

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

HỆ THỐNG QUẢN LÝ NHÀ KHOA TƯ NHÂN
SỬ DỤNG KIẾN TRÚC VUEJS & SPRING BOOT

Người hướng dẫn: **ThS. Nguyễn Thị Xuân Thủy**

Sinh viên thực hiện: **Nguyễn Nhật Thành**

Mã số sinh viên: **2151220143**

Lớp: **21CT3**

Đà Nẵng, 06/2026

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

HỆ THỐNG QUẢN LÝ NHÀ KHOA TƯ NHÂN
SỬ DỤNG KIẾN TRÚC VUE.JS & SPRING BOOT

Người hướng dẫn: **ThS. Nguyễn Thị Xuân Thủy**

Sinh viên thực hiện: **Nguyễn Nhật Thành**

Mã số sinh viên: **2151220143**

Lớp: **21CT3**

Đà Nẵng, 06/2026

TÓM TẮT

Tên đề tài: Hệ thống quản lý nha khoa tư nhân sử dụng kiến trúc VueJS và Spring Boot

Sinh viên thực hiện: NGUYỄN NHẬT THÀNH

Mã số SV: 2151220143

Lớp: 21CT3

Đề tài “**Hệ thống quản lý nha khoa tư nhân sử dụng kiến trúc VueJS và Spring Boot**” được thực hiện với mục tiêu xây dựng một nền tảng trực tuyến hỗ trợ phòng khám nha khoa trong việc quản lý lịch hẹn, hồ sơ bệnh nhân, kế hoạch điều trị, thanh toán và các nghiệp vụ vận hành hằng ngày. Hệ thống hướng đến việc số hóa quy trình làm việc tại phòng khám, giảm thao tác thủ công, hạn chế sai sót trong quản lý dữ liệu và nâng cao trải nghiệm của bệnh nhân khi sử dụng dịch vụ nha khoa.

Về công nghệ, Vue.js được sử dụng để xây dựng giao diện người dùng hiện đại, trực quan và phản hồi nhanh; Spring Boot đảm nhận phát triển backend theo kiến trúc RESTful API với khả năng bảo mật, phân quyền và mở rộng tốt; PostgreSQL đóng vai trò là hệ quản trị cơ sở dữ liệu quan hệ, đảm bảo lưu trữ ổn định, toàn vẹn và phù hợp với các nghiệp vụ y tế có nhiều mối liên kết dữ liệu.

Hệ thống cung cấp đầy đủ các chức năng phục vụ nhiều nhóm đối tượng chính:

Phía bệnh nhân:

- ~ Đăng ký, đăng nhập, quên mật khẩu, cập nhật thông tin cá nhân và đổi mật khẩu.
- ~ Xem trang chủ, thông tin dịch vụ nha khoa, thông tin bác sĩ, nội dung giới thiệu và các thông tin công khai của phòng khám.
- ~ Đặt lịch khám trực tuyến theo ngày, ca khám và theo dõi trạng thái lịch hẹn.
- ~ Nhận email/thông báo xác nhận lịch hẹn, nhắc lịch khám và các thông tin liên quan đến quá trình khám.
- ~ Xem lịch sử lịch hẹn, thông tin tái khám và các kết quả xử lý liên quan đến dịch vụ đã sử dụng.
- ~ Theo dõi hóa đơn, trạng thái thanh toán và thực hiện thanh toán qua tiền mặt hoặc VNPay khi có phát sinh chi phí.

Phía bác sĩ:

- ~ Xem danh sách lịch hẹn được phân công theo ngày và ca làm việc.
- ~ Tiếp nhận bệnh nhân, cập nhật thông tin khám, ghi nhận chẩn đoán và quá trình điều trị.

- ~ Tạo kế hoạch điều trị, lập lịch tái khám và theo dõi tiến độ điều trị của bệnh nhân.
- ~ Kê đơn thuốc, xem thông tin thuốc và hỗ trợ quản lý hồ sơ điều trị.

Phía nhân viên:

- ~ Quản lý hóa đơn phát sinh từ dịch vụ khám và điều trị.
- ~ Ghi nhận thanh toán, cập nhật trạng thái hóa đơn và hỗ trợ thanh toán VNPay.
- ~ Theo dõi các khoản thu, lịch sử giao dịch và thông tin thanh toán của bệnh nhân.

Phía quản trị:

- ~ Quản lý người dùng, tài khoản và phân quyền theo vai trò Admin, Doctor, Cashier và Patient.
- ~ Quản lý bệnh nhân, bác sĩ, lịch làm việc, năng lực tiếp nhận theo ca và phân công bác sĩ cho lịch hẹn.
- ~ Quản lý danh mục dịch vụ nha khoa, thuốc, đơn thuốc, tồn kho thuốc và các dữ liệu nghiệp vụ liên quan.
- ~ Theo dõi, xác nhận, hủy hoặc điều chỉnh lịch hẹn; xử lý lịch tái khám và các yêu cầu phát sinh.
- ~ Quản lý hóa đơn, thanh toán, doanh thu, chi phí và các báo cáo/thống kê hỗ trợ vận hành phòng khám.
- ~ Quản lý nội dung hiển thị công khai như thông tin bác sĩ, dịch vụ, bài giới thiệu và hình ảnh trên website.

Các tính năng nổi bật

- ~ Bảo mật với JWT Authentication và phân quyền theo vai trò người dùng.
- ~ Thông báo realtime bằng WebSocket/toast notifications kết hợp email tự động.
- ~ Tích hợp thanh toán VNPay, hỗ trợ theo dõi hóa đơn và trạng thái thanh toán.
- ~ Quản lý lịch hẹn, lịch tái khám, hồ sơ điều trị và vận hành phòng khám theo quy trình số hóa.

Dự án đã xây dựng thành công một website quản lý phòng khám nha khoa có giao diện thân thiện, đầy đủ các chức năng cốt lõi và phù hợp với nhu cầu thực tế. Kết quả đạt được không chỉ đáp ứng mục tiêu đề ra mà còn giúp sinh viên củng cố kiến thức và kỹ năng lập trình web full-stack theo kiến trúc RESTful API, đồng thời có thêm kinh nghiệm trong việc phân tích nghiệp vụ, thiết kế hệ thống, xây dựng backend, frontend và tích hợp các chức năng thanh toán, thông báo trong một sản phẩm hoàn chỉnh.

LỜI NÓI ĐẦU

Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ, các phòng khám nha khoa ngày càng có nhu cầu ứng dụng hệ thống quản lý trực tuyến nhằm nâng cao hiệu quả vận hành và cải thiện trải nghiệm khách hàng. Việc quản lý thủ công bằng giấy tờ hoặc file rời rạc dễ gây sai sót, mất thời gian và khó theo dõi thông tin điều trị.

Xuất phát từ thực tế đó, đề tài “Hệ thống quản lý nha khoa tư nhân sử dụng kiến trúc VueJS và Spring Boot” được thực hiện nhằm xây dựng một website hỗ trợ quản lý bệnh nhân, lịch hẹn, kế hoạch điều trị, hóa đơn và thanh toán trực tuyến.

Hệ thống sử dụng VueJS để xây dựng giao diện người dùng, Spring Boot để phát triển RESTful API và PostgreSQL để lưu trữ dữ liệu. Ngoài ra, hệ thống còn tích hợp JWT Authentication, gửi email thông báo và thanh toán VNPay nhằm nâng cao tính tiện lợi và bảo mật.

Trong quá trình thực hiện đề tài, em đã vận dụng các kiến thức về phân tích thiết kế hệ thống, lập trình web full-stack, thiết kế cơ sở dữ liệu và triển khai ứng dụng thực tế.

Em xin chân thành cảm ơn cô **ThS. Nguyễn Thị Xuân Thủy** cùng các thầy cô Khoa Công nghệ Thông tin – Trường Đại học Kiến trúc Đà Nẵng đã tận tình hướng dẫn và hỗ trợ em trong suốt quá trình thực hiện đồ án.

CAM ĐOAN

Tôi, Nguyễn Nhật Thành, sinh viên lớp 21CT3, mã số sinh viên 2151220143, sinh viên khóa 21, Khoa Công nghệ Thông tin, Trường Đại học Kiến trúc Đà Nẵng, xin cam đoan rằng đề tài:

“Hệ thống quản lý nha khoa tư nhân sử dụng kiến trúc VueJS và Spring Boot”

Đây là kết quả nghiên cứu và thực hiện của riêng tôi dưới sự hướng dẫn của cô **ThS. Nguyễn Thị Xuân Thủy**. Các số liệu, dữ liệu, hình ảnh và thông tin được sử dụng trong đồ án đều có nguồn gốc rõ ràng, chính xác và trung thực. Tôi hoàn toàn chịu trách nhiệm trước Nhà trường về tính trung thực của toàn bộ nội dung trong đồ án này.

Sinh viên thực hiện

MỤC LỤC

TÓM TẮT	I
LỜI NÓI ĐẦU	I
CAM ĐOAN	II
MỤC LỤC	III
DANH MỤC CÁC BẢNG	VI
DANH MỤC CÁC HÌNH	VII
DANH MỤC KÝ HIỆU, CHỮ VIẾT TẮT	X
MỞ ĐẦU	XII
CHƯƠNG 1. CƠ SỞ LÝ THUYẾT	1
1.1. Tổng quan về hệ thống quản lý nha khoa	1
1.1.1. Tổng quan về hệ thống quản lý phòng khám nha khoa	1
1.1.2. Quy trình đặt lịch khám nha khoa	1
1.1.3. Quản lý hồ sơ bệnh nhân	2
1.1.4. Quản lý lịch hẹn và phân công bác sĩ	2
1.1.5. Quản lý kế hoạch điều trị	2
1.1.6. Quản lý đơn thuốc và thuốc	3
1.1.7. Quản lý hóa đơn và thanh toán	3
1.1.8. Vai trò của AI trong hỗ trợ quản lý nha khoa	3
1.2. Cơ sở lý thuyết về công nghệ	4
1.2.1. Ngôn ngữ HTML/CSS	4
1.2.2. Ngôn ngữ JavaScript	4
1.2.3. Vue.js Framework	4
1.2.4. Ngôn ngữ lập trình Java	7
1.2.5. Spring Boot	7
1.2.6. Cơ sở dữ liệu PostgreSQL	7
1.2.7. Kiến trúc giao tiếp: RESTful API	8

1.2.8. JWT Authentication.....	9
1.3. Kiến trúc và mô hình triển khai hệ thống.....	10
1.3.1. Kiến trúc Client – Server.....	10
1.3.2. Mô hình MVC.....	10
1.3.3. Quy trình giao tiếp hệ thống.....	10
1.3.4. Mô hình triển khai hệ thống.....	10
1.4. Công cụ và cài đặt.....	11
1.4.1. Visual Studio Code.....	11
1.4.2. PgAdmin 4.....	11
1.4.3. GeminiAI.....	11
1.4.4. VNPay.....	12
1.4.5. Cài đặt Node.js và NPM.....	13
1.4.6. Cài đặt Java JDK.....	13
CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.....	14
2.1. Mô tả bài toán.....	14
2.2. Xác định Actor.....	15
2.2.1. Bệnh nhân.....	17
2.2.2. Bác sĩ.....	17
2.2.3. Nhân viên.....	17
2.2.4. Admin.....	18
2.3. Xác định Usecase.....	18
2.4. Phân tích Usecase.....	21
2.4.1. Biểu đồ Usecase.....	21
2.4.2. Đặc tả usecase.....	24
2.5. Biểu đồ hoạt động.....	40
2.5.1. Biểu đồ hoạt động đặt lịch khám và phân công bác sĩ.....	40
2.5.2. Biểu đồ hoạt động chức năng khám bệnh, lập kế hoạch điều trị và thanh toán.....	41
2.5.3. Biểu đồ hoạt động chức năng quản lý thuốc và tồn kho.....	42

2.5.4. Biểu đồ hoạt động chức năng thống kê và AI phân tích đề xuất.....	43
2.5.5. Biểu đồ hoạt động chức năng quản lý hồ sơ bệnh nhân và lịch sử điều trị	44
2.6. Thiết kế cơ sở dữ liệu.....	45
2.6.1. Các bảng trong cơ sở dữ liệu.....	45
2.6.2. Biểu đồ quan hệ của các bảng.....	52
CHƯƠNG 3. TRIỂN KHAI HỆ THỐNG VÀ KẾT QUẢ ĐẠT ĐƯỢC.....	53
3.1. Cấu hình triển khai.....	53
3.1.1. Backend – Spring Boot.....	53
3.1.2. Frontend – Vue.js.....	53
3.1.3. Cơ sở dữ liệu – PostgreSQL.....	53
3.2. Cấu trúc thư mục.....	54
3.2.1. Backend chính.....	54
3.2.1. Frontend.....	55
3.3. Triển khai hệ thống.....	56
3.3.1. Các bước triển khai.....	56
3.3.2. Kết quả sau triển khai.....	57
3.4. Kết quả đạt được.....	58
3.4.1. Giao diện bệnh nhân.....	58
3.4.2. Giao diện bác sĩ.....	63
3.4.3. Giao diện quản trị (ADMIN).....	66
3.4.4. Giao diện nhân viên.....	69
KẾT LUẬN.....	73
TÀI LIỆU THAM KHẢO.....	75

DANH MỤC CÁC BẢNG

Bảng 2.1 Xác định Actor	15
Bảng 2.2 Xác định Usecase.....	18
Bảng 2.3 Đặc tả usecase đặt lịch khám	25
Bảng 2.4 Đặc tả Usecase xác nhận hủy cập nhật lịch hẹn.....	27
Bảng 2.5 Đặc tả usecase Phân công bác sĩ cho lịch hẹn.....	29
Bảng 2.6 Đặc tả Usecase Lập kế hoạch điều trị.....	31
Bảng 2.7 Đặc tả Usecase Kê đơn thuốc.....	33
Bảng 2.8 Đặc tả Usecase Tạo hóa đơn và thanh toán.....	34
Bảng 2.9 Đặc tả Usecase Quản lý thuốc và tồn kho.....	37
Bảng 2.10 Đặc tả Usecase Thống kê và AI phân tích đề xuất.....	39

DANH MỤC CÁC HÌNH

Hình 1.1: Framework thư viện với các tài nguyên không thể thiếu.....	5
Hình 1.2: Mô hình MVC.....	5
Hình 1.3: Logo Vue.js.....	6
Hình 1.4: Logo Spring Boot.....	7
Hình 1.5: Sơ đồ RESTful API.....	8
Hình 1.6: Logo JWT.....	9
Hình 1.7: Logo Visual Studio Code.....	11
Hình 1.8: Logo pgAdmin.....	11
Hình 1.9: Logo Gemini AI.....	12
Hình 1.10: Logo VNPAY.....	12
Hình 2.1: Sơ đồ Usecase khách hàng.....	21
Hình 2.2: Sơ đồ Usecase Admin.....	22
Hình 2.3: Sơ đồ Usecase Bác sĩ.....	23
Hình 2.4: Sơ đồ Usecase Nhân viên.....	23
Hình 2.5: Sơ đồ Usecase tổng quát.....	24
Hình 2.6: Đặc tả Usecase đặt lịch khám.....	24
Hình 2.7: Đặc tả Usecase xác nhận/ Hủy / Cập nhật lịch hẹn.....	26
Hình 2.8: Đặc tả Usecase Phân công bác sĩ cho lịch hẹn.....	28
Hình 2.9: Đặc tả Usecase Lập kế hoạch điều trị.....	30
Hình 2.10: Đặc tả Usecase kê đơn thuốc.....	32
Hình 2.11: Đặc tả Usecase Tạo hóa đơn và thanh toán.....	34
Hình 2.12: Đặc tả Usecase Quản lý thuốc và tồn kho.....	36
Hình 2.13: Đặc tả Usecase Thống kê và AI phân tích đề xuất.....	38
Hình 2.14: Biểu đồ hoạt động đặt lịch khám và phân công bác sĩ.....	40
Hình 2.15: Biểu đồ hoạt động chức năng khám bệnh.....	41
Hình 2.16: Biểu đồ hoạt động chức năng quản lý thuốc và tồn kho.....	42
Hình 2.17: Biểu đồ hoạt động chức năng thống kê và AI phân tích đề xuất.....	43
Hình 2.18: Biểu đồ hoạt động chức năng quản lý hồ sơ bệnh nhân và lịch sử....	44
Hình 2.19: Bảng users.....	45
Hình 2.20: Bảng patients.....	46

Hình 2.21: Bảng appointments.....	47
Hình 2.22: Bảng doctor_shift_capacities.....	48
Hình 2.23: Bảng treatment_plans.....	49
Hình 2.24: Bảng prescriptions.....	50
Hình 2.25: Bảng medicines.....	51
Hình 2.26: Sơ đồ quan hệ ERD.....	52
Hình 3.1: Cấu trúc thư mục backend chính.....	54
Hình 3.2: Cấu trúc thư mục frontend.....	55
Hình 3.3: Trang chủ phòng khám.....	58
Hình 3.4: Đăng Nhập.....	58
Hình 3.5: Đăng Ký.....	59
Hình 3.6: Đặt lịch khám.....	59
Hình 3.7: Xem lịch hẹn.....	60
Hình 3.8: Xem thông tin bác sĩ.....	60
Hình 3.9: Xem dịch vụ nha khoa.....	61
Hình 3.10: Xem lịch sử khám.....	61
Hình 3.11: Thanh toán VNPay.....	62
Hình 3.12: Hồ sơ cá nhân.....	62
Hình 3.13: Danh sách lịch khám.....	63
Hình 3.14: Thông tin bệnh nhân.....	63
Hình 3.15: Kế hoạch điều trị.....	64
Hình 3.16: Kê đơn thuốc.....	64
Hình 3.17: Quản lý tái khám.....	65
Hình 3.18: Theo dõi hồ sơ điều trị.....	65
Hình 3.19: Dashboard thống kê.....	66
Hình 3.20: Quản lý tài khoản.....	66
Hình 3.21: Quản lý lịch hẹn.....	67
Hình 3.22: Quản lý dịch vụ.....	67
Hình 3.23: Quản lý thuốc.....	68
Hình 3.24: Quản lý hóa đơn.....	68
Hình 3.25: Quản lý doanh thu.....	69
Hình 3.26: Tiếp nhận bệnh nhân.....	69
Hình 3.27: Xác nhận lịch hẹn.....	70

Hình 3.28: Quản lý danh sách lịch khám.....	70
Hình 3.29: Hỗ trợ phân công bác sĩ.....	71
Hình 3.30: Quản lý hóa đơn thanh toán.....	71
Hình 3.31: Xác nhận thanh toán VNPay	72
Hình 3.32: Hỗ trợ cập nhật thông tin bệnh nhân	72

DANH MỤC KÝ HIỆU, CHỮ VIẾT TẮT

STT	Ký hiệu viết tắt	Chữ viết đầy đủ	Ghi chú
1	HTML	HyperText Markup Language	Ngôn ngữ đánh dấu tạo cấu trúc trang web
2	CSS	Cascading Style Sheets	Ngôn ngữ định kiểu cho giao diện web
3	JS	JavaScript	Ngôn ngữ lập trình phía client
4	SPA	Single Page Application	Ứng dụng một trang (Vue.js)
5	API	Application Programming Interface	Giao diện lập trình ứng dụng
6	REST	Representational State Transfer	Kiến trúc giao tiếp API
7	RESTful API	REST Application Programming Interface	Chuẩn API trong dự án
8	JSON	JavaScript Object Notation	Định dạng trao đổi dữ liệu
9	JWT	JSON Web Token	Token xác thực & phân quyền
10	MVC	Model – View – Controller	Mô hình kiến trúc phần mềm
11	MVVM	Model – View – ViewModel	Mô hình giao diện (Vue.js)
12	CRUD	Create – Read – Update – Delete	Tác vụ quản lý dữ liệu
13	ORM	Object–Relational Mapping	Spring JPA/Hibernate
14	SQL	Structured Query Language	Ngôn ngữ truy vấn dữ liệu
15	DB	Database	Cơ sở dữ liệu
16	RDBMS	Relational Database Management System	Hệ quản trị CSDL quan hệ
17	QR	Quick Response	Mã QR thanh toán

18	AI	Artificial Intelligence	Trí tuệ nhân tạo
19	NLP	Natural Language Processing	Xử lý ngôn ngữ tự nhiên
20	UI	User Interface	Giao diện người dùng
21	UX	User Experience	Trải nghiệm người dùng
22	VS Code	Visual Studio Code	Công cụ lập trình frontend
23	IDE	Integrated Development Environment	Môi trường phát triển tích hợp
24	DBMS	Database Management System	Hệ thống quản trị CSDL
25	URL	Uniform Resource Locator	Đường dẫn tài nguyên web
26	HTTP	HyperText Transfer Protocol	Giao thức truyền tải web
27	HTTPS	HyperText Transfer Protocol Secure	Giao thức web bảo mật
28	JDK	Java Development Kit	Bộ công cụ lập trình Java
29	JVM	Java Virtual Machine	Máy ảo Java
30	JRE	Java Runtime Environment	Môi trường chạy Java
31	DTO	Data Transfer Object	Đối tượng truyền dữ liệu
32	DAO	Data Access Object	Lớp truy cập dữ liệu
33	QR Pay	QR Payment	Thanh toán bằng mã QR
34	VNPay	Vietnam Payment	Cổng thanh toán QR
35	OTP	One Time Password	Mật khẩu xác thực một lần
36	SSL	Secure Sockets Layer	Bảo mật truyền dữ liệu
37	CPU	Central Processing Unit	Bộ xử lý trung tâm
38	RAM	Random Access Memory	Bộ nhớ truy cập ngẫu nhiên
39	CDN	Content Delivery Network	Mạng phân phối nội dung

MỞ ĐẦU

1. Mục đích thực hiện đề tài

Hiện nay, nhu cầu ứng dụng công nghệ thông tin trong lĩnh vực y tế ngày càng phổ biến, đặc biệt tại các phòng khám nha khoa tư nhân. Việc quản lý thủ công gây khó khăn trong quá trình lưu trữ hồ sơ bệnh nhân, quản lý lịch hẹn và thanh toán.

Đề tài “Hệ thống quản lý nha khoa tư nhân sử dụng kiến trúc VueJS và Spring Boot” được thực hiện nhằm xây dựng một hệ thống hỗ trợ quản lý bệnh nhân, bác sĩ, lịch hẹn, kế hoạch điều trị, hóa đơn và thanh toán trực tuyến.

Hệ thống hướng đến mục tiêu giúp phòng khám tối ưu quy trình vận hành, giảm sai sót trong quản lý và nâng cao trải nghiệm người dùng.

Mục tiêu đề tài

Xây dựng website quản lý phòng khám nha khoa.

Hỗ trợ bệnh nhân đặt lịch khám trực tuyến.

Quản lý bệnh nhân, bác sĩ, lịch hẹn và hồ sơ điều trị.

Quản lý hóa đơn và thanh toán VNPay.

Xây dựng hệ thống phân quyền bằng JWT Authentication.

Thiết kế giao diện trực quan, dễ sử dụng và responsive.

2. Đối tượng, phạm vi nghiên cứu

Đối tượng nghiên cứu

Đề tài tập trung nghiên cứu và xây dựng hệ thống quản lý nha khoa trên nền tảng web với các công nghệ chính:

VueJS: xây dựng giao diện frontend.

Spring Boot: phát triển backend RESTful API.

PostgreSQL: quản lý cơ sở dữ liệu.

JWT Authentication: xác thực và phân quyền.

VNPay API: tích hợp thanh toán trực tuyến.

WebSocket và Email SMTP: gửi thông báo realtime và email.

Phạm vi nghiên cứu

Đề tài tập trung xây dựng hệ thống quản lý phòng khám nha khoa trên nền tảng web với các chức năng chính như quản lý bệnh nhân, bác sĩ, lịch hẹn, kế hoạch điều trị, đơn thuốc, hóa đơn và thanh toán trực tuyến.

Hệ thống hỗ trợ bệnh nhân đặt lịch khám online, theo dõi lịch hẹn và thông tin điều trị. Đồng thời, quản trị viên và nhân viên có thể quản lý thông tin người dùng, lịch khám, dịch vụ, thuốc và thống kê hoạt động của phòng khám.

Đề tài sử dụng VueJS để phát triển frontend, Spring Boot để xây dựng RESTful API và PostgreSQL để lưu trữ dữ liệu. Ngoài ra, hệ thống tích hợp JWT Authentication và VNPay nhằm tăng tính bảo mật và hỗ trợ thanh toán trực tuyến.

Do giới hạn về thời gian và phạm vi thực hiện, đề tài chưa triển khai trên môi trường thực tế quy mô lớn và chưa tích hợp đầy đủ các chức năng AI nâng cao.

Bố cục của đồ án

Đồ án tốt nghiệp được bố cục thành các phần chính như sau:

Chương 1: Cơ sở lý thuyết

Chương 2: Phân tích và thiết kế hệ thống

Chương 3: Triển khai hệ thống và kết quả đạt được

Kết luận và hướng phát triển

CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

1.1. Tổng quan về hệ thống quản lý nha khoa

1.1.1. Tổng quan về hệ thống quản lý phòng khám nha khoa

Hệ thống quản lý phòng khám nha khoa là một hệ thống phần mềm được xây dựng nhằm hỗ trợ quá trình vận hành, quản lý và cung cấp dịch vụ tại các phòng khám nha khoa. Hệ thống giúp số hóa các quy trình truyền thống như quản lý bệnh nhân, đặt lịch khám, phân công bác sĩ, ghi nhận kết quả khám, lập kế hoạch điều trị, kê đơn thuốc, tạo hóa đơn và theo dõi thanh toán.

Trong mô hình quản lý truyền thống, nhiều phòng khám vẫn sử dụng sổ sách, giấy tờ hoặc các file lưu trữ rời rạc để quản lý thông tin bệnh nhân và lịch hẹn. Cách làm này dễ dẫn đến tình trạng trùng lịch, thất lạc thông tin, khó tra cứu lịch sử điều trị và mất nhiều thời gian trong quá trình xử lý nghiệp vụ. Vì vậy, việc xây dựng hệ thống quản lý nha khoa trực tuyến giúp nâng cao hiệu quả vận hành, giảm sai sót và cải thiện trải nghiệm của bệnh nhân.

Một hệ thống quản lý phòng khám nha khoa hiện đại thường bao gồm các nhóm chức năng chính như: quản lý tài khoản người dùng, quản lý bệnh nhân, quản lý bác sĩ, đặt lịch khám, phân công bác sĩ, quản lý kế hoạch điều trị, kê đơn thuốc, quản lý hóa đơn, thanh toán và thống kê báo cáo. Các chức năng này có mối liên hệ chặt chẽ với nhau, tạo thành một quy trình khép kín từ lúc bệnh nhân đặt lịch đến khi hoàn tất khám, điều trị và thanh toán.

1.1.2. Quy trình đặt lịch khám nha khoa

Đặt lịch khám nha khoa là một trong những nghiệp vụ quan trọng của phòng khám, giúp bệnh nhân chủ động lựa chọn thời gian khám phù hợp và giúp phòng khám kiểm soát số lượng bệnh nhân trong từng ca. Thông qua hệ thống đặt lịch trực tuyến, bệnh nhân có thể đăng nhập, chọn ngày khám, chọn ca khám, nhập ghi chú hoặc nhu cầu khám và gửi yêu cầu đặt lịch đến phòng khám.

Sau khi bệnh nhân gửi yêu cầu, hệ thống sẽ tạo lịch hẹn với trạng thái chờ xác nhận. Nhân viên hoặc quản trị viên sẽ tiếp nhận lịch hẹn, kiểm tra thông tin, xác nhận hoặc hủy lịch nếu cần thiết. Đối với các lịch hẹn được tiếp nhận, hệ thống tiếp tục hỗ trợ phân công bác sĩ phù hợp dựa trên lịch làm việc và sức chứa của từng bác sĩ.

Việc quản lý lịch hẹn bằng hệ thống giúp hạn chế tình trạng trùng lịch, quá tải ca khám hoặc bỏ sót yêu cầu của bệnh nhân. Đồng thời, bệnh nhân có thể theo dõi trạng thái lịch

hẹn của mình như chờ xác nhận, đã xác nhận, đã phân công bác sĩ, đã hoàn thành hoặc đã hủy.

1.1.3. Quản lý hồ sơ bệnh nhân

Hồ sơ bệnh nhân là nơi lưu trữ toàn bộ thông tin liên quan đến bệnh nhân trong quá trình sử dụng dịch vụ tại phòng khám. Các thông tin cơ bản thường bao gồm họ tên, ngày sinh, giới tính, số điện thoại, email, địa chỉ và các thông tin liên hệ cần thiết.

Ngoài thông tin cá nhân, hồ sơ bệnh nhân còn liên kết với nhiều dữ liệu quan trọng khác như lịch hẹn, lịch sử khám, kết quả khám, kế hoạch điều trị, đơn thuốc, hóa đơn và lịch sử thanh toán. Việc lưu trữ tập trung các thông tin này giúp bác sĩ dễ dàng tra cứu lịch sử điều trị, nắm được tình trạng răng miệng của bệnh nhân và đưa ra phương án điều trị phù hợp.

Quản lý hồ sơ bệnh nhân bằng hệ thống phần mềm giúp giảm phụ thuộc vào giấy tờ, hạn chế thất lạc dữ liệu và tăng tính chính xác trong quá trình theo dõi điều trị. Đây là một thành phần quan trọng trong việc nâng cao chất lượng chăm sóc bệnh nhân tại phòng khám nha khoa.

1.1.4. Quản lý lịch hẹn và phân công bác sĩ

Quản lý lịch hẹn là quá trình tiếp nhận, xác nhận, cập nhật hoặc hủy các lịch khám của bệnh nhân. Trong hệ thống nha khoa, lịch hẹn thường bao gồm các thông tin như bệnh nhân, ngày khám, ca khám, bác sĩ phụ trách, trạng thái lịch hẹn và ghi chú của bệnh nhân.

Phân công bác sĩ là bước quan trọng sau khi lịch hẹn được tiếp nhận. Hệ thống cần hỗ trợ kiểm tra lịch làm việc của bác sĩ, số lượng bệnh nhân tối đa trong một ca khám và trạng thái khả dụng của bác sĩ. Dựa trên các thông tin này, nhân viên hoặc admin có thể lựa chọn bác sĩ phù hợp cho từng lịch hẹn.

Việc phân công bác sĩ hợp lý giúp phòng khám tối ưu nguồn lực, tránh quá tải cho bác sĩ và đảm bảo bệnh nhân được thăm khám đúng chuyên môn. Đồng thời, sau khi lịch hẹn được phân công, hệ thống có thể gửi email hoặc thông báo đến bệnh nhân để xác nhận thông tin lịch khám.

1.1.5. Quản lý kế hoạch điều trị

Kế hoạch điều trị là nội dung do bác sĩ lập sau khi thăm khám và đánh giá tình trạng răng miệng của bệnh nhân. Một kế hoạch điều trị có thể bao gồm các dịch vụ cần thực hiện, mô tả phương án điều trị, chi phí dự kiến, thời gian thực hiện, trạng thái điều trị và ghi chú chuyên môn.

Trong lĩnh vực nha khoa, nhiều trường hợp điều trị không hoàn thành trong một lần khám mà cần theo dõi qua nhiều giai đoạn, ví dụ như niềng răng, trồng răng, điều trị tủy, phục hình răng hoặc các dịch vụ điều trị chuyên sâu khác. Vì vậy, việc quản lý kế hoạch điều trị giúp bác sĩ và bệnh nhân theo dõi được tiến độ điều trị một cách rõ ràng.

Hệ thống quản lý kế hoạch điều trị giúp lưu trữ thông tin điều trị tập trung, hỗ trợ cập nhật tiến độ sau từng lần khám và tạo cơ sở để tính toán chi phí hoặc tạo hóa đơn. Đồng thời, bệnh nhân có thể xem lại kế hoạch điều trị của mình, từ đó chủ động hơn trong quá trình theo dõi sức khỏe răng miệng.

1.1.6. Quản lý đơn thuốc và thuốc

Sau quá trình khám hoặc điều trị, bác sĩ có thể kê đơn thuốc cho bệnh nhân nếu cần thiết. Đơn thuốc thường bao gồm tên thuốc, số lượng, liều lượng, cách dùng và các ghi chú liên quan. Việc kê đơn thuốc cần đảm bảo chính xác, rõ ràng và phù hợp với tình trạng của bệnh nhân.

Bên cạnh việc quản lý đơn thuốc, hệ thống cũng cần quản lý danh mục thuốc của phòng khám. Thông tin thuốc có thể bao gồm tên thuốc, đơn vị tính, giá bán, mô tả, số lượng tồn kho và trạng thái sử dụng. Nếu hệ thống có quản lý lô thuốc, cần lưu thêm thông tin ngày nhập, hạn sử dụng, số lượng nhập và số lượng còn lại.

Quản lý thuốc và tồn kho giúp phòng khám kiểm soát được tình trạng sử dụng thuốc, hạn chế thiếu thuốc khi kê đơn và cảnh báo các thuốc sắp hết hoặc gần hết hạn. Đây là chức năng hỗ trợ quan trọng trong quá trình điều trị và vận hành phòng khám.

1.1.7. Quản lý hóa đơn và thanh toán

Quản lý hóa đơn là quá trình ghi nhận các chi phí phát sinh trong quá trình bệnh nhân sử dụng dịch vụ tại phòng khám. Hóa đơn có thể bao gồm chi phí khám, dịch vụ điều trị, thuốc và các khoản phí liên quan khác.

Trong hệ thống nha khoa, hóa đơn thường được tạo sau khi bệnh nhân hoàn tất khám hoặc sau khi bác sĩ lập kế hoạch điều trị, kê đơn thuốc. Nhân viên có thể kiểm tra các khoản mục chi phí, tạo hóa đơn và ghi nhận trạng thái thanh toán. Bệnh nhân có thể thanh toán bằng tiền mặt hoặc thông qua cổng thanh toán trực tuyến như VNPay.

Việc tích hợp thanh toán trực tuyến giúp quá trình thanh toán trở nên nhanh chóng, minh bạch và thuận tiện hơn. Đồng thời, hệ thống có thể lưu lại lịch sử giao dịch, cập nhật trạng thái hóa đơn và gửi thông báo thanh toán thành công đến bệnh nhân.

1.1.8. Vai trò của AI trong hỗ trợ quản lý nha khoa

Trí tuệ nhân tạo đang được ứng dụng ngày càng nhiều trong các hệ thống quản lý hiện đại nhằm hỗ trợ phân tích dữ liệu, gợi ý thông tin và nâng cao hiệu quả ra quyết định.

Trong hệ thống quản lý nha khoa, AI có thể được sử dụng để hỗ trợ tư vấn dịch vụ, đề xuất bác sĩ phù hợp và phân tích dữ liệu vận hành của phòng khám.

Đối với bệnh nhân, AI có thể hỗ trợ gợi ý dịch vụ nha khoa dựa trên nhu cầu hoặc vấn đề mà bệnh nhân chia sẻ. Ví dụ, khi bệnh nhân mô tả tình trạng đau răng, muốn tẩy trắng răng hoặc cần tư vấn niềng răng, hệ thống có thể gợi ý nhóm dịch vụ phù hợp để bệnh nhân tham khảo trước khi đặt lịch.

Đối với admin, AI có thể hỗ trợ phân tích các dữ liệu như doanh thu, chi phí, lợi nhuận, số lượng lịch hẹn, số lượng bệnh nhân, tần suất sử dụng dịch vụ và tình hình sử dụng thuốc. Từ các dữ liệu này, hệ thống có thể đưa ra nhận xét hoặc đề xuất nhằm hỗ trợ admin đánh giá hiệu quả hoạt động và định hướng phát triển phòng khám.

Việc tích hợp AI giúp hệ thống không chỉ dừng lại ở mức quản lý dữ liệu, mà còn hỗ trợ phân tích, tư vấn và đề xuất thông tin, góp phần nâng cao chất lượng dịch vụ và hiệu quả vận hành của phòng khám nha khoa.

1.2. Cơ sở lý thuyết về công nghệ

1.2.1. Ngôn ngữ HTML/CSS

HTML và **CSS** là hai công nghệ cơ bản trong phát triển giao diện web. HTML được sử dụng để xây dựng cấu trúc nội dung trang web, trong khi CSS dùng để định dạng giao diện như màu sắc, bố cục và hiển thị trên nhiều thiết bị.

Trong đề tài, HTML/CSS được sử dụng kết hợp với VueJS để xây dựng giao diện người dùng trực quan và responsive.

1.2.2. Ngôn ngữ JavaScript

JavaScript là ngôn ngữ lập trình phổ biến dùng để xử lý tương tác trên website. Ngôn ngữ này hỗ trợ cập nhật dữ liệu động, xử lý sự kiện và cải thiện trải nghiệm người dùng.

Trong hệ thống, JavaScript được sử dụng kết hợp với VueJS để xây dựng các chức năng frontend như đăng nhập, đặt lịch khám và quản lý dữ liệu realtime.

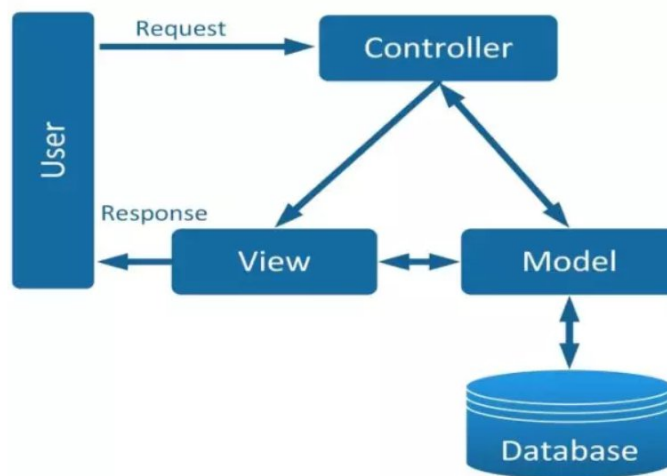
1.2.3. Vue.js Framework

VueJS được sử dụng như nền tảng phát triển frontend cho hệ thống quản lý nha khoa. Framework này hỗ trợ xây dựng giao diện động, phản hồi nhanh và tổ chức mã nguồn theo component, giúp hệ thống dễ mở rộng và quản lý. Các chức năng dành cho bệnh nhân, bác sĩ và quản trị viên đều được triển khai thông qua VueJS.



Hình 1.1: Framework thư viện với các tài nguyên không thể thiếu

1.2.3.1. Mô hình MVC là gì?



Hình 1.2: Mô hình MVC

MVC (Model – View – Controller) là mô hình kiến trúc giúp tách biệt dữ liệu, giao diện và xử lý nghiệp vụ trong ứng dụng.

Model: quản lý dữ liệu và nghiệp vụ.

View: hiển thị giao diện cho người dùng.

Controller: tiếp nhận và xử lý yêu cầu từ người dùng.

Mô hình MVC giúp hệ thống dễ phát triển, bảo trì và mở rộng.

1.2.3.2. Vue.js framework là gì?

VueJS là framework JavaScript dùng để xây dựng giao diện web theo mô hình **SPA** (Single Page Application). VueJS hỗ trợ xây dựng giao diện bằng component giúp dễ tái sử dụng và quản lý mã nguồn.

Trong đề tài, VueJS được sử dụng để phát triển giao diện cho bệnh nhân, bác sĩ và quản trị viên.



Hình 1.3: Logo Vue.js

1.2.3.3. Đặc điểm nổi bật của Vue.js

- Cú pháp đơn giản, dễ học.
- Hỗ trợ xây dựng giao diện bằng component.
- Cập nhật dữ liệu realtime hiệu quả.
- Hỗ trợ SPA giúp website hoạt động mượt mà.
- Dễ tích hợp với các thư viện và API backend.

1.2.3.4. Các thành phần chính trong Vue.js

VueJS gồm các thành phần chính như:

- Template: xây dựng giao diện.
- Script: xử lý logic và dữ liệu.
- Style: định dạng giao diện.
- Directives: hỗ trợ thao tác dữ liệu và DOM.
- Lifecycle Hooks: xử lý theo vòng đời component.

1.2.3.5. Ưu điểm của Vue.js

- Dễ học và dễ sử dụng.
- Hiệu năng tốt nhờ Virtual DOM.
- Hỗ trợ phát triển SPA hiệu quả.
- Dễ bảo trì và mở rộng hệ thống.
- Tài liệu phong phú và cộng đồng lớn.

1.2.3.6. Nhược điểm của Vue.js

- Hệ sinh thái chưa lớn bằng React hoặc Angular.
- Một số thư viện hỗ trợ còn hạn chế.

Quản lý state trong hệ thống lớn có thể phức tạp.

1.2.4. Ngôn ngữ lập trình Java

Java là ngôn ngữ lập trình hướng đối tượng có tính ổn định và bảo mật cao. Java hỗ trợ phát triển các hệ thống web quy mô lớn và dễ mở rộng.

Trong đề tài, Java được sử dụng kết hợp với Spring Boot để xây dựng backend và xử lý nghiệp vụ của hệ thống.

1.2.5. Spring Boot

Spring Boot là framework thuộc hệ sinh thái Spring dùng để phát triển ứng dụng Java và RESTful API. Framework này hỗ trợ cấu hình tự động, tích hợp bảo mật và kết nối cơ sở dữ liệu hiệu quả.

Trong đề tài, Spring Boot được sử dụng để xây dựng backend, xử lý nghiệp vụ và quản lý dữ liệu hệ thống.



Hình 1.4: Logo Spring Boot

1.2.6. Cơ sở dữ liệu PostgreSQL

1.2.6.1. Giới thiệu về cơ sở dữ liệu PostgreSQL

PostgreSQL là hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở được sử dụng phổ biến trong phát triển ứng dụng web. PostgreSQL hỗ trợ lưu trữ dữ liệu ổn định, bảo mật cao và xử lý tốt các hệ thống có lượng dữ liệu lớn.

Trong đề tài, PostgreSQL được sử dụng để lưu trữ dữ liệu bệnh nhân, bác sĩ, lịch hẹn, hóa đơn và các thông tin của hệ thống quản lý nha khoa.

Đặc điểm của PostgreSQL

Hỗ trợ quản lý dữ liệu quan hệ ổn định.

Hỗ trợ truy vấn SQL mạnh mẽ.

Bảo mật và toàn vẹn dữ liệu cao.

Hỗ trợ backup và phục hồi dữ liệu.

Hoạt động tốt với Spring Boot và JPA.

1.2.6.2. Ưu điểm và nhược điểm của PostgreSQL

Ưu điểm:

- Miễn phí và mã nguồn mở.
- Hiệu năng ổn định với hệ thống lớn.
- Hỗ trợ nhiều kiểu dữ liệu.
- Dễ tích hợp với ứng dụng web.
- Có cộng đồng hỗ trợ lớn.

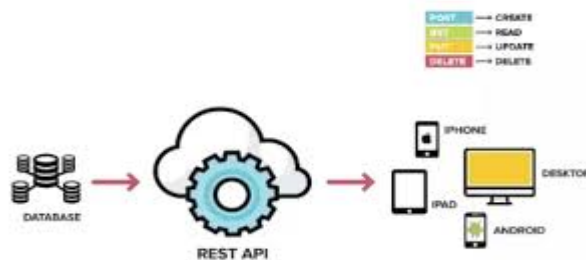
Nhược điểm:

- Cấu hình ban đầu tương đối phức tạp với người mới.
- Tốn nhiều tài nguyên hơn một số hệ quản trị cơ sở dữ liệu nhẹ.
- Một số thao tác tối ưu hiệu năng yêu cầu kiến thức chuyên sâu.

1.2.7. Kiến trúc giao tiếp: RESTful API

RESTful API là kiến trúc dùng để xây dựng và giao tiếp dữ liệu giữa frontend và backend thông qua giao thức HTTP. RESTful API cho phép các chức năng trong hệ thống trao đổi dữ liệu dễ dàng, nhanh chóng và độc lập.

Trong đề tài, RESTful API được xây dựng bằng Spring Boot để xử lý các chức năng như đăng nhập, quản lý bệnh nhân, lịch hẹn, kế hoạch điều trị và thanh toán.



Hình 1.5: Sơ đồ RESTful API

Đặc điểm của RESTful API

- Giao tiếp thông qua giao thức HTTP.
- Dữ liệu thường được trao đổi dưới dạng JSON.
- Hỗ trợ các phương thức như GET, POST, PUT và DELETE.
- Dễ tích hợp giữa frontend và backend.
- Hỗ trợ phát triển hệ thống web và mobile hiệu quả.

1.2.8. JWT Authentication

JWT (JSON Web Token) là cơ chế xác thực người dùng dựa trên token được sử dụng phổ biến trong các hệ thống web hiện đại. Sau khi đăng nhập thành công, hệ thống sẽ tạo token và gửi cho người dùng để xác thực trong các lần truy cập tiếp theo.

Trong đề tài, JWT Authentication được sử dụng kết hợp với Spring Security nhằm hỗ trợ đăng nhập, xác thực người dùng và phân quyền theo vai trò như Admin, Bác sĩ, Nhân viên và Bệnh nhân.

Đặc điểm của JWT Authentication

Xác thực người dùng thông qua token.

Hỗ trợ phân quyền người dùng.

Tăng tính bảo mật cho hệ thống.

Giảm lưu trữ session phía server.

Phù hợp với hệ thống RESTful API và SPA.



Hình 1.6: Logo JWT

1.2.8.1. Ưu điểm của JWT

Không lưu trạng thái (stateless) → server nhẹ, dễ scale.

Nhanh, chỉ cần kiểm tra chữ ký token.

Dễ tích hợp với ứng dụng mobile, web SPA, microservices.

Hỗ trợ phân quyền (role, permission).

Tránh lỗi session timeout hoặc session lưu sai người dùng.

1.2.8.2. Nhược điểm của JWT

Token bị lộ có thể gây mất an toàn dữ liệu.

Khó thu hồi token trước thời gian hết hạn.

Cần xử lý bảo mật token cẩn thận ở phía client..

1.3. Kiến trúc và mô hình triển khai hệ thống

Hệ thống quản lý nha khoa được xây dựng theo mô hình client-server với frontend và backend hoạt động độc lập. Kiến trúc này giúp hệ thống dễ bảo trì, mở rộng và nâng cao hiệu năng xử lý.

Frontend sử dụng VueJS để xây dựng giao diện người dùng, backend sử dụng Spring Boot để xử lý nghiệp vụ và cung cấp RESTful API. PostgreSQL được sử dụng để lưu trữ dữ liệu hệ thống.

1.3.1. Kiến trúc Client – Server

Kiến trúc **Client – Server** là mô hình trong đó client gửi yêu cầu đến server và server xử lý yêu cầu trước khi trả kết quả về cho client.

Trong đề tài, frontend VueJS đóng vai trò client, backend Spring Boot đóng vai trò server xử lý dữ liệu và kết nối cơ sở dữ liệu PostgreSQL.

1.3.2. Mô hình MVC

MVC (Model – View – Controller) là mô hình kiến trúc giúp tách biệt giao diện, dữ liệu và xử lý nghiệp vụ trong hệ thống.

Model: quản lý dữ liệu và nghiệp vụ.

View: hiển thị giao diện người dùng.

Controller: tiếp nhận và xử lý yêu cầu từ client.

Mô hình MVC giúp hệ thống dễ phát triển và bảo trì.

1.3.3. Quy trình giao tiếp hệ thống

Người dùng thao tác trên giao diện VueJS và gửi yêu cầu đến backend thông qua RESTful API. Backend Spring Boot tiếp nhận yêu cầu, xử lý nghiệp vụ và truy xuất dữ liệu từ PostgreSQL trước khi trả kết quả về frontend dưới dạng JSON.

Quy trình này giúp frontend và backend hoạt động độc lập và dễ mở rộng hệ thống.

1.3.4. Mô hình triển khai hệ thống

Hệ thống được triển khai theo mô hình web application gồm frontend, backend và cơ sở dữ liệu hoạt động riêng biệt.

Frontend: VueJS

Backend: Spring Boot RESTful API

Database: PostgreSQL

Mô hình triển khai này giúp hệ thống dễ nâng cấp và thuận tiện trong quá trình bảo trì.

1.4. Công cụ và cài đặt

1.4.1. Visual Studio Code

1.4.1.1. Giới thiệu về Visual Studio Code

Visual Studio Code là môi trường lập trình hỗ trợ phát triển nhiều ngôn ngữ và framework khác nhau. Công cụ này hỗ trợ viết mã nguồn, quản lý extension và kiểm thử ứng dụng hiệu quả.

Trong đề tài, Visual Studio Code được sử dụng để phát triển frontend VueJS và backend Spring Boot.



Hình 1.7: Logo Visual Studio Code

1.4.2. PgAdmin 4

1.4.2.1. Giới thiệu về pgAdmin4

pgAdmin 4 là công cụ hỗ trợ quản lý cơ sở dữ liệu PostgreSQL thông qua giao diện trực quan. Công cụ hỗ trợ tạo database, thực hiện truy vấn SQL và quản lý dữ liệu dễ dàng.

Trong đề tài, pgAdmin được sử dụng để quản lý dữ liệu của hệ thống nha khoa.



Hình 1.8: Logo pgAdmin

1.4.3. GeminiAI

Gemini AI là nền tảng trí tuệ nhân tạo của Google hỗ trợ xử lý và phân tích dữ liệu thông minh. Trong đề tài, AI được định hướng hỗ trợ phân tích dữ liệu và đề xuất thông tin phù hợp cho hệ thống.



Hình 1.9: Logo Gemini AI

1.4.4. VNPay

VNPay là cổng thanh toán điện tử phổ biến tại Việt Nam, hỗ trợ thanh toán qua QR Code, ATM và Mobile Banking.

Trong đề tài, VNPay được tích hợp nhằm hỗ trợ thanh toán hóa đơn trực tuyến cho bệnh nhân.



Hình 1.10: Logo VNPAY

1.4.5. Cài đặt Node.js và NPM

Bước 1: Truy cập <https://nodejs.org>

Bước 2: Chọn bản LTS để đảm bảo ổn định.

Bước 3: Cài đặt và kiểm tra bằng lệnh `node -v` và `npm -v`

```
C:\Users\ADMIN>node -v
v22.19.0

C:\Users\ADMIN>npm -v
10.9.3
```

1.4.6. Cài đặt Java JDK

Bước 1: Truy cập trang: <https://adoptium.net>

Bước 2: Tải Temurin JDK 17 hoặc 21.

Bước 3: Cài đặt và kiểm tra bằng lệnh `java -version` và `javac -version`

```
C:\Users\ADMIN>java -version
java version "24.0.1" 2025-04-15
Java(TM) SE Runtime Environment (build 24.0.1+9-30)
Java HotSpot(TM) 64-Bit Server VM (build 24.0.1+9-30, mixed mode, sharing)

C:\Users\ADMIN>javac -version
javac 24.0.1
```

CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

2.1. Mô tả bài toán

Hệ thống quản lý nha khoa tư nhân được xây dựng nhằm hỗ trợ số hóa quy trình quản lý và vận hành phòng khám nha khoa. Hệ thống giúp quản lý bệnh nhân, lịch hẹn, kế hoạch điều trị, đơn thuốc, hóa đơn và thanh toán trực tuyến một cách hiệu quả hơn so với phương pháp thủ công.

Đối với bệnh nhân, hệ thống hỗ trợ đăng ký tài khoản, đăng nhập, đặt lịch khám trực tuyến, theo dõi lịch hẹn, xem kế hoạch điều trị, đơn thuốc và thanh toán hóa đơn. Ngoài ra, bệnh nhân có thể nhận thông báo và email từ hệ thống.

Đối với bác sĩ, hệ thống hỗ trợ quản lý lịch khám, cập nhật hồ sơ khám bệnh, lập kế hoạch điều trị, kê đơn thuốc và theo dõi thông tin bệnh nhân.

Đối với nhân viên phòng khám, hệ thống hỗ trợ tiếp nhận lịch hẹn, xác nhận lịch khám, quản lý bệnh nhân, thuốc, hóa đơn và hỗ trợ thanh toán.

Đối với quản trị viên, hệ thống hỗ trợ quản lý tài khoản người dùng, dịch vụ nha khoa, thuốc, thống kê doanh thu và phân tích dữ liệu hoạt động của phòng khám.

Hệ thống được xây dựng theo mô hình frontend – backend tách biệt, sử dụng VueJS, Spring Boot và PostgreSQL nhằm đảm bảo khả năng mở rộng, bảo trì và nâng cao hiệu quả quản lý.

2.2. Xác định Actor

Bảng 2.1 Xác định Actor

Actor	Chức năng
Bệnh nhân	Đăng ký, đăng nhập, quên mật khẩu
	Cập nhật thông tin cá nhân và đổi mật khẩu
	Xem danh sách dịch vụ nha khoa
	Đặt lịch khám trực tuyến
	Theo dõi trạng thái lịch hẹn
	Xem lịch sử khám và lịch sử lịch hẹn
	Xem kế hoạch điều trị
	Xem đơn thuốc được bác sĩ kê
	Xem hóa đơn và trạng thái thanh toán
	Thanh toán hóa đơn bằng VNPAY hoặc tiền mặt
	Nhận thông báo, email xác nhận lịch hẹn và nhắc lịch khám
	Sử dụng AI để được gợi ý dịch vụ hoặc bác sĩ phù hợp
Bác sĩ	Đăng nhập hệ thống
	Xem danh sách lịch hẹn được phân công
	Xem thông tin bệnh nhân
	Ghi nhận kết quả khám
	Lập và cập nhật kế hoạch điều trị
	Kê đơn thuốc cho bệnh nhân
	Tạo lịch tái khám
	Theo dõi tiến trình điều trị của bệnh nhân

Nhân viên	Đăng nhập hệ thống
	Xem danh sách lịch hẹn
	Hỗ trợ xác nhận hoặc cập nhật lịch hẹn
	Quản lý hóa đơn
	Ghi nhận thanh toán tiền mặt
	Kiểm tra trạng thái thanh toán VNPay
	In hoặc gửi thông tin hóa đơn cho bệnh nhân
Admin	Đăng nhập hệ thống quản trị
	Quản lý người dùng và phân quyền tài khoản
	Quản lý bệnh nhân
	Quản lý bác sĩ và nhân viên
	Quản lý dịch vụ nha khoa
	Quản lý lịch hẹn và phân công bác sĩ
	Quản lý lịch làm việc, ca khám và sức chứa bác sĩ
	Quản lý kế hoạch điều trị
	Quản lý thuốc, lô thuốc và tồn kho
	Quản lý hóa đơn và thanh toán
	Quản lý thông báo, email
	Xem thống kê doanh thu, chi phí, lợi nhuận
	Sử dụng AI phân tích thống kê và đề xuất hướng phát triển

2.2.1. Bệnh nhân

Bệnh nhân là người sử dụng các dịch vụ khám và điều trị tại phòng khám nha khoa. Hệ thống website hỗ trợ bệnh nhân thực hiện các thao tác trực tuyến thay cho phương thức đặt lịch và quản lý thủ công truyền thống.

Bệnh nhân có thể đăng ký tài khoản, đăng nhập, cập nhật thông tin cá nhân và đặt lịch khám trực tuyến theo ngày khám và ca khám phù hợp. Ngoài ra, hệ thống còn hỗ trợ xem danh sách dịch vụ nha khoa, theo dõi trạng thái lịch hẹn và xem lịch sử khám bệnh.

Trong quá trình điều trị, bệnh nhân có thể xem kế hoạch điều trị, đơn thuốc, lịch tái khám và thực hiện thanh toán hóa đơn bằng tiền mặt hoặc VNPay. Hệ thống cũng hỗ trợ gửi email và thông báo nhắc lịch khám nhằm nâng cao trải nghiệm người dùng.

Ngoài ra, hệ thống định hướng tích hợp AI để hỗ trợ gợi ý dịch vụ hoặc bác sĩ phù hợp với nhu cầu của bệnh nhân.

2.2.2. Bác sĩ

Bác sĩ là người trực tiếp thực hiện quá trình khám và điều trị cho bệnh nhân tại phòng khám nha khoa. Trong hệ thống, bác sĩ có tài khoản riêng để theo dõi lịch hẹn, xem thông tin bệnh nhân và cập nhật kết quả điều trị.

Bác sĩ có thể xem lịch khám được phân công, theo dõi thông tin bệnh nhân, ghi nhận kết quả khám và lập kế hoạch điều trị phù hợp. Ngoài ra, hệ thống hỗ trợ bác sĩ kê đơn thuốc, cập nhật tiến độ điều trị và tạo lịch tái khám cho bệnh nhân khi cần thiết.

Hệ thống giúp bác sĩ quản lý thông tin điều trị hiệu quả hơn, đồng thời hỗ trợ theo dõi lịch sử khám bệnh nhằm nâng cao chất lượng chăm sóc bệnh nhân.

2.2.3. Nhân viên

Nhân viên là người hỗ trợ quá trình tiếp nhận bệnh nhân, xử lý lịch hẹn và thanh toán tại phòng khám nha khoa. Nhân viên đóng vai trò kết nối giữa bệnh nhân, bác sĩ và hệ thống quản lý trong quá trình vận hành phòng khám.

Trong hệ thống, nhân viên có thể xem và quản lý lịch hẹn, hỗ trợ xác nhận hoặc cập nhật trạng thái lịch khám cho bệnh nhân. Ngoài ra, nhân viên còn hỗ trợ quản lý hóa đơn, ghi nhận thanh toán bằng tiền mặt và kiểm tra trạng thái thanh toán qua VNPay.

Hệ thống cũng hỗ trợ nhân viên gửi thông báo hoặc email liên quan đến lịch hẹn và thanh toán nhằm nâng cao hiệu quả chăm sóc bệnh nhân và quản lý phòng khám.

2.2.4. Admin

Admin là người quản trị toàn bộ hệ thống phòng khám nha khoa, có nhiệm vụ quản lý dữ liệu và theo dõi hoạt động của hệ thống nhằm đảm bảo quá trình vận hành diễn ra ổn định và hiệu quả.

Trong hệ thống, admin có thể quản lý tài khoản người dùng, phân quyền theo vai trò, quản lý thông tin bác sĩ, nhân viên, bệnh nhân và các dịch vụ nha khoa. Ngoài ra, admin còn hỗ trợ quản lý lịch hẹn, lịch làm việc, kế hoạch điều trị, thuốc, hóa đơn và thanh toán.

Hệ thống cũng hỗ trợ admin theo dõi thống kê doanh thu, số lượng bệnh nhân, lịch hẹn và tình hình hoạt động của phòng khám. Bên cạnh đó, AI được tích hợp nhằm hỗ trợ phân tích dữ liệu và đề xuất thông tin phù hợp cho quá trình quản lý hệ thống.

2.3. Xác định Usecase

Bảng 2.2 Xác định Usecase

Actor	Usecase
Bệnh nhân	UC01: Xem trang chủ
	UC02: Xem danh sách dịch vụ nha khoa
	UC03: Tìm kiếm dịch vụ nha khoa
	UC04: Đăng ký tài khoản
	UC05: Đăng nhập
	UC06: Quên mật khẩu
	UC07: Xem và cập nhật thông tin cá nhân
	UC08: Đổi mật khẩu
	UC09: Đặt lịch khám trực tuyến
	UC10: Theo dõi trạng thái lịch hẹn
	UC11: Hủy lịch hẹn
	UC12: Xem lịch sử khám và lịch sử lịch hẹn

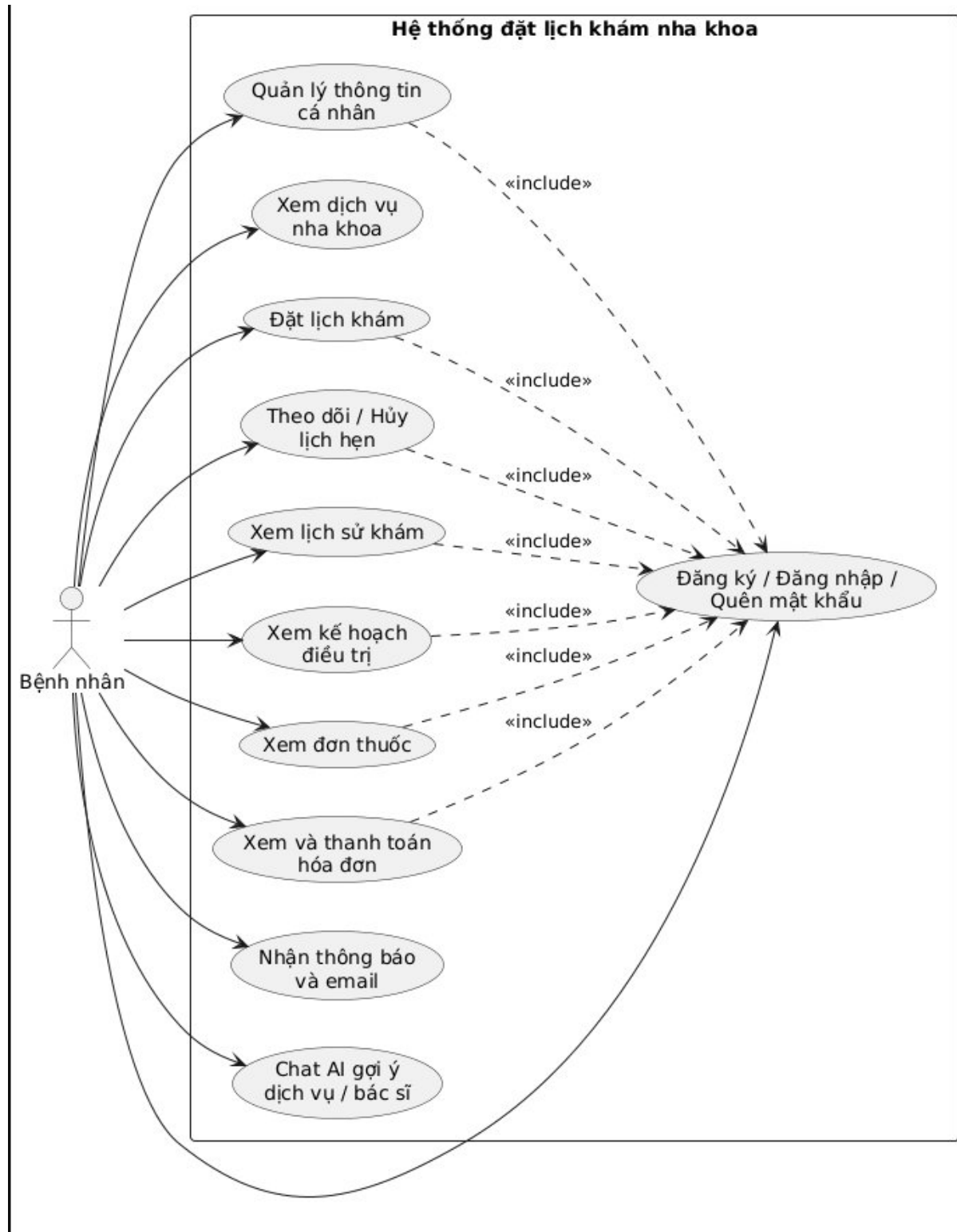
	UC13: Xem kế hoạch điều trị
	UC14: Xem đơn thuốc
	UC15: Xem hóa đơn
	UC16: Thanh toán hóa đơn bằng VNPay hoặc tiền mặt
	UC17: Nhận thông báo và email nhắc lịch khám
	UC18: Chat với AI để được gợi ý dịch vụ hoặc bác sĩ phù hợp
Bác sĩ	UC19: Đăng nhập
	UC20: Xem lịch hẹn được phân công
	UC21: Xem thông tin bệnh nhân
	UC22: Ghi nhận kết quả khám
	UC23: Lập kế hoạch điều trị
	UC24: Cập nhật tiến độ điều trị
	UC25: Kê đơn thuốc cho bệnh nhân
	UC26: Tạo lịch tái khám
Nhân viên	UC27: Đăng nhập
	UC28: Xem danh sách lịch hẹn
	UC29: Hỗ trợ xác nhận hoặc cập nhật lịch hẹn
	UC30: Quản lý hóa đơn
	UC31: Ghi nhận thanh toán tiền mặt
	UC32: Kiểm tra trạng thái thanh toán VNPay
	UC33: In hoặc gửi hóa đơn cho bệnh nhân
Admin	UC34: Đăng nhập

	UC35: Quản lý người dùng và phân quyền
	UC36: Quản lý bệnh nhân
	UC37: Quản lý bác sĩ và nhân viên
	UC38: Quản lý dịch vụ nha khoa
	UC39: Quản lý lịch hẹn
	UC40: Phân công bác sĩ cho lịch hẹn
	UC41: Quản lý lịch làm việc và ca khám
	UC42: Quản lý kế hoạch điều trị
	UC43: Quản lý thuốc, lô thuốc và tồn kho
	UC44: Quản lý đơn thuốc
	UC45: Quản lý hóa đơn và thanh toán
	UC46: Quản lý thông báo và email
	UC47: Xem thống kê doanh thu, chi phí, lợi nhuận
	UC48: Sử dụng AI phân tích thống kê và đề xuất hướng phát triển

2.4. Phân tích Usecase

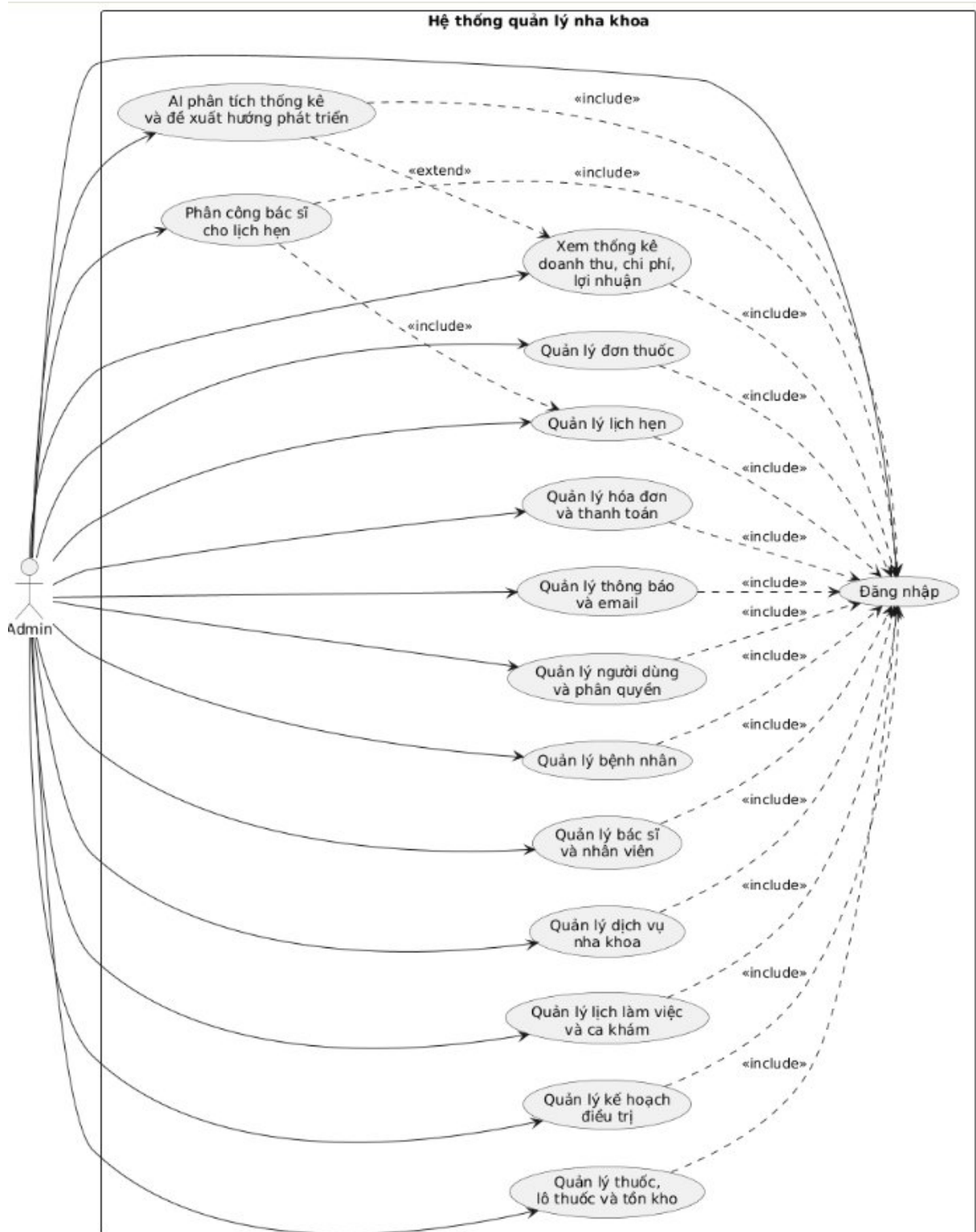
2.4.1. Biểu đồ Usecase

2.4.1.1. Usecase khách hàng



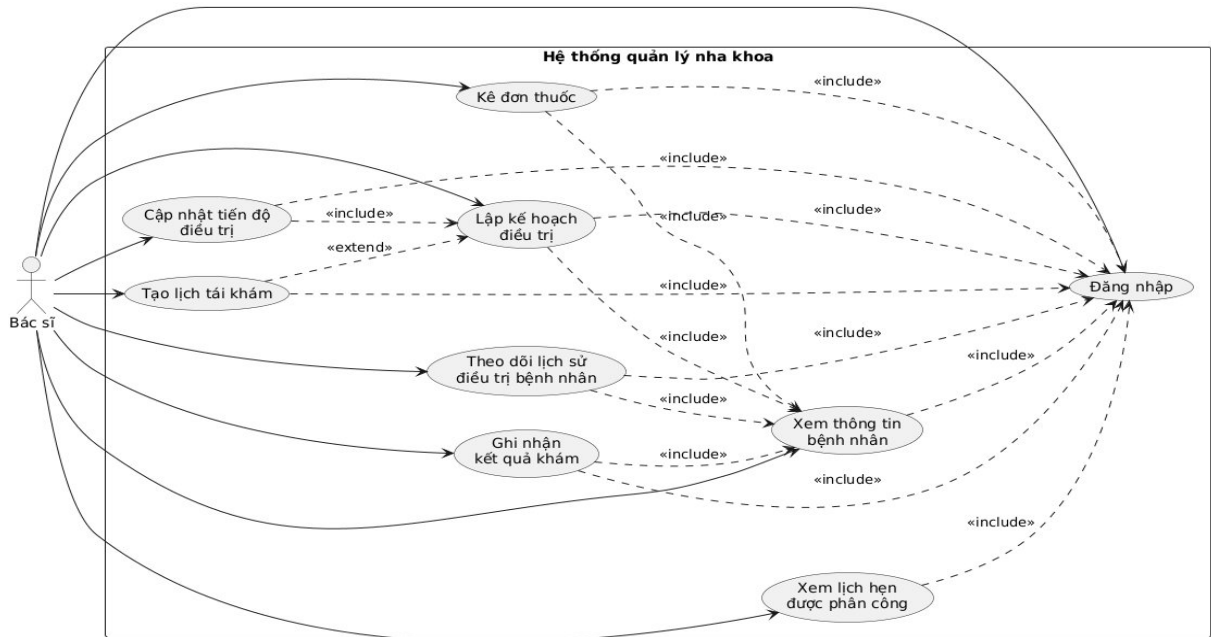
Hình 2.1: Sơ đồ Usecase khách hàng

2.4.1.2. Usecase Admin



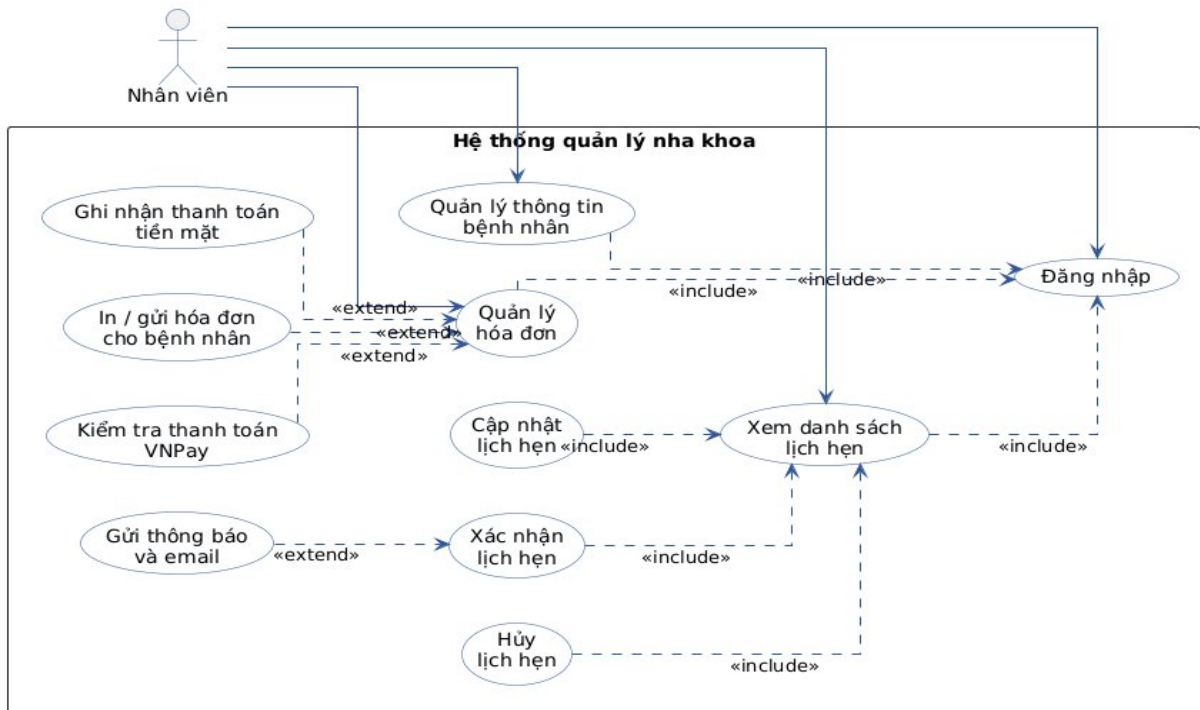
Hình 2.2: Sơ đồ Usecase Admin

2.4.1.3. Usecase Bác sĩ



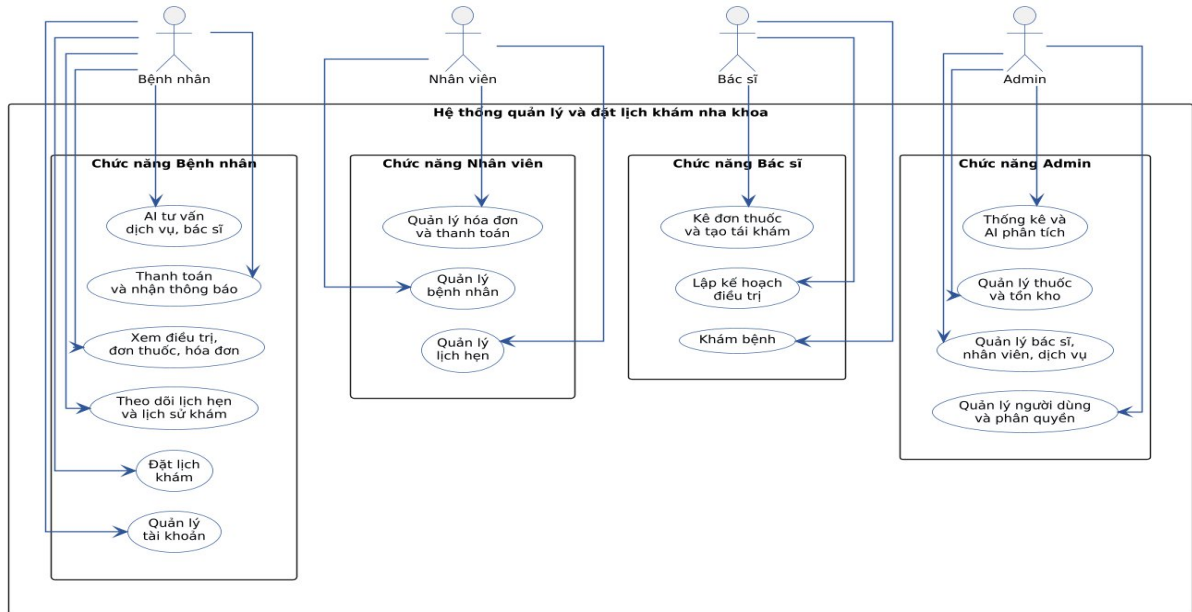
Hình 2.3: Sơ đồ Usecase Bác sĩ

2.4.1.4. Usecase Nhân viên



Hình 2.4: Sơ đồ Usecase Nhân viên

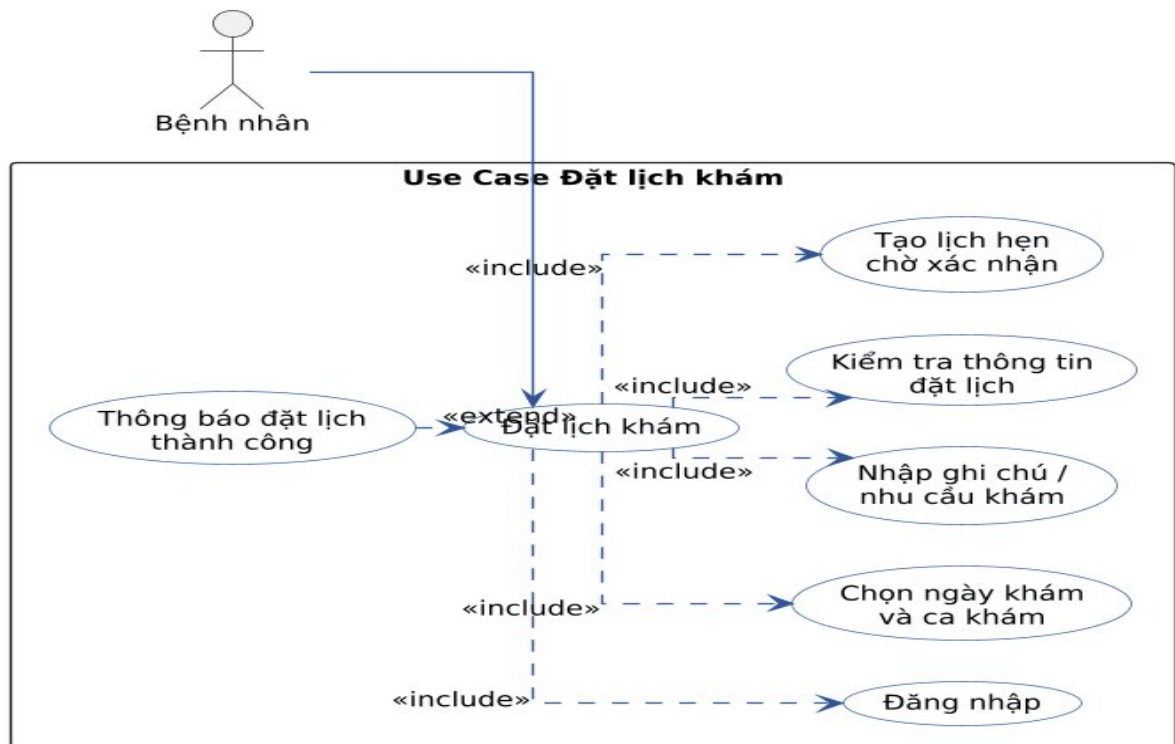
2.4.1.5. Usecase tổng quát



Hình 2.5: Sơ đồ Usecase tổng quát

2.4.2. Đặc tả usecase

2.4.2.1. Usecase đặt lịch khám



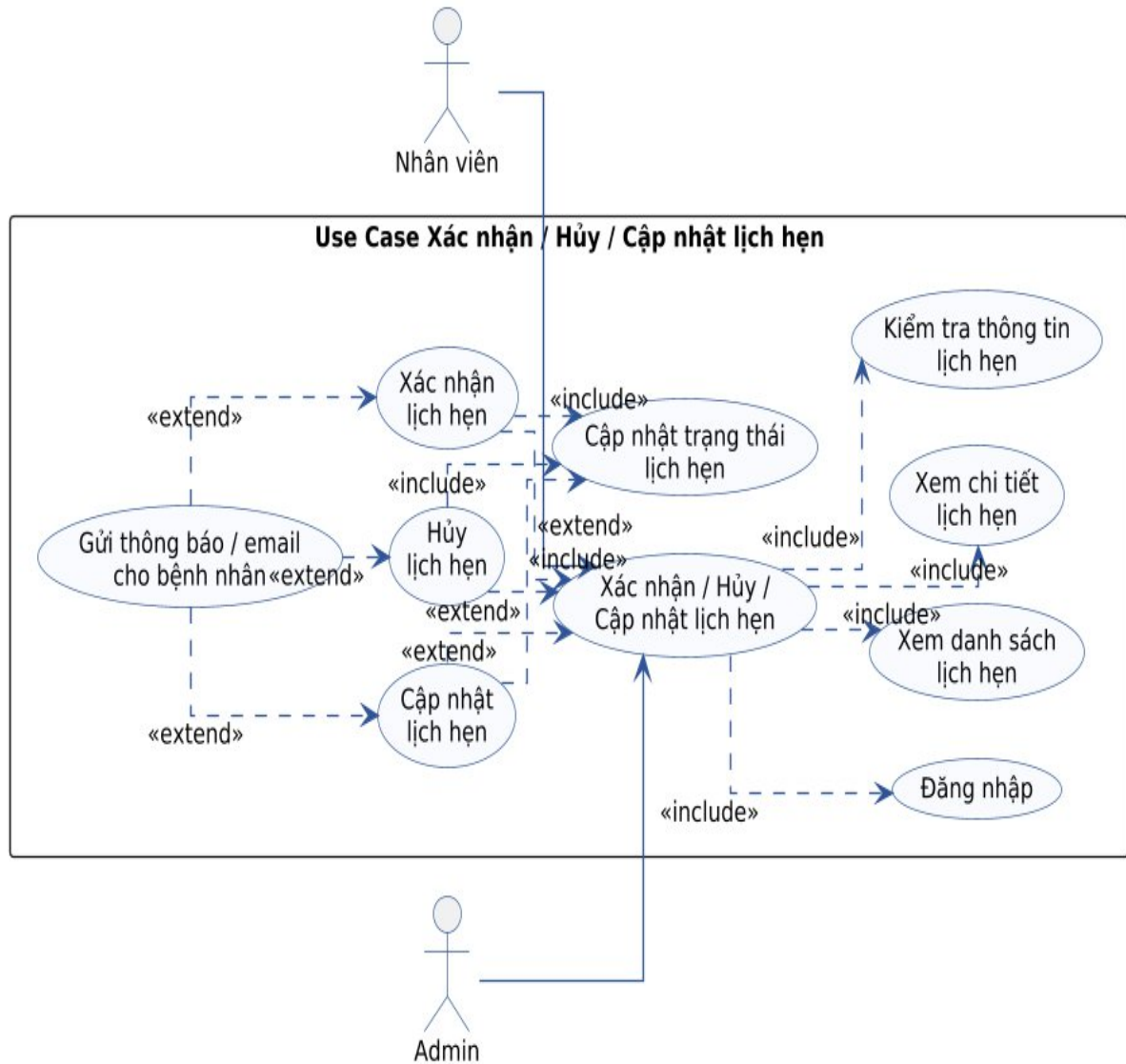
Hình 2.6: Đặc tả Usecase đặt lịch khám

Đặc tả Usecase đặt lịch khám

Bảng 2.3 Đặc tả usecase đặt lịch khám

TT	Nội dung	Mô tả
1	Tên Usecase	Đặt lịch khám
2	Tác nhân	Bệnh nhân
3	Mô tả	Bệnh nhân thực hiện đặt lịch khám nha khoa trực tuyến thông qua website.
4	Điều kiện trước	Bệnh nhân đã đăng nhập vào hệ thống.
5	Luồng sự kiện chính	1. Bệnh nhân truy cập chức năng đặt lịch khám. 2. Hệ thống hiển thị form đặt lịch. 3. Bệnh nhân chọn ngày khám, ca khám và nhập ghi chú hoặc nhu cầu khám nếu có. 4. Bệnh nhân nhấn nút gửi yêu cầu đặt lịch. 5. Hệ thống kiểm tra dữ liệu đặt lịch. 6. Hệ thống tạo lịch hẹn với trạng thái chờ xác nhận. 7. Hệ thống thông báo đặt lịch thành công cho bệnh nhân.
6	Luồng sự kiện phụ	Nếu bệnh nhân nhập thiếu thông tin hoặc ngày khám không hợp lệ, hệ thống hiển thị thông báo lỗi và yêu cầu nhập lại.
7	Điều kiện sau	Lịch hẹn được tạo thành công và chờ nhân viên hoặc admin xác nhận.
8	Yêu cầu đặc biệt	Ngày khám và ca khám phải hợp lệ, không được đặt lịch trong thời gian đã qua.

2.4.2.2. Usecase xác nhận/ chỉnh sửa/ hủy lịch hẹn



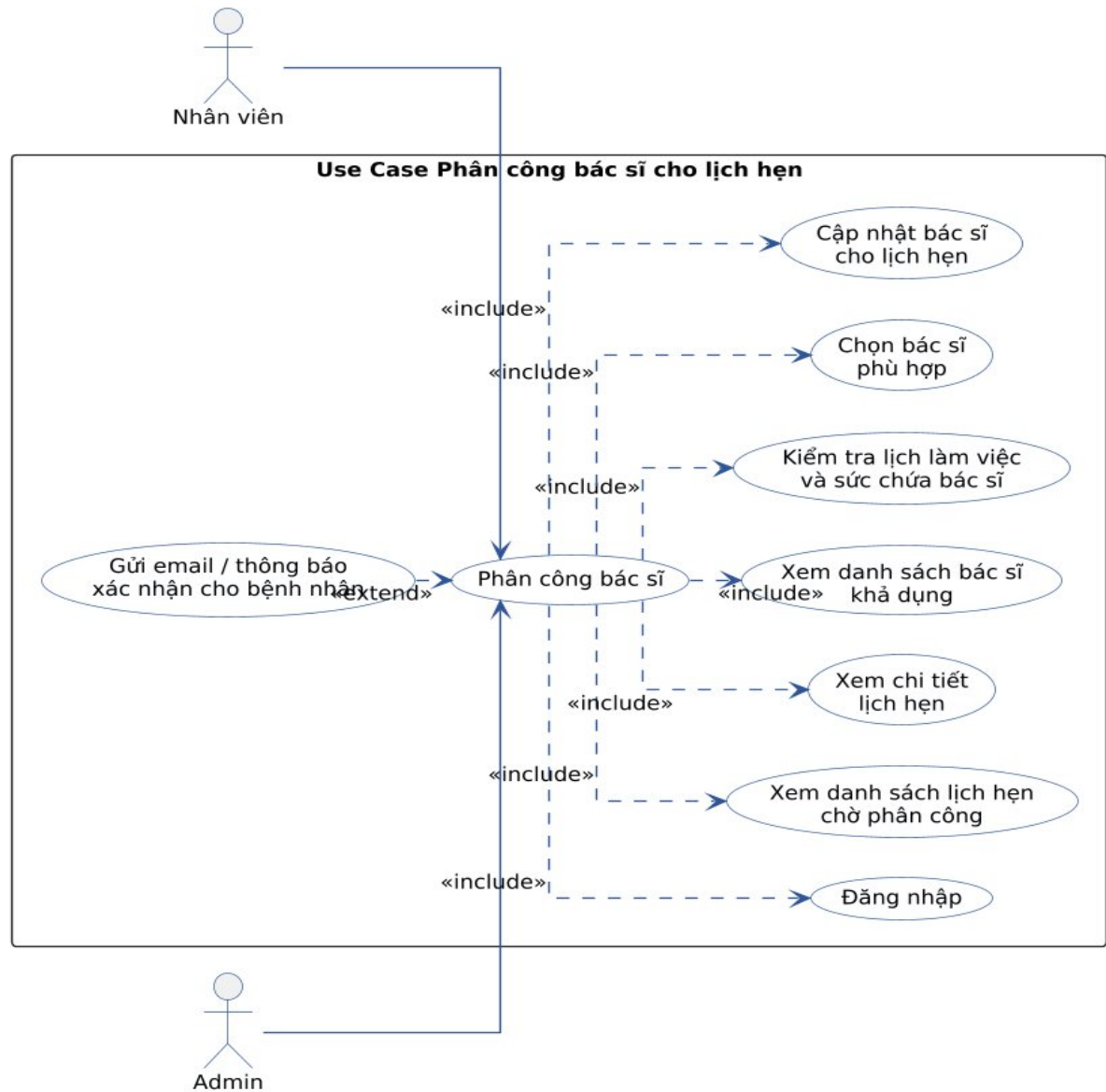
Hình 2.7: Đặc tả Usecase xác nhận/ Hủy / Cập nhật lịch hẹn

Đặc tả Usecase đăng nhập

Bảng 2.4 Đặc tả Usecase xác nhận hủy cập nhật lịch hẹn

TT	Nội dung	Mô tả
1	Tên Usecase	Xác nhận / hủy / cập nhật lịch hẹn
2	Tác nhân	Nhân viên, Admin
3	Mô tả	Nhân viên hoặc Admin xử lý các lịch hẹn do bệnh nhân đặt, bao gồm xác nhận, cập nhật hoặc hủy lịch hẹn.
4	Điều kiện trước	Nhân viên hoặc Admin đã đăng nhập vào hệ thống và lịch hẹn đã tồn tại.
5	Luồng sự kiện chính	1. Nhân viên hoặc Admin truy cập danh sách lịch hẹn. 2. Hệ thống hiển thị các lịch hẹn theo ngày, ca khám hoặc trạng thái. 3. Người dùng chọn một lịch hẹn cần xử lý. 4. Người dùng xem thông tin chi tiết lịch hẹn. 5. Người dùng thực hiện xác nhận, cập nhật thông tin hoặc hủy lịch hẹn. 6. Hệ thống kiểm tra dữ liệu xử lý. 7. Hệ thống cập nhật trạng thái lịch hẹn. 8. Hệ thống gửi thông báo hoặc email đến bệnh nhân nếu cần.
6	Luồng sự kiện phụ	Nếu lịch hẹn không tồn tại hoặc đã bị hủy trước đó, hệ thống hiển thị thông báo lỗi.
7	Điều kiện sau	Trạng thái lịch hẹn được cập nhật trong hệ thống.
8	Yêu cầu đặc biệt	Việc cập nhật lịch hẹn phải đảm bảo không trùng lịch và không vượt quá sức chứa của ca khám.

2.4.2.3. Usecase Phân công bác sĩ cho lịch hẹn



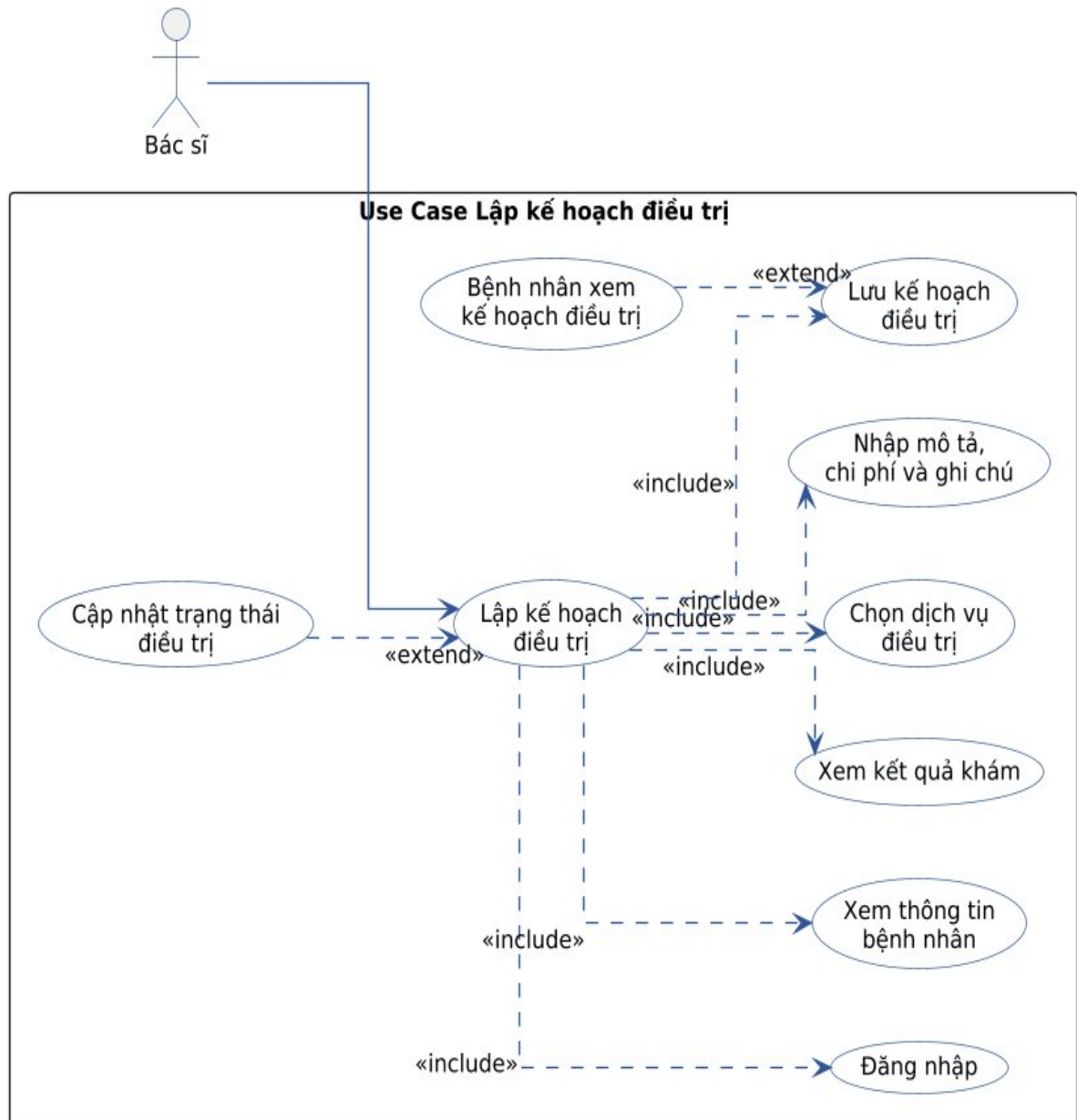
Hình 2.8: Đặc tả Usecase Phân công bác sĩ cho lịch hẹn

Đặc tả Usecase Phân công bác sĩ cho lịch hẹn

Bảng 2.5 Đặc tả usecase Phân công bác sĩ cho lịch hẹn

TT	Nội dung	Mô tả
1	Tên Usecase	Phân công bác sĩ cho lịch hẹn
2	Tác nhân	Admin, Nhân viên
3	Mô tả	Admin hoặc Nhân viên phân công bác sĩ phù hợp cho lịch hẹn của bệnh nhân.
4	Điều kiện trước	Lịch hẹn đã được tạo và bác sĩ có lịch làm việc phù hợp với ngày, ca khám.
5	Luồng sự kiện chính	1. Admin hoặc Nhân viên truy cập danh sách lịch hẹn chờ phân công. 2. Hệ thống hiển thị thông tin lịch hẹn của bệnh nhân. 3. Người dùng chọn chức năng phân công bác sĩ. 4. Hệ thống hiển thị danh sách bác sĩ khả dụng theo ngày và ca khám. 5. Người dùng chọn bác sĩ phù hợp. 6. Hệ thống kiểm tra sức chứa và lịch làm việc của bác sĩ. 7. Hệ thống lưu thông tin bác sĩ được phân công vào lịch hẹn. 8. Hệ thống cập nhật trạng thái lịch hẹn và gửi email xác nhận đến bệnh nhân.
6	Luồng sự kiện phụ	Nếu bác sĩ không còn khả dụng hoặc ca khám đã đầy, hệ thống hiển thị thông báo và yêu cầu chọn bác sĩ khác.
7	Điều kiện sau	Lịch hẹn được gán bác sĩ và bệnh nhân nhận được thông báo xác nhận.
8	Yêu cầu đặc biệt	Một bác sĩ không được vượt quá số lượng bệnh nhân tối đa trong cùng một ca khám.

2.4.2.4. Usecase Lập kế hoạch điều trị



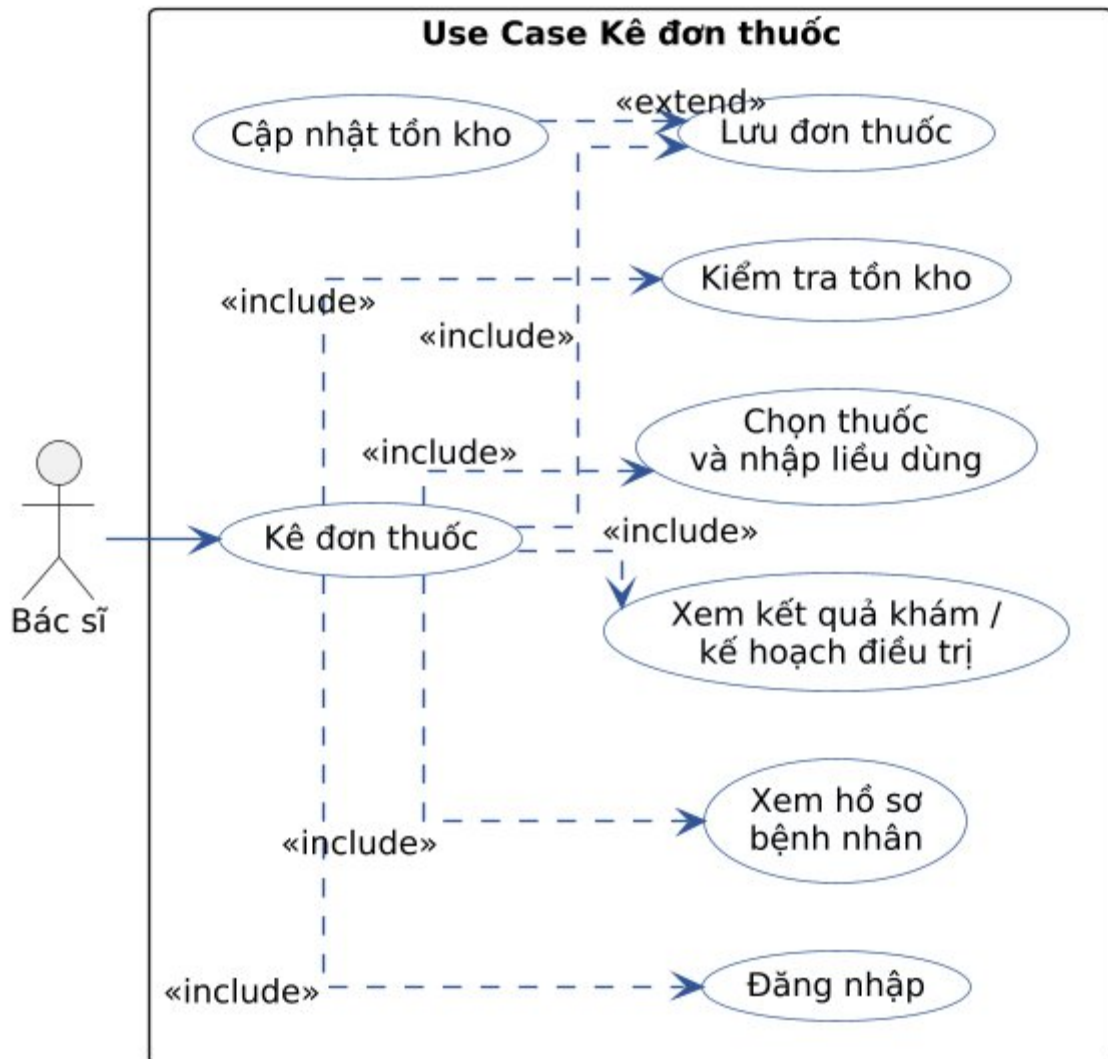
Hình 2.9: Đặc tả Usecase Lập kế hoạch điều trị

Đặc tả Usecase Lập kế hoạch điều trị

Bảng 2.6 Đặc tả Usecase Lập kế hoạch điều trị

TT	Nội dung	Mô tả
1	Tên Usecase	Lập kế hoạch điều trị
2	Tác nhân	Bác sĩ
3	Mô tả	Bác sĩ lập kế hoạch điều trị cho bệnh nhân dựa trên kết quả khám và tình trạng răng miệng.
4	Điều kiện trước	Bệnh nhân đã có kết quả khám hoặc hồ sơ điều trị trong hệ thống.
5	Luồng sự kiện chính	1. Bác sĩ mở hồ sơ bệnh nhân. 2. Hệ thống hiển thị thông tin bệnh nhân và kết quả khám. 3. Bác sĩ chọn chức năng lập kế hoạch điều trị. 4. Bác sĩ nhập thông tin kế hoạch gồm dịch vụ điều trị, mô tả, thời gian dự kiến, chi phí và ghi chú. 5. Bác sĩ lưu kế hoạch điều trị. 6. Hệ thống lưu kế hoạch điều trị vào hồ sơ bệnh nhân. 7. Bệnh nhân có thể xem kế hoạch điều trị trong tài khoản cá nhân.
6	Luồng sự kiện phụ	Nếu thông tin kế hoạch điều trị không hợp lệ, hệ thống hiển thị thông báo lỗi và yêu cầu nhập lại.
7	Điều kiện sau	Kế hoạch điều trị được tạo và gắn với hồ sơ bệnh nhân.
8	Yêu cầu đặc biệt	Chi phí điều trị phải được tính toán rõ ràng và trạng thái điều trị cần được cập nhật theo tiến độ thực tế.

2.4.2.5. Usecase Kế đơn thuốc



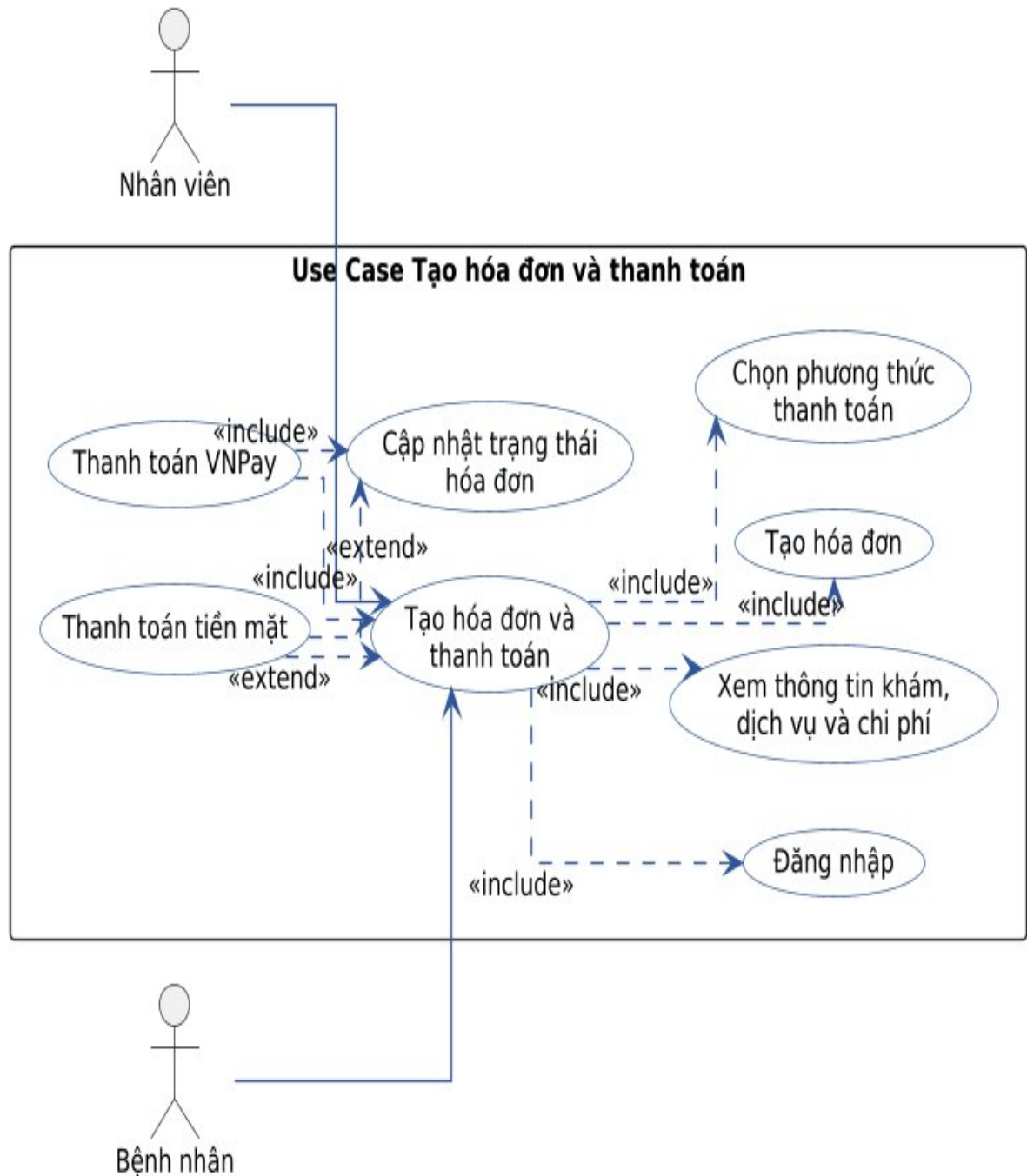
Hình 2.10: Đặc tả Usecase Kế đơn thuốc

Đặc tả Usecase Kê đơn thuốc

Bảng 2.7 Đặc tả Usecase Kê đơn thuốc

TT	Nội dung	Mô tả
1	Tên Usecase	Kê đơn thuốc
2	Tác nhân	Bác sĩ
3	Mô tả	Bác sĩ kê đơn thuốc cho bệnh nhân sau quá trình khám hoặc điều trị.
4	Điều kiện trước	Bệnh nhân đã được khám và có nhu cầu sử dụng thuốc theo chỉ định của bác sĩ.
5	Luồng sự kiện chính	1. Bác sĩ truy cập hồ sơ khám của bệnh nhân. 2. Bác sĩ chọn chức năng kê đơn thuốc. 3. Hệ thống hiển thị danh sách thuốc hiện có. 4. Bác sĩ chọn thuốc, nhập liều lượng, số lượng, cách dùng và ghi chú. 5. Hệ thống kiểm tra thông tin thuốc và số lượng tồn kho. 6. Bác sĩ xác nhận lưu đơn thuốc. 7. Hệ thống lưu đơn thuốc và cập nhật dữ liệu liên quan. 8. Bệnh nhân có thể xem đơn thuốc trong tài khoản cá nhân.
6	Luồng sự kiện phụ	Nếu thuốc không đủ số lượng tồn kho hoặc thông tin kê đơn chưa hợp lệ, hệ thống hiển thị thông báo lỗi.
7	Điều kiện sau	Đơn thuốc được lưu thành công và gắn với hồ sơ khám của bệnh nhân.
8	Yêu cầu đặc biệt	Việc kê đơn phải đảm bảo đúng thuốc, đúng liều lượng và phù hợp với tình trạng bệnh nhân.

2.4.2.6. Usecase Tạo hóa đơn và thanh toán



Hình 2.11: Đặc tả Usecase Tạo hóa đơn và thanh toán

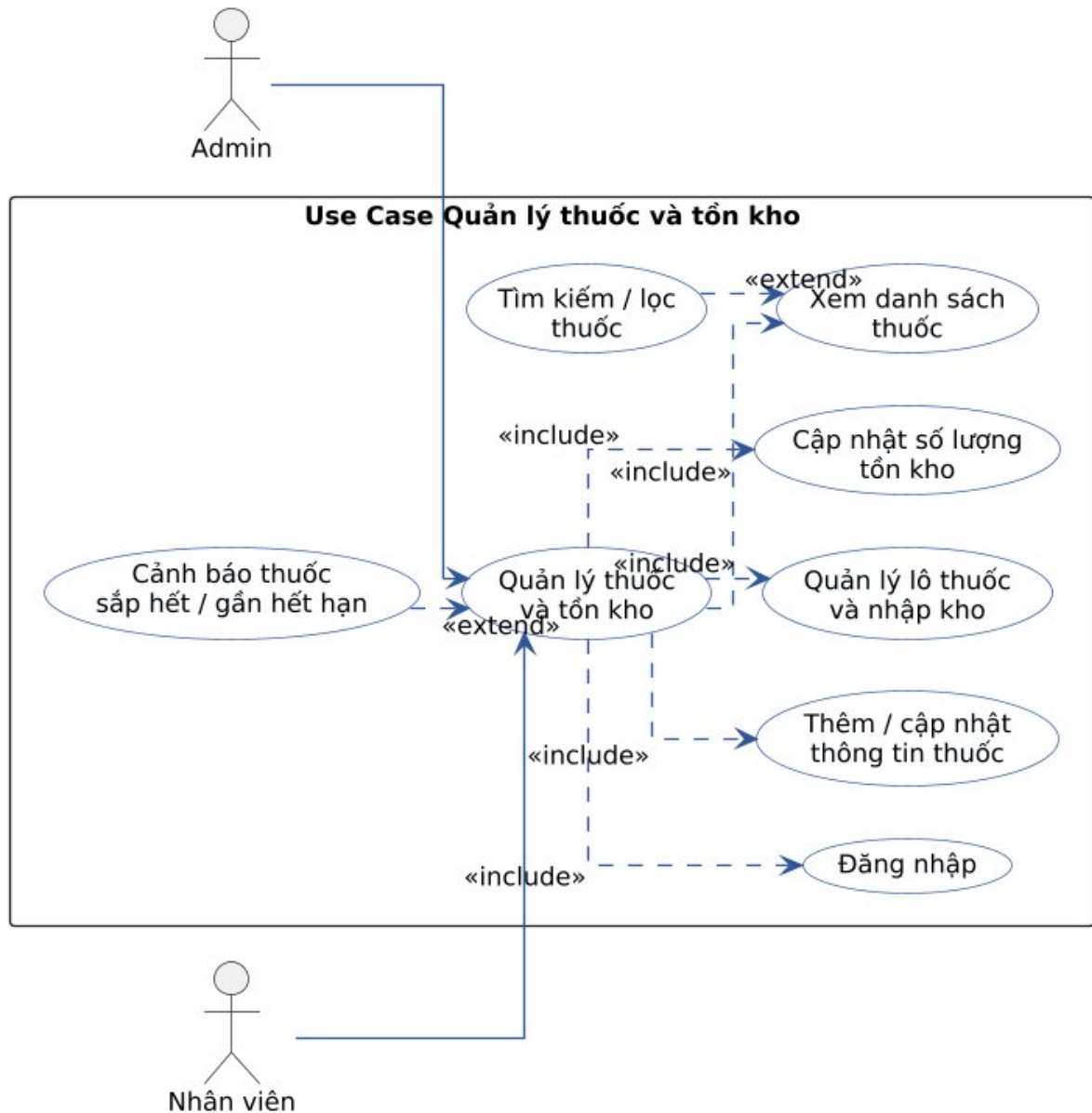
Đặc tả Usecase Tạo hóa đơn và thanh toán

Bảng 2.8 Đặc tả Usecase Tạo hóa đơn và thanh toán

TT	Nội dung	Mô tả
1	Tên Usecase	Tạo hóa đơn và thanh toán

2	Tác nhân	Nhân viên, Bệnh nhân
3	Mô tả	Nhân viên tạo hóa đơn cho bệnh nhân dựa trên dịch vụ điều trị, thuốc hoặc các chi phí phát sinh. Bệnh nhân có thể thanh toán bằng tiền mặt hoặc VNPay.
4	Điều kiện trước	Bệnh nhân đã sử dụng dịch vụ khám hoặc điều trị và có thông tin chi phí trong hệ thống.
5	Luồng sự kiện chính	1. Nhân viên truy cập chức năng quản lý hóa đơn. 2. Nhân viên chọn bệnh nhân hoặc lịch hẹn cần tạo hóa đơn. 3. Hệ thống hiển thị các dịch vụ, thuốc và chi phí liên quan. 4. Nhân viên kiểm tra thông tin và tạo hóa đơn. 5. Hệ thống tính tổng tiền cần thanh toán. 6. Bệnh nhân chọn phương thức thanh toán bằng tiền mặt hoặc VNPay. 7. Nếu thanh toán tiền mặt, nhân viên ghi nhận thanh toán. 8. Nếu thanh toán VNPay, hệ thống tạo giao dịch thanh toán và cập nhật kết quả sau khi thanh toán thành công. 9. Hệ thống cập nhật trạng thái hóa đơn. 10. Hệ thống gửi thông báo hoặc email thanh toán thành công cho bệnh nhân.
6	Luồng sự kiện phụ	Nếu thanh toán VNPay thất bại, hệ thống giữ trạng thái hóa đơn là chưa thanh toán và thông báo cho bệnh nhân.
7	Điều kiện sau	Hóa đơn được tạo và trạng thái thanh toán được cập nhật.
8	Yêu cầu đặc biệt	Tổng tiền hóa đơn phải chính xác, trạng thái thanh toán phải được đồng bộ với kết quả giao dịch thực tế.

2.4.2.7. Usecase Quản lý thuốc và tồn kho



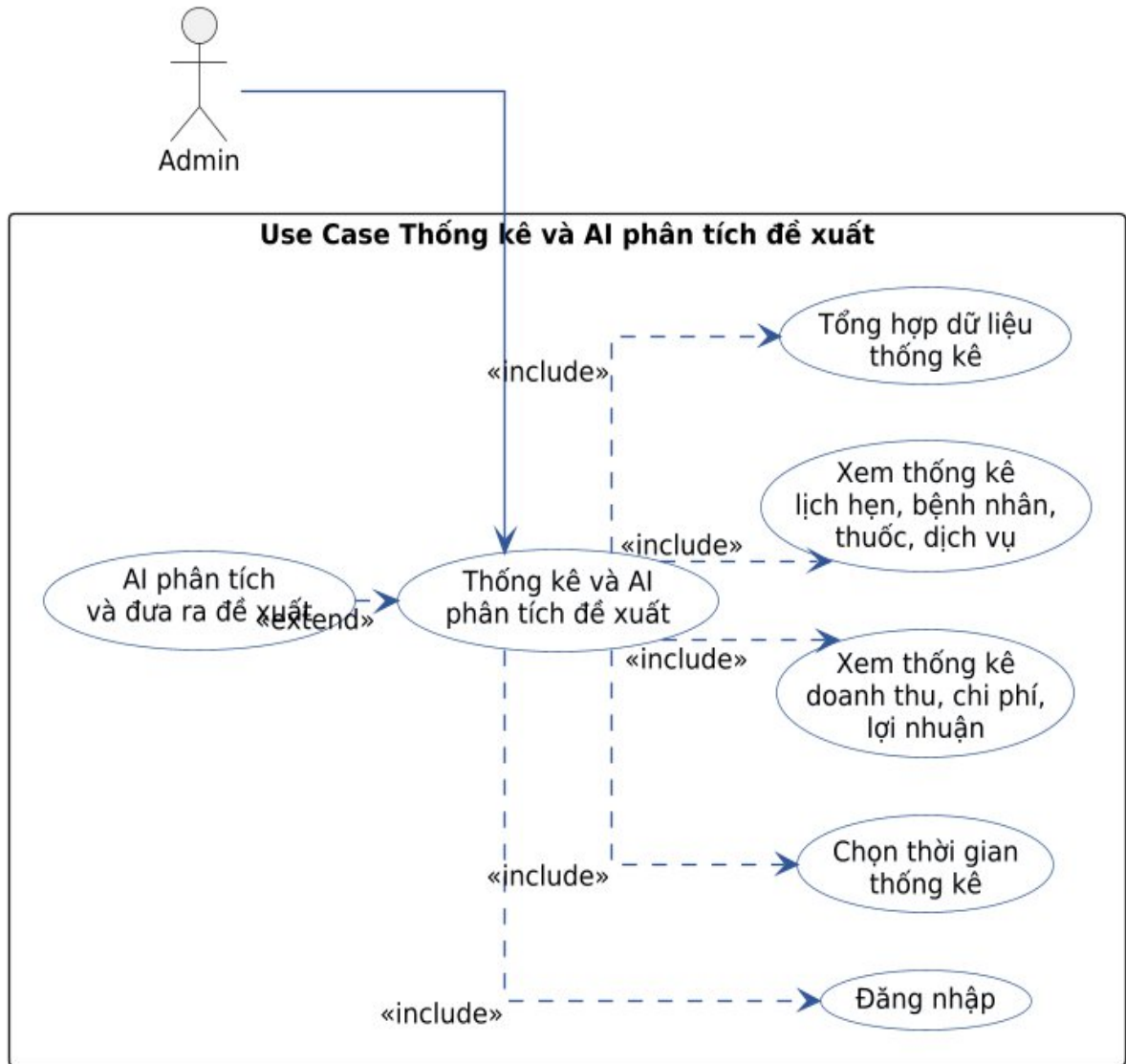
Hình 2.12: Đặc tả Usecase Quản lý thuốc và tồn kho

Đặc tả Usecase Quản lý thuốc và tồn kho

Bảng 2.9 Đặc tả Usecase Quản lý thuốc và tồn kho

TT	Nội dung	Mô tả
1	Tên Usecase	Quản lý thuốc và tồn kho
2	Tác nhân	Admin, Nhân viên
3	Mô tả	Admin hoặc Nhân viên quản lý danh sách thuốc, lô thuốc, số lượng tồn kho và tình trạng sử dụng thuốc trong phòng khám.
4	Điều kiện trước	Người dùng đã đăng nhập và có quyền quản lý thuốc.
5	Luồng sự kiện chính	1. Admin hoặc Nhân viên truy cập chức năng quản lý thuốc. 2. Hệ thống hiển thị danh sách thuốc hiện có. 3. Người dùng thêm mới, cập nhật hoặc tìm kiếm thuốc. 4. Người dùng nhập thông tin thuốc gồm tên thuốc, đơn vị tính, giá, số lượng, ngày nhập, hạn sử dụng và ghi chú. 5. Hệ thống kiểm tra dữ liệu thuốc. 6. Hệ thống lưu thông tin thuốc hoặc lô thuốc. 7. Hệ thống cập nhật số lượng tồn kho khi có nhập thuốc hoặc kê đơn thuốc.
6	Luồng sự kiện phụ	Nếu thuốc đã tồn tại hoặc dữ liệu không hợp lệ, hệ thống hiển thị thông báo lỗi.
7	Điều kiện sau	Thông tin thuốc và tồn kho được cập nhật chính xác trong hệ thống.
8	Yêu cầu đặc biệt	Hệ thống cần cảnh báo khi thuốc sắp hết hoặc gần hết hạn sử dụng.

2.4.2.8. Usecase Thống kê và AI phân tích đề xuất



Hình 2.13: Đặc tả Usecase Thống kê và AI phân tích đề xuất

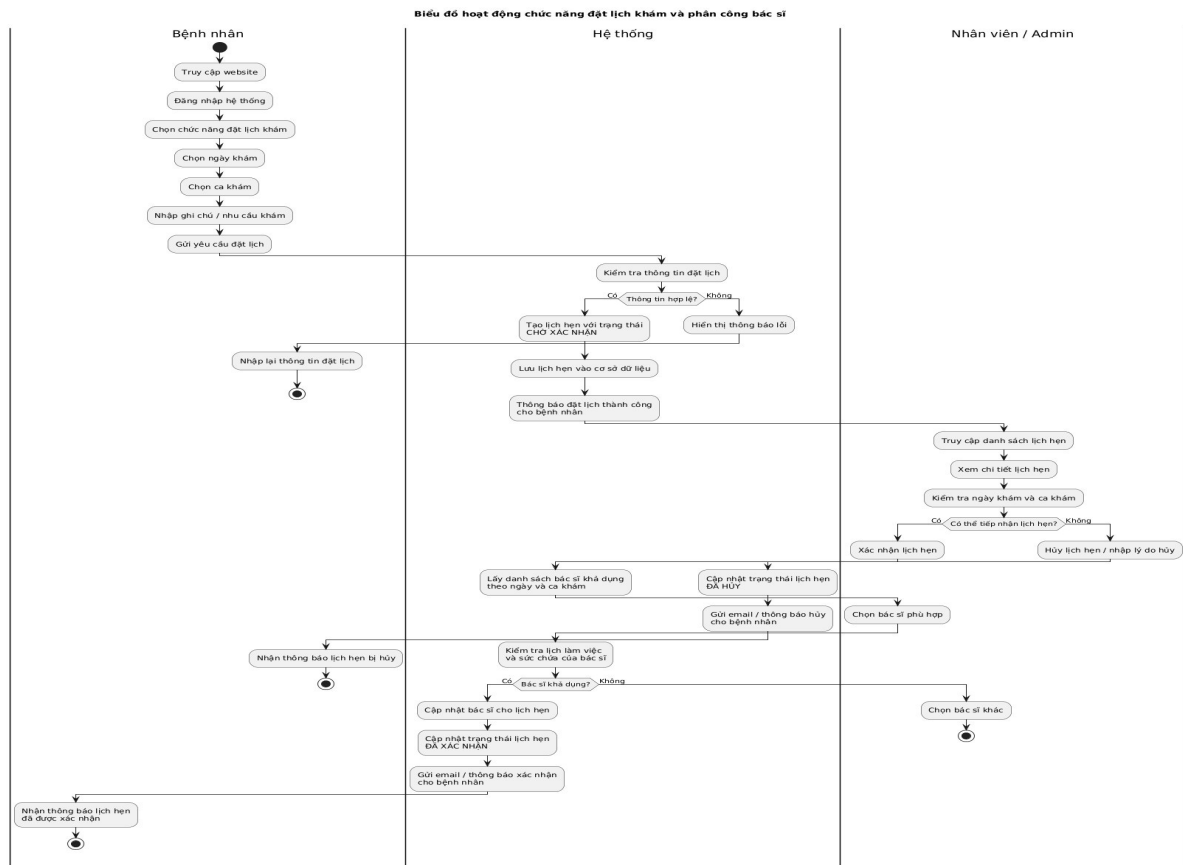
Đặc tả Usecase Thống kê và AI phân tích đề xuất

Bảng 2.10 Đặc tả Usecase Thống kê và AI phân tích đề xuất

TT	Nội dung	Mô tả
1	Tên Usecase	Thống kê và AI phân tích đề xuất
2	Tác nhân	Admin
3	Mô tả	Admin xem các dữ liệu thống kê của phòng khám và sử dụng AI để hỗ trợ phân tích, đề xuất dịch vụ, bác sĩ hoặc hướng phát triển hệ thống.
4	Điều kiện trước	Admin đã đăng nhập vào hệ thống và hệ thống có dữ liệu thống kê cần phân tích.
5	Luồng sự kiện chính	1. Admin truy cập trang thống kê. 2. Hệ thống hiển thị các thông tin như doanh thu, chi phí, lợi nhuận, số lượng lịch hẹn, bệnh nhân, thuốc và dịch vụ. 3. Admin chọn khoảng thời gian hoặc loại dữ liệu cần xem. 4. Hệ thống tổng hợp và hiển thị dữ liệu thống kê. 5. Admin chọn chức năng AI phân tích. 6. Hệ thống gửi dữ liệu thống kê cần thiết cho AI xử lý. 7. AI đưa ra nhận xét, cảnh báo hoặc đề xuất hướng phát triển. 8. Admin xem kết quả phân tích và sử dụng làm cơ sở tham khảo trong quản lý.
6	Luồng sự kiện phụ	Nếu dữ liệu thống kê không đủ hoặc AI không thể phân tích, hệ thống hiển thị thông báo phù hợp.
7	Điều kiện sau	Admin xem được thống kê và các đề xuất hỗ trợ ra quyết định.
8	Yêu cầu đặc biệt	Dữ liệu gửi cho AI cần được kiểm soát, tránh lộ thông tin nhạy cảm của bệnh nhân.

2.5. Biểu đồ hoạt động

2.5.1. Biểu đồ hoạt động đặt lịch khám và phân công bác sĩ



Hình 2.14: Biểu đồ hoạt động đặt lịch khám và phân công bác sĩ

Biểu đồ hoạt động mô tả quy trình bệnh nhân đặt lịch khám trên hệ thống quản lý nha khoa.

Quy trình bắt đầu khi bệnh nhân đăng nhập và chọn chức năng đặt lịch khám. Bệnh nhân nhập thông tin như ngày khám, ca khám và gửi yêu cầu đặt lịch đến hệ thống.

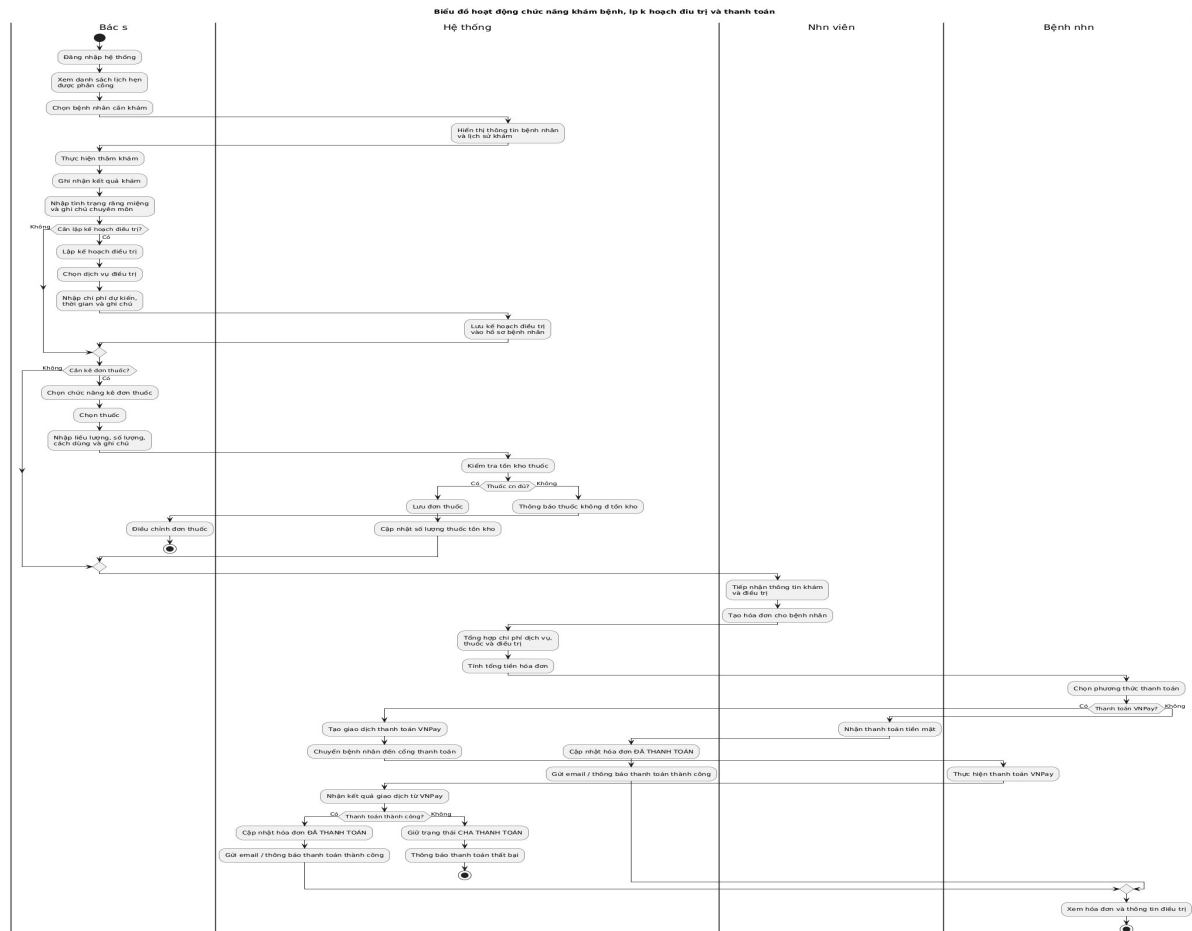
Hệ thống kiểm tra tính hợp lệ của dữ liệu. Nếu thông tin hợp lệ, hệ thống tạo lịch hẹn với trạng thái chờ xác nhận và lưu vào cơ sở dữ liệu.

Nhân viên hoặc admin tiếp nhận yêu cầu đặt lịch, kiểm tra thông tin và thực hiện xác nhận lịch hẹn. Sau đó, hệ thống hỗ trợ phân công bác sĩ phù hợp theo lịch làm việc và ca khám.

Nếu bác sĩ còn khả dụng, hệ thống cập nhật trạng thái lịch hẹn và gửi thông báo xác nhận đến bệnh nhân. Trong trường hợp không thể tiếp nhận lịch khám, hệ thống sẽ cập nhật trạng thái hủy và gửi thông báo cho bệnh nhân.

Cuối cùng, bệnh nhân có thể theo dõi trạng thái lịch hẹn trực tiếp trên hệ thống..

2.5.2. Biểu đồ hoạt động chức năng khám bệnh, lập kế hoạch điều trị và thanh toán



Hình 2.15: Biểu đồ hoạt động chức năng khám bệnh

Biểu đồ hoạt động mô tả quy trình khám bệnh, lập kế hoạch điều trị, kê đơn thuốc và thanh toán trong hệ thống quản lý nha khoa.

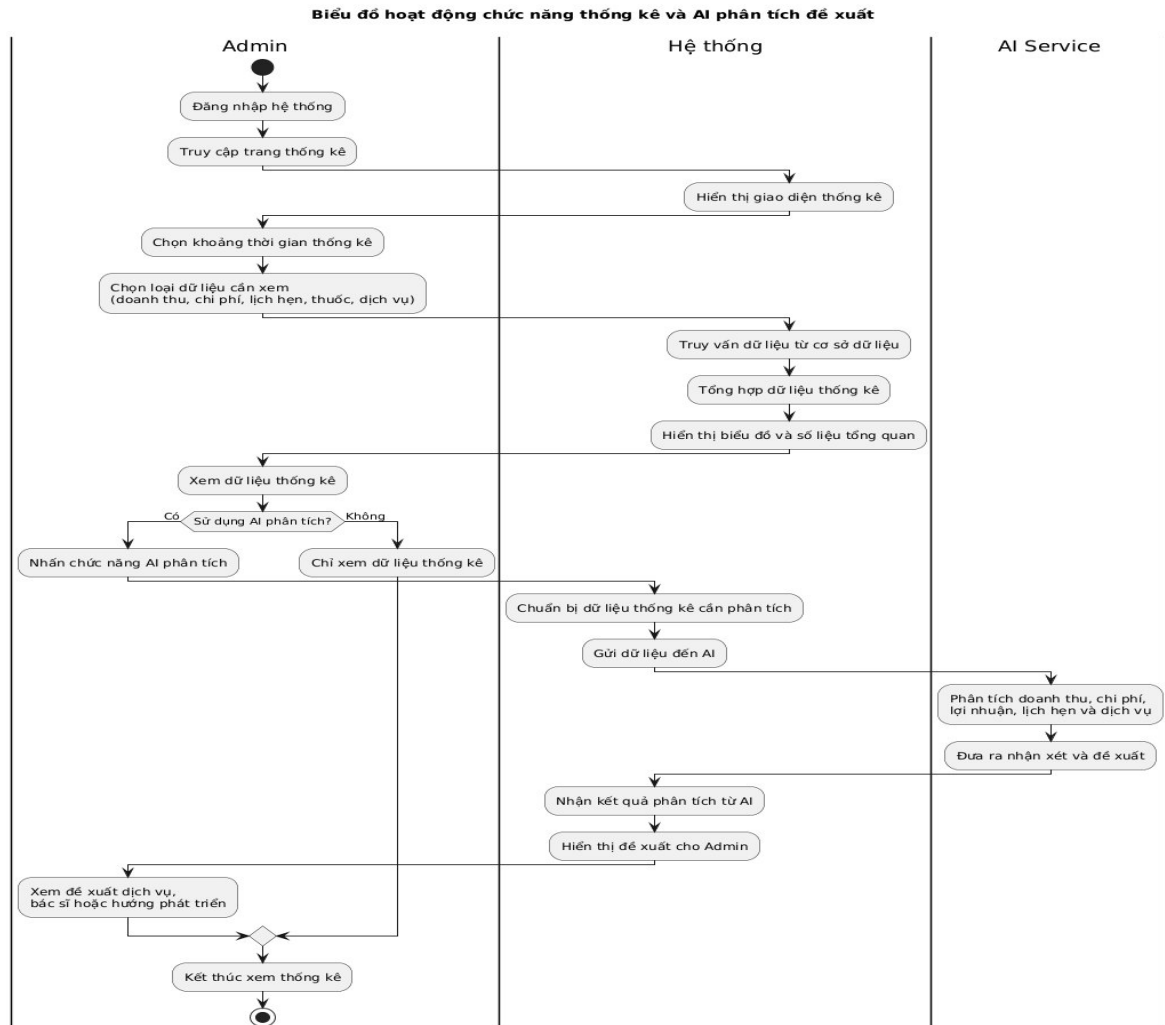
Quy trình bắt đầu khi bác sĩ đăng nhập vào hệ thống và chọn bệnh nhân từ danh sách lịch hẹn được phân công. Hệ thống hiển thị thông tin bệnh nhân và lịch sử khám bệnh để bác sĩ thực hiện thăm khám và cập nhật kết quả điều trị.

Sau khi khám, bác sĩ có thể lập kế hoạch điều trị, kê đơn thuốc và nhập các thông tin liên quan. Hệ thống hỗ trợ kiểm tra tồn kho thuốc trước khi lưu đơn thuốc và cập nhật dữ liệu kho thuốc.

Tiếp theo, nhân viên thực hiện tạo hóa đơn dựa trên các dịch vụ điều trị và thuốc đã sử dụng. Bệnh nhân có thể lựa chọn thanh toán bằng tiền mặt hoặc VNPay.

Nếu thanh toán thành công, hệ thống cập nhật trạng thái hóa đơn và gửi thông báo xác nhận cho bệnh nhân. Cuối cùng, bệnh nhân có thể xem lại thông tin điều trị, đơn thuốc và hóa đơn trực tiếp trên hệ thống.

2.5.4. Biểu đồ hoạt động chức năng thống kê và AI phân tích đề xuất



Hình 2.17: Biểu đồ hoạt động chức năng thống kê và AI phân tích đề xuất

Biểu đồ hoạt động mô tả quy trình admin xem thống kê và sử dụng AI để phân tích dữ liệu trong hệ thống quản lý nha khoa.

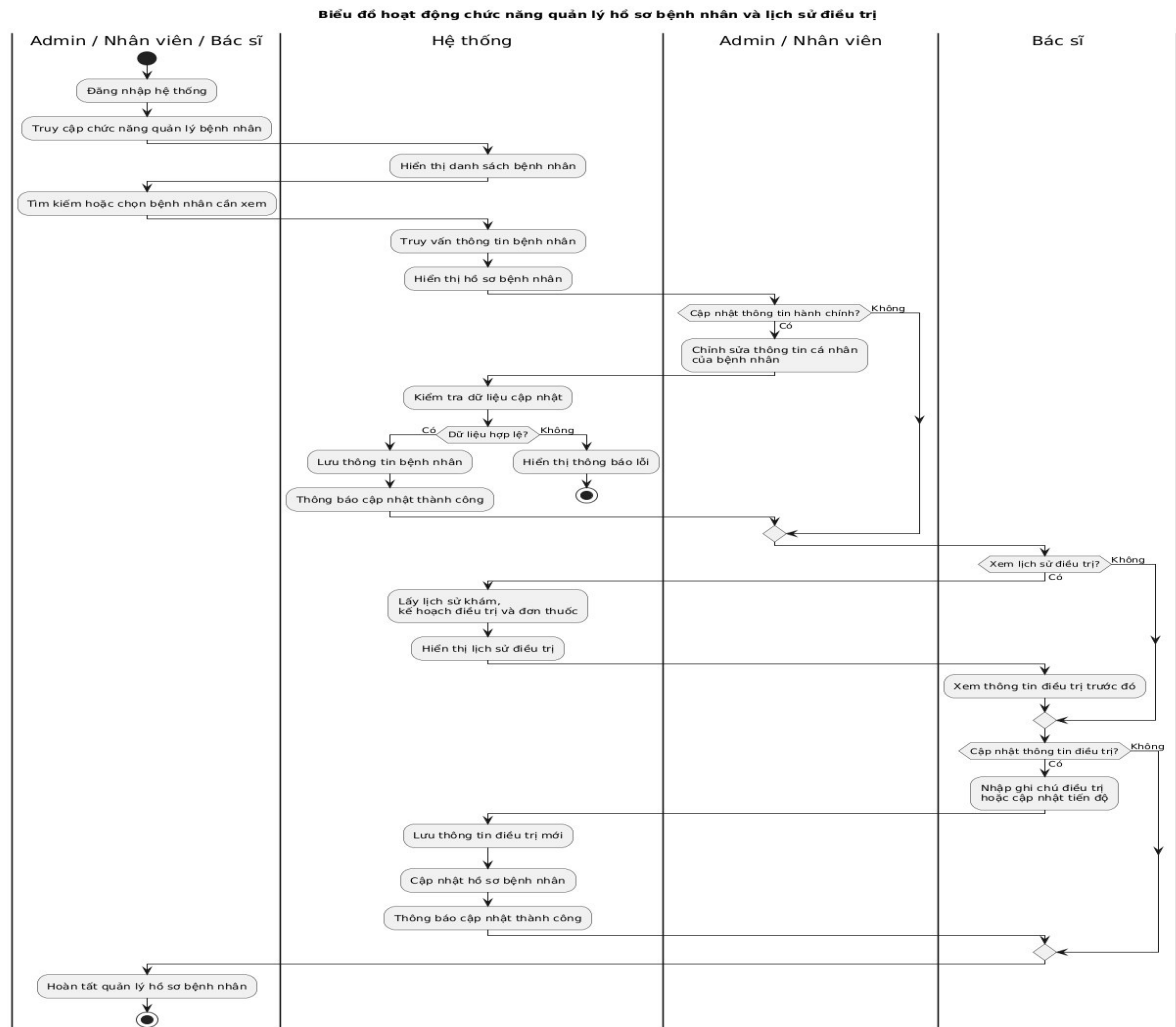
Quy trình bắt đầu khi admin đăng nhập và truy cập chức năng thống kê. Hệ thống hiển thị các dữ liệu như doanh thu, lịch hẹn, bệnh nhân, dịch vụ và thuốc dưới dạng số liệu hoặc biểu đồ trực quan.

Admin có thể lựa chọn khoảng thời gian và loại dữ liệu cần xem. Hệ thống tiến hành truy vấn, tổng hợp và hiển thị kết quả thống kê từ cơ sở dữ liệu.

Ngoài ra, hệ thống hỗ trợ AI phân tích dữ liệu hoạt động của phòng khám nhằm đưa ra các nhận xét và đề xuất phù hợp như dịch vụ cần phát triển, xu hướng sử dụng dịch vụ hoặc hỗ trợ tối ưu hoạt động quản lý.

Kết quả phân tích được hiển thị để admin tham khảo trong quá trình quản lý và vận hành hệ thống.

2.5.5. Biểu đồ hoạt động chức năng quản lý hồ sơ bệnh nhân và lịch sử điều trị



Hình 2.18: Biểu đồ hoạt động chức năng quản lý hồ sơ bệnh nhân và lịch sử

Biểu đồ hoạt động mô tả quy trình quản lý hồ sơ bệnh nhân và theo dõi lịch sử điều trị trong hệ thống quản lý nha khoa.

Quy trình bắt đầu khi admin, nhân viên hoặc bác sĩ truy cập chức năng quản lý bệnh nhân. Hệ thống hiển thị danh sách bệnh nhân và hỗ trợ tìm kiếm theo thông tin cá nhân hoặc mã bệnh nhân.

Sau khi chọn bệnh nhân, hệ thống hiển thị hồ sơ bao gồm thông tin cá nhân, lịch sử khám bệnh, kế hoạch điều trị, đơn thuốc và hóa đơn liên quan.

Admin hoặc nhân viên có thể cập nhật thông tin hành chính của bệnh nhân, trong khi bác sĩ có thể theo dõi lịch sử điều trị và cập nhật ghi chú chuyên môn khi cần thiết.

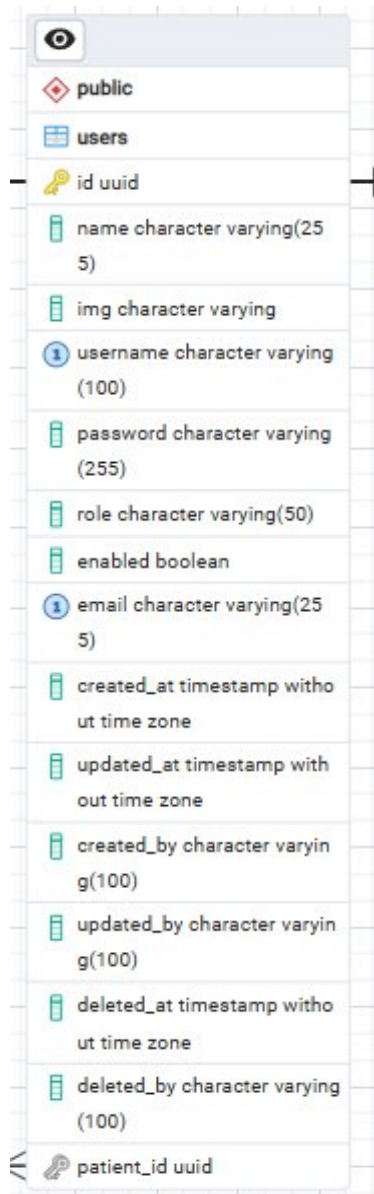
Hệ thống kiểm tra tính hợp lệ của dữ liệu trước khi lưu vào cơ sở dữ liệu nhằm đảm bảo thông tin được quản lý chính xác và thuận tiện cho quá trình theo dõi điều trị.

2.6. Thiết kế cơ sở dữ liệu

2.6.1. Các bảng trong cơ sở dữ liệu

2.6.1.1. Bảng users

Bảng users dùng để lưu trữ thông tin tài khoản của người dùng trong hệ thống. Đây là bảng quan trọng phục vụ cho quá trình xác thực (authentication), phân quyền (authorization) và quản lý hồ sơ người dùng.

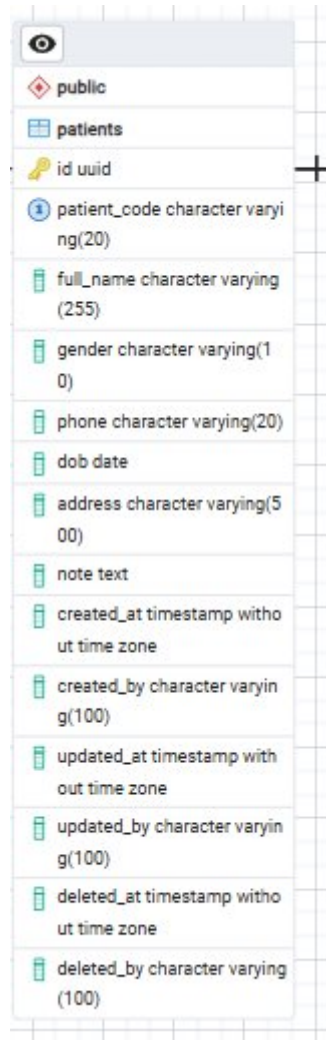


public
users
id uuid
name character varying(255)
img character varying
username character varying(100)
password character varying(255)
role character varying(50)
enabled boolean
email character varying(255)
created_at timestamp without time zone
updated_at timestamp without time zone
created_by character varying(100)
updated_by character varying(100)
deleted_at timestamp without time zone
deleted_by character varying(100)
patient_id uuid

Hình 2.19: Bảng users

2.6.1.2. Bảng patients

Bảng patients dùng để lưu trữ thông tin hồ sơ bệnh nhân trong hệ thống. Đây là bảng quan trọng phục vụ cho quá trình đặt lịch khám, theo dõi lịch sử khám, quản lý kế hoạch điều trị và liên kết thông tin bệnh nhân với các chức năng như lịch hẹn, đơn thuốc, hóa đơn và thanh toán.

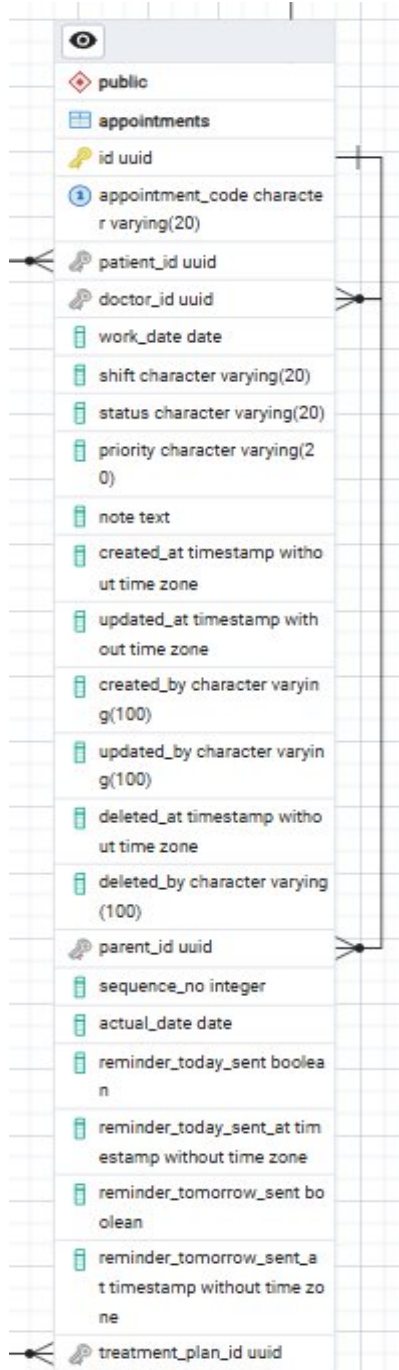


public
patients
id uuid
patient_code character varying(20)
full_name character varying(255)
gender character varying(10)
phone character varying(20)
dob date
address character varying(500)
note text
created_at timestamp with time zone
created_by character varying(100)
updated_at timestamp with time zone
updated_by character varying(100)
deleted_at timestamp with time zone
deleted_by character varying(100)

Hình 2.20: Bảng patients

2.6.1.3. Bảng appointments

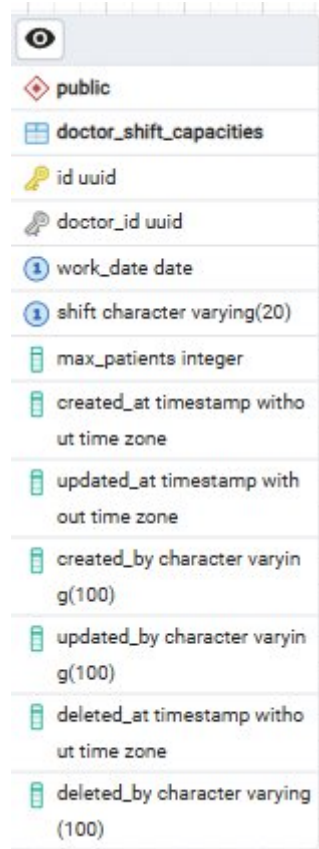
Bảng appointments dùng để lưu trữ thông tin lịch hẹn khám của bệnh nhân trong hệ thống. Bảng này phục vụ cho chức năng đặt lịch khám, xác nhận lịch hẹn, hủy lịch hẹn, phân công bác sĩ và theo dõi trạng thái khám của bệnh nhân.



Hình 2.21: Bảng appointments

2.6.1.4. Bảng `doctor_shift_capacities`

Bảng `doctor_shift_capacities` dùng để lưu trữ thông tin sức chứa và lịch làm việc của bác sĩ theo từng ngày và từng ca khám. Bảng này giúp hệ thống kiểm soát số lượng bệnh nhân tối đa mà bác sĩ có thể tiếp nhận, hỗ trợ quá trình phân công bác sĩ và tránh tình trạng trùng lịch hoặc quá tải lịch khám.

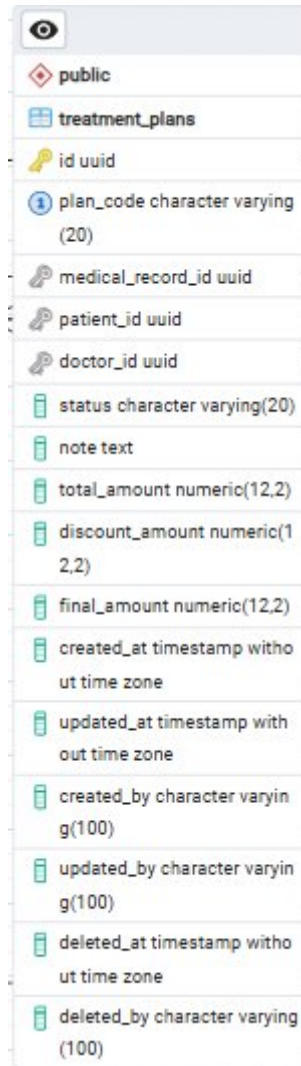


public
doctor_shift_capacities
id uuid
doctor_id uuid
work_date date
shift character varying(20)
max_patients integer
created_at timestamp with time zone
updated_at timestamp with time zone
created_by character varying(100)
updated_by character varying(100)
deleted_at timestamp with time zone
deleted_by character varying(100)

Hình 2.22: Bảng `doctor_shift_capacities`

2.6.1.5. Bảng treatment_plans

Bảng treatment_plans dùng để lưu trữ thông tin kế hoạch điều trị của bệnh nhân do bác sĩ lập sau quá trình thăm khám. Bảng này phục vụ cho việc theo dõi quá trình điều trị, quản lý các dịch vụ cần thực hiện, chi phí dự kiến, trạng thái điều trị và lịch tái khám của bệnh nhân.

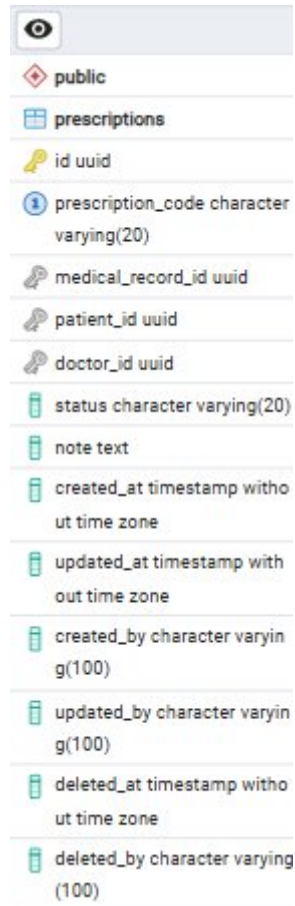


public
treatment_plans
id uuid
plan_code character varying (20)
medical_record_id uuid
patient_id uuid
doctor_id uuid
status character varying(20)
note text
total_amount numeric(12,2)
discount_amount numeric(12,2)
final_amount numeric(12,2)
created_at timestamp with out time zone
updated_at timestamp with out time zone
created_by character varying(100)
updated_by character varying(100)
deleted_at timestamp with out time zone
deleted_by character varying(100)

Hình 2.23: Bảng treatment_plans

2.6.1.6. Bảng prescriptions

Bảng prescriptions dùng để lưu trữ thông tin đơn thuốc được bác sĩ kê cho bệnh nhân sau khi khám hoặc điều trị. Bảng này giúp quản lý quá trình kê đơn, liên kết đơn thuốc với bệnh nhân, bác sĩ, lịch khám và hỗ trợ theo dõi việc sử dụng thuốc trong quá trình điều trị.

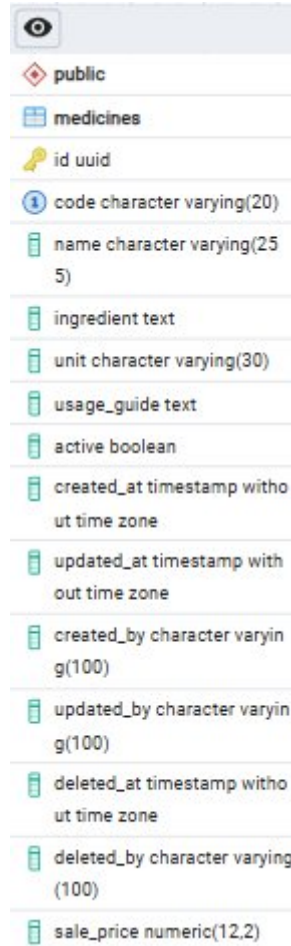


public
prescriptions
id uuid
prescription_code character varying(20)
medical_record_id uuid
patient_id uuid
doctor_id uuid
status character varying(20)
note text
created_at timestamp with time zone
updated_at timestamp with time zone
created_by character varying(100)
updated_by character varying(100)
deleted_at timestamp with time zone
deleted_by character varying(100)

Hình 2.24: Bảng prescriptions

2.6.1.7. Bảng medicines

Bảng medicines dùng để lưu trữ thông tin danh mục thuốc của phòng khám. Bảng này phục vụ cho chức năng quản lý thuốc, kê đơn thuốc, theo dõi tình trạng sử dụng thuốc và hỗ trợ kiểm soát dữ liệu thuốc trong hệ thống.



public
medicines
id uuid
code character varying(20)
name character varying(255)
ingredient text
unit character varying(30)
usage_guide text
active boolean
created_at timestamp without time zone
updated_at timestamp without time zone
created_by character varying(100)
updated_by character varying(100)
deleted_at timestamp without time zone
deleted_by character varying(100)
sale_price numeric(12,2)

Hình 2.25: Bảng medicines

CHƯƠNG 3. TRIỂN KHAI HỆ THỐNG VÀ KẾT QUẢ ĐẠT ĐƯỢC

3.1. Cấu hình triển khai

3.1.1. Backend – Spring Boot

- Framework: Spring Boot 3.5.3
- Ngôn ngữ: Java (JDK 21)
- Quản lý phụ thuộc: Gredlew
- Các module chính:
- Spring Web
- Spring Security + JWT
- Spring Data JPA
- PostgreSQL Driver
- Spring Validation
- Spring AI

3.1.2. Frontend – Vue.js

- Vue.js phiên bản: Vue 3
- Công cụ build: Vite
- Router: Vue Router 4
- State management: Pinia
- UI Framework: TailwindCSS

3.1.3. Cơ sở dữ liệu – PostgreSQL

- Quản lý dữ liệu toàn hệ thống
- Thiết kế theo mô hình quan hệ (ERD)
- Dùng pgAdmin để quản trị

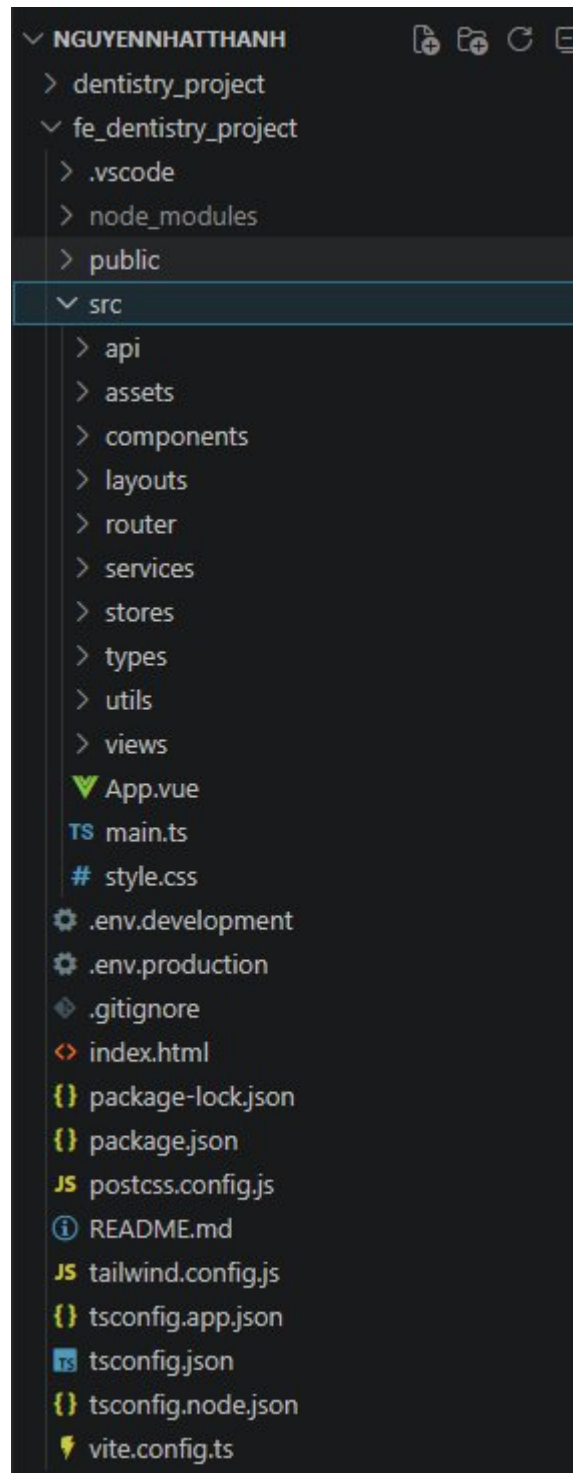
3.2. Cấu trúc thư mục

3.2.1. Backend chính



Hình 3.1: Cấu trúc thư mục backend chính

3.2.1. Frontend



Hình 3.2: Cấu trúc thư mục frontend

3.3. Triển khai hệ thống

3.3.1. Các bước triển khai

Quá trình triển khai hệ thống được thực hiện theo mô hình tách biệt giữa Frontend Vue.js, Backend Spring Boot và Cơ sở dữ liệu PostgreSQL. Các bước triển khai chính như sau:

Bước 1: Chuẩn bị môi trường

Cài đặt các công cụ cần thiết gồm Java JDK 21, Gradle, Node.js, PostgreSQL, pgAdmin, IntelliJ IDEA, Visual Studio Code và Postman/Swagger để phục vụ quá trình phát triển, chạy thử và kiểm thử hệ thống.

Bước 2: Cấu hình cơ sở dữ liệu

Tạo cơ sở dữ liệu PostgreSQL cho hệ thống nha khoa, sau đó cấu hình thông tin kết nối database trong backend. Cơ sở dữ liệu dùng để lưu trữ các thông tin như người dùng, bệnh nhân, bác sĩ, lịch hẹn, kế hoạch điều trị, đơn thuốc, thuốc, hóa đơn và thanh toán.

Bước 3: Triển khai Backend Spring Boot

Backend được chạy bằng Spring Boot, đảm nhiệm xử lý nghiệp vụ, xác thực JWT, phân quyền người dùng, kết nối cơ sở dữ liệu và cung cấp RESTful API cho frontend.

Lệnh chạy backend:

```
./gradlew bootRun
```

Backend sau khi chạy thường hoạt động tại:

<http://localhost:8080>

Bước 4: Kiểm thử API

Sử dụng Postman hoặc Swagger để kiểm tra các API chính như đăng nhập, đặt lịch khám, quản lý lịch hẹn, phân công bác sĩ, lập kế hoạch điều trị, kê đơn thuốc, tạo hóa đơn, thanh toán và thống kê.

Bước 5: Triển khai Frontend Vue.js

Frontend được xây dựng bằng Vue.js, dùng để hiển thị giao diện cho bệnh nhân, bác sĩ, nhân viên và admin. Frontend gọi API từ backend để xử lý các chức năng của hệ thống.

Lệnh cài đặt và chạy frontend:

```
npm install
```

```
npm run dev
```

Frontend sau khi chạy thường hoạt động tại:

`http://localhost:5173`

Bước 6: Kết nối Frontend và Backend

Cấu hình đường dẫn API trong frontend để kết nối với backend. Sau đó kiểm tra các luồng chính như đăng nhập, đặt lịch khám, xác nhận lịch hẹn, phân công bác sĩ, khám bệnh, kê đơn thuốc và thanh toán.

Bước 7: Cấu hình email và thanh toán

Hệ thống được cấu hình gửi email thông báo cho các sự kiện như đặt lịch thành công, xác nhận lịch hẹn, hủy lịch và thanh toán thành công. Ngoài ra, hệ thống tích hợp VNPay để hỗ trợ thanh toán hóa đơn trực tuyến.

Bước 8: Kiểm thử và hoàn thiện

Tiến hành kiểm thử toàn bộ hệ thống, bao gồm kiểm thử chức năng, giao diện, API, phân quyền và thanh toán. Các lỗi phát sinh được ghi nhận và chỉnh sửa để đảm bảo hệ thống hoạt động ổn định.

Bước 9: Build hệ thống

Sau khi kiểm thử hoàn tất, backend và frontend được build để chuẩn bị triển khai.

Build backend:

`./gradlew clean build`

Build frontend:

`npm run build`

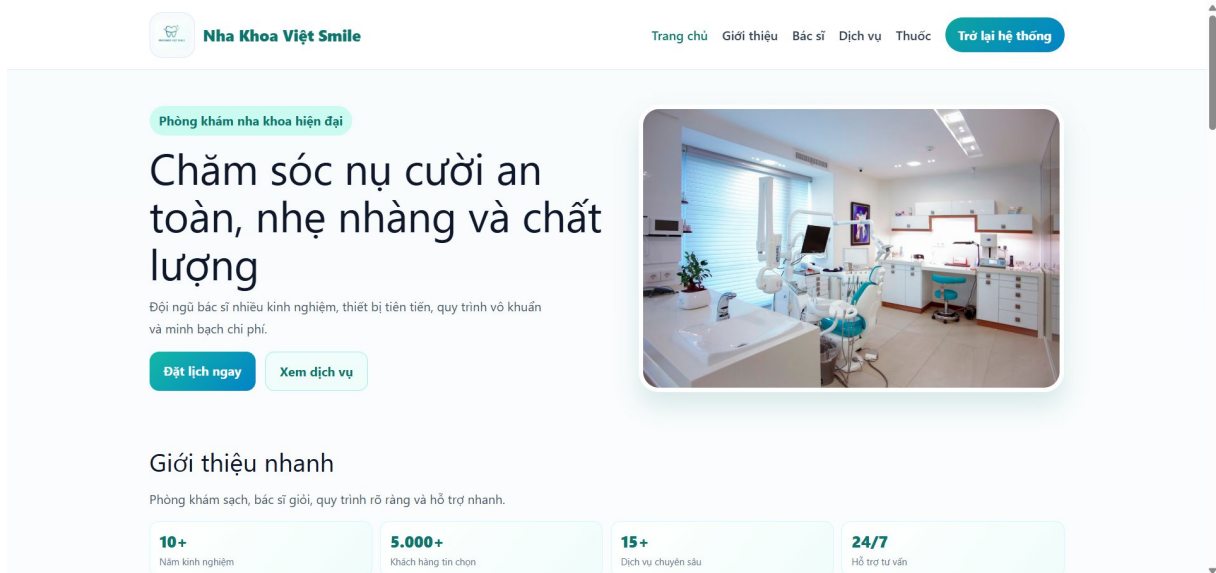
3.3.2. Kết quả sau triển khai

Sau khi triển khai, hệ thống hoạt động ổn định với các chức năng chính như đặt lịch khám, quản lý lịch hẹn, phân công bác sĩ, quản lý bệnh nhân, lập kế hoạch điều trị, kê đơn thuốc, quản lý hóa đơn, thanh toán VNPay, quản lý thuốc, thống kê và AI phân tích đề xuất. Hệ thống góp phần hỗ trợ phòng khám nha khoa tối ưu quy trình quản lý, giảm thao tác thủ công và nâng cao trải nghiệm người dùng.

3.4. Kết quả đạt được

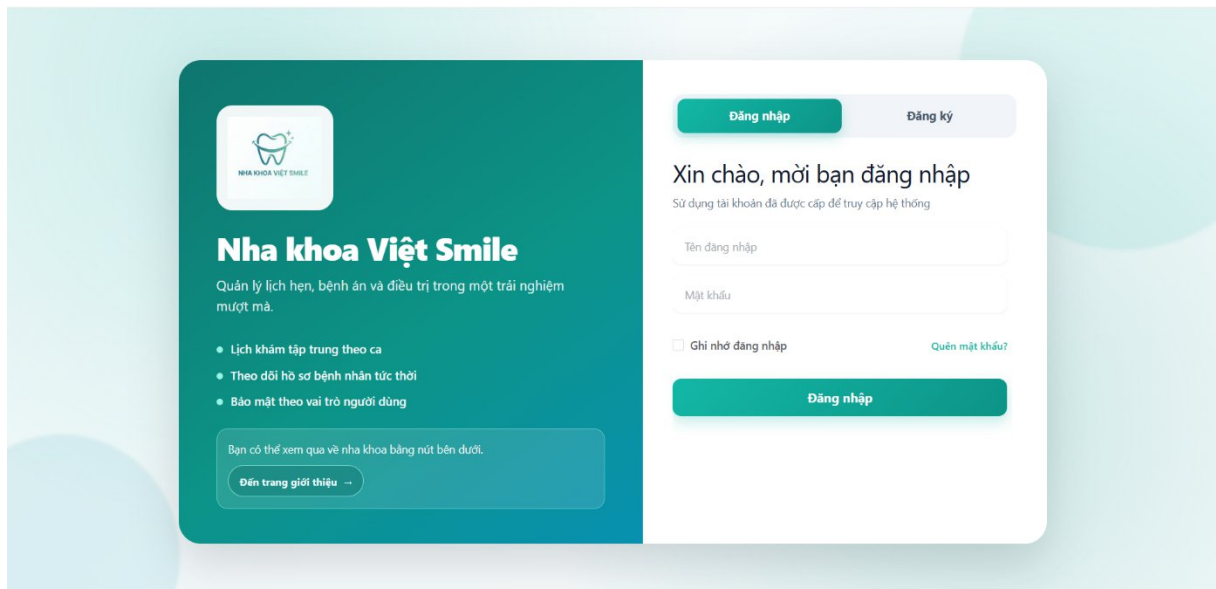
3.4.1. Giao diện bệnh nhân

Trang chủ phòng khám

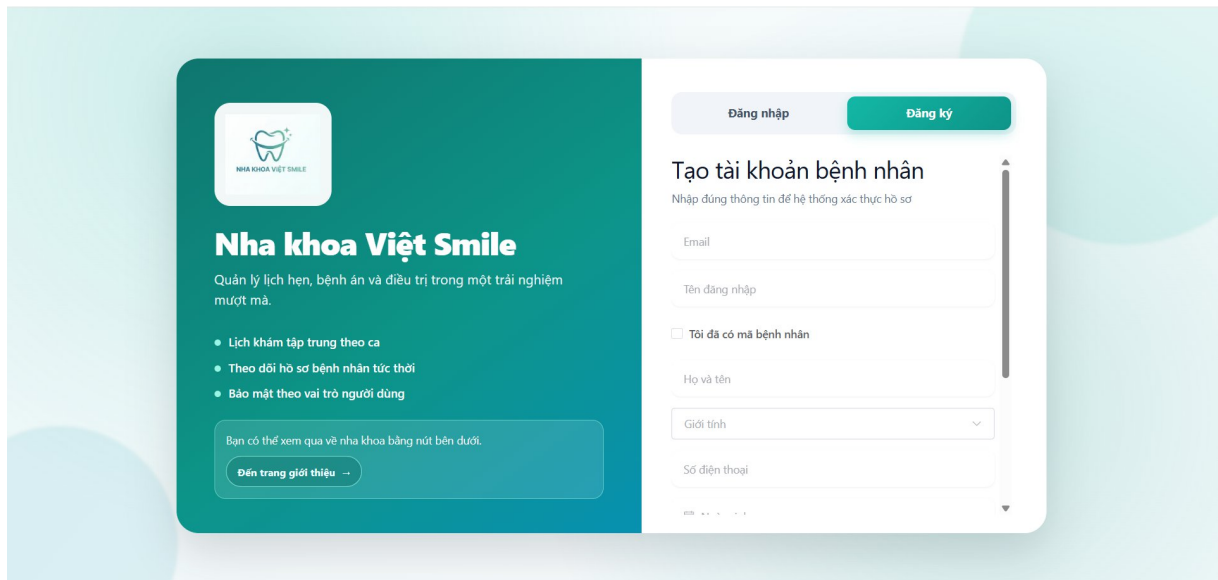


Hình 3.3: Trang chủ phòng khám

Đăng ký / đăng nhập

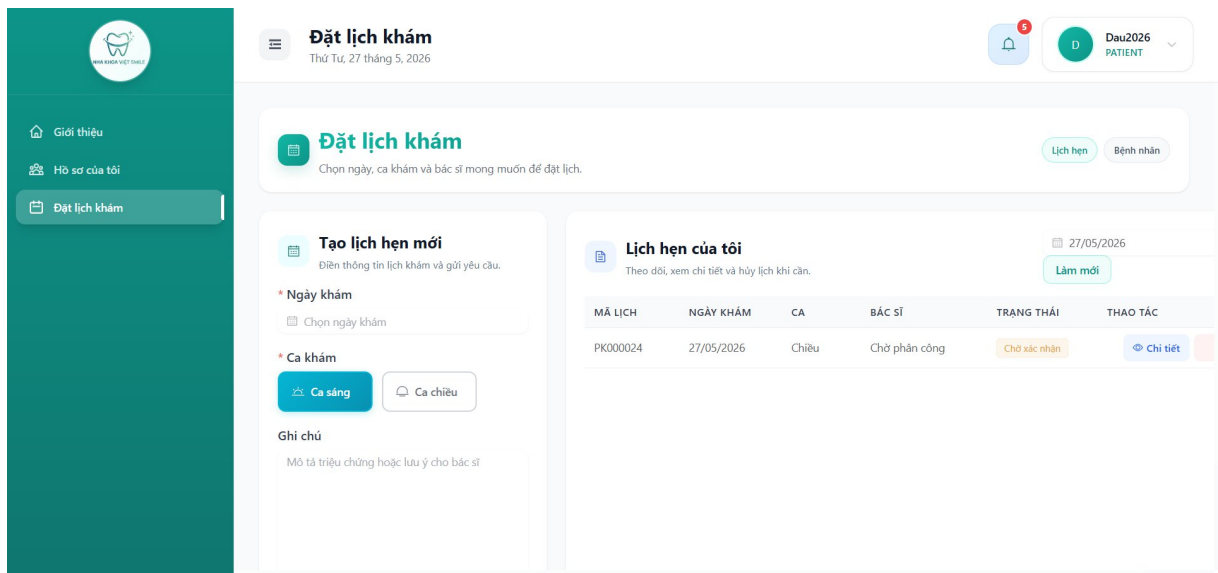


Hình 3.4: Đăng Nhập



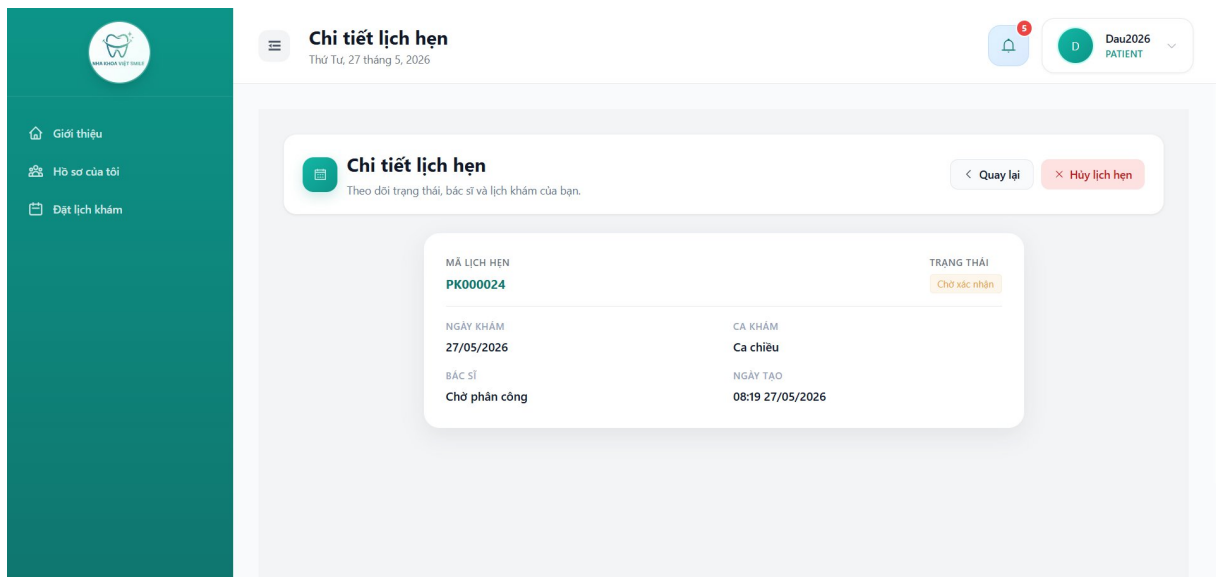
Hình 3.5: Đăng Ký

Đặt lịch khám



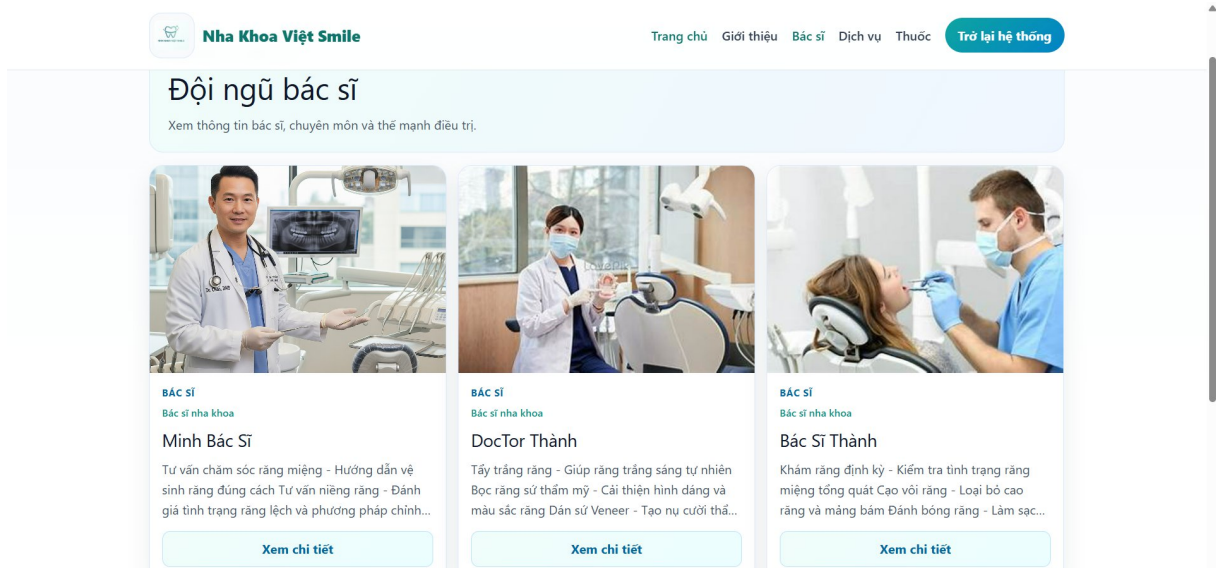
Hình 3.6: Đặt lịch khám

Xem lịch hẹn



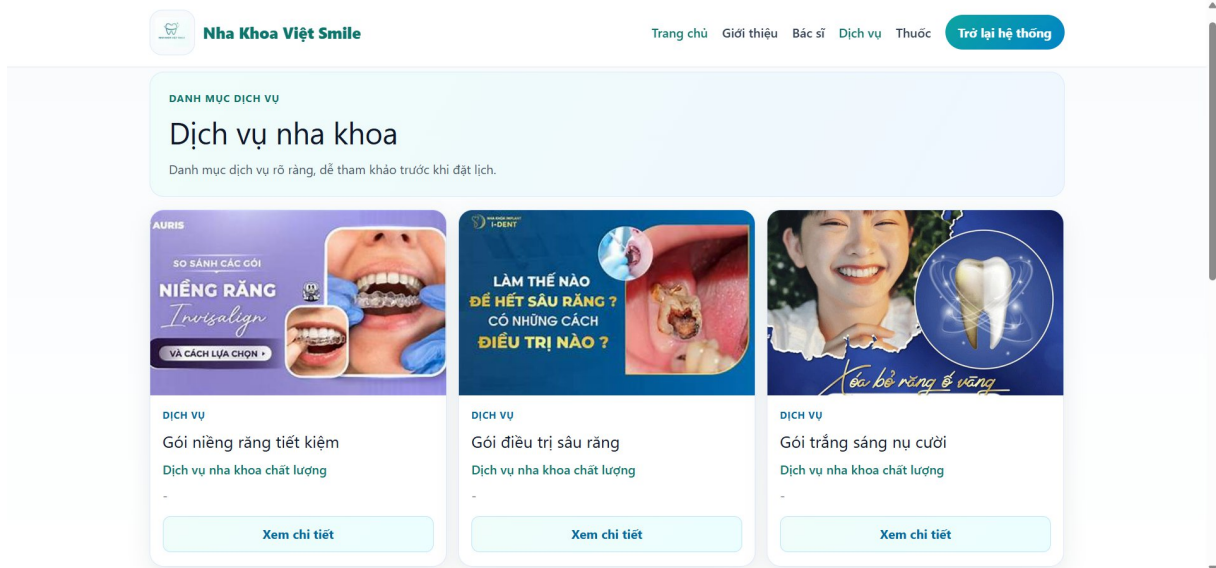
Hình 3.7: Xem lịch hẹn

Xem thông tin bác sĩ



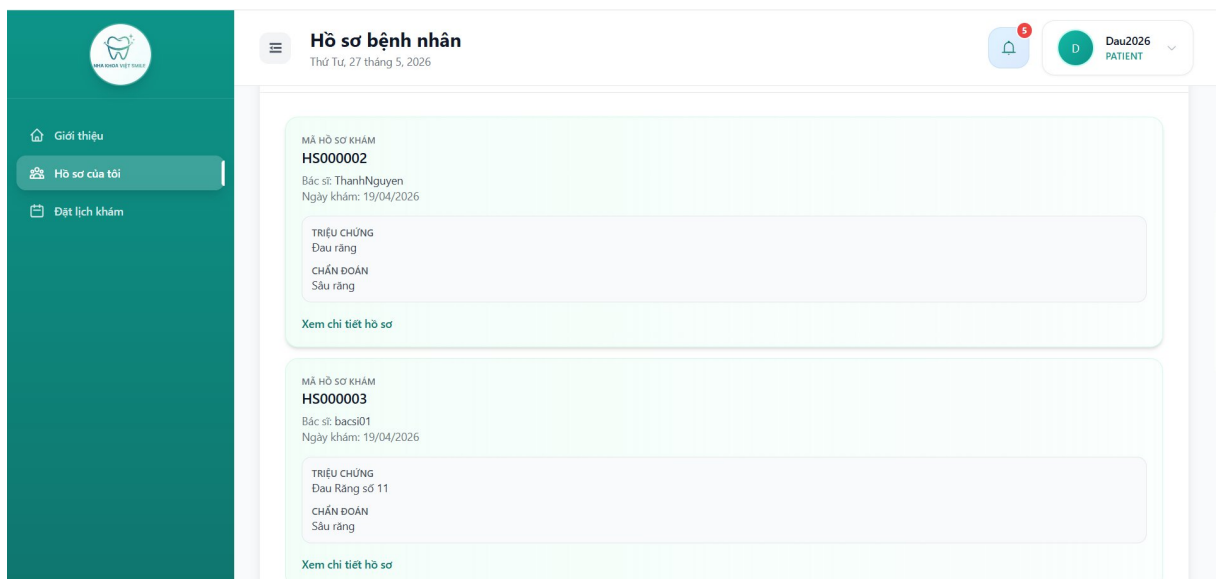
Hình 3.8: Xem thông tin bác sĩ

Xem dịch vụ nha khoa



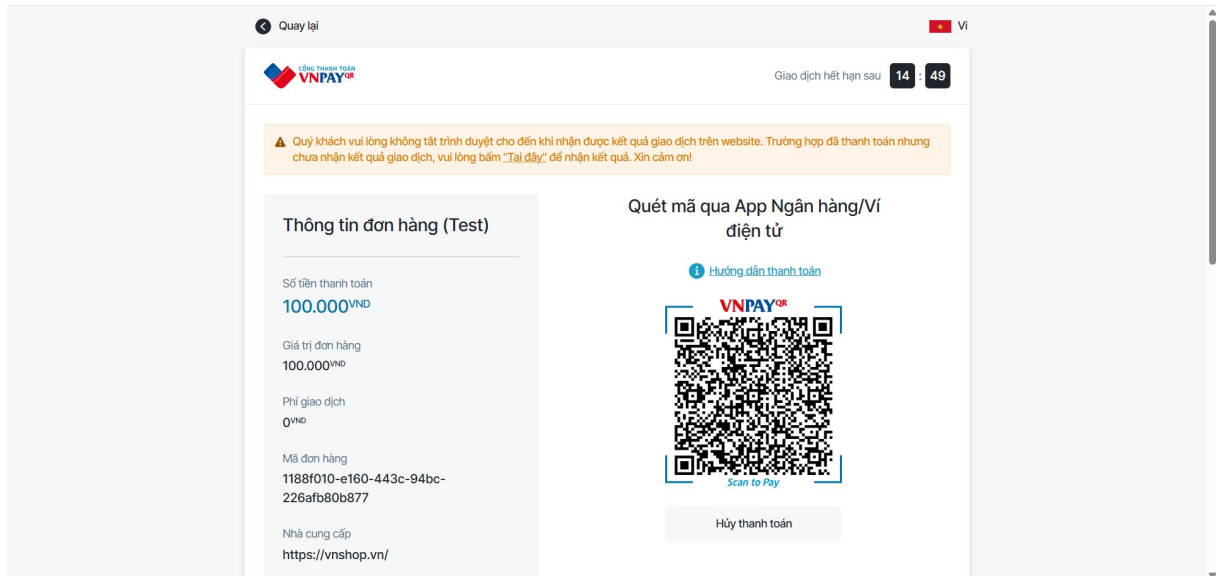
Hình 3.9: Xem dịch vụ nha khoa

Xem lịch sử khám



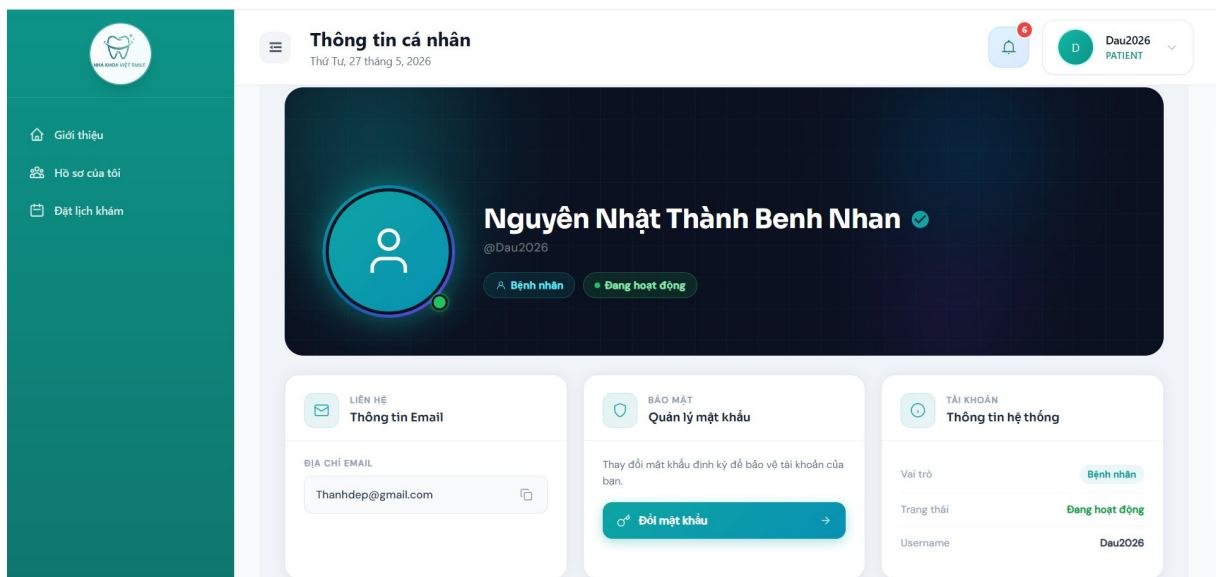
Hình 3.10: Xem lịch sử khám

Thanh toán VNPay



Hình 3.11: Thanh toán VNPay

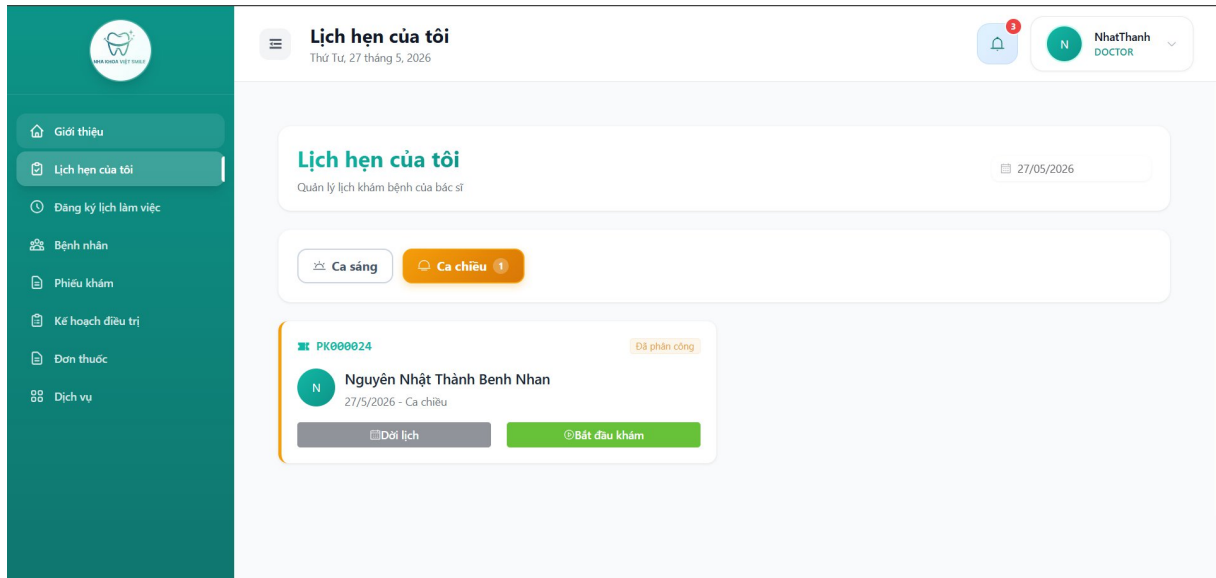
Hồ sơ cá nhân



Hình 3.12: Hồ sơ cá nhân

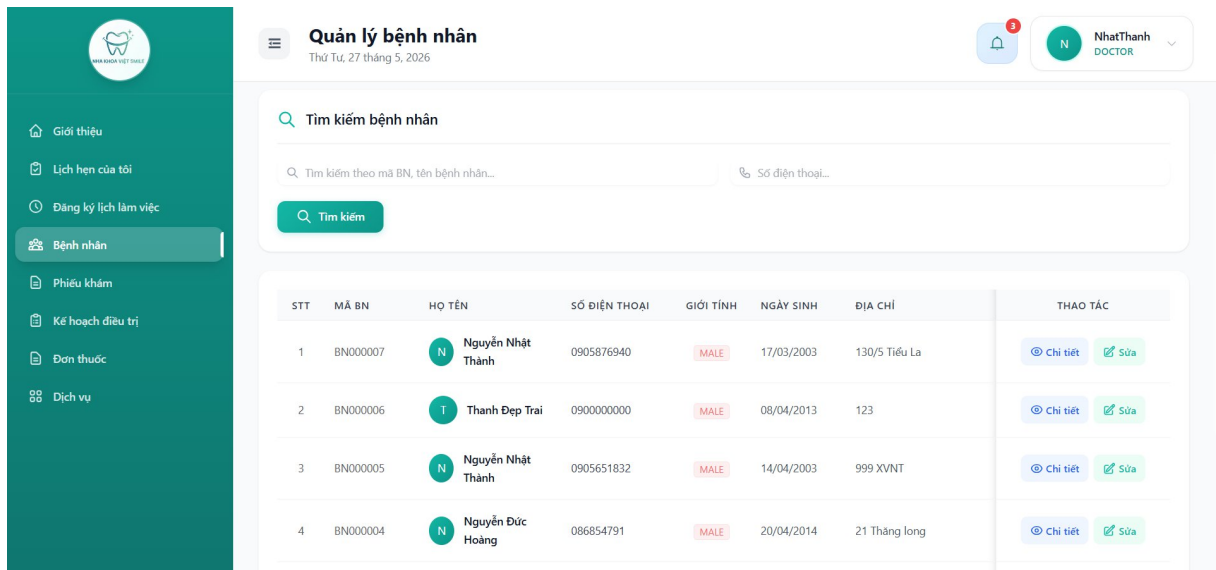
3.4.2. Giao diện bác sĩ

Danh sách lịch khám



Hình 3.13: Danh sách lịch khám

Thông tin bệnh nhân



Hình 3.14: Thông tin bệnh nhân

Kế hoạch điều trị

The screenshot displays the 'Kế hoạch điều trị' (Treatment Plan) page. The sidebar on the left contains navigation links: Giới thiệu, Lịch hẹn của tôi, Đăng ký lịch làm việc, Bệnh nhân, Phiếu khám, Kế hoạch điều trị (highlighted), Đơn thuốc, and Dịch vụ. The header shows the user's name 'NhatThanh DOCTOR' and a notification bell. The main content area has a title 'Kế hoạch điều trị' and a subtitle 'Quản lý kế hoạch điều trị cho bệnh nhân'. Below this is a search bar with the text 'Tìm kiếm kế hoạch điều trị'. The search results show a table with columns: MÃ KH, MÃ BN, BỆNH NHÂN, BÁC SĨ, TRẠNG THÁI, TỔNG TIỀN, GIÁM GI, and THAO TÁC. The table contains two rows of data.

MÃ KH	MÃ BN	BỆNH NHÂN	BÁC SĨ	TRẠNG THÁI	TỔNG TIỀN	GIÁM GI	THAO TÁC
KH000018	BN000002	Nguyễn Nhật Thành Bệnh Nhân	NhatThanh	Hoàn thành	100.000 đ	0 đ	Xem
KH000017	BN000002	Nguyễn Nhật Thành Bệnh Nhân	NhatThanh	Hoàn thành	2.500.000 đ	0 đ	Xem

Hình 3.15: Kế hoạch điều trị

Kê đơn thuốc

The screenshot displays the 'Quản lý đơn thuốc' (Manage Prescription) page. The sidebar on the left contains navigation links: Giới thiệu, Lịch hẹn của tôi, Đăng ký lịch làm việc, Bệnh nhân, Phiếu khám, Kế hoạch điều trị, Đơn thuốc (highlighted), and Dịch vụ. The header shows the user's name 'NhatThanh DOCTOR' and a notification bell. The main content area has a title 'Quản lý đơn thuốc' and a subtitle 'Thư Tư, 27 tháng 5, 2026'. Below this is a 'Tạo đơn thuốc' (Create Prescription) button. A modal window is open for adding a new prescription, with fields for: * Thuốc (TH000001 - Thuốc HDA), Tên kho hiện tại (71 Viên), Liều dùng (9), * Số lượng (18), and Ghi chú (Ghi chú thêm...). The modal has 'Hủy' (Cancel) and 'Thêm' (Add) buttons. The background shows a table of prescriptions with columns: STT, MÃ ĐƠN THUỐC, BỆNH NHÂN, BÁC SĨ, TRẠNG THÁI, NGÀY TẠO, and THAO TÁC.

STT	MÃ ĐƠN THUỐC	BỆNH NHÂN	BÁC SĨ	TRẠNG THÁI	NGÀY TẠO	THAO TÁC
1	DT000009	alabatrap	admin	68 xuất	22:21:44 17/5/2026	Chi tiết
2	DT000009	Thanh Đẹp Trai	Thanh Nguyen	68 xuất	11:42:41 9/5/2026	Chi tiết Hủy

Hình 3.16: Kê đơn thuốc

Quản lý tái khám

The screenshot shows the 'Chi tiết kế hoạch điều trị' (Treatment Plan Details) page for patient KH000011. The page is divided into a sidebar on the left and a main content area. The sidebar contains links to various sections: Giới thiệu, Lịch hẹn của tôi, Đăng ký lịch làm việc, Bệnh nhân, Phiếu khám, Kế hoạch điều trị, Đơn thuốc, and Dịch vụ. The main content area has a header with the patient's name 'NhatThanh DOCTOR' and a 'Tạo tái khám' (Create Re-examination) button. Below the header, there is a section for patient information, including the patient's name, ID, and date of birth. The patient's name is 'Nguyễn Nhật Thành Bệnh Nhân', the ID is 'BN000002', and the date of birth is '21:43:37 6/5/2026'. The patient's status is 'Đang thực hiện' (In progress). Below the patient information, there is a section for the treatment plan, including the patient's name, ID, and date of birth. The patient's name is 'Nguyễn Nhật Thành Bệnh Nhân', the ID is 'BN000002', and the date of birth is '21:44:21 6/5/2026'. The patient's status is 'Đang thực hiện' (In progress). Below the treatment plan, there is a section for the list of services, including the patient's name, ID, and date of birth. The patient's name is 'Nguyễn Nhật Thành Bệnh Nhân', the ID is 'BN000002', and the date of birth is '21:44:21 6/5/2026'. The patient's status is 'Đang thực hiện' (In progress).

Hình 3.17: Quản lý tái khám

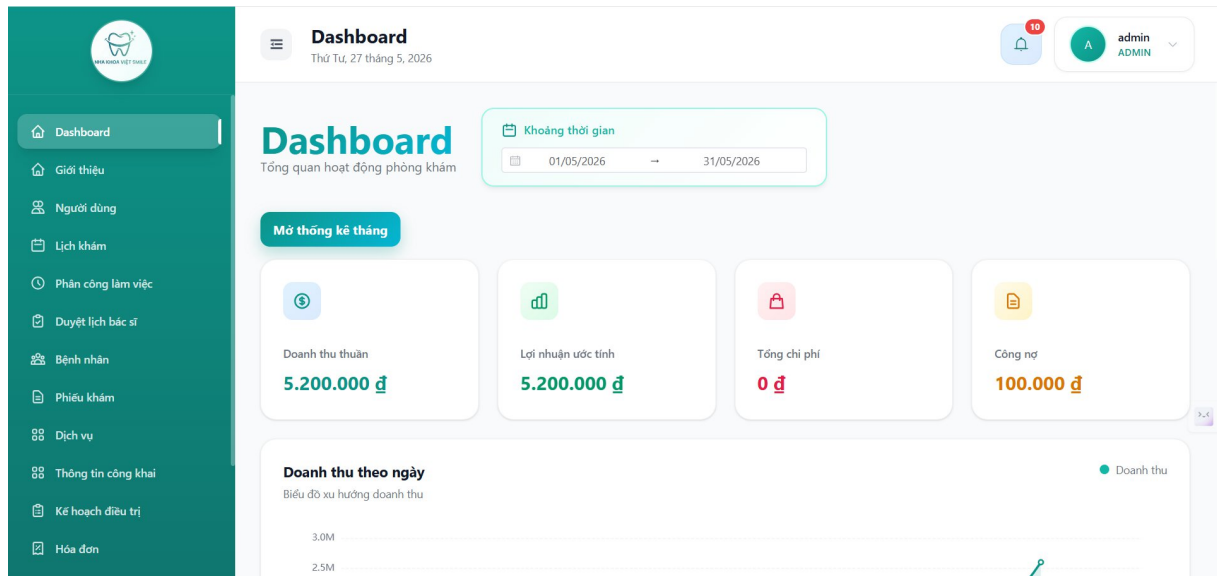
Theo dõi hồ sơ điều trị

The screenshot shows the 'Chi tiết kế hoạch điều trị' (Treatment Plan Details) page for patient KH000018. The page is divided into a sidebar on the left and a main content area. The sidebar contains links to various sections: Giới thiệu, Lịch hẹn của tôi, Đăng ký lịch làm việc, Bệnh nhân, Phiếu khám, Kế hoạch điều trị, Đơn thuốc, and Dịch vụ. The main content area has a header with the patient's name 'NhatThanh DOCTOR' and a 'Hoàn thành' (Complete) button. Below the header, there is a section for patient information, including the patient's name, ID, and date of birth. The patient's name is 'Nguyễn Nhật Thành Bệnh Nhân', the ID is 'BN000002', and the date of birth is '08:44:23 27/5/2026'. The patient's status is 'Hoàn thành' (Complete). Below the patient information, there is a section for the treatment plan, including the patient's name, ID, and date of birth. The patient's name is 'Nguyễn Nhật Thành Bệnh Nhân', the ID is 'BN000002', and the date of birth is '08:44:29 27/5/2026'. The patient's status is 'Hoàn thành' (Complete). Below the treatment plan, there is a section for the list of services, including the patient's name, ID, and date of birth. The patient's name is 'Nguyễn Nhật Thành Bệnh Nhân', the ID is 'BN000002', and the date of birth is '08:44:29 27/5/2026'. The patient's status is 'Hoàn thành' (Complete).

Hình 3.18: Theo dõi hồ sơ điều trị

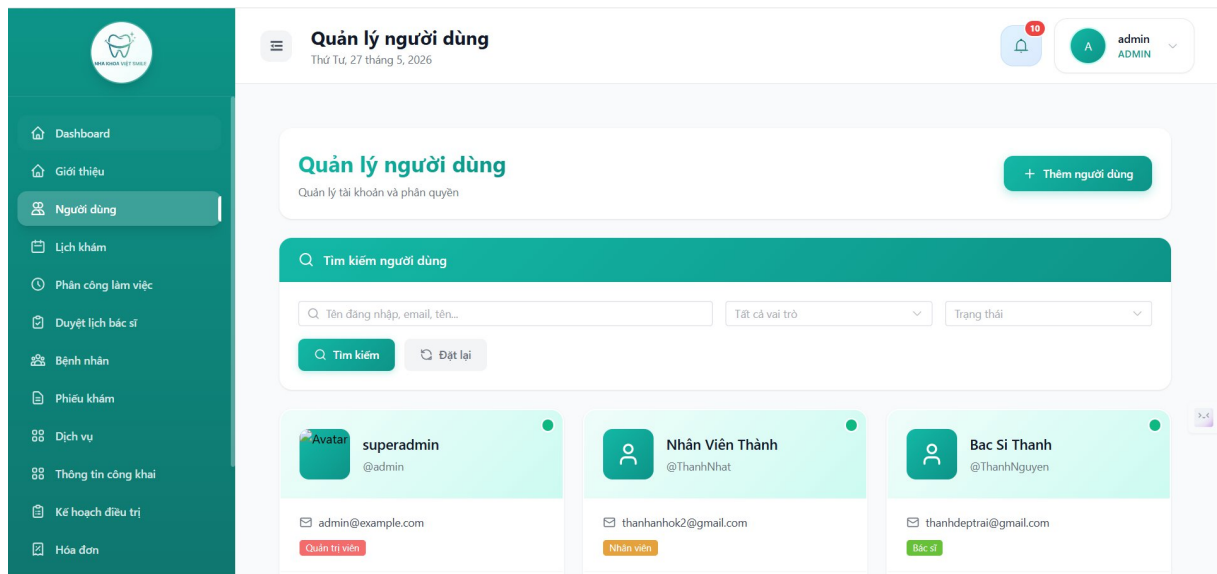
3.4.3. Giao diện quản trị (ADMIN)

Dashboard thống kê



Hình 3.19: Dashboard thống kê

Quản lý tài khoản



Hình 3.20: Quản lý tài khoản

Quản lý lịch hẹn

The screenshot displays the 'Quản lý lịch khám' (Appointment Management) interface. The sidebar on the left contains navigation links: Dashboard, Giới thiệu, Người dùng, Lịch khám (selected), Phân công làm việc, Duyệt lịch bác sĩ, Bệnh nhân, Phiếu khám, Dịch vụ, Thông tin công khai, Kế hoạch điều trị, and Hóa đơn. The main content area has a header with the title 'Quản lý lịch khám' and a subtitle 'Tạo và quản lý lịch hẹn khám bệnh'. A search bar is present with the text 'Tìm kiếm lịch hẹn'. Below the search bar, there is a table with columns: STT, MÃ PHIẾU, BỆNH NHÂN, NGÀY KHÁM, CA, BÁC SĨ, and THAO TÁC. The table contains one row of data for an appointment on 27/05/2026. The bottom of the page shows pagination information: 'Total 1', '10/page', and 'Go to 1'.

STT	MÃ PHIẾU	BỆNH NHÂN	NGÀY KHÁM	CA	BÁC SĨ	THAO TÁC
1	PK000024	Nguyễn Nhật Thành Benh Nhan	27/5/2026	Chiều	NhatThanh	Chi tiết

Hình 3.21: Quản lý lịch hẹn

Quản lý dịch vụ

The screenshot displays the 'Danh mục dịch vụ' (Service Category) interface. The sidebar on the left contains navigation links: Dashboard, Giới thiệu, Người dùng, Lịch khám, Phân công làm việc, Duyệt lịch bác sĩ, Bệnh nhân, Phiếu khám, Dịch vụ (selected), Thông tin công khai, Kế hoạch điều trị, and Hóa đơn. The main content area has a header with the title 'Danh mục dịch vụ' and a subtitle 'Quản lý danh mục dịch vụ nha khoa'. A search bar is present with the text 'Tìm kiếm dịch vụ'. Below the search bar, there is a table with columns: MÃ DỊCH VỤ, TÊN DỊCH VỤ, LOẠI, DANH MỤC, GIÁ, ĐƠN VỊ, and THAO TÁC. The table contains three rows of data for different service categories. The bottom of the page shows pagination information: 'Total 3', '10/page', and 'Go to 3'.

MÃ DỊCH VỤ	TÊN DỊCH VỤ	LOẠI	DANH MỤC	GIÁ	ĐƠN VỊ	THAO TÁC
PK000004	Gói niềng răng tiết kiệm	Gói dịch vụ	Chỉnh nha	25.000.000 đ	gói	Xem Sửa Xóa
PK000003	Gói điều trị sâu răng	Gói dịch vụ	Điều trị răng sâu	1.500.000 đ	gói	Xem Sửa Xóa
PK000002	Gói trắng sáng nụ cười	Gói dịch vụ	Nha khoa thẩm mỹ	2.800.000 đ	gói	Xem Sửa Xóa

Hình 3.22: Quản lý dịch vụ

Quản lý thuốc

Quản lý thuốc
Thứ Tư, 27 tháng 5, 2026

Quản lý thuốc
Quản lý danh mục thuốc và giá bán

Tìm kiếm thuốc

Tên thuốc, thành phần... Trạng thái

Tìm kiếm Đặt lại

STT	MÃ THUỐC	TÊN THUỐC	THÀNH PHẦN	ĐƠN VỊ	GIÁ BÁN	THAO TÁC
1	TH000001	Thuốc HDA	1132	Vien	50.000 đ	Chi tiết Đặt giá Nhập lô Lịch sử lô

Hình 3.23: Quản lý thuốc

Quản lý hóa đơn

Quản lý hóa đơn
Thứ Tư, 27 tháng 5, 2026

Quản lý hóa đơn
Quản lý và theo dõi các hóa đơn thanh toán

Tìm kiếm hóa đơn

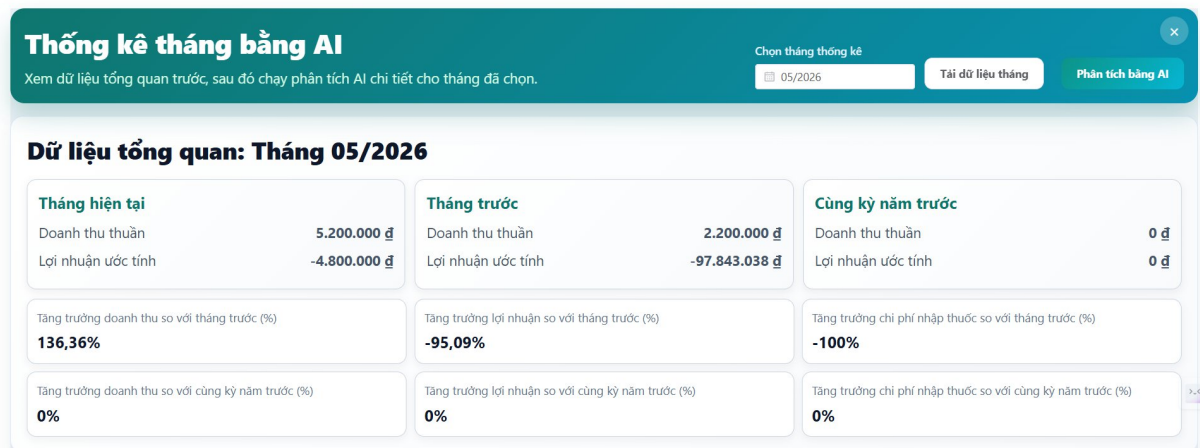
Chọn bệnh nhân Trạng thái 01/05/2026 - 31/05/2026

Tìm kiếm Đặt lại

MÃ HĐ	MÃ BN	BỆNH NHÂN	THU NGÂN	TRẠNG THÁI	TỔNG TIỀN	THAO TÁC
HD000017	BN000002	Nguyễn Nhật Thành Bệnh Nhân	admin	Đã phát hành	100.000 đ	Xem Thanh toán Hủy
HD000016	BN000002	Nguyễn Nhật Thành Bệnh Nhân	admin	Đã thanh toán	2.500.000 đ	Xem

Hình 3.24: Quản lý hóa đơn

Quản lý doanh thu



Hình 3.25: Quản lý doanh thu

3.4.4. Giao diện nhân viên

Tiếp nhận bệnh nhân

Quản lý bệnh nhân
Thứ Tư, 27 tháng 5, 2026

Quản lý bệnh nhân
Quản lý thông tin bệnh nhân

Tìm kiếm bệnh nhân

Q. Tìm kiếm theo mã BN, tên bệnh nhân... Số điện thoại...

Tìm kiếm

STT	MÃ BN	HỌ TÊN	SỐ ĐIỆN THOẠI	GIỚI TÍNH	NGÀY SINH	ĐỊA CHỈ	THAO TÁC
1	BN000007	N Nguyễn Nhật Thành	0905876940	MALE	17/03/2003	130/5 Tiểu La	Chi tiết Sửa
2	BN000006	T Thanh Đẹp Trai	0900000000	MALE	08/04/2013	123	Chi tiết Sửa

Hình 3.26: Tiếp nhận bệnh nhân

Xác nhận lịch hẹn

Chi tiết lịch hẹn
Thứ Tư, 27 tháng 5, 2026

[Quay lại](#)

Chi tiết lịch hẹn [Đăng khám](#)

[Tạo lịch hẹn tiếp theo](#)

Thông tin lịch hẹn

Mã phiếu: PK000024
Ngày khám: 27/05/2026
Lần khám: #1
Ca khám: [Ca chiều](#)
Ngày tạo: 08:19 27/05/2026

Thông tin bệnh nhân

Nguyễn Nhật Thành Benh Nhan
Mã BN: BN000002

Thông tin bác sĩ

NhatThanh

Hình 3.27: Xác nhận lịch hẹn

Quản lý danh sách lịch khám

Quản lý lịch khám
Thứ Tư, 27 tháng 5, 2026

Quản lý lịch khám [+ Tạo lịch hẹn](#)

Tạo và quản lý lịch hẹn khám bệnh

Tìm kiếm lịch hẹn

27/05/2026 [Chọn ca](#) [Trạng thái](#)

[Tìm kiếm](#)

STT	MÃ PHIẾU	BỆNH NHÂN	NGÀY KHÁM	CA	BÁC SĨ	THAO TÁC
1	PK000024	Nguyễn Nhật Thành Benh Nhan	27/5/2026	Chiều	NhatThanh	Chi tiết

Total 1 [10/page](#) [1](#) [Go to](#) [1](#)

Hình 3.28: Quản lý danh sách lịch khám

Hỗ trợ phân công bác sĩ

Tạo lịch hẹn khám

* Bệnh nhân

Nguyễn Nhật Thành Benh Nhan

* Ngày khám

27/05/2026

* Ca khám

☀️ Ca sáng

🌙 Ca chiều

Bác sĩ

Doctor Thanh

Ghi chú

Nhập ghi chú về lịch hẹn (không bắt buộc)...

× Hủy

✓ Tạo mới

Hình 3.29: Hỗ trợ phân công bác sĩ

Quản lý hóa đơn thanh toán

AMH DENTAL

Giới thiệu

Quản lý lịch khám

Bệnh nhân

Hóa đơn

Kho thuốc

Đơn thuốc

Báo cáo tồn kho

Dịch vụ

Thông tin công khai

Chi phí

Quản lý hóa đơn

Thứ Tư, 27 tháng 5, 2026

Nhanvien01

CASHIER

Quản lý hóa đơn

Quản lý và theo dõi các hóa đơn thanh toán

+ Tạo hóa đơn

Tìm kiếm hóa đơn

Chọn bệnh nhân

Trạng thái

01/05/2026 - 31/05/2026

Tìm kiếm

Đặt lại

MÃ HD	MÃ BN	BỆNH NHÂN	THU NGÂN	TRANG THÁI	TỔNG TIỀN	THAO TÁC
HD000017	BN000002	Nguyễn Nhật Thành Benh Nhan	admin	Đã phát hành	100.000	Xem Thanh toán
HD000016	BN000002	Nguyễn Nhật Thành Benh Nhan	admin	Đã thanh toán	2.500.000	Xem

Hình 3.30: Quản lý hóa đơn thanh toán

Xác nhận thanh toán VNPay

The screenshot shows a modal window titled "Thêm thanh toán" (Add Payment) overlaid on a blurred background of the application interface. The modal contains the following information:

- Mã hóa đơn** (Invoice Code): HD000017
- Bệnh nhân** (Patient): Nguyễn Nhật Thành Benh Nhan
- Tổng tiền** (Total Amount): 100.000 đ
- Đã thanh toán** (Paid Amount): 0 đ
- Còn lại** (Remaining Amount): 100.000 đ
- Phương thức thanh toán** (Payment Method): Chuyển khoản (Dropdown menu)
- Số tiền thanh toán** (Payment Amount): 100000
- Thanh toán hết** (Payment Complete) button
- Mã tham chiếu** (Reference Code): # Mã giao dịch, số chứng từ...

Hình 3.31: Xác nhận thanh toán VNPay

Hỗ trợ cập nhật thông tin bệnh nhân

The screenshot shows a modal window titled "Cập nhật bệnh nhân" (Update Patient) overlaid on a blurred background of the application interface. The modal contains the following information:

- Họ tên** (Full Name): Nguyễn Nhật Thành
- Số điện thoại** (Phone Number): 0905876940
- Giới tính** (Gender): MALE (Dropdown menu)
- Ngày sinh** (Date of Birth): 17/03/2003
- Địa chỉ** (Address): 130/5 Tiểu La
- Ghi chú** (Remarks): Tạo từ đăng ký tài khoản
- Buttons**: "Hủy" (Cancel) and "Cập nhật" (Update)

Hình 3.32: Hỗ trợ cập nhật thông tin bệnh nhân

KẾT LUẬN

Đề tài “**Hệ thống quản lý nha khoa tư nhân sử dụng kiến trúc VueJS và Spring Boot**” đã được hoàn thành với mục tiêu xây dựng một hệ thống quản lý và đặt lịch khám nha khoa hiện đại, hỗ trợ số hóa các hoạt động quản lý tại phòng khám. Hệ thống được xây dựng dựa trên kiến trúc frontend và backend tách biệt nhằm đảm bảo tính ổn định, khả năng mở rộng và thuận tiện trong quá trình phát triển và bảo trì.

1. Thành tựu đạt được:

Hệ thống đã xây dựng thành công chức năng quản lý và đặt lịch khám nha khoa trực tuyến. Bệnh nhân có thể đăng ký tài khoản, đăng nhập, đặt lịch khám, theo dõi trạng thái lịch hẹn, xem lịch sử khám bệnh, kế hoạch điều trị, đơn thuốc và hóa đơn trực tiếp trên hệ thống.

Đối với bác sĩ, hệ thống hỗ trợ xem lịch hẹn được phân công, xem thông tin bệnh nhân, cập nhật kết quả khám, lập kế hoạch điều trị, kê đơn thuốc và tạo lịch tái khám. Các dữ liệu điều trị được lưu trữ tập trung giúp quá trình theo dõi và quản lý thuận tiện hơn.

Đối với nhân viên và admin, hệ thống hỗ trợ quản lý lịch hẹn, xác nhận hoặc hủy lịch khám, phân công bác sĩ, quản lý thuốc, hóa đơn và thanh toán. Ngoài ra, hệ thống còn hỗ trợ thống kê doanh thu, số lượng bệnh nhân và tình hình hoạt động của phòng khám.

Đề tài sử dụng VueJS để xây dựng giao diện người dùng, Spring Boot để phát triển backend theo kiến trúc RESTful API và PostgreSQL để lưu trữ dữ liệu. Hệ thống cũng tích hợp JWT Authentication nhằm hỗ trợ xác thực và phân quyền người dùng, đồng thời tích hợp VNPAY để hỗ trợ thanh toán trực tuyến.

Ngoài ra, hệ thống còn định hướng tích hợp AI nhằm hỗ trợ phân tích dữ liệu và đề xuất dịch vụ hoặc bác sĩ phù hợp với nhu cầu của bệnh nhân. AI cũng hỗ trợ admin trong việc phân tích dữ liệu hoạt động và thống kê của phòng khám.

2. Đóng góp và ý nghĩa:

Về mặt học thuật, đề tài giúp làm rõ quy trình xây dựng một hệ thống web full-stack sử dụng VueJS, Spring Boot và PostgreSQL. Thông qua đề tài, các kiến thức về RESTful API, JWT Authentication, thiết kế cơ sở dữ liệu và mô hình SPA đã được áp dụng vào một hệ thống thực tế.

Về mặt thực tiễn, hệ thống góp phần hỗ trợ phòng khám nha khoa số hóa các quy trình quản lý truyền thống như đặt lịch khám, quản lý hồ sơ bệnh nhân, kê đơn thuốc, lập kế

hoạch điều trị và thanh toán hóa đơn. Việc quản lý dữ liệu tập trung giúp giảm sai sót trong quá trình vận hành, nâng cao hiệu quả quản lý và cải thiện trải nghiệm cho bệnh nhân.

3. Hạn chế và bài học kinh nghiệm:

Do giới hạn về thời gian và nguồn lực, hệ thống vẫn còn một số hạn chế như chưa phát triển ứng dụng di động, chưa tối ưu dashboard thống kê nâng cao và chưa triển khai đầy đủ các chức năng AI phân tích chuyên sâu. Một số tính năng như đồng bộ lịch cá nhân, nhắc lịch tự động nâng cao hoặc tích hợp bảo hiểm y tế vẫn chưa được thực hiện.

Qua quá trình thực hiện đề tài, em đã hiểu rõ hơn quy trình phân tích yêu cầu, thiết kế cơ sở dữ liệu, xây dựng RESTful API và phát triển hệ thống web full-stack. Đồng thời, em cũng tích lũy thêm kinh nghiệm trong việc xử lý lỗi, kiểm thử hệ thống, thiết kế giao diện người dùng và tổ chức mã nguồn cho hệ thống có nhiều nghiệp vụ liên kết với nhau.

4. Hướng phát triển:

Trong tương lai, hệ thống có thể được phát triển thêm ứng dụng di động Android/iOS nhằm hỗ trợ bệnh nhân đặt lịch và theo dõi lịch khám thuận tiện hơn. Ngoài ra, hệ thống có thể mở rộng thêm các chức năng AI phân tích dữ liệu nâng cao, dashboard thống kê trực quan và tối ưu hiệu suất xử lý dữ liệu.

Bên cạnh đó, hệ thống có thể bổ sung thêm các chức năng chăm sóc bệnh nhân như nhắc lịch tái khám tự động, lưu trữ hình ảnh điều trị, đồng bộ lịch cá nhân và hỗ trợ đánh giá chất lượng dịch vụ nhằm nâng cao trải nghiệm người dùng.

5. Tổng kết:

Đề tài đã hoàn thành các mục tiêu nghiên cứu và xây dựng được một hệ thống quản lý nha khoa có tính ứng dụng thực tế cao. Hệ thống góp phần hỗ trợ phòng khám nâng cao hiệu quả quản lý, giảm thao tác thủ công và tối ưu quy trình vận hành.

Thông qua quá trình thực hiện đề tài, em đã củng cố được kiến thức chuyên môn về phát triển hệ thống web, đồng thời nâng cao kỹ năng phân tích, thiết kế và triển khai một hệ thống phần mềm thực tế. Đây là nền tảng quan trọng giúp em tiếp tục học tập và phát triển trong lĩnh vực công nghệ phần mềm trong tương lai.

TÀI LIỆU THAM KHẢO

- [1] Nguyễn Văn Ba, *Phân tích và thiết kế hệ thống thông tin*, Nhà xuất bản Đại học Quốc gia Hà Nội, 2010.
- [2] Phạm Hữu Khang, *Lập trình Java cơ bản*, Nhà xuất bản Lao động – Xã hội, 2018.
- [3] Nguyễn Thành Nam, *Giáo trình Cơ sở dữ liệu*, Nhà xuất bản Giáo dục Việt Nam, 2019.
- [4] Trần Minh Triết, *Trí tuệ nhân tạo và ứng dụng*, Nhà xuất bản Khoa học và Kỹ thuật, 2021.
- [5] Craig Walls, *Spring Boot in Action*, Manning Publications, 2016.
- [6] Eric Freeman and Elisabeth Robson, *Head First Design Patterns*, O'Reilly Media, 2020.
- [7] Adam Freeman, *Pro Vue.js 3: Full-Stack Web Development with Vue.js and Node.js*, Apress, 2020.
- [8] Vaskaran Sarcar, *Java Design Patterns*, Apress, 2018.
- [9] Martin Kleppmann, *Designing Data-Intensive Applications*, O'Reilly Media, 2017.
- [10] *VueJS Official Documentation*. Truy cập tại:
<https://vuejs.org/>
- [11] *Spring Boot Documentation*. Truy cập tại:
<https://spring.io/projects/spring-boot>
- [12] *PostgreSQL Documentation*. Truy cập tại:
<https://www.postgresql.org/docs/>
- [13] *JWT Introduction*. Truy cập tại:
<https://jwt.io/introduction>
- [14] *MDN Web Docs – JavaScript*. Truy cập tại:
<https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- [15] *RESTful API Tutorial*. Truy cập tại:
<https://restfulapi.net/>