

Trường Đại học Công nghệ - ĐHQGHN

Khoa công nghệ thông tin

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

Giảng viên: PGS.TS. Đặng Đức Hạnh

ThS. Trần Mạnh Cường



DESCRIBE RUN-TIME ARCHITECTURE
ỨNG DỤNG CHĂM SÓC SỨC KHỎE
TRỰC TUYẾN

Ngày: 05/05/2024

Chuẩn bị bởi: Nhóm 5

Mục lục

Lịch sử sửa đổi	5
1. Tổng quan	6
1.1. Giới thiệu	6
1.2. Đối tượng dự kiến và đề xuất cách đọc	6
1.3. Phạm vi dự án	6
1.4. Tài liệu tham khảo	7
2. Mô tả từng phần	8
2.1. Đăng nhập	8
2.1.1. Mô hình tiến trình	8
2.1.2. Mô tả thành phần tiến trình	8
2.2. Quản lý thông tin cá nhân	9
2.2.1. Mô hình tiến trình	9
2.2.2. Mô tả thành phần tiến trình	9
2.3. Quản lý danh sách tài khoản hệ thống	9
2.3.1. Mô hình tiến trình	9
2.3.2. Mô tả thành phần tiến trình	10
2.4. Quản lý danh sách bệnh viện	10
2.4.1. Mô hình tiến trình	10
2.4.2. Mô tả thành phần tiến trình	10
2.5. Quản lý danh sách tài khoản bác sĩ	11
2.5.1. Mô hình tiến trình	11
2.5.2. Mô tả thành phần tiến trình	11
2.6. Quản lý bệnh viện	12
2.6.1. Mô hình tiến trình	12
2.6.2. Mô tả thành phần tiến trình	12
2.7. Quản lý lịch khám đã được đặt	13
2.7.1. Mô hình tiến trình	13
2.7.2. Mô tả thành phần tiến trình	13
2.8. Cập nhật hồ sơ sức khỏe bệnh nhân	13
2.8.1. Mô hình tiến trình	13
2.8.2. Mô tả thành phần tiến trình	13
2.9. Xem hồ sơ sức khỏe cá nhân	14

2.9.1. Mô hình tiến trình	14
2.9.2. Mô tả thành phần tiến trình	14
2.10. Khám trực tuyến	15
2.10.1. Mô hình tiến trình	15
2.10.2. Mô tả thành phần tiến trình	15
2.11. Xem và trả lời câu hỏi trên diễn đàn hỏi đáp	16
2.11.1. Mô hình tiến trình	16
2.11.2. Mô tả thành phần tiến trình	16
2.12. Đặt câu hỏi trên diễn đàn hỏi đáp	17
2.12.1. Mô hình tiến trình	17
2.12.2. Mô tả thành phần tiến trình	17
2.13. Tìm kiếm dịch vụ/bệnh viện/bác sĩ	18
2.13.1. Mô hình tiến trình	18
2.13.2. Mô tả thành phần tiến trình	18
2.14. Đánh giá	19
2.14.1. Mô hình tiến trình	19
2.14.2. Mô tả thành phần tiến trình	19
2.15. Thanh toán	19
2.15.1. Mô hình tiến trình	19
2.15.2. Mô tả thành phần tiến trình	20
2.16. Quản lý danh sách bệnh nhân đã khám	21
2.16.1. Mô hình tiến trình	21
2.16.2. Mô tả thành phần tiến trình	21
2.17. Đặt lịch khám tại bệnh viện	22
2.17.1. Mô hình tiến trình	22
2.17.2. Mô tả thành phần tiến trình	22
2.18. Đặt lịch khám với bác sĩ	23
2.18.1. Mô hình tiến trình	23
2.18.2. Mô tả thành phần tiến trình	23
2.19. Quản lý lịch sử đặt khám	24
2.19.1. Mô hình tiến trình	24
2.19.2. Mô tả thành phần tiến trình	24
3. Mô tả tương tranh (Concurrency)	25

3.1. Mô hình tiến trình	25
3.2. Mô tả thành phần tiến trình	26

Lịch sử sửa đổi

Họ tên	Thời gian	Lý do sửa đổi	Phiên bản
Dương Nguyễn Việt Anh	05/05/2024	Khởi tạo tài liệu	1.0
Lê Trọng Minh	08/05/2024	Vẽ, mô tả các phần từ 2.1 đến 2.4	1.1
Dương Nguyễn Việt Anh	08/05/2024	Vẽ, mô tả các phần từ 2.5 đến 2.8	1.2
Phạm Hoàng Hải	08/05/2024	Vẽ, mô tả các phần từ 2.16 đến 2.19	1.3
Hoàng Văn Nguyên	08/05/2024	Vẽ, mô tả các phần từ 2.9 đến 2.12	1.4
Nguyễn Thị Ngọc Ánh	08/05/2024	Vẽ, mô tả các phần từ 2.13 đến 2.15	1.5
Dương Nguyễn Việt Anh	12/05/2024	Vẽ, mô tả tương tranh và chỉnh sửa biểu đồ các tiến trình	1.6

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 lựa chọn chủ đề giải quyết vấn đề.

Tài liệu này được sử dụng để xác định các tiến trình, luồng 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:

- Quản trị dự án: Người phụ trách quản lý và chịu trách nhiệm về chất lượng hệ thống. Quản trị dự án nên đọc toàn bộ tài liệu để phục vụ việc lên kế hoạch và phân công công việc.
- 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.
- Người viết tài liệu: Người sẽ viết tài liệu trong tương lai (các báo cáo, biên bản).

Phần này mô tả sự phân rã của hệ thống thành các quy trình nhẹ (các luồng điều khiển đơn lẻ) và các quy trình nặng (nhóm các quy trình nhẹ). Tổ chức phân theo nhóm các quy trình giao tiếp hoặc tương tác. Mô tả các phương thức giao tiếp chính giữa các tiến trình, chẳng hạn như truyền tin nhắn, ngắt và điểm hẹn.

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, ...

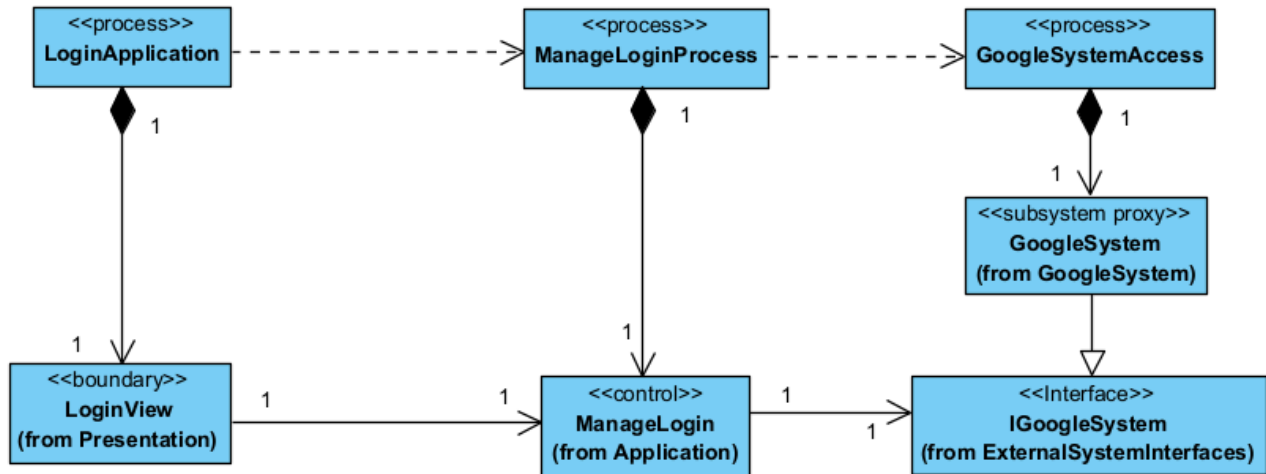
1.4. Tài liệu tham khảo

- [1] IEEE Software Engineering Standards Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”, October 20, 1998.
- [2] Slide môn học Phân tích và thiết kế hướng đối tượng do giảng viên cung cấp.
- [3] Từ điển thuật ngữ của *Phần mềm chăm sóc sức khỏe trực tuyến*.

2. Mô tả từng phần

2.1. Đăng nhập

2.1.1. Mô hình tiến trình

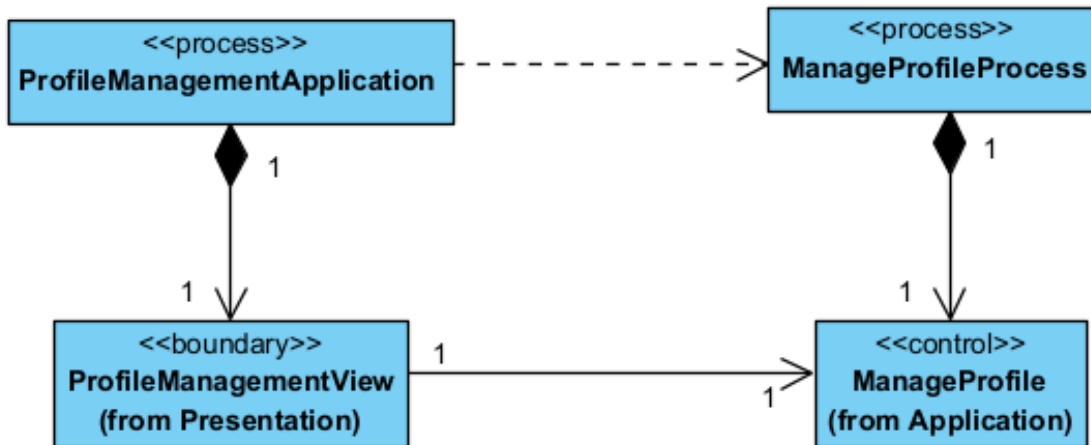


2.1.2. Mô tả thành phần tiến trình

- **LoginApplication:** Điều khiển giao diện của ứng dụng Đăng nhập. Điều khiển loạt các biểu mẫu mà người dùng sử dụng để đăng nhập. Chỉ có một thực thể của tiến trình này.
- **ManageLoginProcess:** Quản lý thực thi của tiến trình đăng nhập sau khi người dùng gửi email và mật khẩu để đăng nhập vào hệ thống. Chỉ có một thực thể và một luồng của quy trình này mỗi khi email và mật khẩu của người dùng được xác thực.
- **GoogleSystem:** Một hệ thống hỗ trợ xác minh người dùng
- **IGoogleSystem:** Điều khiển giao diện giữa hệ thống google và ứng dụng Chăm sóc sức khỏe trực tuyến.
- **GoogleSystemAccess:** Quản lý tất cả truy cập đến hệ thống Google

2.2. Quản lý thông tin cá nhân

2.2.1. Mô hình tiến trình

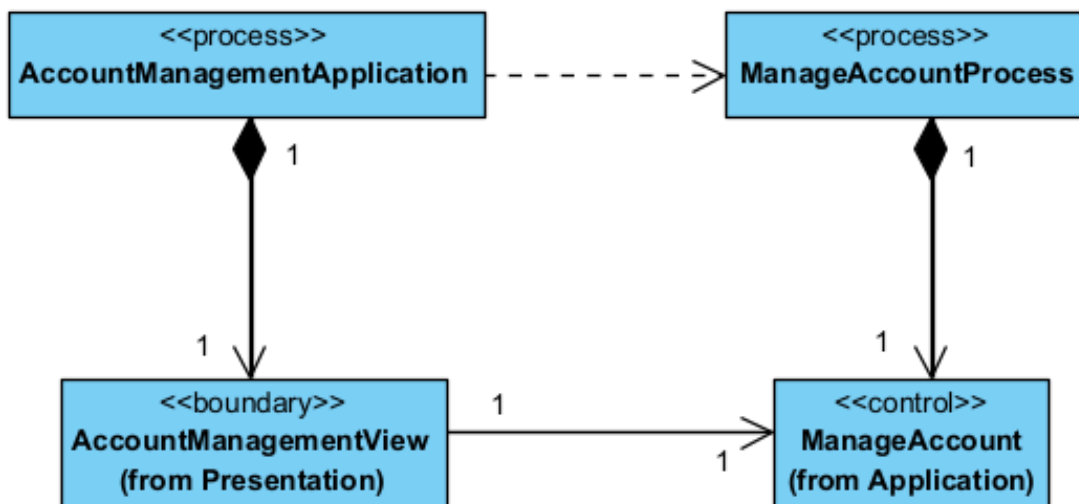


2.2.2. Mô tả thành phần tiến trình

- **ProfileManagementApplication:** quản lý giao diện thông tin cá nhân của người dùng. Đây là một instance của **ProfileManagementView** (lớp Boundary).
- **ManageProfileProcess:** điều khiển việc thực thi quản lý thông tin cá nhân. Một instance của **ManageProfile** (lớp Control) được sử dụng.

2.3. Quản lý danh sách tài khoản hệ thống

2.3.1. Mô hình tiến trình

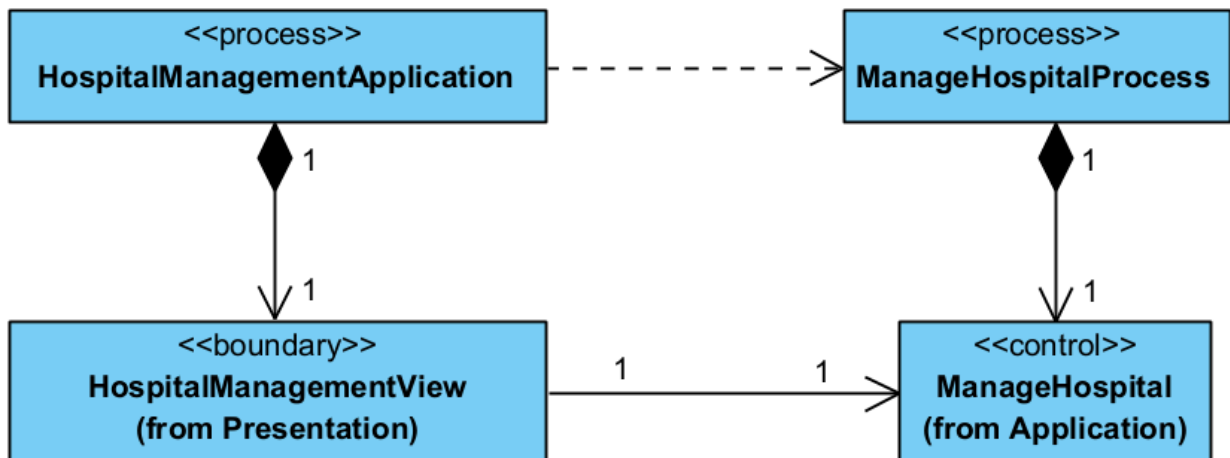


2.3.2. Mô tả thành phần tiến trình

- **AccountManagementApplication:** Quản lý giao diện danh sách tài khoản hệ thống. Đây là một instance của boundary class AccountManagementView.
- **ManageAccountProcess:** điều khiển quá trình thực thi quản lý danh sách tài khoản hệ thống. Một instance của control class ManageAccount được sử dụng.

2.4. Quản lý danh sách bệnh viện

2.4.1. Mô hình tiến trình

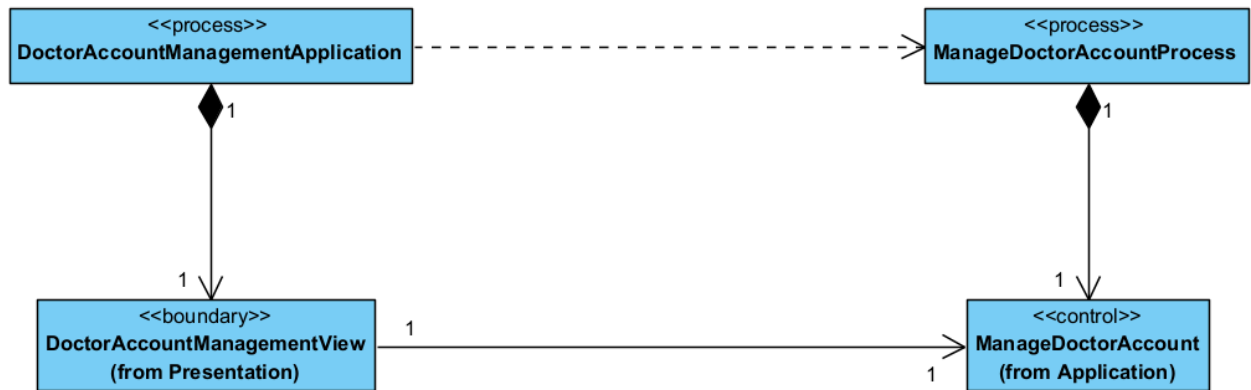


2.4.2. Mô tả thành phần tiến trình

- **HospitalManagementApplication:** quản lý giao diện danh sách bệnh viện. Một instance của boundary class HospitalManagementView được sử dụng.
- **ManageHospitalProcess:** điều khiển quá trình thực thi quản lý danh sách bệnh viện. Một instance của control class ManageHospital được sử dụng.

2.5. Quản lý danh sách tài khoản bác sĩ

2.5.1. Mô hình tiến trình

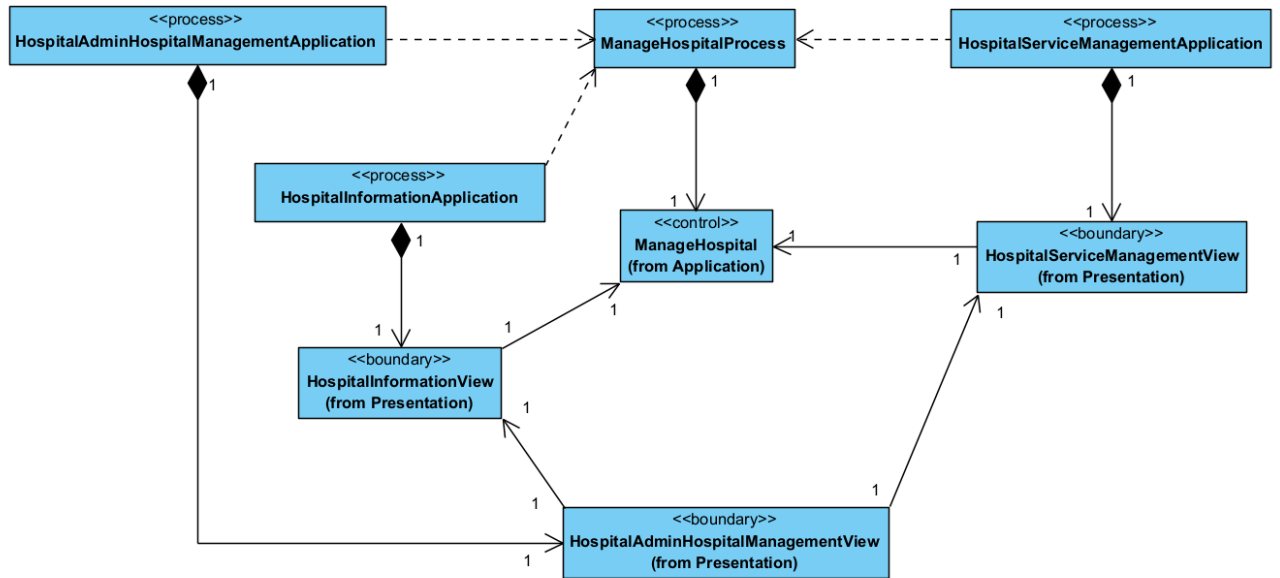


2.5.2. Mô tả thành phần tiến trình

- **DoctorAccountManagementApplication:** Quản lý giao diện danh sách tài khoản bác sĩ. Đây là một instance của boundary class `DoctorAccountManagementView`.
- **ManageDoctorAccountProcess:** điều khiển quá trình thực thi quản lý danh sách tài khoản bác sĩ. Một instance của control class `ManageDoctorAccount` được sử dụng.

2.6. Quản lý bệnh viện

2.6.1. Mô hình tiến trình

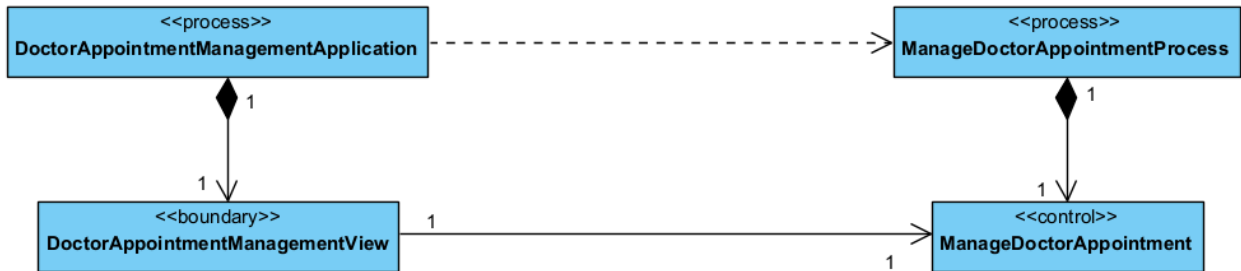


2.6.2. Mô tả thành phần tiến trình

- **HospitalAdminHospitalManagementApplication:** Quản lý giao diện quản lý bệnh viện. Đây là một instance của boundary class HospitalAdminHospitalManagementView.
- **HospitalInformationApplication:** Quản lý giao diện quản lý thông tin bệnh viện. Đây là một instance của boundary class HospitalInformationView.
- **HospitalServiceManagementApplication:** Quản lý giao diện quản lý dịch vụ bệnh viện. Đây là một instance của boundary class HospitalServiceManagementView.
- **ManageHospitalProcess:** điều khiển quá trình thực thi quản lý bệnh viện. Một instance của control class ManageHospital được sử dụng.

2.7. Quản lý lịch khám đã được đặt

2.7.1. Mô hình tiến trình

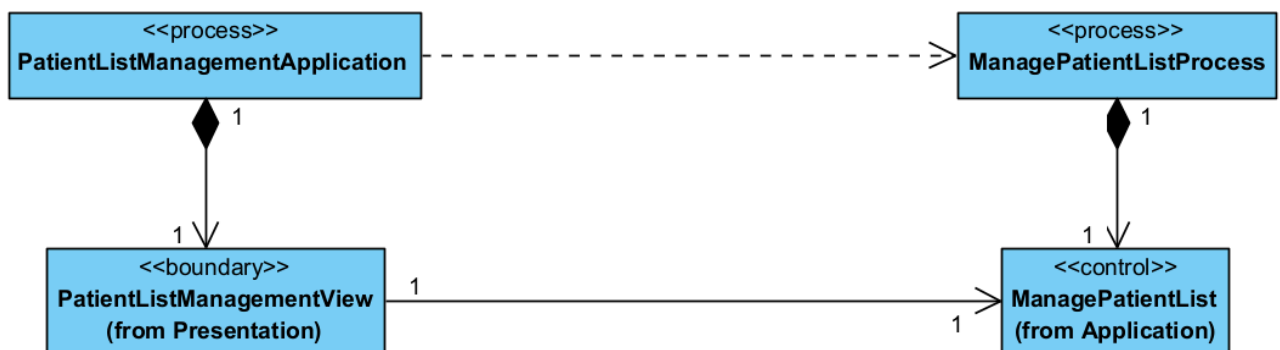


2.7.2. Mô tả thành phần tiến trình

- **DoctorAppointmentManagementApplication:** Quản lý giao diện quản lý lịch khám đã được đặt của bác sĩ. Đây là một instance của boundary class **DoctorAppointmentManagementView**.
- **ManageDoctorAppointmentProcess:** điều khiển quá trình thực thi quản lý lịch khám đã được đặt của bác sĩ. Một instance của control class **ManageDoctorAppointment** được sử dụng.

2.8. Cập nhật hồ sơ sức khỏe bệnh nhân

2.8.1. Mô hình tiến trình



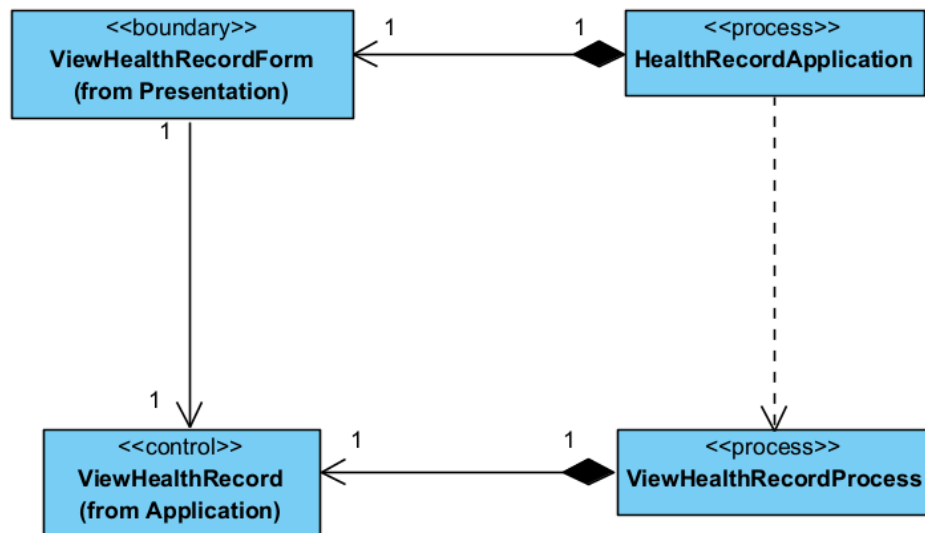
2.8.2. Mô tả thành phần tiến trình

- **PatientListManagementApplication:** Quản lý giao diện quản lý danh sách bệnh nhân đã khám của bác sĩ, đồng thời quản lý hồ sơ sức khỏe bệnh nhân. Đây là một instance của boundary class **PatientListManagementView**.

- **ManagePatientListProcess:** điều khiển quá trình thực thi quản lý danh sách bệnh nhân đã khám của bác sĩ. Một instance của control class ManagePatientList được sử dụng.

2.9. Xem hồ sơ sức khỏe cá nhân

2.9.1. Mô hình tiến trình

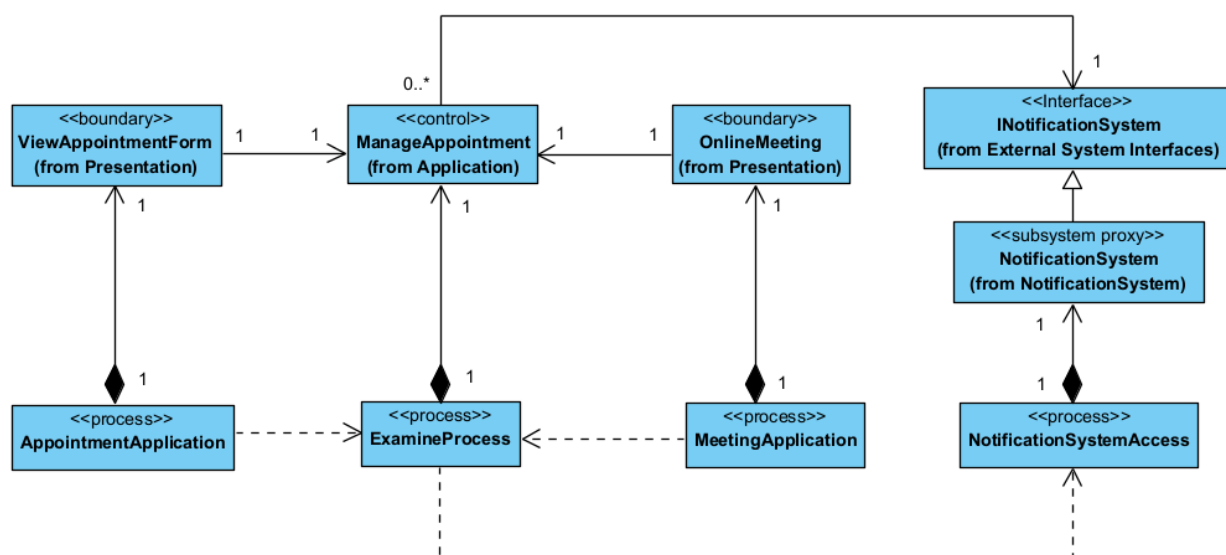


2.9.2. Mô tả thành phần tiến trình

- **HealthRecordApplication:** Điều khiển các giao diện, biểu mẫu cho người dùng thao tác với các chức năng liên quan tới hồ sơ sức khỏe. Tiến trình này có thể hiện của lớp *ViewHealthRecordForm*, có nhiệm vụ giúp bệnh nhân xem hồ sơ sức khỏe cá nhân. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **ViewHealthRecordProcess:** Quản lý quá trình, thực hiện xử lý quá trình xem hồ sơ sức khỏe của bệnh nhân. Có một thể hiện của tiến trình này tương ứng với mỗi bệnh nhân tự xem hồ sơ sức khỏe.

2.10. Khám trực tuyến

2.10.1. Mô hình tiến trình



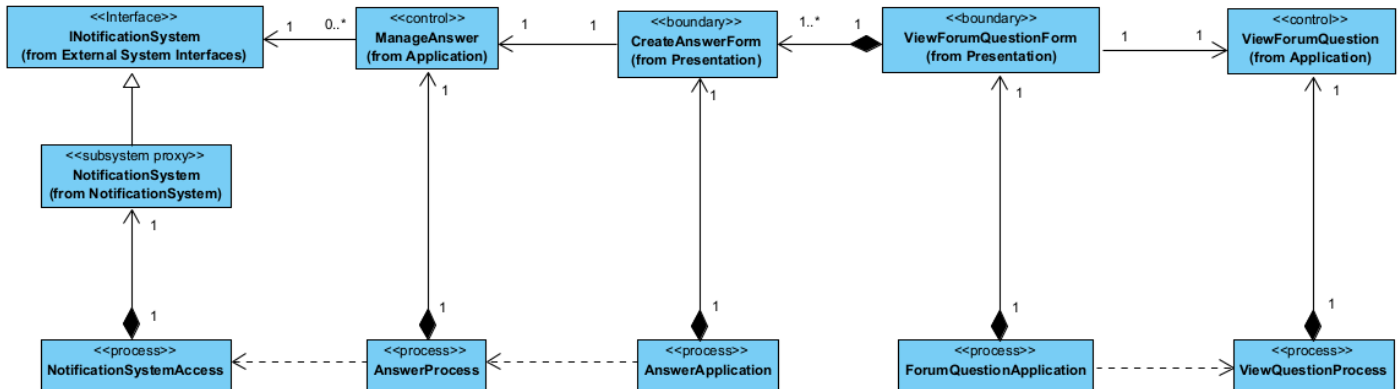
2.10.2. Mô tả thành phần tiến trình

- **AppointmentApplication:** Điều khiển các giao diện, biểu mẫu cho người dùng thao tác với các chức năng liên quan tới lịch khám giữa bác sĩ và bệnh nhân. Tiến trình này có thể hiện của lớp *ViewAppointmentForm*, có nhiệm vụ giúp bác sĩ thực hiện việc xem lịch khám cá nhân. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **ExamineProcess:** Quản lý quá trình, thực hiện xử lý quá trình khám trực tuyến. Có một thể hiện của tiến trình này tương ứng với mỗi lần người dùng thực hiện việc khám trực tuyến.
- **MeetingApplication:** Điều khiển các giao diện cho người dùng thao tác với các chức năng liên quan tới cuộc gọi video trực tuyến giữa bác sĩ và bệnh nhân. Tiến trình này có thể hiện của lớp *OnlineMeeting*, có nhiệm vụ giúp bác sĩ và bệnh nhân thực hiện việc gọi video trực tuyến. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **NotificationSystemAccess:** Quản lý tất cả truy cập đến hệ thống con *NotificationSystem*. Tiến trình này cũng đồng bộ hóa quyền truy cập vào hệ

thông con *NotificationSystem* từ các tiến trình hệ thống khác. Chỉ có một thể hiện của tiến trình *NotificationSystemAccess*.

2.11. Xem và trả lời câu hỏi trên diễn đàn hỏi đáp

2.11.1. Mô hình tiến trình



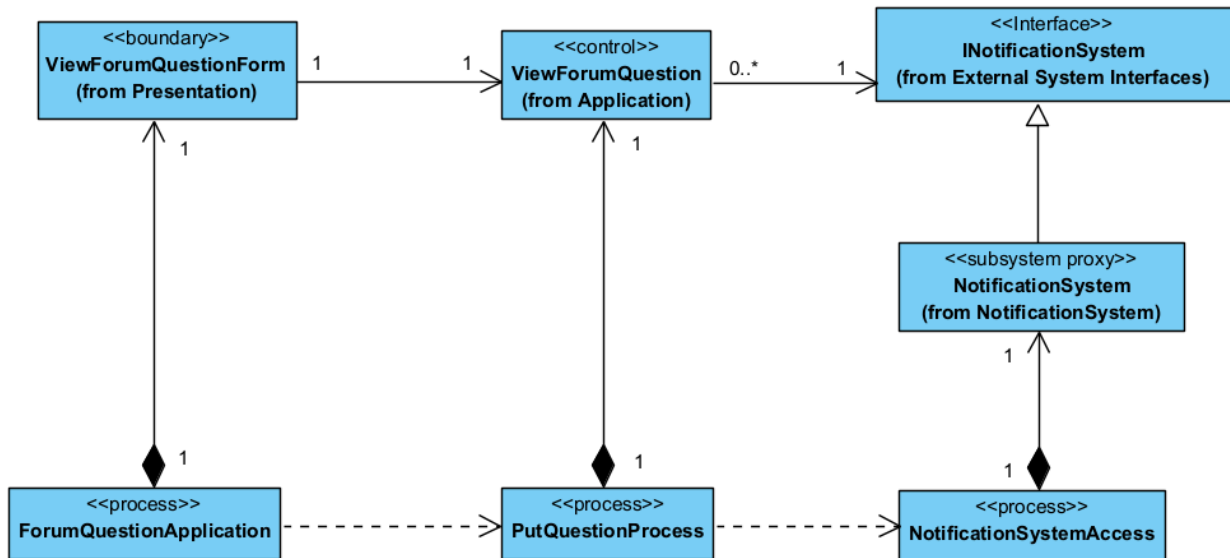
2.11.2. Mô tả thành phần tiến trình

- **ForumQuestionApplication:** Điều khiển các giao diện, biểu mẫu cho người dùng thao tác với các chức năng liên quan tới câu hỏi trên diễn đàn. Tiến trình này có thể hiện của lớp *ViewForumQuestionForm*, có nhiệm vụ giúp người dùng đặt câu hỏi trên diễn đàn hỏi đáp. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **ViewQuestionProcess:** Quản lý quá trình, thực hiện xử lý quá trình xem câu hỏi trên diễn đàn hỏi đáp. Có một thể hiện của tiến trình này tương ứng với mỗi lần người dùng xem câu hỏi trên diễn đàn.
- **AnswerApplication:** Điều khiển các giao diện, biểu mẫu cho người dùng thao tác với các chức năng liên quan tới câu trả lời của câu hỏi trên diễn đàn. Tiến trình này có thể hiện của lớp *CreateAnswerForm*, có nhiệm vụ giúp người dùng trả lời câu hỏi trên diễn đàn. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **AnswerProcess:** Quản lý quá trình, thực hiện xử lý quá trình trả lời câu hỏi trên diễn đàn hỏi đáp. Có một thể hiện của tiến trình này tương ứng với mỗi lần người dùng trả lời câu hỏi trên diễn đàn.

- **NotificationSystemAccess:** Quản lý tất cả truy cập đến hệ thống con *NotificationSystem*. Tiến trình này cũng đồng bộ hóa quyền truy cập vào hệ thống con *NotificationSystem* từ các tiến trình hệ thống khác. Chỉ có một thể hiện của tiến trình *NotificationSystemAccess*.

2.12. Đặt câu hỏi trên diễn đàn hỏi đáp

2.12.1. Mô hình tiến trình

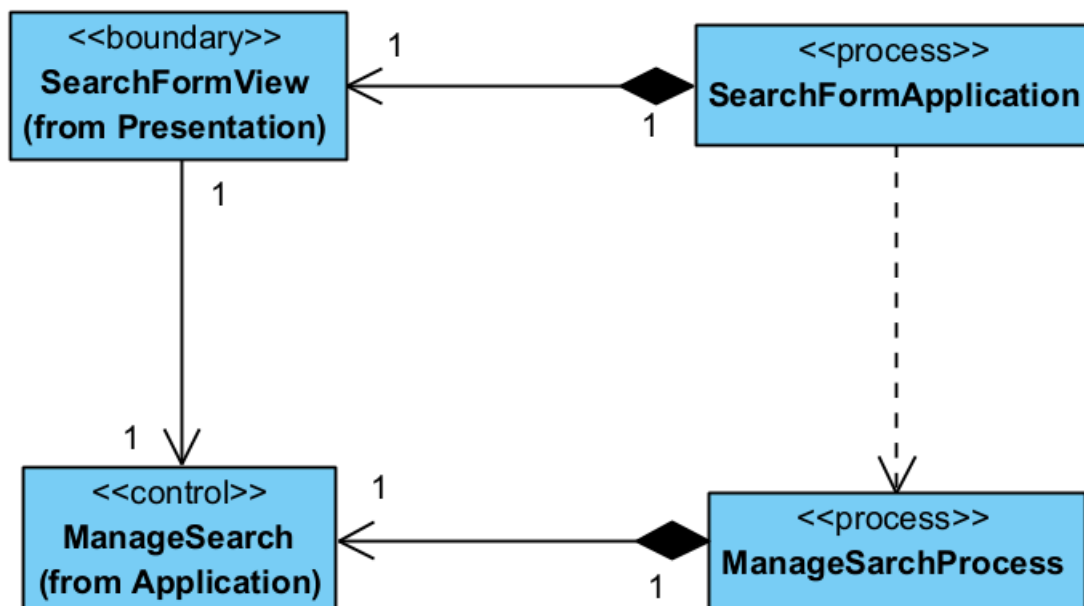


2.12.2. Mô tả thành phần tiến trình

- **ForumQuestionApplication:** Điều khiển các giao diện, biểu mẫu cho người dùng thao tác với các chức năng liên quan tới câu hỏi trên diễn đàn. Tiến trình này có thể hiện của lớp *ViewForumQuestionForm*, có nhiệm vụ giúp người dùng đặt câu hỏi trên diễn đàn hỏi đáp. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **PutQuestionProcess:** Quản lý quá trình, thực hiện xử lý quá trình đặt câu hỏi trên diễn đàn hỏi đáp. Có một thể hiện của tiến trình này tương ứng với mỗi lần người dùng đặt câu hỏi trên diễn đàn.
- **NotificationSystemAccess:** Quản lý tất cả truy cập đến hệ thống con *NotificationSystem*. Tiến trình này cũng đồng bộ hóa quyền truy cập vào hệ thống con *NotificationSystem* từ các tiến trình hệ thống khác. Chỉ có một thể hiện của tiến trình *NotificationSystemAccess*.

2.13. Tìm kiếm dịch vụ/bệnh viện/bác sĩ

2.13.1. Mô hình tiến trình

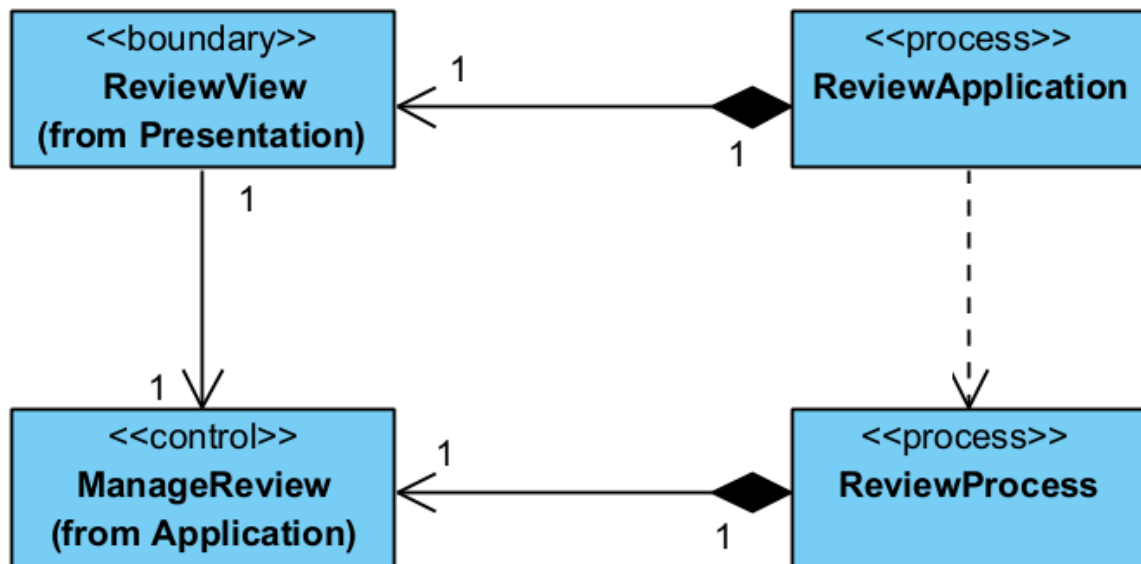


2.13.2. Mô tả thành phần tiến trình

- **SearchFormApplication**: Điều khiển các giao diện, biểu mẫu dành cho người cần tìm kiếm dịch vụ/bệnh viện/bác sĩ. Tiến trình này có thể hiện của lớp **SearchFormView** có nhiệm vụ để người dùng có thể xem danh sách các dịch vụ/bệnh viện/bác sĩ cần tìm và thực hiện các thao tác tiếp theo. Có một instance của tiến trình này tương ứng với người truy cập hệ thống.
- **ManageSearchProcess**: Quản lý quá trình thực hiện người dùng từ lúc truy cập trang tìm kiếm đến khi chuyển sang trang khác của hệ thống. Có một instance của tiến trình này tương ứng với người truy cập hệ thống.

2.14. Đánh giá

2.14.1. Mô hình tiến trình

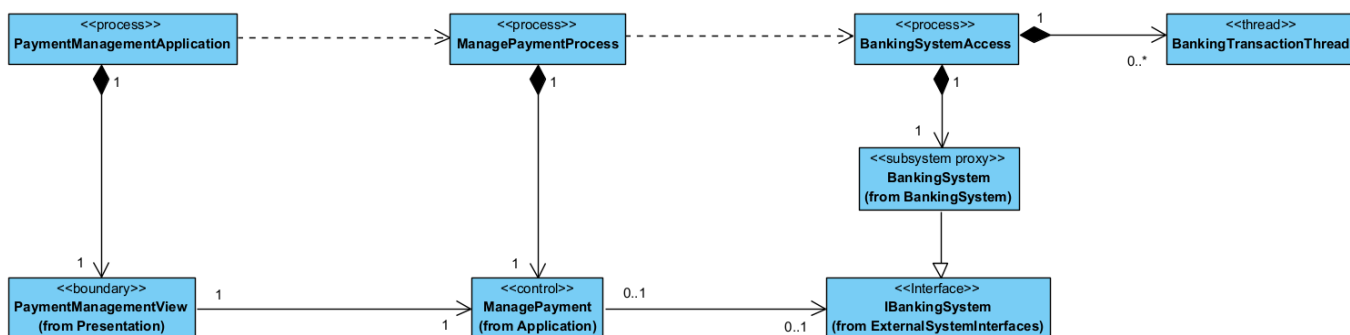


2.14.2. Mô tả thành phần tiến trình

- **ReviewApplication**: quản lý các giao diện, biểu mẫu mà người dùng sử dụng để quản lý dịch vụ. Đây là một instance của ReviewView.
- **ReviewProcess**: điều khiển việc thực thi đánh giá dịch vụ. Một instance của ManageReview được sử dụng.

2.15. Thanh toán

2.15.1. Mô hình tiến trình

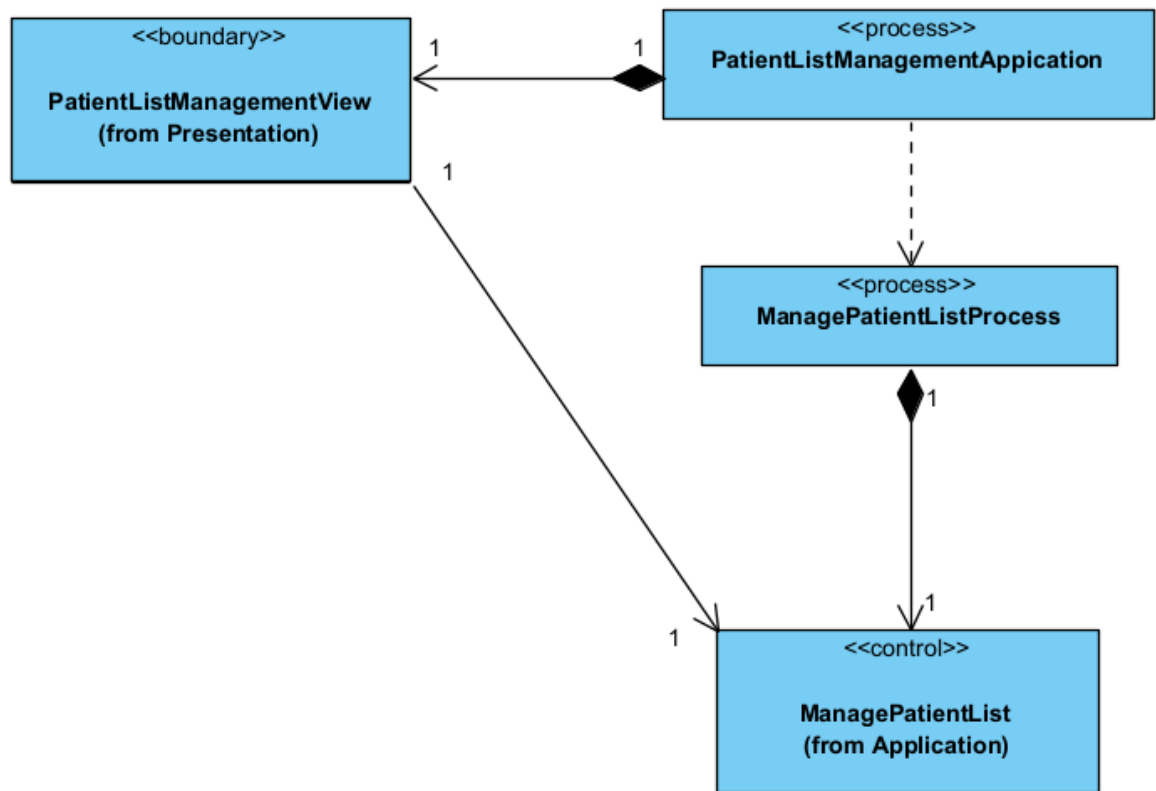


2.15.2. Mô tả thành phần tiến trình

- **PaymentManagementApplication**: quản lý giao diện, biểu mẫu mà người dùng tương tác để thanh toán. Một instance của boundary class **PaymentManagementView** được sử dụng.
- **ManagePaymentProcess**: điều khiển quá trình thực thi thanh toán. Một instance của control class **ManagePayment** được sử dụng.
- **BankingSystem**: hệ thống ngân hàng, ví điện tử hỗ trợ quản lý tài chính của người dùng.
- **IBankingSystem**: giao diện tương tác giữa hệ thống với hệ thống ngân hàng điện tử.
- **BankingSystemAccess**: quản lý tất cả truy cập đến hệ thống ngân hàng điện tử.
- **BankTransactionThread**: đóng gói một giao dịch đến hệ thống ngân hàng điện tử.

2.16. Quản lý danh sách bệnh nhân đã khám

2.16.1. Mô hình tiến trình

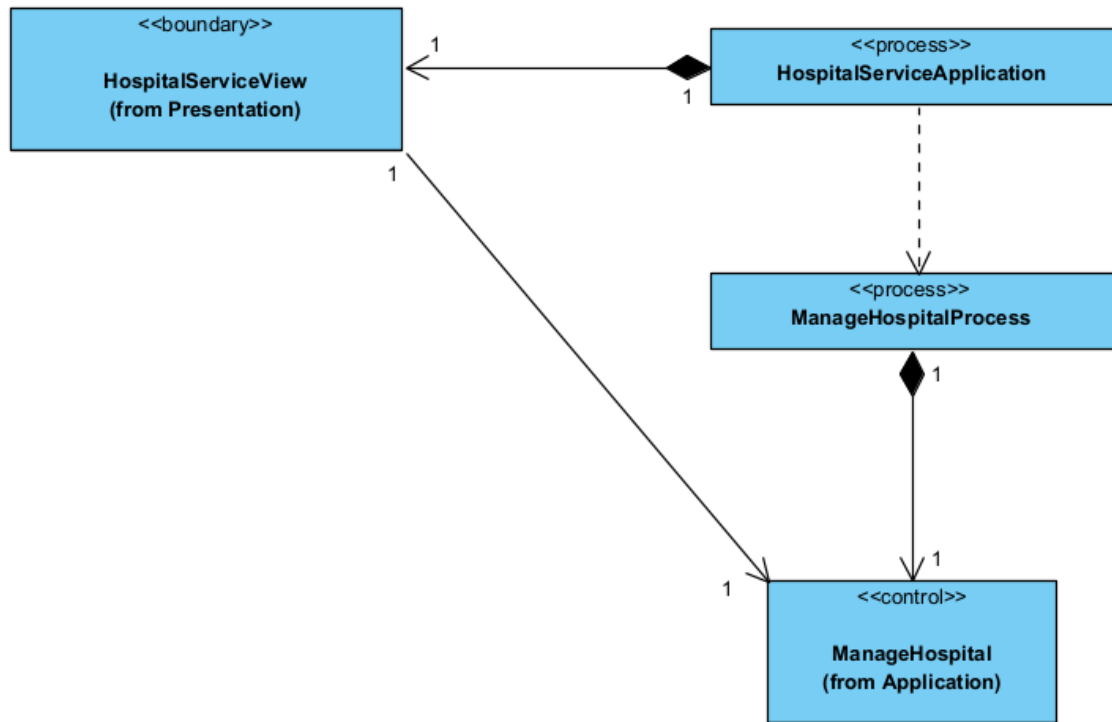


2.16.2. Mô tả thành phần tiến trình

- **PatientListManagementApplication**: quản lý giao diện, biểu mẫu để bác sĩ tương tác để xem thông tin các bệnh nhân của mình và cập nhật hồ sơ sức khỏe của họ. Một instance của boundary class **PatientListManagementView** được sử dụng.
- **ManagePatientListProcess**: Quản lý quá trình bác sĩ thực hiện việc xem thông tin các bệnh nhân của mình và cập nhật hồ sơ sức khỏe của họ. Một instance của control class **ManagePatientList** được sử dụng.

2.17. Đặt lịch khám tại bệnh viện

2.17.1. Mô hình tiến trình

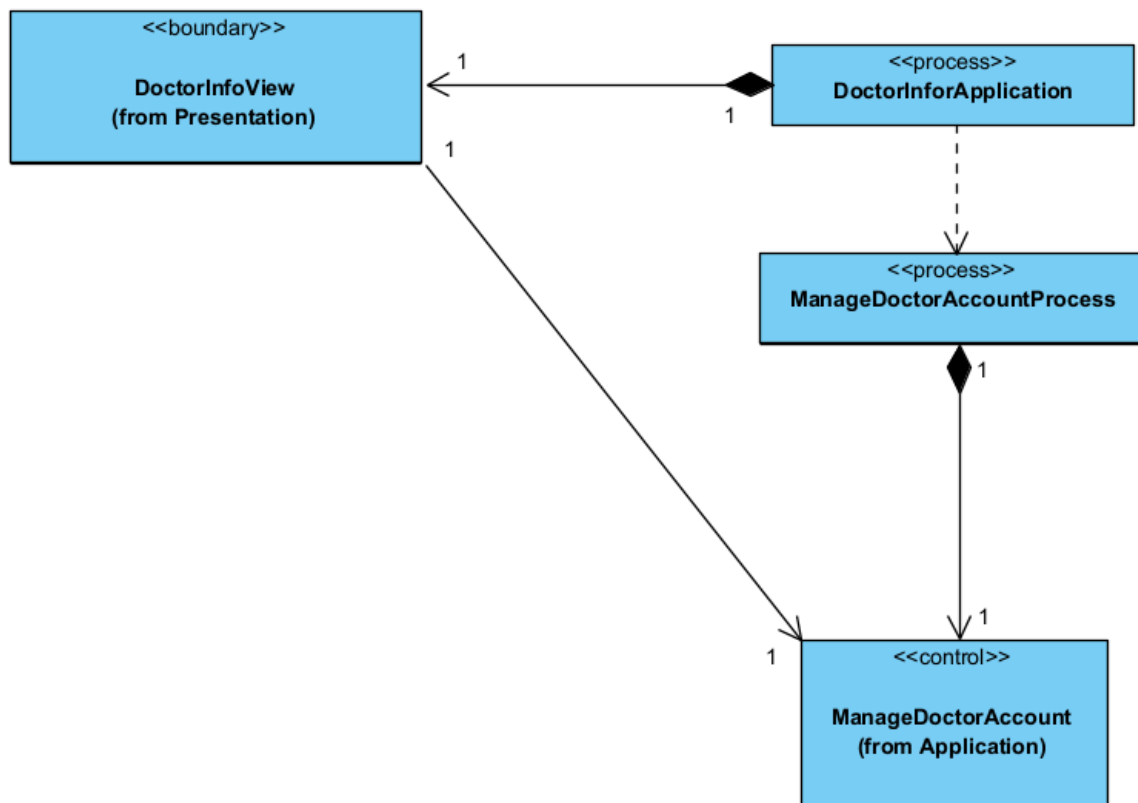


2.17.2. Mô tả thành phần tiến trình

- **HospitalServiceApplication:** quản lý giao diện, biểu mẫu để người dùng xem các dịch vụ của bệnh viện và đặt lịch khám tại bệnh viện. Một instance của boundary class **HospitalServiceView** được sử dụng.
- **ManageHospitalProcess:** Quản lý quá trình người dùng xem các dịch vụ của bệnh viện và đặt lịch khám tại bệnh viện. Một instance của control class **ManageHospital** được sử dụng.

2.18. Đặt lịch khám với bác sĩ

2.18.1. Mô hình tiến trình

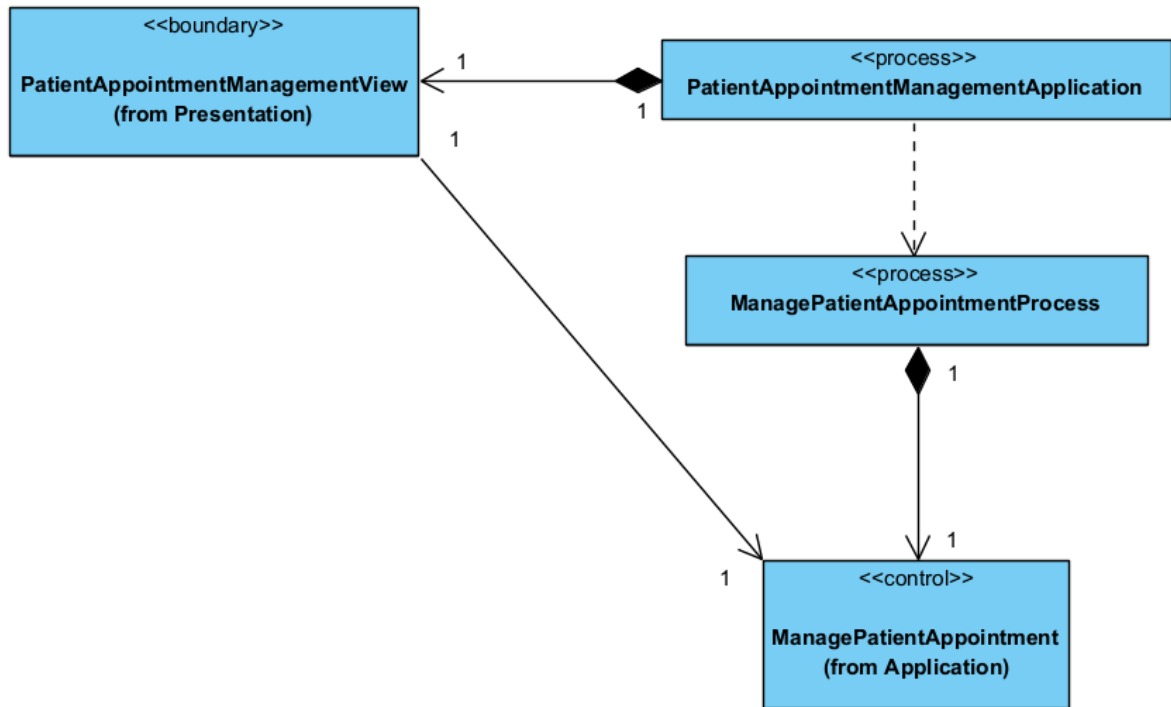


2.18.2. Mô tả thành phần tiến trình

- **DoctorInforApplication:** quản lý giao diện, biểu mẫu để người dùng xem thông tin bác sĩ và đặt lịch khám với bác sĩ. Một instance của boundary class **DoctorInfoView** được sử dụng.
- **ManageDoctorAccountProcess:** quản lý quá trình người dùng xem thông tin bác sĩ và đặt lịch khám với bác sĩ. Một instance của control class **ManageDoctorAccount** được sử dụng.

2.19. Quản lý lịch sử đặt khám

2.19.1. Mô hình tiến trình

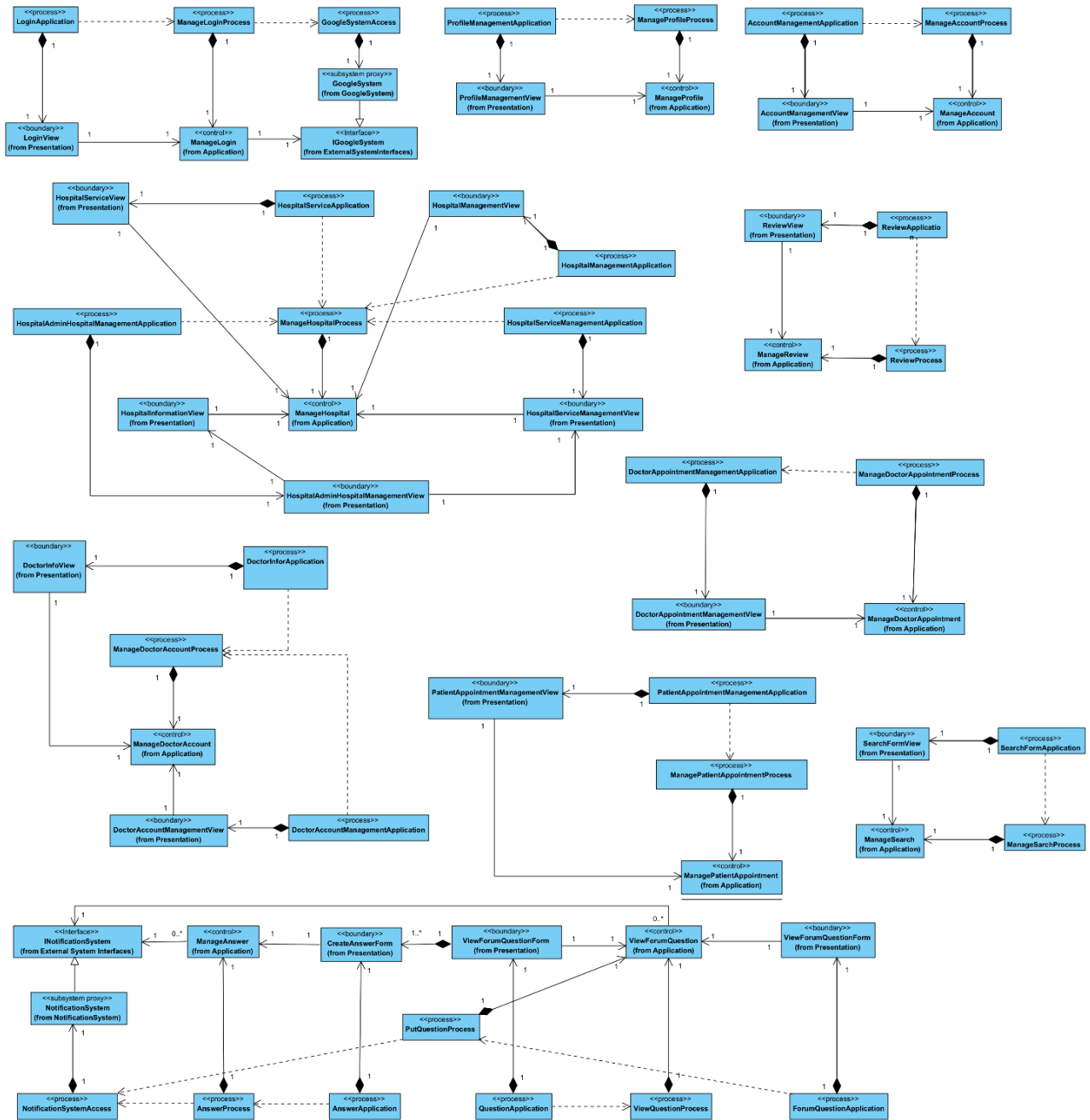


2.19.2. Mô tả thành phần tiến trình

- **PatientAppointmentManagementApplication:** quản lý giao diện, biểu mẫu để bệnh nhân tương tác để xem thông tin các lịch khám đã đặt. Một instance của boundary class **PatientAppointmentManagementView** được sử dụng.
- **ManagePatientAppointmentProcess:** Quản lý quá trình bệnh nhân thực hiện việc xem thông tin các lịch khám đã đặt. Một instance của control class **ManagePatientAppointment** được sử dụng.

3. Mô tả tương tranh (Concurrency)

3.1. Mô hình tiến trình



- **ManageAccountProcess:** điều khiển quá trình thực thi quản lý danh sách tài khoản hệ thống. Một instance của control class ManageAccount được sử dụng.
- **HospitalManagementApplication:** quản lý giao diện danh sách bệnh viện. Một instance của boundary class HospitalManagementView được sử dụng.
- **ManageHospitalProcess:** điều khiển quá trình thực thi quản lý danh sách bệnh viện trong hệ thống, quản lý thông tin và dịch vụ bệnh viện, quản lý quá trình người dùng xem các dịch vụ của bệnh viện và đặt lịch khám tại bệnh viện. Một instance của control class ManageHospital được sử dụng.
- **DoctorAccountManagementApplication:** Quản lý giao diện danh sách tài khoản bác sĩ. Đây là một instance của boundary class DoctorAccountManagementView.
- **ManageDoctorAccountProcess:** điều khiển quá trình thực thi quản lý danh sách tài khoản bác sĩ, quản lý quá trình người dùng xem thông tin bác sĩ và đặt lịch khám với bác sĩ. Một instance của control class ManageDoctorAccount được sử dụng.
- **HospitalAdminHospitalManagementApplication:** Quản lý giao diện quản lý bệnh viện. Đây là một instance của boundary class HospitalAdminHospitalManagementView.
- **HospitalInformationApplication:** Quản lý giao diện quản lý thông tin bệnh viện. Đây là một instance của boundary class HospitalInformationView.
- **HospitalServiceManagementApplication:** Quản lý giao diện quản lý dịch vụ bệnh viện. Đây là một instance của boundary class HospitalServiceManagementView.
- **DoctorAppointmentManagementApplication:** Quản lý giao diện quản lý lịch khám đã được đặt của bác sĩ. Đây là một instance của boundary class DoctorAppointmentManagementView.
- **ManageDoctorAppointmentProcess:** điều khiển quá trình thực thi quản lý lịch khám đã được đặt của bác sĩ. Một instance của control class ManageDoctorAppointment được sử dụng.

- **PatientListManagementApplication:** Quản lý giao diện quản lý danh sách bệnh nhân đã khám của bác sĩ, đồng thời quản lý hồ sơ sức khỏe bệnh nhân. Đây là một instance của boundary class *PatientListManagementView*.
- **ManagePatientListProcess:** điều khiển quá trình thực thi quản lý danh sách bệnh nhân đã khám của bác sĩ, quản lý quá trình bác sĩ thực hiện việc xem thông tin các bệnh nhân của mình và cập nhật hồ sơ sức khỏe của họ. Một instance của control class *ManagePatientList* được sử dụng.
- **HealthRecordApplication:** Điều khiển các giao diện, biểu mẫu cho người dùng thao tác với các chức năng liên quan tới hồ sơ sức khỏe. Tiến trình này có thể hiện của lớp *ViewHealthRecordForm*, có nhiệm vụ giúp bệnh nhân xem hồ sơ sức khỏe cá nhân. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **ViewHealthRecordProcess:** Quản lý quá trình, thực hiện xử lý quá trình xem hồ sơ sức khỏe của bệnh nhân. Có một thể hiện của tiến trình này tương ứng với mỗi bệnh nhân tự xem hồ sơ sức khỏe.
- **AppointmentApplication:** Điều khiển các giao diện, biểu mẫu cho người dùng thao tác với các chức năng liên quan tới lịch khám giữa bác sĩ và bệnh nhân. Tiến trình này có thể hiện của lớp *ViewAppointmentForm*, có nhiệm vụ giúp bác sĩ thực hiện việc xem lịch khám cá nhân. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **ExamineProcess:** Quản lý quá trình, thực hiện xử lý quá trình khám trực tuyến. Có một thể hiện của tiến trình này tương ứng với mỗi lần người dùng thực hiện việc khám trực tuyến.
- **MeetingApplication:** Điều khiển các giao diện cho người dùng thao tác với các chức năng liên quan tới cuộc gọi video trực tuyến giữa bác sĩ và bệnh nhân. Tiến trình này có thể hiện của lớp *OnlineMeeting*, có nhiệm vụ giúp bác sĩ và bệnh nhân thực hiện việc gọi video trực tuyến. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **NotificationSystemAccess:** Quản lý tất cả truy cập đến hệ thống *NotificationSystem*. Tiến trình này cũng đồng bộ hóa quyền truy cập vào hệ

thông con *NotificationSystem* từ các tiến trình hệ thống khác. Chỉ có một thể hiện của tiến trình *NotificationSystemAccess*.

- **ForumQuestionApplication:** Điều khiển các giao diện, biểu mẫu cho người dùng thao tác với các chức năng liên quan tới câu hỏi trên diễn đàn. Tiến trình này có thể hiện của lớp *ViewForumQuestionForm*, có nhiệm vụ giúp người dùng đặt câu hỏi trên diễn đàn hỏi đáp. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **ViewQuestionProcess:** Quản lý quá trình, thực hiện xử lý quá trình xem câu hỏi trên diễn đàn hỏi đáp. Có một thể hiện của tiến trình này tương ứng với mỗi lần người dùng xem câu hỏi trên diễn đàn.
- **AnswerApplication:** Điều khiển các giao diện, biểu mẫu cho người dùng thao tác với các chức năng liên quan tới câu trả lời của câu hỏi trên diễn đàn. Tiến trình này có thể hiện của lớp *CreateAnswerForm*, có nhiệm vụ giúp người dùng trả lời câu hỏi trên diễn đàn. Có một thể hiện của tiến trình này tương ứng với mỗi người dùng sử dụng hệ thống.
- **AnswerProcess:** Quản lý quá trình, thực hiện xử lý quá trình trả lời câu hỏi trên diễn đàn hỏi đáp. Có một thể hiện của tiến trình này tương ứng với mỗi lần người dùng trả lời câu hỏi trên diễn đàn.
- **PutQuestionProcess:** Quản lý quá trình, thực hiện xử lý quá trình đặt câu hỏi trên diễn đàn hỏi đáp. Có một thể hiện của tiến trình này tương ứng với mỗi lần người dùng đặt câu hỏi trên diễn đàn.
- **SearchFormApplication:** Điều khiển các giao diện, biểu mẫu dành cho người cần tìm kiếm dịch vụ/bệnh viện/bác sĩ. Tiến trình này có thể hiện của lớp *SearchFormView* có nhiệm vụ để người dùng có thể xem danh sách các dịch vụ/bệnh viện/bác sĩ cần tìm và thực hiện các thao tác tiếp theo. Có một instance của tiến trình này tương ứng với người truy cập hệ thống.
- **ManageSearchProcess:** Quản lý quá trình thực hiện người dùng từ lúc truy cập trang tìm kiếm đến khi chuyển sang trang khác của hệ thống. Có một instance của tiến trình này tương ứng với người truy cập hệ thống.

- **ReviewApplication:** quản lý các giao diện, biểu mẫu mà người dùng sử dụng để quản lý dịch vụ. Đây là một instance của ReviewView.
- **ReviewProcess:** điều khiển việc thực thi đánh giá dịch vụ. Một instance của ManageReview được sử dụng.
- **PaymentManagementApplication:** quản lý giao diện, biểu mẫu mà người dùng tương tác để thanh toán. Một instance của boundary class PaymentManagementView được sử dụng.
- **ManagePaymentProcess:** điều khiển quá trình thực thi thanh toán. Một instance của control class ManagePayment được sử dụng.
- **BankingSystem:** hệ thống ngân hàng, ví điện tử hỗ trợ quản lý tài chính của người dùng.
- **IBankingSystem:** giao diện tương tác giữa hệ thống với hệ thống ngân hàng điện tử.
- **BankingSystemAccess:** quản lý tất cả truy cập đến hệ thống ngân hàng điện tử.
- **BankTransactionThread:** đóng gói một giao dịch đến hệ thống ngân hàng điện tử.
- **PatientListManagementAppication:** quản lý giao diện, biểu mẫu để bác sĩ tương tác để xem thông tin các bệnh nhân của mình và cập nhật hồ sơ sức khỏe của họ. Một instance của boundary class PatientListManagementView được sử dụng.
- **HospitalServiceApplication:** quản lý giao diện, biểu mẫu để người dùng xem các dịch vụ của bệnh viện và đặt lịch khám tại bệnh viện. Một instance của boundary class HospitalServiceView được sử dụng.
- **DoctorInforApplication:** quản lý giao diện, biểu mẫu để người dùng xem thông tin bác sĩ và đặt lịch khám với bác sĩ. Một instance của boundary class DoctorInfoView được sử dụng.
- **PatientAppointmentManagementApplication:** quản lý giao diện, biểu mẫu để bệnh nhân tương tác để xem thông tin các lịch khám đã đặt. Một

instance của boundary class PatientAppointmentManagementView được sử dụng.

- **ManagePatientAppointmentProcess:** Quản lý quá trình bệnh nhân thực hiện việc xem thông tin các lịch khám đã đặt. Một instance của control class ManagePatientAppointment được sử dụng.