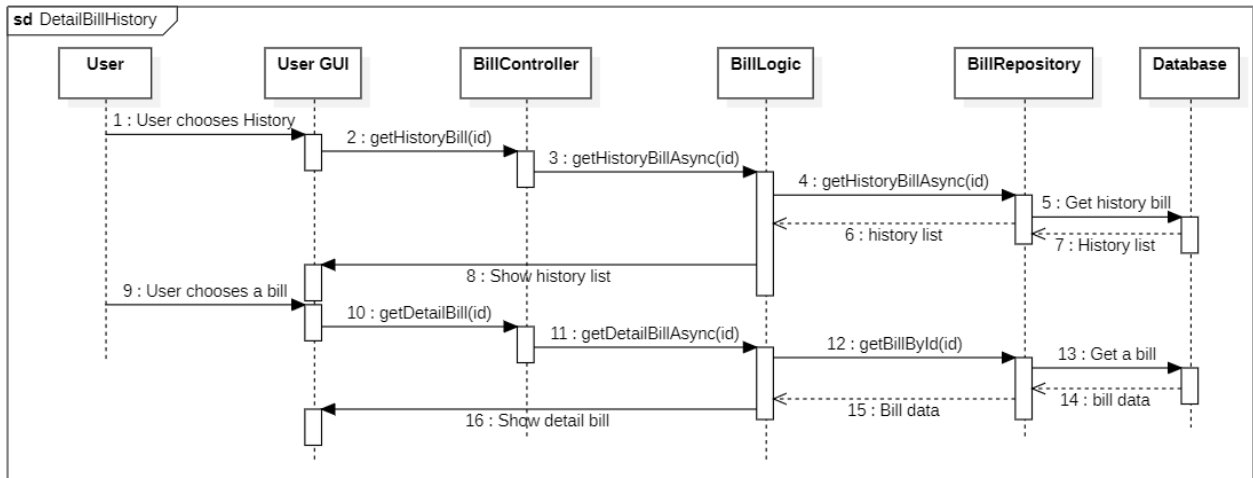


Hình 17. Sơ đồ tuần tự trường hợp sử dụng Lịch sử thanh toán

2.2.5. Xem chi tiết hóa đơn và đơn thuốc

Usecase ID	REQ_11
Tên usecase	Xem chi tiết hóa đơn và đơn thuốc
Mô tả	Khi bệnh nhân muốn xem chi tiết một lịch sử khám chữa bệnh
Tác nhân	Bệnh nhân
Độ ưu tiên	Cao
Kích hoạt	Khi bệnh nhân nhấp vào một lịch sử khám chữa bệnh
Tiền điều kiện	Bệnh nhân đã đăng nhập vào hệ thống Hóa đơn phải trong trạng thái chờ thanh toán hoặc đã thanh toán
Kết quả	Hiển thị thông tin chi tiết bao gồm hóa đơn và toa thuốc
Luồng xử lý bình thường	1. Bệnh nhân chọn Tiện ích → Hóa đơn 2. Hệ thống hiển thị danh sách hóa đơn đang chờ và đã thanh toán 3. Bệnh nhân chọn một lịch sử thanh toán để xem chi tiết 4. Hệ thống hiển thị thông tin chi tiết bao gồm hóa đơn và đơn thuốc
Luồng thay thế/ đặc biệt	

Bảng 11 Trường hợp sử dụng Xem chi tiết hóa đơn và đơn thuốc



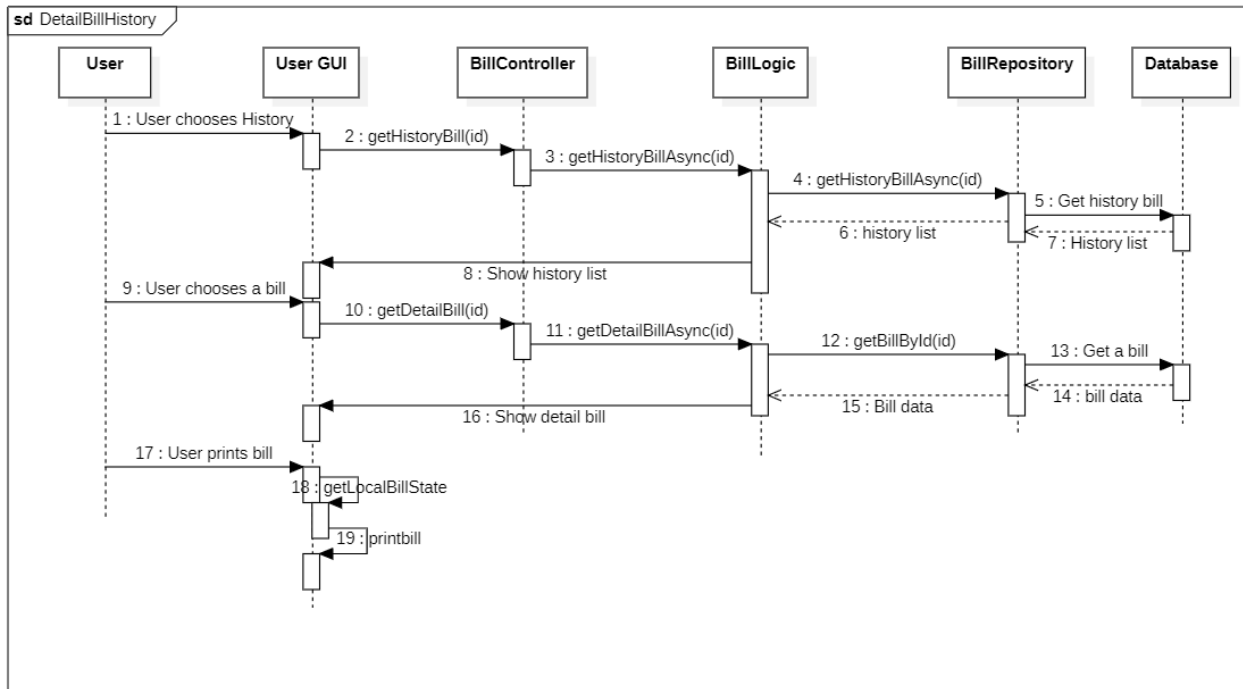
Hình 18. Sơ đồ tuần tự trường hợp sử dụng Xem chi tiết hóa đơn và đơn thuốc

2.2.6. In hóa đơn

Usecase ID	REQ_12
Tên usecase	In hóa đơn
Mô tả	Bệnh nhân có thể sử dụng chức năng này để tải xuống hóa đơn đã thanh toán
Tác nhân	Bệnh nhân
Độ ưu tiên	Trung bình
Kích hoạt	Khi bệnh nhân muốn tải xuống hóa đơn đã thanh toán
Tiền điều kiện	Bệnh nhân đã đăng nhập vào hệ thống Hóa đơn phải trong trạng thái chờ thanh toán hoặc đã thanh toán
Kết quả	Hóa đơn được tải xuống
Luồng xử lý bình thường	1. Bệnh nhân chọn Tiện ích → Hóa đơn 2. Hệ thống hiển thị danh sách hóa đơn đang chờ và đã thanh toán 3. Bệnh nhân chọn một lịch sử thanh toán để xem chi tiết 4. Hệ thống hiển thị thông tin chi tiết bao gồm hóa đơn và đơn thuốc 5. Bệnh nhân chọn Tải xuống hóa đơn 6. Hệ thống chuẩn bị tệp tin pdf của hóa đơn và tải xuống ở thiết bị của người dùng.

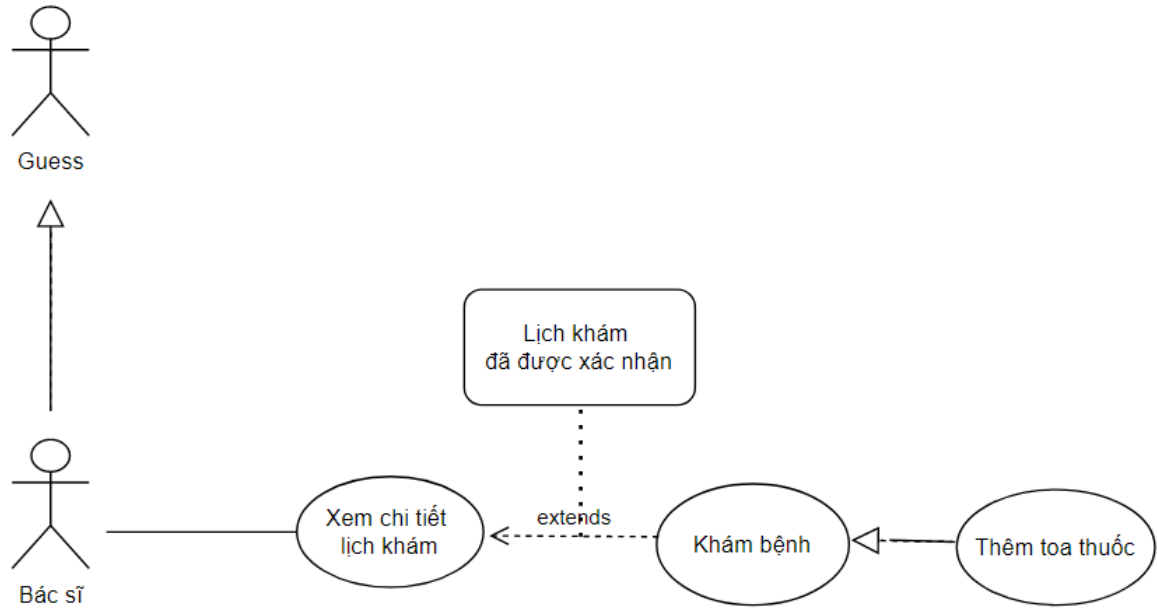
**Luồng thay thế/
đặc biệt**

Bảng 12 Trường hợp sử dụng In hóa đơn



Hình 19. Sơ đồ tuần tự trường hợp sử dụng In hóa đơn.

2.3. Trường hợp sử dụng của bác sĩ



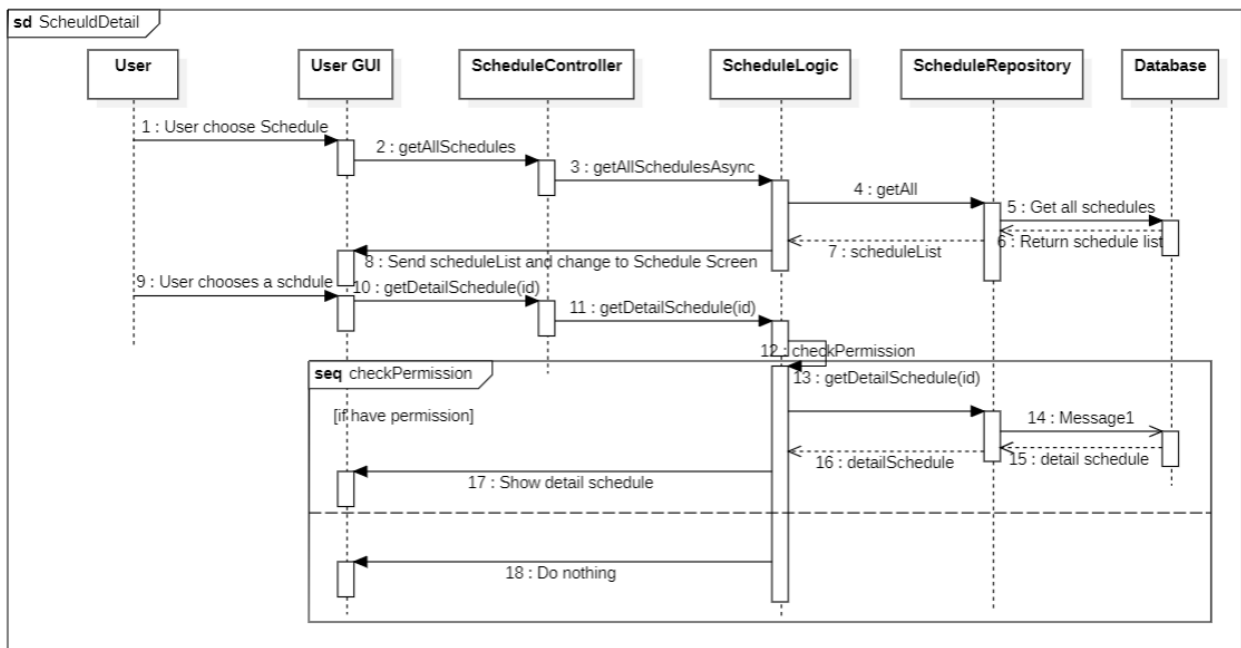
Hình 20 Trường hợp sử dụng của bác sĩ.

2.3.1 Xem chi tiết lịch khám

Usecase ID	REQ_13
Tên usecase	Xem chi tiết lịch khám
Mô tả	Được sử dụng khi bác sĩ muốn xem chi tiết một lịch khám
Tác nhân	Bác sĩ
Độ ưu tiên	Cao
Kích hoạt	Khi bác sĩ muốn xem chi tiết một lịch khám
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Hiển thị chi tiết lịch khám

Luồng xử lý bình thường	<ol style="list-style-type: none"> 1. Tại trang chủ của bác sĩ hoặc nhấp vào Xem lịch tại header của trang web 2. Bác sĩ chuyển sang chế độ lịch tuần 2. Chọn một lịch khám của bệnh nhân cần xem 3. Hệ thống hiển thị chi tiết thông tin của lịch khám đã chọn
Luồng thay thế/đặc biệt	Nút “Bắt đầu khám” khi và chỉ khi lịch khám đang ở trạng thái “Đã xác nhận”

Bảng 13 Trường hợp sử dụng Xem chi tiết lịch khám



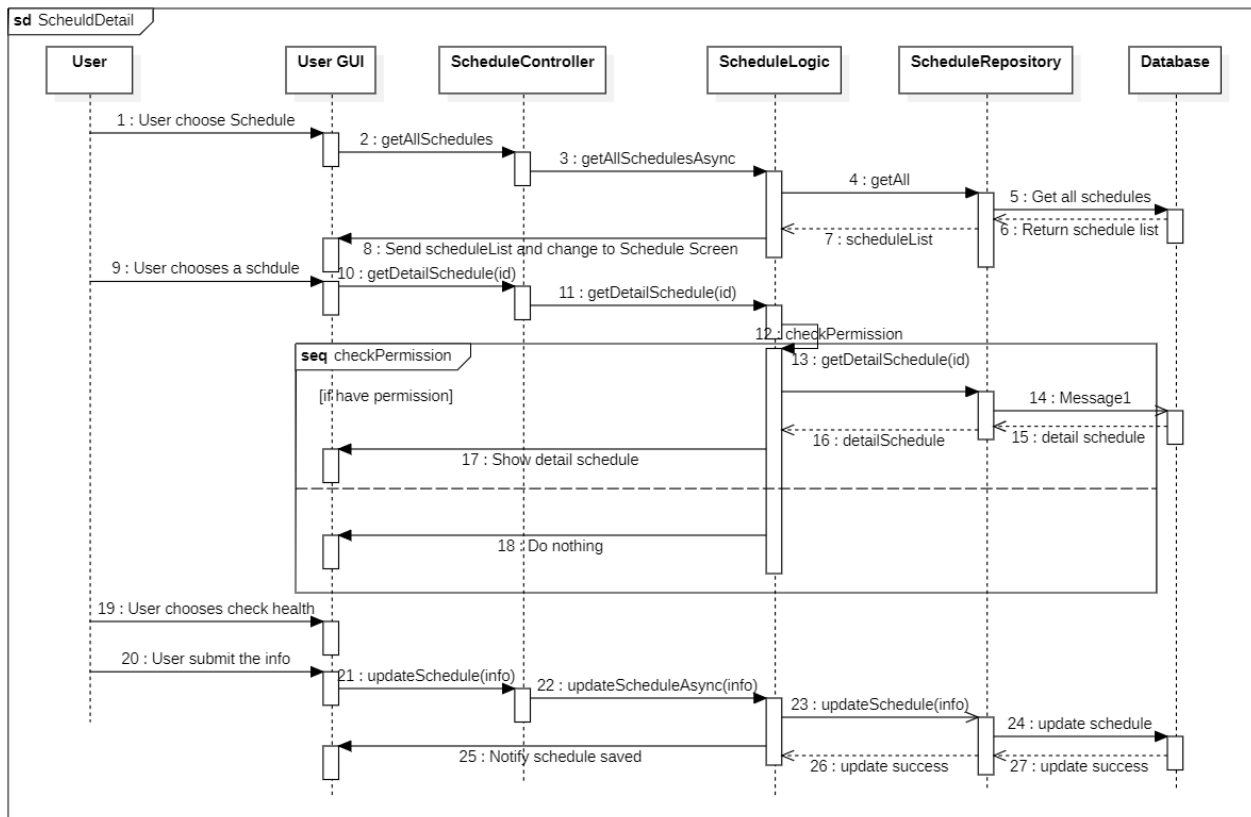
Hình 21. Sơ đồ tuần tự trường hợp sử dụng Xem chi tiết lịch khám.

2.3.2. Khám bệnh

Usecase ID	REQ_14
Tên usecase	Khám bệnh
Mô tả	Lưu trữ lại thông tin phiên khám chữa bệnh của bệnh nhân
Tác nhân	Bác sĩ
Độ ưu tiên	Cao

Kích hoạt	Khi bác sĩ tiến hành khám bệnh cho bệnh nhân
Tiền điều kiện	Đã đăng nhập vào hệ thống Lịch khám ở trạng thái “Đã xác nhận”
Kết quả	Hiển thị biểu mẫu khám bệnh, bác sĩ có thể điền thông tin và xác nhận
Luồng xử lý bình thường	1. Bác sĩ chọn “Bắt đầu khám” 2. Bác sĩ nhập đầy đủ thông tin khám chữa bệnh 3. Bác sĩ chọn xác nhận nếu đã hoàn thành phiên khám chữa bệnh
Luồng thay thế/ đặc biệt	Trong quá trình khám chữa bệnh, bác sĩ có thể thêm hoặc không thêm toa thuốc (kê đơn)

Bảng 14 Trường hợp sử dụng Khám bệnh



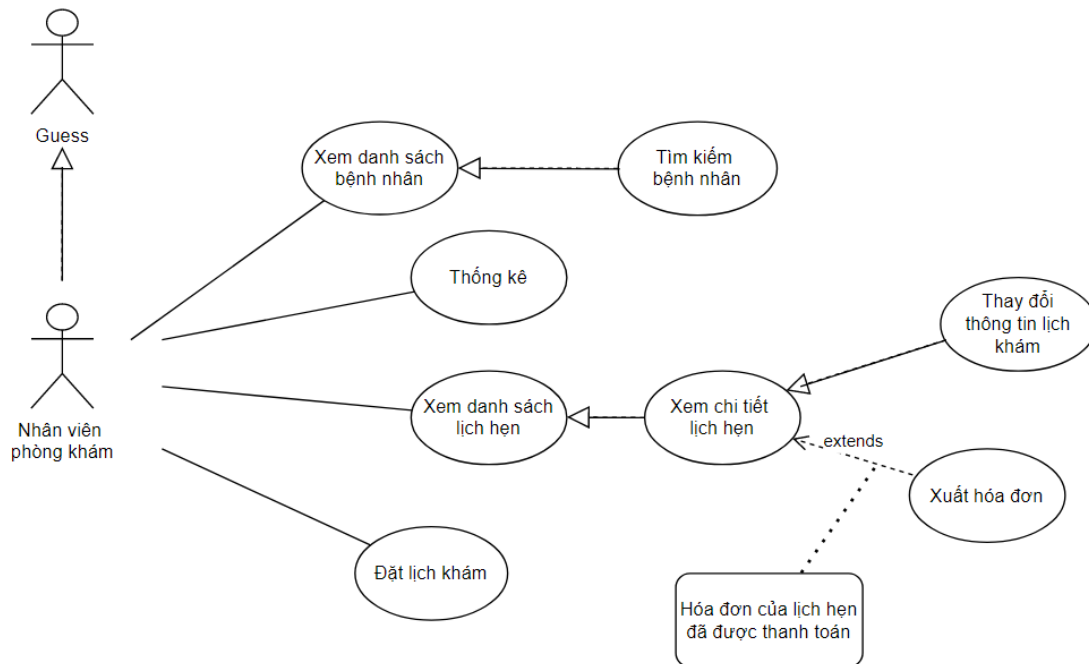
Hình 22. Sơ đồ tuần tự trường hợp sử dụng Khám bệnh.

2.3.3. Thêm toa thuốc (luồng bổ sung cho Khám bệnh)

Usecase ID	REQ_15
Tên usecase	Thêm toa thuốc
Mô tả	Bác sĩ có thể thêm toa thuốc cho bệnh nhân
Tác nhân	Bác sĩ
Độ ưu tiên	Cao
Kích hoạt	Trong quá trình khám bệnh, yêu cầu cần toa thuốc
Tiền điều kiện	Đang thực hiện khám bệnh
Kết quả	Toa thuốc được thêm vào
Luồng xử lý bình thường	<ol style="list-style-type: none"> 1. Bác sĩ nhập tên thuốc 2. Hệ thống gợi ý các tên thuốc theo từ khóa đã nhập 3. Bác sĩ chọn một trong những thuốc đã gợi ý, nhập số lượng, nhập ghi chú về thuốc cho bệnh nhân (nếu có hệ thống quản lý chính của phòng khám thì sẽ chịu trách nhiệm bổ sung thông tin thuốc) 4. Bác sĩ nhấp vào nút “+” để thêm thuốc vừa nhập. 5. Quay lại bước 1.
Luồng thay thế/ đặc biệt	

Bảng 15 Trường hợp sử dụng Thêm toa thuốc

2.4 Trường hợp sử dụng của nhân viên phòng khám



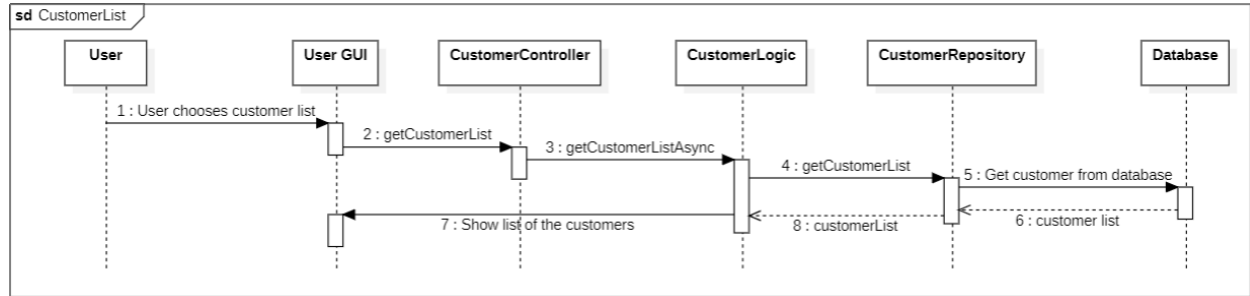
Hình 23 Trường hợp sử dụng của nhân viên phòng khám.

2.3.1 Xem danh sách bệnh nhân

Usecase ID	REQ_16
Tên usecase	Xem danh sách bệnh nhân
Mô tả	Nhân viên phòng khám có thể xem danh sách bệnh nhân
Tác nhân	Nhân viên
Độ ưu tiên	Cao
Kích hoạt	Khi nhân viên muốn xem danh sách nhân viên phòng khám
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Hiển thị danh sách bệnh nhân
Luồng xử lý bình thường	1. Nhân viên chọn Công cụ → Bệnh nhân

	2. Hệ thống hiển thị danh sách bệnh nhân đã có tài khoản tại phòng khám
Luồng thay thế/ đặc biệt	

Bảng 16 Trường hợp sử dụng Xem danh sách bệnh nhân



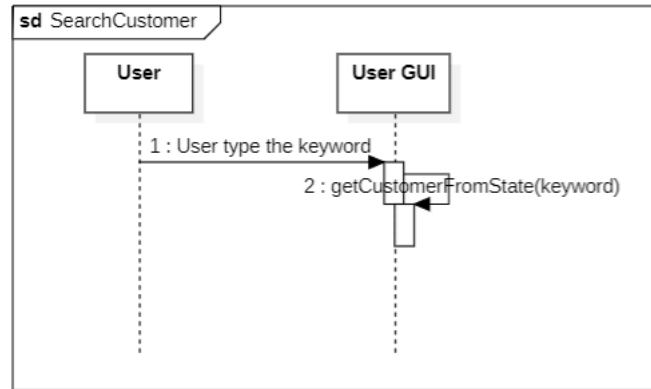
Hình 24. Sơ đồ tuần tự trường hợp sử dụng Xem danh sách bệnh nhân.

2.3.2. Tìm kiếm bệnh nhân

Usecase ID	REQ_17
Tên usecase	Tìm kiếm bệnh nhân
Mô tả	Nhân viên có thể tìm kiếm thông tin bệnh nhân
Tác nhân	Nhân viên
Độ ưu tiên	Thấp
Kích hoạt	Khi nhân viên có nhu cầu tìm kiếm bệnh nhân
Tiền điều kiện	Đã đăng nhập hệ thống
Kết quả	Hiển thị danh sách bệnh nhân khớp với từ khóa, ngược lại hiện toàn bộ danh sách
Luồng xử lý bình thường	1. Nhân viên chọn Công cụ → Bệnh nhân 2. Hệ thống hiển thị danh sách bệnh nhân đã có tài khoản tại phòng khám 3. Nhân viên nhập từ khóa tìm kiếm vào ô tìm kiếm 4. Hệ thống hiển thị danh sách bệnh nhân khớp với từ khóa

Luồng thay thế/ đặc biệt	Khi không thông tin nào của bệnh nhân khớp với từ khóa, hệ thống hiển thị toàn bộ danh sách.
---------------------------------	--

Bảng 17 Trường hợp sử dụng Tìm kiếm bệnh nhân

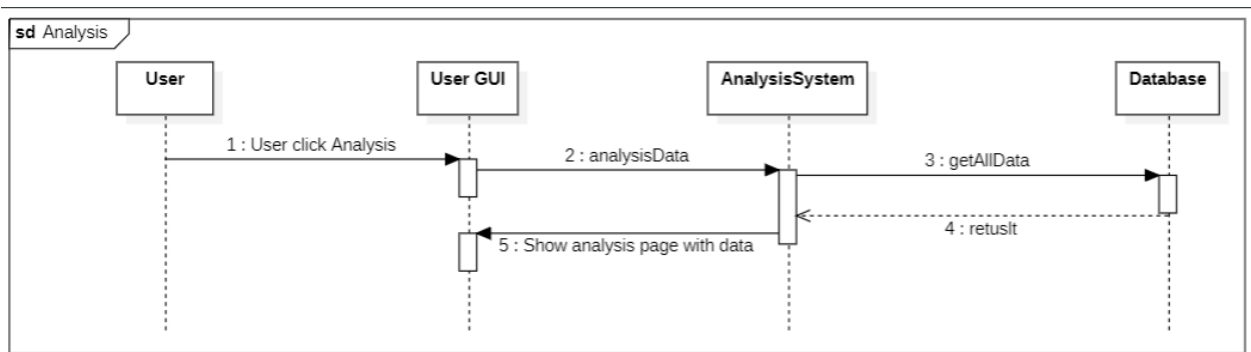


Hình 25. Sơ đồ tuần tự trường hợp sử dụng Tìm kiếm bệnh nhân

2.3.3. Thống kê

Usecase ID	REQ_18
Tên usecase	Thống kê
Mô tả	Nhân viên có thể xem thống kê của phòng khám
Tác nhân	Nhân viên
Độ ưu tiên	Trung bình
Kích hoạt	Khi nhân viên muốn xem thống kê về phòng khám
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Hiển thị trang thống kê
Luồng xử lý bình thường	1. Nhân viên chọn Thống kê tại header của trang web 2. Hệ thống hiển thị trang thống kê
Luồng thay thế/ đặc biệt	Nhân viên có thể xem hiển thị tất cả hoặc theo một khoảng thời gian

Bảng 18 Trường hợp sử dụng Thống kê

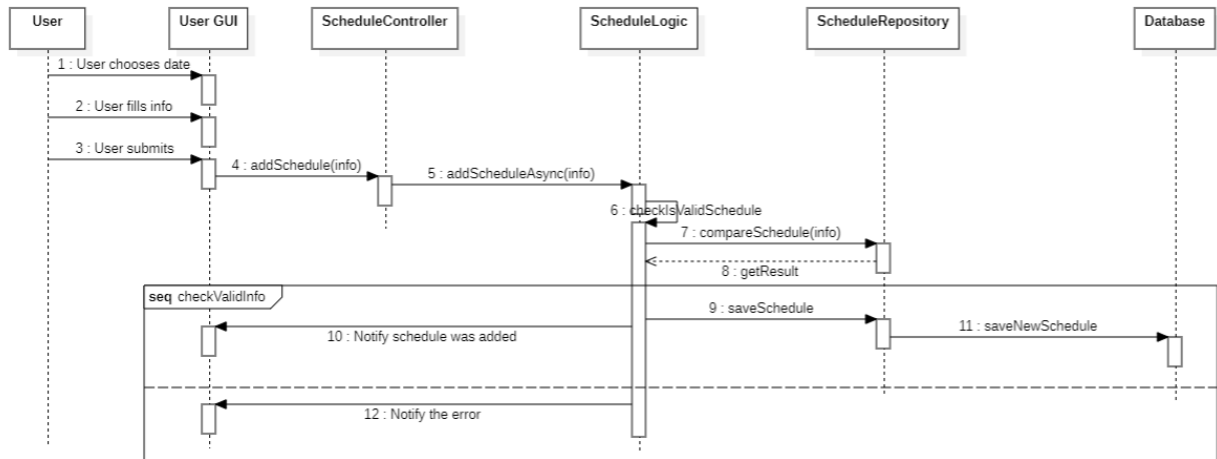


Hình 26. Sơ đồ tuần tự trường hợp sử dụng Thống kê

2.3.4 Đặt lịch khám

Usecase ID	REQ_19
Tên usecase	Đặt lịch khám
Mô tả	Nhân viên sử dụng chức năng này để đặt lịch cho bệnh nhân đến khám trực tiếp
Tác nhân	Nhân viên
Độ ưu tiên	Cao
Kích hoạt	Khi nhân muốn đặt lịch cho bệnh nhân
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Lịch được đặt cho bệnh nhân
Luồng xử lý bình thường	Tương tự như luồng xử lý REQ_07
Luồng thay thế/ đặc biệt	

Bảng 19 Trường hợp sử dụng Đặt lịch khám

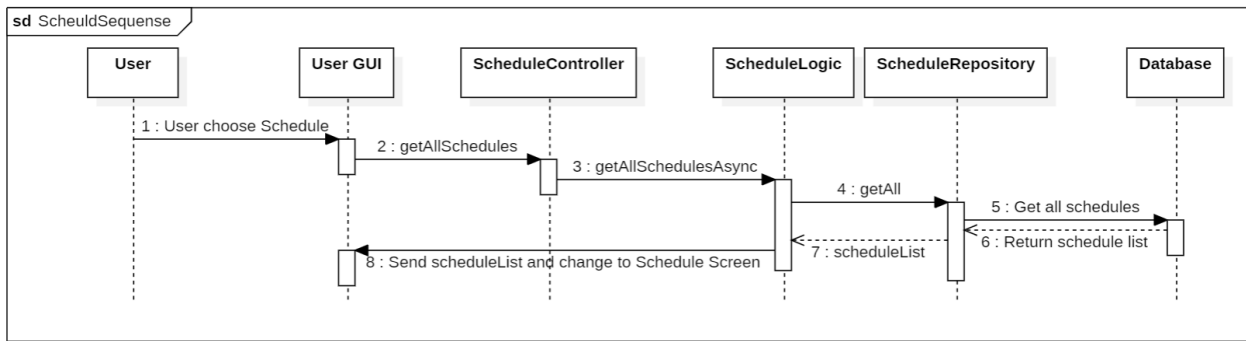


Hình 27. Sơ đồ tuần tự trường hợp sử dụng Đặt lịch khám

2.3.5 Xem danh sách lịch hẹn

Usecase ID	REQ_20
Tên usecase	Xem danh sách lịch hẹn
Mô tả	Nhân viên có thể xem tất cả danh sách lịch hẹn
Tác nhân	Nhân viên
Độ ưu tiên	Cao
Kích hoạt	Khi nhân viên cần xem danh sách lịch hẹn
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Hiển thị danh sách lịch hẹn
Luồng xử lý bình thường	Lịch hẹn hiển thị tại trang chủ của nhân viên hoặc khi nhân viên nhấp vào logo của phòng khám
Luồng thay thế/ đặc biệt	

Bảng 20 Trường hợp sử dụng Xem danh sách lịch hẹn

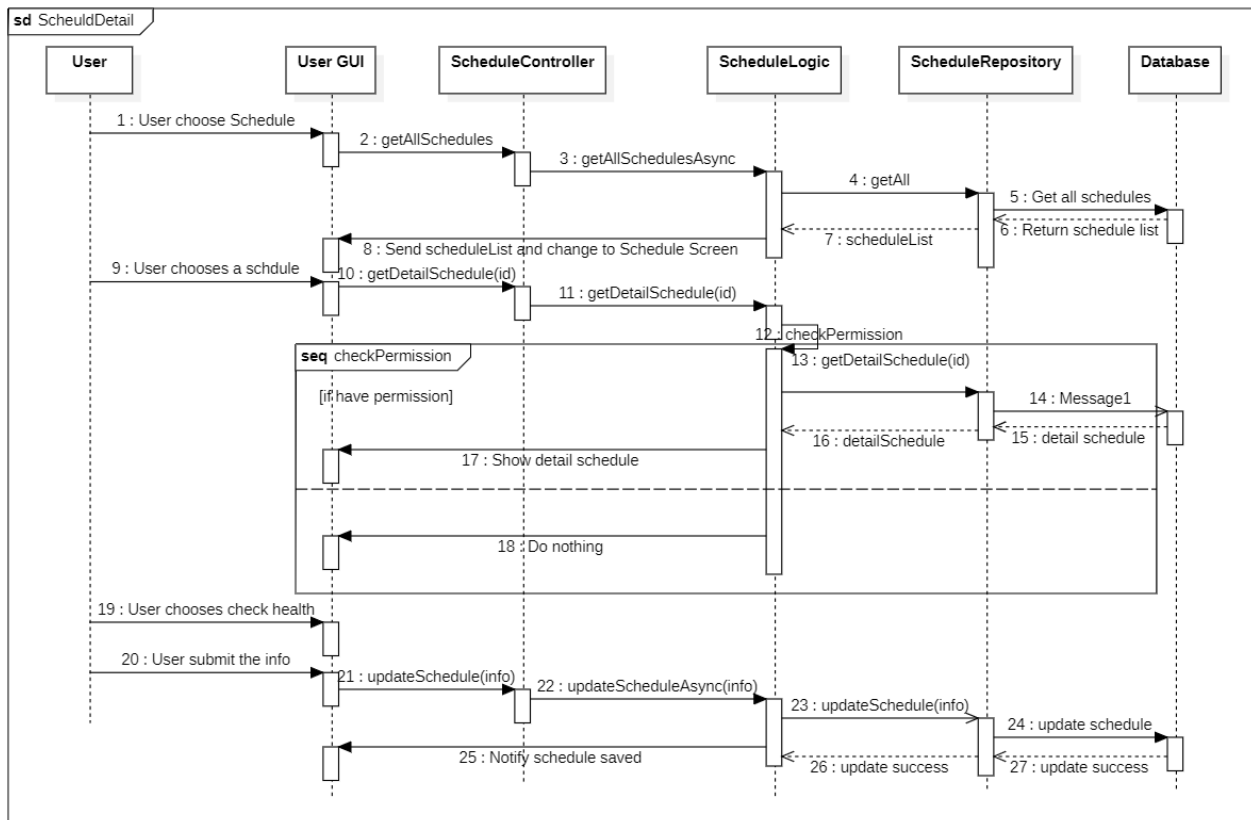


Hình 28. Sơ đồ tuần tự trường hợp sử dụng Xem danh sách lịch hẹn

2.3.6 Xem chi tiết lịch hẹn

Usecase ID	REQ_21
Tên usecase	Xem chi tiết lịch hẹn
Mô tả	Nhân viên có thể chi tiết của lịch hẹn, thay đổi thông tin lịch hẹn tùy vào điều kiện ràng buộc.
Tác nhân	Nhân viên
Độ ưu tiên	Cao
Kích hoạt	Khi nhân viên cần xem, chỉnh sửa lịch hẹn
Tiền điều kiện	Đã đăng nhập vào hệ thống Bệnh nhân đã đặt lịch hẹn
Kết quả	Nhân viên có thể xem/ chỉnh sửa lịch hẹn
Luồng xử lý bình thường	1. Nhân viên đứng tại trang chủ hoặc khi nhân viên nhấp vào logo của phòng khám 2. Hệ thống hiển thị danh sách lịch đã được đặt 3. Nhân viên chọn một lịch hẹn cần xem chi tiết
Luồng thay thế/ đặc biệt	Nhân viên có thể thay đổi trạng thái của lịch hẹn - Nhân viên chỉ có thể thay đổi thời gian bắt đầu khám chữa bệnh và thời gian khám khi lịch hẹn ở trạng thái “Đã tạo” - Nhân viên chỉ thêm được các dịch vụ khám khi lịch hẹn ở trạng thái “Chờ kết quả”

Bảng 21 Trường hợp sử dụng Xem chi tiết lịch hẹn



Hình 29. Sơ đồ tuần tự trường hợp sử dụng Xem chi tiết lịch hẹn

2.3.7 Thay đổi thông tin lịch khám (luồng bổ sung cho Xem chi tiết lịch khám của Nhân viên phòng khám)

Usecase ID	REQ_22
Tên usecase	Thay đổi thông tin lịch khám
Mô tả	Nhân viên có thể thay đổi trạng thái của lịch hẹn khi bệnh nhân có nhu cầu
Tác nhân	Nhân viên
Độ ưu tiên	Cao
Kích hoạt	Khi bệnh nhân có nhu cầu thay đổi thời gian khám chữa bệnh
Tiền điều kiện	Đã đăng nhập vào hệ thống
Kết quả	Thông tin mới được cập nhật

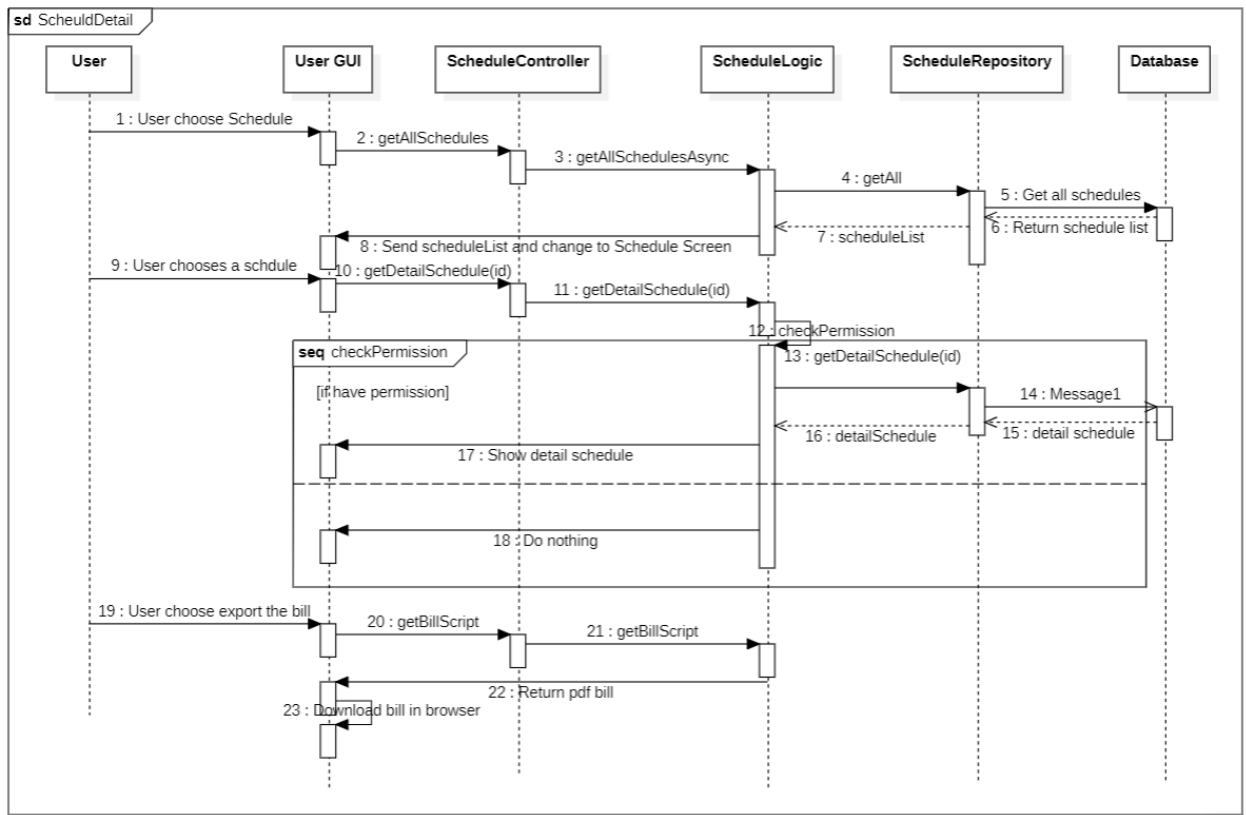
Luồng xử lý bình thường	<ol style="list-style-type: none"> 1. Nhân viên đứng tại trang chủ hoặc khi nhân viên nhấp vào logo của phòng khám 2. Hệ thống hiển thị danh sách lịch đã được đặt 3. Nhân viên chọn một lịch hẹn cần xem chi tiết 4. <ol style="list-style-type: none"> 4.1 Ở trạng thái “Đã tạo”: nhân viên có thể thay đổi thời gian khám và thời gian bắt đầu 4.2 Ở trạng thái “Chờ kết quả”: nhân viên có thể bổ sung các phụ phí cho lịch hẹn
Luồng thay thế/ đặc biệt	

Bảng 22 Trường hợp sử dụng Thay đổi thông tin lịch khám

2.3.8 Xuất hóa đơn

Usecase ID	<u>REQ_23</u>
Tên usecase	<u>Xuất hóa đơn</u>
Mô tả	<u>Xuất hóa đơn thanh toán cho bệnh nhân</u>
Tác nhân	<u>Nhân viên</u>
Độ ưu tiên	<u>Cao</u>
Kích hoạt	<u>Khi lịch hẹn ở trạng thái hoàn tất</u>
Tiền điều kiện	<u>Khi lịch hẹn ở trạng thái hoàn tất</u>
Kết quả	<u>Trình duyệt tải xuống tệp tin pdf</u>
Luồng xử lý bình thường	<ol style="list-style-type: none"> 1. Nhân viên đứng tại trang chủ hoặc khi nhân viên nhấp vào logo của phòng khám 2. Hệ thống hiển thị danh sách lịch đã được đặt 3. Nhân viên chọn một lịch hẹn cần xem chi tiết ở trạng thái “Hoàn tất” 4. <u>Nhân viên chọn Xuất hóa đơn</u>
Luồng thay thế/ đặc biệt	

Bảng 23 Trường hợp sử dụng Xuất hóa đơn



Hình 30.Sơ đồ tuần tự trường hợp sử dụng Xuất hóa đơn.

3. Kiến trúc hệ thống

Khi client thực hiện một yêu cầu (request) đến server, thông tin sẽ được gửi dưới dạng JSON đến server, định tuyến (router) sẽ phân chia hướng đi cho yêu cầu, sau đó thông qua bộ xử lý trung gian sẽ kiểm tra yêu cầu có hợp lệ hay không.

Client (giao diện phía người dùng) được cài đặt bằng React có chức năng kết xuất giao diện người dùng, xử lý các sự kiện.

Định tuyến (Router) có chức năng phân luồng của các điểm cuối (endpoint) của yêu cầu để server xử lý.

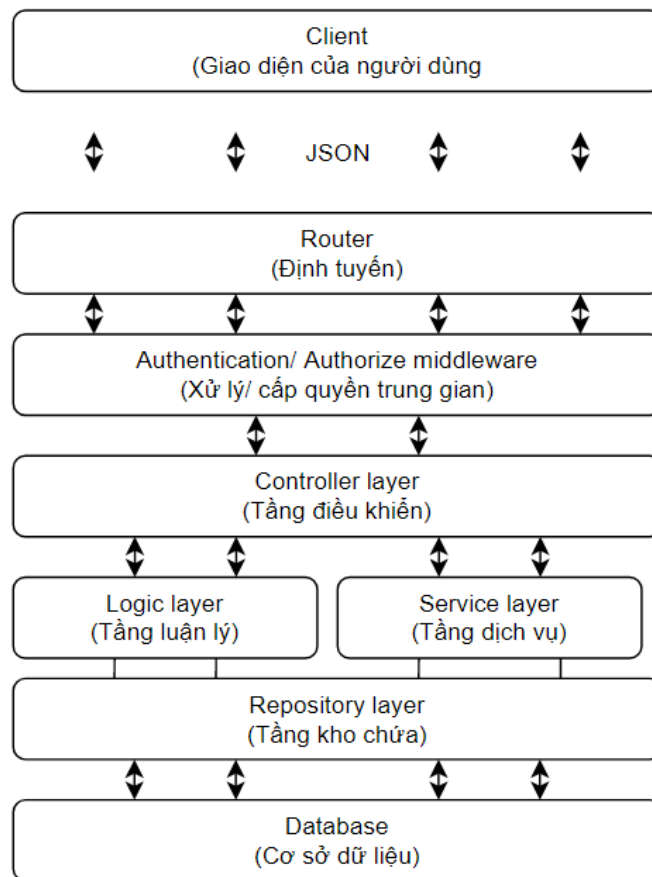
Lớp điều khiển (Controller Layer) có chức năng định nghĩa chức năng cho định tuyến đó, giúp người có kiến thức lập trình không cao có thể đọc, nó có ý nghĩa về mặt nghiệp vụ (business).

Lớp luận lý (Logic layer) là lớp cốt lõi thực hiện xử lý các yêu cầu. Tại đây, lớp luận lý thực hiện gọi các kho chứa dữ liệu ở tầng thấp hơn, các lớp luận lý cùng cấp khác, các dịch vụ đáp ứng cho yêu cầu, sau đó thực hiện xử lý các yêu cầu và trả lại phản hồi (response) cho client nếu cần thiết.

Lớp kho chứa (Repository layer) chịu trách nhiệm giao tiếp với cơ sở dữ liệu bao gồm các hoạt động đọc ghi dữ liệu.

Lớp dịch vụ (Service layer) cung cấp các dịch vụ thực hiện các tác vụ đơn giản cho hệ thống như dịch vụ mail, dịch vụ thời gian thực,...

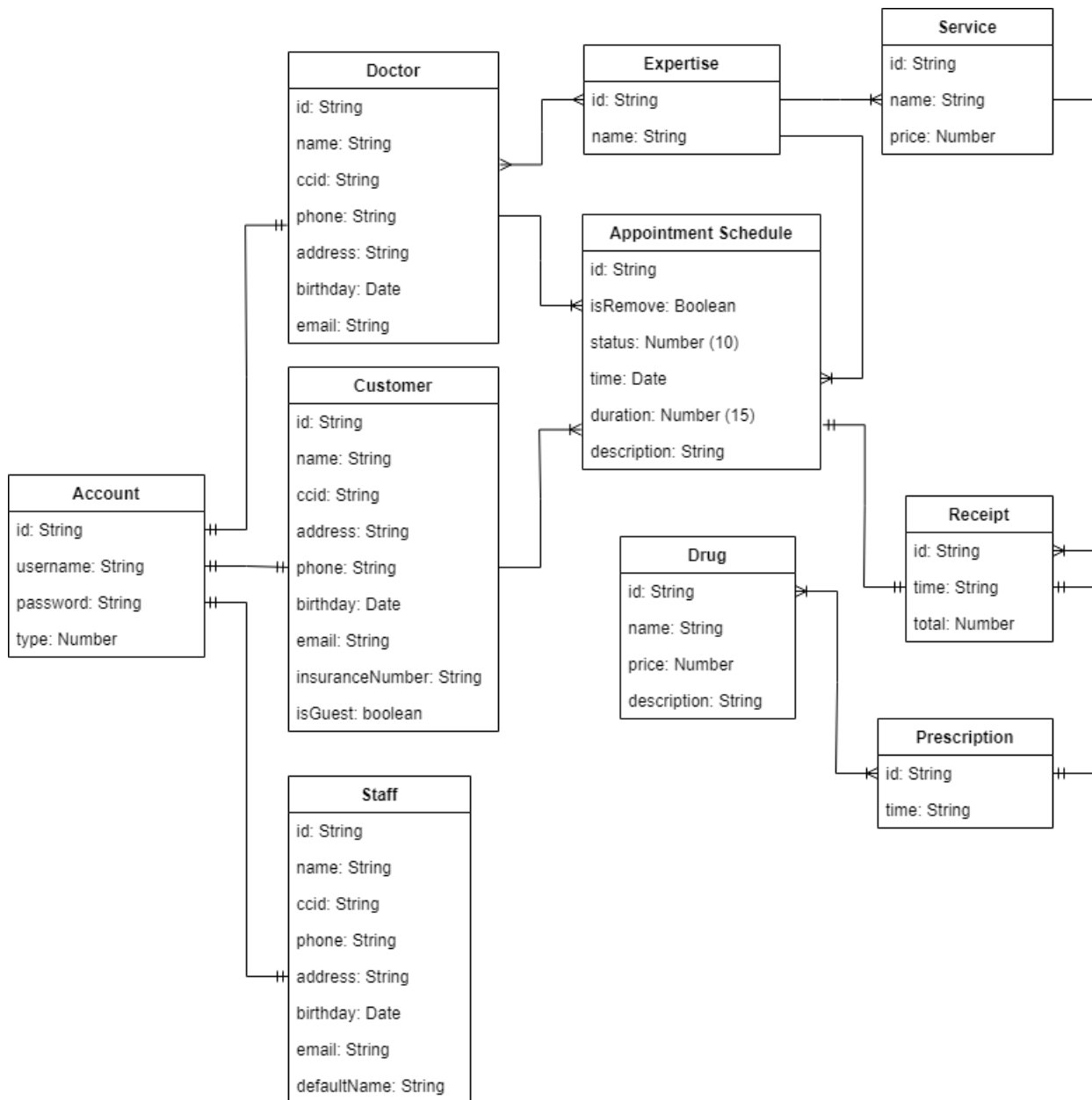
Database sử dụng cơ sở dữ liệu phi quan hệ MongoDB.



Hình 31 Mô hình kiến trúc phân tầng của hệ thống.

4. Thiết kế dữ liệu

4.1. Mô hình dữ liệu



Hình 32 Mô hình dữ liệu mức quan niệm của hệ thống

Để đáp ứng vai trò cho một hệ thống đặt lịch trực tuyến, mô hình dữ liệu tập trung vào bảng Appointment Schedule (Lịch hẹn), các bảng Staff, Customer, Doctor Account, và

các bảng liên quan đến chuyên môn và thuộc mang tính chất tổng quát, nhằm đáp ứng cho bảng dữ liệu chính, các bảng đó sẽ được quản lý bởi một hệ thống quản lý của phòng khám.

Các kiểu dữ liệu được sử dụng tham chiếu theo dạng dữ liệu BSON (Binary JavaScript Object Notation) của cơ sở dữ liệu MongoDB

4.2. Phân tích dữ liệu

Bảng Account (Tài khoản) - accounts

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã định danh cho tài khoản, được tự động tạo bằng một chuỗi uuid
2	username	String		X			Tên tài khoản
3	password	String					Mật khẩu
4	type	Number			10		Loại tài khoản. Mặc định 10 là bệnh nhân, 20 là nhân viên phòng khám và 30 là bác sĩ

Bảng 24 Bảng Account (Tài khoản).

Bảng Doctor (Bác sĩ) - doctors

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã định danh cho bác sĩ
2	name	String					Họ tên
3	ccid	String		X			CMCD/CCCD
4	address	Number					Địa chỉ của bác sĩ
5	phone	String		X			Số điện thoại
6	birthday	Date					Ngày tháng năm sinh
7	email	String		X			Địa chỉ email
8	accountId	String				X	Mã tài khoản của bác sĩ

Bảng 25 Bảng Doctor (Bác sĩ).

Bảng Customer (Bệnh nhân /Khách hàng) - customers

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã định danh cho bác sĩ
2	name	String					Họ tên
3	ccid	String		X			CMCD/CCCD
4	address	Number					Địa chỉ của bác sĩ
5	phone	String		X			Số điện thoại
6	birthday	Date					Ngày tháng năm sinh
7	email	String		X			Địa chỉ email
8	assuranceNumber	String					Mã số bảo hiểm
9	isGuest	Boolean			true		Xác định bệnh nhân có tài khoản hay không
10	accountId	String					Mã tài khoản

Bảng 26 Bảng Customer (Bệnh nhân / Khách hàng)

Bảng Staff (Nhân viên phòng khám) - staffs

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã định danh cho bác sĩ
2	name	String					Họ tên
3	ccid	String		X			CMCD/CCCD
4	address	Number					Địa chỉ của bác sĩ
5	phone	String		X			Số điện thoại
6	birthday	Date					Ngày tháng năm sinh
7	email	String		X			Địa chỉ email
8	defaultName	String			name		Tên thay thế cho nhân viên
	accountId	String		X			Mã tài khoản

Bảng 27 Bảng Staff (Nhân viên phòng khám)

Bảng Expertise (Chuyên môn) - expertise

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã chuyên môn
2	name	String					Tên chuyên môn

Bảng 28 Bảng Expertise (Chuyên môn)

Bảng Doctor – Expertise (Bác sĩ – Chuyên môn) – doctors_expertise

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã chuyên môn
2	expertiesId	String				X	Mã chuyên môn
3	doctorId	String				X	Mã bác sĩ

Bảng 29 Bảng Doctor - Expertise

Bảng Service (Dịch vụ)

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã dịch vụ
2	name	String					Tên dịch vụ
3	price	Number				X	Giá dịch vụ
4	exportiesId	String				X	Mã chuyên môn

Bảng 30 Bảng Service (Dịch vụ)

Bảng Appointment Schedule (Lịch hẹn) - appointmentschedules

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã chuyên môn
2	isRemove	String			false	X	Tên chuyên môn
3	status	Boolean			10	X	Trạng thái
4	time	Date			15	X	Thời gian bắt đầu
5	duration	Number				X	Thời gian khám
6	description	String					Mô tả
7	doctorId	String				X	Mã bác sĩ
8	customerId	String				X	Mã bệnh nhân

9	expertiseId	String				X	Mã chuyên môn
---	-------------	--------	--	--	--	---	---------------

Bảng 31 Bảng Appointment Schedule (Lịch hẹn)

Bảng Receipt (Hóa đơn)

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã dịch vụ
2	time	Date					Tên dịch vụ
3	total	Number					Giá dịch vụ
4	scheduleId	String				X	Mã lịch hẹn
5	serviceId	String				X	Mã dịch vụ

Bảng 32 Bảng Receipt (Hóa đơn)

Bảng Prescription (Toa thuốc) - prescription

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã định danh toa thuốc
2	time	String		X		X	Thời gian tạo toa thuốc

Bảng 33 Bảng Prescription (Toa thuốc)

Bảng Drug (Thuốc) - drugs

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã định danh cho thuốc, được tự động tạo bằng một chuỗi uuid
2	name	String		X		X	Tên thuốc
3	price	Number				X	Đơn giá của thuốc
4	description	String					Mô tả của thuốc

Bảng 34 Bảng Drug (Thuốc)

Bảng Prescription – Drug (Toa thuốc – thuốc) prescriptions_drugs

STT	Tên thuộc tính	Kiểu	Khóa	Duy nhất	Mặc định	Bắt buộc	Mô tả
1	id	String	X	X	uuid	X	Mã định danh tự sinh
2	prescriptionId	String				X	Mã toa thuốc thuốc
3	drugId	String				X	Mã thuốc
4	note	String					Ghi chú thuốc

Bảng 35 Bảng Prescription - Drug (Toa thuốc - Thuốc)


5. Thiết kế giao diện

Bao gồm các giao diện chức năng chính mà hệ thống cung cấp phục vụ cho từng nhóm người dùng


5.1. Giao diện xem lịch

Khi chọn một bác sĩ trong danh sách bác sĩ, người dùng có thể xem lịch làm việc của bác sĩ đó. Có thể xem lịch ở chế độ xem tháng hoặc xem tuần. Ở chế độ xem lịch tháng, người dùng sẽ thấy số cuộc hẹn trong mỗi ngày, và thời gian còn trống còn lại của bác sĩ. Ở chế độ xem tuần, bệnh nhân sẽ xem được thời gian của các cuộc hẹn, từ đó chọn những khoảng thời gian còn trống để đặt lịch hẹn

Đây là giao diện dùng cho cho bác sĩ, bệnh nhân và nhân viên phòng khám, tuy nhiên, các thông tin chi tiết về lịch khám chỉ bác sĩ có thể xem, bệnh nhân không thể xem chi tiết lịch khám của bệnh nhân khác, nhân viên phòng khám có thể xem các thông tin của lịch khám bệnh qua một chức năng nghiệp vụ khác.

<div>  Trang chính Đặt lịch Đăng nhập </div>					
<div> <div>Lịch khám bệnh</div> <div>Nguyễn Thành Long</div> <div>Lịch tháng</div> <div>← Tháng 5 năm 2022 →</div> </div>					
Thứ hai	Thứ ba	Thứ tư	Thứ năm	Thứ sáu	Thứ bảy
2	3	4	5	6	7
Trống	Trống	Trống	Trống	Trống	Trống
9	10	11	12	13	14
Trống	<div>Số cuộc hẹn</div> <div>1</div> <div>Số giờ còn lại</div> <div>7.25h</div>	Trống	Trống	<div>Số cuộc hẹn</div> <div>1</div> <div>Số giờ còn lại</div> <div>7.25h</div>	<div>Số cuộc hẹn</div> <div>1</div> <div>Số giờ còn lại</div> <div>7.25h</div>
16	17	18	19	20	21
<div>Số cuộc hẹn</div> <div>4</div> <div>Số giờ còn lại</div> <div>6.25h</div>	<div>Số cuộc hẹn</div> <div>4</div> <div>Số giờ còn lại</div> <div>6h</div>	<div>Số cuộc hẹn</div> <div>3</div> <div>Số giờ còn lại</div> <div>6h</div>	<div>Số cuộc hẹn</div> <div>3</div> <div>Số giờ còn lại</div> <div>6.5h</div>	<div>Số cuộc hẹn</div> <div>2</div> <div>Số giờ còn lại</div> <div>6.75h</div>	<div>Số cuộc hẹn</div> <div>3</div> <div>Số giờ còn lại</div> <div>6h</div>
23	24	25	26	27	28

Hình 33 Giao diện xem lịch hẹn theo tháng

<div>  Trang chính Đặt lịch Đăng nhập </div>					
<div> <div>Lịch khám bệnh</div> <div>Nguyễn Thành Long</div> <div>Lịch tuần</div> <div>← Tuần 3 →</div> <div>← Tháng 5 năm 2022 →</div> </div>					
Thứ hai - 16	Thứ ba - 17	Thứ tư - 18	Thứ năm - 19	Thứ sáu - 20	Thứ bảy - 21
<div>7:30 - 7:45</div> <div>Đã tạo</div> <div>Nguyễn Thanh Kha</div>	<div>7:30 - 7:45</div> <div>Đã tạo</div> <div>Đỗ Duy Thanh</div>	<div>7:30 - 8:00</div> <div>Đã tạo</div> <div>Trịnh Lộc Uyển</div>	<div>7:30 - 7:45</div> <div>Đã tạo</div> <div>Nguyễn Ngọc Ngà</div>	<div>7:30 - 8:15</div> <div>Đã tạo</div> <div>Nguyễn Kim Ngân</div>	<div>7:30 - 8:00</div> <div>Đã tạo</div> <div>Phạm Hoàng Minh</div>
<div>7:45 - 8:00</div> <div>Đã tạo</div> <div>Thạch Chí Tâm</div>	<div>7:45 - 8:15</div> <div>Đã tạo</div> <div>Đỗ Ngọc Lan</div>		<div>7:45 - 8:15</div> <div>Đã tạo</div> <div>Phạm Hà My</div>		
<div>8:00 - 8:30</div> <div>Đã tạo</div> <div>Lê Duy Hải</div>	<div>8:15 - 9:00</div> <div>Đã tạo</div> <div>Lê Thanh Ngân</div>	<div>8:00 - 8:45</div> <div>Đã tạo</div> <div>Nguyễn Ngọc Duy</div>	<div>8:15 - 9:00</div> <div>Đã tạo</div> <div>Thạch Chí Tâm</div>	<div>8:15 - 8:45</div> <div>Đã tạo</div> <div>Nguyễn Thị Thị</div>	<div>8:00 - 8:45</div> <div>Đã tạo</div> <div>Thạch Chí Tâm</div>
<div>8:30 - 9:15</div> <div>Đã tạo</div> <div>Vũ Ngọc Thắng</div>	<div>9:00 - 9:30</div> <div>Đã tạo</div> <div>Trần Hoàng Thu</div>	<div>8:45 - 9:30</div> <div>Đã tạo</div> <div>Nguyễn Thị Hiền</div>			<div>8:45 - 9:30</div> <div>Đã tạo</div> <div>Lê Ngọc Thu</div>

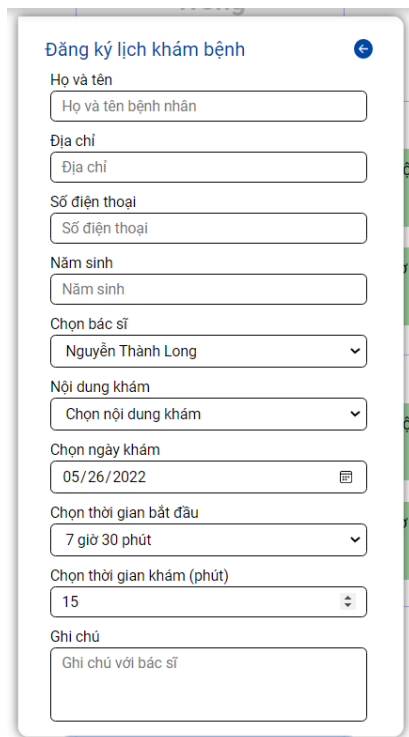
Hình 34 Giao diện xem lịch hẹn theo tuần

5.2. Giao diện đặt lịch

Giao diện đặt lịch của bệnh nhân chưa có tài khoản và bệnh nhân có tài khoản là như nhau, những bệnh nhân có tài khoản sẽ được điền sẵn các thông tin cơ bản, người dùng chưa có tài khoản phải nhập đầy đủ thông tin.

Nếu lịch hẹn bệnh nhân muốn đặt trùng với lịch hẹn sẵn có thì bệnh nhân không thể đặt được, bệnh nhân cần xem lại khoảng thời gian khi xem lịch và điều chỉnh lại khoảng thời gian của lịch hẹn.

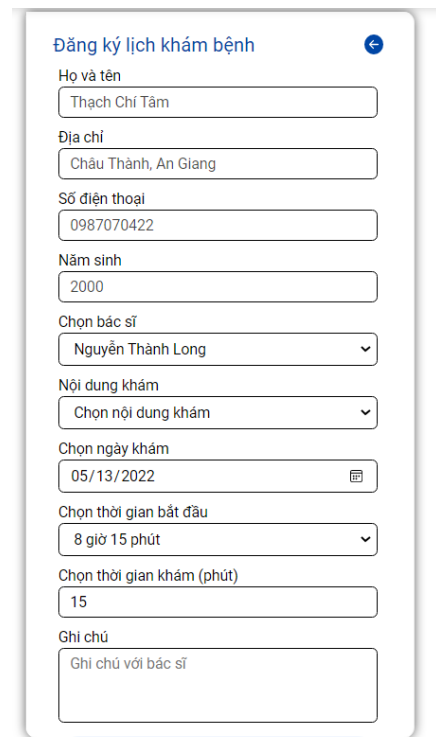
Bệnh nhân và nhân viên phòng khám có thể sử dụng chức năng này. Nhân viên phòng khám có thể đặt lịch giúp bệnh nhân hoặc đặt thời gian khám khi bệnh nhân đến khám bệnh trực tiếp mà chưa đặt lịch hẹn.



The screenshot shows a mobile application interface for booking an appointment. The title is "Đăng ký lịch khám bệnh". The form contains the following fields: "Họ và tên" (Last name and first name) with a placeholder "Họ và tên bệnh nhân"; "Địa chỉ" (Address) with a placeholder "Địa chỉ"; "Số điện thoại" (Phone number) with a placeholder "Số điện thoại"; "Năm sinh" (Year of birth) with a placeholder "Năm sinh"; "Chọn bác sĩ" (Select doctor) with a dropdown menu showing "Nguyễn Thành Long"; "Nội dung khám" (Consultation content) with a dropdown menu showing "Chọn nội dung khám"; "Chọn ngày khám" (Select consultation date) with a date picker showing "05/26/2022"; "Chọn thời gian bắt đầu" (Select start time) with a dropdown menu showing "7 giờ 30 phút"; "Chọn thời gian khám (phút)" (Select consultation time in minutes) with a dropdown menu showing "15"; and "Ghi chú" (Remarks) with a text area containing "Ghi chú với bác sĩ".

Hình 35 Giao diện đặt lịch của khách

(Không được điền sẵn thông tin)



The screenshot shows the same mobile application interface for booking an appointment, but with pre-filled information. The title is "Đăng ký lịch khám bệnh". The form contains the following fields: "Họ và tên" (Last name and first name) with the value "Thạch Chí Tâm"; "Địa chỉ" (Address) with the value "Châu Thành, An Giang"; "Số điện thoại" (Phone number) with the value "0987070422"; "Năm sinh" (Year of birth) with the value "2000"; "Chọn bác sĩ" (Select doctor) with a dropdown menu showing "Nguyễn Thành Long"; "Nội dung khám" (Consultation content) with a dropdown menu showing "Chọn nội dung khám"; "Chọn ngày khám" (Select consultation date) with a date picker showing "05/13/2022"; "Chọn thời gian bắt đầu" (Select start time) with a dropdown menu showing "8 giờ 15 phút"; "Chọn thời gian khám (phút)" (Select consultation time in minutes) with a dropdown menu showing "15"; and "Ghi chú" (Remarks) with a text area containing "Ghi chú với bác sĩ".

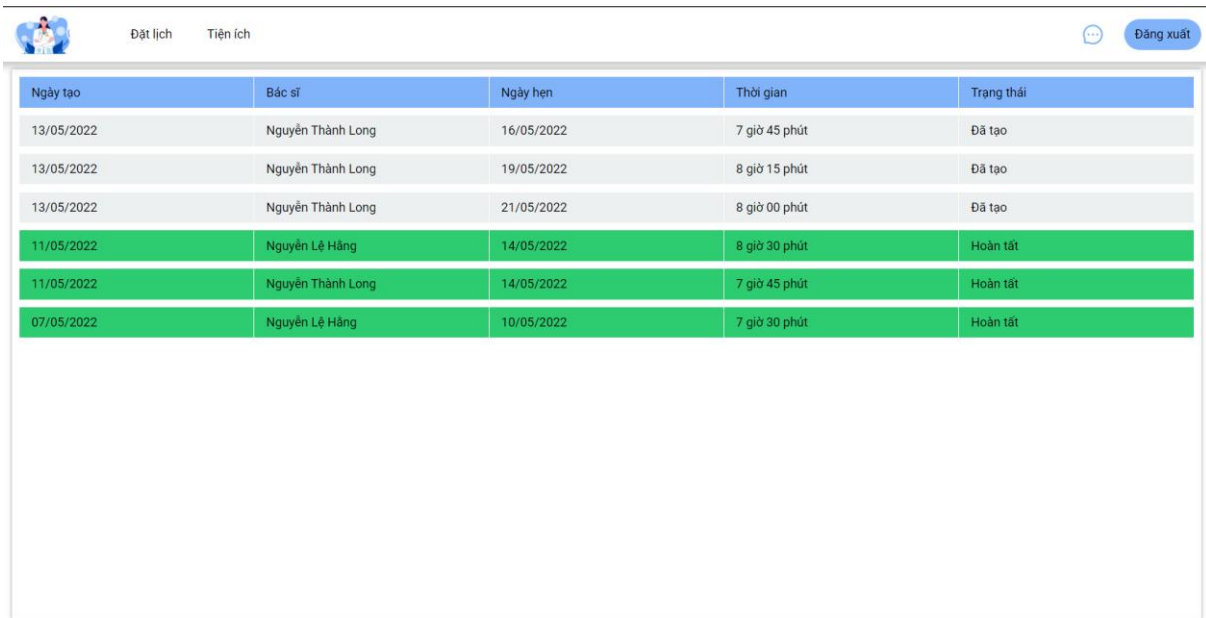
Hình 36 Giao diện đặt lịch của bệnh

nhân có tài khoản

(Được điền sẵn thông tin)


5.3. Giao diện lịch hẹn cá nhân của bệnh nhân


Đây là chức năng chỉ dành cho bệnh nhân có tài khoản, tại đây, bệnh nhân có thể xem trạng thái của các lịch hẹn, tùy vào trạng thái mà lịch hẹn có một màu sắc riêng, để có thể xem thêm thông tin, bệnh nhân nhấp vào từng lịch hẹn.




Ngày tạo	Bác sĩ	Ngày hẹn	Thời gian	Trạng thái
13/05/2022	Nguyễn Thành Long	16/05/2022	7 giờ 45 phút	Đã tạo
13/05/2022	Nguyễn Thành Long	19/05/2022	8 giờ 15 phút	Đã tạo
13/05/2022	Nguyễn Thành Long	21/05/2022	8 giờ 00 phút	Đã tạo
11/05/2022	Nguyễn Lê Hằng	14/05/2022	8 giờ 30 phút	Hoàn tất
11/05/2022	Nguyễn Thành Long	14/05/2022	7 giờ 45 phút	Hoàn tất
07/05/2022	Nguyễn Lê Hằng	10/05/2022	7 giờ 30 phút	Hoàn tất

Hình 37 Giao diện lịch hẹn cá nhân của bệnh nhân


Đặt lịch
Tien ich


Đăng xuất

Ngày tạo	Bác sĩ	Ngày hẹn	Thời gian	Trạng thái
13/05/2022	Nguyễn Thành Long	16/05/2022	7 giờ 45 phút	Đã tạo
13/05/2022	Nguyễn Thành Long	19/05/2022	8 giờ 15 phút	Đã tạo
13/05/2022	Nguyễn Thành Long	21/05/2022	8 giờ 00 phút	Đã hủy
11/05/2022	Nguyễn Lệ Hằng	14/05/2022	8 giờ 30 phút	Hoàn tất
11/05/2022	Nguyễn Thành Long	14/05/2022	7 giờ 45 phút	Hoàn tất
07/05/2022	Nguyễn Lệ Hằng	10/05/2022	7 giờ 30 phút	Hoàn tất

CHI TIẾT LỊCH KHÁM


Họ và tên: Thạch Chí Tâm
Năm sinh: 2000
Địa chỉ: Châu Thành, An Giang
Bác sĩ: Nguyễn Thành Long
Chuyên môn: Nha khoa, Tai mũi họng, Chụp X quang, ...
Ngày khám: 16/05/2022
Giờ khám: 7 giờ 45 phút
Trạng thái: Đã tạo
Thời gian tạo: 13/05/2022
Ghi chú: Không


Hủy lịch hẹn

Hình 38 Xem chi tiết khi chọn 1 lịch hẹn

5.4. Giao diện nhật ký thanh toán của bệnh nhân

Giao diện nhật ký thanh toán gồm danh sách các lần bệnh nhân đến khám chữa bệnh và thanh toán, bệnh nhân có thể nhấp vào từng dòng để xem chi tiết, nếu hóa đơn đó đã thanh toán có thể xuất ra tệp để có thể in.


Đặt lịch
Tien ich


Đăng xuất

Ngày khám	Tiền thuốc	Phụ phí	Tổng thanh toán	Người thanh toán	Trạng thái
14/05/2022	638.340 đ	15.000 đ	1.153.340 đ	Thạch Chí Tâm	Hoàn tất
14/05/2022	250.500 đ	20.000 đ	510.500 đ	Thạch Chí Tâm	Hoàn tất
10/05/2022	100.000 đ	25.000 đ	425.000 đ	Thạch Chí Tâm	Hoàn tất

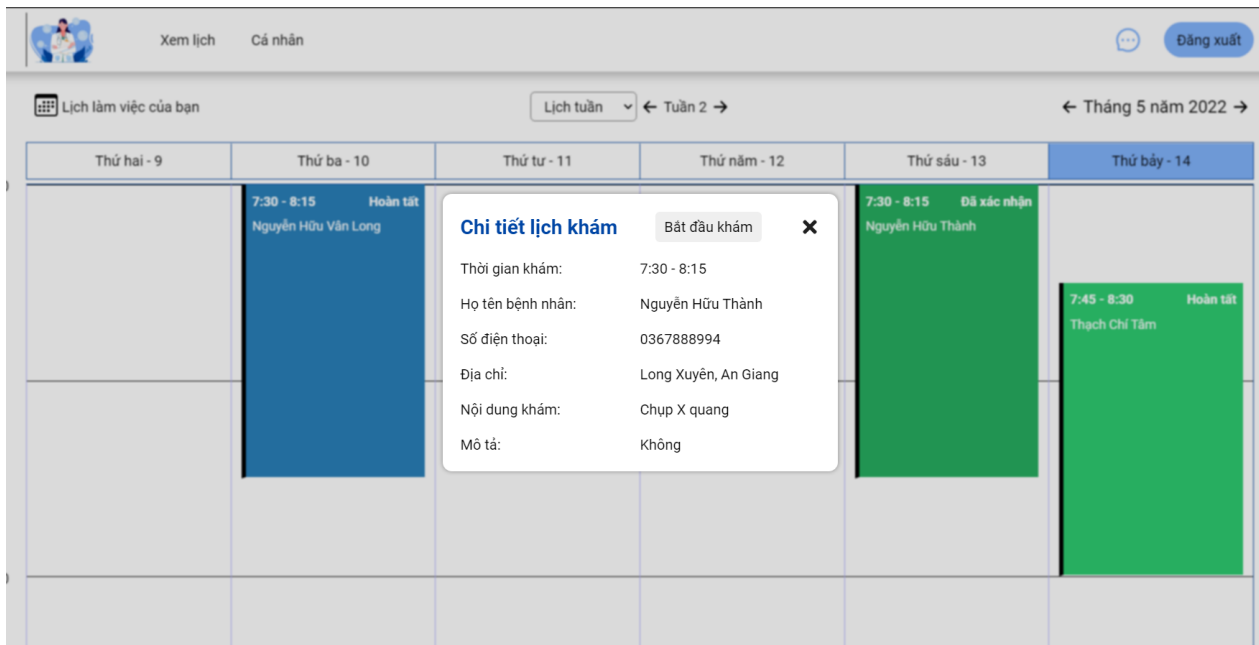
Hình 39 Giao diện nhật ký/ lịch sử thanh toán



Hình 40 Giao diện xem chi tiết thanh toán

5.5. Giao diện xem chi tiết lịch khám của bác sĩ

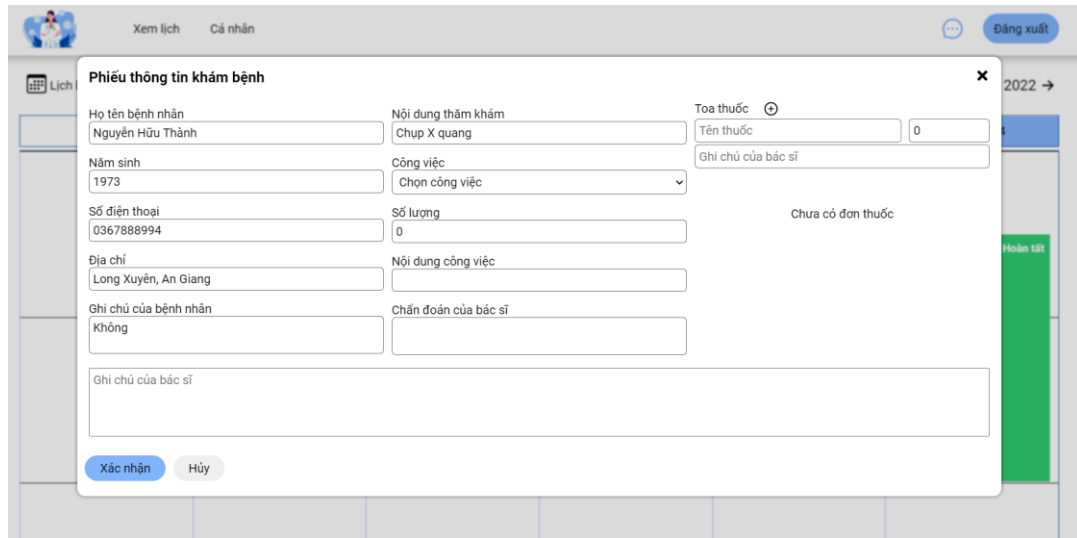
Tại trang lịch làm việc, bác sĩ có thể xem được chi tiết thông tin của lịch hẹn, nếu lịch hẹn đó được xác nhận, bác sĩ có thể tiến hành khám bệnh cho bệnh nhân



Hình 41 Giao diện xem chi tiết lịch khám của bác sĩ

5.6. Giao diện khám bệnh của bác sĩ

Bác sĩ có thể chọn công việc sẽ thực hiện cho bệnh nhân, số lượng công việc, nội dung chi tiết của công việc và thông tin chẩn đoán cho bệnh nhân. Nếu trong trường hợp có đơn thuốc cho bệnh nhân, bác sĩ có thể kê toa và thêm ghi chú của mình.



The screenshot shows a web application interface for a doctor's examination. At the top, there are tabs for 'Xem lịch' (View schedule) and 'Cá nhân' (Personal), along with a 'Đăng xuất' (Logout) button. The main form is titled 'Phiếu thông tin khám bệnh' (Examination information form). It contains several input fields and dropdown menus for patient and examination details. The patient information includes name, birth year, phone number, and address. The examination details include the type of examination, work type, quantity, and content. There is also a section for prescriptions and a note field for the doctor. The form is styled with a light blue header and a white body, with a green sidebar on the right.

Phiếu thông tin khám bệnh		
Họ tên bệnh nhân	Nội dung thăm khám	Toa thuốc
Nguyễn Hữu Thành	Chụp X quang	Tên thuốc
Năm sinh	Công việc	Ghi chú của bác sĩ
1973	Chọn công việc	
Số điện thoại	Số lượng	Chưa có đơn thuốc
0367888994	0	
Địa chỉ	Nội dung công việc	
Long Xuyên, An Giang		
Ghi chú của bệnh nhân	Chẩn đoán của bác sĩ	
Không		
Ghi chú của bác sĩ		
<button>Xác nhận</button> <button>Hủy</button>		

Hình 42 Giao diện nhập thông tin khám bệnh của bác sĩ

5.7. Giao diện quản lý thông tin lịch hẹn (Trang chủ của nhân viên phòng khám)

Tại giao diện này, nhân viên phòng khám có thể xem danh sách lịch hẹn của bệnh nhân.

Thời gian tạo	Thời gian bắt đầu	Thời gian khám	Tên bệnh nhân	Tên bác sĩ khám	Trạng thái
10:08 13/05/2022	07:30 17/05/2022	15 phút	Đỗ Duy Thanh	Nguyễn Thành Long	Đã tạo
10:30 13/05/2022	08:30 16/05/2022	45 phút	Vũ Ngọc Thắng	Nguyễn Thành Long	Đã tạo
10:25 13/05/2022	08:00 16/05/2022	30 phút	Lê Duy Hải	Nguyễn Thành Long	Đã tạo
10:21 13/05/2022	07:45 16/05/2022	15 phút	Thạch Chí Tâm	Nguyễn Thành Long	Đã tạo
10:07 13/05/2022	07:30 16/05/2022	15 phút	Nguyễn Thanh Kha	Nguyễn Thành Long	Đã tạo
11:39 11/05/2022	08:30 14/05/2022	30 phút	Thạch Chí Tâm	Nguyễn Lê Hằng	Hoàn tất
01:47 11/05/2022	09:15 14/05/2022	60 phút	Nguyễn Hồng Ân	Nguyễn Lê Hằng	Đã tạo
01:36 11/05/2022	07:45 14/05/2022	45 phút	Thạch Chí Tâm	Nguyễn Thành Long	Hoàn tất
12:52 11/05/2022	07:30 14/05/2022	30 phút	Trần Đăng Khoa	Đỗ Kim Long	Hoàn tất
03:23 10/05/2022	07:30 13/05/2022	45 phút	Nguyễn Hữu Thành	Nguyễn Thành Long	Đã xác nhận
03:24 07/05/2022	07:30 10/05/2022	45 phút	Thạch Chí Tâm	Nguyễn Lê Hằng	Hoàn tất
01:31 07/05/2022	07:30 10/05/2022	45 phút	Nguyễn Hữu Văn Long	Nguyễn Thành Long	Hoàn tất

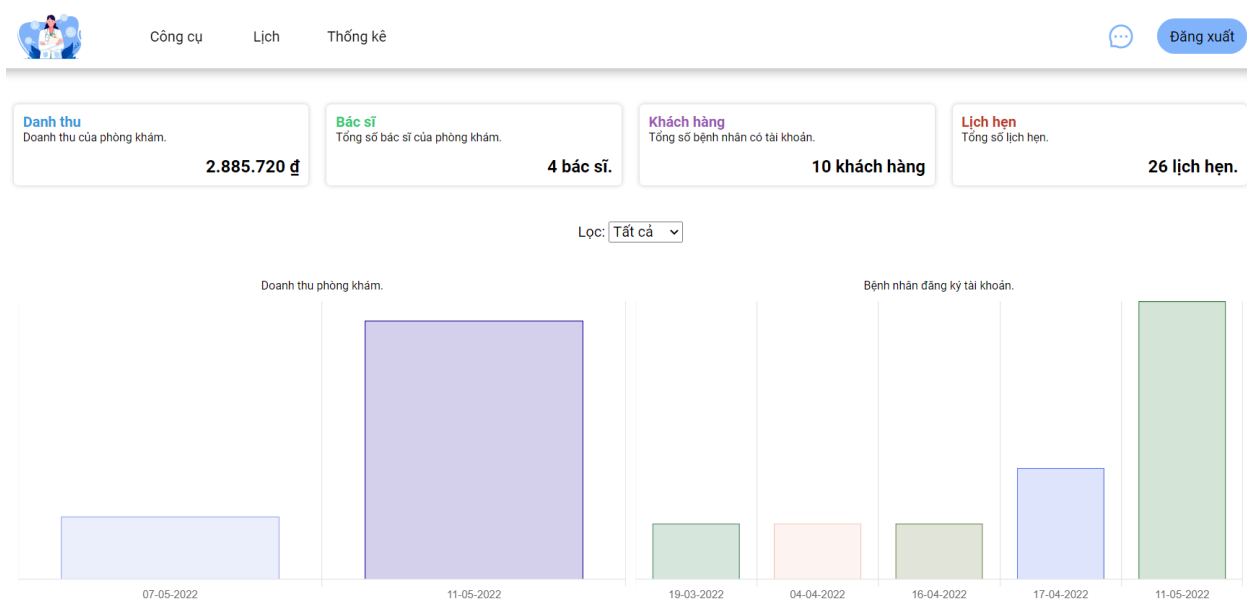
Hình 43 Trang chủ của nhân viên phòng khám

Thông tin lịch khám của Thạch Chí Tâm			
Trạng thái		Thời gian khám	
Hoàn tất		45 phút	
Thông tin			
Nội dung khám bệnh: Chụp X quang			
Họ và tên:	Thạch Chí Tâm	Bác sĩ:	Nguyễn Thành Long
Năm sinh:	2000	Ngày khám:	14/05/2022
Liên hệ:	0987070422	Thời gian:	7 giờ 45 phút
Địa chỉ:	Châu Thành, An Giang	Chi phí:	510.500 đ
Thông tin thanh toán			
Tên sản phẩm/dịch vụ	Đơn giá	Số lượng	Thành tiền
Chụp X quang ngực thẳng	240.000 đ	1	240.000 đ
Air-X	5.000 đ	30	150.000 đ
Eurocaljium-M	3.350 đ	30	100.500 đ
Dịch vụ chữa bệnh theo thời gian	5.000 đ	3	15.000 đ
Xuất hóa đơn			

Hình 44 Giao diện xem chi tiết lịch khám

5.8. Thống kê thông tin phòng khám

Chức năng thống kê các thông tin của phòng khám thông qua các thẻ và biểu đồ



Hình 45 Giao diện thống kê của phòng khám

CHƯƠNG 3 KIỂM THỬ VÀ ĐÁNH GIÁ

1. Mục tiêu kiểm thử:

- Đảm bảo rằng các điểm cuối cung cấp tài nguyên cho hệ thống hoạt động bình thường.
- Đảm bảo các thành phần, luồng thực hiện công việc của giao diện người dùng đúng như mong đợi.

2. Trường hợp kiểm thử

2.1. Kiểm thử thủ công cho giao diện người dùng

Mã kiểm thử	Mục đích	Bước thực hiện	Kết quả mong đợi
QA_01	Kiểm tra chức năng đăng nhập với thông tin đúng	1. Nhấp Đăng nhập 2. Nhập thông tin đúng 3. Nhấp Đăng nhập	Đăng nhập thành công, chuyển đến trang chủ của từng nhóm người dùng
QA_02	Kiểm tra chức năng đăng nhập với thông tin sai	1. Nhấp đăng nhập 2. Nhập thông tin sai hoặc để trống 3. Nhấp đăng nhập	Đăng nhập không thành công, hiện thông báo thông tin đăng nhập không chính xác
QA_03	Kiểm tra chức năng xem lịch hẹn	1. Nhấp vào Đặt lịch 2. Chọn ra một bác sĩ 3. Chuyển sang chế độ xem lịch tuần	Hiện thị chế độ tuần của bác sĩ đã chọn
QA_04	Kiểm tra chức năng đặt lịch	1. Nhấp vào Đặt lịch. 2. Nhấp đặt lịch 3. Chuyển sang chế độ xem lịch tuần. 4. Nhấp vào một ngày 5. Điền đầy đủ thông tin 5. Nhấp đặt lịch hẹn.	Nếu thiếu thông tin, báo lỗi thiếu thông tin. Nếu lịch được đặt vào chủ nhật, hiện thị gợi ý không đặt lịch vào chủ nhật Nếu lịch bị trùng, hiện thị thông báo lịch bị trùng

			Nếu các thông tin đã hợp lệ, hiển thị thông báo đặt lịch thành công.
QA_05	Kiểm tra chức năng khám bệnh của bác sĩ	<ol style="list-style-type: none"> 1. Bác sĩ đã đăng nhập thành công 2. Bác sĩ truy cập vào lịch làm việc. 3. Bác sĩ chọn một lịch hẹn có trạng thái là Đã xác nhận 	Hiển thị cửa sổ nhập thông tin khám bệnh
QA_06	Kiểm tra chức năng xem lịch sử chi trả của bệnh nhân	<ol style="list-style-type: none"> 1. Bệnh nhân đã đăng nhập vào hệ thống 2. Bệnh nhân chọn Tiện ích 3. Bệnh nhân chọn Hóa đơn 	Hiển thị danh sách hóa đơn của bệnh nhân
QA_07	Kiểm tra chức năng xem chi tiết hóa đơn	<ol style="list-style-type: none"> 1. Bệnh nhân đã đăng nhập vào hệ thống 2. Bệnh nhân chọn Tiện ích 3. Bệnh nhân chọn Hóa đơn. 4. Chọn ngẫu nhiên một hóa đơn 	Hiển thị chi tiết hóa đơn
QA_08	Kiểm tra chức năng in hóa đơn đã thanh toán	<ol style="list-style-type: none"> 1. Bệnh nhân đã đăng nhập vào hệ thống 2. Bệnh nhân chọn Tiện ích 3. Bệnh nhân chọn Hóa đơn. 	Trình duyệt tự động tải xuống một tệp tin chứa thông tin thanh toán của hóa đơn

		4. Chọn ngẫu nhiên một hóa đơn đã thanh toán thành công. 5. Chọn in hóa đơn	
QA_9	Kiểm tra chức năng xem danh sách đặt lịch	1. Nhân viên phòng khám đăng nhập vào hệ thống	Hiển thị danh sách các lịch hẹn
QA_10	Kiểm tra chức năng xuất hóa đơn cho hóa đơn đã thanh toán	1. Nhân viên đăng nhập vào hệ thống. 2. Chọn một lịch đã được hoàn tất 3. Chọn in hóa đơn	Trình duyệt tự động tải xuống

Bảng 36 Trường hợp kiểm thử thủ công công (các chức năng chính)

2.2 Kiểm thử tự động dành cho server web api

Kiểm thử kết nối

Mã kiểm thử	Mục đích	Điểm cuối	Dữ liệu đầu vào	Kết quả mong đợi
QAA_01	Kiểm tra kết nối đến server.	GET: /		- Status: 200 - Body.message: "Welcome to Hospital booking resource"
QAA_02	Kiểm tra phương thức GET	GET: /api/test		Status: 200
QAA_03	Kiểm tra phương thức POST	POST: /api/test		Status: 200
QAA_04	Kiểm tra một url không hợp lệ	GET: /api/invalidurl		Status: 400

Kiểm thử xác thực tài nguyên

Mã kiểm thử	Mục đích	Điểm cuối	Dữ liệu đầu vào	Kết quả mong đợi
QAA_05	Kiểm tra lấy dữ liệu khi có key bí mật	GET: /api/testsecret	Header: { Secret-key: testsecret }	- Status: 200
QAA_06	Kiểm tra lấy dữ liệu khi không có key bí mật	GET: /api/testsecret		Status: 400. Body.message: "Secret key is invalid!"

Kiểm thử xác thực thông tin người dùng

Mã kiểm thử	Mục đích	Điểm cuối	Dữ liệu đầu vào	Kết quả mong đợi
QAA_07	Kiểm tra đăng nhập với thông tin chính xác	POST: /api/auth/login	Header: { Secret-key: testsecret } Body: { username: "tamthachchi", password: "hash<123456>" }	- Status: 200; - Body.message: "Đăng nhập thành công" - Body.status: 200 - Body.body: {token}
QAA_08	Kiểm tra đăng nhập với thông tin không chính xác	POST: /api/auth/login	Header: { Secret-key: testsecret } Body: { username: "tamthachchi", password: "hash<abcded>" }	- Status: 404 - Body.message: "Account's not found!"
QAA_09	Kiểm tra đăng nhập	POST: /api/auth/login	Header: { Secret-key: testsecret } Body: {	- Status: 400 - Body.message: "Tài khoản là bắt buộc"

	thiếu tài khoản		password: 123456 }	
QAA_10	Kiểm tra đăng nhập thiếu mật khẩu	POST: /api/auth/login	Header: { Secret-key: testsecret } Body: { username: "tamthachchi" }	- Status: 400 - Body.message: "Mật khẩu là bắt buộc"
QAA_11	Kiểm tra lấy thông tin sau khi đăng nhập thành công	GET: /api/auth/:token	Header: { Secret-key: testsecret } token: dữ liệu trả về sau khi đăng nhập thành công (QAA_05)	- Status: 200 - Body.body.username: "tamthachchi" - Body.body.type: "customer" - Body.message: "Thông tin hợp lệ"
QAA_12	Đăng ký tài khoản bệnh nhân với thông tin chưa tồn tại	POST: /api/auth/register	Header: { Secret-key: testsecret } Body: { name: "Hữu Lộc", phone: "09878765821", ccid: "126765864475", email: hloc@gmail.com, username: hloc123, password: hash<12345> }	- Status: 200 - Body.message: "Tạo tài khoản khách hàng thành công!"
QAA_13	Đăng ký tài khoản với thông tin tồn tại	POST: /api/auth/register	Header: { Secret-key: testsecret } Body: { name: "Trần Đăng Khoa", phone: "0486787654", ccid: "348156781509", email: "dangkhoa1@gmail.com", username: "dangkhoa1",	Status: 400 Body.message: "Thông tin đã được sử dụng"

			password: hash<123456> }	
QAA_14	Đăng ký tài khoản thiếu thông tin	POST: /api/auth/register	Header:{ Secret-key: testsecret } Body:{ name:"Huy Vũ", phone: "09835765821", ccid: "126724864475", email: hvu@gmail.com, username: hvu123 }	Status: 400 Body.message: "Mật khẩu là bắt buộc"

Kiểm thử các điểm cuối của đối tượng bệnh nhân

Mã kiểm thử	Mục đích	Điểm cuối	Dữ liệu đầu vào	Kết quả mong đợi
QAA_15	Lấy thông tin bệnh nhân bằng id chính xác	GET: /api/customer/:id	Header:{ Secret-key: testsecret } id: 932fdd5f-152a-4150-bc1d-76418ed845f0	- Status: 200; - Body.message: "Lấy dữ liệu thành công" - Body.status: 200 - type of (body.body): object
QAA_16	Lấy thông tin bệnh nhân với id không chính xác	POST: /api/customer/:id	Header:{ Secret-key: testsecret } id: 932fdd5f-152a-4150-bc1d-76418ed857ff	- Status: 404 - Body.message: "Không tìm thấy"
QAA_17	Lấy danh sách	POST: /api/customer/list	Header:{ Secret-key: testsecret }	- Status: 200 - Body.message: "OK"

	bệnh nhân			- typeof (Body.body) : array
QAA_18	Lấy danh sách hóa đơn bằng id của bệnh nhân	POST: /api/customer/bill/:id	Header: { Secret-key: testsecret } id: 932fdd5f-152a-4150-bc1d-76418ed845f0	- Status: 200 - typeof (Body.body) : array

Kiểm thử các điểm cuối của đối tượng bác sĩ

Mã kiểm thử	Mục đích	Điểm cuối	Dữ liệu đầu vào	Kết quả mong đợi
QAA_19	Tìm kiếm bác sĩ với dữ liệu rỗng	GET: /api/doctor/search	Header: { Secret-key: testsecret }	- Status: 200; - type of (body.body): array
QAA_20	Tìm kiếm bác sĩ với từ khóa không tìm thấy	GET: /api/doctor/search/:text	Header: { Secret-key: testsecret } text: aaaaaa	Status: 200; - type of (body.body): array
QAA_21	Lấy thông tin bác sĩ bằng id chính xác	GET: /api/doctor/:id	Header: { Secret-key: testsecret } id: d4ed722a-d522-49ee-a5f1-cf87658906ec-15185-7123382	- Status: 200 - Body.message: "OK" - typeof (Body.body) : object
QAA_22	Lấy thông tin bác sĩ bằng id không	GET: /api/doctor/:id	Header: { Secret-key: testsecret }	- Status: 404 - Body.message: "Không tìm thấy thông tin"

	chính xác		id: d4ed7a-d522-49ee-a5f1-cf87658906ec-15185-7123382	
--	-----------	--	--	--

Kiểm thử điểm cuối của đối tượng nhân viên phòng khám

Mã kiểm thử	Mục đích	Điểm cuối	Dữ liệu đầu vào	Kết quả mong đợi
QAA_23	Lấy thông tin thống kê	GET: /api/cashier/analysis	Header: { Secret-key: testsecret }	- Status: 200; - type of (body.body): array
QAA_24	Lấy thông tin nhân viên với id chính xác	GET: /api/cashier/:id	Header: { Secret-key: testsecret } id: 59fbaef2-e322-459e-aec4-1f849f57b45e-613949-9723853	- Status: 200; - Body.message: "Ok" - type of (body.body): object
QAA_25	Lấy thông tin nhân viên với id không chính xác	GET: /api/cashier/:id	Header: { Secret-key: testsecret } id: 59f322-459e-aec4-1f849f57b45e-613949-9723853	- Status: 404 - Body.message: "Not found"

Kiểm tra điểm cuối của lịch hẹn

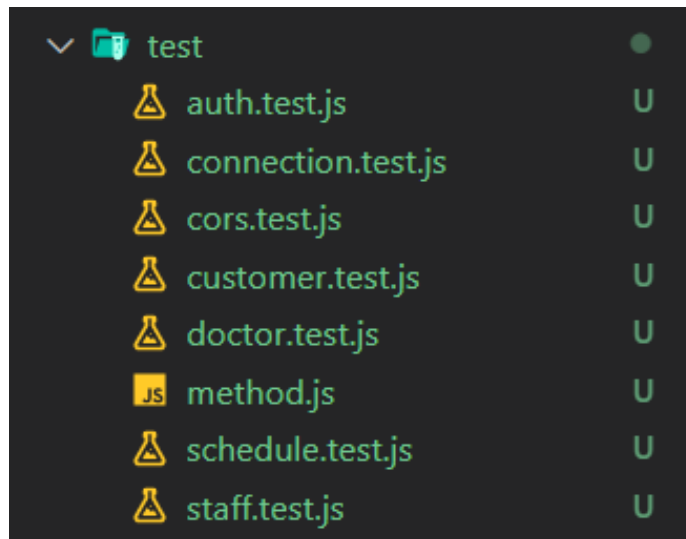
Mã kiểm thử	Mục đích	Điểm cuối	Dữ liệu đầu vào	Kết quả mong đợi
QAA_26	Lấy lịch hẹn bằng id bệnh nhân	GET: /api/schedule/customer/:id	Header: { Secret-key: testsecret } id: 932fdd5f-152a-4150-bc1d-76418ed845f0	- Status: 200; - type of (body.body): array
QAA_27	Lấy hóa đơn của lịch hẹn	GET: /api/schedule/bill/:id	Header: { Secret-key: testsecret }	- Status: 200; - Body.message: "Ok"

	bảng id lịch hẹn chính xác		id: 6df0b943-750d-43ce- b549-fc01541cade6- 97479-6666362	- type of (body.body): object
QAA_28	Lấy hóa đơn của lịch hẹn bảng id lịch hẹn không chính xác	GET: /api/schedule /bill/:id	Header: { Secret-key: testsecret } id: 6df0b943-750d-43ce- e7494	- Status: 404 - Body.message: “Not found”
QAA_29	Lấy tất danh sách lịch hẹn	GET: /api/schedule/list	Header: { Secret-key: testsecret }	- Status: 200 - typeof (Body.body): array
QAA_30	Lấy lịch hẹn với id chính xác	GET: /api/schedule/:id	Header: { Secret-key: testsecret } id: 6df0b943-750d-43ce- b549-fc01541cade6- 97479-6666362	- Status: 200 - typeof (Body.body): object - Body.message: “Lấy dữ liệu thành công”
QAA_31	Lấy lịch hẹn với id không chính xác	GET: /api/schedule/:id	Header: { Secret-key: testsecret } id: 6df0b943-750d-43ce- b549-fc01541cade6- 97479-6666362	- Status: 404 - Body.message: “Không tìm thấy dữ liệu”

Tạo thư mục test bên trong dự án, viết các trường hợp kiểm thử với phần mở rộng là .test.js.

Trong tệp tin package.json của dự án, tại phần scripts, thêm hoặc thay thế key test thành *test: “jest”*

Để tiến hành kiểm thử, thực hiện *npm test*



Hình 46 Thư mục và tệp tin kiểm thử

```
PS H:\workspace\HospitalBooking\hospitalbooking-nodejs> npm test
> hospitalbooking-nodejs@1.0.0 test
> jest
PASS test/cors.test.js
PASS test/connection.test.js
PASS test/doctor.test.js
PASS test/customer.test.js
PASS test/staff.test.js
PASS test/schedule.test.js
PASS test/auth.test.js

Test Suites: 7 passed, 7 total
Tests: 31 passed, 31 total
Snapshots: 0 total
Time: 4.942 s
Ran all test suites.
PS H:\workspace\HospitalBooking\hospitalbooking-nodejs>
```

Hình 47 Kết quả kiểm thử API

3. Đánh giá

Hệ thống đáp ứng được nhu cầu bình thường của người sử dụng, không xảy ra lỗi, ngoại lệ khi kiểm thử hệ thống. Các luồng xử lý trên giao diện đúng như theo mô tả của trường hợp kiểm thử.

Kiểm thử tự động đảm bảo cái API hoạt động bình thường, thời gian thực thi ít độ trễ.

PHẦN KẾT LUẬN

1. Kết quả đạt được

- Hoàn thành được đề tài và xây dựng được hệ thống đặt lịch trực tuyến theo các chức năng được yêu cầu.
- Cung cố kiến thức lý thuyết, cách thức xây dựng mô hình cho hệ thống.
- Cung cố kiến thức lập trình cũng như tìm hiểu thêm các công nghệ mới.
- Cải thiện khả năng làm việc độc lập, tự tìm hiểu và nghiên cứu để giải quyết vấn đề.
- Hệ thống được triển khai thành công tại <https://hospitalclient.vercel.app> thực thi phân giao diện người dùng qua máy chủ của Vercel, Server Web API được triển khai trên máy chủ của Heroku.

2. Hạn chế

- Mặc dù hệ thống đáp ứng được nhu cầu của người sử dụng nhưng vẫn còn một số hạn chế như: hệ thống thực thi chậm trên môi trường triển khai thực tế so với môi trường phát triển (do thiếu chi phí).

3. Hướng phát triển

- Về hệ thống đặt lịch trực tuyến, tiếp tục hoàn thiện và bổ sung thêm nhiều chức năng phục vụ cho người dùng, từ đó có thể đáp ứng được nhu cầu của người dùng thực tế, giúp tiết kiệm thời gian và chi phí khám chữa bệnh. Nâng cấp bằng thông cơ sở để hệ thống có thể hoạt động nhanh chóng và hiệu quả hơn.
- Về công nghệ, React.js, MongoDB, Node.js là những công nghệ xu thế trong những năm tới đây vì mức độ tiện dụng, dễ học đối với lập trình viên mới bắt đầu. Bên cạnh đó những hệ thống này có đặt tính hiện suất cao, dễ mở rộng, dễ bảo trì.

TÀI LIỆU THAM KHẢO

- [1] Phạm Gia Tiến, Phạm Thế Phi. Giáo trình hệ quản trị cơ sở dữ liệu. Nhà xuất bản Đại học Cần Thơ. Cần Thơ.
- [2] Phan Tấn Tài, [et.al], 2010. Giáo trình phân tích thiết kế hệ thống thông tin. Nhà xuất bản Đại học Cần Thơ, Cần Thơ.
- [3] Huỳnh Xuân Hiệp, Phan Phương Lan, giáo trình Nhập môn công nghệ phần mềm, NXB Đại học Cần Thơ, 2011.
- [4] Trần Văn Hoàng, bài giảng Phân tích yêu cầu phần mềm, khoa CNTT&TT, Đại học Cần Thơ, 2015.
- [5] Huỳnh Xuân Hiệp, Giáo trình Thiết kế phần mềm, khoa CNTT& TT, Đại học Cần Thơ, 2015.
- [6] Trần Cao Đệ, Đỗ Thanh Nghị, Giáo trình Kiểm thử phần mềm, NXB Đại học Cần Thơ, 2012.
- [7] Wikipedia, Node.js, <https://en.wikipedia.org/wiki/Node.js>, 28-04-2022.
- [8] Wikipedia, MongoDB, <https://en.wikipedia.org/wiki/MongoDB>, 28-04-2022.
- [9] Wikipedia, React (JavaScript library), [https://en.wikipedia.org/wiki/React_\(JavaScript_library\)](https://en.wikipedia.org/wiki/React_(JavaScript_library)), 28-04-2022.
- [10] Lê Tấn Lộc, Ứng dụng di động hỗ trợ công tác bảo tồn đa dạng sinh học ở Tràm Chim, https://drive.google.com/file/d/1NxLCkI2EwuH02SoczNA6VnLRNeRQvk_6/view, 04-2020.
- [11] Viblo, React là gì? Những điều cần biết về ReactJSC, <https://viblo.asia/p/React.js-la-gi-nhung-dieu-can-biet-ve-React.jsc-djeZ1GJo5Wz>, 28-04-2022
- [12] Bệnh viện Đại học Y dược TP.HCM (umcclinic.com.vn), Đăng ký khám, <https://umcclinic.com.vn/dang-ky-kham>, 30-3-2022.
- [13] W3School, Node.js Send an Email, https://www.w3schools.com/Node.js/Node.js_email.asp, 10-4-2022.

[14] Node.js, The V8 JavaScript Engine, <https://Node.js.dev/learn/the-v8-javascript-engine>, 28-04-2022.

HẾT