

Trường Đại học Công nghệ - ĐHQGHN
Khoa Công nghệ thông tin

BÀI TẬP LỚN: PHÂN TÍCH & THIẾT KẾ HƯỚNG ĐỐI TƯỢNG

Giảng viên: Đặng Đức Hạnh

Trần Mạnh Cường



ARCHITECTURE ANALYSIS

ỨNG DỤNG CHĂM SÓC SỨC KHỎE TRỰC TUYẾN

Ngày: 01/04/2024

Chuẩn bị bởi: Nhóm 5

Mục lục

Lịch sử sửa đổi	3
1. Tổng quan	4
1.1. Giới thiệu	4
1.2. Đối tượng dự kiến và đề xuất cách đọc	4
1.3. Phạm vi dự án	6
2. Cơ chế phân tích	6
2.1. Cơ chế về tính nhất quán	6
2.2. Cơ chế về việc giao tiếp	7
2.3. Cơ chế về bảo mật	7
2.4. Các cơ chế khác	8
3. Các trừu tượng chính	9
3.1. Biểu đồ	9
3.2. Định nghĩa	10
4. Biểu diễn kiến trúc hệ thống	11
5. Mục tiêu và ràng buộc	11
6. Khung nhìn ca sử dụng	11
6.1. Hiện thực hóa Use case	12
6.2. Mô tả Use case quan trọng	15
7. Khung nhìn logic	16
7.1. Tổng quan	16
7.2. Các gói thiết kế kiến trúc	17
7.2.1. Gói giao diện	17
7.2.2. Gói ứng dụng	19
7.2.4. Gói nhất quán	23
8. Khung nhìn tiến trình	23
8.1. Mô hình tiến trình	23
8.2. Mô tả các phần tử tiến trình	24
9. Khung nhìn triển khai	25
10. Quy mô và hiệu năng	26
11. Chất lượng	27

Lịch sử sửa đổi

Họ tên	Thời gian	Lý do sửa đổi	Phiên bản
Dương Nguyễn Việt Anh	01/04/2024	Khởi tạo tài liệu	1.0
Nguyễn Thị Ngọc Ánh	05/04/2024	Viết, hoàn thiện nội dung phần 2, phần 4, phần 5, phần 10 & 11	1.1
Dương Nguyễn Việt Anh	06/04/2024	Viết, hoàn thiện nội dung phần 3	1.2
Lê Trọng Minh	06/04/2024	Viết, hoàn thiện nội dung phần 6	1.3
Phạm Hoàng Hải	06/04/2024	Viết, hoàn thiện nội dung phần 7	1.4
Hoàng Văn Nguyên	06/04/2024	Viết, hoàn thiện nội dung phần 9	1.5

1. Tổng quan

1.1. Giới thiệu

Đây là một báo cáo về chủ đề Phân tích và thiết kế hướng đối tượng của nhóm 5 (về sau gọi tắt là nhóm tác giả) về Kiến trúc hệ thống.

Tài liệu được viết dựa theo định dạng báo cáo “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”.

Tài liệu này cung cấp một tổng quan toàn diện về kiến trúc của hệ thống, sử dụng một số các khung nhìn kiến trúc khác nhau để mô tả các khía cạnh khác nhau của hệ thống.

1.2. Đối tượng dự kiến và đề xuất cách đọc

Các đối tượng đọc khác nhau dành cho tài liệu này là:

- Nhà phát triển: Người thực hiện nhiệm vụ phát triển hệ thống từ đầu vào là bản thiết kế và tài liệu để tạo thành đầu ra là một phiên bản có thể chạy được.
- Khách hàng: Khách hàng là người đặt hàng hệ thống và muốn có một hệ thống mới (system-to-be) tốt hơn hệ thống hiện thời (system-as-is). Trong khóa học này, khách hàng có thể coi như là giáo viên.
- Người viết tài liệu: Người sẽ viết tài liệu trong tương lai (các báo cáo, biên bản).

Tài liệu cung cấp một mô tả tổng quan về các mục tiêu của kiến trúc, các ca sử dụng hỗ trợ bởi hệ thống và các kiểu và thành phần kiến trúc đã được chọn để đạt được các ca sử dụng phù hợp nhất. Khung làm việc này sau đó cho phép phát triển các tiêu chí thiết kế và tài liệu xác định các tiêu chuẩn kỹ thuật và miền một cách chi tiết.

Tài liệu này giúp người đọc có được cái nhìn tổng quan về kiến trúc hệ thống. Các nội dung chính trong báo cáo bao gồm: xác định các cơ chế phân tích, các trừu tượng chính, và các phần kiến trúc chính:

- Phần 1 - Biểu diễn kiến trúc: Phần này mô tả kiến trúc phần mềm nào cho hệ thống hiện tại và cách nó được trình bày. Bao gồm cả sử dụng, khung nhìn logic, khung nhìn tiến trình, khung nhìn triển khai và khung nhìn thực thi nó liệt kê các khung nhìn cần thiết và cho mỗi khung nhìn, giải thích các loại phân tử mô hình mà nó chứa.
- Phần 2 - Các mục tiêu và ràng buộc về kiến trúc: Phần này mô tả các yêu cầu và mục tiêu phần mềm có ảnh hưởng đáng kể đến kiến trúc, ví dụ: an toàn, bảo mật, quyền riêng tư, sử dụng sản phẩm có sẵn, tính di động, phân phối và tái sử dụng. Nó cũng nắm bắt các ràng buộc đặc biệt có thể áp dụng: chiến lược thiết kế và triển khai, các công cụ phát triển, cấu trúc nhóm, lịch biểu, mã kế thừa, v.v.
- Phần 3 - Khung nhìn ca sử dụng: Phần này liệt kê các trường hợp sử dụng hoặc kịch bản từ mô hình ca sử dụng nếu chúng thể hiện một số chức năng trung tâm quan trọng của hệ thống cuối cùng hoặc nếu chúng có độ bao phủ kiến trúc lớn - chúng thực hiện nhiều yếu tố kiến trúc hoặc nếu chúng nhấn mạnh hoặc minh họa cụ thể, điểm tinh tế của kiến trúc.
- Phần 4 - Khung nhìn logic: Phần này mô tả các phần có ý nghĩa về mặt kiến trúc của mô hình thiết kế, chẳng hạn như phân tách thành các hệ thống con và gói. Và đối với mỗi gói quan trọng, phân tách của nó thành các lớp và các tiện ích lớp.
- Phần 5 - Khung nhìn tiến trình: Phần này mô tả sự phân rã của hệ thống thành các quy trình nhẹ (các luồng điều khiển đơn) và các quy trình nặng (nhóm các quy trình nhẹ). Tổ chức các phần theo nhóm các tiến trình giao tiếp hoặc tương tác. Mô tả các chế độ giao tiếp chính giữa các tiến trình, chẳng hạn như chuyển tin nhắn, ngắt và điểm hẹn.
- Phần 6 - Khung nhìn triển khai: Phần này mô tả một hoặc nhiều tiến trình cấu hình mạng vật lý (phần cứng) khi phần mềm được triển khai và chạy.
- Phần 7 - Khung nhìn thực thi: Phần này mô tả cấu trúc tổng thể của mô hình triển khai, phân tách phần mềm thành các lớp và hệ thống

con trong mô hình triển khai và bất kỳ thành phần quan trọng nào về mặt kiến trúc.

- Phần 8 - Quy mô và hiệu năng: Mô tả về các đặc điểm kích thước chính của phần mềm tác động đến kiến trúc, cũng như các ràng buộc về hiệu suất.
- Phần 9 - Chất lượng: Một mô tả về cách kiến trúc phần mềm thỏa mãn các yêu cầu khác (ngoài yêu cầu chức năng) của hệ thống: khả năng mở rộng, độ tin cậy, tính di động, v.v.

1.3. Phạm vi dự án

Phần mềm chăm sóc sức khỏe trực tuyến được thiết kế nhằm cung cấp các dịch vụ chăm sóc sức khỏe thông qua nền tảng trực tuyến. Phần mềm sẽ được phát triển dưới dạng ứng dụng web để có thể truy cập từ mọi thiết bị kết nối internet. Người dùng cuối bao gồm các cá nhân quan tâm đến việc duy trì và cải thiện sức khỏe cá nhân, bao gồm người bệnh, người tìm kiếm thông tin sức khỏe, cũng như nhà cung cấp dịch vụ y tế. Phần mềm cho phép người dùng có thể tạo và quản lý hồ sơ sức khỏe cá nhân, bao gồm thông tin về lịch sử bệnh lý, thuốc đã dùng, kết quả xét nghiệm, và các thông tin liên quan khác. Người bệnh cũng có thể đặt lịch khám, tương tác trực tiếp với các bác sĩ, chuyên gia y tế thông qua cuộc gọi video, tin nhắn, hoặc hệ thống thảo luận trực tuyến. Các bác sĩ có thể quản lý hồ sơ bệnh án của bệnh nhân, theo dõi sức khỏe, tư vấn, giải đáp các câu hỏi của bệnh nhân, ...

2. Cơ chế phân tích

2.1. Cơ chế về tính nhất quán

Cần phải xác định được những đặc điểm sau để đảm bảo các thực thể của các lớp trở nên nhất quán:

- Dung lượng: Số lượng tối đa các đối tượng cần lưu trữ bền vững.
- Phạm vi: Xác định tất cả các lớp mà thể hiện của chúng vẫn được lưu trữ cho các lần thực thi sau của hệ thống.
- Quy mô: Xác định miền số lượng các đối tượng cần lưu trữ bền vững.

- Thời lượng: Xác định khoảng thời gian tối đa mà các đối tượng cần lưu trữ bền vững.
- Cơ chế truy cập: Vấn đề đặt ra là làm thế nào để một đối tượng được xác định và truy xuất một cách chính xác (duy nhất).
- Tần suất cập nhật: Các đối tượng có thường xuyên giữ nguyên trạng thái không hay thường xuyên được cập nhật.
- Tính ổn định (Tin cậy): Các đối tượng có cần phải tồn tại được nếu có sự cố xảy ra ở một tiến trình, vi xử lý hay là cả hệ thống?

2.2. Cơ chế về việc giao tiếp

Với tất cả các phần tử mô hình cần giao tiếp với các thành phần hoặc dịch vụ chạy trên một tiến trình hoặc luồng khác, ta cần phải xác định:

- Độ trễ: Các tiến trình phải giao tiếp với nhau trong bao lâu?
- Tính đồng bộ: Các giao tiếp không đồng bộ
- Kích thước của thông điệp: Nên để thành một phổ thay vì một con số đơn lẻ
- Giao thức: Quản lý, bộ đệm luồng, . . .

2.3. Cơ chế về bảo mật

Với mỗi lớp, hệ thống con, gói, ta cần xác định được những tiêu chí về bảo mật sau:

- Độ chi tiết của dữ liệu: Mức độ sâu của dữ liệu được biểu diễn bởi bảng sự thật hoặc chiều trong kho dữ liệu.
- Độ chi tiết của người dùng: Xác định hệ thống có bao nhiêu quyền?
- Các quy định an ninh: Các tiêu chuẩn về bảo mật nhằm bảo vệ dữ liệu cá nhân của người dùng.
- Các loại đặc quyền: Với mỗi một role, cần xác định họ có những quyền gì với hệ thống.

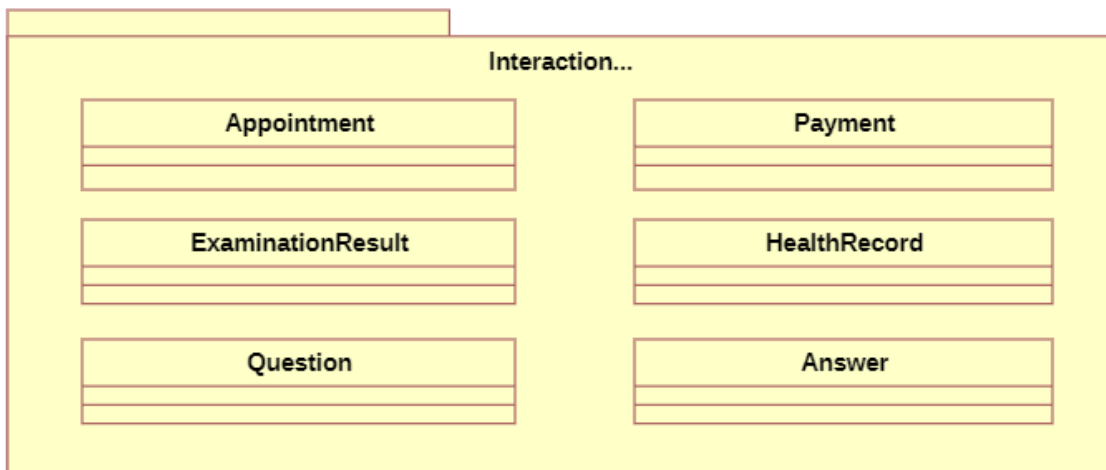
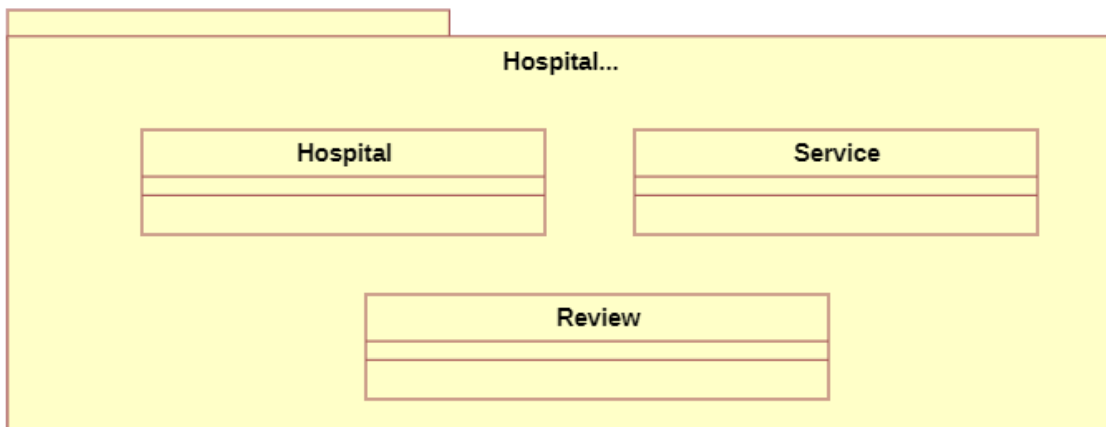
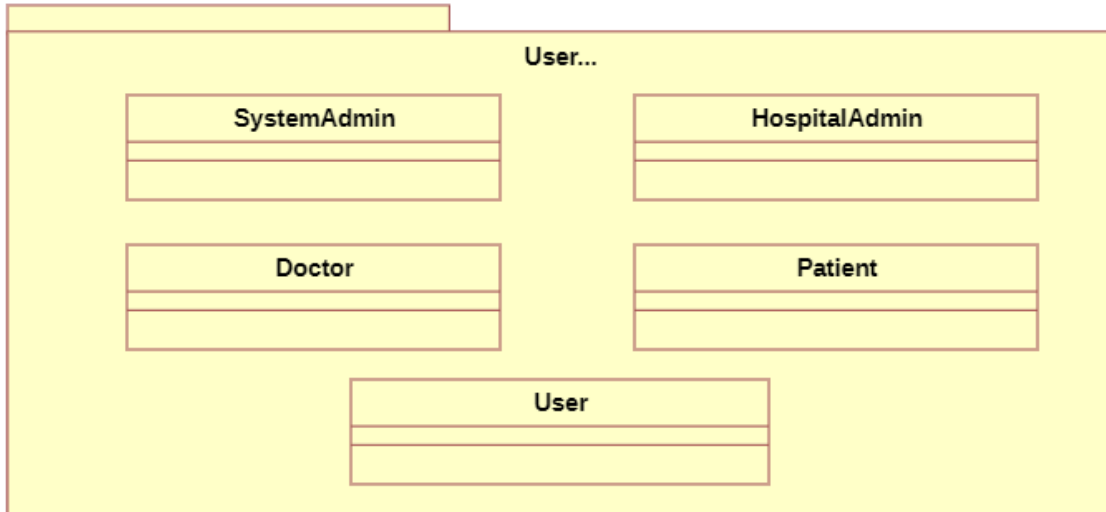
2.4. Các cơ chế khác

Các cơ chế khác mà hệ thống cần quan tâm:

- Cơ chế phân tán: Dữ liệu sẽ được tổ chức lưu trong các máy chủ ra sao để đảm bảo hệ thống luôn hoạt động?
- Cơ chế điều khiển lỗi và thất bại: Các lỗi của hệ thống được báo cáo và xử lý như thế nào?
- Cơ chế quản lý giao dịch: Làm sao các giao dịch trong hệ thống an toàn khỏi các mã độc đồng thời vẫn nhanh, ít bước xác thực nhất.
- Cơ chế về dư thừa thông tin: Tổ chức lưu trữ thông tin hiệu quả, tối ưu cho bộ nhớ. Dữ liệu sẽ được lưu trong hệ thống bao lâu trước khi bị xóa bỏ.

3. Các trừu tượng chính

3.1. Biểu đồ



3.2. Định nghĩa

- **User:** Người dùng của hệ thống, truy cập vào hệ thống qua đăng nhập. User là một lớp trừu tượng.
- **SystemAdmin:** Các quản trị viên hệ thống.
- **HospitalAdmin:** Các quản trị viên tại các bệnh viện.
- **Doctor:** Các bác sĩ tại các bệnh viện có đăng ký với hệ thống.
- **Patient:** Các bệnh nhân/người khám có thể đặt lịch khám với bệnh viện/bác sĩ.
- **Hospital:** Các bệnh viện có hợp tác với hệ thống.
- **Service:** Các dịch vụ khám bệnh tại một bệnh viện cụ thể có hợp tác với hệ thống. Các thông tin cơ bản của một dịch vụ bao gồm id, tên dịch vụ, mô tả dịch vụ và giá dịch vụ.
- **Review:** Đánh giá của người khám về dịch vụ khám hoặc bệnh viện, bao gồm đánh giá bằng số và bằng chữ.
- **Appointment:** Một cuộc hẹn khám giữa bệnh nhân và bác sĩ. Các thông tin bao gồm id, hình thức khám (online hay offline), giờ bắt đầu và kết thúc khám, triệu chứng của bệnh nhân (cung cấp khi đặt lịch khám), ghi chú và tổng hợp các đánh giá (Review) từ người khám.
- **Payment:** Giao dịch thanh toán cho một cuộc hẹn khám, là số tiền bệnh nhân đã trả để đặt lịch khám (online) hoặc khám trực tiếp.
- **HealthRecord:** Hồ sơ sức khỏe của bệnh nhân. Khi khám xong với bác sĩ, bác sĩ sẽ cập nhật hồ sơ sức khỏe lưu trữ thông tin thăm khám của người bệnh.
- **ExaminationResult:** Kết quả một lần khám của bệnh nhân. Một kết quả khám bao gồm các thông tin: triệu chứng, các xét nghiệm kèm theo, chẩn đoán, kết luận, đơn thuốc (nếu có) và ghi chú cho người bệnh. Mỗi lần thăm khám với bác sĩ, kết quả khám của bệnh nhân đều được cập nhật vào hồ sơ sức khỏe. Một HealthRecord có thể bao gồm nhiều ExaminationResult.

- **Question:** Các câu hỏi đã được đặt trên diễn đàn trao đổi của hệ thống.
- **Answer:** Các câu trả lời cho các câu hỏi đã được đặt trên diễn đàn trao đổi của hệ thống.

4. Biểu diễn kiến trúc hệ thống

Tài liệu mô tả kiến trúc của hệ thống theo nhiều khung nhìn: khung nhìn ca sử dụng, khung nhìn tiến trình, khung nhìn triển khai và khung nhìn cài đặt. Qua đó có thể mô tả kiến trúc một cách đầy đủ và tổng quan nhất với từng đối tượng đọc tài liệu. Những khung nhìn trên được biểu diễn bằng Unified Modeling Language (UML).

5. Mục tiêu và ràng buộc

Các yêu cầu chính và các ràng buộc hệ thống có ảnh hưởng lớn đến kiến trúc hệ thống:

- Hệ thống hoạt động được trên tất cả các trình duyệt web phổ biến.
- Các thông tin người dùng và các giao dịch tài chính cần phải được truyền đi một cách bảo mật.
- Tất cả các yêu cầu hiệu suất, khả năng chịu tải như đã được miêu tả trong tài liệu đặc tả bổ sung cần phải được xem xét khi phát triển hệ thống.

6. Khung nhìn ca sử dụng

Phần này mô tả việc phân rã hệ thống thành các quá trình nhẹ (luồng điều khiển đơn) và các quá trình nặng (nhóm các quá trình nhẹ). Phần này được tổ chức theo các nhóm quá trình tương tác hoặc trao đổi thông tin. Mô tả các chế độ chính của việc giao tiếp giữa các quá trình, chẳng hạn như truyền thông tin nhắn, ngắt và hẹn gặp. Biểu đồ quá trình của hệ thống có thể được xem như sau:

Những ca sử dụng (use case) của hệ thống được liệt kê dưới đây. Những Use case được bôi đậm là những Use case quan trọng với kiến trúc hệ thống.

- Đăng nhập
- Quản lý thông tin cá nhân

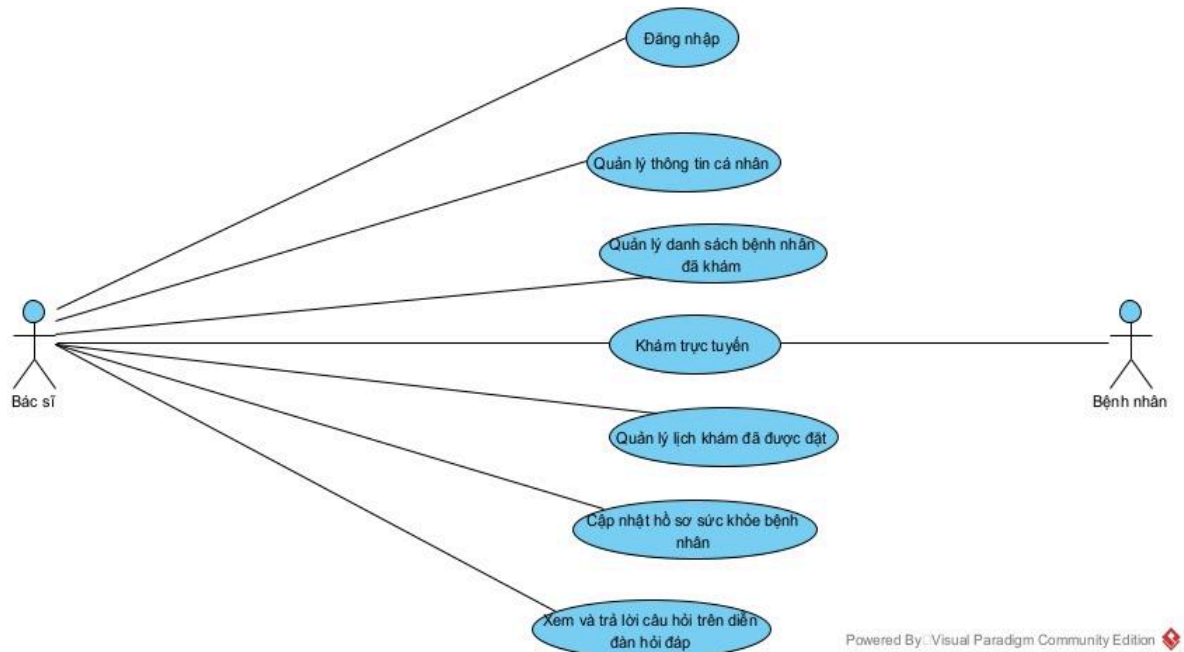
- **Quản lý danh sách tài khoản hệ thống**
- **Quản lý danh sách bệnh viện**
- **Quản lý danh sách tài khoản bác sĩ**
- **Quản lý bệnh viện**
- **Quản lý lịch khám đã được đặt**
- **Cập nhật hồ sơ sức khỏe bệnh nhân**
- **Quản lý danh sách bệnh nhân đã khám**
- **Đặt lịch khám tại bệnh viện**
- **Đặt lịch khám với bác sĩ**
- **Quản lý lịch sử đặt khám**
- **Xem hồ sơ sức khỏe cá nhân**
- **Khám trực tuyến**
- **Xem và trả lời câu hỏi trên diễn đàn hỏi đáp**
- **Đặt câu hỏi trên diễn đàn hỏi đáp**
- **Tìm kiếm bệnh viện/dịch vụ khám/bác sĩ**
- **Đánh giá**
- **Thanh toán**

6.1. Hiện thực hóa Use case

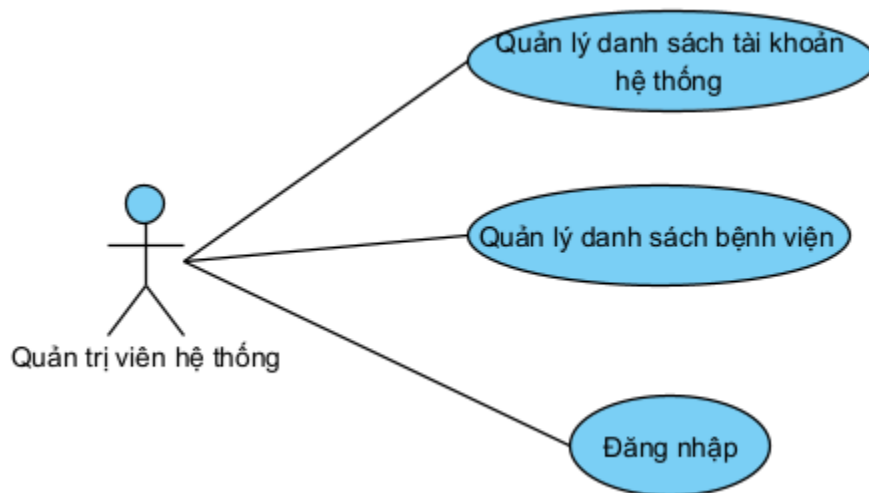
- Use case cho Bệnh nhân (Patient):



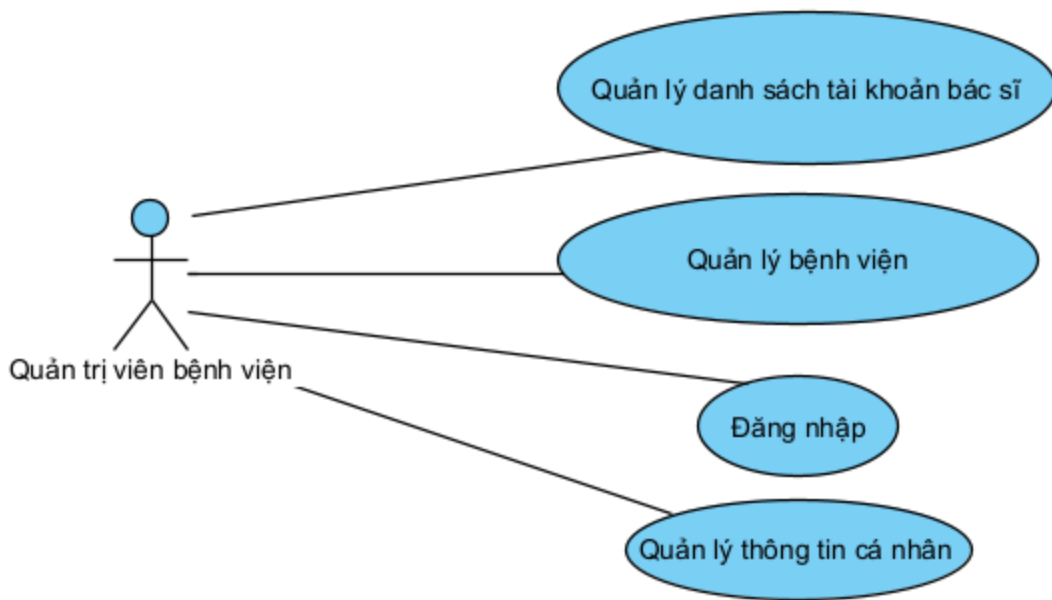
- Use case cho Bác sĩ (Doctor):



- Use case cho Quản trị viên hệ thống (System Admin):



- Use case cho Quản trị viên bệnh viện (Hospital Admin):



6.2. Mô tả Use case quan trọng

- **Quản lý danh sách tài khoản hệ thống:** Ca sử dụng này mô tả cách quản trị viên hệ thống quản lý (thêm, cấm tạm thời, cấm vĩnh viễn, mở khóa, xóa) danh sách tài khoản người dùng trên toàn hệ thống (danh sách tài khoản quản trị viên bệnh viện, danh sách tài khoản bác sĩ và bệnh nhân).
- **Quản lý danh sách bệnh viện:** Ca sử dụng này mô tả cách quản trị viên hệ thống quản lý (thêm, ngừng liên kết, xóa) danh sách bệnh viện trên toàn hệ thống.
- **Quản lý danh sách tài khoản bác sĩ:** Ca sử dụng này mô tả cách quản trị viên bệnh viện quản lý (thêm, cấm, mở cấm, xóa) danh sách tài khoản bác sĩ trong bệnh viện đó.
- **Quản lý bệnh viện:** Ca sử dụng này mô tả cách quản trị viên bệnh viện quản lý (thêm, chỉnh sửa, xóa) dịch vụ bệnh viện và quản lý các thông tin chung về bệnh viện.
- **Quản lý lịch khám đã được đặt:** Ca sử dụng này mô tả cách bác sĩ xem và quản lý trạng thái các lịch khám cá nhân.

- **Cập nhật hồ sơ sức khỏe bệnh nhân:** Ca sử dụng này mô tả cách bác sĩ cập nhật hồ sơ sức khỏe của bệnh nhân mình đã khám.
- **Đặt lịch khám tại bệnh viện:** Ca sử dụng này mô tả cách bệnh nhân đặt lịch khám tại bệnh viện trên hệ thống.
- **Đặt lịch khám với bác sĩ:** Ca sử dụng này mô tả cách bệnh nhân đặt lịch khám trực tiếp hoặc trực tuyến với bác sĩ trên hệ thống.
- **Xem hồ sơ sức khỏe cá nhân:** Ca sử dụng này mô tả cách bệnh nhân xem hồ sơ sức khỏe cá nhân trên hệ thống.
- **Khám trực tuyến:** Ca sử dụng này mô tả cách bệnh nhân và bác sĩ video call để khám trực tuyến theo lịch đã đặt.
- **Tìm kiếm bệnh viện/dịch vụ khám/bác sĩ:** Ca sử dụng này mô tả cách người dùng tìm kiếm bệnh viện, các dịch vụ khám hoặc bác sĩ trên hệ thống.
- **Thanh toán:** Ca sử dụng này mô tả cách mà một thanh toán được tạo giữa bệnh nhân và bác sĩ trên hệ thống. Hệ thống thanh toán sẽ xác nhận và hoàn thành giao dịch.

7. Khung nhìn logic

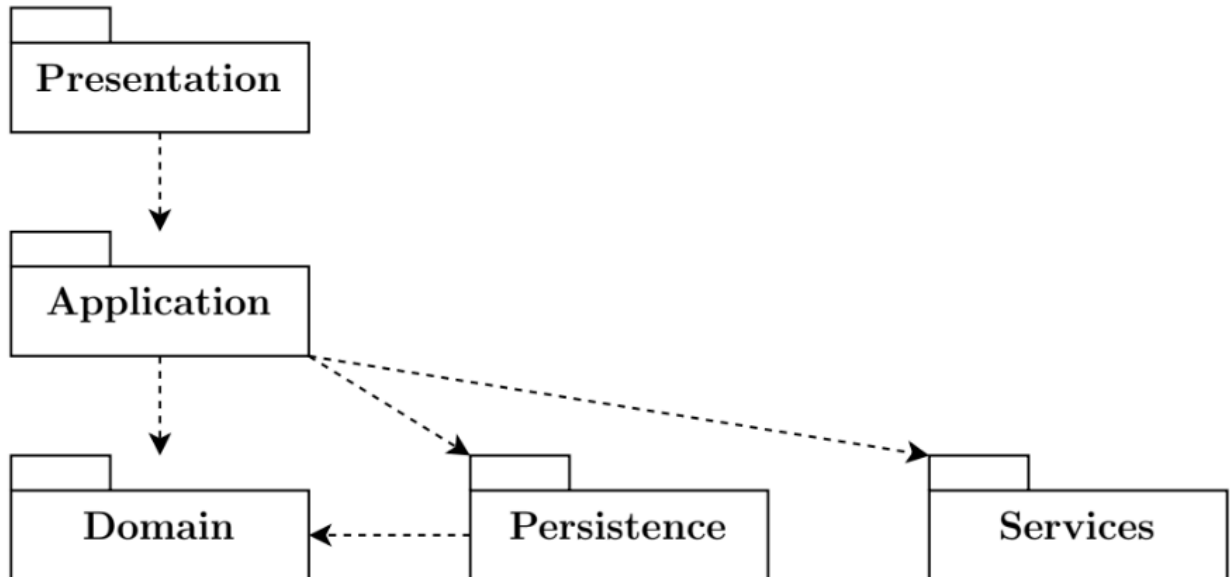
7.1. Tổng quan

Đây là một mô tả logic về kiến trúc của hệ thống. Mô tả những lớp quan trọng và cách tổ chức những gói dịch vụ và hệ thống con, và cách tổ chức những hệ thống con này vào các lớp. Trong phần này cũng mô tả những ca sử dụng quan trọng nhất, ví dụ những khía cạnh linh hoạt của kiến trúc hệ thống. Biểu đồ lớp có thể được đưa vào để minh họa các mối quan hệ giữa kiến trúc quan trọng các lớp, hệ thống con, gói và lớp.

Khung nhìn logic của hệ thống bao gồm 5 gói:

- **Giao diện (Presentation):** chứa các lớp cho mỗi biểu mẫu mà các tác nhân sử dụng để giao tiếp với Hệ thống.
- **Ứng dụng (Application):** chứa các lớp xử lý chính cho hệ thống.

- **Miền** (Domain): chứa các gói hỗ trợ các thực thể chính của hệ thống.
- **Nhất quán** (Persistence): chứa các lớp để đảm bảo tính nhất quán của dữ liệu.
- **Dịch vụ** (Services): chứa các lớp để cung cấp các lớp hệ thống cho mục đích bảo trì.



7.2. Các gói thiết kế kiến trúc

7.2.1. Gói giao diện

Mô tả ngắn gọn: Trong gói này chứa các lớp cho mỗi mẫu mà các tác nhân sử dụng để giao tiếp với hệ thống. Các lớp ranh giới tồn tại để hỗ trợ duy trì xác thực, quản lý các appointment, forum hỏi đáp, tìm kiếm bác sĩ, dịch vụ, triển khai việc khám trực tuyến.

Biểu đồ:

<<boundary>> DoctorAccountManagementView +viewDoctorAccountList() +displayDoctorAccountList() +viewDoctorAccountInfo() +displayDoctorAccountInfo() +inputSearchKeyword() +banDoctorAccount() +displayBanDoctorAccountForm() +unbanDoctorAccount() +displaySearchResult() +displayBanSuccessfullyNotify() +displayConfirm() +confirmUnbanDoctorAccount() +displayUnbanSuccessfullyNotify() +confirmDeleteDoctorAccount() +deleteDoctorAccount() +displayDeleteReasonForm() +displayDeleteSuccessfullyNotify() +addAnDoctorAccount() +displayDoctorAccountInfoForm() +submitAccountInfoForm() +displayAddSuccessfullyNotify() +displayAddUnsuccessfullyNotify()
--

<<boundary>> HospitalServiceView +viewHospitalServiceList() +viewHospitalService() +scheduleAppointment() +displayHospitalServiceList() +displayHospitalService() +displayAppointmentInformation()

<<boundary>> PaymentView +payForService() +displayPaymentOption() +fillPaymentInfo() +submitPayment() +displayPaymentInfo() +displayInvalidPaymentNotify()

<<boundary>> ViewForumQuestionForm +viewQuestionList() +setFilter(filter) +retrieveData() +selectQuestionFormList() +requestToAnswerTheQuestion() +requestToPostQuestion()

<<boundary>> DoctorInfoView +viewDoctor() +displayAppointmentInformation() +scheduleAppointment() +displayDoctorAccountInfo()
--

<<boundary>> OnlineMeeting +openAnOnlineMeetingRoom() +joinMeeting() +endMeeting()
--

<<boundary>> ReviewView +makeReview() +displayReview()

<<boundary>> SearchFormView +type() +seeDetail()

<<boundary>> HospitalManagementView +displayHospitalList() +displayHospitalInfo() +displayAddHospitalForm() +displayAddHospitalSuccessfullyNotify() +displayAddHospitalAdminForm() +displayAddHospitalAdminSuccessfullyNotify() +displayUnconnectSuccessfullyNotify() +displayDeleteSuccessfullyNotify() +viewHospitalList() +viewHospitalInfo() +searchOrFilterHospital() +addHospital() +fillAndSubmitAddHospitalForm() +fillAndSubmitAddHospitalAdminForm() +unconnectHospital() +deleteHospital()
--

<<boundary>> DoctorAppointmentManagementView +viewAppointmentList() +displayAppointmentList() +viewAnAppointment() +displayAppointmentDetail() +inputSearchKeyword() +displaySearchResult() +chooseAppointmentFilter() +displayFilterResult() +changeAppointmentStatus() +displayChangeSuccessfullyNotify()
--

<<boundary>> PatientAppointmentManagementView +viewPatientAppointmentList() +displayPersonalAppointmentList() +viewPersonalAppointment() +cancelAppointment()
--

<<boundary>> HospitalAdminHospitalManagementView +viewManageOptions() +displayManageOptions() +chooseHospitalServiceManage() +chooseHospitalInformationManage()
--

<<boundary>> ViewHealthRecordForm +requestToViewHealthRecord()
--

<<boundary>> PatientListManagementView +viewPatientList() +displayPatientList() +viewPatientInformation() +displayPatientInformation() +updatePatientHealthRecord() +displayUpdateHealthRecordForm() +submitUpdateHealthRecordForm() +displayUpdateSuccessfullyNotify()
--

<<boundary>> ProfileManagementView +viewProfile() +updateUserProfile()

<<boundary>> Notification +notifyActionStatus()

<<boundary>> HospitalServiceManagementView +transferServiceView() +displayServiceList() +viewServiceInfo() +displayServiceInformation() +AddAService() +displayAddServiceForm() +submitAddServiceForm() +displayAddSuccessfullyNotify() +displayAddUnsuccessfullyNotify() +editAService() +displayEditServiceForm() +submitEditAServiceForm() +displayEditSuccessfullyNotify() +displayEditUnsuccessfullyNotify() +deleteAService() +displayDeleteConfirm() +confirmDelete() +displayDeleteSuccessfullyNotify()
--

<<boundary>> AccountManagementView +viewAccountType() +chooseAccountTypeList() +viewAccountInfo() +searchOrFilterAccount() +banAccount() +unbanAccount() +addAccount() +fillAndSubmitAddAccountForm() +deleteAccount() +displayAccountList() +displayAccountType() +displayAccountInfo() +displayBanSuccessfullyNotify() +displayUnbanSuccessfullyNotify() +displayAddAccountForm() +displayAddSuccessfullyNotify() +displayInvalidInfoNotify() +displayDeleteSuccessfullyNotify()

<<boundary>> HospitalInformationView +transferInformationView() +displayHospitalInformation() +editHospitalInformation() +displayRequireInformationForm() +submitInformationForm() +displayEditSuccessfullyNotify() +displayEditUnsuccessfullyNotify()
--

<<boundary>> ViewAppointmentForm +viewAppointmentList() +selectAppointmentFromList() +requestVideoCall() +requestToEndVideoCall()
--

<<boundary>> LoginView +loginWithGoogle() +displayLoginSuccessfullyNotify() +redirectToPageAfterLogin() +displayLoginFailNotify()
--

<<boundary>> CreateAnswerForm +createForm() +createAnswer()
--

<<boundary>> GoogleSystemInterface +loginWithGoogle()

7.2.2. Gói ứng dụng

Mô tả ngắn gọn: Gói này chứa các lớp cho các logic nghiệp vụ trong hệ thống. Các lớp điều khiển tồn tại để hỗ trợ duy trì xác thực, quản lý quản lý các appointment, forum hỏi đáp, tìm kiếm bác sĩ, dịch vụ, triển khai việc khám trực tuyến.

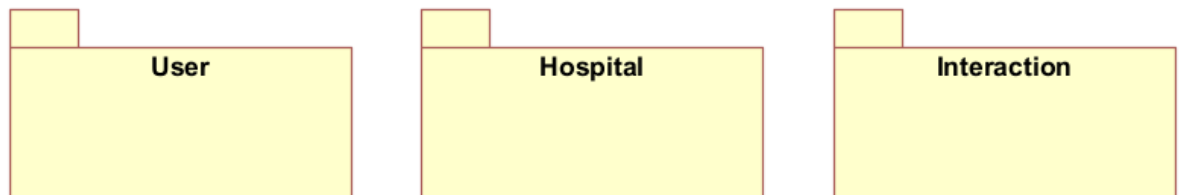
Biểu đồ:



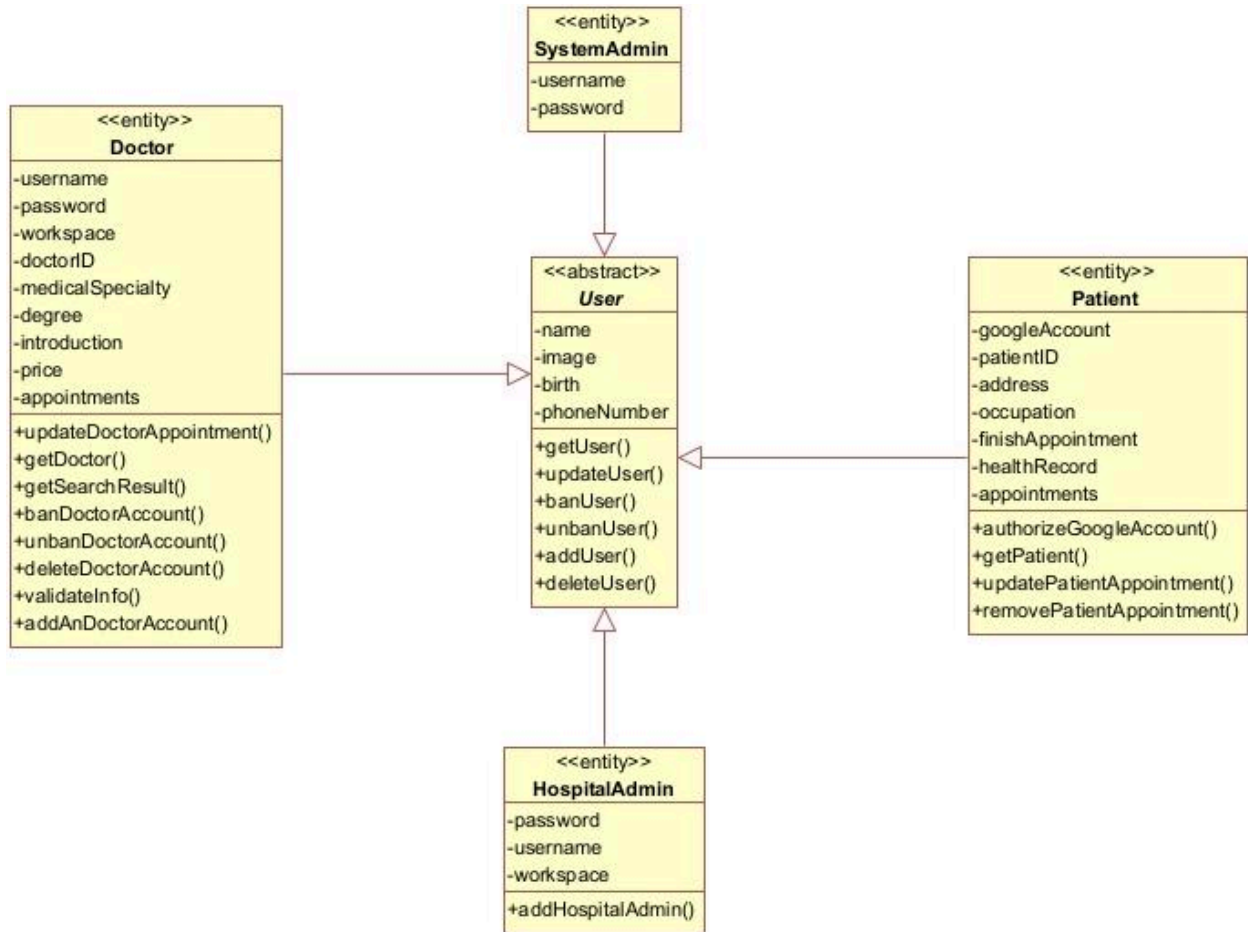
7.2.3. Gói miền

Mô tả ngắn gọn: Gói này chứa các gói chứa các lớp để hỗ trợ các thực thể người dùng, appointment, forum hỏi đáp. Gói miền bao gồm 3 gói con sau đây:

- **Gói người dùng:** chứa tất cả các lớp phục vụ quản lý các người dùng.
- **Gói hospital:** chứa các yêu cầu, dịch vụ lớp để hỗ trợ những nhiệm vụ liên quan đến bác sĩ, bệnh viện.
- **Gói interaction:** chứa tất cả lớp phục vụ việc tương tác giữa các người dùng.



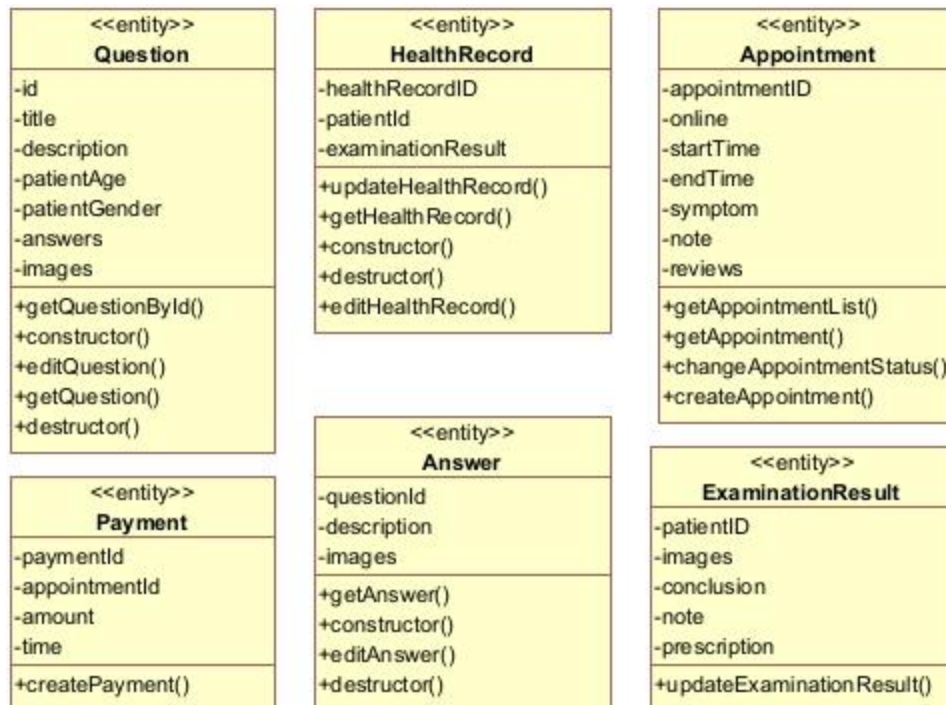
Gói người dùng



Gói Hospital



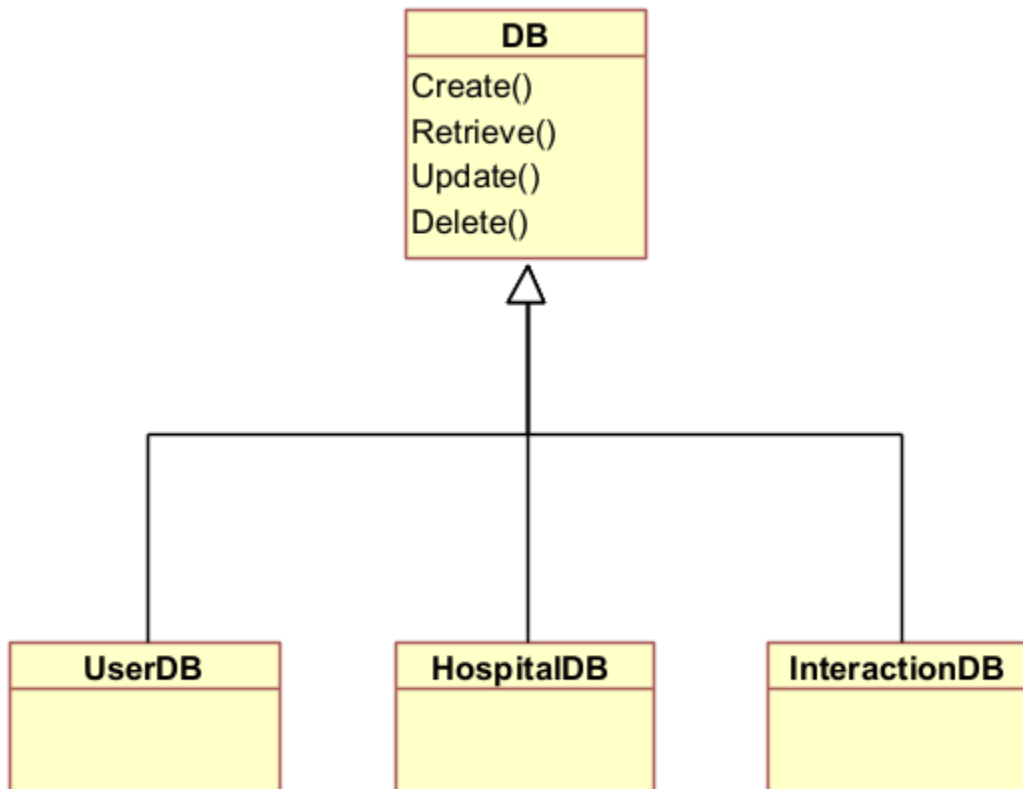
Gói Interaction



7.2.4. Gói nhất quán

Mô tả ngắn gọn: Gói này chứa các gói dữ liệu để đảm bảo tính nhất quán của dữ liệu. Bốn toán tử: thêm, sửa, xóa, cập nhật là bốn chức năng chính được thực hiện trong các ứng dụng cơ sở dữ liệu.

Biểu đồ:



8. Khung nhìn tiến trình

8.1. Mô hình tiến trình

Phần này sẽ mô tả sự phân hóa của hệ thống thành các quá trình nhẹ (các luồng điều khiển đơn) và các quy trình nặng (nhóm của các quy trình nhẹ), tổ chức các phần này theo nhóm các tiến trình giao tiếp hoặc tương tác. Mô tả các chế độ giao tiếp chính giữa các quá trình, chẳng hạn như chuyển tiếp tin nhắn, gián đoạn.

8.2. Mô tả các phần tử tiến trình

Tiến trình Quản lý người dùng: Đây là tiến trình quản lý thông tin người dùng và các hoạt động của họ trên ứng dụng.

Tiến trình này bao gồm các bước sau:

- Nhận yêu cầu đăng ký, đăng nhập hoặc cập nhật thông tin người dùng từ giao diện người dùng.
- Xử lý yêu cầu và truy vấn cơ sở dữ liệu để lưu trữ hoặc cập nhật thông tin người dùng.
- Kiểm tra tính hợp lệ của thông tin người dùng và xử lý các lỗi khi cần thiết.

Tiến trình thanh toán

Tiến trình này bao gồm các bước sau:

- Bệnh nhân chọn bác sĩ mà mình muốn được khám, sau đó điền/chọn các thông tin cần thiết và bấm “Đặt khám”.
- Bệnh nhân thanh toán thông qua các phương thức thanh toán qua ví điện tử.
- Hệ thống xử lý thanh toán, kiểm tra tính hợp lệ của thông tin thanh toán và cập nhật trạng thái thanh toán.
- Hệ thống gửi thông tin thanh toán thành công hoặc không thành công cho người dùng.

Tiến trình đánh giá bệnh viện/dịch vụ khám/bác sĩ cho phép bệnh nhân đánh giá các bệnh viện/dịch vụ khám/bác sĩ mà họ đã trải nghiệm, đồng thời cung cấp thông tin đánh giá cho người dùng khác tham khảo.

Tiến trình này bao gồm các bước sau:

- Khởi động tiến trình: Bệnh nhân truy cập vào ứng dụng Chăm sóc sức khỏe trực tuyến, sau đó truy cập trang đánh giá sau khám. Bệnh nhân

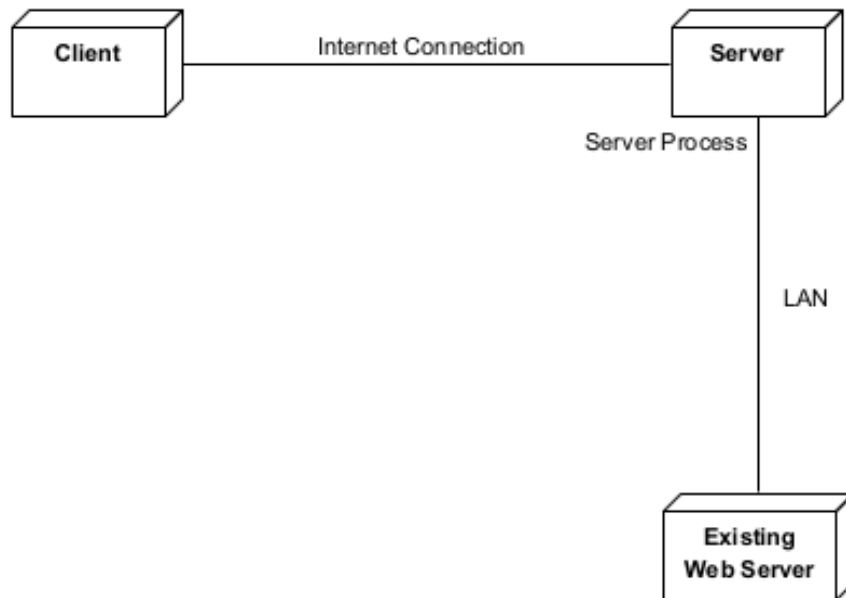
chọn một lịch khám chưa được đánh giá và truy cập chức năng đánh giá.

- **Hiển thị giao diện đánh giá:** Hệ thống hiển thị giao diện đánh giá cho bệnh nhân nhập thông tin về đánh giá, bao gồm điểm đánh giá (thang 5), và phản hồi chi tiết.
- **Nhập thông tin đánh giá:** Bệnh nhân nhập thông tin đánh giá vào các trường tương ứng trên giao diện đánh giá.
- **Lưu thông tin đánh giá:** Hệ thống lưu thông tin đánh giá vào cơ sở dữ liệu của ứng dụng.
- **Xử lý thông tin đánh giá:** Hệ thống sử dụng các thuật toán đánh giá để tính toán điểm đánh giá trung bình cho bệnh viện/dịch vụ khám/bác sĩ dựa trên đánh giá của bệnh nhân.
- **Cập nhật thông tin đánh giá:** Hệ thống cập nhật điểm đánh giá trung bình và số lượng đánh giá cho bệnh viện/dịch vụ khám/bác sĩ vào cơ sở dữ liệu của ứng dụng.
- **Hiển thị thông tin đánh giá:** Hệ thống hiển thị thông tin đánh giá của bệnh viện/dịch vụ khám/bác sĩ cho người dùng khác tham khảo trên trang chi tiết bệnh viện/dịch vụ khám/bác sĩ.

Kết thúc quá trình: Tiến trình đánh giá bệnh viện/dịch vụ khám/bác sĩ kết thúc và quay trở lại giao diện hồ sơ của bệnh viện/dịch vụ khám/bác sĩ vừa đánh giá.

9. Khung nhìn triển khai

Phần này mô tả các cấu hình mạng vật lý (phần cứng) mà phần mềm được triển khai và hoạt động trên đó. Các nút vật lý (máy tính, CPU) chạy phần mềm và có các kết nối giữa chúng (bus, LAN, point-to-point). Có một ánh xạ của các tiến trình trong khung nhìn tiến trình lên các nút vật lý. Server là một máy chủ UNIX, Client là thiết bị có thể chạy trình duyệt web như PC, điện thoại và kết nối với Server thông qua Internet.



Cấu trúc phần cứng của hệ thống bao gồm một máy chủ (Server) để lưu trữ và xử lý dữ liệu, và các thiết bị người dùng như điện thoại thông minh, máy tính bảng hoặc máy tính để bàn để cung cấp giao diện người dùng. Máy chủ sẽ được đặt tại một trung tâm dữ liệu (data center) để đảm bảo tính an toàn, tin cậy và mở rộng của hệ thống.

Phần mềm của hệ thống được phát triển, khai thác trên nền tảng điện toán đám mây để đảm bảo tính linh hoạt và khả năng mở rộng.

Phần mềm của hệ thống được viết bằng ngôn ngữ lập trình TypeScript và các Framework phổ biến như Next.js, ReactJs.

Để đảm bảo tính toàn vẹn và bảo mật của dữ liệu, hệ thống sử dụng các phương thức bảo mật như mã hóa SSL (Secure Sockets Layer) để mã hóa dữ liệu giao tiếp và JWT (JSON Web Token) để xác thực người dùng.

10. Quy mô và hiệu năng

Phần mềm này được thiết kế để có thể chịu tới 200,000 người dùng cùng lúc. Nếu muốn nâng cao khả năng chịu tải, có thể dễ dàng cài đặt bằng cách tăng quy mô dịch vụ và cơ sở dữ liệu.

11. Chất lượng

Kiến trúc phần mềm liên quan đến tất cả các khả năng của hệ thống: khả năng mở rộng, độ tin cậy, tính di động,... Nếu tất cả các yêu cầu về chất lượng được đáp ứng thì thời gian trung bình giữa các lần thất bại (Mean Time Between Failures - MTBF) sẽ lớn hơn 3 tháng.