

BỘ GIAO THÔNG VẬN TẢI
TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TP.HCM
VIỆN ĐÀO TẠO CHẤT LƯỢNG CAO

—o0o—



BÀI BÁO CÁO MÔN ĐỒ ÁN CÔNG NGHỆ PHẦN MỀM

Tên đề tài

PHẦN MỀM QUẢN LÝ PHÒNG KHÁM TƯ NHÂN (CMS)

TP. Hồ Chí Minh, tháng 11 / 2025

STT	Họ và tên	Mã số sinh viên	Lớp
1	Hồ Ngọc Gia Hân	079305030460	CN2304CLCB
2	Lê Trần Vũ	052205008073	CN2308CLCB
3	Điền Đình	070205005858	CN2305CLCB

LỜI CẢM ƠN

Để hoàn thành bài báo cáo tổng kết dự án "Phần Mềm Quản Lý Phòng Khám Tư Nhân (Cms)", chúng em xin bày tỏ lòng biết ơn chân thành đến Thầy Lê Văn Quốc Anh. Trong suốt quá trình thực hiện, Thầy đã tận tình hướng dẫn, đóng góp nhiều ý kiến quý báu và giúp chúng em giải quyết các vấn đề gặp phải, từ đó hoàn thiện đề tài một cách tốt nhất. Do kiến thức và kinh nghiệm còn hạn chế, bài báo cáo không thể tránh khỏi những thiếu sót. Chúng em kính mong nhận được sự góp ý của Thầy.

Chúng em xin chân thành cảm ơn!

MỤC LỤC

LỜI MỞ ĐẦU	1
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI	2
1.1. Đặt vấn đề.....	2
1.2. Mục tiêu của đề tài.....	2
1.2.1. Mục tiêu chung.....	2
1.2.2. Mục tiêu cụ thể.....	3
1.3. Đối tượng và phạm vi nghiên cứu	3
1.3.1. Đối tượng nghiên cứu	3
1.3.2. Phạm vi nghiên cứu	3
1.4. Công nghệ thực hiện	4
CHƯƠNG 2: KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU	5
2.1. Khảo sát hiện trạng.....	5
2.1.1. Mô tả quy trình nghiệp vụ thủ công.....	5
2.1.2. Các vấn đề tồn tại	5
2.1.3. Giải pháp đề xuất.....	5
2.2. Phân tích yêu cầu	6
2.2.1. Xác định tác nhân (Actors).....	6
2.2.2. Các chức năng chính (Use Cases).....	6
2.2.3. Đặc tả Use Case.....	7
2.3. Yêu cầu phi chức năng.....	9
CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	12
3.1. Thiết kế kiến trúc hệ thống.....	12
3.2. Thiết kế dữ liệu (Database).....	13
3.2.1. Sơ đồ Quan hệ Thực thể (ERD - Entity Relationship Diagram)	13
3.2.2. Usecase	14
3.2.3. Thiết kế chi tiết các Bảng (Table Design).....	17

3.3. Thiết kế giao diện	18
3.3.1. Giao diện Đăng nhập	18
3.3.2. Giao diện Tiếp tân	19
3.3.3. Giao diện Admin	20
3.3.4. Giao diện Bác sĩ	21
3.3.5. Giao diện Dược sĩ	21
CHƯƠNG 4: CÀI ĐẶT VÀ TRIỂN KHAI	22
4.1. Môi trường cài đặt	22
4.1.1. Yêu cầu về môi trường	22
4.1.2. Hướng dẫn cài đặt	22
4.2. Cấu trúc dự án	22
4.3. Các chức năng chính	24
KẾT LUẬN	25

DANH MỤC HÌNH ẢNH

<i>Hình 1. Sơ đồ kiến trúc 3 lớp</i>	12
<i>Hình 2. Sơ đồ ERD của hệ thống Quản lý phòng khám</i>	14
<i>Hình 3. Admin use-case diagram</i>	15
<i>Hình 5. Tiếp tân use-case diagram</i>	16
<i>Hình 6. Dược sĩ use-case diagram</i>	17
<i>Hình 6. Giao diện Đăng nhập</i>	19
<i>Hình 7. Giao diện Tiếp tân</i>	20
<i>Hình 8. Giao diện Admin</i>	20
<i>Hình 9. Giao diện Bác sĩ</i>	21
<i>Hình 10. Giao diện Dược sĩ</i>	21
<i>Hình 10. Đoạn code tạo tài khoản mẫu</i>	24

DANH MỤC BẢNG BIỂU

<i>Bảng 1. Đặc tả Use Case Đăng nhập</i>	8
<i>Bảng 2. Đặc tả Use Case Khám bệnh và Kê đơn thuốc</i>	9
<i>Bảng 3. Bảng Users (Người dùng hệ thống)</i>	17
<i>Bảng 4. Bảng Patients (Hồ sơ bệnh nhân)</i>	18
<i>Bảng 5. Bảng Medicines (Kho thuốc)</i>	18
<i>Bảng 6. Bảng Medical_Records (Phiếu khám bệnh)</i>	18

LỜI MỞ ĐẦU

Trong bối cảnh xã hội hiện đại, nhu cầu chăm sóc sức khỏe ngày càng tăng cao, dẫn đến sự phát triển mạnh mẽ của hệ thống các phòng khám tư nhân nhằm giảm tải áp lực cho các bệnh viện công. Tại các cơ sở này, sự linh hoạt và chất lượng phục vụ là yếu tố then chốt để thu hút bệnh nhân. Tuy nhiên, thực tế cho thấy nhiều phòng khám quy mô nhỏ và vừa vẫn đang duy trì phương thức quản lý thủ công dựa trên sổ sách giấy tờ.

Phương pháp truyền thống này tuy đơn giản nhưng lại bộc lộ nhiều hạn chế lớn như: tốn thời gian trong khâu tiếp đón và tìm kiếm hồ sơ cũ, dễ xảy ra sai sót khi kê đơn thuốc bằng tay, và đặc biệt là khó khăn cho chủ phòng khám trong việc thống kê doanh thu hay kiểm soát lượng thuốc tồn kho một cách chính xác.

Xuất phát từ nhu cầu thực tiễn đó, việc xây dựng một "Phần mềm Quản lý Phòng khám (Clinic Management System)" tinh gọn là vô cùng cần thiết. Hệ thống được thiết kế để tin học hóa các quy trình nghiệp vụ cốt lõi phù hợp với mô hình 3 tác nhân: Lễ tân (tiếp đón), Bác sĩ (khám bệnh, kê đơn) và Quản trị viên (quản lý thuốc, nhân sự). Giải pháp này không chỉ giúp lưu trữ thông tin bệnh nhân một cách khoa học, bảo mật mà còn hỗ trợ bác sĩ ra quyết định nhanh chóng nhờ các công cụ tra cứu và in ấn tự động.

Bài báo cáo này sẽ tổng kết lại toàn bộ quá trình từ khảo sát yêu cầu, phân tích thiết kế hệ thống, đến quy trình cài đặt và kiểm thử sản phẩm, đồng thời trình bày những kết quả đạt được và phương hướng phát triển trong tương lai.

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Đặt vấn đề

Trong xã hội hiện đại, bên cạnh hệ thống bệnh viện công, các phòng khám tư nhân đóng vai trò quan trọng trong việc giảm tải áp lực y tế và cung cấp dịch vụ khám chữa bệnh linh hoạt cho người dân. Tuy nhiên, đặc thù của các phòng khám quy mô nhỏ là nguồn nhân lực mỏng, thường chỉ gồm một vài bác sĩ và nhân viên hỗ trợ.

Hiện nay, nhiều phòng khám tư vẫn duy trì phương thức quản lý thủ công truyền thống (sổ sách, Excel rời rạc). Điều này dẫn đến những bất cập lớn:

- **Thất lạc thông tin:** Hồ sơ bệnh án giấy dễ bị hư hỏng, khó tìm kiếm lại lịch sử khám của bệnh nhân cũ, làm giảm chất lượng chẩn đoán.
- **Quy trình chồng chéo:** Việc xếp hàng đợi, gọi tên và điều phối bệnh nhân vào phòng khám thường gây lộn xộn vào giờ cao điểm.
- **Khó kiểm soát kho thuốc:** Chủ phòng khám khó nắm bắt chính xác số lượng thuốc tồn kho để bổ sung kịp thời, dẫn đến thất thoát hoặc thiếu thuốc điều trị.
- **Hạn chế trong hỗ trợ:** Các bác sĩ thiếu các công cụ tra cứu nhanh về tương tác thuốc hoặc thông tin y khoa ngay trên phần mềm quản lý.

Từ những thực trạng trên, việc phát triển một "Phần mềm Quản lý Phòng khám (Clinic Management System)" nhỏ gọn, chạy trên nền tảng Desktop với cơ sở dữ liệu nội bộ là giải pháp tối ưu. Nó giúp chuẩn hóa quy trình làm việc, tiết kiệm thời gian và nâng cao sự chuyên nghiệp cho phòng khám.

1.2. Mục tiêu của đề tài

Dựa trên nhu cầu thực tế và khả năng công nghệ, đề tài hướng tới các mục tiêu sau:

1.2.1. Mục tiêu chung

Xây dựng một ứng dụng Desktop hoàn chỉnh bằng ngôn ngữ Python, phục vụ công tác quản lý toàn diện cho một phòng khám tư nhân quy mô vừa và

nhỏ. Hệ thống đảm bảo tính ổn định, giao diện thân thiện và dễ dàng triển khai cài đặt.

1.2.2. Mục tiêu cụ thể

- Quản lý hồ sơ bệnh nhân: Lưu trữ tập trung thông tin hành chính và lịch sử khám bệnh trong cơ sở dữ liệu số (SQLite), cho phép tra cứu tức thì.
- Tin học hóa quy trình khám: Hỗ trợ quy trình khép kín: Lễ tân tiếp nhận -> Bác sĩ khám/kê đơn -> Xuất phiếu và hóa đơn.
- Quản lý kho dược: Theo dõi danh mục thuốc, tự động trừ tồn kho khi bác sĩ kê đơn.
- Tích hợp công nghệ AI: Ứng dụng mô hình ngôn ngữ lớn (thông qua OpenAI API) để xây dựng trợ lý ảo (Chatbot), hỗ trợ bác sĩ tra cứu thông tin nhanh chóng.
- Chức năng in ấn: Tự động xuất phiếu khám và đơn thuốc ra định dạng PDF để in ấn cho bệnh nhân.

1.3. Đối tượng và phạm vi nghiên cứu

1.3.1. Đối tượng nghiên cứu

- Quy trình nghiệp vụ tại phòng khám tư nhân: Tiếp đón, Khám lâm sàng, Kê đơn, Cấp phát thuốc.
- Các công nghệ lập trình ứng dụng Desktop với Python.

1.3.2. Phạm vi nghiên cứu

Về nghiệp vụ: Hệ thống tập trung vào các chức năng cốt lõi cho phòng khám ngoại trú. Không bao gồm các nghiệp vụ phức tạp của bệnh viện lớn như: Quản lý nội trú (giường bệnh), Phẫu thuật, hay Quản lý bảo hiểm y tế chuyên sâu.

Về người dùng (Tác nhân): Hệ thống phân quyền cho 3 đối tượng chính:

- Quản trị viên (Admin): Quản lý tài khoản người dùng, quản lý danh mục thuốc và cài đặt hệ thống.
- Bác sĩ (Doctor): Xem hàng đợi, thực hiện khám, chỉ định dịch vụ, kê đơn thuốc và sử dụng trợ lý AI.

- Lễ tân (Receptionist): Đăng ký thông tin bệnh nhân, đặt lịch khám, điều phối hàng đợi và thu phí.

1.4. Công nghệ thực hiện

- Ngôn ngữ lập trình: Python (Phiên bản 3.12). Đây là ngôn ngữ mạnh mẽ, cú pháp rõ ràng, hỗ trợ tốt cho việc phát triển nhanh ứng dụng.
- Giao diện người dùng (GUI): Sử dụng thư viện Tkinter tích hợp sẵn trong Python để xây dựng giao diện Desktop trực quan.
- Cơ sở dữ liệu: SQLite (data/clinic.db). Đây là hệ quản trị cơ sở dữ liệu quan hệ dạng file, không cần cài đặt máy chủ (Server), rất phù hợp cho ứng dụng quy mô nhỏ, dễ dàng sao chép và backup.
- Tích hợp AI: Sử dụng thư viện openai để kết nối API, tạo tính năng Chatbot tư vấn y khoa thông minh.
- Xử lý văn bản: Sử dụng thư viện hỗ trợ tạo file PDF (như ReportLab hoặc FPDF) để xuất phiếu khám và đơn thuốc (output/*.pdf).

CHƯƠNG 2: KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU

2.1. Khảo sát hiện trạng

2.1.1. Mô tả quy trình nghiệp vụ thủ công

Tại các phòng khám tư nhân quy mô vừa và nhỏ chưa ứng dụng phần mềm, quy trình khám chữa bệnh thường diễn ra theo trình tự sau:

Tiếp đón: Bệnh nhân đến quầy lễ tân. Nhân viên ghi chép thông tin hành chính (tên, tuổi, địa chỉ) vào sổ tay hoặc nhập vào file Excel rời rạc.

Xếp hàng: Bệnh nhân ngồi chờ và được gọi tên bằng miệng để vào phòng khám. Việc quản lý thứ tự khám hoàn toàn thủ công.

Khám bệnh & Kê đơn: Bác sĩ khám lâm sàng, viết chẩn đoán và chỉ định thuốc vào sổ khám bệnh giấy hoặc toa thuốc viết tay.

Cấp thuốc/Thanh toán: Bệnh nhân mang toa thuốc ra quầy. Nhân viên (hoặc dược sĩ) nhìn toa thuốc để lấy thuốc, tính tiền bằng máy tính bỏ túi và thu phí.

2.1.2. Các vấn đề tồn tại

Phương thức quản lý thủ công này bộc lộ nhiều nhược điểm lớn:

- **Tốn thời gian:** Lễ tân mất nhiều thời gian để tìm lại thông tin bệnh nhân cũ trong chồng sổ sách.
- **Sai sót trong kê đơn:** Chữ viết tay của bác sĩ đôi khi khó đọc, hoặc việc tính toán liều lượng, kiểm tra tương tác thuốc hoàn toàn dựa vào trí nhớ, dễ gây nhầm lẫn.
- **Khó kiểm soát kho dược:** Không thể trừ kho tự động ngay khi kê đơn, dẫn đến tình trạng thuốc hết mà không biết hoặc thất thoát hàng hóa.
- **Thiếu báo cáo:** Chủ phòng khám không thể xem ngay lập tức doanh thu trong ngày hoặc lịch sử khám của một bệnh nhân cụ thể.

2.1.3. Giải pháp đề xuất

Xây dựng phần mềm Quản lý Phòng khám (CMS) chạy trên máy tính (Desktop App) với cơ sở dữ liệu tập trung. Hệ thống sẽ số hóa toàn bộ quy trình

từ lúc bệnh nhân bước vào đến khi nhận thuốc ra về, tích hợp thêm công cụ AI để hỗ trợ bác sĩ.

2.2. Phân tích yêu cầu

2.2.1. Xác định tác nhân (Actors)

Hệ thống sẽ bao gồm các nhóm chức năng chính, tương ứng với các tác nhân (actors) sử dụng:

Tác nhân (Actors):

- Quản trị viên (Admin): Người có quyền cao nhất, chịu trách nhiệm cấu hình hệ thống, quản lý nhân sự và kiểm soát kho dược.
- Nhân viên lễ tân (Receptionist): Người đầu mối tiếp nhận bệnh nhân, kiêm nhiệm vụ thu ngân và điều phối hàng đợi.
- Bác sĩ (Doctor): Người trực tiếp thực hiện chuyên môn khám chữa bệnh, chỉ định dịch vụ và kê đơn thuốc.

2.2.2. Các chức năng chính (Use Cases)

Nhóm chức năng chung:

- Đăng nhập (Xác thực tài khoản từ CSDL).
- Đăng xuất.

Nhóm chức năng của Quản trị viên (Admin):

- Quản lý tài khoản người dùng: Thêm mới, cập nhật, khóa tài khoản bác sĩ/lễ tân, phân quyền truy cập.
- Quản lý kho thuốc & vật tư:
 - Thêm mới danh mục thuốc (Tên, hoạt chất, đơn vị tính).
 - Nhập kho (Cập nhật số lượng, hạn sử dụng).
 - Xem danh sách thuốc tồn kho (để kịp thời bổ sung).
- Báo cáo thống kê: Xem tổng quan lượt khám và doanh thu phòng khám.

Nhóm chức năng của Nhân viên lễ tân (Receptionist):

- Tiếp nhận bệnh nhân:
 - Tra cứu hồ sơ bệnh nhân cũ (theo tên, số điện thoại).
 - Thêm mới hồ sơ hành chính cho bệnh nhân mới.

- Quản lý hàng đợi: Tạo phiếu khám và đẩy bệnh nhân vào danh sách chờ của bác sĩ.
- Đặt lịch hẹn: Tiếp nhận thông tin đặt lịch (qua điện thoại hoặc web).
- Thu ngân: Tính tổng chi phí (phí khám + thuốc + dịch vụ) và xác nhận thanh toán.

Nhóm chức năng của Bác sĩ (Doctor):

- Quản lý khám bệnh:
 - Xem danh sách bệnh nhân đang chờ.
 - Ghi nhận triệu chứng lâm sàng, tiền sử bệnh.
 - Nhập kết luận chẩn đoán.
- Chỉ định & Kê đơn:
 - Chỉ định dịch vụ cận lâm sàng (Siêu âm, xét nghiệm...).
 - Kê đơn thuốc điện tử (Hệ thống tự động trừ tồn kho).
- Tiềm ích hỗ trợ:
 - Sử dụng trợ lý AI (AI Chat): Tra cứu nhanh thông tin thuốc, tương tác thuốc hoặc phác đồ điều trị thông qua cửa sổ chat thông minh.
 - In ấn: Xuất phiếu khám và đơn thuốc ra file PDF để in cho bệnh nhân.

2.2.3. Đặc tả Use Case

Tên Use Case	Đăng nhập hệ thống
Tác nhân	Admin, Bác sĩ, Lễ tân
Mục đích	Cho phép người dùng truy cập vào hệ thống bằng tài khoản đã được cấp.
Điều kiện	Người dùng đã có tài khoản được Admin tạo trước trong cơ sở dữ liệu (clinic.db).

Cách hoạt động	<p>Người dùng mở phần mềm (file login.py).</p> <p>Nhập Tên đăng nhập và Mật khẩu.</p> <p>Nhấn nút "Đăng nhập".</p> <p>Hệ thống mã hóa mật khẩu và đối chiếu với bảng users trong CSDL.</p> <p>Nếu hợp lệ, hệ thống đóng form đăng nhập và mở giao diện chính tương ứng với quyền (Admin Panel, Giao diện Khám, hoặc Giao diện Lễ tân).</p>
----------------	--

Bảng 1. Đặc tả Use Case Đăng nhập

Tên Use Case	Khám bệnh và Kê đơn
Tác nhân	Bác sĩ
Mục đích	Ghi nhận kết quả khám lâm sàng, chỉ định thuốc điều trị và xuất hồ sơ cho bệnh nhân.
Điều kiện	Bác sĩ đã đăng nhập và có bệnh nhân đang chờ trong hàng đợi (được Lễ tân chuyển qua).
Cách hoạt động	<p>Tại giao diện bác sĩ, chọn bệnh nhân từ "Danh sách chờ khám".</p> <p>Nhập thông tin: Triệu chứng, Tiền sử bệnh, Chẩn đoán (forms/tao_phieu_kham.py).</p>

	<p>Chuyển sang tab "Kê đơn thuốc" (forms/don_thuoc.py).</p> <p>Tìm kiếm thuốc trong kho. Hệ thống hiển thị số lượng tồn khả dụng.</p> <p>Chọn thuốc, nhập số lượng và liều dùng (Sáng/Trưa/Chiều).</p> <p>Nhấn nút "Lưu và In".</p> <p>Hệ thống tự động trừ số lượng thuốc trong bảng medicines.</p> <p>Hệ thống gọi module print_phieu_kham.py để tạo file PDF kết quả.</p>
Kết quả	<p>Hồ sơ bệnh án được lưu vào CSDL.</p> <p>Kho thuốc được cập nhật giảm số lượng.</p> <p>File PDF phiếu khám được tạo thành công trong thư mục output/.</p>

Bảng 2. Đặc tả Use Case Khám bệnh và Kê đơn thuốc

2.3. Yêu cầu phi chức năng

- Bảo mật (Security)
 - Xác thực và Phân quyền: Hệ thống phải phân quyền truy cập chặt chẽ dựa trên vai trò của người dùng (Role-based Access Control). Cụ thể: Lễ tân không được phép truy cập vào chức năng quản lý kho thuốc hay cấu hình hệ thống của Admin, bác sĩ chỉ thực hiện các nghiệp vụ chuyên môn.
 - An toàn dữ liệu: Mật khẩu của người dùng khi lưu trữ trong cơ sở dữ liệu (clinic.db) phải được mã hóa (Hashing) một chiều, không lưu dưới dạng văn bản thuần để tránh bị lộ khi mất file dữ liệu.

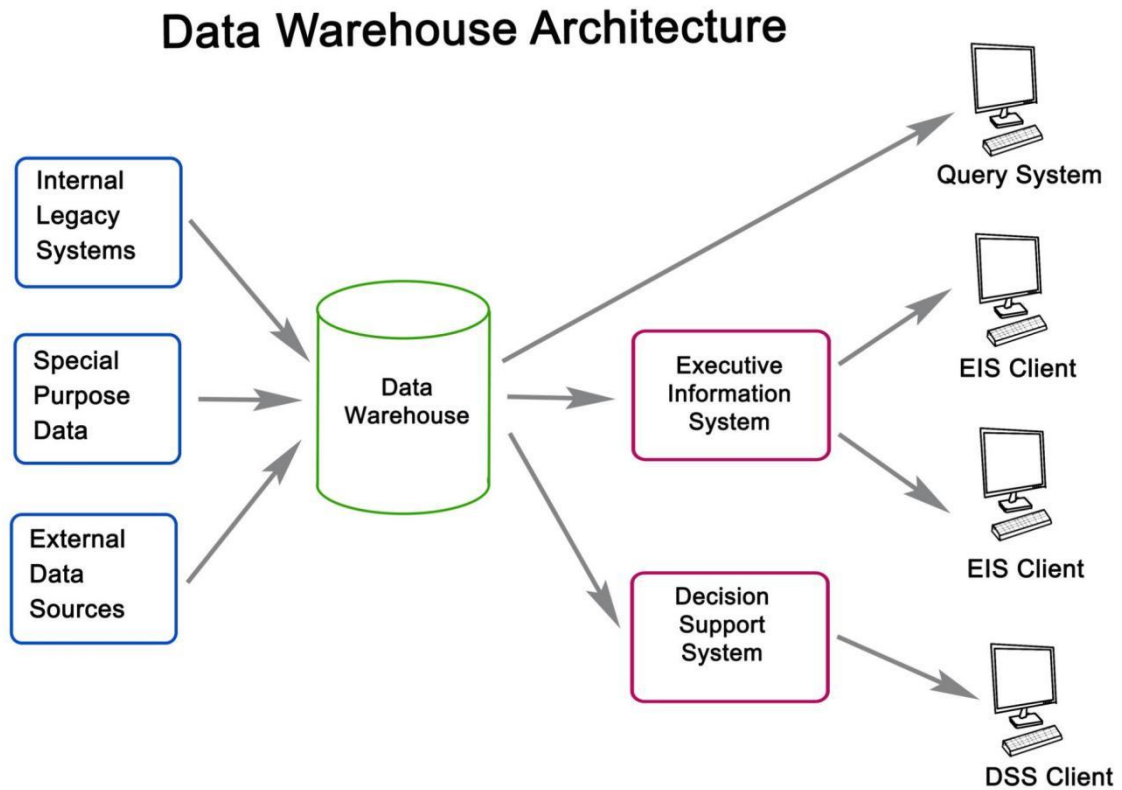
- Bảo mật hồ sơ: Thông tin cá nhân và lịch sử khám của bệnh nhân được lưu trữ cục bộ, hạn chế rủi ro bị tấn công qua mạng so với các hệ thống web công khai.
- Hiệu năng (Performance):
 - Tốc độ phản hồi: Vì sử dụng cơ sở dữ liệu SQLite cục bộ, các thao tác cơ bản như tìm kiếm bệnh nhân, lưu phiếu khám, tải danh sách thuốc phải được thực hiện tức thì, với thời gian phản hồi dưới 2 giây.
 - Khởi động: Ứng dụng phải khởi động nhanh chóng, tài nguyên chiếm dụng (RAM, CPU) ở mức thấp, phù hợp với các máy tính văn phòng cấu hình trung bình (Core i3, RAM 4GB) thường dùng tại các phòng khám.
 - Kết nối API: Chức năng AI Chat phụ thuộc vào tốc độ Internet, nhưng hệ thống phải xử lý bất đồng bộ để giao diện không bị "đơ" (treo) trong lúc chờ phản hồi từ máy chủ OpenAI.
- Tính Khả dụng (Usability):
 - Giao diện thân thiện: Giao diện người dùng (GUI) được thiết kế bằng thư viện Tkinter với bố cục rõ ràng, sử dụng phông chữ tiếng Việt chuẩn (Unicode), giúp nhân viên không rành công nghệ cũng có thể sử dụng dễ dàng sau thời gian ngắn đào tạo.
 - Thông báo rõ ràng: Hệ thống phải có các thông báo hướng dẫn cụ thể khi thao tác thành công hoặc cảnh báo lỗi (ví dụ: "Thuốc trong kho không đủ", "Lưu phiếu thành công").
 - Quy trình tối ưu: Giảm thiểu số lượng thao tác chuột và bàn phím cần thiết để hoàn thành một phiếu khám.
- Độ Tin cậy (Reliability):
 - Hoạt động ổn định: Phần mềm phải hoạt động liên tục trong giờ làm việc mà không bị lỗi thoát đột ngột (crash).

- Toàn vẹn dữ liệu: Cơ chế Transaction của SQLite phải đảm bảo dữ liệu không bị sai lệch khi có sự cố (ví dụ: mất điện khi đang lưu phiếu).
- Sao lưu (Backup): Vì dữ liệu nằm gọn trong file data/clinic.db, hệ thống cho phép sao lưu dễ dàng bằng cách copy file này sang thiết bị lưu trữ khác.
- Khả năng Bảo trì và Cài đặt (Maintainability & Portability):
 - Dễ dàng cài đặt: Phần mềm được đóng gói gọn nhẹ, có thể chạy ngay sau khi cài đặt môi trường Python và các thư viện phụ thuộc (được liệt kê trong requirements.txt) mà không cần cài đặt các hệ quản trị CSDL phức tạp như SQL Server hay MySQL.
 - Cấu trúc mã nguồn: Mã nguồn được tổ chức theo module (tách biệt thư mục forms, data, scripts) giúp dễ dàng nâng cấp hoặc sửa lỗi sau này.

CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1. Thiết kế kiến trúc hệ thống

Để đảm bảo tính linh hoạt, dễ bảo trì và mở rộng, dự án "Hệ thống Quản lý Bệnh viện (HMS)" được thiết kế theo mô hình kiến trúc 3 lớp (3-Tier Architecture). Mô hình này tách biệt rõ ràng các thành phần logic của ứng dụng.



Hình 1. Sơ đồ kiến trúc 3 lớp

Cấu trúc 3 lớp của hệ thống bao gồm:

- Tầng Giao diện (Presentation Layer - GUI/UI):
 - Đây là lớp tương tác trực tiếp với người dùng (Admin, Bác sĩ, Lễ tân).
 - Được xây dựng bằng thư viện đồ họa của Python (như Tkinter/PyQt) nằm trong thư mục forms/.

- Bao gồm các thành phần chính: Màn hình đăng nhập (login.py), Bảng điều khiển Admin (admin_panel.py), Form khám bệnh (tao_phieu_kham.py), Cửa sổ Chat AI (ai_chat.py).
- Nhiệm vụ: Nhận dữ liệu nhập từ bàn phím/chuột và hiển thị kết quả xử lý lên màn hình.
- Tầng Nghiệp vụ (Business Logic Layer - BLL):
 - Xử lý các quy tắc nghiệp vụ cốt lõi của phòng khám.
 - Được cài đặt trong các file như main_app.py và các logic xử lý sự kiện trong forms/.
 - Chức năng: Kiểm tra tồn kho thuốc trước khi kê đơn, tính toán tổng tiền viện phí, xử lý gửi yêu cầu đến OpenAI API, điều phối hàng đợi khám bệnh.
- Tầng Truy cập Dữ liệu (Data Access Layer - DAL):
 - Chịu trách nhiệm giao tiếp trực tiếp với cơ sở dữ liệu SQLite (data/clinic.db).
 - Được cài đặt tập trung tại file database.py.
 - Nhiệm vụ: Thực hiện các câu lệnh SQL (SELECT, INSERT, UPDATE, DELETE) để lưu trữ hồ sơ bệnh án, cập nhật kho thuốc và xác thực tài khoản.

3.2. Thiết kế dữ liệu (Database)

3.2.1. Sơ đồ Quan hệ Thực thể (ERD - Entity Relationship Diagram)

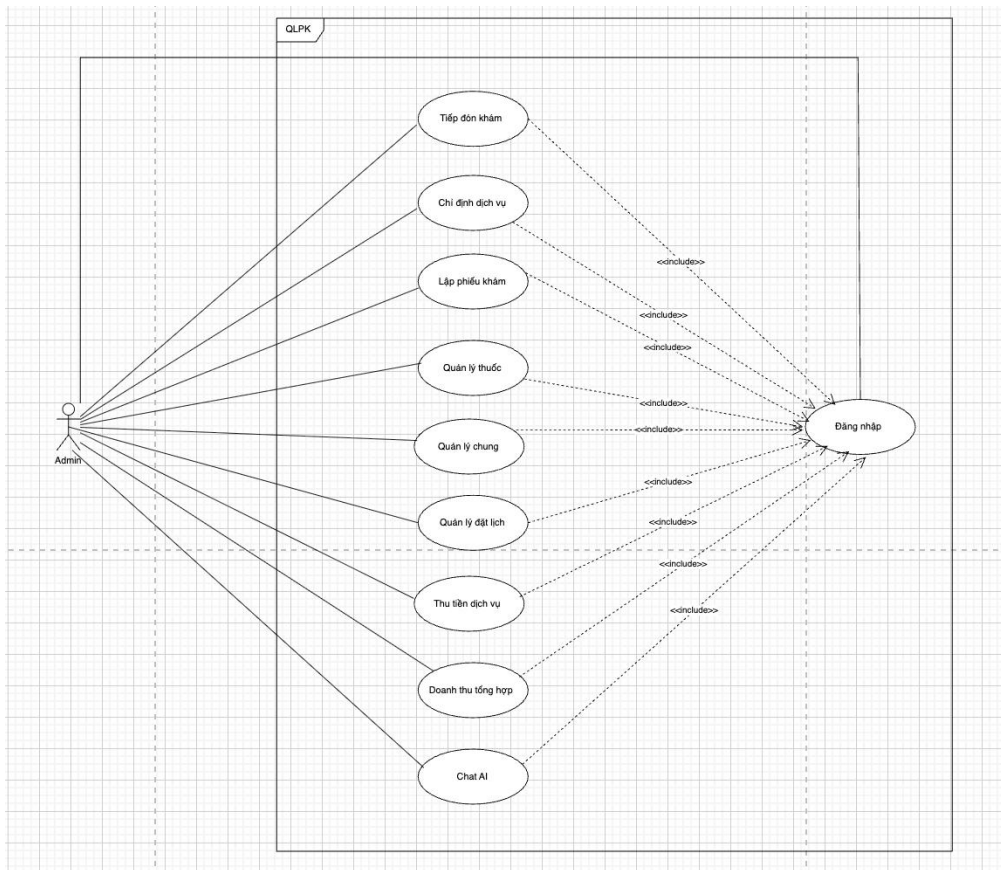
Sơ đồ ERD mô tả các thực thể chính trong hệ thống và mối quan hệ (1-1, 1-nhiều, nhiều-nhiều) giữa chúng.

- TaiKhoan
- VaiTro (Quyền: Admin, Bác sĩ, Lễ tân, Dược sĩ)
- BenhNhan
- PhieuKham (Thực thể trung tâm, liên kết Bệnh nhân và Bác sĩ)
- LichHen
- DonThuoc

- [illegible]

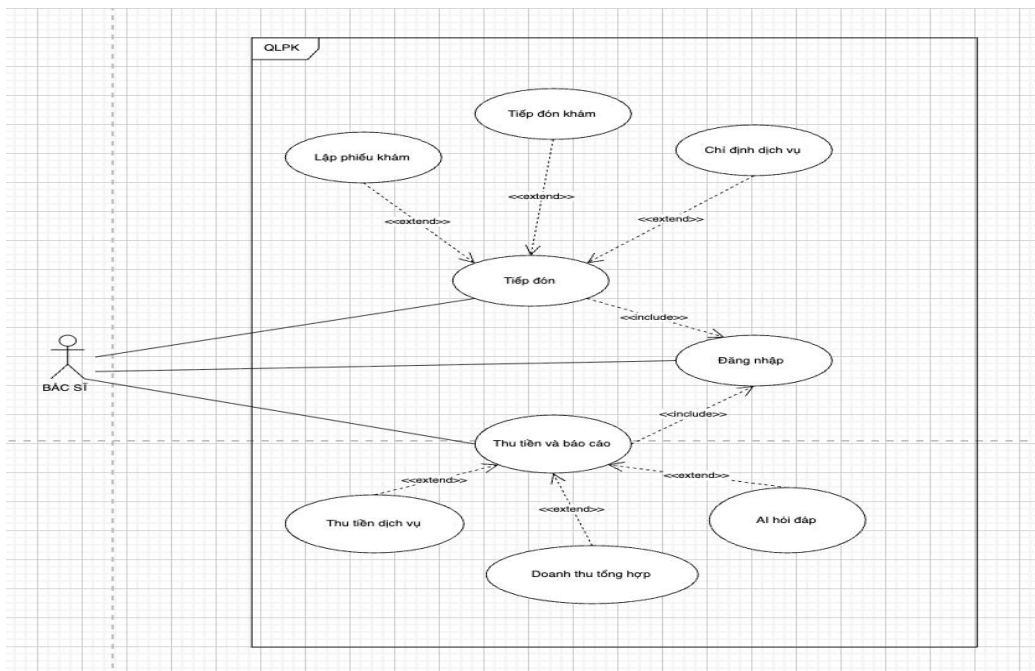
3.2.2. Usecase

14



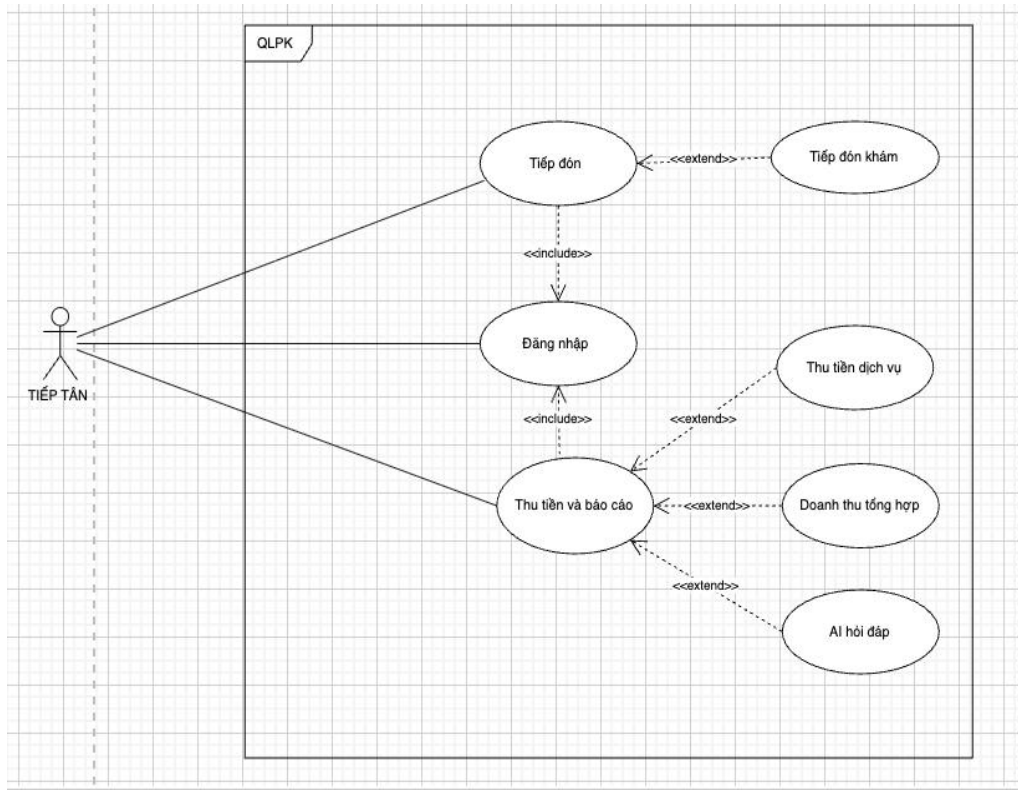
Hình 3. Admin use-case diagram

Biểu đồ use-case thể hiện sự thao tác của actor Bác Sĩ với hệ thống:



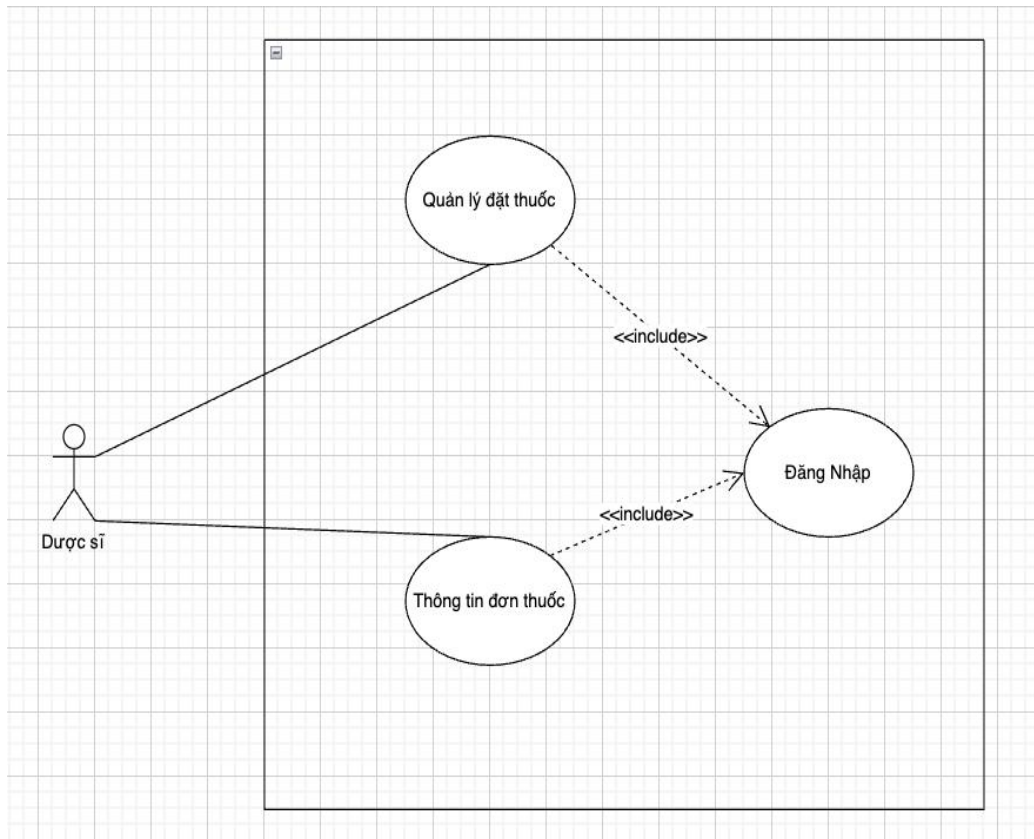
Hình 4. Bác sĩ use-case diagram

Biểu đồ use-case thể hiện sự thao tác của actor Tiếp Tân với hệ thống:



Hình 5. Tiếp tân use-case diagram

Biểu đồ use-case thể hiện sự thao tác của actor Dược Sĩ với hệ thống:



Hình 6. Dược sĩ use-case diagram

3.2.3. Thiết kế chi tiết các Bảng (Table Design)

Dưới đây là thiết kế chi tiết các bảng dữ liệu chính trong file clinic.db:

Tên cột	Kiểu dữ liệu	Mô tả
id	integer (PK)	Mã người dùng tự tăng
username	text	Tên đăng nhập (duy nhất)
password_hash	text	Mật khẩu đã mã hóa
role	text	Vai trò (admin, doctor, receptionist)
full_name	text	Họ tên người dùng

Bảng 3. Bảng Users (Người dùng hệ thống)

Tên cột	Kiểu dữ liệu	Mô tả
id	integer (PK)	Mã người dùng tự tăng
name	text	Họ và tên bệnh nhân

gender	text	Giới tính
dob	text	Ngày tháng năm sinh
phone	text	Số điện thoại liên hệ
address	text	Địa chỉ

Bảng 4. Bảng Patients (Hồ sơ bệnh nhân)

Tên cột	Kiểu dữ liệu	Mô tả
id	integer (PK)	Mã thuốc tự tăng
name	text	Tên thuốc
unit	text	Đơn vị tính (Viên, Vi, Chai)
price	text	Đơn giá bán
stock_qty	integer	Số lượng tồn kho hiện tại

Bảng 5. Bảng Medicines (Kho thuốc)

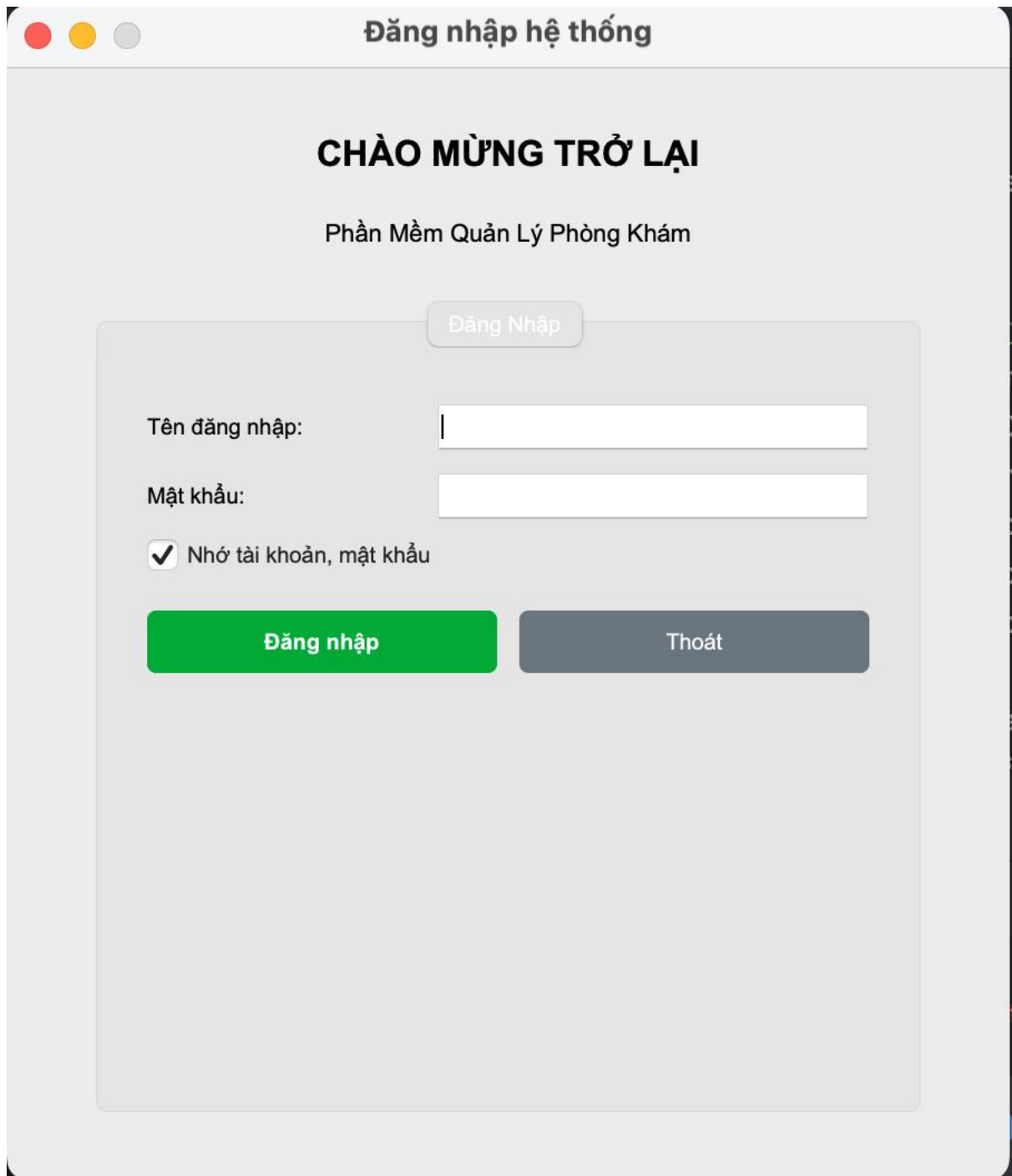
Tên cột	Kiểu dữ liệu	Mô tả
id	integer (PK)	Mã phiếu khám tự tăng
patient_id	integer (PK)	Liên kết bảng Patients
doctor_id	integer (PK)	Liên kết bảng Users (Bác sĩ khám)
visit_date	datetime	Ngày giờ khám
symptoms	text	Triệu chứng lâm sàng
diagnosis	text	Chẩn đoán bệnh
total_fee	real	Tổng chi phí khám và thuốc

Bảng 6. Bảng Medical_Records (Phiếu khám bệnh)

3.3. Thiết kế giao diện

Phần này trình bày một số thiết kế (mockup) cho các giao diện chính của hệ thống, giúp hình dung rõ hơn về luồng tương tác của người dùng.

3.3.1. Giao diện Đăng nhập



Đăng nhập hệ thống

CHÀO MỪNG TRỞ LẠI

Phần Mềm Quản Lý Phòng Khám

Đăng Nhập

Tên đăng nhập:

Mật khẩu:

☒ Nhớ tài khoản, mật khẩu

Đăng nhập Thoát

Hình 6. Giao diện Đăng nhập

3.3.2. Giao diện Tiếp tân

3.3.4. Giao diện Bác sĩ

bacsi

Vai trò: bác_sĩ

TIẾP ĐÓN

Tiếp đón khám

Quản lý đặt lịch

Chỉ định dịch vụ

Lập phiếu khám

AI Hỏi đáp

THÔNG TIN BỆNH NHÂN

Họ và tên

Giới tính

Chọn giới tính

Ngày sinh

14/12/2025

Tuổi

Địa chỉ

Điện thoại

Đối tượng

-Chọn đối tượng-

Nghề nghiệp

Người giới thiệu

-Chọn người giới thiệu-

Loại khám

-Nhấn để chọn-

Số CCCD

THÔNG TIN TIẾP ĐÓN BAN ĐẦU

Số hồ sơ

HS005

Tình trạng BN

-Nhấn để chọn-

Bác sĩ khám

-Nhấn để chọn-

Phòng khám

-Nhấn để chọn-

Ngày lập

14/12/2025

Nhịp thở (lần/phút)

Nhịp tim (bpm)

Huyết áp (mmHg)

Nhiệt độ (°C)

Chiều cao (cm)

Cân nặng (kg)

DANH SÁCH PHIẾU TIẾP ĐÓN KCB

Số hồ sơ	Ngày lập	Phòng khám	Họ tên BN	Bác sĩ khám	Tình trạng
HS002	2025-12-11	-Nhấn để chọn-	ãs	-Nhấn để chọn-	-Nhấn để chọn-
HS003	2025-12-11	-Nhấn để chọn-	q	-Nhấn để chọn-	-Nhấn để chọn-
HS004	2025-12-11	-Nhấn để chọn-	lê trần vũ	-Nhấn để chọn-	-Nhấn để chọn-
HS001	2025-12-10	Phòng Tai - Mũi - Họng	lê vũ	BS. Trần Thị B	Nặng

Nhập mới

Lưu

Sửa

Xóa

In phiếu

In STT

Tải lại

Hình 9. Giao diện Bác sĩ

3.3.5. Giao diện Dược sĩ

duocsĩ

Vai trò: duocsĩ

TIẾP ĐÓN

Quản lý đặt lịch

Thông tin đơn thuốc

AI Hỏi đáp

QUẢN LÝ XUẤT THUỐC

Chọn bệnh nhân:

Tải đơn

Đơn Thuốc Thường

Đơn Thuốc Bổ Sung

Ngày kê	Bác sĩ	Tổng tiền	Trạng thái
---------	--------	-----------	------------

Danh mục tất cả thuốc (Tồn kho sẽ cập nhật khi xuất):

Tên thuốc	Mã thuốc	Đơn vị	Tồn kho	Ghi chú
-----------	----------	--------	---------	---------

Ghi chú:

Xem chi tiết

Xuất thuốc

Hình 10. Giao diện Dược sĩ

CHƯƠNG 4: CÀI ĐẶT VÀ TRIỂN KHAI

4.1. Môi trường cài đặt

4.1.1. Yêu cầu về môi trường

Do ứng dụng được xây dựng trên nền tảng Desktop với cơ sở dữ liệu SQLite gọn nhẹ, yêu cầu phần cứng không quá khắt khe, phù hợp với các máy tính văn phòng thông thường tại phòng khám:

- Vi xử lý (CPU): Intel Core i3 (thế hệ 8 trở lên) hoặc tương đương (Apple M1/M2).
- Bộ nhớ (RAM): Tối thiểu 4GB (Khuyến nghị 8GB để chạy mượt mà các tác vụ in ấn và AI).
- Ổ cứng: Dung lượng trống tối thiểu 500MB (để lưu trữ mã nguồn, thư viện và file cơ sở dữ liệu clinic.db).

4.1.2. Hướng dẫn cài đặt

Hệ thống được cài đặt và vận hành theo các bước cơ bản sau:

- Tải mã nguồn: Sao chép (clone) kho lưu trữ của dự án từ Github về máy trạm:

git clone: <https://github.com/LeTranVu00/CNPM.git>

- Tạo môi trường ảo: (Khuyến nghị) Tạo một môi trường ảo Python để quản lý các thư viện:

```
python -m venv venv
```

- Kích hoạt môi trường ảo:
 - Trên Windows: venv\Scripts\activate
 - Trên macOS/Linux: source venv/bin/activate
- Cài đặt thư viện: Cài đặt tất cả các thư viện phụ thuộc từ tệp requirements.txt:

```
pip install -r requirements.txt
```

- Khởi chạy: Chạy tệp Python chính để khởi động ứng dụng:

```
python main.py
```

4.2. Cấu trúc dự án

Mã nguồn chương trình được tổ chức theo hướng module hóa, tách biệt giữa giao diện (Forms), dữ liệu (Data) và logic xử lý (Scripts). Cấu trúc thư mục cụ thể như sau:

- Thư mục gốc (Root):
 - main.py: Tập tin khởi chạy chính của chương trình.
 - login.py: Module xử lý đăng nhập và xác thực người dùng.
 - database.py: Lớp xử lý kết nối CSDL SQLite, chứa các hàm create_connection, execute_query.
 - main_app.py: Điều phối luồng chương trình chính sau khi đăng nhập.
 - app_signals.py / signals.py: Quản lý các tín hiệu (signals) để giao tiếp giữa các form.
- Thư mục forms/ (Giao diện người dùng):
 - admin_panel.py: Giao diện dành cho Quản trị viên (Quản lý User, Cấu hình).
 - tiep_don_kham.py: Giao diện Lễ tân (Tiếp nhận bệnh nhân, xếp hàng đợi).
 - tao_phieu_kham.py: Giao diện chính của Bác sĩ (Khám lâm sàng).
 - don_thuoc.py & quan_ly_thuoc.py: Giao diện kê đơn và quản lý kho thuốc.
 - ai_chat.py: Giao diện Chatbot hỗ trợ y khoa (tích hợp OpenAI).
 - print_phieu_kham.py: Module xử lý logic dàn trang và xuất file PDF.
- Thư mục data/:
 - clinic.db: File cơ sở dữ liệu SQLite chứa toàn bộ bảng (Users, Patients, Medicines...).
- Thư mục output/:
 - Nơi lưu trữ các file PDF phiếu khám và đơn thuốc sau khi xuất (Ví dụ: phieu_kham_PK0001_20251123.pdf).

4.3. Các chức năng chính

4.3.1. Chức năng Đăng nhập (login.py)

- Giao diện đầu tiên xuất hiện khi chạy chương trình.
- Người dùng nhập Username/Password: thông tin đăng nhập sẽ do admin tạo và quản lý.

```
4 from database import create_user, get_connection
5
6 def ensure_user(username, password, role):
7     try:
8         created = create_user(username, password, role=role)
9         if created:
10             print(f"Created user: {username} with role {role}")
11         else:
12             print(f"User {username} not created (possibly exists)")
13     except Exception as e:
14         print(f"Could not create user {username}: {e}")
15
16 if __name__ == '__main__':
17     # Create sample accounts for testing roles
18     ensure_user('bacsi', 'bacsi', 'bac_si')
19     ensure_user('tieptan', 'tiếp_tan', 'tiếp_tan')
20     create_user('bacsi1', '123456', role='bac_si')
21     print('Đã tạo tài khoản bác sĩ: bacsi1 / 123456')
22     create_user('tieptan1', '123456', role='tieptan')
23     print('Đã tạo tài khoản tiếp tân: tieptan1 / 123456')
24
25     # Print existing users
26     conn = get_connection()
27     cur = conn.cursor()
28     cur.execute('SELECT id, username, role, created_at FROM users ORDER BY id')
29     rows = cur.fetchall()
30     print('\nCurrent users:')
31     for r in rows:
32         print(f"id={r[0]}\tusername={r[1]}\trole={r[2]}\tcreated_at={r[3]}")
33     conn.close()
34
```

Hình 10. Đoạn code tạo tài khoản mẫu

KẾT LUẬN

Qua quá trình nghiên cứu và thực hiện đề tài "Phần mềm Quản lý Phòng khám Tư nhân (CMS)", nhóm chúng em đã hoàn thành việc xây dựng một ứng dụng Desktop hoàn chỉnh, đáp ứng sát sao nhu cầu quản lý thực tế tại các phòng khám quy mô vừa và nhỏ. Kết quả lớn nhất của dự án là đã triển khai thành công quy trình nghiệp vụ khép kín với 3 tác nhân chính: Tiếp đón (Lễ tân) – Khám bệnh/Kê đơn (Bác sĩ) – Quản trị hệ thống (Admin). Hệ thống giúp thay thế hoàn toàn phương thức quản lý sổ sách thủ công, giải quyết triệt để các vấn đề như thất lạc hồ sơ bệnh nhân, sai sót trong tính toán chi phí hay khó khăn trong việc kiểm soát tồn kho thuốc. Về mặt kỹ thuật, nhóm đã vận dụng hiệu quả ngôn ngữ Python và cơ sở dữ liệu SQLite để tạo ra một phần mềm gọn nhẹ, dễ triển khai cài đặt mà không yêu cầu hạ tầng máy chủ phức tạp. Điểm sáng nổi bật của dự án là việc tích hợp thành công Trí tuệ nhân tạo (AI Chatbot) hỗ trợ tra cứu y khoa và chức năng xuất phiếu khám tự động (PDF), giúp nâng cao tính chuyên nghiệp và hiệu suất làm việc của người dùng. Tuy nhiên, do giới hạn về thời gian và đặc thù dự án, phần mềm hiện tại có thể còn một số chỗ chưa hoàn thiện. Nhưng đây sẽ là tiền đề quan trọng để nhóm tiếp tục nghiên cứu, nâng cấp hệ thống lên nền tảng Web hoặc ứng dụng di động trong tương lai, nhằm đáp ứng tốt hơn nhu cầu mở rộng của các phòng khám hiện đại.