

# **Trường Đại học Bách Khoa Hà Nội**

## **Viện Công nghệ thông tin và truyền thông**



### **Đồ án tốt nghiệp:**

### **Hệ thống quản lý bệnh án**

**Vũ Lê Hoàng**

20166163@student.hust.edu.vn

**Ngành Công nghệ thông tin**

**Giảng viên hướng dẫn:** Trần Đình Khang

**Bộ môn:** Hệ thống thông tin

**Viện:** Công nghệ thông tin và truyền thông

---

Chữ ký của GVHD

**HÀ NỘI, 6/2018**

# Lời cam kết

Họ và tên sinh viên : Vũ Lê Hoàng .....  
Điện thoại liên lạc : 0919066123 ..... Email: 20166163@student.hust.edu.vn .  
Lớp : CN – CNTT 2 K61 ..... Hệ đào tạo : Chính quy .....

Tôi – *Vũ Lê Hoàng* – cam kết Đồ án Tốt nghiệp (ĐATN) là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của *PGS. TS. Trần Đình Khang*. Các kết quả nêu trong ĐATN là trung thực, là thành quả của riêng tôi, không sao chép theo bất kỳ công trình nào khác. Tất cả những tham khảo trong ĐATN – bao gồm hình ảnh, bảng biểu, số liệu, và các câu từ trích dẫn – đều được ghi rõ ràng và đầy đủ nguồn gốc trong danh mục tài liệu tham khảo. Tôi xin hoàn toàn chịu trách nhiệm với dù chỉ một sao chép vi phạm quy chế của nhà trường.

*Hà Nội, ngày 24 tháng 06 năm 2020*

Tác giả ĐATN

*Vũ Lê Hoàng*

Xác nhận của giảng viên về mức độ hoàn thành ĐATN và cho phép bảo vệ:

.....  
.....  
.....  
.....

*Hà Nội, ngày      tháng      năm 2020*

Giảng viên hướng dẫn

PGS. TS. Trần Đình Khang

## Lời cảm ơn

Lời đầu tiên, em xin phép được gửi lời cảm ơn chân thành tới thầy Trần Đình Khang – giảng viên Bộ môn Hệ thống thông tin – Viện Công nghệ thông tin và truyền thông – Đại học Bách Khoa Hà Nội. Thầy đã dành thời gian đưa ra những ý kiến rất chính xác về nghiệp vụ hệ thống đối với em, đồng thời thầy vô cùng tận tình và tâm huyết khi giúp đỡ sinh viên tìm hiểu thêm và khai phá dữ liệu về đề tài được giao.

Lời thứ hai, em xin cảm ơn anh Nguyễn Đức Anh, cựu sinh viên Đại học Bách Khoa Hà Nội K53 đã đưa ra những ý kiến đóng góp mang tính định hướng về công nghệ cũng như sự giúp đỡ hết sức nhiệt tình trong việc phản biện trong khóa cạnh người dùng

Em cũng xin cảm ơn các thầy cô và bạn bè trong Viện Công nghệ Thông tin và Truyền thông cũng như trường Đại học Bách Khoa Hà Nội đã trang bị cho em những kiến thức cần thiết để em có thể hoàn thành đồ án.

Do thời gian có hạn và trình độ hiểu biết còn nhiều hạn chế, đồ án này khó có thể tránh khỏi những thiếu sót. Rất mong nhận được sự đóng góp từ các thầy cô để đồ án hoàn thiện hơn.

Em xin chân thành cảm ơn !

# Tóm tắt

Hiện nay, chất lượng và giá cả dịch vụ y tế ngày càng có xu hướng phù hợp với đại đa số bệnh nhân trong nước. Bên cạnh đó, tỉ lệ người già ở Việt Nam đang tăng ở mức cao nhất trong lịch sử đất nước <sup>[1]</sup>. Trong số đó, hơn 90% số lượng người già mắc các bệnh mãn tính <sup>[2]</sup>. Điều đó có nghĩa nhu cầu khám chữa bệnh thường xuyên là vô cùng cần thiết, kéo theo đó là việc quản lý sổ bệnh án bệnh nhân cũng trở nên quan trọng hơn. Tuy nhiên hầu hết các sổ y bạ hay bệnh án hiện tại đều rất khó giữ gìn, dễ hỏng, rách, mất. Vậy nên việc đưa ra một hệ thống quản lý bệnh án là cần thiết

Bên cạnh đó, thế hệ trẻ hiện nay, lực lượng lao động chính trong thời kì dân số vàng lại đa phần mắc các bệnh văn phòng như đau lưng, đau xương khớp, các bệnh về cột sống, các bệnh về tiêu hóa. Nhu cầu về khám chữa bệnh của người lao động trẻ rất lớn. Đồng thời đây là thế hệ tiếp cận nhanh và có niềm hứng thú với công nghệ. Việc áp dụng công nghệ mới sẽ thu hút các bệnh nhân trải nghiệm dịch vụ khám chữa bệnh thuận tiện hơn, đồng thời mang lại chất lượng sức khỏe tốt hơn cho cộng đồng.

Vì vậy em xin chọn đề tài “Hệ thống quản lý bệnh án” để triển khai nhờ tính thực tiễn đề tài nhắm đến. Hệ thống thực hiện trên nền web, gồm 2 phần là client và server riêng. Ứng dụng đã hoàn thành các chức năng cốt lõi, trong lúc sử dụng còn nhiều hạn chế. Trong thời gian tới em sẽ hoàn thiện ứng dụng để đưa ra một giải pháp hoàn thiện hơn cho người dùng.

Treatment Records Management System	
-------------------------------------	--

### Lịch sử sửa đổi

Phiên bản	Sửa đổi	Ngày sửa đổi	Người sửa đổi	Lý do sửa đổi	Người kiến nghị sửa đổi
1.1	Method scope	04.02.2020	Vũ Lê Hoàng	Sai phạm vi phương thức (tất cả phương thức để public)	Vũ Lê Hoàng

# Mục lục

<b>Chương 1: Giới thiệu và đề xuất</b>	8
1.1. Thực trạng	8
1.2. Yêu cầu cần giải quyết	8
1.3. Thông tin cần giải quyết	9
1.4. Đối tượng cá nhân	10
1.5. Tính năng	11
1.6. Công nghệ sử dụng	11
1.7. Kế hoạch thực hiện	11
1.7.1. Phân tích yêu cầu	11
1.7.2. Thiết kế hệ thống	11
1.7.3. Lập trình hệ thống	12
1.7.4. Báo cáo	12
<b>Chương 2: Phân tích và thiết kế hệ thống</b>	13
2.1. Tìm hiểu các hệ thống tương tự	13
2.1.1. Hệ thống quản lý bệnh viện của hệ thống bệnh viện Hồng Ngọc	13
2.1.2. Kết luận	13
2.2. Thiết kế hệ thống	14
2.2.1. Phân tích yêu cầu	14
2.2.2. Phân tích kiến trúc	29
2.2.3. Phân tích chi tiết	40
2.2.4. Thiết kế cơ sở dữ liệu	54
2.3. Thiết kế giao diện	59
<b>Chương 3: Kết luận</b>	64
3.1. Kết luận	64
3.2. Hướng phát triển	65
<b>Tài liệu tham khảo</b>	66
<b>Phụ lục</b>	67

## Mục lục biểu đồ và ảnh

Figure 1: Usecase tổng quan .....	14
Figure 2: Biểu đồ usecase quản lý bệnh án .....	15
Figure 3: Biểu đồ usecase quản lý cuộc hẹn.....	16
Figure 4: Biểu đồ usecase quản lý thuốc .....	16
Figure 5: Biểu đồ usecase quản trị tài khoản.....	17
Figure 6: Kiến trúc hệ thống .....	29
Figure 7: Kiến trúc logic.....	30
Figure 8: Sequence diagram: Thêm mới bệnh án .....	31
Figure 9: Sequence diagram: Sửa bệnh án .....	31
Figure 10: Sequence diagram: Thêm mới cuộc hẹn.....	32
Figure 11: Sequence diagram: Sửa thông tin cuộc hẹn.....	32
Figure 12: Sequence diagram: Kết thúc cuộc hẹn .....	33
Figure 13: Sequence diagram: Thêm mới thuốc.....	33
Figure 14: Sequence diagram: Sửa thông tin thuốc .....	34
Figure 15: Activity diagram: Thêm bệnh án .....	35
Figure 16: Activity diagram: Sửa bệnh án.....	36
Figure 17: Activity diagram: Thêm cuộc hẹn.....	37
Figure 18: Activity diagram: Sửa cuộc hẹn.....	37
Figure 19: Activity diagram: Kết thúc cuộc hẹn.....	38
Figure 20: Activity diagram: Thêm thuốc .....	39
Figure 21: Activity diagram: Sửa thuốc .....	39
Figure 22 Class diagram: Overview .....	40
Figure 23: Đăng nhập .....	59
Figure 24: Danh sách bệnh án.....	59
Figure 25: Form thêm mới bệnh án.....	59
Figure 26: Popup xác nhận xóa bệnh án.....	60
Figure 27: Chi tiết bệnh án (Danh sách lần khám) .....	60
Figure 28: Popup xác nhận xóa lần khám .....	60
Figure 29: Form thông tin lần khám: khám sơ bộ.....	61
Figure 30: Form thông tin lần khám: Chẩn đoán.....	61
Figure 31: Form thông tin lần khám: Điều trị.....	61
Figure 32: Danh sách thuốc .....	62
Figure 33: Form thêm mới thuốc .....	62
Figure 34: Danh sách cuộc hẹn.....	62
Figure 35: Danh sách cuộc hẹn theo lịch.....	63
Figure 36: Form thêm mới cuộc hẹn.....	63

Figure 37: Form Sửa thông tin cuộc hẹn .....63



# Chương 1: Giới thiệu và đề xuất

## 1.1. Thực trạng

Hiện nay công nghệ đã đi vào đời sống, những bước phát triển mạnh mẽ của công nghệ thông tin đã mang lại ứng dụng to lớn trong đời sống con người. Cách thức làm việc của con người đã được cải thiện hoàn toàn mới mẻ và sáng tạo, linh hoạt, giúp cho công việc nâng cao hiệu quả và tăng độ chính xác. Đặc biệt, nó đã đánh dấu vai trò quan trọng trong việc phát triển các hệ thống quản lý nói chung và quản lý bệnh án cho bệnh viện nói riêng.

Tuy nhiên đối với các hệ thống bệnh viện truyền thống, hệ thống quản lý bệnh án chưa được phát triển nhiều. Đa phần người bệnh đều sử dụng sổ để lưu lại mỗi lần đi khám bệnh. Và đối với những người bệnh với tần suất khám bệnh thấp, thường họ phải mua sổ mới mỗi lần khám. Đồng thời bên bệnh viện cũng mất tài nguyên để lưu trữ lại bệnh án của bệnh nhân. Những lần bệnh nhân đi khám lại, đa phần là bệnh nhân là phải mang theo sổ y bạ để bác sĩ khám lại, thay vì bác sĩ xem lại bệnh án được lưu lại của bệnh nhân trong bệnh viện. Việc này cản trở cả bác sĩ, y tá và bệnh nhân trong quá trình điều trị.

Còn đối với các hệ thống đã triển khai hệ thống quản lý bệnh án điện tử, thường đây là hệ thống quản lý nội bộ trong bệnh viện. Nhược điểm của nó là bệnh án của người bệnh được lưu lại sẽ không được chia sẻ cho người bệnh, còn đối với hệ thống chia sẻ để lưu trữ lại cho người bệnh thì lại khó sử dụng, đặc biệt là đối với những người có tuổi, khó tiếp cận với công nghệ, nhưng lại có nhu cầu khám bệnh cao

Vì vậy chúng ta cần một hệ thống lưu trữ được bệnh án cho cả bệnh viện và người bệnh, đồng thời dễ dàng sử dụng đối với phần đông người dùng

## 1.2. Yêu cầu cần giải quyết

Như thực trạng đã nêu trên, bài toán được đặt ra một cách cụ thể và rõ ràng:

Đưa ra hệ thống quản lý bệnh án điện tử để quản lý bệnh án của người bệnh, đồng thời quản lý các cuộc hẹn của người bệnh và quá trình điều trị của người bệnh tại bệnh viện, như vậy sẽ làm tăng hiệu quả của quá trình khám chữa bệnh của người bệnh tại bệnh viện.

Đồng thời hệ thống cũng phải áp dụng chức năng nhận diện khuôn mặt để tối ưu hóa lượng bệnh nhân cao tuổi chậm hoặc khó thích nghi với công nghệ quản lý bệnh án thông thường. Chức năng nhận diện khuôn mặt sau khi nhận diện được khuôn mặt của người bệnh, tìm ra số mã bệnh án tương ứng và trích ra bệnh án của người bệnh cho bác sĩ. Từ đó, giảm thời gian tìm kiếm thông tin bệnh nhân và bệnh án cho bác sĩ và thời gian giao tiếp của bệnh nhân.

### 1.3. Thông tin cần giải quyết

Trong phạm vi đồ án tốt nghiệp cá nhân, thông tin cần giải quyết được rút gọn cho phù hợp với đề tài đồ án tốt nghiệp đồng thời đảm bảo chất lượng và lượng thông tin đủ để tạo tính chân thực cho sản phẩm.

Thông tin chính cần giải quyết là về bệnh án của bệnh nhân. Khi bệnh nhân đi khám, họ sẽ có thông tin cá nhân của mình được lưu lại. Khi bệnh nhân được điều trị, bác sĩ sẽ lưu lại quá trình điều trị của bệnh nhân vào bệnh án của họ, bao gồm lời chuẩn đoán cũng như các nhận xét về bệnh tình của bệnh nhân rồi từ đó đưa ra pháp đồ điều trị hiệu quả. Pháp đồ này sẽ bao gồm thuốc cũng như hướng dẫn sử dụng thuốc. Sau quá trình thực hiện pháp đồ điều trị, bác sĩ sẽ gặp lại bệnh nhân vào một buổi hẹn trước. Đối với các bệnh thông thường sẽ là 2 tuần sau khi bệnh nhân hoàn thành pháp đồ điều trị để quan sát tình trạng bệnh nhân trong sinh hoạt bình thường. Như vậy các thông tin cần giải quyết sẽ là: thông tin về bệnh án, thông tin chung về người bệnh, thông tin chung về điều trị, cuộc hẹn, thuốc.

Các thông tin sẽ được giải quyết theo trình tự khám bệnh của người bệnh đến khám lần đầu như sau:

Bước 1: Bệnh nhân đến bệnh viện, y tá tiếp nhận, nếu là bệnh nhân đến lần đầu thì thêm bệnh nhân, sau đó y tá thêm cuộc hẹn cho bệnh nhân với bác sĩ tương ứng.

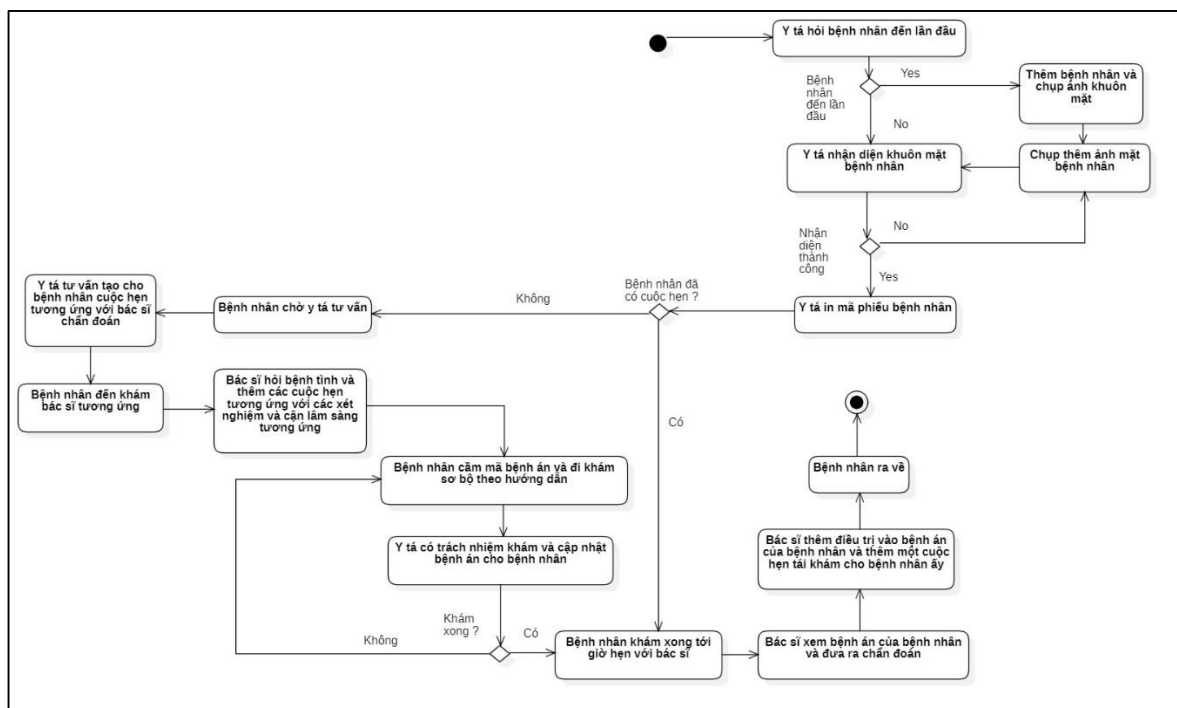
Bước 2: Bác sĩ khám bệnh cho bệnh nhân, cập nhật bệnh án, thêm các cuộc hẹn để xét nghiệm nếu cần thiết

Bước 3: Y tá thực hiện xét nghiệm và cập nhật vào bệnh án cho bệnh nhân

Đối với bệnh nhân đã khám nhiều lần, trình tự diễn ra theo các bước: 2 – (3) – 2

Đối với bệnh nhân đã khám lần đầu, trình tự diễn ra theo các bước: 1 – 2 – (3) – 2

Trong đó bước 3 có thể được lặp lại nhiều lần



**Quy trình khám bệnh**

#### 1.4. Đối tượng cá nhân

Hệ thống quản lý bệnh án tích hợp nhận diện khuôn mặt nằm trong khuôn khổ đề án tốt nghiệp được thực hiện bởi sinh viên Vũ Lê Hoàng dưới sự hướng dẫn của giảng viên Trần Đình Khang

## 1.5. Tính năng

Từ bài toán đề ra, các chức năng bắt buộc phải có bao gồm các chức năng nhận diện khuôn mặt cho phần nhận diện khuôn mặt người bệnh; sau đó khi đã nhận diện được khuôn mặt người bệnh, chức năng cần có tiếp theo là quản lý bệnh án người bệnh, cuộc hẹn; ngoài ra trong quá trình hoạt động của bệnh viện, hệ thống còn cung cấp các chức năng như quản lý thuốc, điều trị cho bệnh nhân, thông tin bệnh nhân. Như vậy hệ thống gồm 3 chức năng chính là nhận diện khuôn mặt, quản lý bệnh án và quản lý điều trị cho bệnh nhân.

## 1.6. Công nghệ sử dụng

- Frontend: Reactjs – Ant Design – Nextjs framework

React.js là một thư viện Javascript đang nổi lên trong những năm gần đây với xu hướng Single Page Application. Việc sử dụng Reactjs tăng trải nghiệm người dùng lên cao, đồng thời giảm tải cho hệ thống nặng về phần client khi xử lý nhiều tác vụ ở phía người dùng. Trong đó Ant design là một thư viện của reactjs gồm nhiều components dễ dùng và đẹp, đáp ứng đầy đủ các quy chuẩn về thiết kế giao diện; Nextjs là framework reactjs chạy theo mô hình server side rendering có tác dụng giảm nhẹ phần xử lý của phía khách hàng sang server.

- Backend: Python – Framework Django – graphql

Framework Django là framework python nổi tiếng về làm model về quản lý và quản trị, có cộng đồng lớn và được hỗ trợ nhiều thư viện. GraphQL là một Graph Query Language được dành cho API, thay thế cho restApi nhờ sự hiệu quả và linh hoạt hơn trong truy vấn cơ sở dữ liệu.

- Database: MySql
- Nhận dạng khuôn mặt: OpenCV library

OpenCV là thư viện cho thị giác máy tính hỗ trợ việc nhận diện khuôn mặt, đồng thời nó cũng hỗ trợ ngôn ngữ Python, giúp server được đồng nhất hơn.

## 1.7. Kế hoạch thực hiện

### 1.7.1. Phân tích yêu cầu

- Tổng quan về hệ thống lớn, phân tích chi tiết vào các chức năng Quản lý thông tin bệnh án bao gồm:
    - ❖ Thêm, sửa, xóa người bệnh
    - ❖ Thêm, sửa, xóa bệnh án
    - ❖ Thêm, sửa, xóa cuộc hẹn
- Hạn nộp: ngày 17/02/2020

### 1.7.2. Thiết kế hệ thống

#### a. Hệ thống quản lý bệnh án

- Tổng quan về kiến trúc hệ thống lớn là microservice, mô tả chi tiết hệ thống ứng với từng use case
- ❖ Các thành phần cần mô tả cho từng use – case trên tài liệu thiết kế hệ thống (Software Architecture Design – SAD)
  - Service: Danh sách các đầu API của service đó (theo chuẩn Rest Service)
  - Sequence diagram cho mỗi service tương tác với người dùng và client thông qua website
- ❖ Tài liệu thiết kế chi tiết
  - Thiết kế hoạt động chức năng của mỗi service dưới dạng biểu đồ flow diagram

#### b. Hệ thống nhận diện khuôn mặt

- ❖ Tài liệu thiết kế chi tiết
    - Thiết kế hoạt động của hệ thống nhận diện khuôn mặt dưới dạng biểu đồ flow diagram
- Hạn nộp: ngày 16/03/2020

### 1.7.3. Lập trình hệ thống

#### a. Giao diện website

1. Thiết kế giao diện website dạng single page application
  2. Service RestService
  3. Công nghệ: Reactjs
- Hạn nộp: ngày 11/05/2020

#### b. Nhận diện khuôn mặt

- Nhận diện khuôn mặt
  - Tích hợp camera
- Hạn nộp: ngày 25/05/2020

### 1.7.4. Báo cáo

- Kết quả test và sản phẩm phần mềm
  - Tài liệu SRS, SAD, SDD
- Hạn nộp: ngày 8/06/2020

## Chương 2: Phân tích hệ thống

### 2.1. Tìm hiểu các hệ thống tương tự

#### 2.1.1. Hệ thống quản lý bệnh viện của hệ thống bệnh viện Hồng Ngọc

- Hệ thống quản lý bệnh án và bệnh viện
- Lưu trữ nội bộ trong bệnh viện
- Đăng nhập bằng cách truy xuất thông tin bệnh nhân bằng số điện thoại và tên bệnh nhân

#### **So sánh với hệ thống cần làm:**

- Chưa có nhận diện khuôn mặt nên khó sử dụng hơn đối với người bệnh, đặc biệt là người bệnh cao tuổi khi đến bệnh viện cần cung cấp số an sinh xã hội (chứng minh thư) để tìm ra thông tin cá nhân. Việc mang theo nhiều giấy tờ hoặc nhớ lại mã số chứng minh thư hoặc căn cước công dân là khó khăn đối với các bệnh nhân cao tuổi.
- Hệ thống bệnh viện không chia sẻ bệnh án của bệnh nhân ra ngoài. Điều đó có nghĩa nếu bệnh nhân sử dụng hệ thống bệnh viện khác do nhu cầu về địa điểm hoặc nhu cầu cá nhân thì sẽ phải khám lại từ đầu. Điều này mang lại khó khăn cho đa số bệnh nhân thường xuyên phải di chuyển.
- Chưa có lưu trữ bệnh án cho người bệnh trong trường hợp người bệnh chuyển địa điểm khám thì sẽ không có bệnh án đi kèm người bệnh, buộc người bệnh nếu muốn giữ bệnh án thì phải giữ đơn thuốc và chẩn đoán đi kèm. Chức năng quản lý bệnh án có thể phát triển thêm để bệnh nhân có thể xem trực tiếp bệnh án của mình trên điện thoại, cũng như mang lại sự tự do tối đa về khám chữa bệnh cho bệnh nhân.

#### 2.1.2. Kết luận

Chức năng cơ bản	<ul style="list-style-type: none"><li>- Quản lý bệnh án</li><li>- Quản lý cuộc hẹn</li><li>- Quản lý thuốc</li></ul>
Chức năng nâng cao	<ul style="list-style-type: none"><li>- Quản lý bệnh án cho bệnh nhân</li></ul>
Chức năng chủ đạo	<ul style="list-style-type: none"><li>- Quản lý bệnh án</li></ul>

## Chương 3: Thiết kế hệ thống

### 3.1. Phân tích yêu cầu

Nội dung phân tích: Phân tích chi tiết vào các chức năng quản lý thông tin project dưới dạng biểu đồ use case và chi tiết interaction cho từng use case bao gồm 3 use case chính:

- Tạo mới, sửa thông tin cơ bản, xóa project
- Xây dựng kế hoạch cho project
- Update tiến trình nhiệm vụ.

Và đồng thời kèm theo các use case cho các hoạt động. Kết quả của phần này là tài liệu SRS.

#### 3.1.1. Usecase diagram

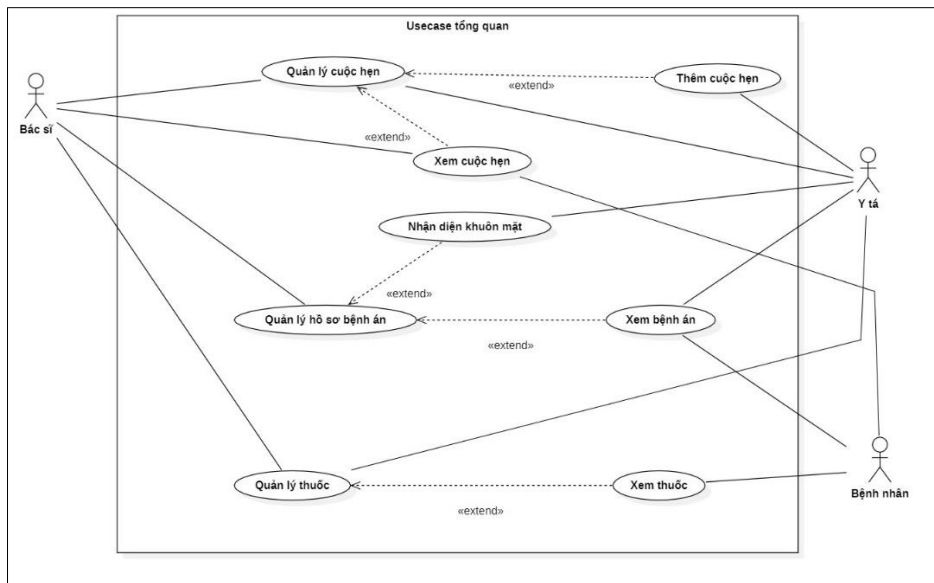


Figure 1: Usecase tổng quan

- Biểu đồ Use Case với 3 usecase chính:
- ❖ Quản lý bệnh án
- ❖ Quản lý thuốc
- ❖ Quản lý cuộc hẹn

#### Chi tiết usecase: Quản lý bệnh án

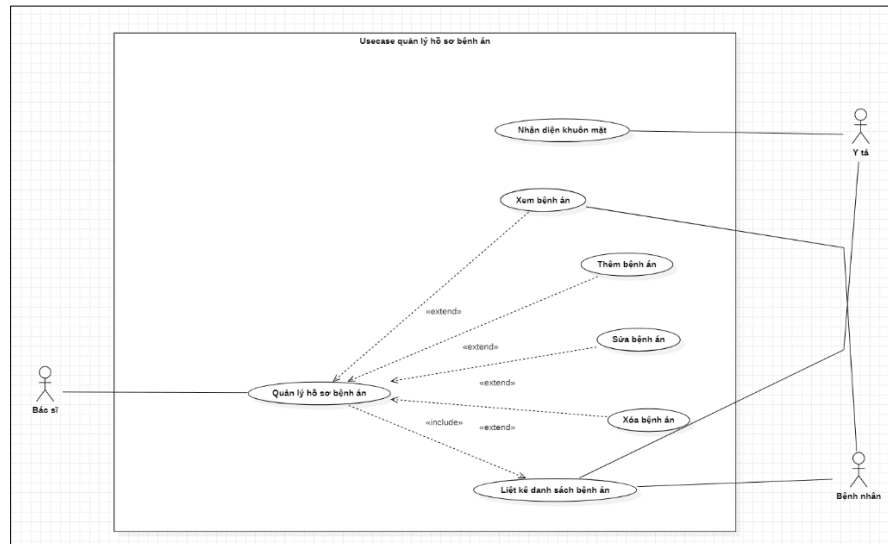


Figure 2: Biểu đồ usecase quản lý bệnh án



### Chi tiết usecase: Quản lý cuộc hẹn

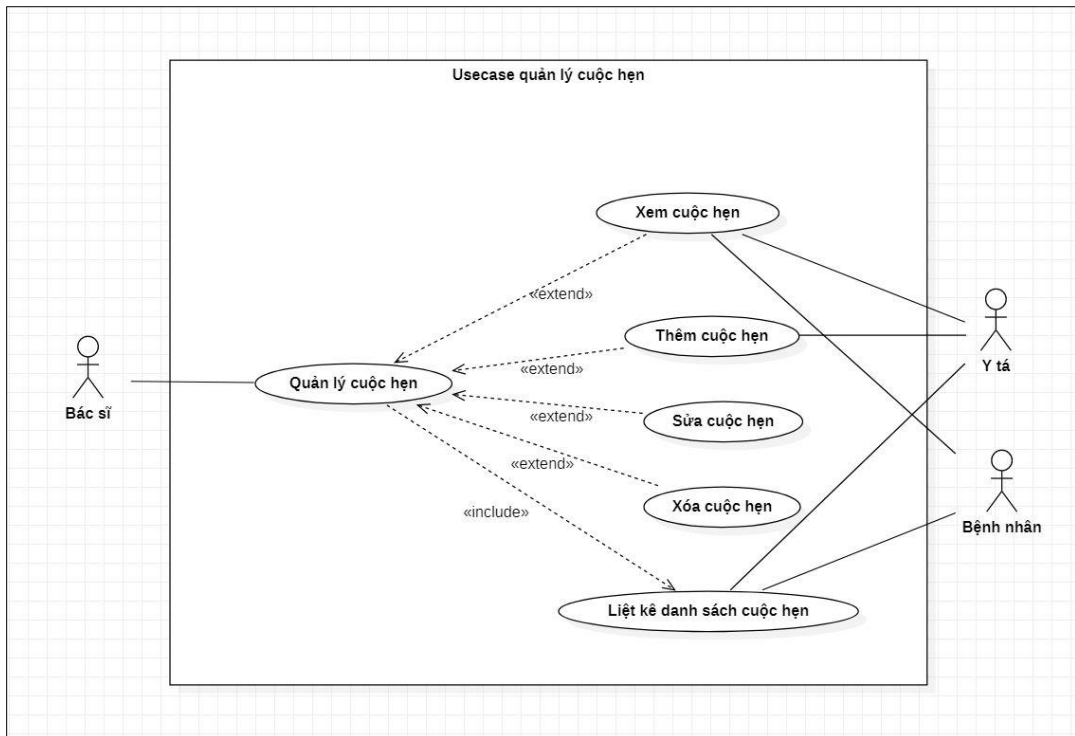


Figure 3: Biểu đồ usecase quản lý cuộc hẹn

### Chi tiết usecase: Quản lý thuốc

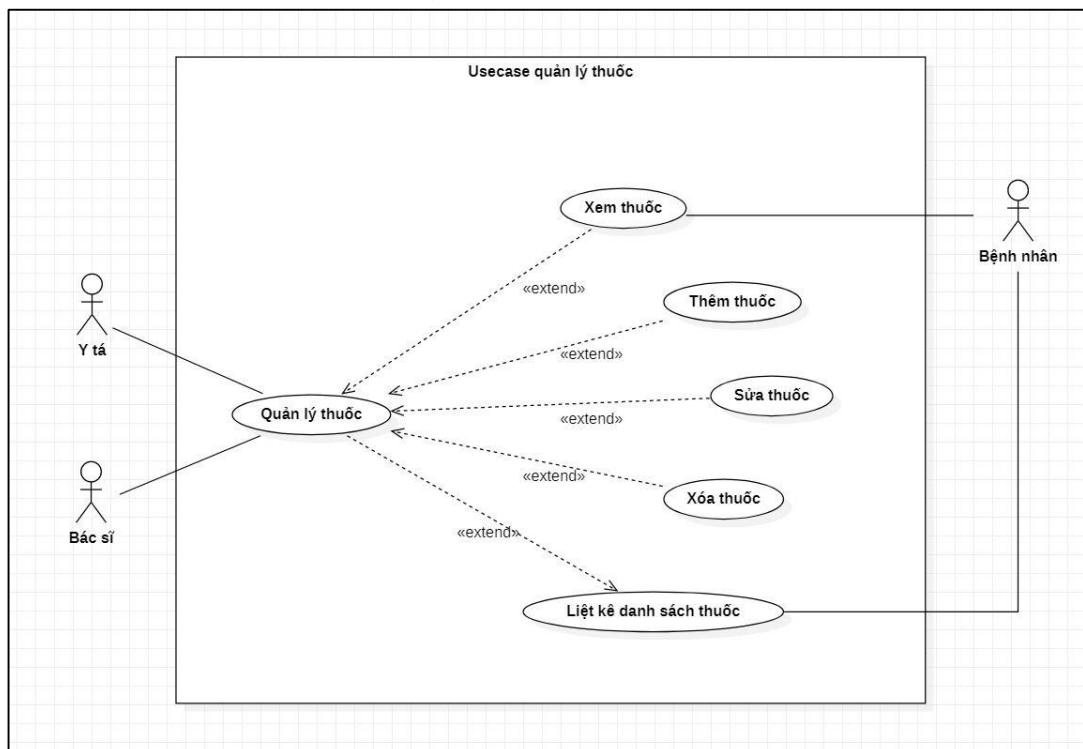


Figure 4: Biểu đồ usecase quản lý thuốc

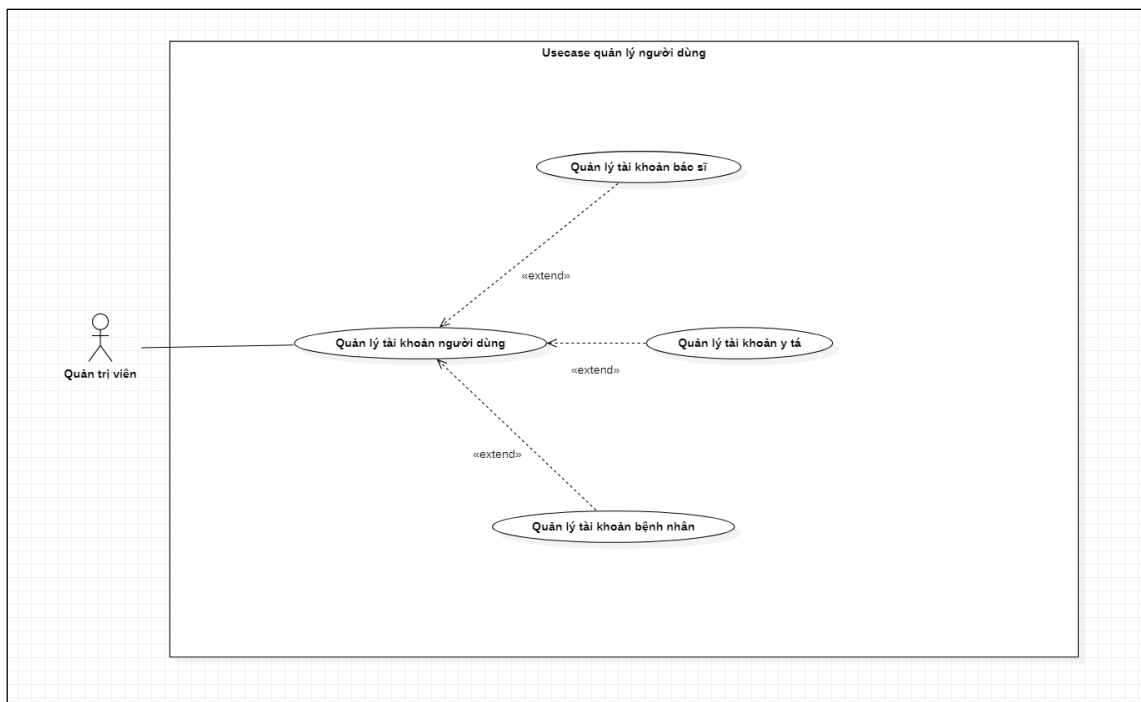


Figure 5: Biểu đồ usecase quản trị tài khoản

### 3.1.2. Các tác nhân

#### 1. Bác sĩ

- Mô tả: là người trực tiếp khám bệnh cho bệnh nhân và đưa ra chẩn đoán cũng như điều trị cho bệnh nhân được khám.
- Tác động: Các chức năng liên quan đến thêm, sửa, xóa các đối tượng như cuộc hẹn, thuốc, bệnh án.

#### 2. Y tá

- Mô tả: là người chịu trách nhiệm cho phần hỗ trợ bác sĩ và bệnh nhân.
- Tác động: Các chức năng liên quan đến quản lý cuộc hẹn và thuốc, xem bệnh án của bệnh nhân để đưa ra các hành động thích hợp hoặc giúp đỡ bệnh nhân thực hiện liệu trình chữa trị.

#### 3. Bệnh nhân

- Mô tả: là người đi khám bệnh, cần được chăm sóc và tư vấn từ đội ngũ y, bác sĩ
- Tác động: Xem và đọc về thuốc và bệnh án của bản thân.

### 3.1.3. Usecase 1: Quản lý hồ sơ bệnh án

**Yêu cầu:** Usecase cho phép bác sĩ quản lý hồ sơ bệnh án hoặc cho người bệnh và y tá xem hồ sơ bệnh án.

**Tác nhân:** Bác sĩ, y tá, bệnh nhân

**Tiền điều kiện:** Người dùng ấn vào bệnh án hoặc danh sách bệnh án.

**Mô tả khái quát:**

1. Hệ thống cho phép một bác sĩ tạo mới một hồ sơ bệnh án cho bệnh nhân sau khi khám bệnh cho bệnh nhân đó lần đầu hoặc thêm mới và sửa hồ sơ bệnh án của bệnh nhân đã có hồ sơ bệnh án.
2. Sau mỗi cuộc hẹn với bệnh nhân, luôn có hồ sơ bệnh án cần sửa hoặc thêm mới tương ứng với cuộc hẹn đó.

**Mô tả chi tiết các bước cần làm:**

Dòng hành động logic chung: Bác sĩ chọn cuộc hẹn đối với bệnh nhân, hệ thống sẽ đưa hành động thích hợp như thêm mới hoặc sửa bệnh án, bác sĩ sẽ sửa hoặc thêm mới chẩn đoán vào bệnh án của bệnh nhân sau khi khám và chẩn đoán cho bệnh nhân của mình.

**Mô tả kiểu dữ liệu nhập vào:**

Tên dữ liệu	Mô tả	Kiểu	Phạm vi hợp lệ
PatientId	Mã bệnh nhân	Text	- Độ dài tối đa: 50 - Không trùng với mã bệnh nhân đã có bệnh án

#### a. Tạo mới bệnh án

Các thông tin cần nhập khi tạo mới một bệnh án:

1. Mã bệnh nhân (Tự động sinh khi tạo mới, được sinh theo uuid v4)
2. Thông tin bệnh nhân như họ tên, địa chỉ, số điện thoại, tuổi tác, tiền sử bệnh lý, người bảo lãnh và liên lạc của người bảo lãnh

**Dòng hành động chính:**

1. Y tá ấn vào tạo mới bệnh án hoặc sửa bệnh án cho người bệnh
2. Y tá chọn tên bệnh nhân để tạo bệnh án hoặc sửa bệnh án (thêm chẩn đoán chữa trị) và thêm chẩn đoán, liệu trình
3. Y tá xác nhận tạo hoặc sửa bệnh án

Hành động của tác nhân	Hành động của hệ thống
1. Y tá chọn thêm chẩn đoán cho bệnh nhân	Hiển thị pop-up thêm chẩn đoán
2. Y tá nhập/chọn tên của bệnh nhân cần thêm chẩn đoán và nhập chẩn đoán và liệu trình cho bệnh nhân	Hiển thị pop-up xác nhận

3. Y tá ấn xác nhận có	Hệ thống kiểm tra xem bệnh nhân có hồ sơ bệnh án chưa, nếu chưa có thì tạo mới và thêm chẩn đoán, nếu có rồi thì thêm chẩn đoán vào hồ sơ bệnh án của bệnh nhân
4.	Hệ thống tắt pop-up

➔ Scenario: 1 – 2 – 3

### **Dòng hành động con thay thế:**

Vào lúc xác nhận thay đổi, y tá ấn hủy bỏ

<b>Hành động của tác nhân</b>	<b>Hành động của hệ thống</b>
1. Y tá chọn thêm chẩn đoán cho bệnh nhân	Hiện thị pop-up thêm chẩn đoán
2. Y tá nhập/chọn tên của bệnh nhân cần thêm chẩn đoán và nhập chẩn đoán và liệu trình cho bệnh nhân	Hiện thị pop-up xác nhận
3A. Y tá ấn hủy bỏ	Hệ thống tắt pop-up

➔ Scenario: 1 – 2 – 3

Vào lúc chọn tên bệnh nhân để sửa bệnh án hoặc thêm bệnh án cho bệnh nhân, chưa có bệnh nhân nào trong hệ thống, use case kết thúc

### **b. Sửa bệnh án**

#### **Dòng hành động chính:**

1. Tại giao diện khám chữa bệnh hoặc giao diện thông tin bệnh án của bệnh nhân, bác sĩ chọn sửa bệnh án
2. Bác sĩ thay đổi thông tin chẩn đoán của bệnh án đó
3. Bác sĩ ấn lưu lại thay đổi
4. Bác sĩ xác nhận thay đổi

<b>Hành động của tác nhân</b>	<b>Hành động của hệ thống</b>
1. Tại khám chữa bệnh hoặc giao diện thông tin bệnh án của bệnh nhân, bác sĩ ấn chọn sửa bệnh án	Hệ thống chuyển sang trang sửa bệnh án
2. Bác sĩ chọn thay đổi chẩn đoán hoặc liệu trình điều trị cho bệnh nhân	Hệ thống hiện thị pop-up (modal) cho việc sửa chẩn đoán hoặc liệu trình điều trị tương ứng
3. Bác sĩ ấn lưu thay đổi	Hệ thống hiện thị pop-up xác nhận lưu thay đổi

4. Bác sĩ xác nhận có	Hệ thống lưu lại thay đổi và tắt pop-up (modal) tương ứng
-----------------------	---

**Dòng hành động con thay thế:**

Vào lúc xác nhận thay đổi, bác sĩ ấn hủy bỏ

Hành động của tác nhân	Hành động của hệ thống
3. Bác sĩ ấn lưu thay đổi	Hệ thống hiển thị pop-up xác nhận lưu thay đổi
4A. Bác sĩ ấn hủy bỏ	Hệ thống tắt pop-up (modal) tương ứng

**c. Xóa bệnh án**

**Dòng hành động logic chính:**

1. Tại giao diện quản lý bệnh án, bác sĩ chọn xóa bệnh án
2. Bác sĩ chọn xóa bệnh án

Hành động của tác nhân	Hành động của hệ thống
1. Tại giao diện quản lý bệnh án, bác sĩ chọn xóa bệnh án	Hệ thống hiển thị pop-up xác nhận xóa bệnh án
2. Bác sĩ xác nhận xóa bệnh án	Hệ thống xóa các thông tin của bệnh án trên cơ sở dữ liệu

**Dòng hành động con thay thế:**

2A. Bác sĩ ấn hủy bỏ

Hành động của tác nhân	Hành động của hệ thống
1. Tại giao diện quản lý bệnh án, bác sĩ chọn xóa bệnh án	Hệ thống hiển thị pop-up xác nhận xóa bệnh án
2A. Bác sĩ ấn hủy bỏ	Hệ thống tắt pop-up

**d. Nhận diện khuôn mặt**

**Dòng hành động logic chính:**

1. Tại giao diện quản lý bệnh án, y tá chọn nhận diện bệnh nhân
2. Y tá xem bệnh án của bệnh nhân

Hành động của tác nhân	Hành động của hệ thống
1. Tại giao diện quản lý bệnh án, y tá chọn nhận diện khuôn mặt	Hệ thống sử dụng camera nhìn vào khuôn mặt của bệnh nhân và nhận dạng, xác định bệnh nhân và hiển thị bệnh án lên giao diện màn hình của y tá

### **Dòng hành động con thay thế:**

Trong trường hợp hệ thống không định dạng chính xác được khuôn mặt bệnh nhân và xuất hiện nhiều hơn 1 kết quả

<b>Hành động của tác nhân</b>	<b>Hành động của hệ thống</b>
1. Tại giao diện quản lý bệnh án, y tá chọn nhận diện khuôn mặt	Hệ thống sử dụng camera nhìn vào khuôn mặt của bệnh nhân và nhận dạng, xác định được từ 2 bệnh nhân có khuôn mặt được đưa vào nhận dạng
2. Y tá xác nhận thêm thông tin bệnh nhân và chọn bệnh nhân khớp với thông tin người bệnh được đưa ra	Hệ thống hiển thị thông tin của bệnh nhân được y tá chọn khớp với thông tin bệnh nhân đưa ra

### **3.1.4. Usecase 2: Quản lý cuộc hẹn**

**Yêu cầu:** Use case cho phép y tá hoặc bác sĩ quản lý các cuộc hẹn với các bệnh nhân.

**Tác nhân:** Bác sĩ hoặc y tá

**Tiền điều kiện:** Người quản lý cuộc hẹn phải là y tá hoặc bác sĩ

**Mô tả khái quát:**

Hệ thống cho phép bác sĩ hoặc y tá thêm hoặc sửa hoặc xóa các cuộc hẹn liên quan tới bệnh nhân, tùy vào điều kiện thay đổi.

Một cuộc hẹn có các trạng thái: Chưa đến hẹn, đang thực hiện và kết thúc cuộc hẹn. Một cuộc hẹn khi được kết thúc cần có thêm chẩn đoán và điều trị cho bệnh nhân

**Mô tả chi tiết các bước cần làm:**

- Xác định người tham gia cuộc hẹn
- Xác định thời gian và địa điểm của cuộc hẹn

Usecase được chia thành các usecase nhỏ hơn:

- Thêm mới cuộc hẹn
- Sửa cuộc hẹn
- Xóa cuộc hẹn

#### **a. Thêm mới cuộc hẹn**

**Dòng hành động chính:**

1. Bác sĩ hoặc y tá thêm mới cuộc hẹn
2. Bác sĩ hoặc y tá thêm thông tin về bệnh nhân và bác sĩ trong cuộc hẹn, thời gian và địa điểm cuộc hẹn.
3. Bác sĩ xác nhận thêm cuộc hẹn

Hành động của tác nhân	Hành động của hệ thống
1. Bác sĩ hoặc y tá ấn nút thêm cuộc hẹn	Chuyển sang giao diện thêm mới cuộc hẹn
2. Bác sĩ hoặc y tá thêm thông tin về bệnh nhân, bác sĩ, thời gian và địa điểm của cuộc hẹn	Kiểm tra thông tin đầu vào có hợp lệ hay không
3. Bác sĩ hoặc y tá xác nhận thêm cuộc hẹn	Hệ thống xác nhận cuộc hẹn hợp lệ
	Chuyển sang giao diện quản lý danh sách cuộc hẹn

#### **Dòng hành động con thay thế:**

Khi cuộc hẹn không hợp lệ, hệ thống hiển thị thông báo thông tin cuộc hẹn không hợp lệ ở thời gian hoặc địa điểm, dòng hành động quay lại bước thứ 2

Hành động của tác nhân	Hành động của hệ thống
3A. Tại giao diện thêm mới cuộc hẹn, bác sĩ hoặc y tá chọn thêm cuộc hẹn	Hệ thống kiểm tra và xác nhận cuộc hẹn không hợp lệ về thời gian hoặc địa điểm

Scenario: 1 – 2 – 3A – 2

#### **Mô tả dữ liệu tạo mới:**

Tên dữ liệu	Mô tả	Kiểu	Phạm vi hợp lệ
appointmentId	Mã cuộc hẹn	Text	- Độ dài: 50 - Tự động sinh
doctorId	Mã bác sĩ	Text	- Độ dài: 50 - Tự động điền khi chọn
patientId	Mã bệnh nhân	Text	- Độ dài: 50 - Tự động điền khi chọn
time	Thời gian cuộc hẹn	Text	- Độ dài: 20 - Tự động điền khi chọn
address	Địa điểm cuộc hẹn	Text	- Độ dài tối đa: 100
status	Trạng thái cuộc hẹn	Text	- Độ dài: 1
treatmentId	Điều trị	Text	- Độ dài: 50 - Tự động sinh khi thêm

#### **b. Sửa cuộc hẹn**

#### **Dòng hành động chính:**

1. Bác sĩ hoặc y tá ấn vào nút sửa của cuộc hẹn cần sửa
2. Bác sĩ hoặc y tá sửa thông tin của cuộc hẹn
3. Bác sĩ hoặc y tá xác nhận sửa cuộc hẹn



Hành động của tác nhân	Hành động của hệ thống
1. Bác sĩ hoặc y tá ấn vào nút sửa của cuộc hẹn cần sửa	Hiển thị màn hình sửa cuộc hẹn của cuộc hẹn tương ứng
2. Bác sĩ hoặc y tá tiến hành sửa thông tin cuộc hẹn hoặc thêm điều trị khi kết thúc cuộc hẹn	Kiểm tra thông tin đầu vào có hợp lệ hay không
3. Bác sĩ hoặc y tá xác nhận sửa	Kiểm tra thông tin đầu vào có hợp lệ hay không
	Hiển thị kết quả kiểm tra thông tin

➔ Scenario: 1 – 2 – 3 – 4

- **Dòng hành động con thay thế:**

Khi hệ thống kiểm tra thông tin đầu vào không hợp lệ:

Hành động của tác nhân	Hành động của hệ thống
3A. Bác sĩ hoặc y tá nhập thông tin của cuộc hẹn không hợp lệ	Hệ thống kiểm tra thời gian và địa điểm không hợp lệ, hiển thị yêu cầu nhập lại
4A. Bác sĩ hoặc y tá nhập thông tin về thời gian và địa điểm hợp lệ và xác nhận	Hệ thống xác nhận thông tin hợp lệ và tiến hành lưu dữ liệu đã nhập vào hệ thống

➔ Scenario: 1 – 2 – 3A – 4A – 4 – 5.

### 3.1.5. Use Case 3: Quản lý thuốc

**Yêu cầu:** Use case cho phép y tá hoặc bác sĩ quản lý các loại thuốc trong bệnh viện.

**Tác nhân:** Bác sĩ hoặc y tá

**Tiền điều kiện:** Người quản lý thuốc phải là y tá hoặc bác sĩ

**Mô tả khái quát:**

Hệ thống cho phép bác sĩ hoặc y tá thêm hoặc sửa hoặc xóa các loại thuốc

**Mô tả chi tiết các bước cần làm:**

Xác định các thông tin về loại thuốc cần thêm vào như tên thuốc, số lượng, tên công ty sản xuất, mô tả, hướng dẫn sử dụng, loại thuốc, lưu ý, được phép mua tự do, loại hàng ổn định giá

Usecase được chia thành các usecase nhỏ hơn:

- Thêm mới thuốc
- Sửa thông tin thuốc
- Xóa thuốc

#### a. Thêm mới thuốc

**Dòng hành động chính:**

1. Bác sĩ hoặc y tá thêm mới loại thuốc

2. Bác sĩ hoặc y tá thêm thông tin về loại thuốc mới trong bệnh viện
3. Bác sĩ xác nhận thêm loại thuốc

Hành động của tác nhân	Hành động của hệ thống
1. Bác sĩ hoặc y tá ấn nút thêm cuộc hẹn	Chuyển sang giao diện thêm mới cuộc hẹn
2. Bác sĩ hoặc y tá thêm thông tin về thuốc	Kiểm tra thông tin đầu vào có hợp lệ hay không
3. Bác sĩ hoặc y tá xác nhận thêm thuốc	Hệ thống xác nhận cuộc hẹn hợp lệ
	Chuyển sang giao diện quản lý danh sách cuộc hẹn

**Mô tả dữ liệu tạo mới:**

STT	Tên dữ liệu	Mô tả	Kiểu	Phạm vi hợp lệ
1	medicationId	Mã cuộc hẹn	Text	- Độ dài: 50 - Tự động sinh
2	medicationName	Tên thuốc	Text	- Độ dài: 10
3	quantity	Số lượng	Number	- Độ dài: 10
4	companyName	Công ty sản xuất	Text	- Độ dài: 50 - Tự động điền khi chọn
5	description	Mô tả	Text	- Độ dài tối đa: 512
6	medicationGuide	Hướng dẫn sử dụng	Text	- Độ dài: 512
7	typeId	Mã loại thuốc	Text	- Độ dài: 50 - Tự động sinh khi chọn
8	notion	Lưu ý	Text	- Độ dài: 256
9	isFreeBuy	Được mua tự do hay không	Bool	- Độ dài: 1 - Tự động sinh khi chọn
10	isFinedMedication	Là loại hàng ổn định giá	Bool	- Độ dài: 1 - Tự động sinh khi chọn

**b. Sửa thuốc**

**Dòng hành động chính:**

1. Bác sĩ hoặc y tá ấn vào nút sửa của cuộc hẹn cần sửa
2. Bác sĩ hoặc y tá sửa thông tin của cuộc hẹn
3. Bác sĩ hoặc y tá xác nhận sửa cuộc hẹn

Hành động của tác nhân	Hành động của hệ thống
1. Bác sĩ hoặc y tá ấn vào nút sửa của thuốc cần sửa	Hiển thị màn hình sửa thuốc của thuốc tương ứng
2. Bác sĩ hoặc y tá tiến hành sửa thông tin thuốc	Kiểm tra thông tin đầu vào có hợp lệ hay không
3. Bác sĩ hoặc y tá xác nhận sửa	Kiểm tra thông tin đầu vào có hợp lệ hay không

	Hiện thị kết quả kiểm tra thông tin
--	-------------------------------------

➔ Scenario: 1 – 2 – 3 – 4

### 3.1.6. Usecase 4: Nhận diện khuôn mặt

#### Dòng hành động chính:

1. Bác sĩ ấn vào nút nhận diện khuôn mặt
2. Hệ thống nhận diện khuôn mặt và đưa ra bệnh án tương đương cho bác sĩ

Hành động của tác nhân	Hành động của hệ thống
1. Bác sĩ ấn vào nút nhận diện khuôn mặt	Kiểm tra khuôn mặt của bệnh nhân trong cơ sở dữ liệu khuôn mặt của bệnh viện và đưa ra bệnh án tương ứng
2. Bác sĩ nhận bệnh án người bệnh	

➔ Scenario: 1 – 2

#### Dòng hành động phụ:

1. Bác sĩ ấn vào nút nhận diện khuôn mặt
2. Hệ thống nhận diện khuôn mặt và đưa ra bệnh án tương đương cho bác sĩ

Hành động của tác nhân	Hành động của hệ thống
1. Bác sĩ ấn vào nút nhận diện khuôn mặt	Kiểm tra khuôn mặt của bệnh nhân trong cơ sở dữ liệu khuôn mặt của bệnh viện nhưng không có gương mặt phù hợp
2. Bác sĩ yêu cầu bệnh nhân quay lại gặp y tá để kiểm tra thông tin (*)	

➔ Scenario: 1 – 2

(\*): Y tá sẽ tiến hành kiểm tra xem bệnh nhân đã được tạo trong danh sách bệnh nhân chưa. Nếu bệnh nhân đã được tạo, lỗi nhận diện thuộc về hệ thống, y tá tiến hành lấy lại ảnh của bệnh nhân và nhận diện lại, nếu không thành công thì báo cho admin để tiến hành xử lý. Nếu bệnh nhân chưa được tạo, y tá tiến hành tạo mới bệnh nhân và bệnh án cho bệnh nhân đó.

### 3.1.7. Usecase quản trị tài khoản

#### Dòng hành động chính:

1. Quản trị viên chọn loại đối tượng muốn quản lý
2. Quản trị viên thực hiện thao tác quản lý bao gồm thêm, sửa, xóa
3. Quản trị viên xác nhận thực hiện thao tác
4. Quản trị viên xác nhận có

Hành động của tác nhân	Hành động của hệ thống
1. Quản trị viên chọn loại đối tượng muốn quản lý	Hiện thị trang quản lý đối tượng tương ứng
2. Quản trị viên thực hiện thao tác quản lý	
3. Quản trị viên xác nhận thực hiện thao tác	Hiện thị popup xác nhận thực hiện thao tác
4. Quản trị viên xác nhận có	Hệ thống lưu lại kết quả thao tác thực hiện của quản trị viên

➔ Scenario: 1 – 2 – 3 – 4

#### Dòng hành động phụ:

3. Quản trị viên xác nhận thực hiện thao tác
4. Quản trị viên xác nhận không

Hành động của tác nhân	Hành động của hệ thống
3. Quản trị viên xác nhận thực hiện thao tác	Hiện thị popup xác nhận thực hiện thao tác
4. Quản trị viên xác nhận không thực hiện	Hệ thống bỏ qua thao tác quản trị viên đã thực hiện, quay lại trang quản lý

➔ Scenario: 1 – 2

(\*): Do cả 3 đối tượng là tài khoản bác sĩ, y tá và bệnh nhân đều chung cách thức quản lý, ở đây chỉ đưa ra đối tượng nói chung để thao tác

### 3.1.8. Các chức năng khác

Các chức năng như tìm kiếm và thống kê mặc định được thêm vào tại trang quản lý liệt kê của từng đối tượng như bệnh án:

- Chức năng thống kê: Thống kê theo ngày, tháng, năm hoặc theo tổng thời gian
- Chức năng tìm kiếm: Tìm kiếm theo từng trường của đối tượng sở hữu

## 3.2. Phân tích kiến trúc

### 3.2.1. Mục tiêu kiến trúc và ràng buộc

#### a. Tổng quan

Tài liệu thiết kế kiến trúc của phần mềm quản lý bệnh án được tạo ra với các mục tiêu sau đây:

1. Tạo thuận lợi cho người bệnh và bác sĩ trong quá trình tham gia khám chữa bệnh thông qua việc dễ dàng lưu trữ và theo dõi thông tin bệnh án.
2. Tối ưu hóa thời gian khám chữa bệnh cho bác sĩ và bệnh nhân

Các ràng buộc thiết kế và thực thi chính cho phần mềm quản lý bệnh án là:

#### 1. Dễ sử dụng

Người bệnh và bác sĩ dễ dàng thực hiện các chức năng mong muốn với số thao tác tối thiểu, đồng thời giao diện thân thiện và dễ hiểu đối với người dùng

#### 2. Tính linh hoạt

Hệ thống có thể được sử dụng trên điện thoại cũng như web để tiện lợi cho mọi người cùng sử dụng

Danh sách các yêu cầu cho phần mềm hệ thống quản lý bệnh án đã được mô tả chi tiết trong tài liệu SRS.

#### b. Kiến trúc chi tiết hệ thống

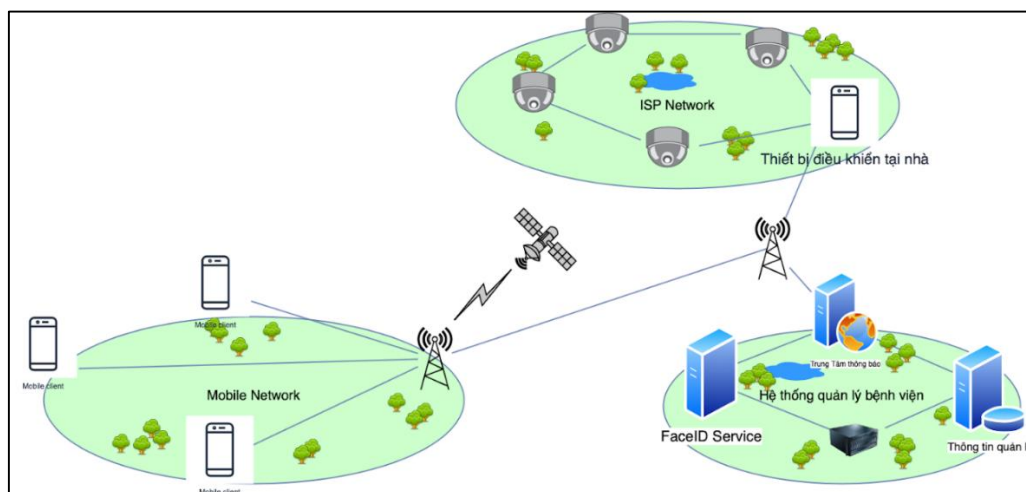


Figure 6: Kiến trúc hệ thống

Tổng quan về hệ thống bao gồm 1 webcam, điện thoại hoặc máy tính bàn và server để hoạt động.

Luồng hoạt động của hệ thống sẽ là: Người dùng bước vào, webcam quan sát khuôn mặt người dùng và gửi lên server, server nhận dạng khuôn mặt của người bệnh, xuất ra id người bệnh tương ứng, chuyển id sang cho hệ thống quản lý bệnh án, hệ thống quản lý bệnh án sẽ từ id trích xuất ra thông tin bệnh án của người bệnh và gửi về cho bác sĩ thông qua thiết bị kết nối với webcam.

### 3.2.2. Góc nhìn kiến trúc

Phần này sẽ đưa ra góc nhìn về kiến trúc chi tiết của hệ thống, bao gồm luồng thực hiện trên server trong quá trình thao tác của người dùng bao gồm bác sĩ, y tá, bệnh nhân đối với hệ thống và phản hồi tương ứng của hệ thống.

#### a. Góc nhìn logic

Phần này mô tả góc nhìn logic của kiến trúc: nhìn một cách tổng thể về kiến trúc của hệ thống bao gồm các lớp quan trọng và cách tổ chức của chúng trong hệ thống.

Góc nhìn kiến trúc của hệ thống chia hệ thống ra làm 3 gói chính:

- Gói giao diện người dùng bao gồm các lớp hoạt động với người dùng, các form để người dùng giao tiếp với hệ thống, hỗ trợ việc kiểm tra độ hợp lệ của dữ liệu đầu vào và hiển thị kết quả kiểm tra dữ liệu, hiển thị thông báo và giao diện cơ bản cho các chức năng cơ bản CRUD.
- Gói dịch vụ nghiệp vụ bao gồm các lớp điều khiển cho giao diện với hệ thống như hệ thống quản lý các đối tượng nghiệp vụ, tính toán và xác định nhận diện khuôn mặt người bệnh.
- Gói đối tượng nghiệp vụ gồm các đối tượng thực thể hoạt động trong hệ thống như bệnh nhân, bệnh án, thuốc.

Biểu đồ các tầng hoạt động của hệ thống:

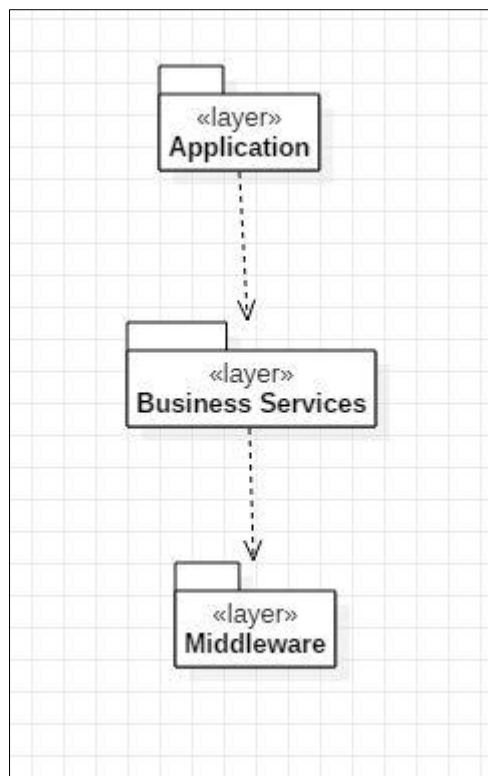


Figure 7: Kiến trúc logic

Chi tiết mô tả các tầng trong bảng bên dưới:

#	Tầng	Mô tả
1	Ứng dụng	Hiển thị giao diện mà người dùng nhìn thấy
2	Dịch vụ nghiệp vụ	Hiển thị và phục vụ các use-case quản lý các hành vi tương tác của người dùng hoặc quản trị viên ở tầng ứng dụng
3	Trung gian	Hỗ trợ truy xuất cơ sở dữ liệu

### 3.2.3. Góc nhìn quy trình

Góc nhìn quy trình sẽ mô tả hệ thống dưới dạng phân rã cũng như form giao tiếp giữa các thành phần hệ thống. Các thành phần của góc nhìn quy trình bao gồm các biểu đồ tuần tự và biểu đồ hoạt động.

#### a. Biểu đồ tuần tự

#### Usecase Quản lý hồ sơ bệnh án

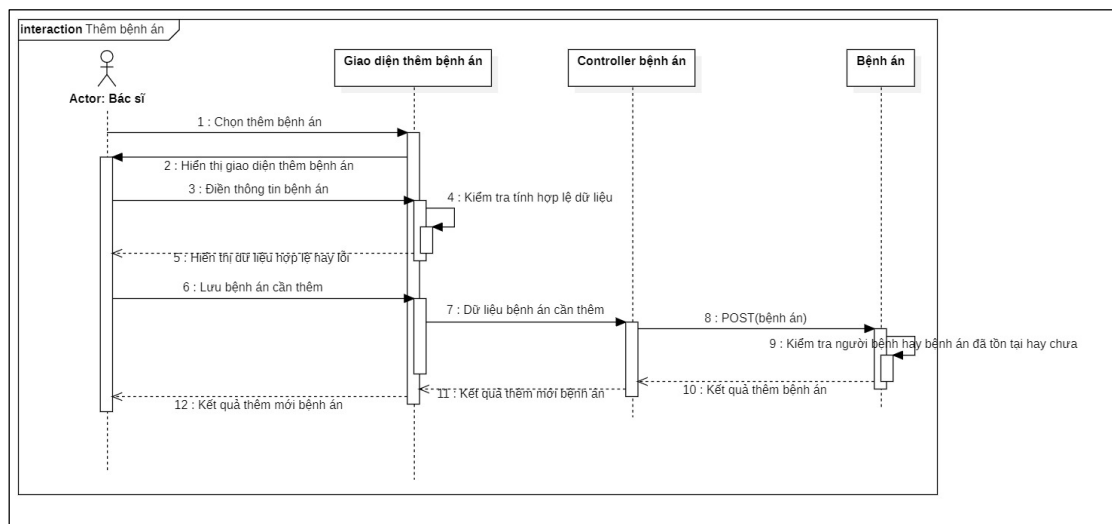


Figure 8: Sequence diagram: Thêm mới bệnh án

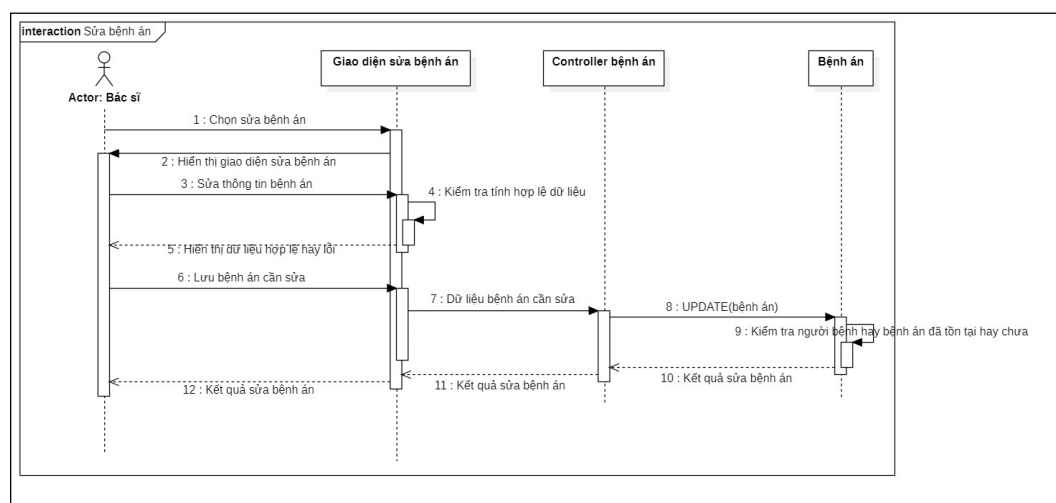


Figure 9: Sequence diagram: Sửa bệnh án



**Mô tả:** Luồng hoạt động chính của usecase thêm mới, sửa bệnh án theo thứ tự trong biểu đồ

Đối với các trường hợp kiểm tra không hợp lệ, người dùng quay lại thực hiện luồng từ người dùng của luồng kiểm tra tương ứng:

- Với luồng kiểm tra số 4 không hợp lệ, người dùng thực hiện lại từ luồng 3
- Với luồng kiểm tra số 9 không hợp lệ, người dùng thực hiện lại từ luồng 6
- Với mỗi quá trình sửa bệnh án, người dùng cần lưu lại thao tác thực hiện

### Usecase Quản lý cuộc hẹn

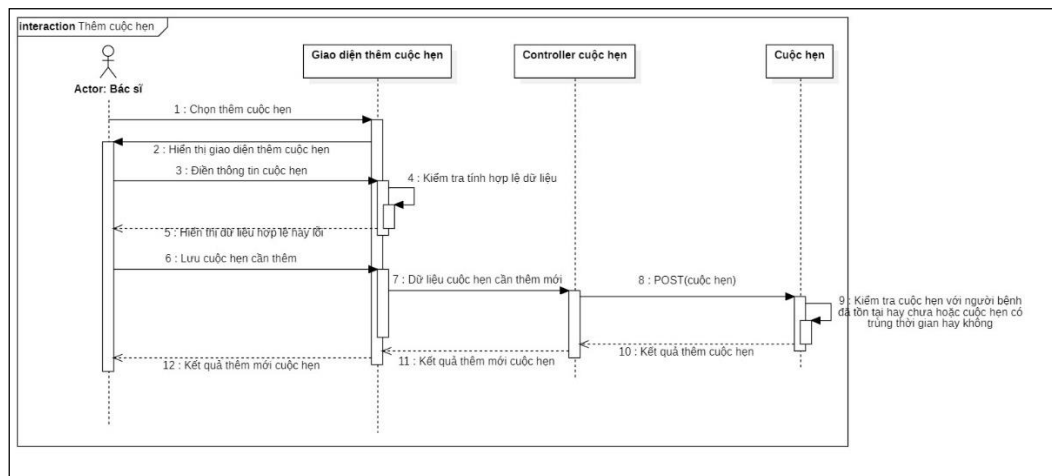


Figure 10: Sequence diagram: Thêm mới cuộc hẹn

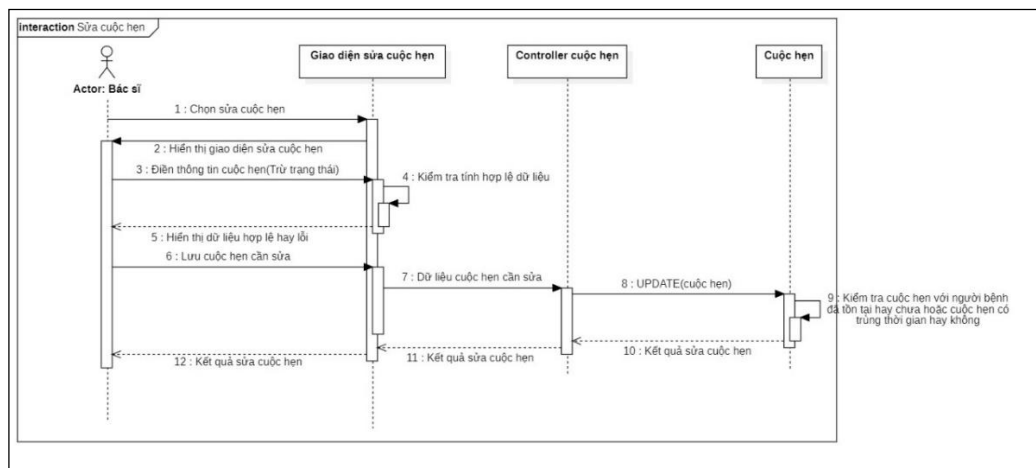


Figure 11: Sequence diagram: Sửa thông tin cuộc hẹn

**Mô tả:** Luồng hoạt động chính của usecase thêm mới, sửa cuộc hẹn theo thứ tự trong biểu đồ

Đối với các trường hợp kiểm tra không hợp lệ, người dùng quay lại thực hiện luồng từ người dùng của luồng kiểm tra tương ứng:

- Với luồng kiểm tra số 4 không hợp lệ, người dùng thực hiện lại từ luồng 3
- Với luồng kiểm tra số 9 không hợp lệ, người dùng thực hiện lại từ luồng 6

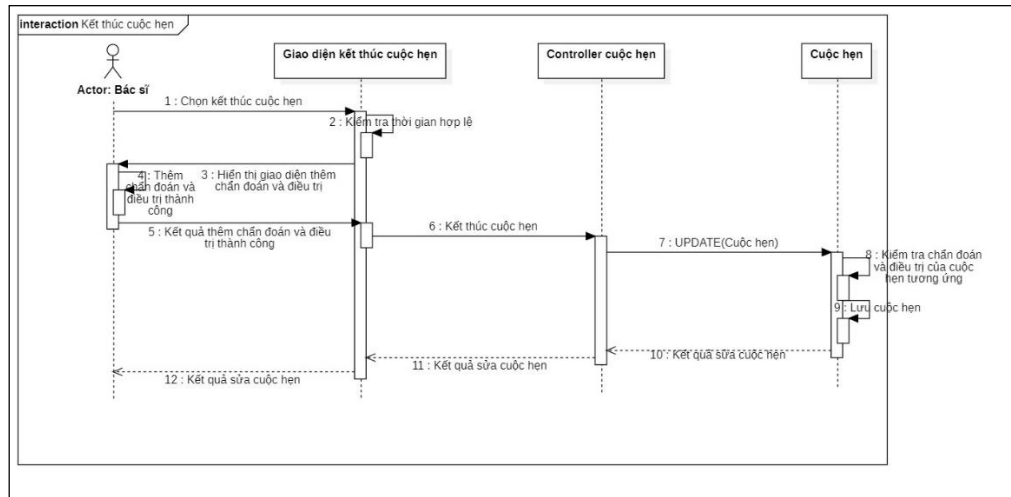


Figure 12: Sequence diagram: Kết thúc cuộc hẹn

**Mô tả:** Luồng hoạt động của usecase kết thúc cuộc hẹn theo thứ tự trong biểu đồ

Đối với các trường hợp kiểm tra không hợp lệ, người dùng quay lại thực hiện luồng từ người dùng của luồng kiểm tra tương ứng:

- Với luồng kiểm tra số 2 không hợp lệ, người dùng thực hiện lại từ luồng 1
- Với luồng kiểm tra số 8 không hợp lệ, người dùng thực hiện lại từ luồng 1 (trường hợp kẻ xấu có được token gửi gói tin đến hệ thống mà không qua xác thực từ client)

### Usecase Quản lý thuốc

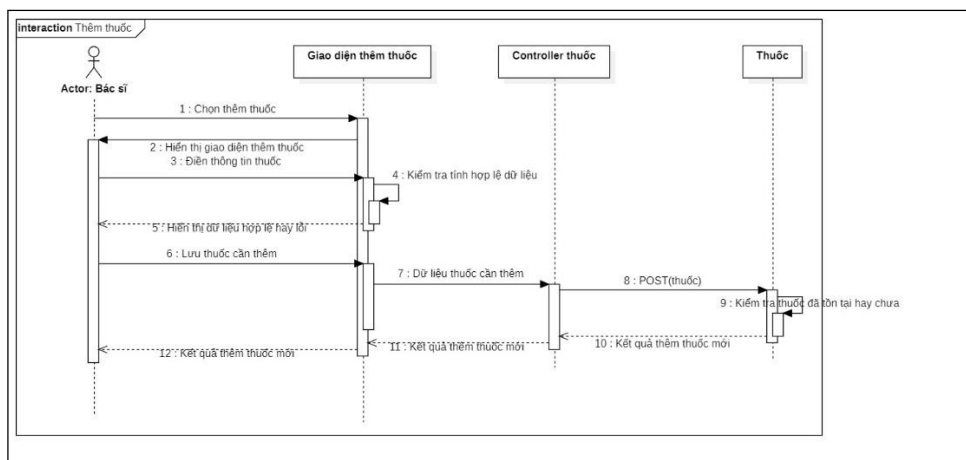


Figure 13: Sequence diagram: Thêm mới thuốc

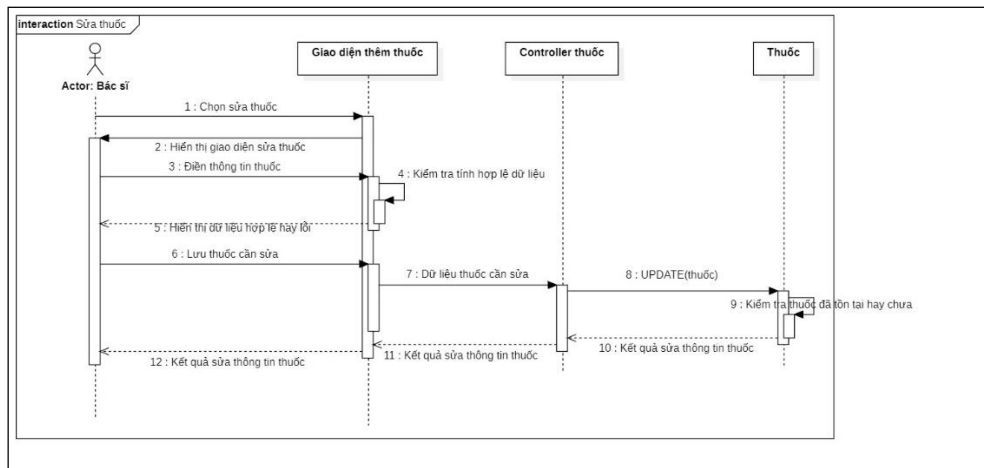


Figure 14: Sequence diagram: Sửa thông tin thuốc

**Mô tả:** Luồng hoạt động của usecase thêm mới thuốc và sửa thông tin thuốc theo thứ tự trong biểu đồ tuần tự

Đối với các trường hợp kiểm tra không hợp lệ, người dùng quay lại thực hiện luồng từ người dùng của luồng kiểm tra tương ứng:

- Với luồng kiểm tra số 4 không hợp lệ, người dùng thực hiện lại từ luồng 3
- Với luồng kiểm tra số 9 không hợp lệ, người dùng thực hiện lại từ luồng 6

b. Biểu đồ hoạt động

Usecase quản lý bệnh án

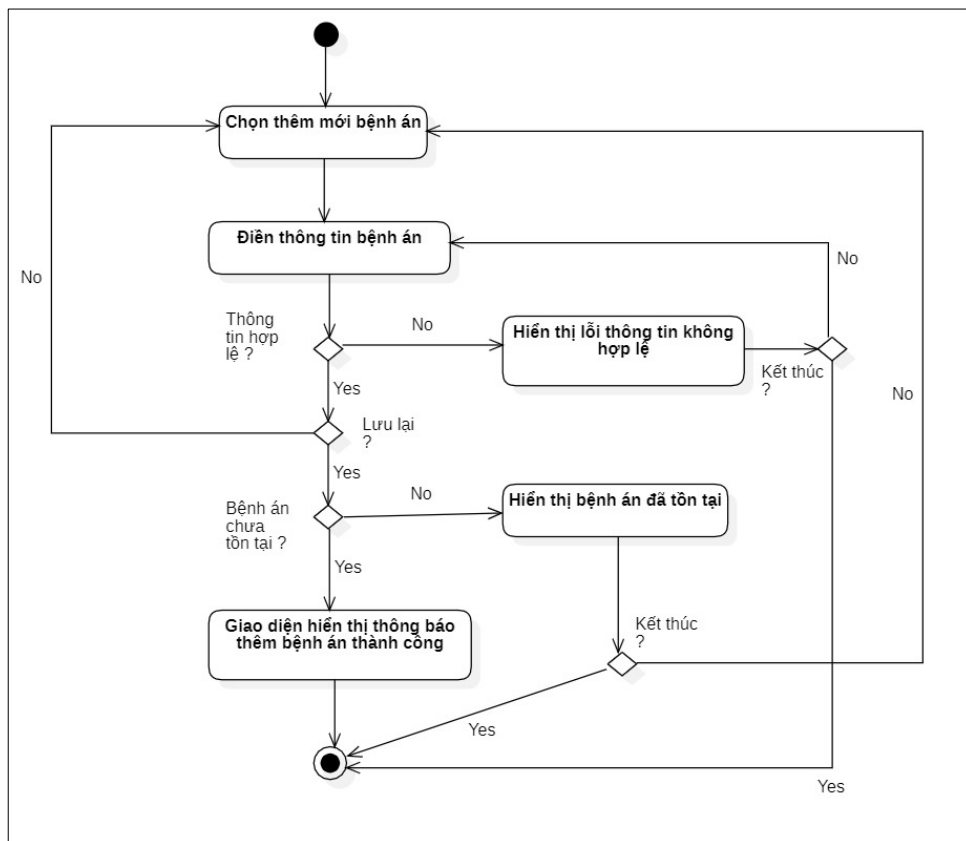


Figure 15: Activity diagram: Thêm bệnh án

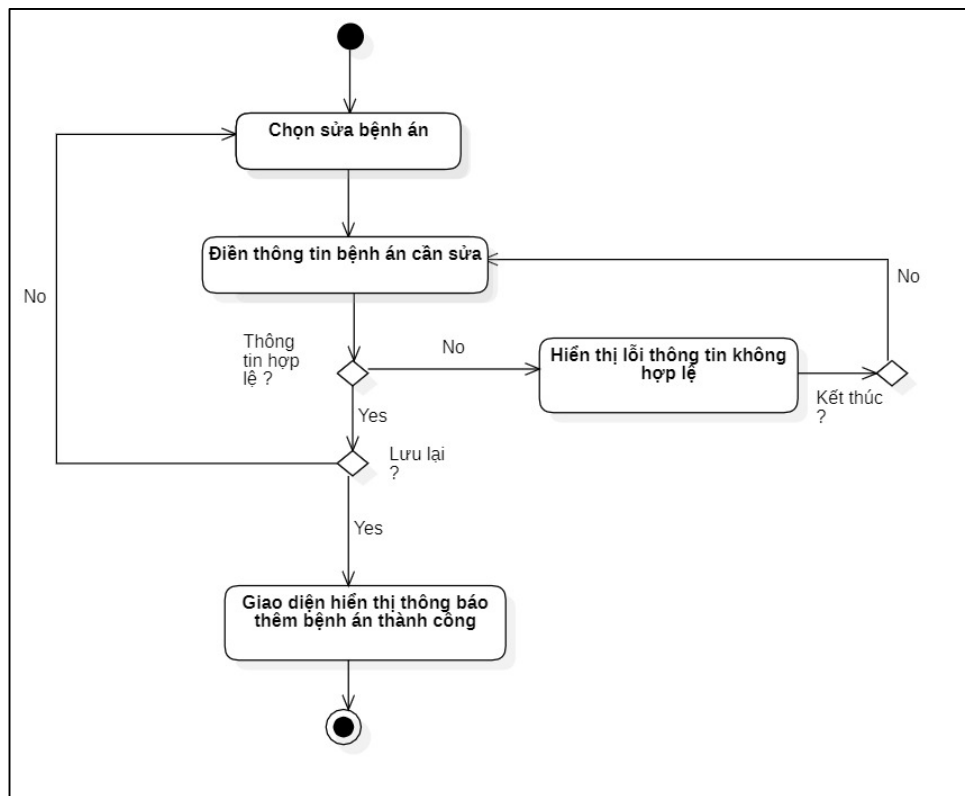


Figure 16: Activity diagram: Sửa bệnh án

## Usecase Quản lý cuộc hẹn

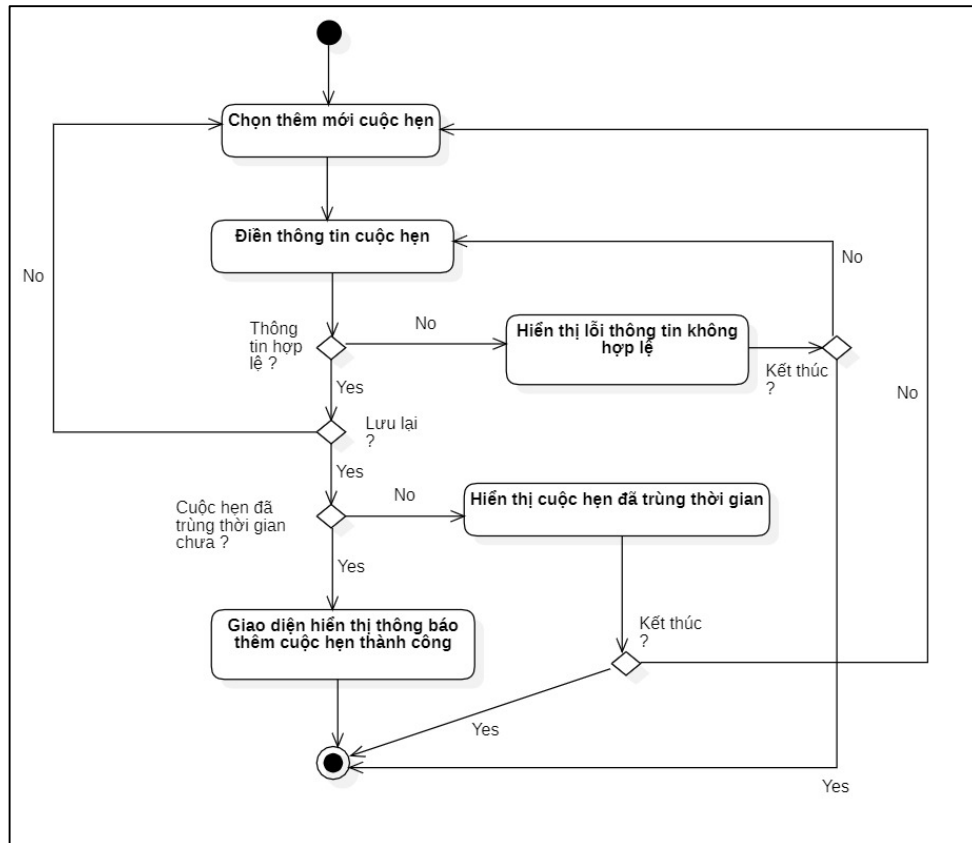


Figure 17: Activity diagram: Thêm cuộc hẹn

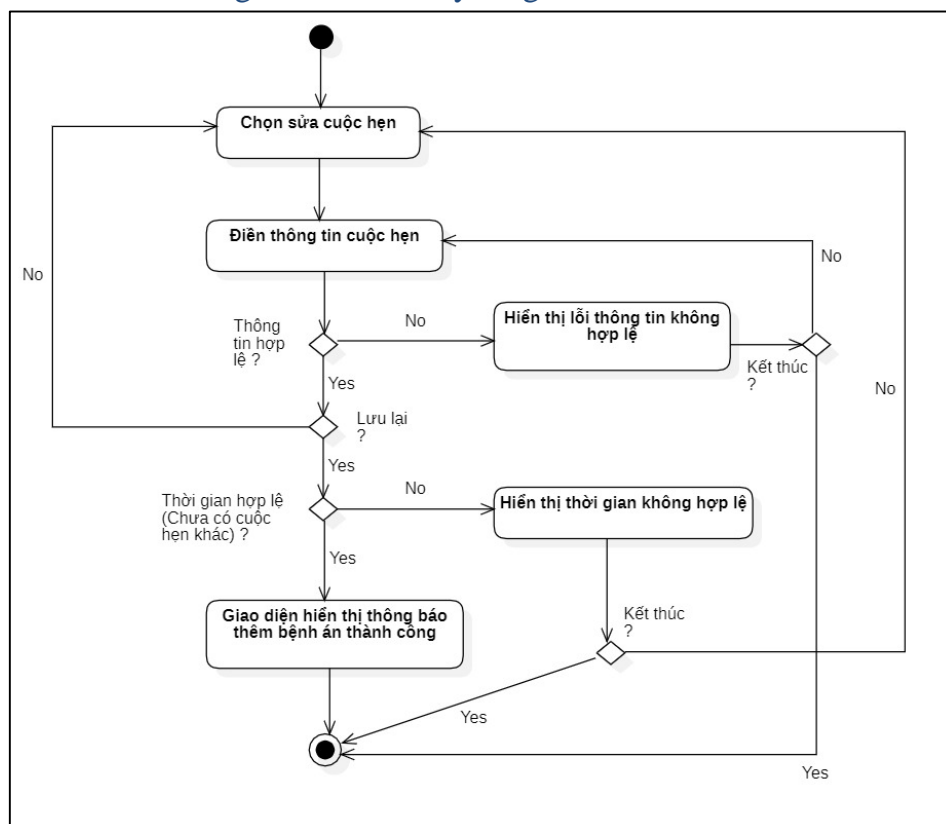


Figure 18: Activity diagram: Sửa cuộc hẹn

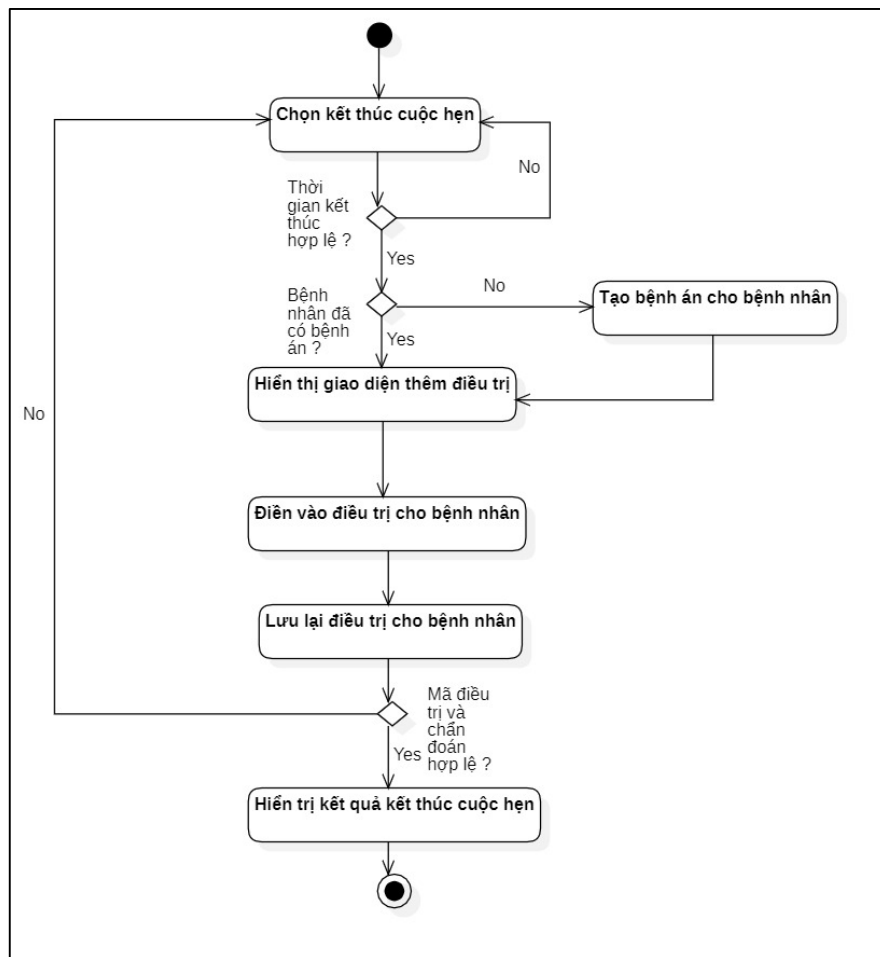


Figure 19: Activity diagram: Kết thúc cuộc hẹn

## Usecase Quản lý thuốc

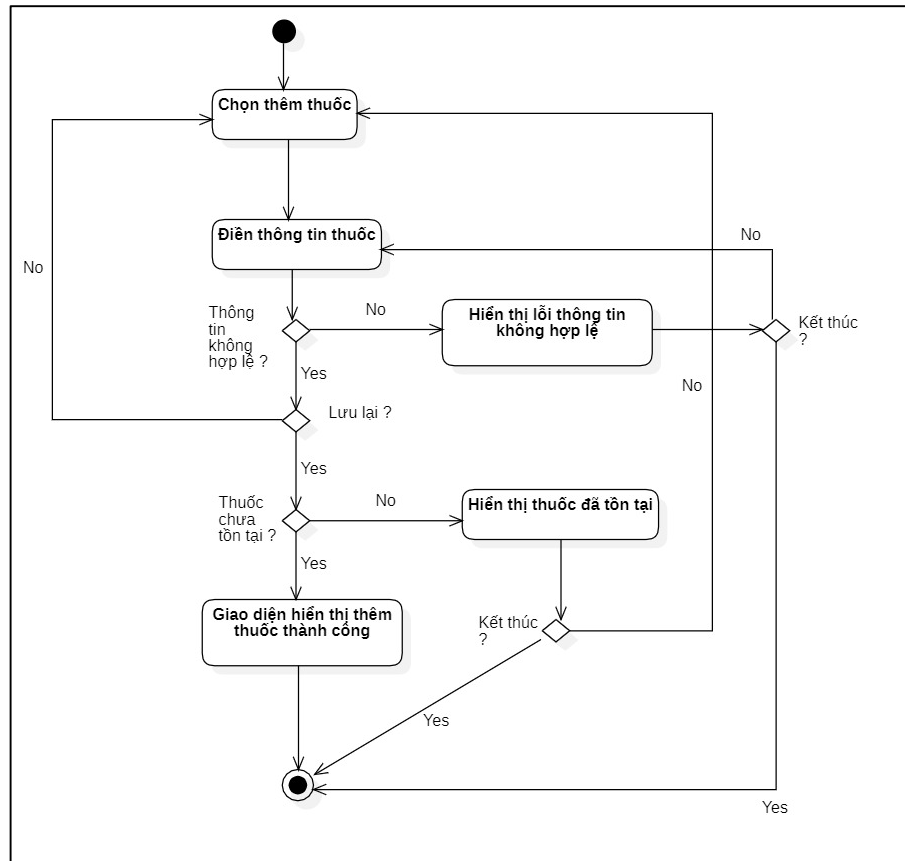


Figure 20: Activity diagram: Thêm thuốc

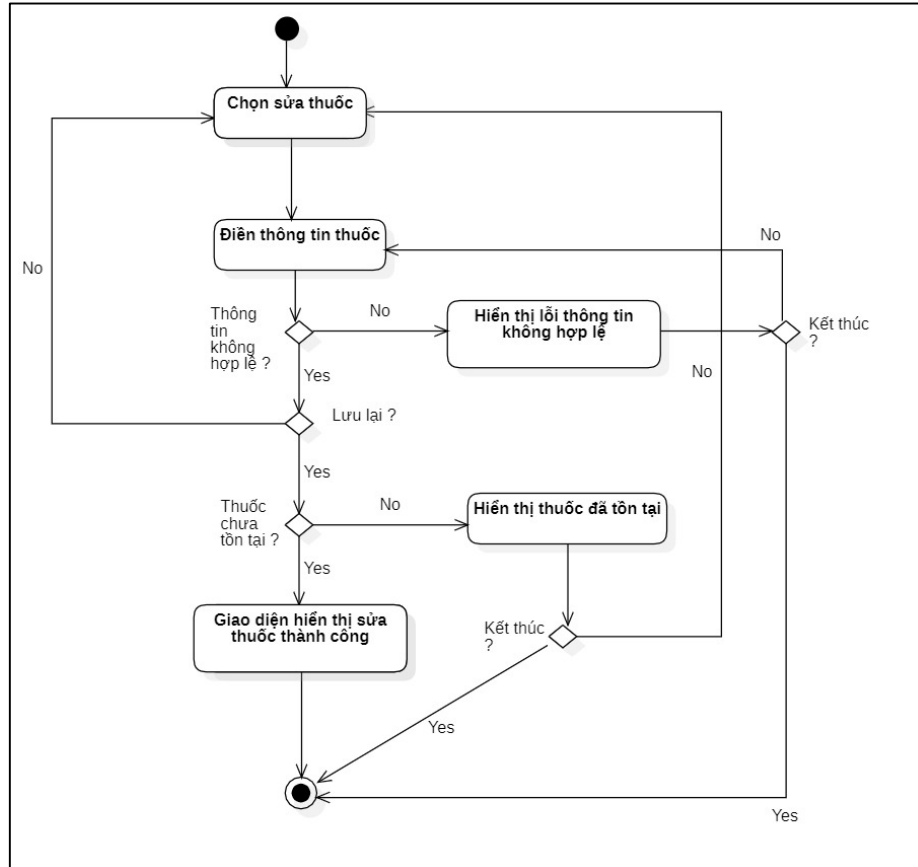


Figure 21: Activity diagram: Sửa thuốc



### 3.2.4. Phân tích chi tiết

#### a. Biểu đồ lớp tổng quan

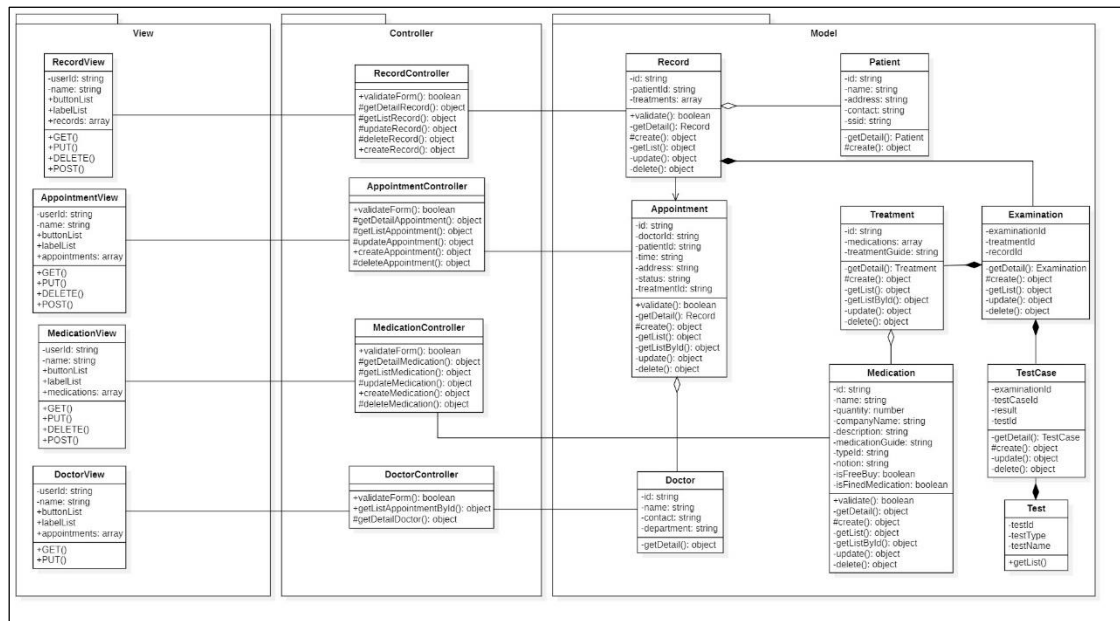


Figure 22: Class diagram: Overview

#### Package View:

- RecordView: Lớp hiển thị giao diện quản lý bệnh án
- AppointmentView: Lớp hiển thị giao diện quản lý cuộc hẹn
- MedicationView: Lớp hiển thị giao diện quản lý thuốc
- DoctorView: Lớp hiển thị giao diện chính cá nhân của bác sĩ

#### Package Controller:

- RecordController: Lớp điều khiển các phương thức hiển thị tới giao diện quản lý bệnh án bao gồm lấy thông tin từ model, việc xử lý thông tin trước hiển thị và việc xử lý thông tin đầu vào từ người dùng, thay đổi giao diện cho hợp lý với các thao tác giao diện của người dùng
- AppointmentController: Lớp điều khiển các phương thức hiển thị tới giao diện quản lý cuộc hẹn bao gồm lấy thông tin từ model, việc xử lý thông tin trước hiển thị và việc xử lý thông tin đầu vào từ người dùng, thay đổi giao diện cho hợp lý với các thao tác giao diện của người dùng
- MedicationController: Lớp điều khiển các phương thức hiển thị tới giao diện quản lý thuốc bao gồm lấy thông tin từ model, việc xử lý thông tin trước hiển thị và việc xử lý thông tin đầu vào từ người dùng, thay đổi giao diện cho hợp lý với các thao tác giao diện của người dùng
- DoctorController: Lớp điều khiển các phương thức hiển thị tới giao diện quản lý bệnh án bao gồm lấy thông tin từ model, việc xử lý thông tin trước hiển thị và thay đổi giao diện cho hợp lý với các thao tác giao diện của người dùng.

**Package Model:** Gói bao gồm các lớp với chức năng xử lý và lưu trữ thông tin trong hệ thống

- Record: Bệnh án
- Appointment: Cuộc hẹn
- Doctor: Bác sĩ
- Patient: Bệnh nhân
- Treatment: Điều trị
- Medication: Thuốc

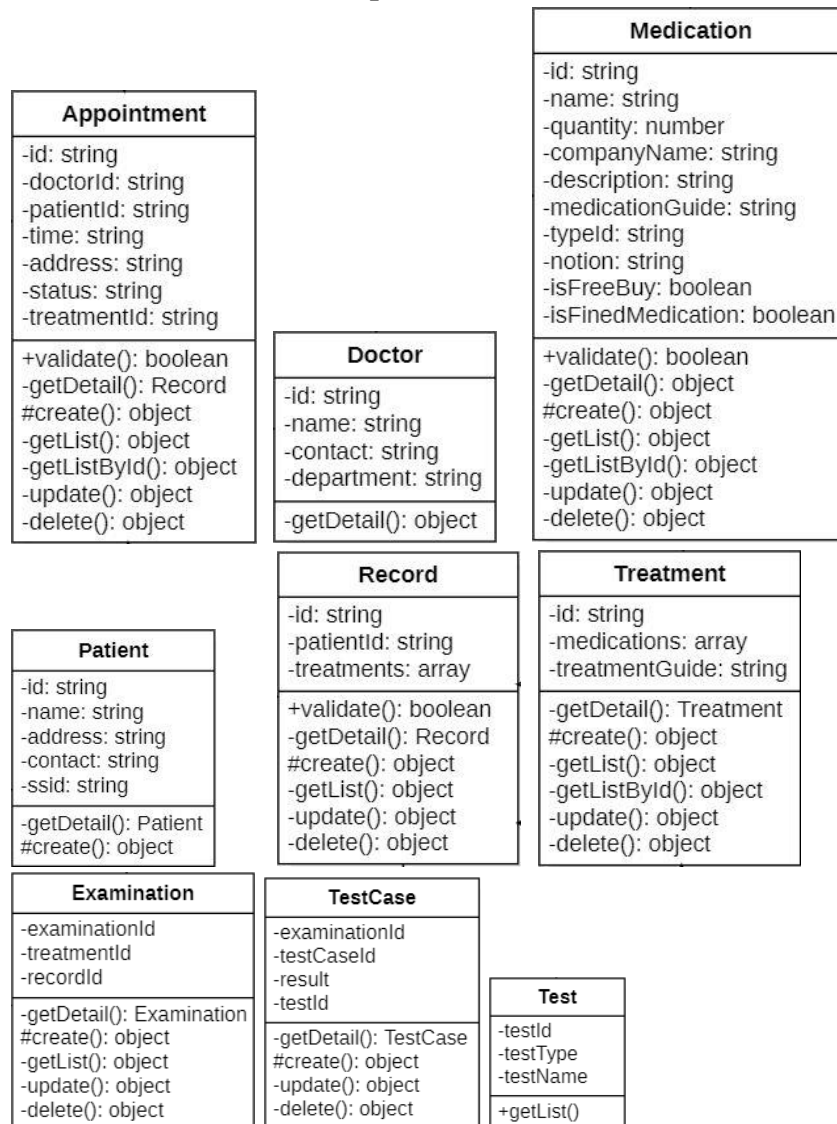
- Nhóm các lớp View:

AppointmentView	DoctorView	MedicationView	RecordView
-userId: string -name: string +buttonList +labelList +appointments: array	-userId: string -name: string +buttonList +labelList +appointments: array	-userId: string -name: string +buttonList +labelList +medications: array	-userId: string -name: string +buttonList +labelList +records: array
+GET() +PUT() +DELETE() +POST()	+GET() +PUT()	+GET() +PUT() +DELETE() +POST()	+GET() +PUT() +DELETE() +POST()

- Nhóm các lớp Controller:

AppointmentController	DoctorController
+validateForm(): boolean #getDetailAppointment(): object #getListAppointment(): object #updateAppointment(): object +createAppointment(): object #deleteAppointment(): object	+validateForm(): boolean +getListAppointmentById(): object #getDetailDoctor(): object
MedicationController	RecordController
+validateForm(): boolean #getDetailMedication(): object #getListMedication(): object #updateMedication(): object +createMedication(): object #deleteMedication(): object	+validateForm(): boolean #getDetailRecord(): object #getListRecord(): object #updateRecord(): object #deleteRecord(): object +createRecord(): object

- Nhóm các lớp Model:



b. Mô tả các mối quan hệ

No.	Type	Lớp	Mô tả
1.	Kết tập	Record - Patient	Một patient (bệnh nhân) thì có 1 và chỉ 1 record (bệnh án)
2.	Kết tập	Appointment - Record	Một appointment (cuộc hẹn) thì có 1 bệnh án được đối chiếu
3.	Hợp thành	Record – Examination	Một bệnh án thì có nhiều lần khám, các lần khám thuộc về bệnh án đó
4.	Kết tập	Appointment – Doctor	Một cuộc hẹn phải có sự tham gia của 1 bác sĩ
5.	Kết tập	Treatment – Medication	Một lần điều trị có 1 hoặc nhiều loại thuốc
6.	Hợp thành	Examination – Treatment	Một lần khám thì có một lần điều trị tương ứng với lần khám đó
7.	Hợp thành	Examination – TestCase	Một lần khám thì có nhiều lần xét nghiệm hoặc khám sơ bộ tương ứng
8.	Hợp thành	Test - TestCase	Một xét nghiệm được thực hiện nhiều lần đối với những lần khám khác nhau
9.	Liên hệ	Các lớp còn lại	Các lớp liên hệ với nhau thông qua việc sử dụng phương thức của nhau

### c. Chi tiết các lớp

#### Các lớp gói View

##### Lớp RecordView

- Bảng mô tả thuộc tính:

No.	Attribute	Valid	Type	Description
1.	userId		string	Mã người dùng
2.	name	Length < 64	string	Tên người dùng
3.	buttonList			Danh sách các nút trên giao diện
4.	labelList			Danh các các nhãn trên giao diện
5.	records		array	Danh sách các bệnh án

- Bảng mô tả phương thức:

No.	Method	Return value	Description
1.	GET()	object	Phương thức yêu cầu trả về giao diện danh sách bệnh án hoặc chi tiết bệnh án hoặc thông báo
2.	PUT()	object	Phương thức yêu cầu cập nhật bệnh án
3.	DELETE()	object	Phương thức yêu cầu xóa bệnh án
4.	POST()	object	Phương thức yêu cầu tạo mới bệnh án

##### Lớp AppointmentView

- Bảng mô tả thuộc tính:

No.	Attribute	Valid	Type	Description
1.	userId		string	Mã người dùng
2.	name	Length < 64	string	Tên người dùng
3.	buttonList			Danh sách các nút trên giao diện
4.	labelList			Danh các các nhãn trên giao diện
5.	appointments		array	Danh sách các cuộc hẹn

- Bảng mô tả phương thức:

No.	Method	Return value	Description
1.	GET()	object	Phương thức yêu cầu trả về giao diện danh sách cuộc hẹn hoặc chi tiết cuộc hẹn hoặc thông báo
2.	PUT()	object	Phương thức yêu cầu cập nhật cuộc hẹn
3.	DELETE()	object	Phương thức yêu cầu xóa cuộc hẹn
4.	POST()	object	Phương thức yêu cầu tạo mới cuộc hẹn

### Lớp MedicationView

- Bảng mô tả thuộc tính:

No.	Attribute	Valid	Type	Description
1.	userId		string	Mã người dùng
2.	name	Length < 64	string	Tên người dùng
3.	buttonList			Danh sách các nút trên giao diện
4.	labelList			Danh các các nhãn trên giao diện
5.	medications		array	Danh sách các thuốc

- Bảng mô tả phương thức:

No.	Method	Return value	Description
1.	GET()	object	Phương thức yêu cầu trả về giao diện danh sách thuốc hoặc chi tiết thuốc hoặc thông báo
2.	PUT()	object	Phương thức yêu cầu cập nhật thuốc
3.	DELETE()	object	Phương thức yêu cầu xóa thuốc
4.	POST()	object	Phương thức yêu cầu tạo mới thuốc

### Lớp DoctorView

- Bảng mô tả thuộc tính:

No.	Attribute	Valid	Type	Description
1.	userId		string	Mã người dùng
2.	name	Length < 64	string	Tên người dùng
3.	buttonList			Danh sách các nút trên giao diện
4.	labelList			Danh các các nhãn trên giao diện
5.	appointments		array	Danh sách các cuộc hẹn của bản thân bác sĩ

- Bảng mô tả phương thức:

No.	Method	Return value	Description
1.	GET()	object	Phương thức yêu cầu trả về giao diện các cuộc hẹn của bản thân bác sĩ hoặc thông báo
2.	PUT()	object	Phương thức yêu cầu cập nhật thông tin cá nhân bác sĩ

## Các lớp gói Controller

### Lớp RecordController

- Bảng mô tả phương thức:

No.	Method	Return value	Description
1.	validateForm()	boolean	Phương thức kiểm tra hợp lệ form input
2.	getDetailRecord()	object	Phương thức yêu cầu thông tin chi tiết của một bệnh án từ phía server
3.	getListRecord()	object	Phương thức yêu cầu thông tin về danh sách bệnh án từ server
4.	updateRecord()	object	Phương thức gửi yêu cầu cập nhật thông tin bệnh án đến server
5.	deleteRecord()	object	Phương thức gửi yêu cầu xóa bệnh án tới server
6.	createRecord()	object	Phương thức yêu cầu tạo mới bệnh án tới server

### Lớp AppointmentController

- Bảng mô tả phương thức:

No.	Method	Return value	Description
1.	validateForm()	boolean	Phương thức kiểm tra hợp lệ form input
2.	getDetailAppointment()	object	Phương thức yêu cầu thông tin chi tiết của một cuộc hẹn từ phía server
3.	getListAppointment()	object	Phương thức yêu cầu thông tin về danh sách cuộc hẹn từ server
4.	updateAppointment()	object	Phương thức gửi yêu cầu cập nhật thông tin cuộc hẹn đến server
5.	deleteAppointment()	object	Phương thức gửi yêu cầu xóa cuộc hẹn tới server
6.	createAppointment()	object	Phương thức yêu cầu tạo mới cuộc hẹn tới server



### Lớp MedicationController

- Bảng mô tả phương thức:

No.	Method	Return value	Description
1.	validateForm()	boolean	Phương thức kiểm tra hợp lệ form input
2.	getDetailMedication()	object	Phương thức yêu cầu thông tin chi tiết của một thuốc từ phía server
3.	getListMedication()	object	Phương thức yêu cầu thông tin về danh sách thuốc từ server
4.	updateMedication()	object	Phương thức gửi yêu cầu cập nhật thông tin thuốc đến server
5.	deleteMedication()	object	Phương thức gửi yêu cầu xóa thuốc tới server
6.	createMedication()	object	Phương thức yêu cầu tạo mới thuốc tới server

### Lớp DoctorController

- Bảng mô tả phương thức:

No.	Method	Return value	Description
1.	validateForm()	boolean	Phương thức kiểm tra hợp lệ form input
2.	getListAppointmentById()	object	Phương thức yêu cầu thông tin về danh sách cuộc hẹn từ server của bản thân bác sĩ
3.	getDetailDoctor()	object	Phương thức yêu cầu thông tin chi tiết của bản thân bác sĩ

## Các lớp gói Model

### Lớp Record

- Bảng mô tả thuộc tính:

No	Attribute	Valid	Type	Description
1.	id		string	Mã bệnh án
2.	patientId		string	Mã bệnh nhân
3.	examinations		array	Danh sách lần khám

- Bảng mô tả phương thức:

No	Method	Return value	Description
1.	validate()	boolean	Xác nhận xem bệnh nhân đã có bệnh án chưa
2.	getDetail()	Record	Trả về thông tin chi tiết của bệnh án
3.	create()	object	Tạo mới bệnh án
4.	getList()	object	Trả về danh sách bệnh án
5.	update()	object	Cập nhật thông tin bệnh án
6.	delete()	object	Xóa bệnh án

### Lớp Patient

- Bảng mô tả thuộc tính:

No	Attribute	Valid	Type	Description
1.	id		string	Mã bệnh nhân
2.	name	Length < 64	string	Tên bệnh nhân
3.	address	Length < 256	string	Địa chỉ bệnh nhân
4.	contact	Length < 64	string	Số liên lạc của bệnh nhân
5.	ssid	Length < 32	string	Mã an sinh xã hội

- Bảng mô tả phương thức:

No	Method	Return value	Description
1.	getDetail()	Patient	Phương thức yêu cầu trả về thông tin chi tiết bệnh nhân
2.	create()	object	Phương thức yêu cầu tạo mới bệnh nhân

### Lớp Treatment

- Bảng mô tả thuộc tính:

No	Attribute	Valid	Type	Description
1.	id		string	Mã điều trị
2.	examinationId		string	Mã lần khám
3.	medications		array	Các thuốc trong quá trình điều trị
4.	treatmentGuide	Length < 2048	string	Hướng dẫn điều trị

- Bảng mô tả phương thức:

No	Method	Return value	Description
1.	getDetail()	Treatment	Trả về thông tin chi tiết của điều trị
2.	create()	object	Tạo mới điều trị
3.	getList()	object	Trả về danh sách điều trị
4.	getListById()	object	Trả về danh sách điều trị theo mã bệnh án
5.	update()	object	Cập nhật điều trị
6.	delete()	object	Xóa điều trị

### Lớp Appointment

- Bảng mô tả thuộc tính:

No	Attribute	Valid	Type	Description
1.	id		string	Mã cuộc hẹn
2.	doctorId		string	Mã bác sĩ
3.	patientId		string	Mã bệnh nhân
4.	time	time > Date.now()	string	Thời gian cuộc hẹn
5.	address	Length < 256	string	Địa chỉ cuộc hẹn
6.	status	Length < 32	string	Trạng thái cuộc hẹn
7.	treatmentId		string	Mã điều trị

- Bảng mô tả phương thức:

No	Method	Return value	Description
1.	validate()	boolean	Xác nhận xem cuộc hẹn có bị trùng thời gian với cuộc hẹn khác chưa
2.	getDetail()	Appointment	Trả về thông tin chi tiết của cuộc hẹn
3.	create()	object	Tạo mới cuộc hẹn
4.	getList()	object	Trả về danh sách cuộc hẹn
5.	getListById()	object	Trả về danh sách cuộc hẹn theo mã bác sĩ
6.	update()	object	Cập nhật cuộc hẹn
7.	delete()	object	Xóa cuộc hẹn

Lớp Doctor

- Bảng mô tả thuộc tính:

No	Attribute	Valid	Type	Description
1.	id		string	Mã bác sĩ
2.	name	Length < 64	string	Tên bác sĩ
3.	contact	Length < 64	string	Số liên lạc của bác sĩ
4.	department	Length < 128	string	Khoa

- Bảng mô tả phương thức:

No	Method	Return value	Description
1.	getDetail()	object	Trả về thông tin chi tiết của bác sĩ

### Lớp Medication

- Bảng mô tả thuộc tính:

No	Attribute	Valid	Type	Description
1.	id		string	Mã thuốc
2.	name	Length < 64	string	Tên thuốc
3.	quantity	quantity >= 0	number	Số lượng
4.	companyName	Length < 256	string	Công ty sản xuất
5.	description	Length < 2048	string	Mô tả
6.	medicationGuide	Length < 256	string	Hướng dẫn sử dụng thuốc
7.	typeId	Length < 32	string	Loại thuốc
8.	notion	Length < 256	string	Lưu ý khi sử dụng
9.	isFreeBuy		boolean	Được mua tự do ?
10.	isFinMedication		boolean	Loại thuốc ổn định giá ?

- Bảng mô tả phương thức:

No	Method	Return value	Description
1.	validate()	boolean	Xác nhận xem thuốc có tồn tại hay chưa
2.	getDetail()	Appointment	Trả về thông tin chi tiết của thuốc
3.	create()	object	Tạo mới thuốc
4.	getList()	object	Trả về danh sách thuốc
5.	getListById()	object	Trả về danh sách thuốc theo mã điều trị
6.	update()	object	Cập nhật thuốc
7.	delete()	object	Xóa thuốc

### Lớp Examination

- Bảng mô tả thuộc tính:

No	Attribute	Valid	Type	Description
1.	examinationId		string	Mã lần khám
2.	treatmentId		string	Mã chữa trị
3.	recordId		number	Mã bệnh án

- Bảng mô tả phương thức:

No	Method	Return value	Description
1.	getList()	boolean	Lấy danh sách lần khám của một bệnh án tương ứng
2.	getDetail()	Appointment	Trả về thông tin chi tiết lần khám
3.	create()	object	Thêm mới lần khám
4.	update ()	object	Cập nhật lần khám
5.	delete()	object	Xóa lần khám

### Lớp TestCase

- Bảng mô tả thuộc tính:

No	Attribute	Valid	Type	Description
1.	examinationId		string	Mã lần khám
2.	testCaseId			
3.	result	Length < 128	string	Mã chữa trị
4.	testId		number	Mã xét nghiệm

- Bảng mô tả phương thức:

No	Method	Return value	Description
1.	getDetail()	TestCase	Trả về thông tin chi tiết lần xét nghiệm
2.	create()	object	Thêm mới lần xét nghiệm
3.	update ()	object	Cập nhật lần xét nghiệm
4.	delete()	object	Xóa lần xét nghiệm

### Lớp Test

- Bảng mô tả thuộc tính:

No	Attribute	Valid	Type	Description
1.	testName	Length < 64	string	Tên xét nghiệm
2.	testType	Length < 64	string	Loại xét nghiệm
3.	testId		number	Mã xét nghiệm

- Bảng mô tả phương thức:

No	Method	Return value	Description
1.	getList()		Trả về danh sách các loại xét nghiệm

### 3.2.5. Thiết kế cơ sở dữ liệu

Danh sách các bảng:

No	Tên thực thể	Diễn giải
1.	Record	Ghi lại thông tin về bệnh án
2.	Patient	Ghi lại thông tin về bệnh nhân
3.	Treatment	Ghi lại thông tin về các lần điều trị của bệnh nhân
4.	Appointment	Ghi lại thông tin về các cuộc hẹn
5.	Doctor	Ghi lại thông tin về bác sĩ
6.	Medication	Ghi lại thông tin về thuốc
7.	Examination	Ghi lại thông tin về lần khám
8.	TestCase	Ghi lại thông tin về xét nghiệm
9.	Test	Thông tin các loại xét nghiệm

a. Quan hệ các bảng dữ liệu

No	Quan hệ	Bảng 1	Bảng 2	Giải thích
1.	1 – 1	Record	Patient	1 bệnh nhân có 1 và chỉ 1 bệnh án duy nhất
2.	1 – n	Record	Examination	1 bệnh án của 1 bệnh nhân có nhiều lần khám theo thời gian
3.	1 – n	Record	Appointment	Một bệnh án xuất hiện trong nhiều lần hẹn gặp bác sĩ
4.	1 – 1	Doctor	Appointment	Một cuộc hẹn có 1 bác sĩ
5.	1 – n	Treatment	Medication	Một lần điều trị dùng 1 đến nhiều thuốc
6.	1 – 1	Examination	Treatment	Một lần khám có 1 lần điều trị tương ứng
7.	1 – n	Examination	TestCase	Một lần khám có nhiều lần xét nghiệm
8.	1 – n	Test	TestCase	Một loại xét nghiệm được sử dụng để xét nghiệm nhiều lần

b. Chi tiết các bảng

Bảng 1: Record

No	Thông tin	Quy định			Ràng buộc	Cho phép để rỗng	Chuẩn hóa
		Tên trường	Kiểu	Độ rộng			
1.	Mã bệnh án	id	string	64			x
2.	Mã bệnh nhân	patientId	string	64			x
3.	Danh sách điều trị	treatments	array				x



Bảng 2: Patient

No	Thông tin	Quy định			Ràng buộc	Cho phép để rỗng	Chuẩn hóa
		Tên trường	Kiểu	Độ rộng			
1.	Mã bệnh nhân	id	string	64			x
2.	Tên bệnh nhân	name	string	64			x
3.	Địa chỉ bệnh nhân	addresses	string	256			x
4.	Liên lạc bệnh nhân	contact	string	256			x
5.	Mã an sinh xã hội	ssid	string	32			x

Bảng 3: Treatment

No	Thông tin	Quy định			Ràng buộc	Cho phép để rỗng	Chuẩn hóa
		Tên trường	Kiểu	Độ rộng			
1.	Mã điều trị	id	string	64			x
2.	Tên thuốc	medications	array				x
3.	Hướng dẫn điều trị	treatmentGuide	string	2048			x

Bảng 4: Appointment

No	Thông tin	Quy định			Ràng buộc	Cho phép để rỗng	Chuẩn hóa
		Tên trường	Kiểu	Độ rộng			
1.	Mã cuộc hẹn	id	string	64			x
2.	Mã bác sĩ	doctorId	string	64			x
3.	Mã bệnh nhân	patientId	string	64			x
4.	Thời gian hẹn	time	string				x
5.	Địa chỉ hẹn	address	string	256			x
6.	Trạng thái cuộc hẹn	status	string	32			x
7.	Mã điều trị	treatmentId	string	64			x

Bảng 5: Doctor

No	Thông tin	Quy định			Ràng buộc	Cho phép để rỗng	Chuẩn hóa
		Tên trường	Kiểu	Độ rộng			
1.	Mã bác sĩ	id	string	64			x
2.	Tên bác sĩ	name	string	64			x
3.	Số liên lạc của bác sĩ	contact	string	64			x
4.	Khoa	department	string	128			x

Bảng 6: Medication

No	Thông tin	Quy định			Ràng buộc	Cho phép để rỗng	Chuẩn hóa
		Tên trường	Kiểu	Độ rộng			
1.	Mã thuốc	id	string	64			x
2.	Tên thuốc	name	string	64			x
3.	Số lượng	quantity	number	0			x
4.	Công ty sản xuất	companyName	string	256			x
5.	Mô tả	description	string	2048			x
6.	Hướng dẫn sử dụng thuốc	medicationGuide	string	256			x
7.	Loại thuốc	typeId	string	32			x
8.	Lưu ý khi sử dụng	notion	string	256			x
9.	Được mua tự do ?	isFreeBuy	boolean	1			x
10.	Loại thuốc ổn định giá ?	isFixedMedication	boolean	1			x

Bảng 7: Examination

No	Thông tin	Quy định			Ràng buộc	Cho phép để rỗng	Chuẩn hóa
		Tên trường	Kiểu	Độ rộng			
1.	Mã lần khám	examinationId	string	64			x
2.	Mã điều trị	treatmentId	string	64			x
3.	Mã bệnh án	recordId	string	64			x

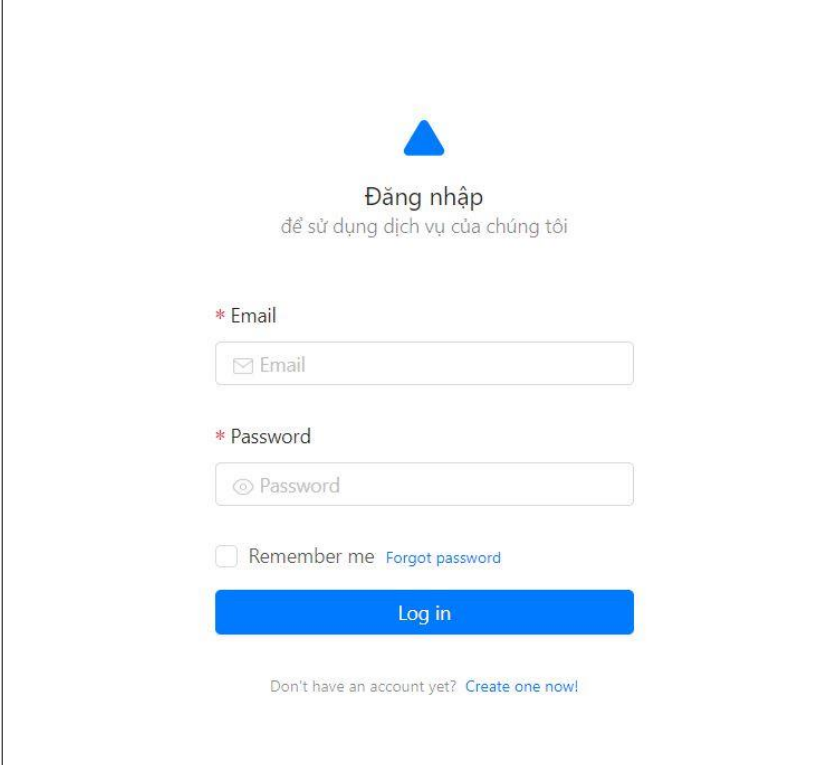
Bảng 8: TestCase

No	Thông tin	Quy định			Ràng buộc	Cho phép để rỗng	Chuẩn hóa
		Tên trường	Kiểu	Độ rộng			
1.	Mã lần khám	examinationId	string	64			x
2.	Mã lần xét nghiệm	testCaseId	string	64			x
3.	Kết quả xét nghiệm	result	string	128			x
4.	Mã xét nghiệm	testId	string	64			x

Bảng 9: Test

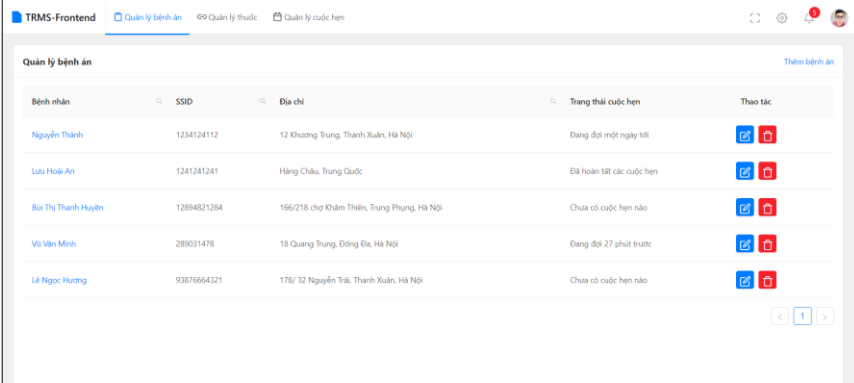
No	Thông tin	Quy định			Ràng buộc	Cho phép để rỗng	Chuẩn hóa
		Tên trường	Kiểu	Độ rộng			
1.	Mã xét nghiệm	testId	string	64			x
2.	Loại xét nghiệm	testType	string	64			x
3.	Tên xét nghiệm	testName	string	64			x

### 3.3. Thiết kế giao diện



The login form features a blue triangle icon at the top. Below it, the text "Đăng nhập" (Login) is centered, followed by "để sử dụng dịch vụ của chúng tôi" (to use our service). The form includes two input fields: "Email" and "Password", both preceded by an asterisk. Below the password field is a "Remember me" checkbox and a "Forgot password" link. A blue "Log in" button is positioned below these fields. At the bottom, there is a link "Don't have an account yet? Create one now!".

Figure 23: Đăng nhập



The table displays a list of patients with columns for name, SSID, address, appointment status, and actions. The data is as follows:











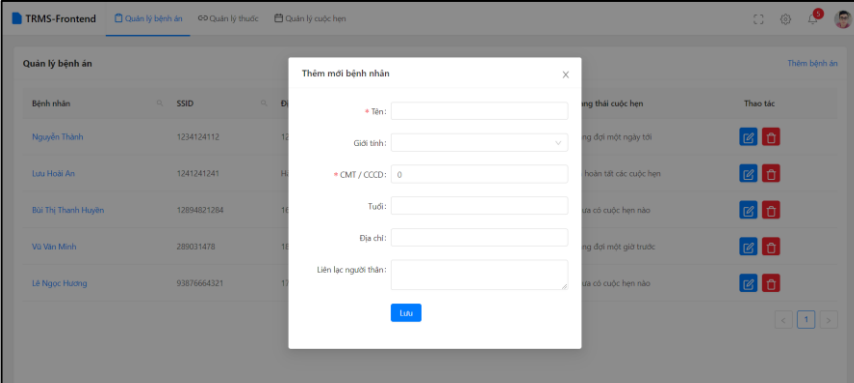
Bệnh nhân	SSID	Địa chỉ	Trạng thái cuộc hẹn	Thao tác
Nguyễn Thành	1234124112	12 Đường Trung, Thanh Xuân, Hà Nội	Đang đợi một ngày tới	 
Lưu Hoài An	1241241241	Hàng Châu, Trung Quốc	Đã hoàn tất các cuộc hẹn	 
Bùi Thị Thanh Huyền	12894821284	166/218 chợ Khâm Thiên, Trung Phụng, Hà Nội	Chưa có cuộc hẹn nào	 
Vũ Văn Minh	289031478	18 Quang Trung, Đống Đa, Hà Nội	Đang đợi 27 phút trước	 
Lê Ngọc Hương	93876664321	178/ 32 Nguyễn Trãi, Thanh Xuân, Hà Nội	Chưa có cuộc hẹn nào	 

Figure 24: Danh sách bệnh án



The form for adding a new patient is titled "Thêm mới bệnh nhân". It includes fields for: "Tên" (Name), "Giới tính" (Gender), "CMT / CCCD" (ID Number), "Tuổi" (Age), "Địa chỉ" (Address), and "Liên lạc người thân" (Contact person). A "Lưu" (Save) button is at the bottom.

Figure 25: Form thêm mới bệnh án

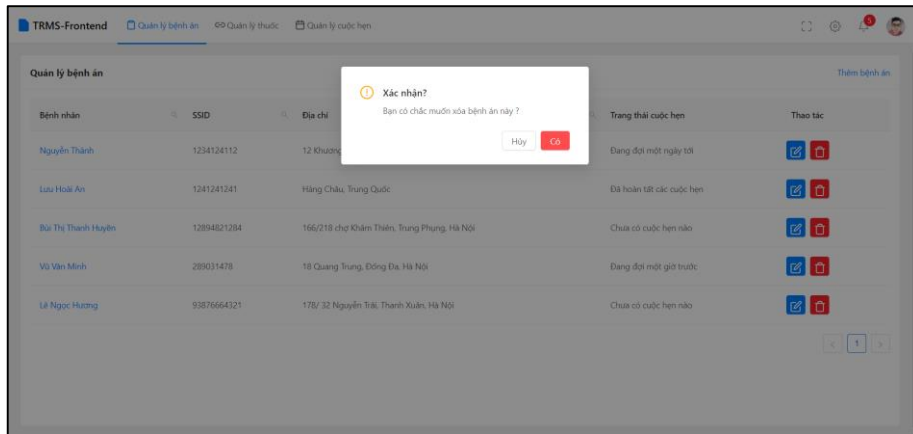


Figure 26: Popup xác nhận xóa bệnh án

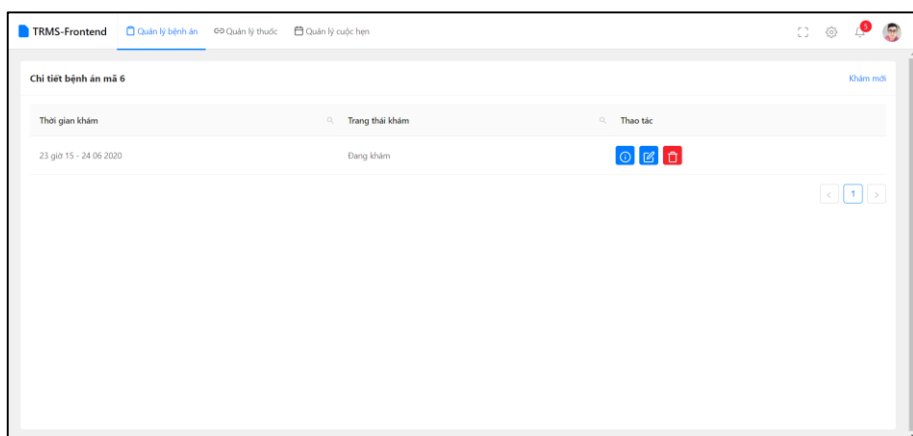


Figure 27: Chi tiết bệnh án (Danh sách lần khám)

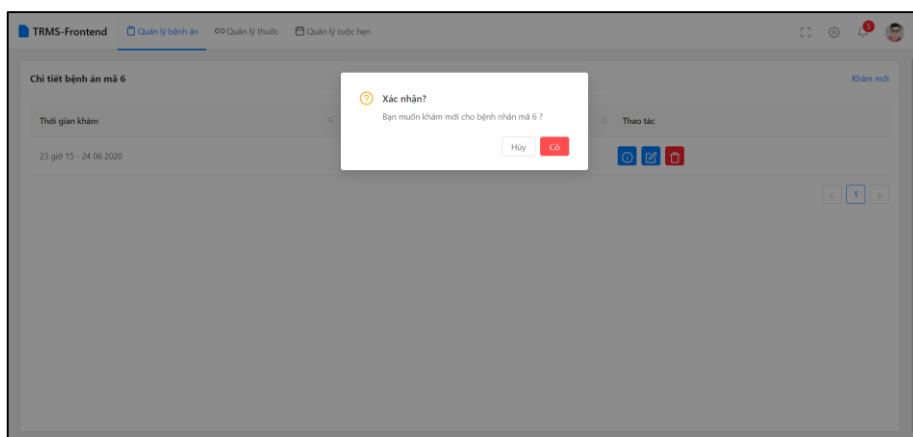


Figure 28: Popup xác nhận xóa lần khám

Figure 29: Form thông tin lần khám: khám sơ bộ

Figure 30: Form thông tin lần khám: Chẩn đoán

Figure 31: Form thông tin lần khám: Điều trị

Tên thuốc	Số lượng	Tên công ty sản xuất	Loại thuốc	Được mua tự do	Loại thuốc ấn định giá	Thao tác
EtorcB6S21	10000	Công ty cổ phần dược phẩm Hà Tây	Thuốc kê đơn - null	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<a href="#">Thêm</a> <a href="#">Xóa</a>
EtorcB1	10000	Công ty cổ phần dược phẩm Hà Tây	Thuốc kê đơn - null	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<a href="#">Thêm</a> <a href="#">Xóa</a>
Sorbitol delalande	10000	Công ty Cổ phần Dược phẩm Senohi Synthelabo Việt Nam (bản nhận giá công)	null - Tên được	<input type="checkbox"/>	<input type="checkbox"/>	<a href="#">Thêm</a> <a href="#">Xóa</a>
Dodevitart	10000	Công ty TNHH SX DP MEDICAL PHARMA ITALY	Thuốc kê đơn - null	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<a href="#">Thêm</a> <a href="#">Xóa</a>
An thần ACP	10000	Công ty cổ phần dược phẩm Me Di Sun (Cổ sở nhận giá công)	Thuốc không kê đơn - null	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<a href="#">Thêm</a> <a href="#">Xóa</a>
Ibatonic-Calcit	10000	(Cổ sở nhận giá công), Công ty cổ phần	null - Tên được	<input type="checkbox"/>	<input type="checkbox"/>	<a href="#">Thêm</a> <a href="#">Xóa</a>

Figure 32: Danh sách thuốc

Tên thuốc:

Số lượng: 0

Công ty sản xuất:

Mô tả:

Hướng dẫn sử dụng thuốc:

Loại thuốc:

Lưu ý khi dùng:

Được mua tự do: ☒

Figure 33: Form thêm mới thuốc

Bác sĩ	Bệnh nhân	Thời gian hẹn	Địa chỉ	Trạng thái	Thao tác
Vũ Lê Hoàng	Vũ Văn Minh	22 phút trước	701 B1 Huat	Đang đợi	<a href="#">Thêm</a> <a href="#">Xóa</a>
Vũ Lê Hoàng	Nguyễn Thành	3 giờ trước	605 Huyện Thuat Khang	Đang đợi	<a href="#">Thêm</a> <a href="#">Xóa</a>
Vũ Lê Hoàng	Lưu Hoài An	8 giờ trước	704 B1	Đang đợi	<a href="#">Thêm</a> <a href="#">Xóa</a>
Vũ Lê Hoàng	Lưu Hoài An	7 ngày trước	Phong 501 Nha B1	Đang đợi	<a href="#">Thêm</a> <a href="#">Xóa</a>

Figure 34: Danh sách cuộc hẹn

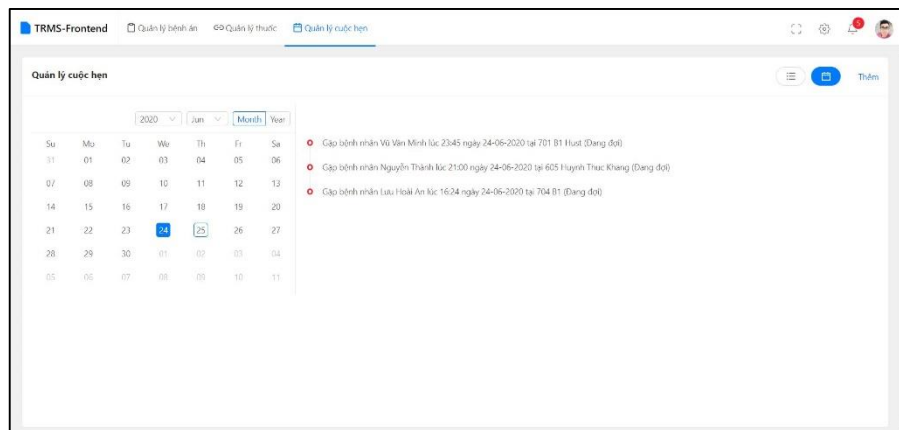


Figure 35: Danh sách cuộc hẹn theo lịch

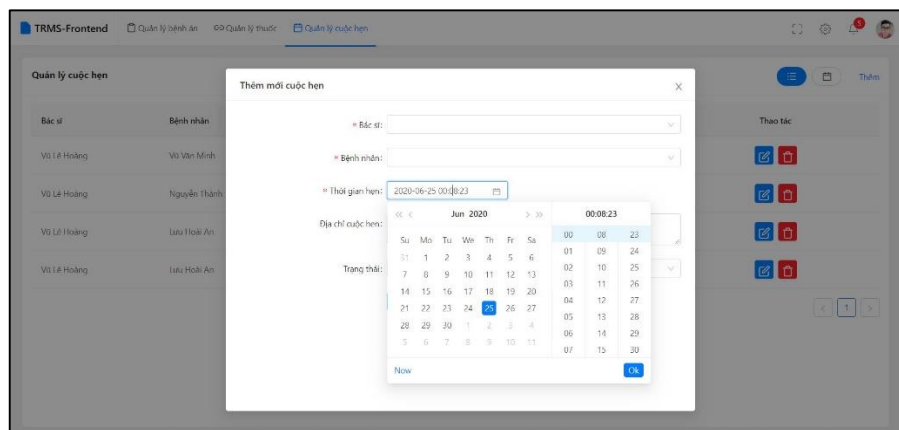


Figure 36: Form thêm mới cuộc hẹn

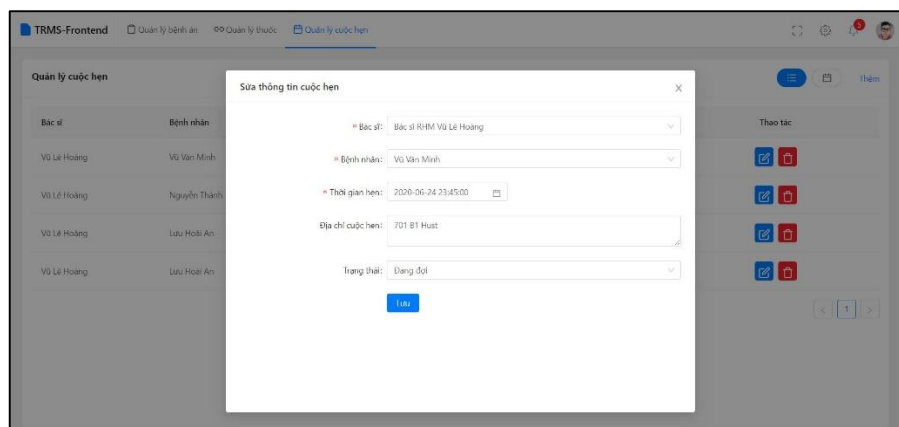


Figure 37: Form Sửa thông tin cuộc hẹn



## Chương 4: Kết luận

### 4.1. Kết luận

Sau đây là những kết quả em đã đạt được trong quá trình thực hiện thực hiện đồ án tốt nghiệp trong thời gian qua.

Về quy trình, em đã thực hiện quy trình theo mô hình phát triển thác nước, nhằm nắm rõ hơn về mô hình phát triển phần mềm trong tương lai. Tuy đồ án nằm trong mục đồ án cá nhân nhưng nhờ phát triển theo mô hình phát triển một cách chặt chẽ, em đã phát triển các tài liệu thống nhất về mặt thông tin và dữ liệu với nhau, đồng thời cũng quá trình phát triển cũng rõ ràng hơn.

Về cấu trúc, em triển khai hệ thống theo mô hình micro-service, tách back-end và front-end ra riêng, nhằm đáp ứng nhu cầu về mở rộng trong tương khai khi phải thay đổi hay cập nhật công nghệ cho phía back-end hay fron-end thì hệ thống vẫn hoạt động tốt

Về công nghệ, phía back-end em sử dụng Django kết hợp với graphql. Trong đó graphql là một loại truy vấn dữ liệu dạng cây, giúp câu truy vấn được dễ dàng hơn khi cần thực hiện các truy vấn cần lấy thông tin từ nhiều bảng khác nhau.

Cũng với đó, em sử dụng thêm bảo mật jwt (JsonWebToken) kết hợp cùng graphql để tăng tính bảo mật cho hệ thống. Còn phía front-end, để tương tác graphql với phía backend, em sử dụng module Apollo-graphql kết hợp với reactjs và react-hook. Việc sử dụng hook nhằm kiểm soát tối đa việc binding DOM cũng như render dữ liệu thừa và giảm kích thước bundle khi tải trên phía client. Làm việc cùng framework loại server side render cũng cho em kinh nghiệm về xử lý cookie và token xác thực của người dùng.

Về sản phẩm, hệ thống của em có thể quản lý người dùng, quản lý thuốc, quản lý bệnh án cho bệnh nhân và các cuộc hẹn của bác sĩ đối với bệnh nhân. Ngoài ra hệ thống còn có xác thực tài khoản đăng nhập bằng lưu xác thực ở cookie.

Về hạn chế, hệ thống chưa có khả năng chịu tải cao ở phía server và giao diện chưa được đẹp ở phía client. Luồng thiết kế của hệ thống cũng chưa được tối ưu cho người sử dụng, ở đây đặc biệt là bác sĩ và y tá. Ngoài ra tính ổn định của hệ thống chưa cao, thể hiện ở việc các thao tác xử lý chậm, sai, và việc xử lý lỗi chưa được tốt.

## 4.2. Hướng phát triển

Trước tiên, để hệ thống có thể phát triển, cần khắc phục các hạn chế đã nêu trên:

- Vấn đề về thiết kế hệ thống chịu tải cao do bệnh viện có nhiều phòng ban và bệnh nhân, cần sử dụng queue để tách nhỏ lượng request lên server và tiến hành xử lý bất đồng bộ. Ở đây công nghệ đề xuất sẽ là sử dụng redis kết hợp framework hiện tại là Django. Ngoài ra redis cũng hỗ trợ cache dữ liệu làm giảm khối lượng tính toán cho server
- Vấn đề về dữ liệu chậm do truy vấn, có thể sử dụng pagination cho dữ liệu, đồng thời với việc sử dụng các search engine trên giao diện, có thể áp dụng thêm ELK stack (ElasticSearch) để tăng hiệu quả truy vấn trên bảng khi dữ liệu lớn
- Vấn đề về front-end, cần thiết kế lại giao diện sao cho người dùng thực hiện được mục đích với ít thao tác nhất
- Vấn đề về xử lý lỗi cần được thực hiện trên cả server và client. Hiện tại trên server đã xử lý và gửi lỗi về cho phía client, tuy nhiên client chưa xử lý được lỗi sao cho thân thiện với người dùng

Sau đó, về phát triển thêm, hệ thống có thể phát triển thành app trên di động để thuận tiện cho việc sử dụng linh hoạt cho bệnh nhân. (Bệnh nhân có thể sử dụng app để tra cứu thay vì sử dụng web) hoặc chức năng nhận diện khuôn mặt bệnh nhân khi bệnh nhân đi khám

## Tài liệu tham khảo

[1] Tỷ lệ dân số già ở Việt Nam:

[http://vnca.molisa.gov.vn/default.aspx?page=news&do=detail&category\\_id=76&id=558](http://vnca.molisa.gov.vn/default.aspx?page=news&do=detail&category_id=76&id=558)

[2] Tỷ lệ mắc bệnh mãn tính ở người già: <https://baotintuc.vn/van-de-quan-tam/cham-soc-y-te-cho-nguoi-cao-tuoi-bai-1-trung-binh-mot-nguoi-gia-mac-3-benh-man-tinh-20171001075734729.htm>

## Phụ lục

### A1. Dữ liệu thuốc

Dữ liệu về thuốc được lấy từ Ngân hàng dữ liệu ngành dược tại

<https://drugbank.vn/>

Thông qua public api:

<https://drugbank.vn/services/drugbank/api/thuoc>

**Bản quyền dữ liệu thuốc về: Cục quản lý dược – Bộ Y tế**

### A2. Dữ liệu bệnh án

Dữ liệu về bệnh án được sử dụng dưới sự cho phép của bệnh viện

Đa khoa Y học cổ truyền Thái Nguyên với mục đích phục vụ nghiên cứu và công tác giảng dạy

### A3. Mẫu API

Mẫu api sử dụng trong chương trình được khai báo theo loại graphql, sử dụng postman export engine, được public tại:

<https://github.com/hoangvlbk61/HUST-Final-project-2019B-TRMS/blob/dev/Doc/Api-postman-sample.json>

**Bản quyền dữ liệu thuốc về: Vũ Lê Hoàng – 20166163 – CN CNTT2 K 61 – MIT License 2.0**

### A4. Source Code

Source code được viết bởi **Vũ Lê Hoàng – 20166163 – CN**

**CNTT2 K 61 – MIT License 2.0**

<https://github.com/hoangvlbk61/HUST-Final-project-2019B-TRMS/tree/dev>

### A5. Hướng dẫn chạy chương trình

Tài liệu được viết bởi **Vũ Lê Hoàng – 20166163 – CN CNTT2 K 61 – MIT License 2.0**

<https://github.com/hoangvlbk61/HUST-Final-project-2019B-TRMS/blob/dev/README.md>