

**ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



**BÀI TẬP LỚN
CÔNG NGHỆ PHẦN MỀM**

Chủ đề: Hệ thống đặt phòng khách sạn

Lớp: INT2208_7

Nhóm: 7

Thành viên:

- Phạm Đức Thiện
- Phạm Tuấn Việt
- Đoàn Văn Tuyền
- Phạm Văn Minh

Mục lục

I Đặc tả yêu cầu	6
1 Đặt vấn đề	6
2 Bảng thuật ngữ	7
3 Đặc tả bổ sung	8
4 Mô hình Use Case	9
4.1 Danh sách tác nhân	9
4.2 Các ca sử dụng bậc cao	10
4.3 Các ca sử dụng dưới góc nhìn người dùng	11
4.4 Ca sử dụng Đăng nhập	12
4.5 Ca sử dụng Tra cứu thông tin phòng khách sạn	13
4.6 Ca sử dụng Tra cứu thông tin phương tiện di chuyển	14
4.7 Ca sử dụng Cho thuê phòng	14
5 Đặc tả Use Case	15
5.1 Đặt phòng khách sạn	15
5.2 Theo dõi phòng đã đặt	17
5.3 Đánh giá phòng đã thuê	19
5.4 Liên hệ với chủ khách sạn	21
5.5 Xem danh sách phòng yêu thích	23
5.6 Cho thuê phòng	25
5.7 Sửa thông tin phòng	27
5.8 Tra cứu thông tin phòng đang cho thuê	29
5.9 Xóa phòng khách sạn	31
5.10 Thanh toán	33

5.11	Tra cứu phương tiện	36
5.12	Đăng ký tài khoản	39
5.13	Đăng nhập	41
5.14	Lựa chọn ngôn ngữ	43
II	Phân tích hệ thống	45
1	Phân tích kiến trúc	45
1.1	Key abstraction	45
1.2	Thành phần cấp cao và sự phụ thuộc	47
2	Phân tích Use case	48
2.1	Biểu đồ tuần tự các use case	48
2.1.1	Đặt phòng khách sạn	48
2.1.2	Theo dõi phòng đã đặt	49
2.1.3	Đánh giá phòng đã thuê	50
2.1.4	Liên hệ với chủ khách sạn	51
2.1.5	Xem danh sách phòng yêu thích	52
2.1.6	Cho thuê phòng	53
2.1.7	Chỉnh sửa thông tin phòng khách sạn	54
2.1.8	Tra cứu thông tin phòng cho thuê	55
2.1.9	Xóa phòng khách sạn	57
2.1.10	Thanh toán	58
2.1.11	Tra cứu phương tiện	60
2.1.12	Đăng ký tài khoản	61
2.1.13	Đăng nhập	62
2.1.14	Lựa chọn ngôn ngữ	65
2.2	Biểu đồ lớp pha phân tích	66
2.2.1	Đặt phòng khách sạn	66
2.2.2	Theo dõi phòng đã đặt	67
2.2.3	Đánh giá phòng đã thuê	68
2.2.4	Liên hệ với chủ khách sạn	69

2.2.5	Xem danh sách phòng yêu thích	70
2.2.6	Cho thuê phòng	70
2.2.7	Chỉnh sửa thông tin phòng khách sạn	71
2.2.8	Tra cứu thông tin phòng cho thuê	71
2.2.9	Xóa phòng khách sạn	72
2.2.10	Thanh toán	73
2.2.11	Tra cứu phương tiện	73
2.2.12	Đăng ký tài khoản	74
2.2.13	Đăng nhập	74
2.2.14	Lựa chọn ngôn ngữ	76
2.3	Ánh xạ từ lớp phân tích tới cơ chế phân tích	76

III Thiết kế hệ thống 78

1	Xác định các thành phần cần thiết kế	78
1.1	Subsystem Context	78
1.2	Analysis-to-Design-to-Implementation Mechanisms Map	80
1.2.1	Cơ chế Persistency - ObjectStore OODBMS	81
1.2.2	Cơ chế Distribution	85
1.2.3	Cơ chế Security	86
1.3	Analysis-Class-to-Design-Element Map	88
1.4	Design-Element-to-Owning-Package Map	89
1.5	Packages and Their Dependency	90
2	Mô tả kiến trúc thực thi	91
3	Thiết kế Use Case	92
3.1	Thiết kế biểu đồ tuần tự	92
3.1.1	Đặt phòng khách sạn	92
3.1.2	Theo dõi phòng đã đặt	93
3.1.3	Đánh giá phòng đã thuê	94
3.1.4	Liên hệ với chủ khách sạn	95
3.1.5	Xem danh sách phòng yêu thích	96

3.1.6	Cho thuê phòng	97
3.1.7	Sửa thông tin phòng	98
3.1.8	Tra cứu thông tin phòng đang cho thuê	99
3.1.9	Xóa phòng khách sạn	101
3.1.10	Thanh toán	102
3.1.11	Tra cứu phương tiện	104
3.1.12	Đăng ký tài khoản	105
3.1.13	Đăng nhập	106
3.1.14	Lựa chọn ngôn ngữ	109
3.2	Thiết kế biểu đồ lớp	110
3.2.1	Đặt phòng khách sạn	110
3.2.2	Theo dõi phòng đã đặt	110
3.2.3	Đánh giá phòng đã thuê	111
3.2.4	Liên hệ với chủ khách sạn	111
3.2.5	Xem danh sách phòng yêu thích	111
3.2.6	Cho thuê phòng	112
3.2.7	Sửa thông tin phòng	112
3.2.8	Tra cứu thông tin phòng đang cho thuê	113
3.2.9	Xóa phòng khách sạn	114
3.2.10	Thanh toán	115
3.2.11	Tra cứu phương tiện	116
3.2.12	Đăng ký tài khoản	117
3.2.13	Đăng nhập	118
3.2.14	Lựa chọn ngôn ngữ	120
4	Thiết kế Hệ thống con	121
4.1	User Subsystem	121
4.2	Hotel Subsystem	127
5	Thiết kế lớp	134
6	Thiết kế Cơ sở dữ liệu	146

I Đặc tả yêu cầu

1 Đặt vấn đề

Hiện nay, với sự bùng nổ của ngành du lịch, các loại hình tham quan trải nghiệm cùng sự phát triển mạnh mẽ của điện thoại thông minh, nhu cầu tìm kiếm và đặt phòng khách sạn trực tuyến tại Việt Nam cũng như trên toàn thế giới đang ngày càng gia tăng. Tuy nhiên, giữa khối lượng thông tin khổng lồ và không phải lúc nào cũng chính xác, du khách thường gặp khó khăn và mất nhiều thời gian để chọn lựa khách sạn phù hợp với nhu cầu của mình.

Chính vì vậy, chúng tôi đã phát triển ứng dụng "TravelHelper"—một công cụ đặt phòng khách sạn và phương tiện thông minh, với mục tiêu cung cấp cho du khách trong và ngoài nước một giải pháp đặt lịch tiện lợi, an toàn và tiết kiệm. Ứng dụng này cho phép người dùng tìm kiếm khách sạn và phương tiện di chuyển đáp ứng các tiêu chí như vị trí, giá cả, đánh giá, và tiện nghi, giúp du khách dễ dàng lựa chọn khách sạn phù hợp nhất với yêu cầu của mình. Bên cạnh đó, TravelHelper còn hỗ trợ chủ khách sạn trong việc cho thuê phòng, quản lý lịch đặt phòng và kết nối với khách hàng.

TravelHelper là một ứng dụng di động tương thích với cả hai nền tảng IOS và Android, tích hợp Google Maps để cung cấp bản đồ trực quan. Du khách chỉ cần kết nối Internet là có thể dễ dàng tìm kiếm thông tin về khách sạn, phương tiện di chuyển, kiểm tra giá phòng, xem đánh giá từ các khách hàng trước, và thực hiện giao dịch thanh toán trực tuyến hoặc trực tiếp một cách nhanh chóng và an toàn. Ứng dụng đảm bảo rằng lịch đặt phòng sẽ không bị hủy bỏ mà không có sự đồng ý của cả du khách và chủ khách sạn. Sau khi đặt phòng, cả du khách và chủ khách sạn đều nhận được thông báo và thông tin cần thiết để liên lạc và trao đổi thêm. Khi kết thúc kỳ nghỉ, du khách có thể để lại đánh giá về khách sạn và các đánh giá này sẽ được công khai để giúp các du khách khác dễ dàng tham khảo. Chủ khách sạn cần xác thực thông tin để đăng ký sử dụng dịch vụ cho thuê khách sạn, vì vậy có thể đảm bảo được sự an toàn và minh bạch cho du khách.

Đối tượng người dùng của TravelHelper là bất kỳ ai có nhu cầu đặt phòng khách sạn qua điện thoại thông minh, từ du khách trong nước đến quốc tế. Ứng dụng hỗ trợ nhiều ngôn ngữ, bao gồm Tiếng Việt và Tiếng Anh, và sẽ mở rộng thêm các ngôn ngữ khác trong tương lai.

2 Bảng thuật ngữ

Bảng thuật ngữ xác định các thuật ngữ cụ thể cho ứng dụng TravelHelper, giải thích các thuật ngữ người đọc có thể không biết trong mô tả ca sử dụng và các tài liệu khác.

Người dùng:

Là người sử dụng hệ thống, có thể có hoặc không có tài khoản của hệ thống. Bao gồm cả du khách và chủ khách sạn.

Hệ quản trị cơ sở dữ liệu:

Là phần mềm quản lý, tổ chức và điều phối các hoạt động liên quan đến cơ sở dữ liệu, bao gồm lưu trữ, truy vấn, cập nhật và bảo mật dữ liệu.

Cơ sở dữ liệu:

Là một tập hợp dữ liệu được tổ chức, lưu trữ và quản lý một cách có cấu trúc, cho phép dễ dàng truy xuất, cập nhật và xử lý thông tin.

API Bản đồ (Google Map API):

Là API bản đồ được tích hợp cho hệ thống, có chức năng định vị và xác định các vị trí của khách sạn, phòng, địa điểm du lịch và hướng dẫn đường đi cho người dùng.

Hệ thống thanh toán:

Là hệ thống con có thể truy cập, truy vấn và xử lý cơ sở dữ liệu thanh toán và hóa đơn. Hệ thống này hỗ trợ các phương thức thanh toán như thẻ tín dụng, thẻ ghi nợ, ví điện tử và thanh toán trực tuyến.

Khách sạn:

Là cơ sở lưu trú phục vụ cho du khách với các dịch vụ phòng ngủ, nhà hàng, giải trí và các tiện ích khác. Phòng: Là một đơn vị cung cấp chỗ ở trong khách sạn, có thể được phân theo loại phòng (phòng đơn, phòng đôi, phòng suite) và mức giá.

Phương tiện:

Là các dịch vụ di chuyển được hệ thống cung cấp như máy bay, taxi, xe bus,... Du khách: Là người dùng hệ thống với mục đích đặt phòng và phương tiện.

Chủ khách sạn:

Là người dùng hệ thống với mục đích cho thuê phòng khách sạn.

Đánh giá khách sạn:

Là quá trình mà du khách cung cấp ý kiến về trải nghiệm của mình tại khách sạn, bao gồm các yếu tố như chất lượng phòng. Các đánh giá có thể được hiển thị công khai để hỗ trợ các khách hàng khác trong việc lựa chọn khách sạn.

Đặt phòng:

Là hành động mà người dùng lựa chọn và xác nhận việc thuê phòng tại một khách sạn vào một thời điểm nhất định thông qua hệ thống.

Cho thuê phòng:

Chủ khách sạn cung cấp dịch vụ đặt phòng trên hệ thống. Đặt phương tiện: Là hành động của du khách lựa chọn đặt vé trước của một phương tiện.

Hủy phòng:

Là hành động người dùng yêu cầu hủy bỏ đơn đặt phòng đã xác nhận. Các quy định

về hủy phòng có thể áp dụng phí hủy dựa trên thời gian yêu cầu hủy.

Tìm kiếm phòng:

Là tính năng cho phép người dùng tìm kiếm các phòng khách sạn có sẵn dựa trên các tiêu chí như địa điểm, loại phòng, giá cả và thời gian lưu trú.

3 Đặc tả bổ sung

Mục tiêu

Mục đích của tài liệu này là xác định các yêu cầu của ứng dụng đặt phòng khách sạn TravelHelper. Đặc tả bổ sung này liệt kê các yêu cầu nằm ngoài mô hình usecase để hoàn thiện các chức năng cần đạt được của hệ thống.

Phạm vi

Đặc tả bổ sung này áp dụng cho ứng dụng TravelHelper. Đặc tả này xác định các yêu cầu phi chức năng của ứng dụng; chẳng hạn như độ tin cậy, khả năng sử dụng, hiệu suất và khả năng hỗ trợ, cũng như các yêu cầu chức năng chung trong một số Use Case.

Chức năng

Nhiều người dùng có thể dùng ứng dụng cùng lúc.

Tính khả dụng

Hệ thống sẽ dễ dàng tương tác, các tác vụ xử lý nhanh, hoạt động trên cả hệ điều hành Android và IOS. Hệ thống yêu cầu kết nối mạng để truy cập.

Tính tin cậy

Hệ thống có thể sử dụng 24/7, đáp ứng được tần suất truy cập cao.

Tính bảo mật

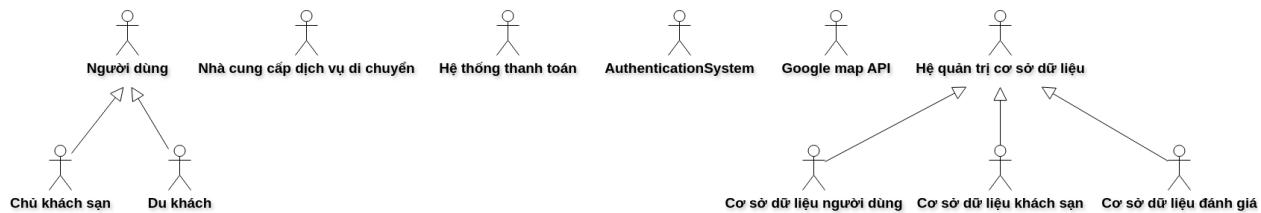
Hệ thống ngăn chặn các hành vi truy cập và đánh cắp dữ liệu trái phép. Hệ thống đảm bảo chỉ chủ tài khoản mới có thể nhìn thấy thông tin cá nhân của bản thân. Chỉ có chủ mới có thể thay đổi thông tin phòng.

Ràng buộc thiết kế

Hệ thống cung cấp giao diện cho điện thoại thông minh.

4 Mô hình Use Case

4.1 Danh sách tác nhân

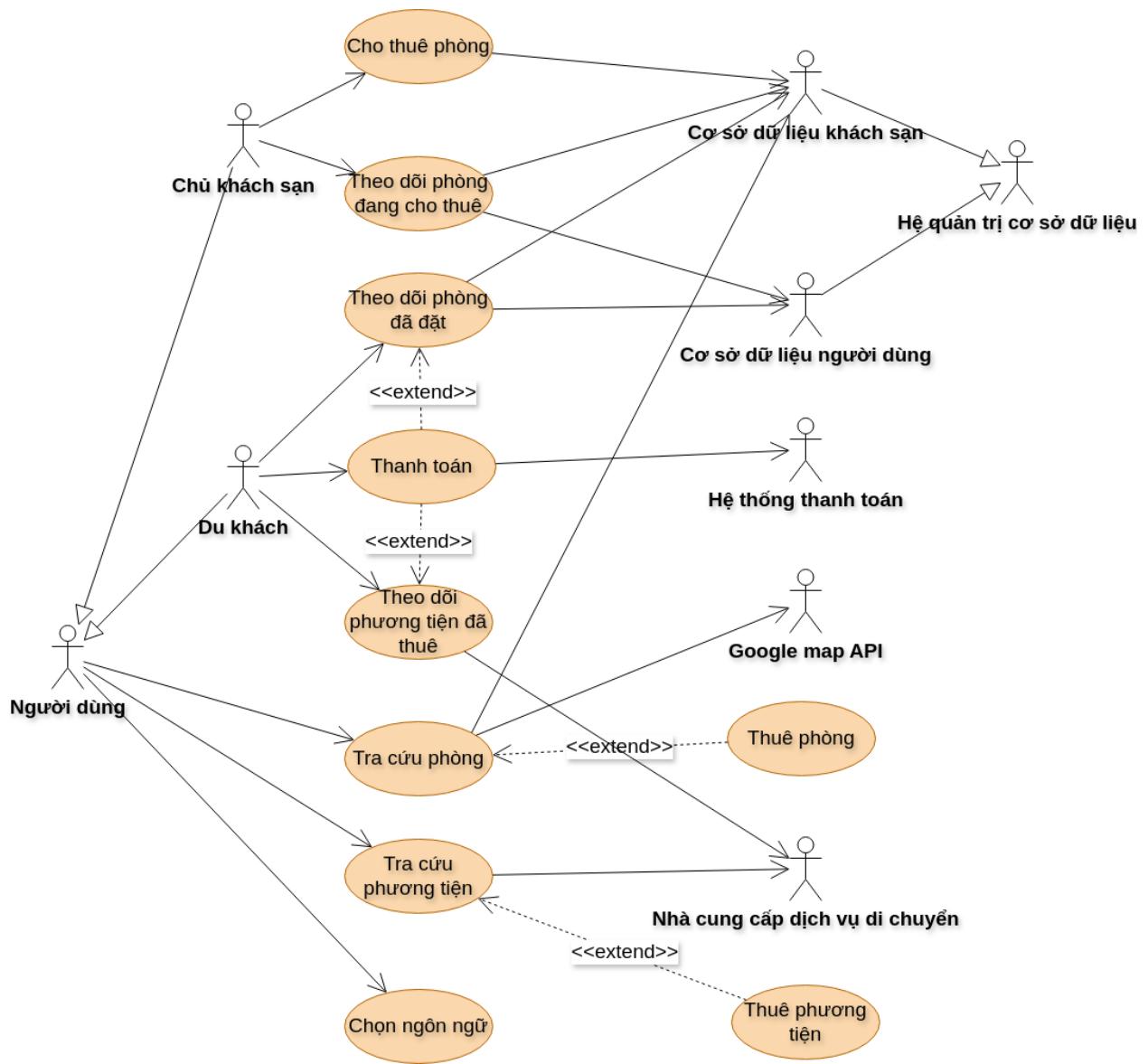


Hình 4.1.1: Biểu đồ ca sử dụng về tác nhân và sự phụ thuộc lẫn nhau giữa các tác nhân

Danh sách các tác nhân của hệ thống bao gồm:

- **Người dùng**: là khách hàng sử dụng ứng dụng gồm 2 loại là du khách(người dùng với mục đích đặt phòng và phương tiện) và chủ khách sạn(người dùng với mục đích cho thuê và quản lý phòng khách sạn)
- **Nhà cung cấp dịch vụ di chuyển**: là các công ty vận chuyển như taxi, xe buýt, máy bay. Được tích hợp với hệ thống để hiển thị các lựa chọn phương tiện di chuyển và thực hiện thanh toán.
- **Hệ thống thanh toán**: xử lý giao dịch tài chính giữa người dùng với chủ khách sạn hoặc nhà cung cấp dịch vụ vận chuyển.
- **Hệ quản trị cơ sở dữ liệu**: là phần mềm cung cấp dịch vụ quản lý cơ sở dữ liệu. Trong đó có 3 cơ sở dữ liệu chính: cơ sở dữ liệu người dùng, cơ sở dữ liệu khách sạn, cơ sở dữ liệu đánh giá.
- **AuthenticationSystem**: là phần mềm cung cấp dịch vụ xác thực người dùng và khách sạn như CCCD, giấy xác nhận kinh doanh.
- **Google map API**: API cung cấp khả năng tích hợp bản đồ và định vị vào ứng dụng.

4.2 Các ca sử dụng bậc cao



Hình 4.2.1: Biểu đồ ca sử dụng bậc cao

Tác nhân người dùng sử dụng ứng dụng để đặt lịch trực tuyến. Các ca sử dụng bậc cao bao gồm **Tra cứu phòng**, **Tra cứu phương tiện**, **Cho thuê phòng**, **Theo dõi phòng đã thuê**, **Theo dõi phương tiện đã thuê**, **Theo dõi phòng đang cho thuê** và **Thanh toán**. Tra cứu phòng và tra cứu phương tiện dành cho du khách muốn tìm kiếm và thuê các loại hình dịch vụ này. Theo dõi phòng và theo dõi phương tiện đã thuê dành cho du khách đã thực hiện thuê và muốn kiểm tra tình trạng các dịch vụ. Cho thuê phòng và theo dõi phòng đang cho thuê dành cho chủ khách sạn muốn cung cấp dịch vụ khách sạn và quản lý dịch vụ của mình. Thanh toán là ca sử dụng phát sinh khi người dùng đã thuê một dịch vụ mất phí như phòng khách sạn và phương tiện và dùng để thanh toán cho các dịch vụ này.

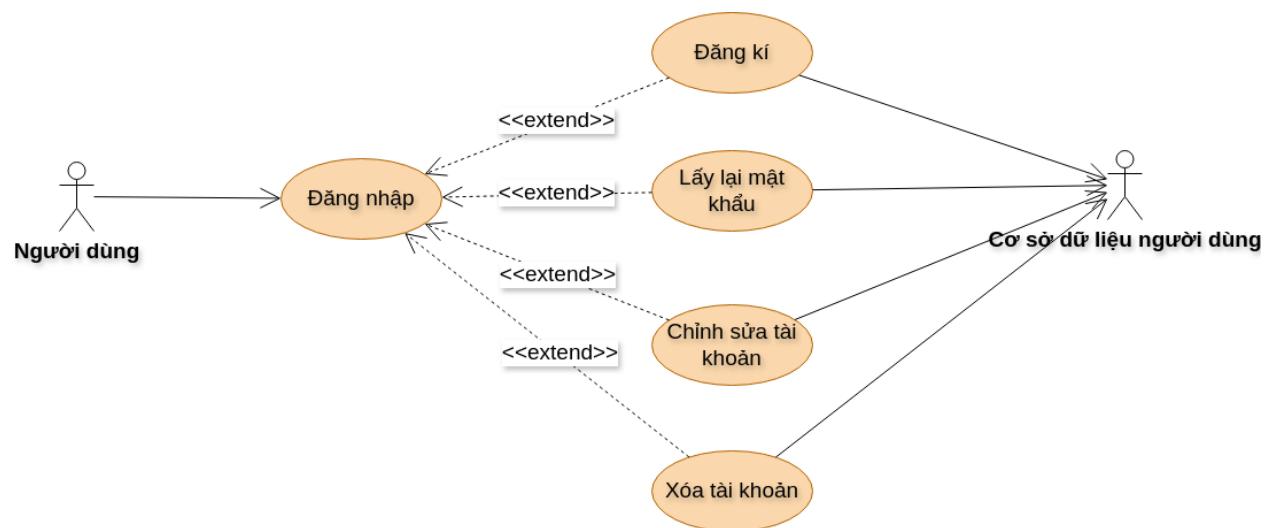
4.3 Các ca sử dụng dưới góc nhìn người dùng



Hình 4.3.1: Biểu đồ ca sử dụng dưới góc nhìn của Người dùng

Ca sử dụng dưới góc nhìn người dùng phân tích cụ thể các hoạt động và mục tiêu của người dùng khi sử dụng các ca sử dụng và bổ sung thêm các chi tiết cho biểu đồ ca sử dụng bậc cao.

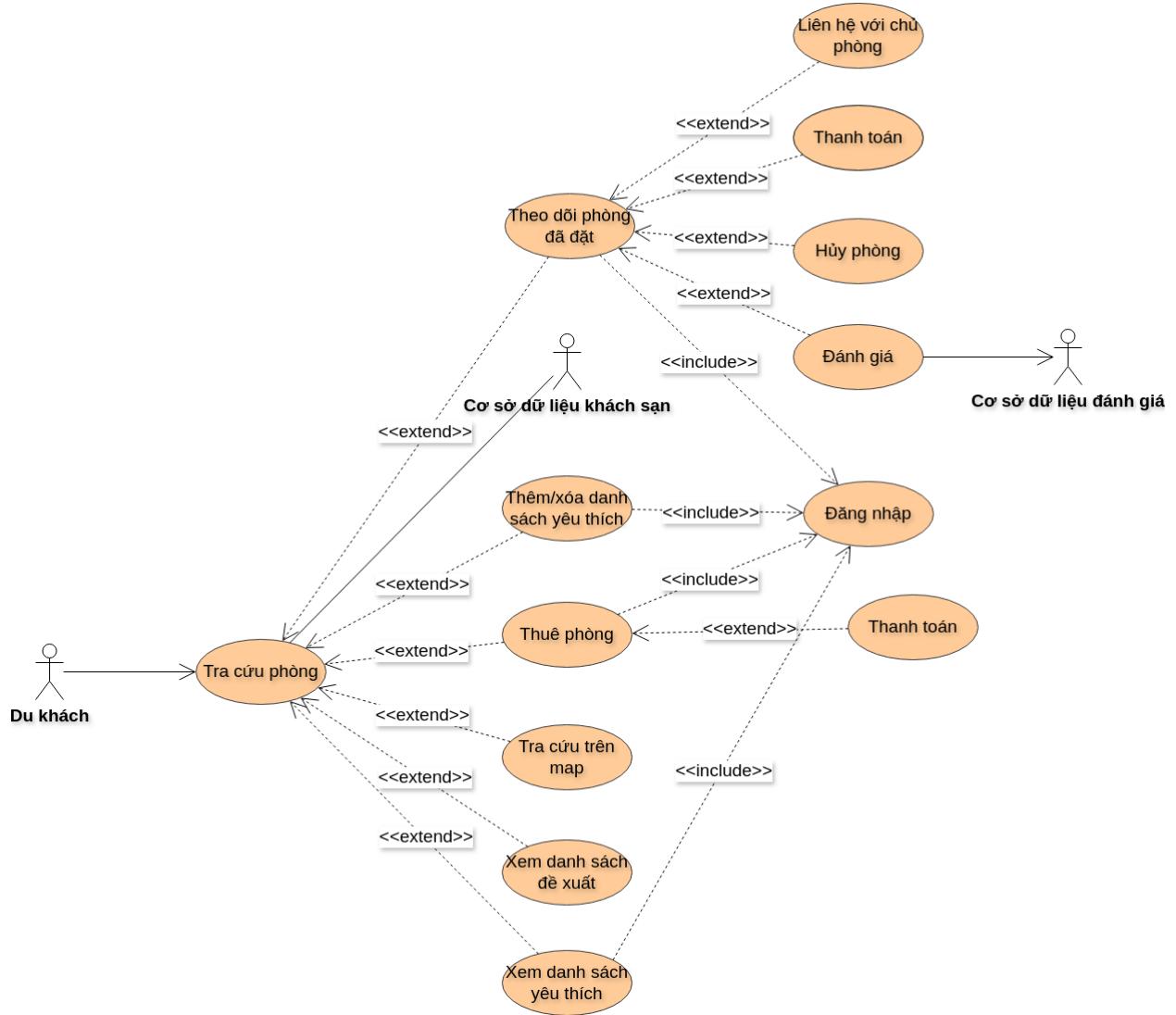
4.4 Ca sử dụng Đăng nhập



Hình 4.4.1: Biểu đồ ca sử dụng Đăng nhập

Ca sử dụng **Đăng nhập** có thể dùng sau khi người dùng đã đăng ký, người dùng có thể lựa chọn các tùy chọn thêm như lấy lại mật khẩu và ghi nhớ mật khẩu ở ca sử dụng này.

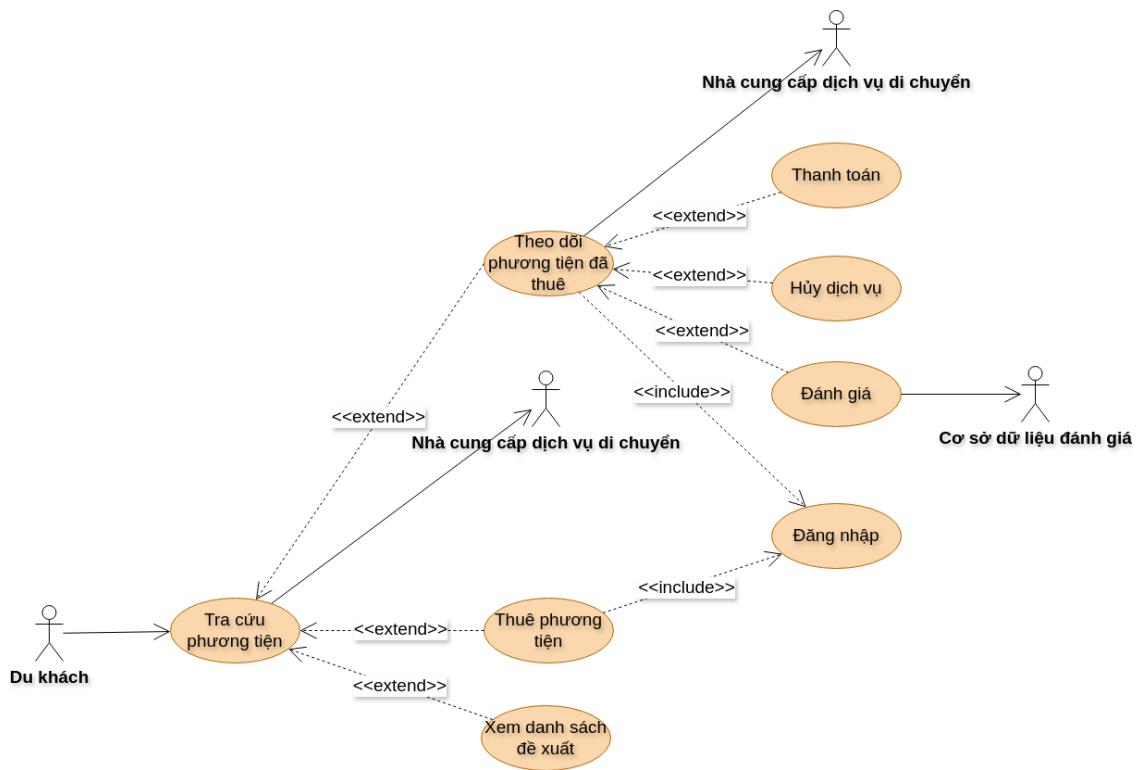
4.5 Ca sử dụng Tra cứu thông tin phòng khách sạn



Hình 4.5.1: Biểu đồ ca sử dụng Tra cứu thông tin phòng khách sạn

Ca sử dụng **Tra cứu phòng** được người dùng sử dụng khi muốn tìm kiếm thông tin về phòng khách sạn. Người dùng cần phải đăng nhập để có thể sử dụng các tính năng bổ sung như cập nhật và xem danh sách yêu thích, đặt phòng và theo dõi phòng đã đặt. Người dùng cũng được cung cấp dịch vụ tìm phòng trên bản đồ trực quan thông qua dịch vụ của Google map.

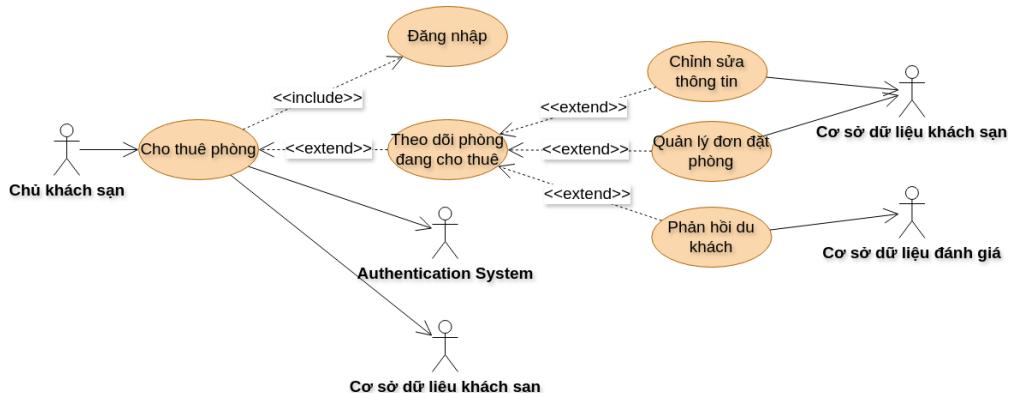
4.6 Ca sử dụng Tra cứu thông tin phương tiện di chuyển



Hình 4.6.1: Biểu đồ ca sử dụng Tra cứu thông tin phương tiện

Ca sử dụng **Tra cứu phương tiện** được người dùng sử dụng khi muốn tìm kiếm thông tin về phương tiện di chuyển. Người dùng cần phải đăng nhập để có thể sử dụng các tính năng bổ sung như đặt phương tiện và theo dõi phương tiện đã đặt.

4.7 Ca sử dụng Cho thuê phòng

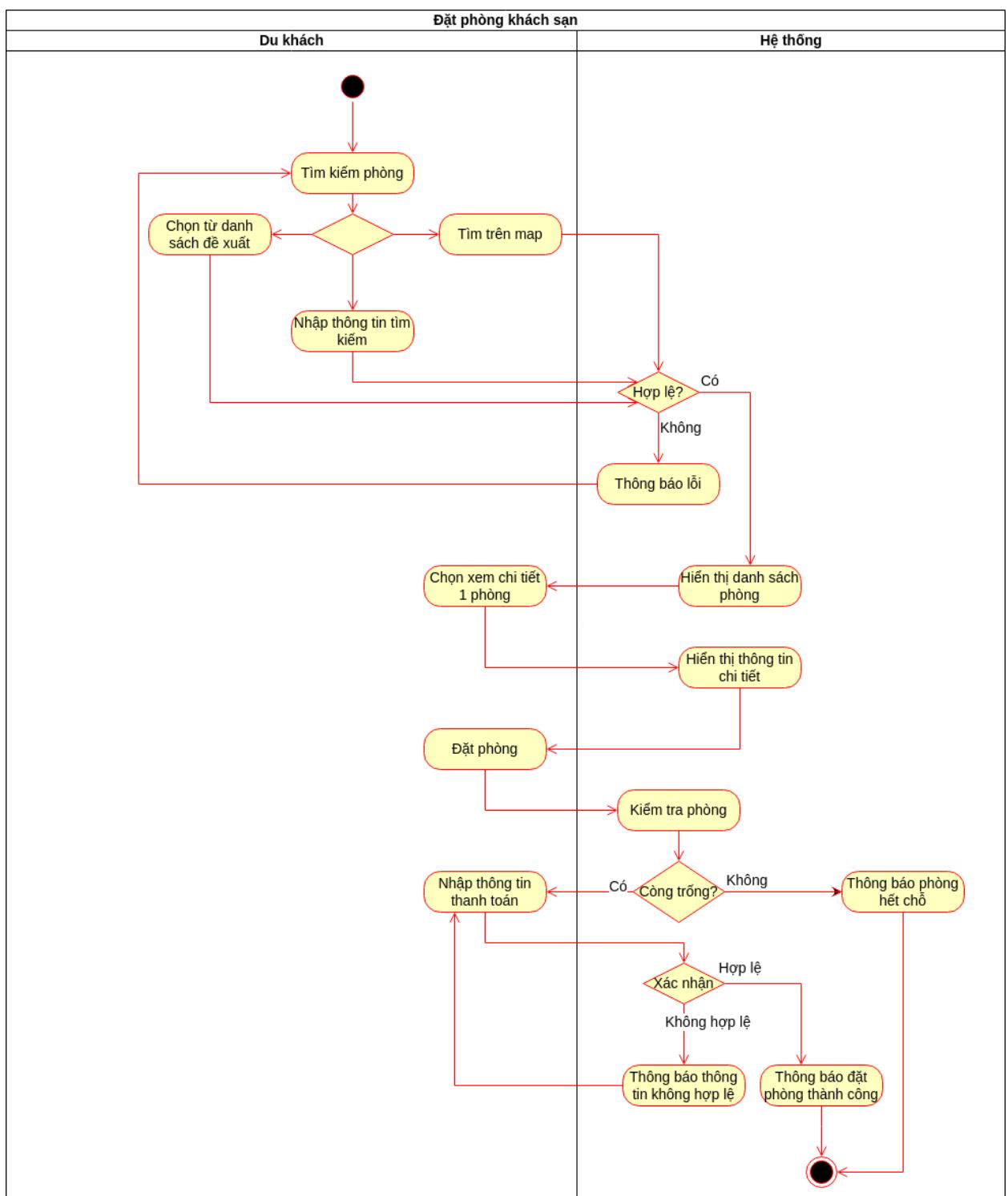


Hình 4.7.1: Biểu đồ ca sử dụng Cho thuê phòng

Ca sử dụng **Cho thuê phòng** được người dùng chủ khách sạn đã xác thực tài khoản sử dụng để cung cấp dịch vụ phòng khách sạn cho du khách và quản lý khách sạn đang cho thuê.

5 Đặc tả Use Case

5.1 Đặt phòng khách sạn



Hình 5.1.1: Biểu đồ hoạt động ca sử dụng Đặt phòng khách sạn

Mô tả

Chức năng này cho phép người dùng tìm kiếm phòng dựa trên các tiêu chí như vị trí, giá cả, tiện nghi, đánh giá... Người dùng có thể xem chi tiết phòng và tiến hành đặt phòng.

Các tác nhân

- **Du khách:** Tìm kiếm và đặt phòng.
- **Cơ sở dữ liệu khách sạn:** Lưu trữ và cung cấp dữ liệu về các phòng khách sạn.

Điều kiện kích hoạt ca sử dụng

- Người dùng đã đăng nhập vào hệ thống.

Tiền điều kiện

- Hệ thống đã có dữ liệu phòng.
- Kết nối mạng ổn định.

Hậu điều kiện

- Nếu đặt phòng thành công, thông tin đặt phòng được lưu lại.
- Nếu chỉ tìm kiếm, hệ thống có thể lưu lại lịch sử tìm kiếm để gợi ý sau này.

Các luồng sự kiện

Luồng cơ bản

1. Người dùng chọn tìm kiếm phòng theo các phương thức như theo thông tin, tìm kiếm trên map, chọn từ danh sách đề xuất.
2. Hệ thống kiểm tra tính hợp lệ của thông tin khách sạn.
3. Hệ thống hiển thị danh sách phòng theo yêu cầu của du khách.
4. Du khách chọn một phòng để xem chi tiết.
5. Hệ thống hiển thị thông tin chi tiết của phòng đã chọn.
6. Người dùng chọn đặt phòng.
7. Hệ thống kiểm tra xem phòng còn trống không.
8. Người dùng nhập thông tin tài khoản và xác nhận đặt phòng.
9. Hệ thống xác nhận thông tin thanh toán.
10. Hệ thống hiển thị thông báo thành công và kết thúc ca sử dụng.

Luồng thay thế

- 2.1 Nếu thông tin không hợp lệ, hệ thống thông báo và quay lại bước 1.
- 7.1 Nếu phòng đã hết chỗ thì hệ thống hiện thông báo và kết thúc ca sử dụng.
- 9.1 Nếu thông tin thanh toán không hợp lệ, hệ thống hiển thị thông báo và quay lại bước 8.

Business rules

- Chỉ hiển thị phòng khả dụng trong khoảng thời gian người dùng yêu cầu.

- Nếu người dùng hủy đặt phòng, áp dụng chính sách hoàn tiền của từng phòng.
- Có thể có giá ưu đãi cho khách hàng thân thiết hoặc đặt phòng sớm.

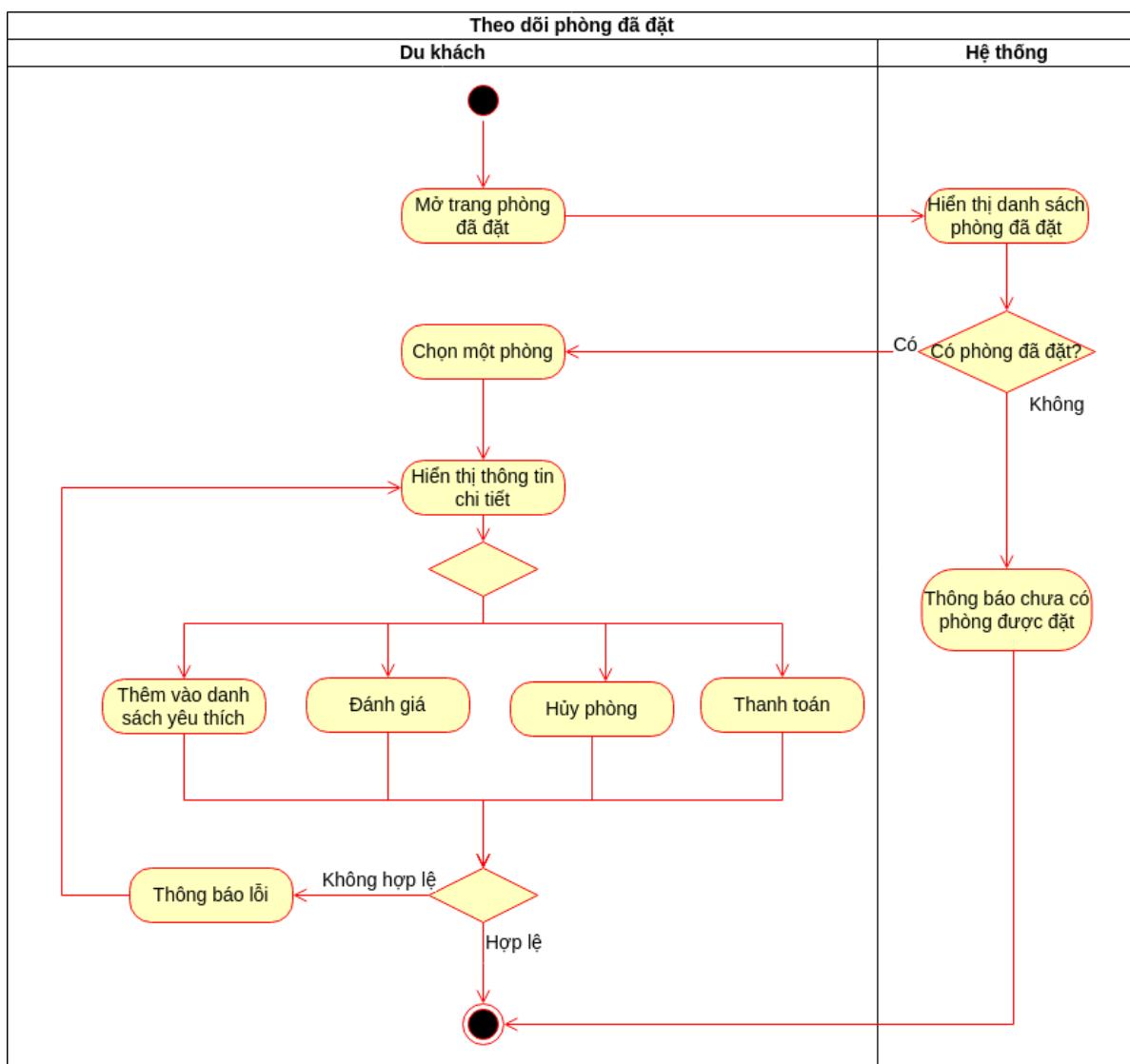
Yêu cầu phi chức năng

- Tốc độ tìm kiếm nhanh, tối ưu thời gian phản hồi dưới 3 giây.
- Giao diện thân thiện, hiển thị rõ ràng thông tin quan trọng.

Extension point

- Hệ thống có thể tích hợp AI để đề xuất phòng phù hợp dựa trên lịch sử tìm kiếm.
- Hỗ trợ bản đồ tương tác để tìm kiếm phòng theo vị trí trực quan.

5.2 Theo dõi phòng đã đặt



Hình 5.2.1: Biểu đồ hoạt động ca sử dụng Theo dõi phòng đã đặt

Mô tả

Chức năng này cho phép người dùng xem danh sách phòng đã thuê, kiểm tra thông tin đặt phòng và hủy đặt phòng nếu cần.

Các tác nhân

- **Du khách:** Xem danh sách và chi tiết phòng đã thuê.

Điều kiện kích hoạt ca sử dụng

- Người dùng đã đăng nhập vào hệ thống.

Tiền điều kiện

- Người dùng đã có ít nhất một phòng được đặt trước đó.

Hậu điều kiện

- Nếu người dùng chỉ xem, không có thay đổi nào trong dữ liệu đặt phòng.
- Nếu người dùng hủy phòng, hệ thống cập nhật trạng thái đặt phòng và hoàn tiền (nếu chính sách cho phép).

Các luồng sự kiện

Luồng cơ bản

1. Người dùng mở trang “Phòng đã đặt”.
2. Hệ thống hiển thị danh sách các phòng đã đặt.
3. Người dùng chọn một phòng để xem chi tiết.
4. Hệ thống hiển thị thông tin đặt phòng.
5. Người dùng có thể thao tác hủy phòng, thanh toán, đánh giá và thêm vào danh sách yêu thích.
6. Hệ thống xử lý yêu cầu và hiển thị kết quả.

Luồng thay thế

- 2.1 Nếu người dùng chưa đặt phòng nào, hệ thống hiển thị thông báo “Bạn chưa có phòng nào được đặt”.
 - 6.1 Nếu yêu cầu không hợp lệ hoặc không phù hợp với chính sách thì thông báo lỗi và quay lại bước 4.

Business rules

- Chỉ cho phép hủy phòng trước thời gian check-in theo chính sách của từng phòng.
- Hiển thị đầy đủ thông tin về lịch sử đặt phòng, bao gồm cả những phòng đã hoàn thành.

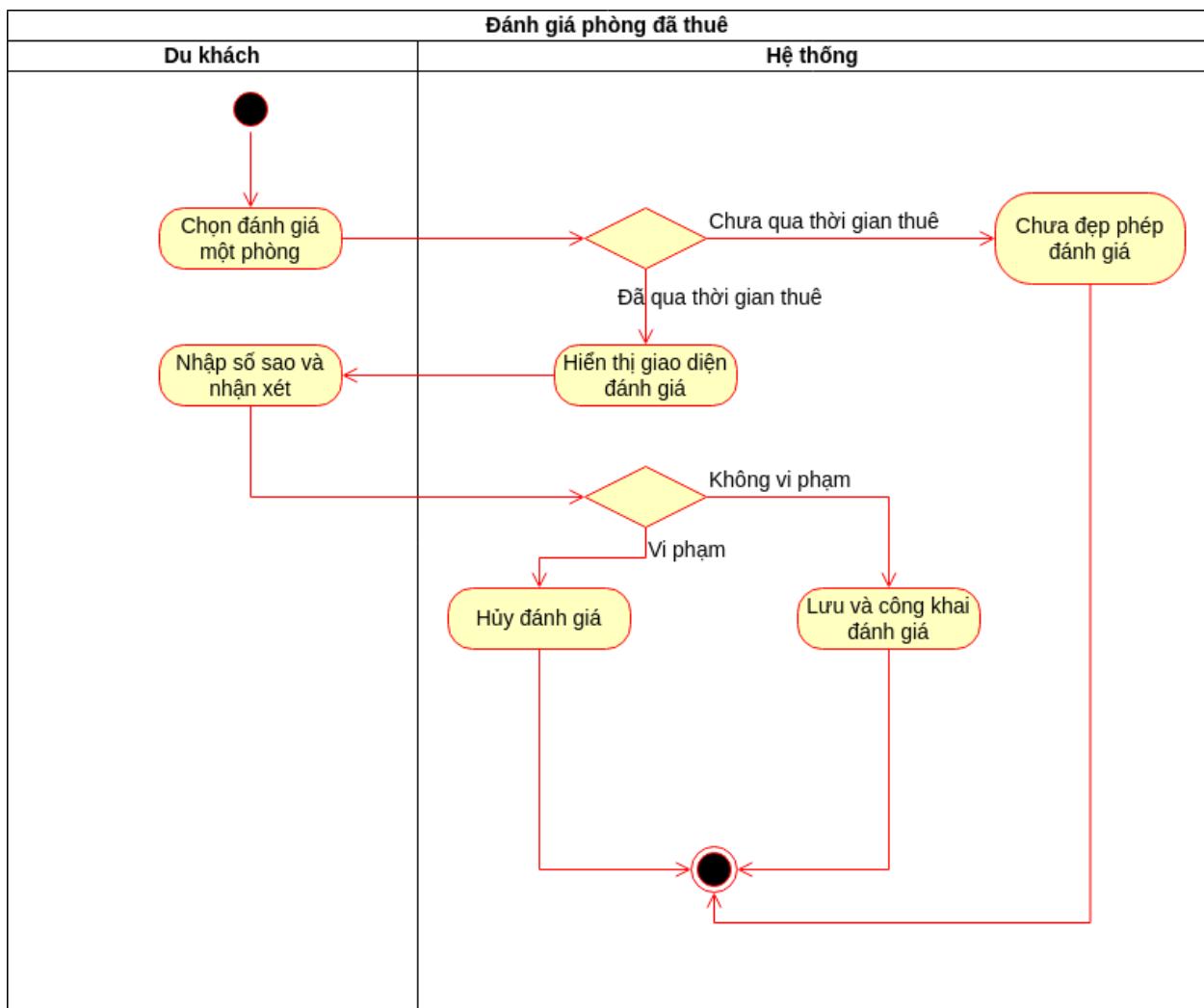
Yêu cầu phi chức năng

- Giao diện hiển thị rõ ràng, dễ dàng điều hướng giữa các phòng đã đặt.

Extension point

- Hệ thống có thể hiển thị lịch trình check-in/check-out sắp tới.
- Hỗ trợ xuất hóa đơn điện tử cho khách hàng.

5.3 Đánh giá phòng đã thuê



Hình 5.3.1: Biểu đồ hoạt động cho ca sử dụng Đánh giá phòng đã thuê

Mô tả

Sau khi thuê phòng, người dùng có thể đánh giá bằng số sao và viết nhận xét về trải nghiệm của mình.

Các tác nhân

- **Người dùng:** Đánh giá phòng đã thuê.

Điều kiện kích hoạt ca sử dụng

- Người dùng đã hoàn thành ít nhất một lần thuê phòng.

Tiền điều kiện

- Phòng đã qua thời gian thuê (không cho phép đánh giá trước khi trải nghiệm).

Hậu điều kiện

- Đánh giá được lưu và hiển thị trên trang phòng.

Các luồng sự kiện

Luồng cơ bản

1. Người dùng chọn đánh giá phòng đã thuê.
2. Hệ thống kiểm tra phòng có đủ điều kiện để đánh giá chưa.
3. Người dùng nhập số sao và nhận xét.
4. Hệ thống kiểm tra đánh giá.
5. Hệ thống lưu đánh giá và hiển thị công khai.

Luồng thay thế

- 2.1 Nếu phòng chưa qua thời gian thuê, hệ thống không cho phép đánh giá.
- 4.1 Nếu đánh giá chứa nội dung vi phạm, hệ thống từ chối lưu.

Business rules

- Mỗi phòng chỉ có thể được đánh giá sau khi thời gian thuê kết thúc.
- Không cho phép chỉnh sửa hoặc xóa đánh giá sau khi đã gửi.

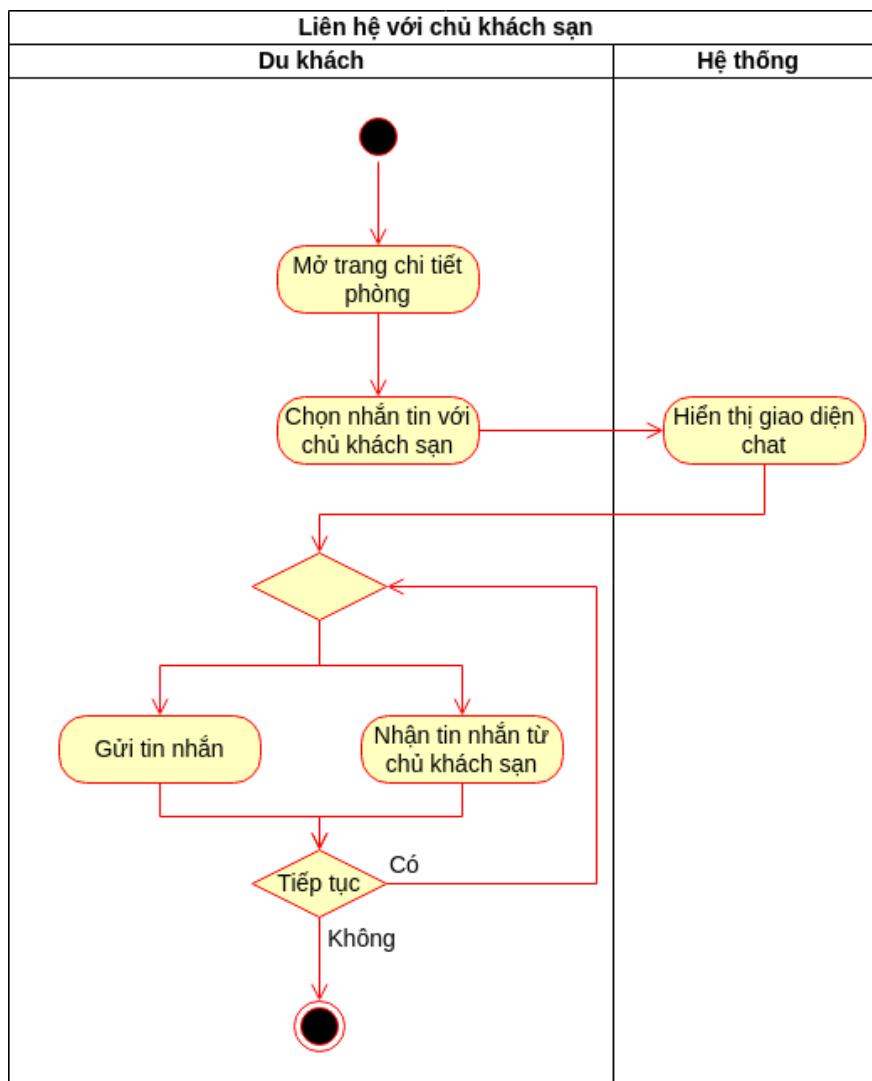
Yêu cầu phi chức năng

- Đánh giá hiển thị theo thời gian thực, phản hồi nhanh.

Extension point

- Hệ thống có thể đề xuất các phòng tương tự dựa trên đánh giá của người dùng.

5.4 Liên hệ với chủ khách sạn



Hình 5.4.1: Biểu đồ hoạt động cho ca sử dụng Liên hệ với chủ khách sạn

Mô tả

Người dùng có thể chat trực tiếp với chủ phòng để đặt câu hỏi về phòng hoặc yêu cầu hỗ trợ trong quá trình thuê.

Các tác nhân

- **Người dùng (Khách hàng):** Gửi tin nhắn.
- **Chủ khách sạn:** Trả lời tin nhắn.

Điều kiện kích hoạt ca sử dụng

- Người dùng đã đăng nhập.
- Người dùng đang xem một phòng cụ thể hoặc đã thuê phòng đó.

Tiền điều kiện

- Chủ phòng có bật tính năng nhắn tin.

Hậu điều kiện

- Tin nhắn được lưu lại để người dùng có thể xem lại sau.

Các luồng sự kiện

Luồng cơ bản

- Người dùng mở trang chi tiết phòng.
- Người dùng nhấn vào nút "Nhắn tin với chủ phòng".
- Hệ thống mở giao diện chat.
- Người dùng gửi và nhận tin nhắn với chủ khách sạn.
- Người dùng chọn kết thúc trò chuyện.

Business rules

- Hệ thống chỉ cho phép nhắn tin giữa khách hàng và chủ phòng khi có liên quan đến một phòng cụ thể.
- Không cho phép gửi nội dung vi phạm chính sách nền tảng.

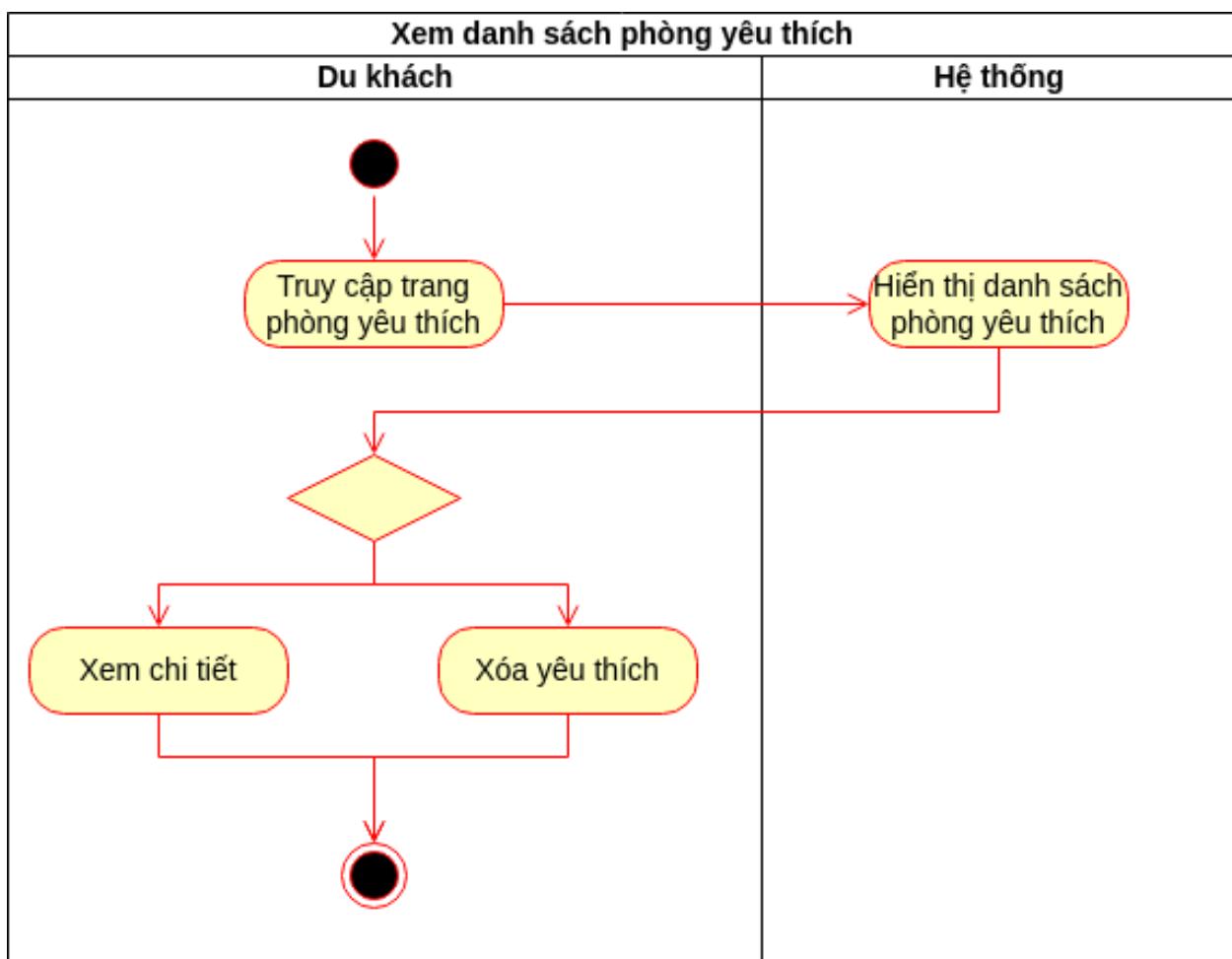
Yêu cầu phi chức năng

- Tin nhắn phải hiển thị ngay lập tức (real-time).

Extension point

- Hệ thống có thể tích hợp chatbot AI để hỗ trợ tự động trả lời các câu hỏi phổ biến.

5.5 Xem danh sách phòng yêu thích



Hình 5.5.1: Biểu đồ hoạt động cho ca sử dụng Xem danh sách phòng yêu thích

Mô tả

Chức năng này cho phép người dùng lưu các phòng yêu thích để xem lại sau mà không cần tìm kiếm lại. Đồng thời, người dùng có thể truy cập danh sách này để quản lý các phòng đã lưu.

Các tác nhân

- **Người dùng (Khách hàng):** Lưu, xem và xóa phòng yêu thích.

Điều kiện kích hoạt ca sử dụng

- Người dùng đã đăng nhập vào hệ thống.

Tiền điều kiện

- Hệ thống có dữ liệu phòng hợp lệ.

Hậu điều kiện

- Nếu người dùng lưu phòng, phòng được thêm vào danh sách yêu thích.
- Nếu người dùng xóa phòng khỏi danh sách, hệ thống cập nhật lại danh sách.

Các luồng sự kiện

Luồng cơ bản

1. Người dùng truy cập trang phòng yêu thích
2. Hiển thị danh sách phòng yêu thích
3. Người dùng có thể xem chi tiết phòng hoặc xóa phòng khỏi danh sách yêu thích.

Business rules

- Người dùng có thể lưu nhiều phòng mà không bị giới hạn.
- Nếu một phòng bị chủ phòng xóa khỏi hệ thống, nó cũng bị xóa khỏi danh sách yêu thích của người dùng.

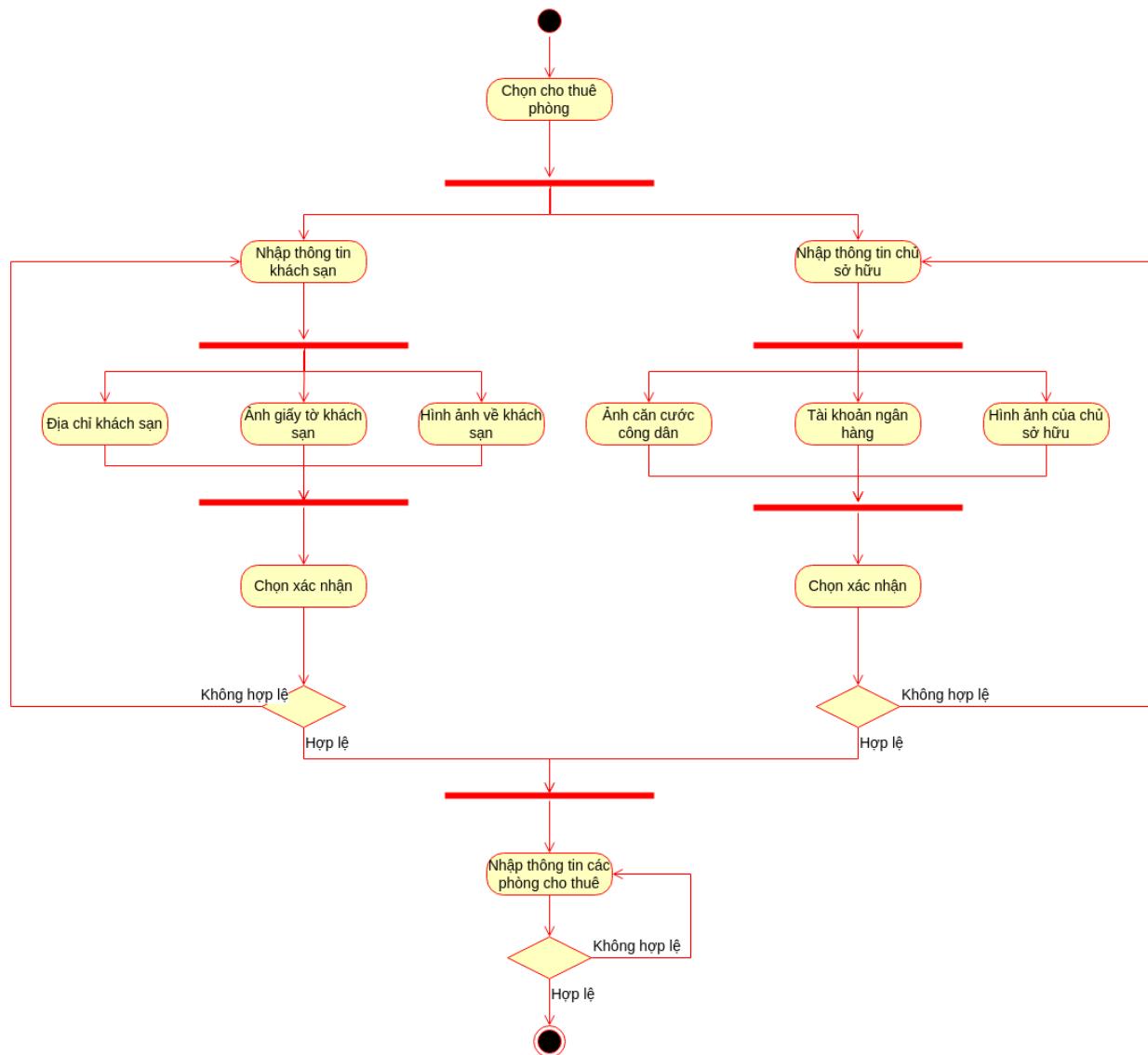
Yêu cầu phi chức năng

- Danh sách yêu thích phải tải nhanh, không làm gián đoạn trải nghiệm người dùng.

Extension point

- Hệ thống có thể gợi ý các phòng tương tự dựa trên danh sách yêu thích của người dùng.

5.6 Cho thuê phòng



Hình 5.6.1: Biểu đồ hoạt động cho ca sử dụng Cho thuê phòng

Mô tả

Người dùng (chủ khách sạn hoặc người quản lý) thực hiện việc đăng ký khách sạn và phòng cho thuê trên hệ thống, hệ thống sẽ kiểm tra tính hợp lệ của thông tin trước khi xác nhận việc cho thuê phòng.

Các tác nhân

Người dùng, authentication system.

Điều kiện kích hoạt ca sử dụng

- Người dùng muốn đăng ký khách sạn và cho thuê phòng.

Tiền điều kiện

- Người dùng phải có tài khoản hợp lệ để truy cập hệ thống.
- Hệ thống phải hoạt động bình thường.
- Người dùng phải có internet trong suốt thời gian hoạt động.

Hậu điều kiện

- Nếu thông tin hợp lệ, phòng được đăng ký thành công trong hệ thống.
- Nếu thông tin không hợp lệ, hệ thống sẽ từ chối và yêu cầu chỉnh sửa.

Các luồng sự kiện

Luồng cơ bản

1. Người dùng chọn cho thuê phòng
2. Người dùng nhập thông tin của cá nhân và khách sạn
3. Người dùng chọn xác nhận
4. Hệ thống kiểm tra các thông tin người dùng đã nhập
5. Người dùng nhập thông tin các phòng cho thuê
6. Hệ thống kiểm tra thông tin phòng.
7. Hiển thị thông báo thành công và kết thúc

Luồng thay thế

- 4.1 Thông báo thông tin không hợp lệ và quay lại bước 2
- 6.1 Hiển thị thông tin phòng không hợp lệ và quay lại bước 5

Business rules

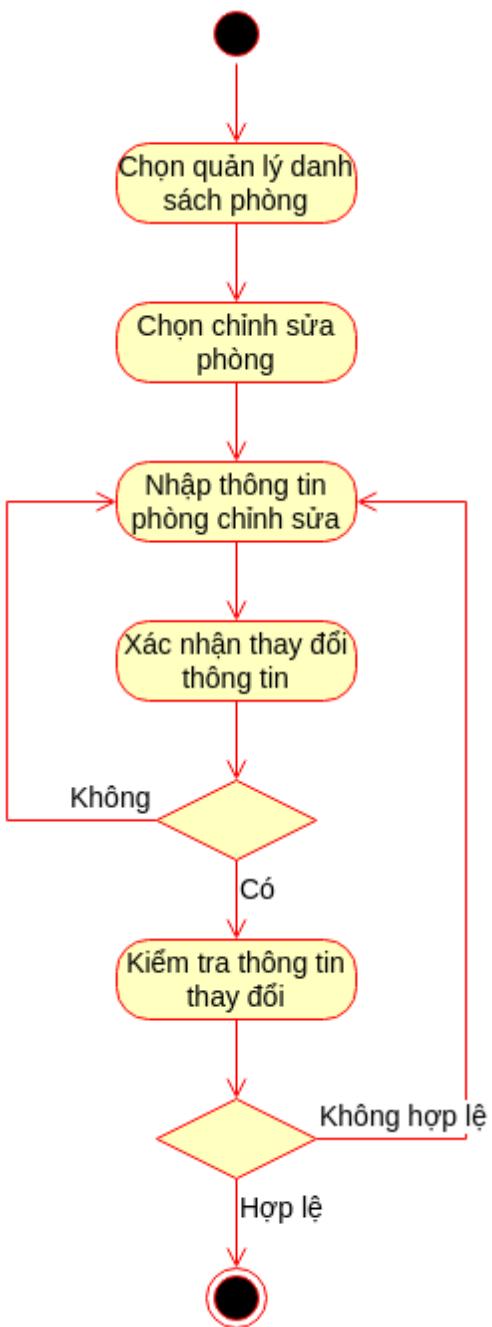
- Thông tin khách hàng và chủ sở hữu phải đầy đủ chính xác.
- Ảnh CCCD và giấy tờ phải rõ ràng, không bị mờ.
- Tài khoản ngân hàng phải khớp với tên chủ sở hữu để tránh gian lận.
- Chỉ khi mọi thông tin hợp lệ thì phòng mới được đưa lên hệ thống.

Yêu cầu phi chức năng

- Bảo mật: bảo vệ thông tin cá nhân của chủ khách sạn.
- Tích hợp: hệ thống có thể kết nối với ngân hàng để xác thực tài khoản.

Extension point: Không có

5.7 Sửa thông tin phòng



Hình 5.7.1: Biểu đồ hoạt động cho ca sử dụng Sửa thông tin phòng

Mô tả

Hành động này cho phép người quản lý cập nhật thông tin của một phòng đã cho thuê trên ứng dụng.

Các tác nhân

- **Người quản lý phòng:** tác nhân chính.

Điều kiện kích hoạt ca sử dụng

- Người quản lý đã đăng nhập vào ứng dụng.
- Phòng cần cập nhật đã tồn tại trong hệ thống.

Tiền điều kiện

- Phòng không ở trạng thái đang cho thuê.
- Tài khoản có phòng đã được đăng lên hệ thống.

Hậu điều kiện

- Thông tin phòng được cập nhập nếu người dùng nhập đúng và hoàn tất kiểm tra.
- Thông tin phòng giữ nguyên nếu người dùng không xác nhận hoặc nhập thông tin không chính xác.

Các luồng sự kiện

Luồng cơ bản

1. Người dùng chọn danh sách phòng đã cho thuê.
2. Chọn chỉnh sửa phòng muốn thay đổi.
3. Nhập những thông tin cần chỉnh sửa.
4. Người dùng xác nhận lưu thay đổi.
5. Hệ thống kiểm tra xác thực dữ liệu đầu vào.

Luồng thay thế

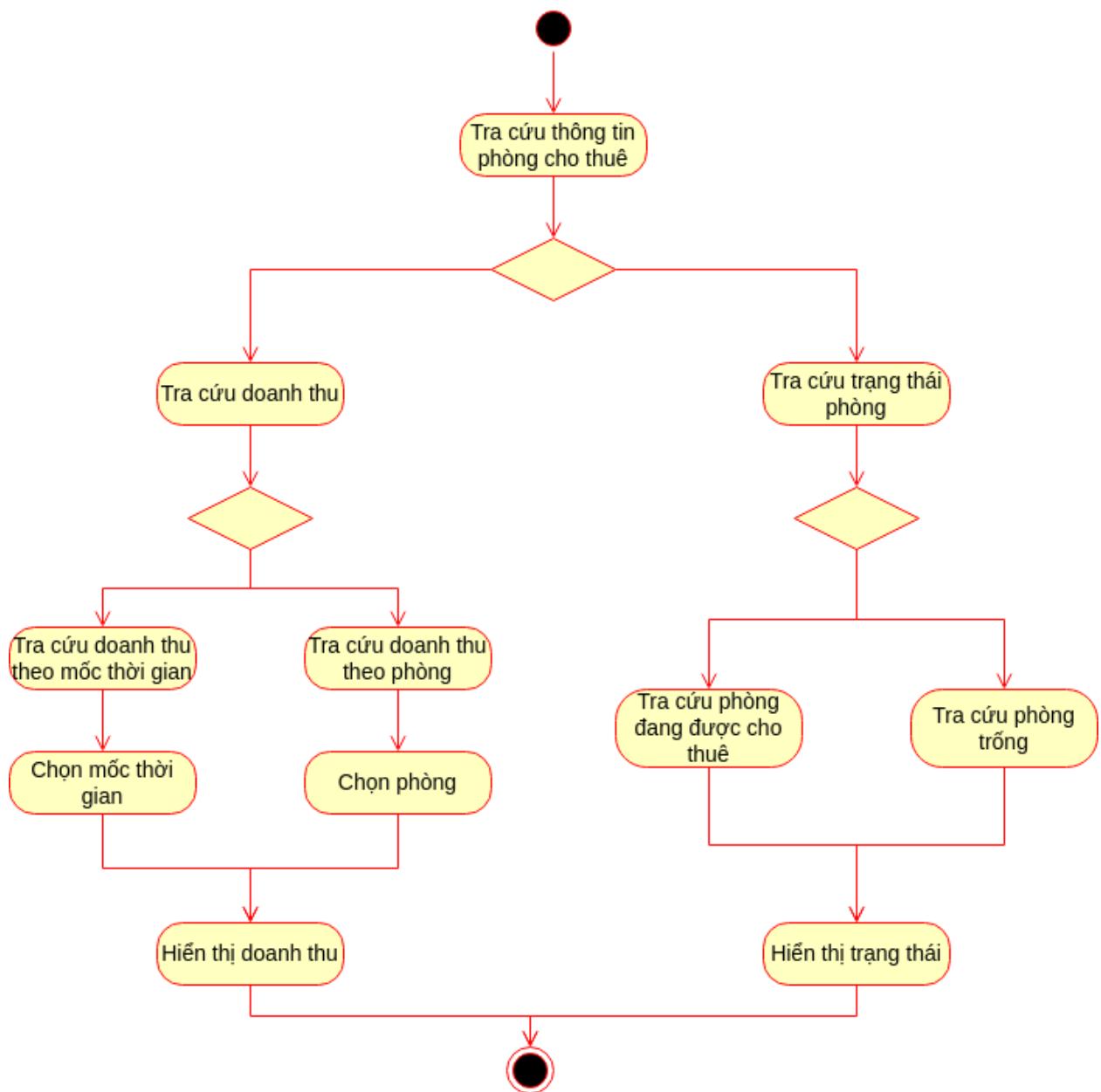
- 4.1 Nếu người quản lý nhấn "Hủy", hệ thống sẽ không lưu bất kỳ thay đổi nào.
- 5.1 Hiển thị thông báo thông tin phòng không hợp lệ, yêu cầu người dùng nhập lại thông tin phòng.

Business rules

- Thông tin phòng phải đầy đủ chính xác hình ảnh phải rõ nét.
- Chỉ khi mọi thông tin hợp lệ thì phòng mới được đưa lên hệ thống.

Extension point: Không có

5.8 Tra cứu thông tin phòng đang cho thuê



Hình 5.8.1: Biểu đồ hoạt động cho ca sử dụng Tra cứu thông tin phòng đang cho thuê

Mô tả

- Doanh thu (theo khoảng thời gian hoặc theo phòng).
- Trạng thái các phòng (phòng đang được thuê hoặc phòng trống).

Các tác nhân

- **Người quản lý phòng:** tác nhân chính.

Điều kiện kích hoạt ca sử dụng

- Người dùng muốn xem thông tin về doanh thu hoặc tình trạng (trống/đang thuê) của phòng.

Tiền điều kiện

- Người dùng đã đăng nhập hệ thống và có phòng cho thuê.
- Dữ liệu doanh thu và thông tin trạng thái phòng đã được hệ thống cập nhật.

Hậu điều kiện

- Người dùng có được thông tin chính xác theo kết quả được tìm kiếm.
- Hệ thống hiển thị thông tin theo lựa chọn của người dùng.

Các luồng sự kiện

Luồng cơ bản

1. Người dùng tra cứu thông tin phòng cho thuê
2. Người dùng chọn Tra cứu doanh thu hoặc Tra cứu trạng thái phòng
3. Người dùng chọn thông tin cần tra cứu
4. Hệ thống hiển thị thông tin

Luồng thay thế

- 3.1 Hệ thống không tìm kiếm được thời gian đã chọn, hiển thị thông báo không có thông tin về thời gian đã chọn.
- 3.2 Hệ thống không tìm kiếm được phòng đã chọn, hiển thị thông báo không có thông tin về phòng đã chọn.
- 3.3 Hệ thống không thấy phòng còn trống, hiển thị thông báo không có phòng nào còn trống.
- 3.4 Hệ thống không thấy phòng nào được thuê, hiển thị thông báo không có phòng nào được thuê.

Business rules

- Khoảng thời gian tra cứu phải hợp lệ (ví dụ: ngày bắt đầu trước ngày kết thúc).
- Thông tin về trạng thái phòng phải được cập nhật thường xuyên để đảm bảo tính chính xác.
- Người dùng chỉ xem được doanh thu nếu có quyền hạn phù hợp.

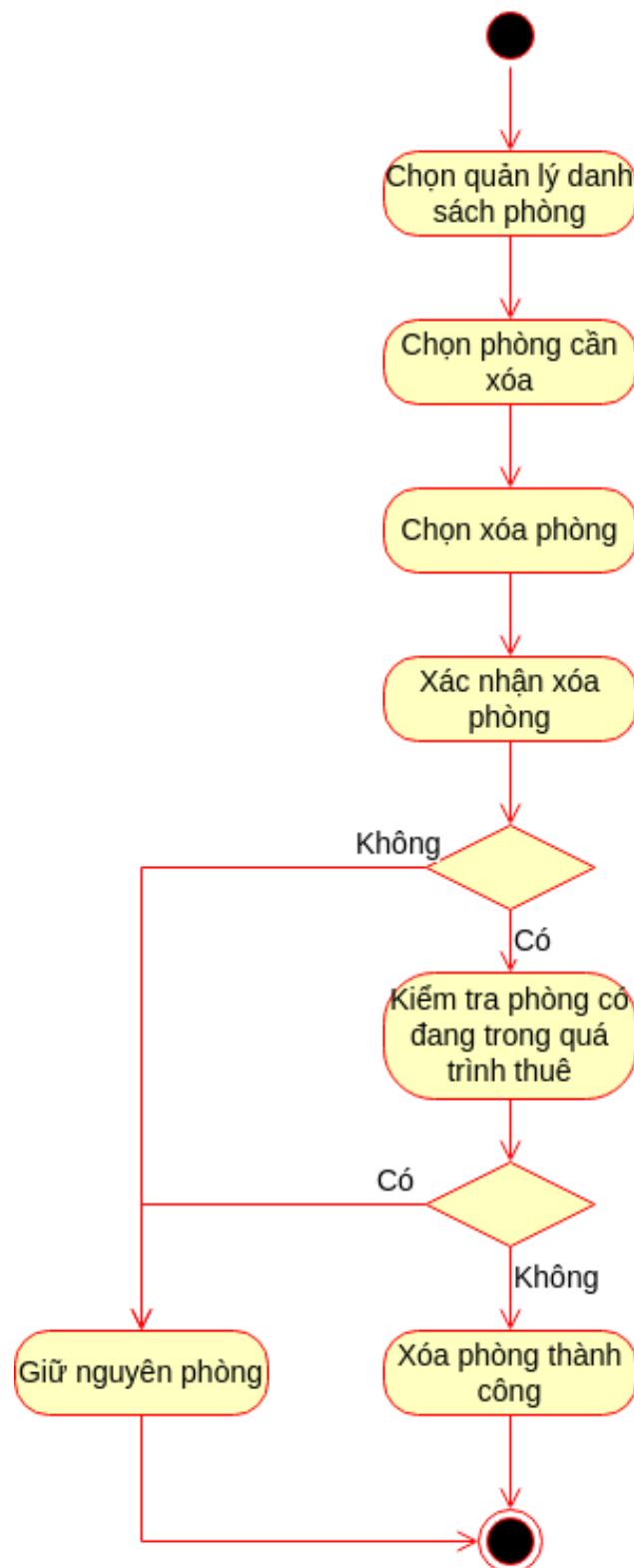
Yêu cầu phi chức năng

- Hiệu suất: Hệ thống phải trả kết quả tra cứu doanh thu và trạng thái phòng nhanh
- Bảo mật: Chỉ những tài khoản có quyền xem doanh thu mới được phép xem báo cáo.

Extension point

- Lọc nâng cao: Lọc trạng thái phòng theo nhiều tiêu chí (loại phòng, giá, vị trí...).

5.9 Xóa phòng khách sạn



Hình 5.9.1: Biểu đồ hoạt động cho ca sử dụng Xóa phòng khách sạn

Mô tả

Hành động cho phép người quản lý phòng cho thuê xóa phòng của mình trên hệ thống.

Các tác nhân

- **Người quản lý phòng:** tác nhân chính.

Điều kiện kích hoạt ca sử dụng

- Người quản lý đã đăng nhập vào ứng dụng.
- Phòng cần cập nhật đã tồn tại trong hệ thống.

Tiền điều kiện

- Tài khoản có phòng đã được đăng lên hệ thống.

Hậu điều kiện

- Phòng sẽ được xóa nếu hoàn tất kiểm tra.
- Phòng sẽ không thay đổi nếu người dùng xác nhận hoặc phòng đang trong quá trình thuê.

Các luồng sự kiện

Luồng cơ bản

1. Người dùng chọn danh sách phòng đã cho thuê.
2. Chọn phòng muốn xóa.
3. Chọn nút xóa phòng.
4. Người dùng xác nhận xóa phòng.
5. Hệ thống kiểm tra phòng có đang trong quá trình thuê không.
6. Hệ thống thông báo xóa phòng thành công.

Luồng thay thế

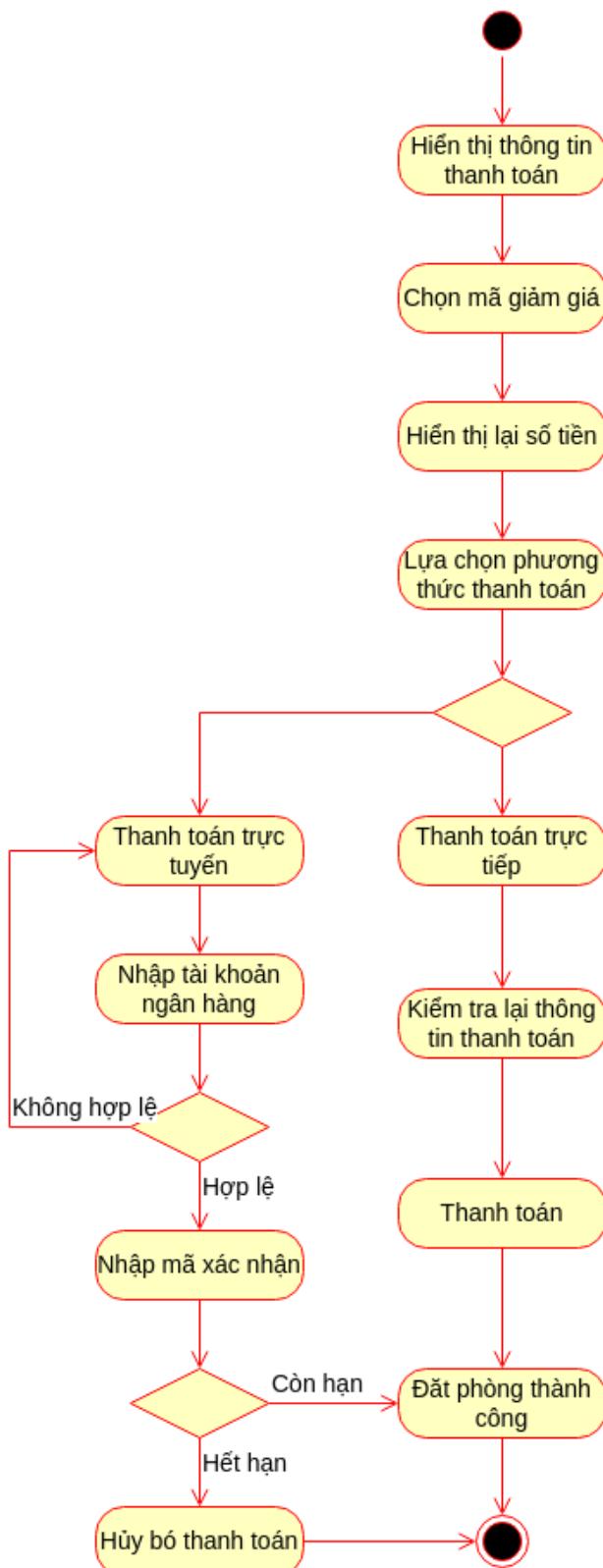
- 4.1 Nếu người quản lý nhấn "Hủy", hệ thống sẽ không lưu bất kỳ thay đổi nào.
- 5.1 Hiển thị thông báo thông phòng đang trong quá trình thuê, yêu cầu người dùng xóa phòng khi không có khách thuê.

Business rules: Không có.

Yêu cầu phi chức năng: Không có.

Extension point: Không có.

5.10 Thanh toán



Hình 5.10.1: Biểu đồ hoạt động cho ca sử dụng Thanh toán

Mô tả

Người dùng thanh toán khoản tiền cần trả cho việc đặt phòng và tìm phương tiện (nếu có) bằng các hình thức khác nhau.

Các tác nhân: Người dùng, Ngân hàng.

Điều kiện kích hoạt ca sử dụng

- Người dùng chọn chức năng thanh toán sau khi đặt phòng (hoặc đặt phương tiện).

Tiền điều kiện

- Người dùng đã đăng nhập vào hệ thống.
- Thiết bị của người dùng kết nối Internet trong khi sử dụng dịch vụ.
- Người dùng đã đặt phòng (hoặc đặt phương tiện) và chọn mục thanh toán.

Hậu điều kiện

- Thanh toán thành công: Hệ thống cập nhật lại trạng thái phòng và thông báo cho người dùng.
- Thanh toán thất bại: Hệ thống hiển thị thông báo lỗi.

Các luồng sự kiện

Luồng cơ bản

- Người dùng đặt phòng khách sạn (đặt phương tiện) và sau đó chọn mục thanh toán.
- Hệ thống hiển thị thông tin dịch vụ và số tiền cần phải thanh toán.
- Hệ thống hiển thị các phương thức thanh toán khả dụng.
- Người dùng chọn 1 phương thức thanh toán phù hợp.
- Người dùng nhập thông tin thanh toán.
- Hệ thống xác nhận thông tin.
- Hệ thống thông báo thanh toán thành công

Luồng thay thế

- Hiển thị thông báo lỗi và quay lại bước 5

Business rules

- Hệ thống chỉ xác nhận đặt phòng thành công khi thanh toán được chấp nhận hoặc đặt cọc thành công.
- Nếu thanh toán thất bại, đặt phòng sẽ không được giữ cho đến khi người dùng hoàn tất giao dịch.
- Hệ thống phải tự động gửi email/SMS xác nhận sau khi thanh toán thành công.
- Người dùng có thể hủy phòng và nhận lại toàn bộ số tiền đã thanh toán.
- Hệ thống không lưu trữ trực tiếp thông tin thẻ tín dụng của khách hàng chỉ lưu mã token hóa do cổng thanh toán cung cấp.
- Hệ thống hỗ trợ 2FA (xác thực hai yếu tố).

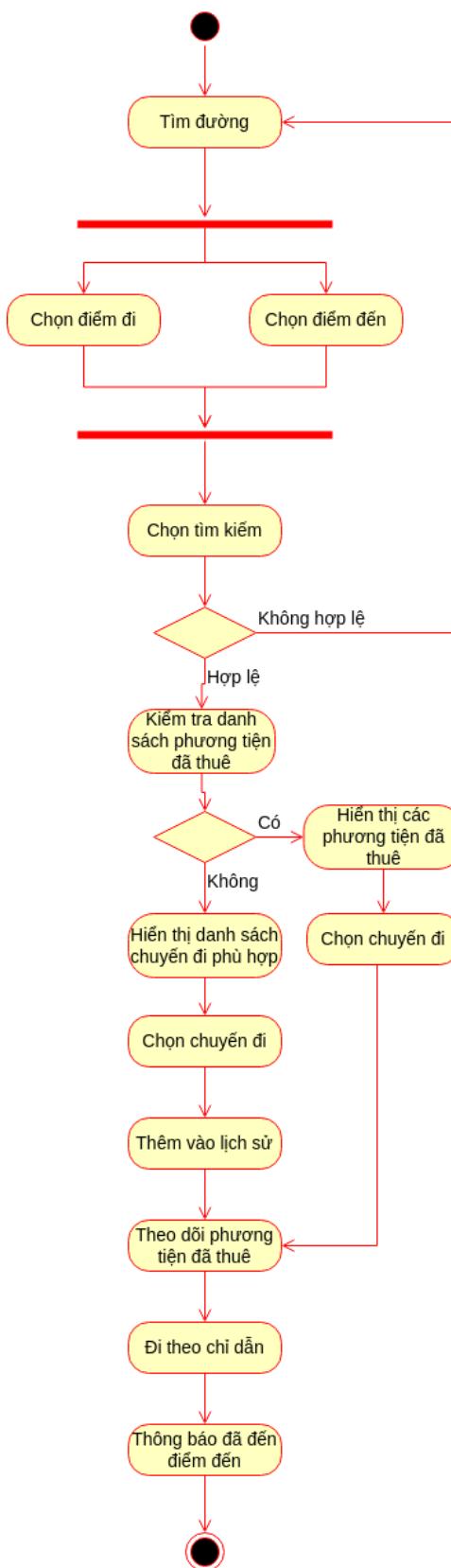
- Nếu hệ thống báo thanh toán thành công nhưng đặt phòng không được xác nhận, hệ thống phải hoàn tiền trong vòng 3-5 ngày làm việc.
- Nếu giao dịch bị lỗi do hệ thống nhưng tiền đã bị trừ, hệ thống sẽ tự động xử lý hoàn tiền hoặc liên hệ hỗ trợ.

Yêu cầu phi chức năng

- Hệ thống xử lý giao dịch trong vòng 5 giây để đảm bảo trải nghiệm mượt mà.
- Hỗ trợ nhiều giao dịch thanh toán đồng thời (hơn 1000) mà không bị gián đoạn.
- Hỗ trợ tất cả các trình duyệt phổ biến (Chrome, Safari, Firefox, ...)
- Tương thích với Android, IOS.

Extension point: Mã giảm giá: Người dùng chọn mã giảm giá tại bước thanh toán, nếu mã hợp lệ thì hiển thị lại số tiền cần thanh toán, và nếu mã k hợp lệ thì hệ thống hiển thị lỗi và yêu cầu nhập lại

5.11 Tra cứu phương tiện



Hình 5.11.1: Biểu đồ hoạt động cho ca sử dụng Tra cứu phương tiện

Mô tả

Người dùng sử dụng chức năng tìm phương tiện để được chỉ dẫn về các phương thức di chuyển từ vị trí hiện tại (hoặc một địa điểm cụ thể) đến khách sạn đã đặt (hoặc một địa điểm cụ thể). Hệ thống sẽ hiển thị các tùy chọn phương tiện phù hợp.

Các tác nhân: Người dùng, Google map.

Điều kiện kích hoạt ca sử dụng

- Người dùng muốn tìm phương tiện để di chuyển đến khách sạn đã đặt.

Tiền điều kiện

- Người dùng đã đặt phòng khách sạn thành công.
- Hệ thống hoạt động ổn định.
- Đảm bảo thiết bị có kết nối Internet.

Hậu điều kiện

- Hệ thống hiển thị các phương tiện di chuyển từ điểm xuất phát đến khách sạn.
- Người dùng chọn phương tiện phù hợp và hệ thống sẽ hiển thị chi phí cần thanh toán.
- Hệ thống cập nhật trạng thái dịch vụ.

Các luồng sự kiện

Luồng cơ bản

1. Ở giao diện chính của ứng dụng, sau khi đặt phòng thành công thì người dùng chọn mục Tìm phương tiện.
2. Người dùng chọn điểm xuất phát và điểm đến.
3. Sau đó chọn vào mục Tìm phương tiện và hệ thống sẽ hiển thị những phương tiện phù hợp cho người dùng dựa vào chi phí tốt nhất và thời gian di chuyển ngắn nhất.
4. Người dùng chọn phương tiện phù hợp với mình trong danh sách hiển thị, hệ thống hiển thị chi tiết thông tin chi tiết lộ trình và bản đồ theo dõi vị trí so với đường đi được chỉ dẫn.
5. Người dùng thanh toán chi phí và sử dụng dịch vụ.

Luồng thay thế

- 1.1 Người dùng mất kết nối Internet trong quá trình tìm kiếm phương tiện. Hệ thống sẽ hiển thị “Kết nối Internet bị gián đoạn. Vui lòng kiểm tra lại mạng và thử lại”. Hệ thống lưu trữ thông tin điểm đi/đến của người dùng để tiếp tục tìm kiếm ngay khi có lại kết nối.
- 2.1 Người dùng nhập điểm xuất phát không hợp lệ. Hệ thống sẽ hiển thị “Không tìm thấy”.
- 3.1 Nếu hai điểm đến và đi quá gần nhau trong vòng 200m, hệ thống sẽ thông báo: “Hai điểm bạn chọn quá gần để sử dụng phương tiện”. Hệ thống gợi ý tuyến đường đi bộ trên bản đồ. hoản không hợp lệ, Vui lòng kiểm tra lại”.

Business rules

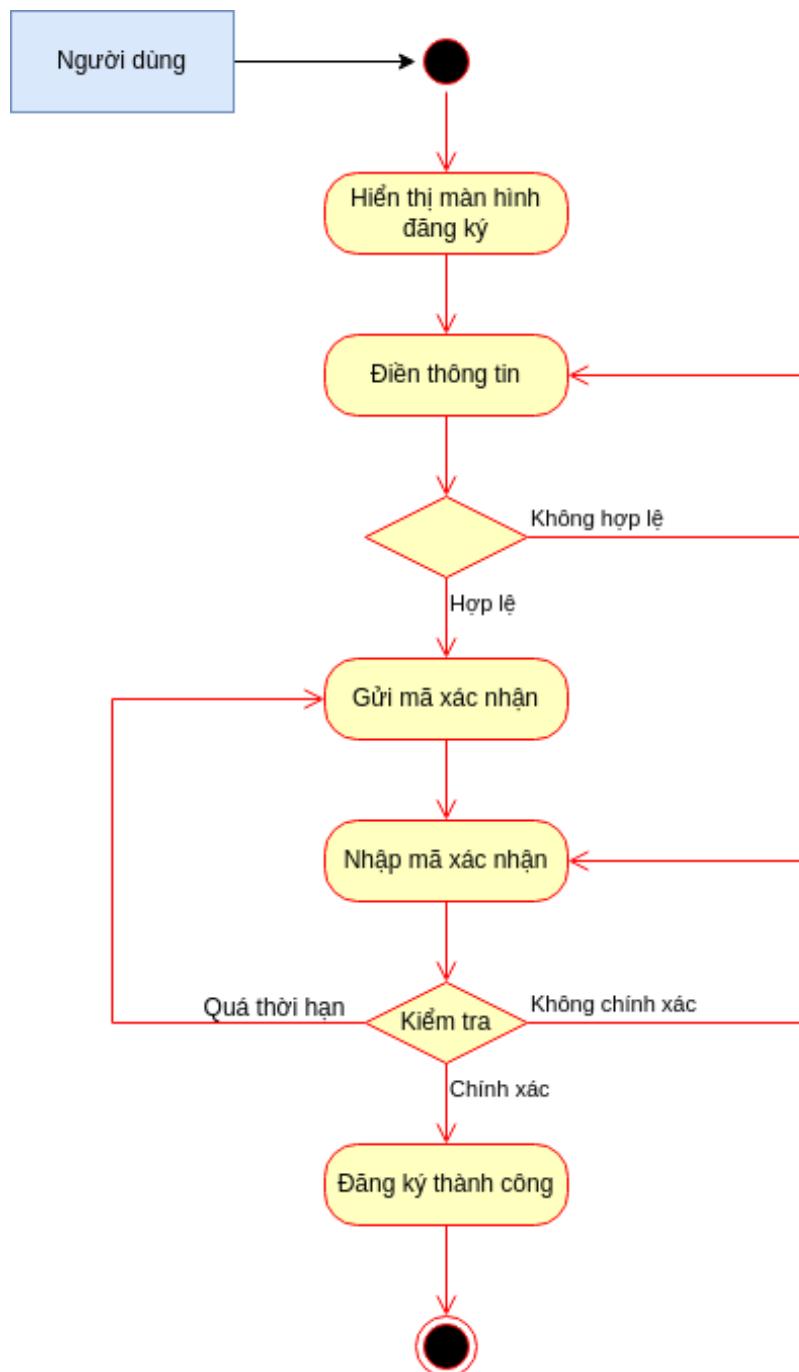
- Hệ thống chỉ tìm kiếm phương tiện trong phạm vi khu vực có dịch vụ giao thông hỗ trợ.
- Nếu điểm đi hoặc điểm đến nằm ngoài khu vực hỗ trợ, hệ thống sẽ thông báo lỗi.
- Người dùng có thể lọc để sắp xếp lại thứ tự ưu tiên khi hiển thị kết quả.
- Hệ thống cung cấp thông tin về các loại phương tiện phù hợp

Yêu cầu phi chức năng

- Hệ thống phải trả về kết quả tìm kiếm phương tiện trong vòng dưới 10 giây trong điều kiện mạng ổn định.
- Tốc độ hiển thị bản đồ và chỉ dẫn đường đi phải mượt mà, không bị giật lag.
- Hệ thống có thể xử lý đồng thời nhiều lượt tìm kiếm mà không ảnh hưởng đến hiệu suất.
- Dữ liệu vị trí của người dùng phải được mã hóa trước khi gửi lên server.
- Không chia sẻ dữ liệu vị trí cá nhân cho bên thứ ba nếu không có sự cho phép của người dùng.

Extension point: Không có.

5.12 Đăng ký tài khoản



Hình 5.12.1: Biểu đồ hoạt động cho ca sử dụng Đăng ký tài khoản

Mô tả

Người dùng khai báo thông tin để tạo tài khoản và truy cập đầy đủ các chức năng của hệ thống.

Các tác nhân: Người dùng.

Điều kiện kích hoạt ca sử dụng

- Người dùng chọn chức năng đăng ký tài khoản.

Tiền điều kiện

- Người dùng muốn tạo tài khoản.
- Người dùng có đầy đủ thông tin mà hệ thống yêu cầu.
- Thiết bị của người dùng được kết nối internet trong suốt ca sử dụng.

Hậu điều kiện

- Người dùng đăng ký tài khoản thành công

Các luồng sự kiện

Luồng cơ bản

1. Người dùng truy cập vào ứng dụng.
2. Người dùng chọn chức năng đăng ký tài khoản.
3. Hệ thống cung cấp form điền thông tin đăng ký tài khoản.
4. Người dùng điền thông tin đầy đủ vào các trường dữ liệu trong form đăng ký.
5. Hệ thống xác nhận tính hợp lệ của thông tin mà người dùng điền vào form.
6. Hệ thống gửi đường dẫn mã xác nhận kích hoạt tài khoản bằng tin nhắn đến.
7. Người dùng nhập mã xác nhận kích hoạt tài khoản.
8. Hệ thống kiểm tra mã xác nhận.
9. Hệ thống hiển thị thông báo đăng ký thành công.
10. Kết thúc ca sử dụng.

Luồng thay thế

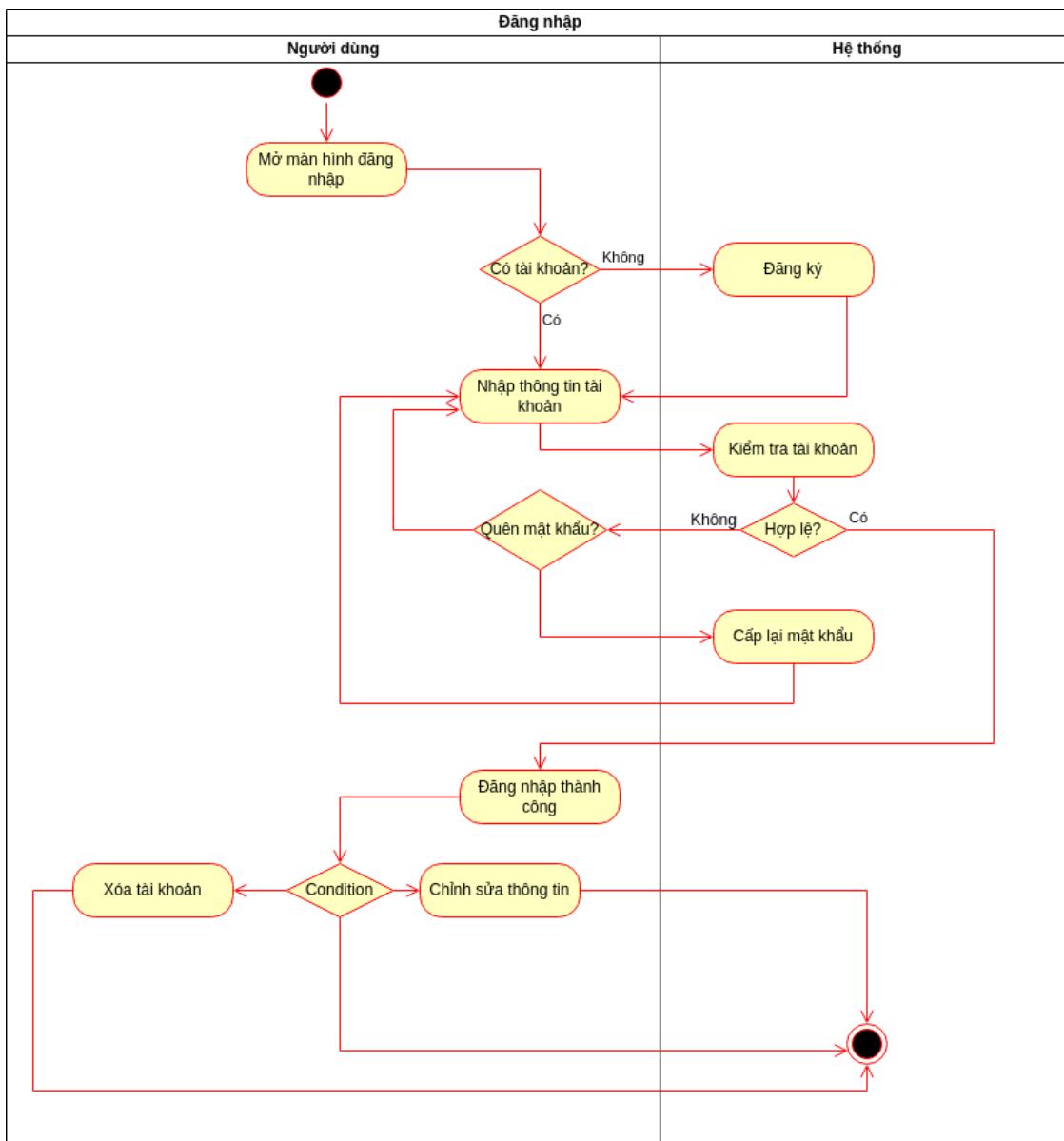
- 5.1 Hiển thị thông báo thông tin không hợp lệ và quay lại bước 4.
- 8.1 Hiển thị thông báo mã xác nhận không chính xác và quay lại bước 7.
- 8.2 Hiển thị thông báo mã xác nhận đã quá hạn và quay lại bước 6.

Business rules: Không có

Yêu cầu phi chức năng: Không có

Extension point: Không có

5.13 Đăng nhập



Hình 5.13.1: Biểu đồ hoạt động cho ca sử dụng Đăng nhập

Mô tả

Người dùng xác minh danh tính của mình để đăng nhập sử dụng hệ thống và chỉnh sửa tài khoản.

Các tác nhân: Người dùng, Cơ sở dữ liệu người dùng.

Điều kiện kích hoạt ca sử dụng

- Người dùng chọn chức năng đăng nhập tài khoản

Tiền điều kiện

- Hệ thống hiển thị giao diện đăng nhập ở trạng thái chờ

- Thiết bị của người dùng được kết nối internet trong suốt ca sử dụng

Hậu điều kiện: Không có

Các luồng sự kiện

Luồng cơ bản

- Người dùng truy cập vào ứng dụng.
- Người dùng chọn chức năng đăng nhập.
- Nếu người dùng có tài khoản, hệ thống cung cấp form điền thông tin đăng nhập tài khoản.
- Người dùng điền số điện thoại và mật khẩu vào form đăng nhập.
- Hệ thống xác minh thông tin của người dùng.
- Hệ thống hiển thị đăng nhập thành công.
- Kết thúc ca sử dụng

Luồng thay thế

- Người dùng không có tài khoản có thể chuyển sang chức năng đăng ký.
- Người dùng chọn quên mật khẩu.
- Hiển thị thông báo thông tin bị sai và yêu cầu người dùng nhập lại tài khoản, mật khẩu.
- Người dùng chọn chỉnh sửa thông tin sau đó kết thúc ca sử dụng.
- Người dùng chọn xóa tài khoản sau đó kết thúc ca sử dụng.

Business rules: Không có

Yêu cầu phi chức năng: Không có

Extension point:

Lấy lại mật khẩu:

- Người dùng chọn chức năng quên mật khẩu.
- Hệ thống yêu cầu người dùng nhập số điện thoại để xác minh tài khoản.
- Người dùng nhập số điện thoại vào trường dữ liệu hệ thống hiển thị.
- Hệ thống xác nhận tài khoản trùng khớp với số điện thoại và yêu cầu nhập mật khẩu mới.
Nếu tài khoản không tồn tại, hệ thống báo lỗi.
- Người dùng nhập mật khẩu mới và bấm tiếp tục.
- Hệ thống thông báo tạo mật khẩu mới thành công.

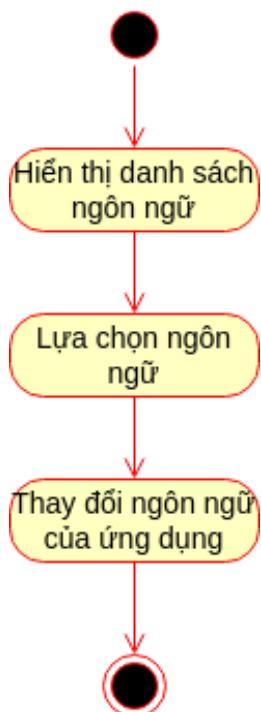
Chỉnh sửa thông tin hồ sơ

- Tại giao diện chính, người dùng chọn chức năng thay đổi thông tin.
- Hệ thống hiển thị form dữ liệu mà người dùng cung cấp khi đăng ký tài khoản.
- Người dùng thay đổi thông tin trên các trường dữ liệu cần đổi và bấm hoàn thành.
- Hệ thống hiển thị thông báo xác nhận thay đổi thông tin thành công.

Xóa tài khoản

- Tại giao diện chính, người dùng chọn chức năng xóa tài khoản.
- Hệ thống hiển thị thông báo xác nhận yêu cầu xóa tài khoản.
- Người dùng bấm chọn xác nhận lại yêu cầu.
- Hệ thống xóa dữ liệu tài khoản người dùng.

5.14 Lựa chọn ngôn ngữ



Hình 5.14.1: Biểu đồ hoạt động cho ca sử dụng Lựa chọn ngôn ngữ

Mô tả

Người dùng chọn ngôn ngữ phù hợp với bản thân.

Các tác nhân

Người dùng

Điều kiện kích hoạt ca sử dụng

- Người dùng lần đầu truy cập hệ thống.

Tiền điều kiện

- Người dùng truy cập hệ thống.

Hậu điều kiện

- Người dùng chọn ngôn ngữ thành công.
- Hệ thống lưu lại lựa chọn và thay đổi ngôn ngữ tương ứng.

Các luồng sự kiện

Luồng cơ bản

1. Hệ thống đưa lựa chọn ngôn ngữ.

2. Người dùng chọn ngôn ngữ tương ứng.
3. Hiển thị thông báo thay đổi thành công và kết thúc ca sử dụng.

Business rules: Không có.

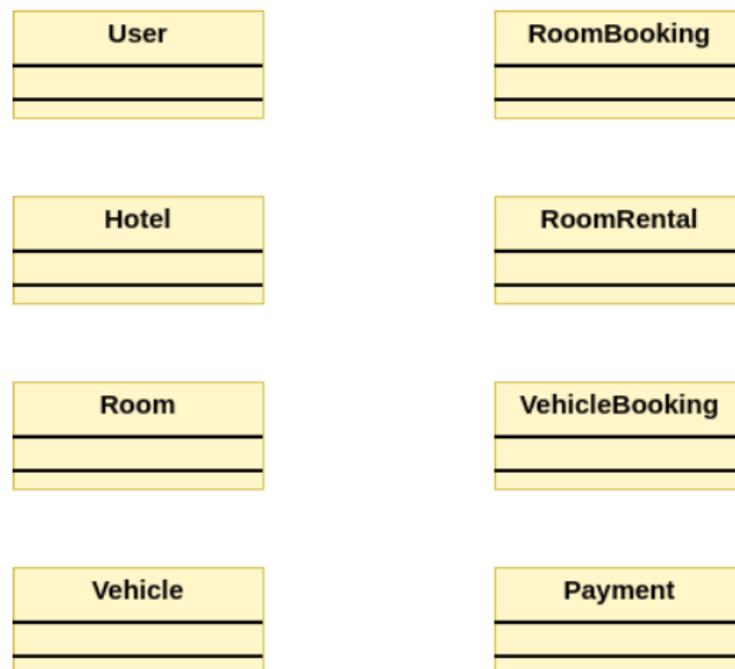
Yêu cầu phi chức năng: Không có

Extension point: Không có.

II Phân tích hệ thống

1 Phân tích kiến trúc

1.1 Key abstraction



Hình 1.1.1: Danh sách Key abstraction

User

- Định nghĩa: Là tài khoản của người dùng để thực hiện các chức năng như thuê phòng, thuê phương tiện và cho thuê phòng
- Cơ chế phân tích: Persistency, Security

Hotel

- Định nghĩa: là đối tượng chứa thông tin về các phòng và dịch vụ.
- Cơ chế phân tích: Persistency

Room

- Định nghĩa: là đối tượng đại diện cho các phòng có thể được thuê của một khách sạn.
- Cơ chế phân tích: Persistency

Vehicle

- Định nghĩa: là đối tượng đại diện cho các phương tiện có thể được thuê.
- Cơ chế phân tích: Persistency

RoomBooking

- Định nghĩa: là đối tượng quản lý việc đặt phòng của người dùng.
- Cơ chế phân tích: Persistency

RoomRental

- Định nghĩa: là đối tượng quản lý việc cho thuê phòng trọ của người dùng.
- Cơ chế phân tích: Persistency

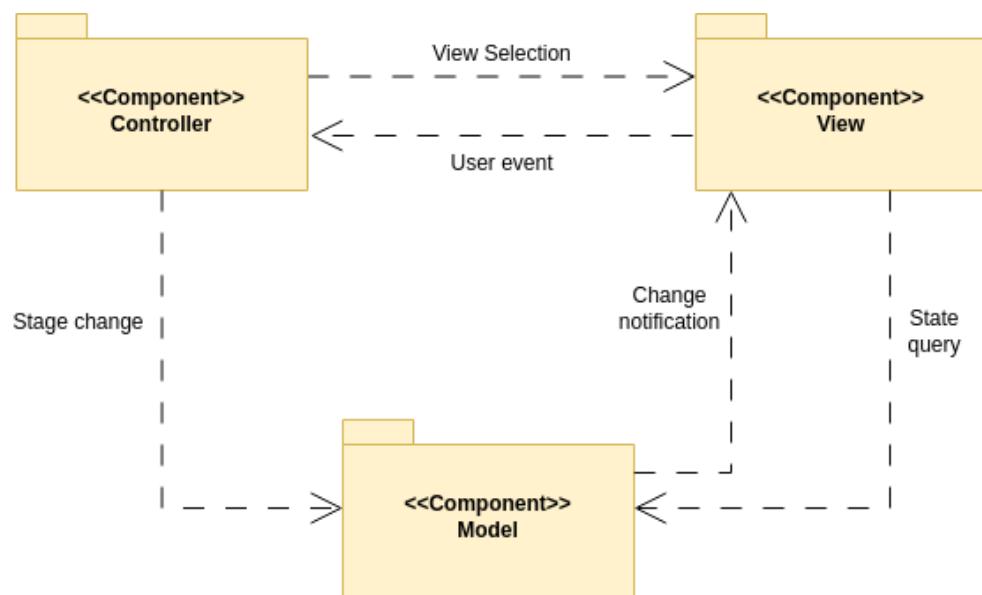
VehicleBooking

- Định nghĩa: là đối tượng quản lý việc đặt phương tiện của người dùng.
- Cơ chế phân tích: Persistency

Payment

- Định nghĩa: là đối tượng quản lý việc thanh toán của người dùng.
- Cơ chế phân tích: Persistency, Security

1.2 Thành phần cấp cao và sự phụ thuộc



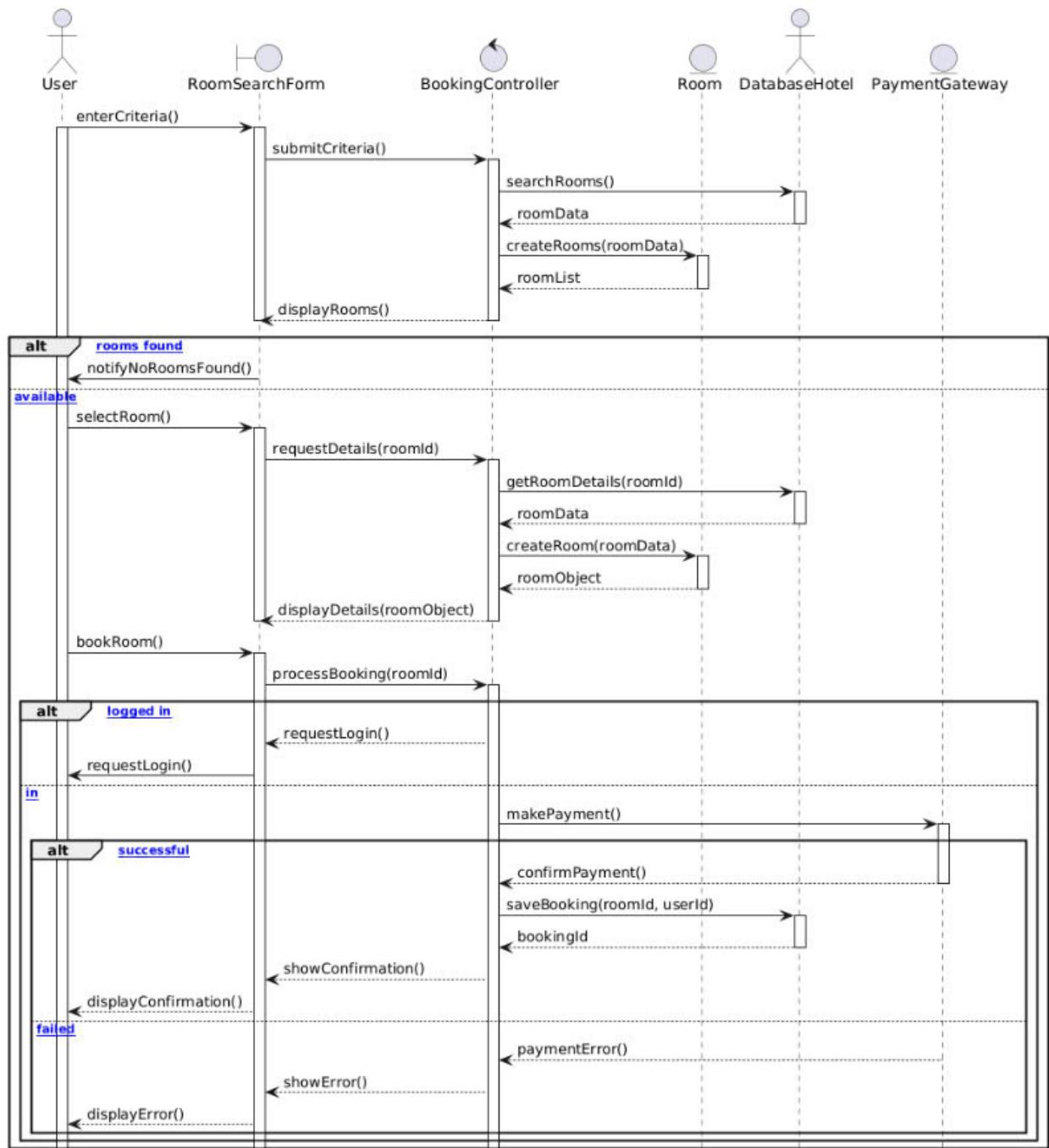
Hình 1.2.1: Mô hình MVC

- **Controller:** Điều phối luồng xử lý giữa View và Model
- **Model:** Tính toán nghiệp vụ, xử lý và kiểm tra dữ liệu
- **View:** hiển thị thông tin, tương tác trực tiếp với người dùng, chuyển tiếp các yêu cầu tới hệ thống và hiển thị kết quả đầu ra cho người dùng.

2 Phân tích Use case

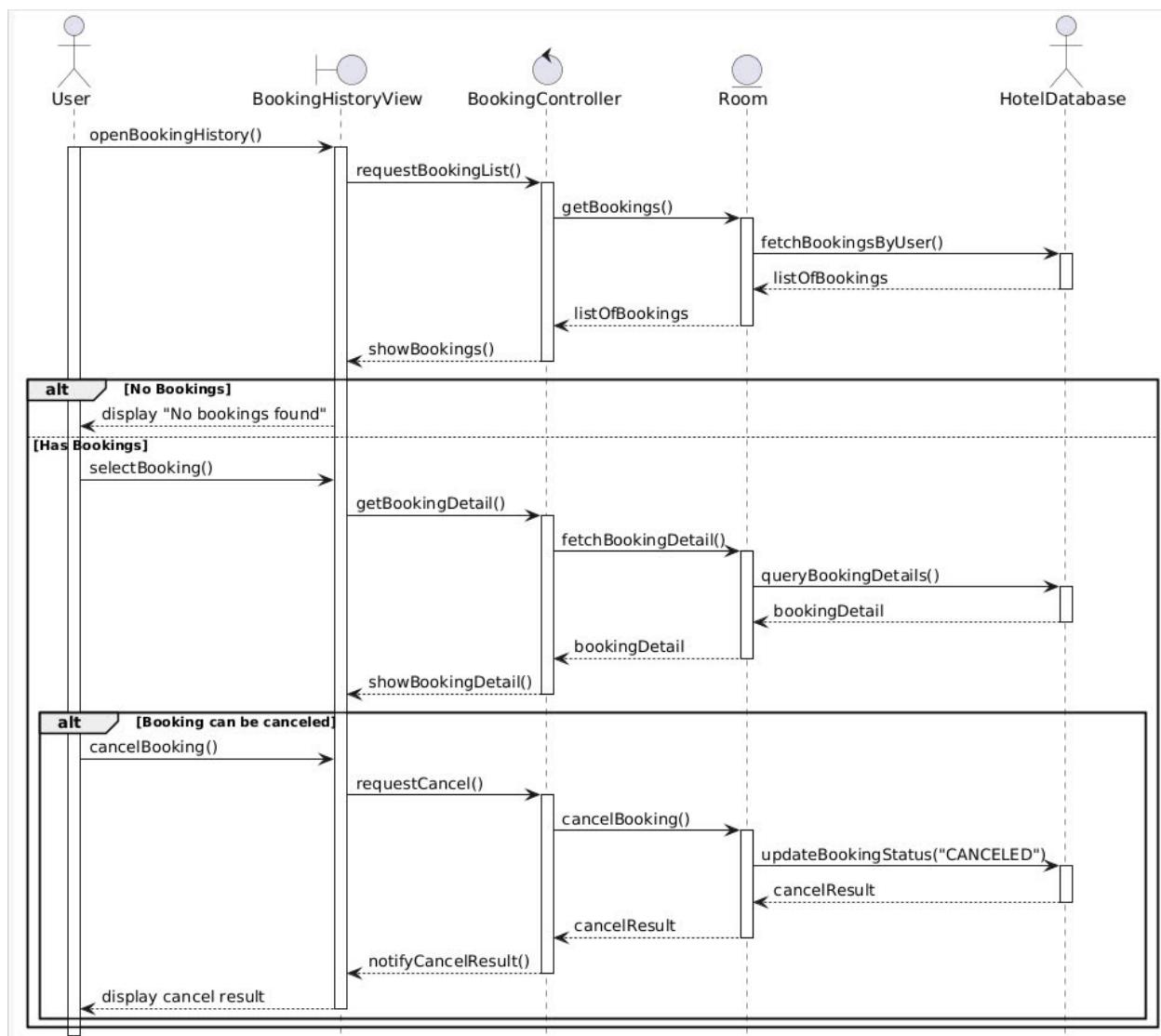
2.1 Biểu đồ tuần tự các use case

2.1.1 Đặt phòng khách sạn



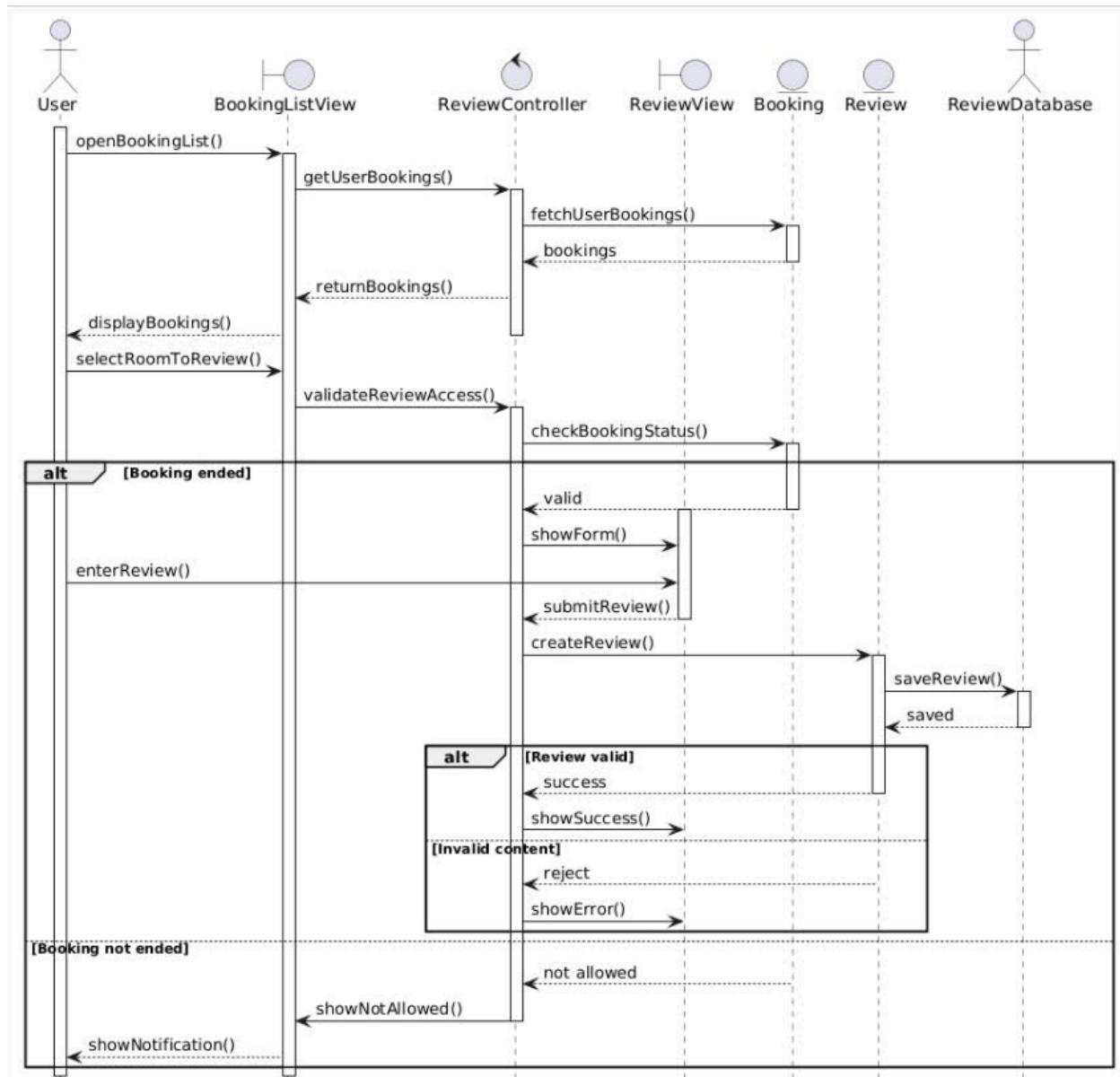
Hình 2.1.1: Biểu đồ tuần tự ca sử dụng Đặt phòng khách sạn

2.1.2 Theo dõi phòng đã đặt



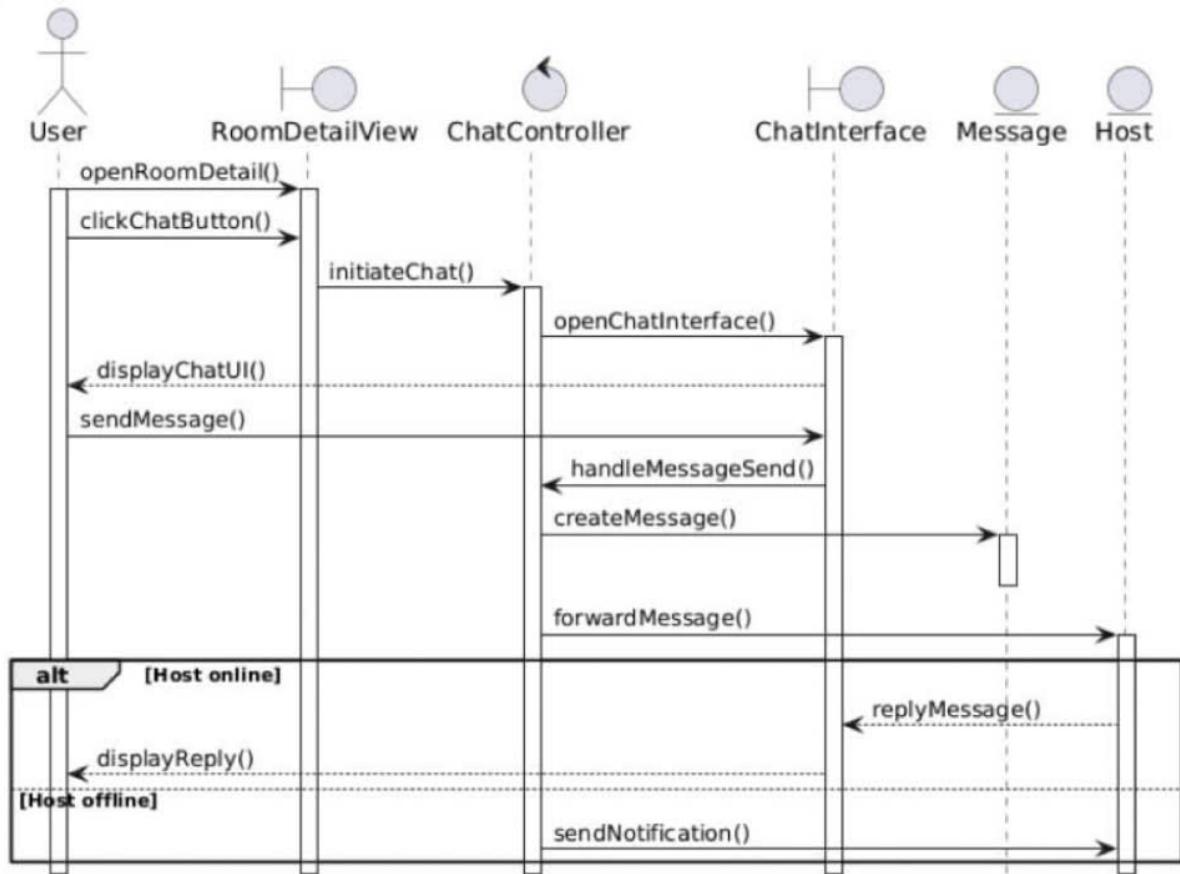
Hình 2.1.2: Biểu đồ tuần tự ca sử dụng Theo dõi phòng đã đặt

2.1.3 Đánh giá phòng đã thuê



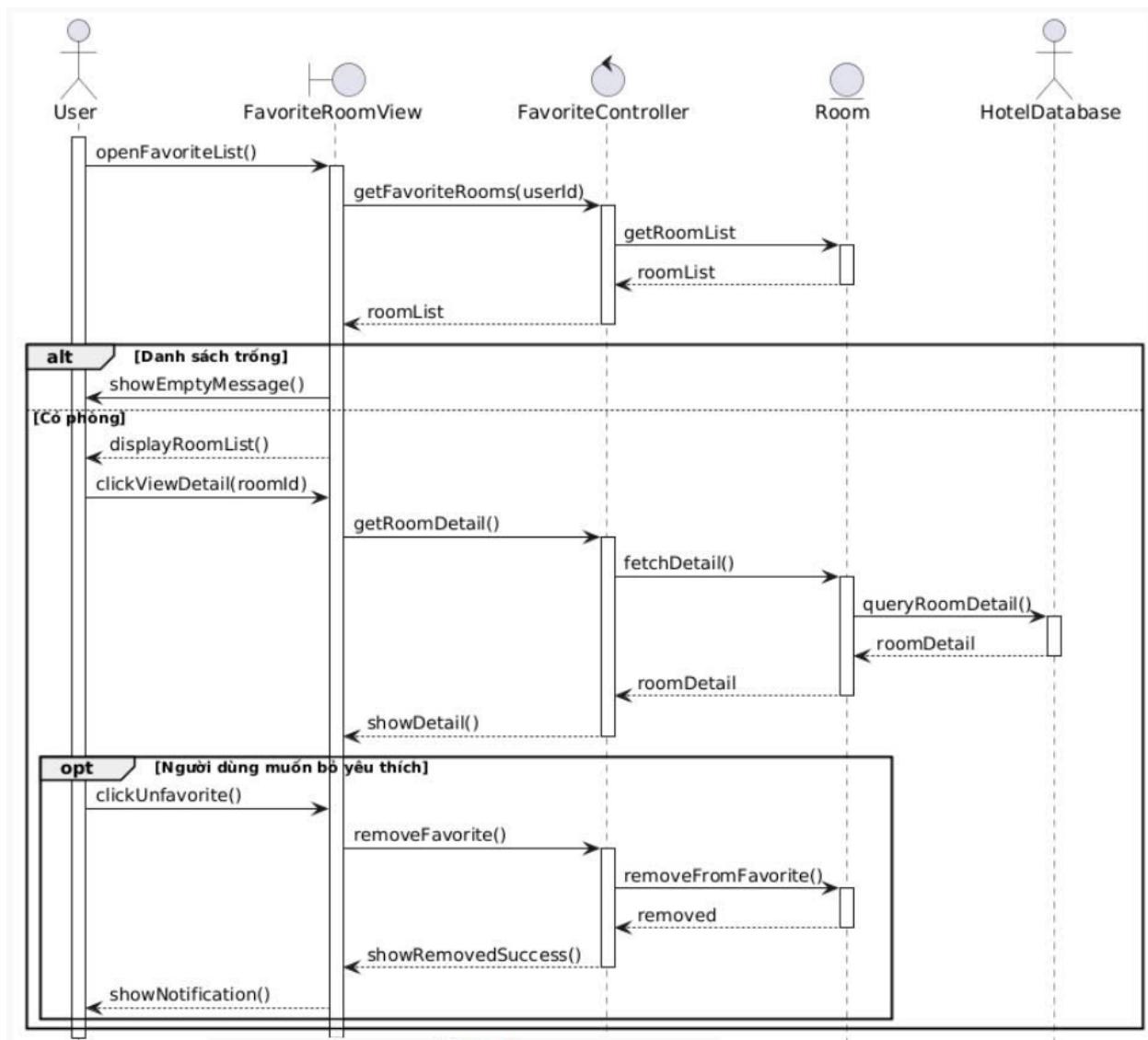
Hình 2.1.3: Biểu đồ tuần tự ca sử dụng Đánh giá phòng đã thuê

2.1.4 Liên hệ với chủ khách sạn



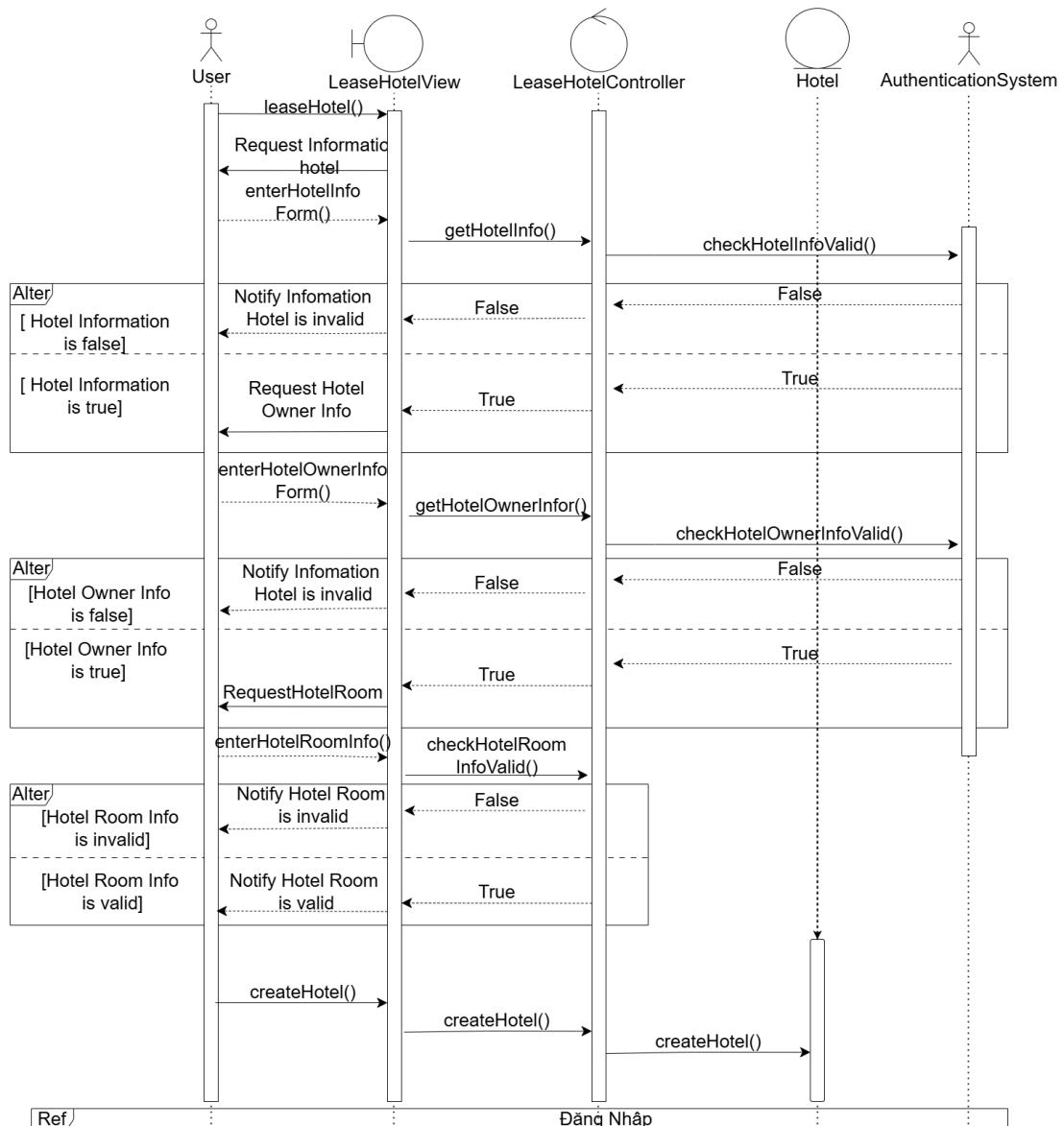
Hình 2.1.4: Biểu đồ tuần tự ca sử dụng Liên hệ với chủ khách sạn

2.1.5 Xem danh sách phòng yêu thích



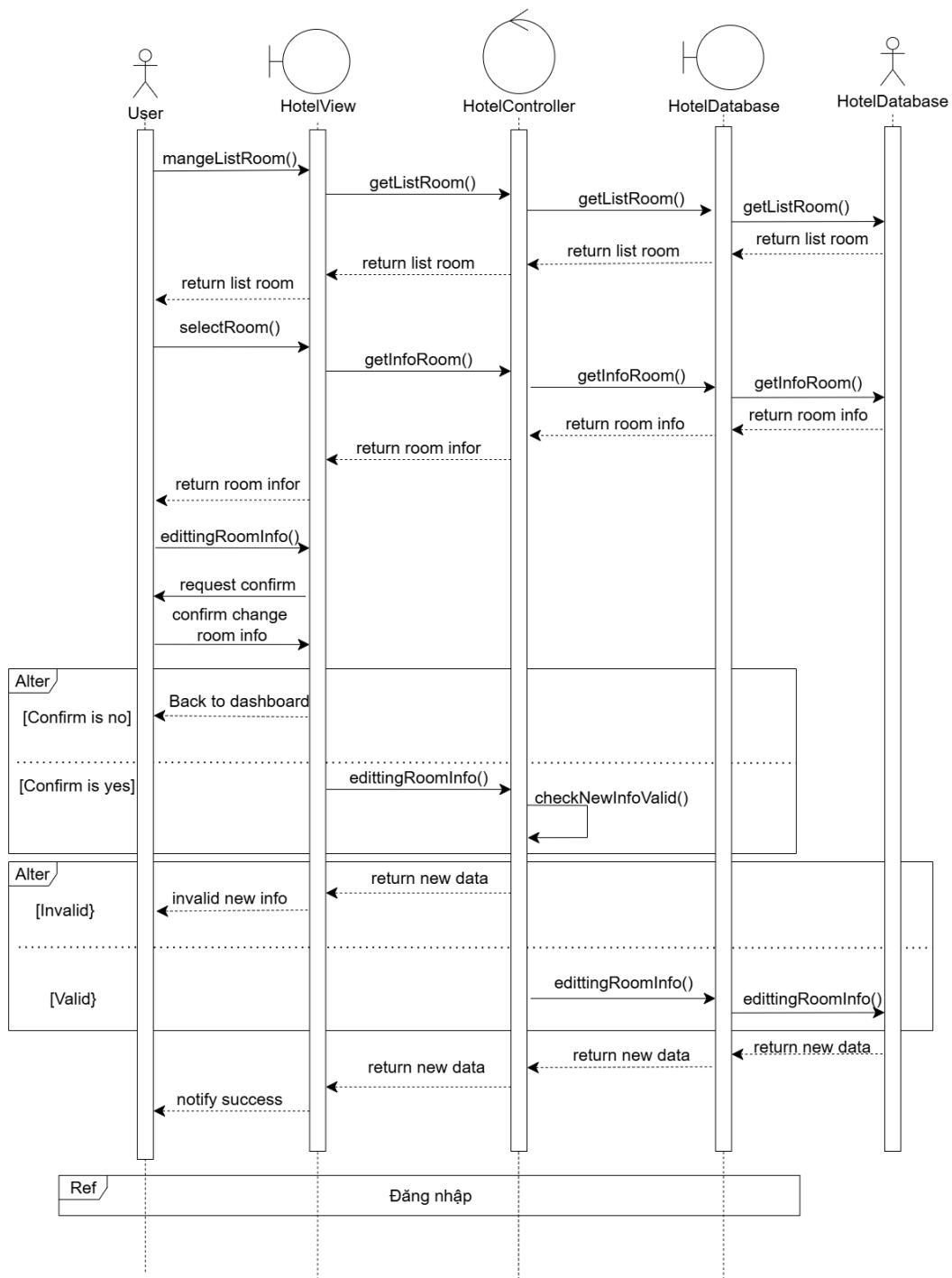
Hình 2.1.5: Biểu đồ tuần tự ca sử dụng Xem danh sách phòng yêu thích

2.1.6 Cho thuê phòng



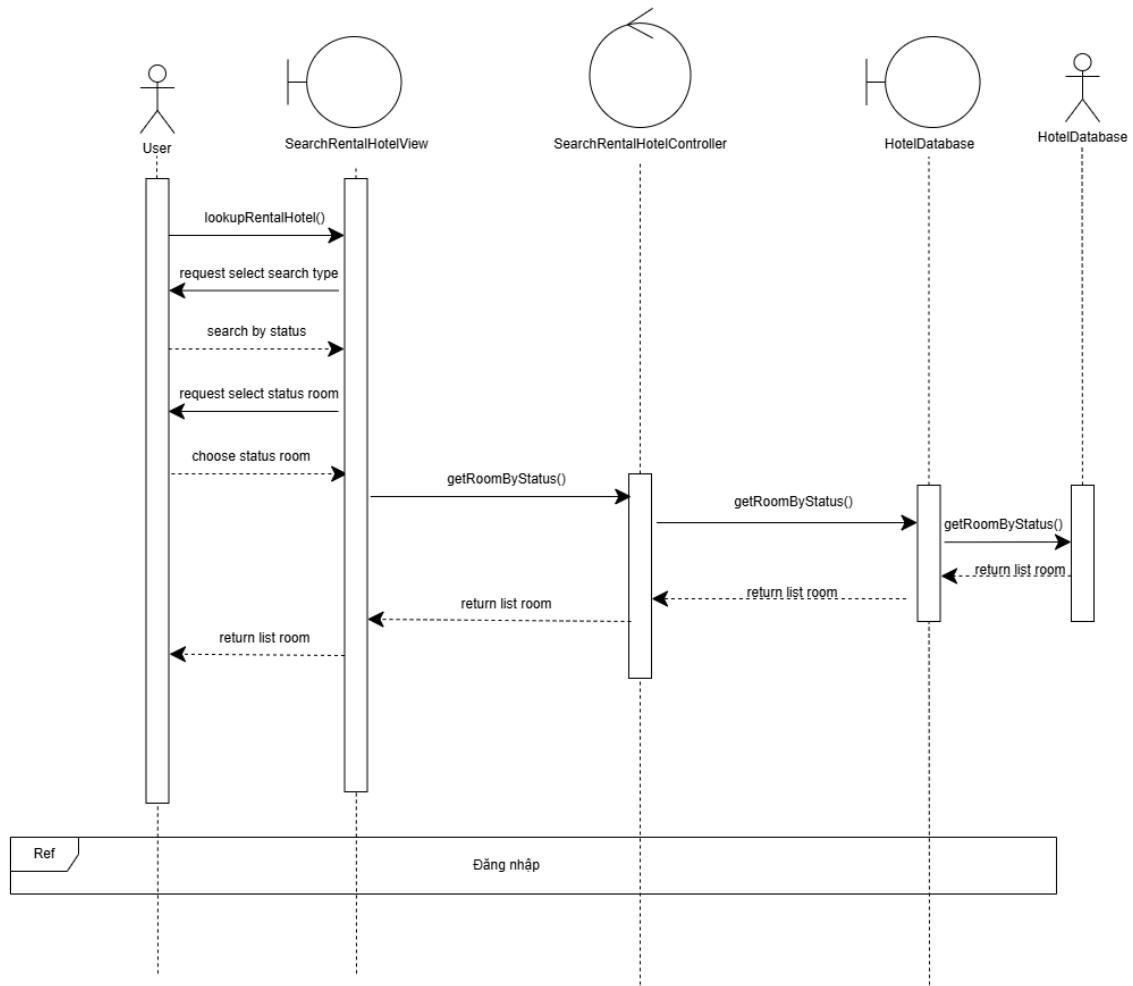
Hình 2.1.6: Biểu đồ tuần tự ca sử dụng Cho thuê phòng

2.1.7 Chỉnh sửa thông tin phòng khách sạn

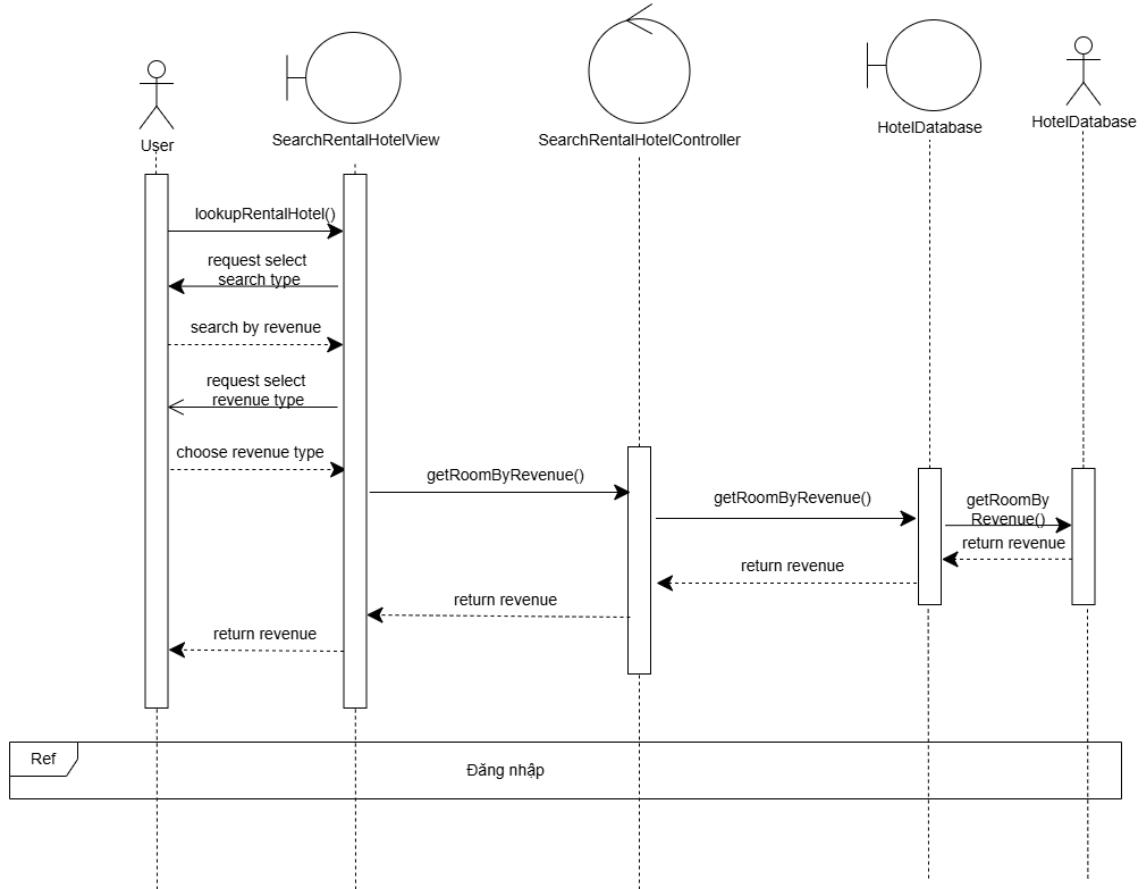


Hình 2.1.7: Biểu đồ tuần tự ca sử dụng Chỉnh sửa thông tin phòng khách sạn

2.1.8 Tra cứu thông tin phòng cho thuê

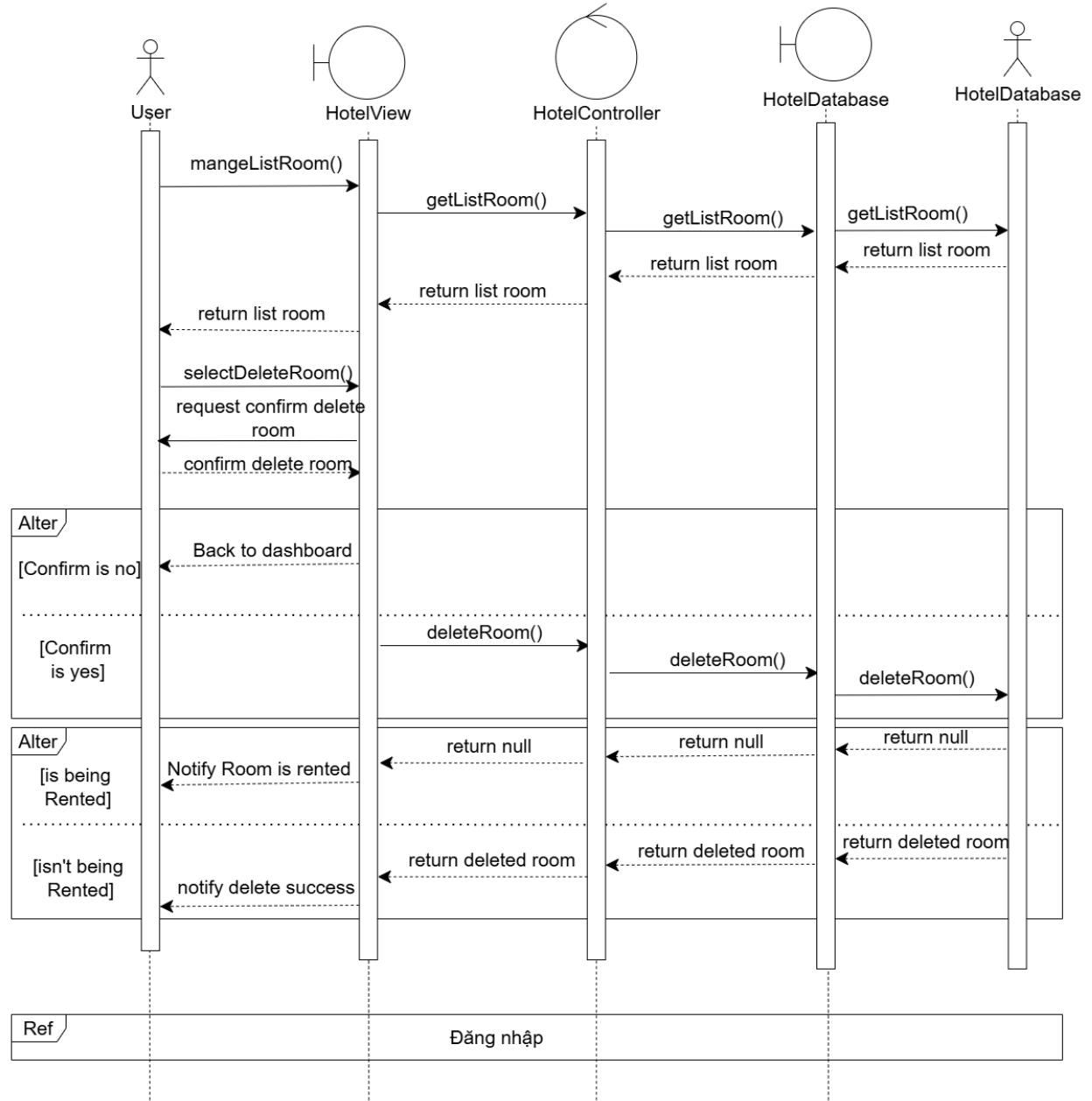


Hình 2.1.8: Biểu đồ tuần tự ca sử dụng Tra cứu phòng cho thuê theo status



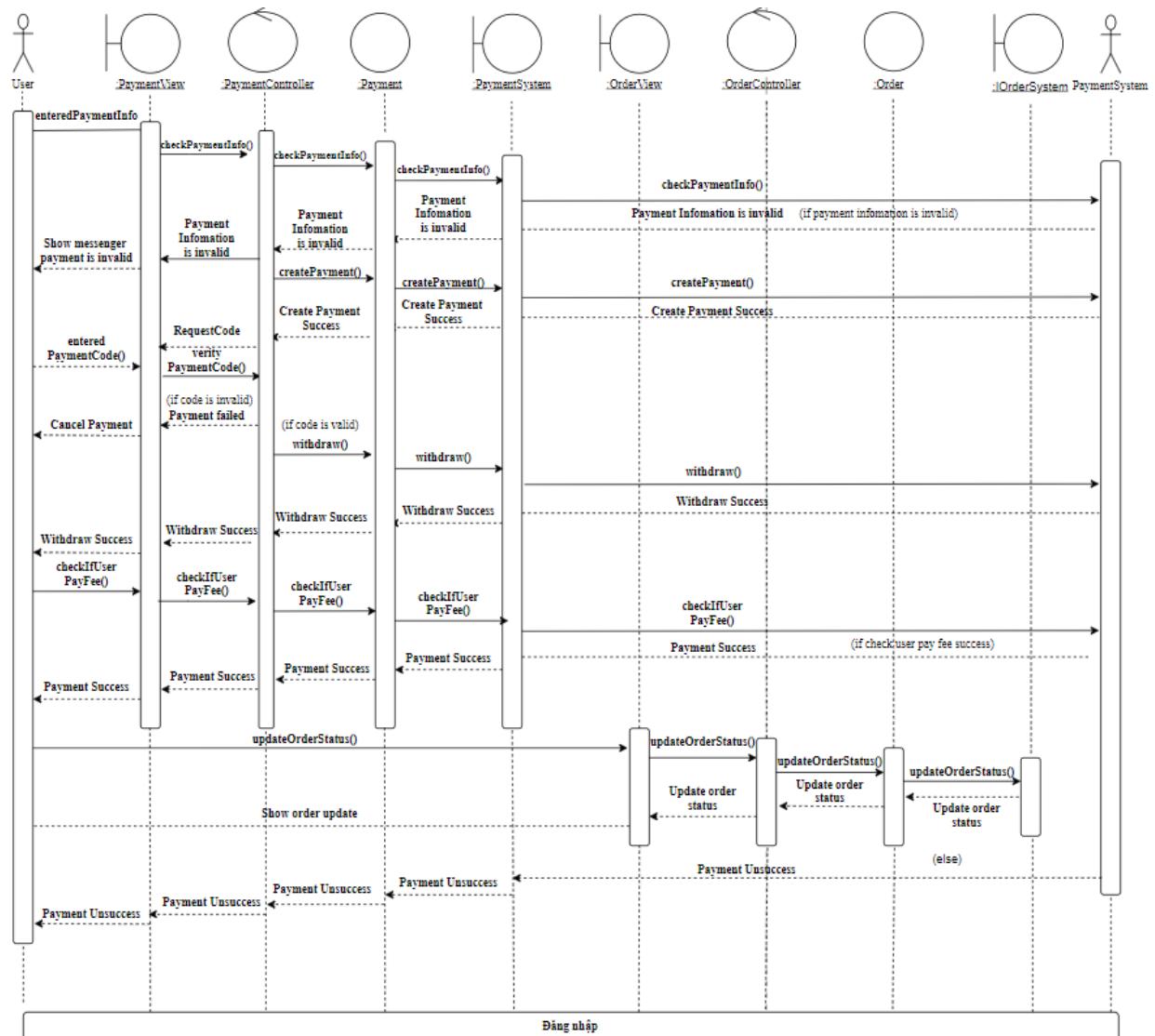
Hình 2.1.9: Biểu đồ tuần tự ca sử dụng Tra cứu phòng cho thuê theo revenue

2.1.9 Xóa phòng khách sạn

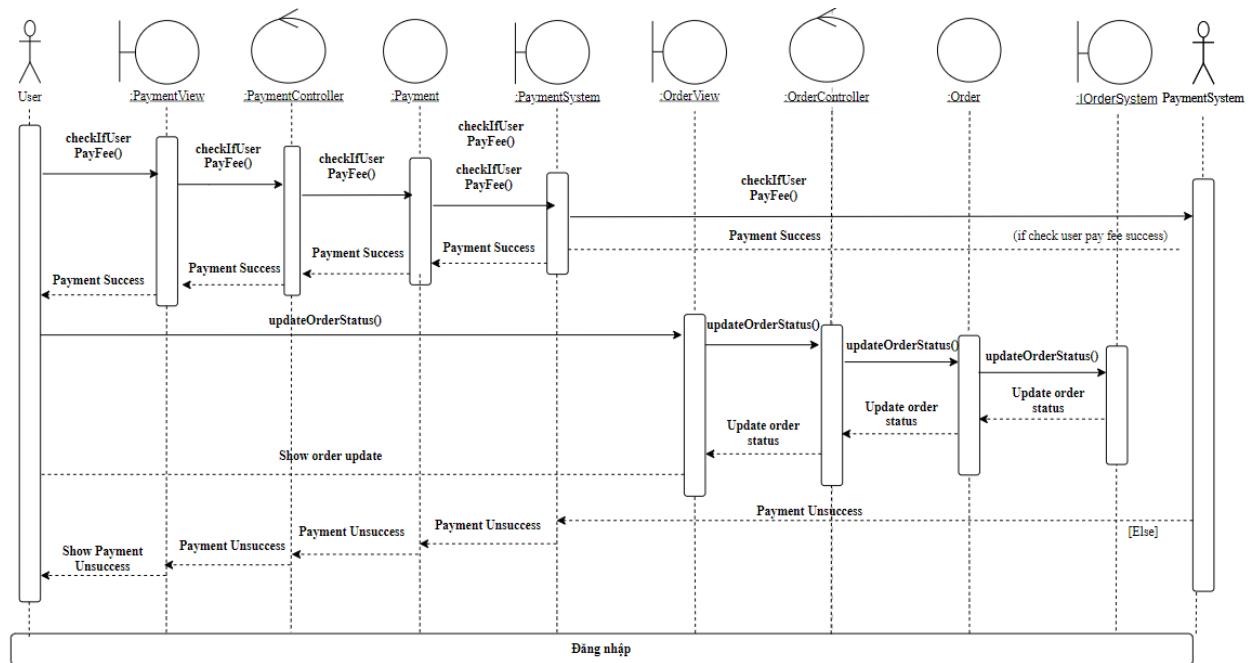


Hình 2.1.10: Biểu đồ tuần tự ca sử dụng Xóa phòng khách sạn

2.1.10 Thanh toán

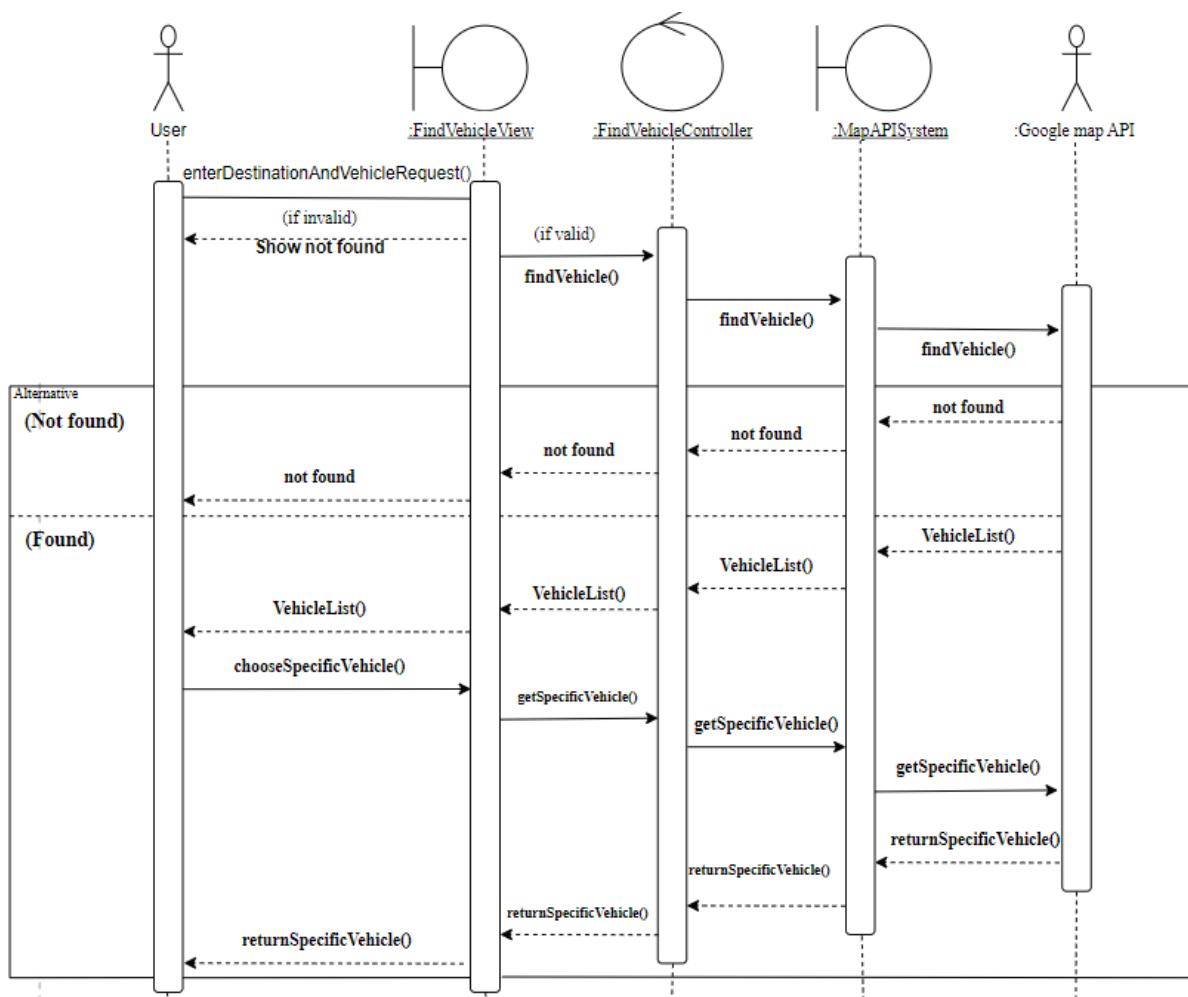


Hình 2.1.11: Biểu đồ tuần tự ca sử dụng Thanh toán trực tuyến



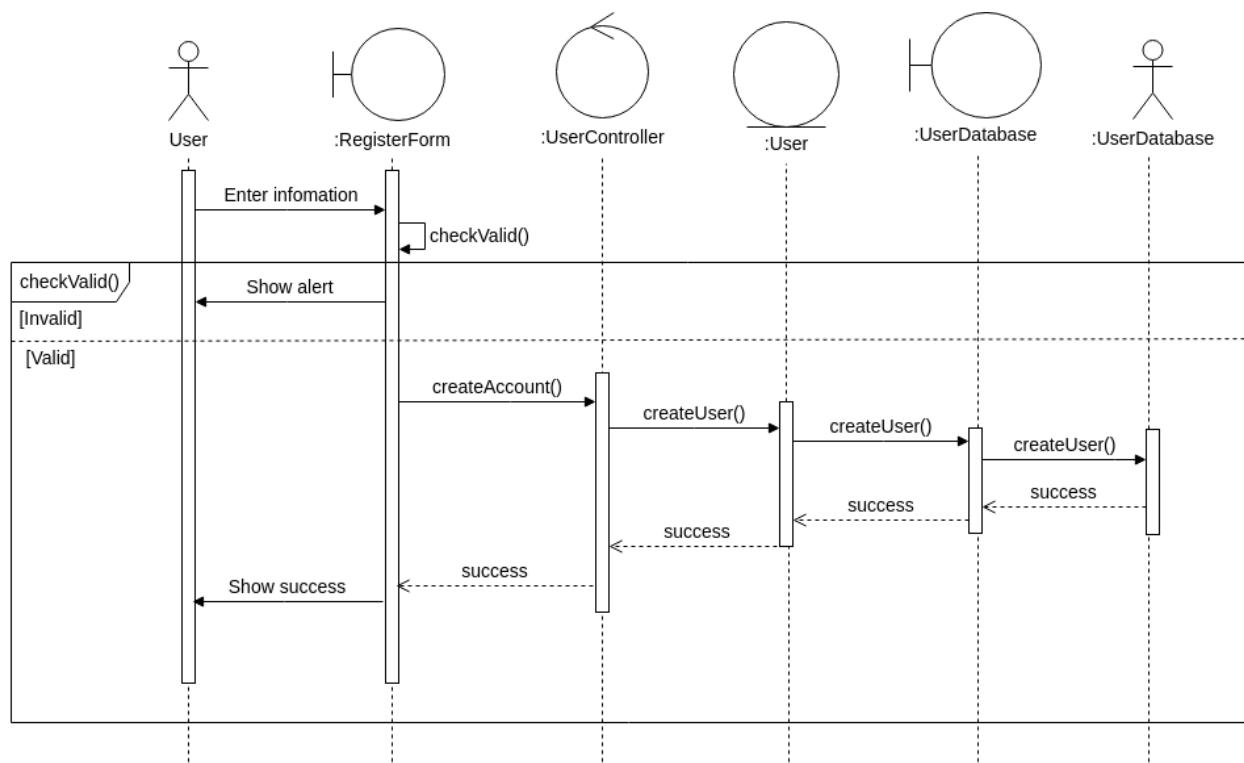
Hình 2.1.12: Biểu đồ tuần tự ca sử dụng Thanh toán trực tiếp

2.1.11 Tra cứu phương tiện



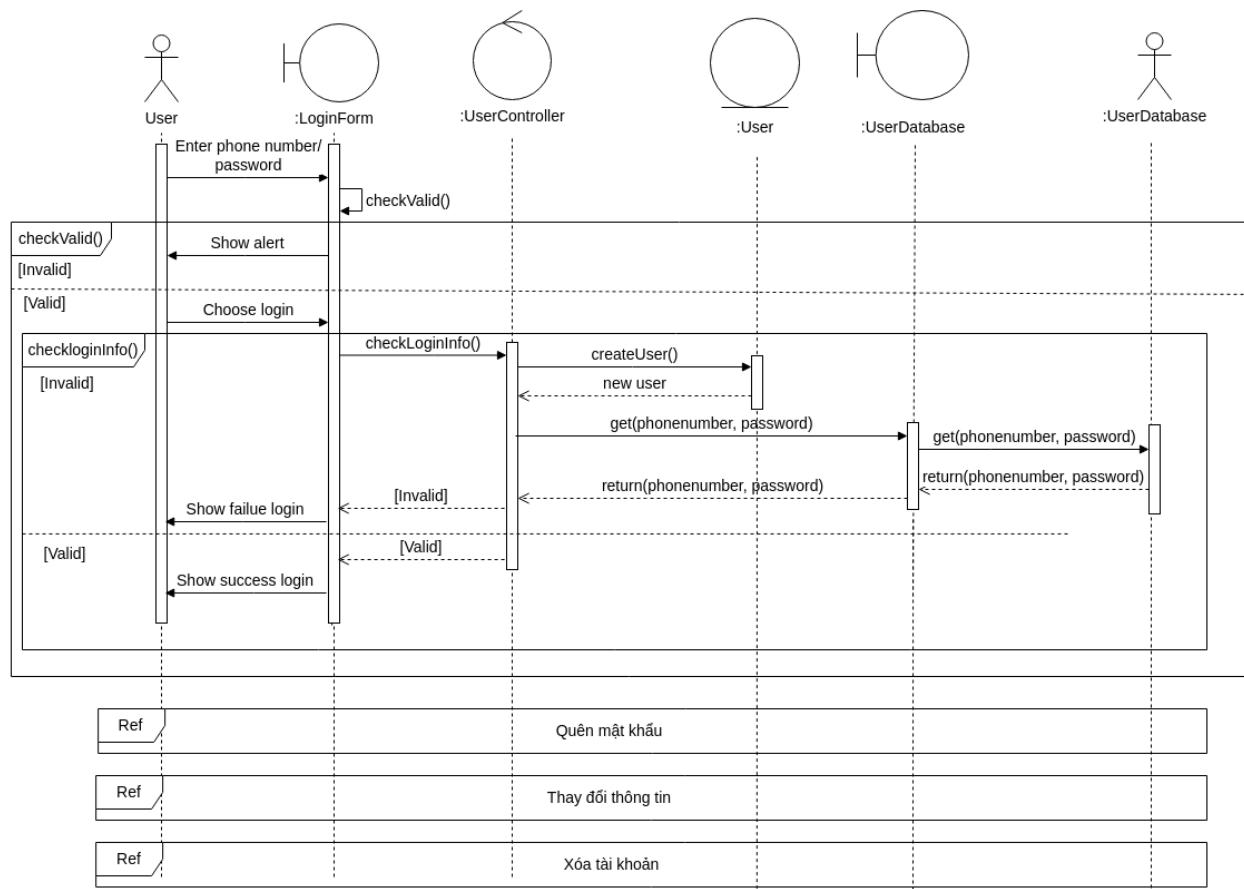
Hình 2.1.13: Biểu đồ tuần tự ca sử dụng Tra cứu phương tiện

2.1.12 Đăng ký tài khoản

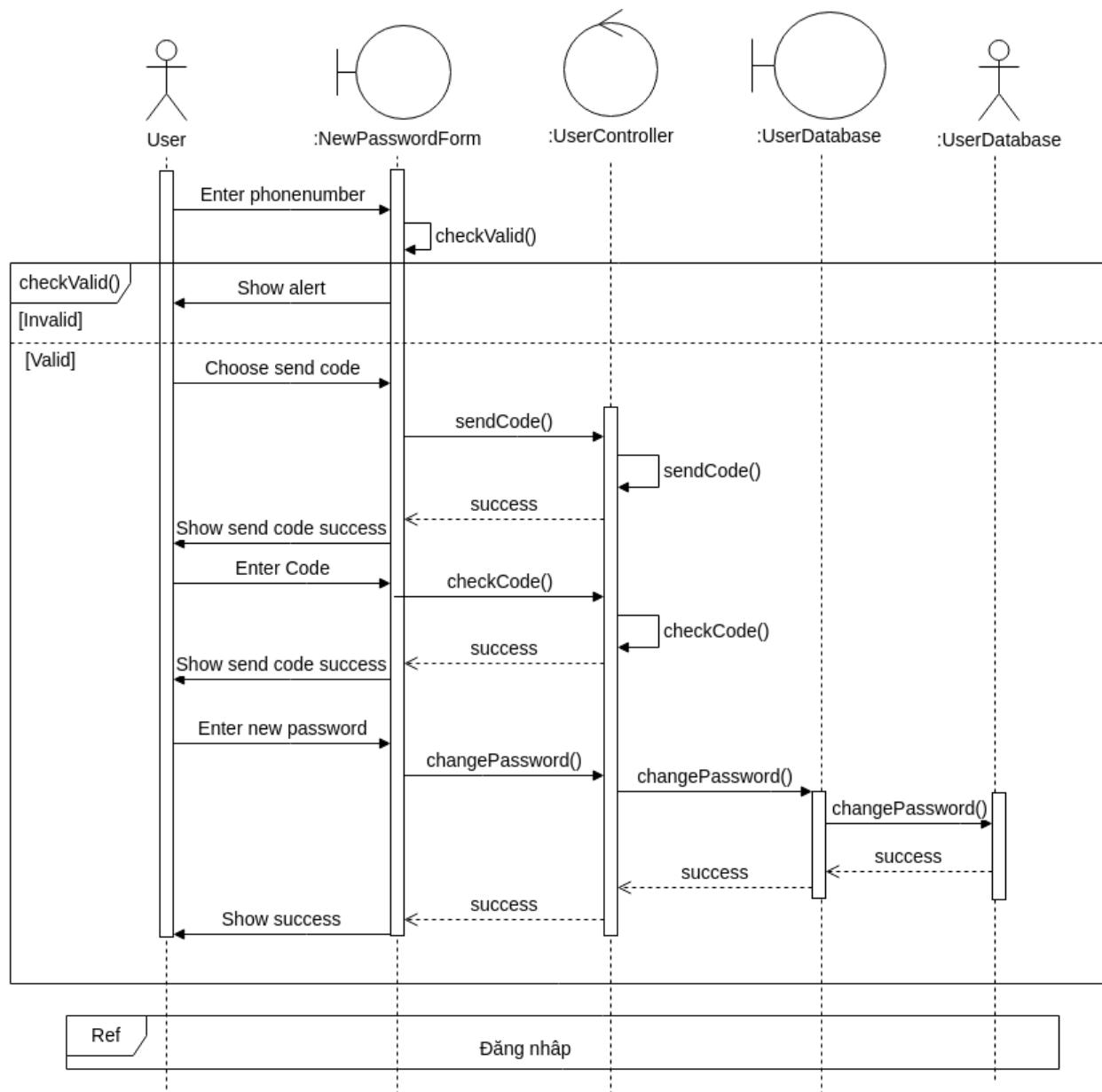


Hình 2.1.14: Biểu đồ tuần tự ca sử dụng Đăng ký tài khoản

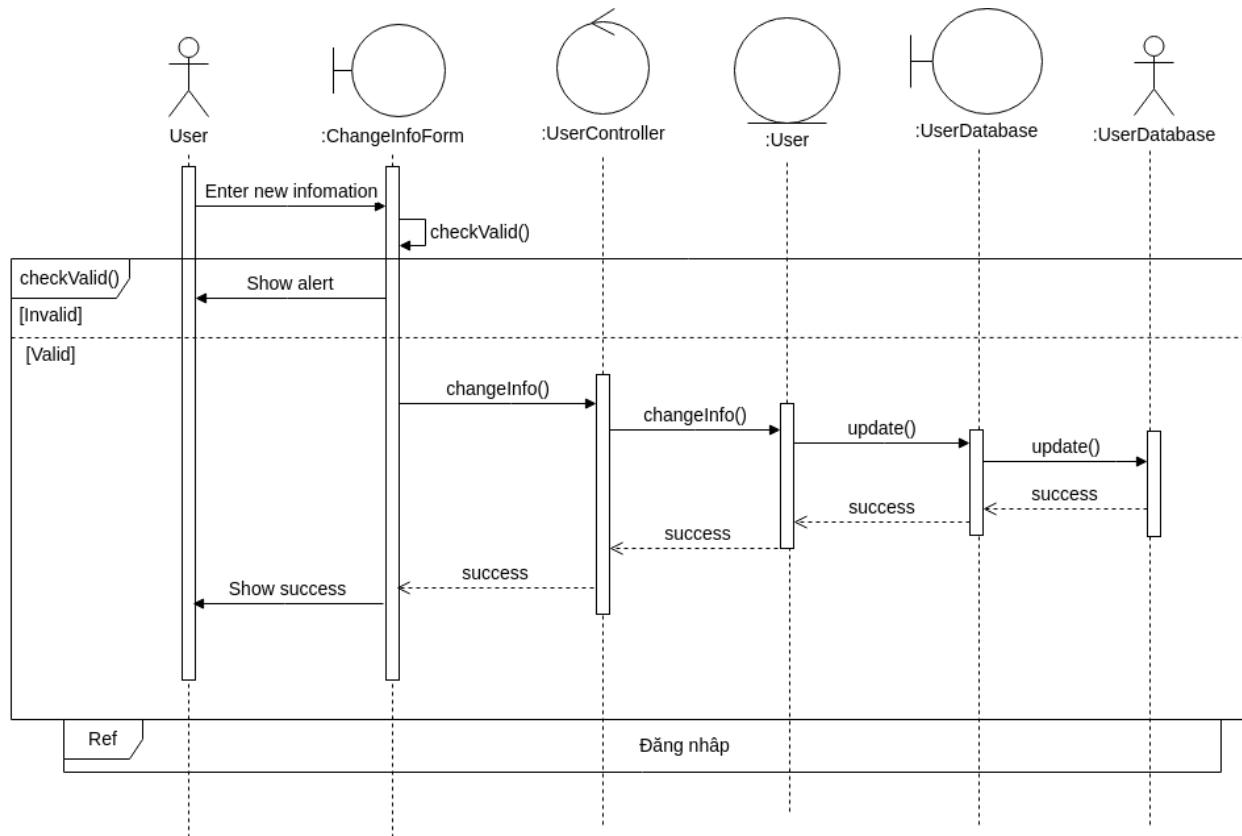
2.1.13 Đăng nhập



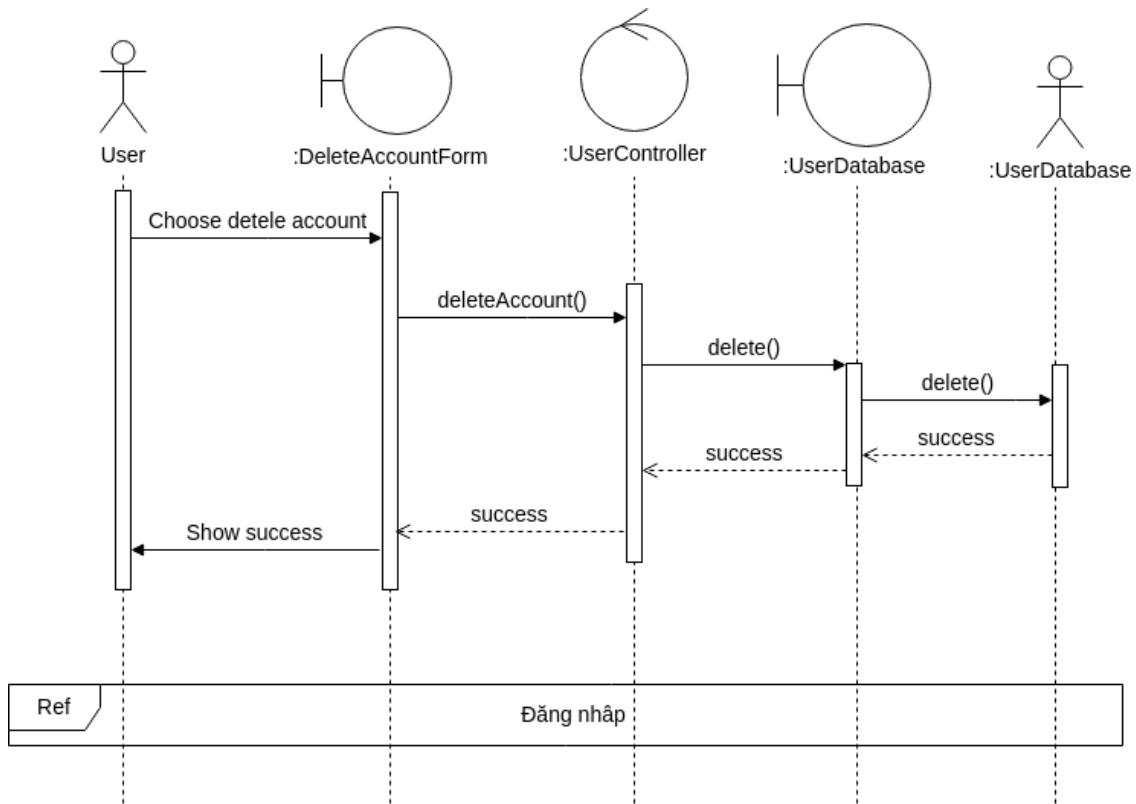
Hình 2.1.15: Biểu đồ tuần tự ca sử dụng Đăng nhập



Hình 2.1.16: Biểu đồ tuần tự ca sử dụng Quên mật khẩu

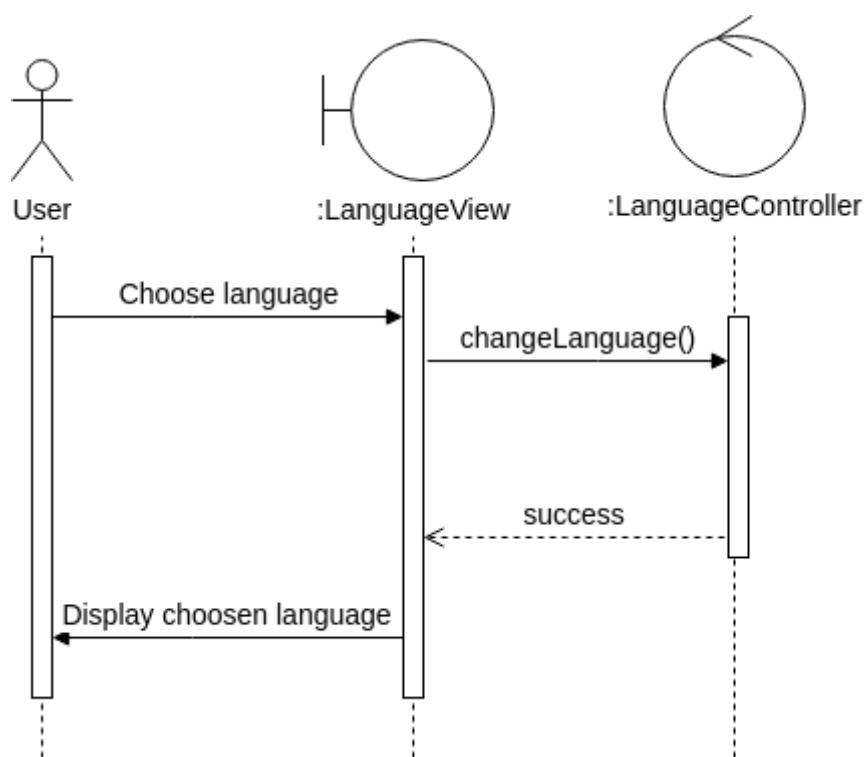


Hình 2.1.17: Biểu đồ tuần tự ca sử dụng Thay đổi thông tin tài khoản



Hình 2.1.18: Biểu đồ tuần tự ca sử dụng Xóa tài khoản

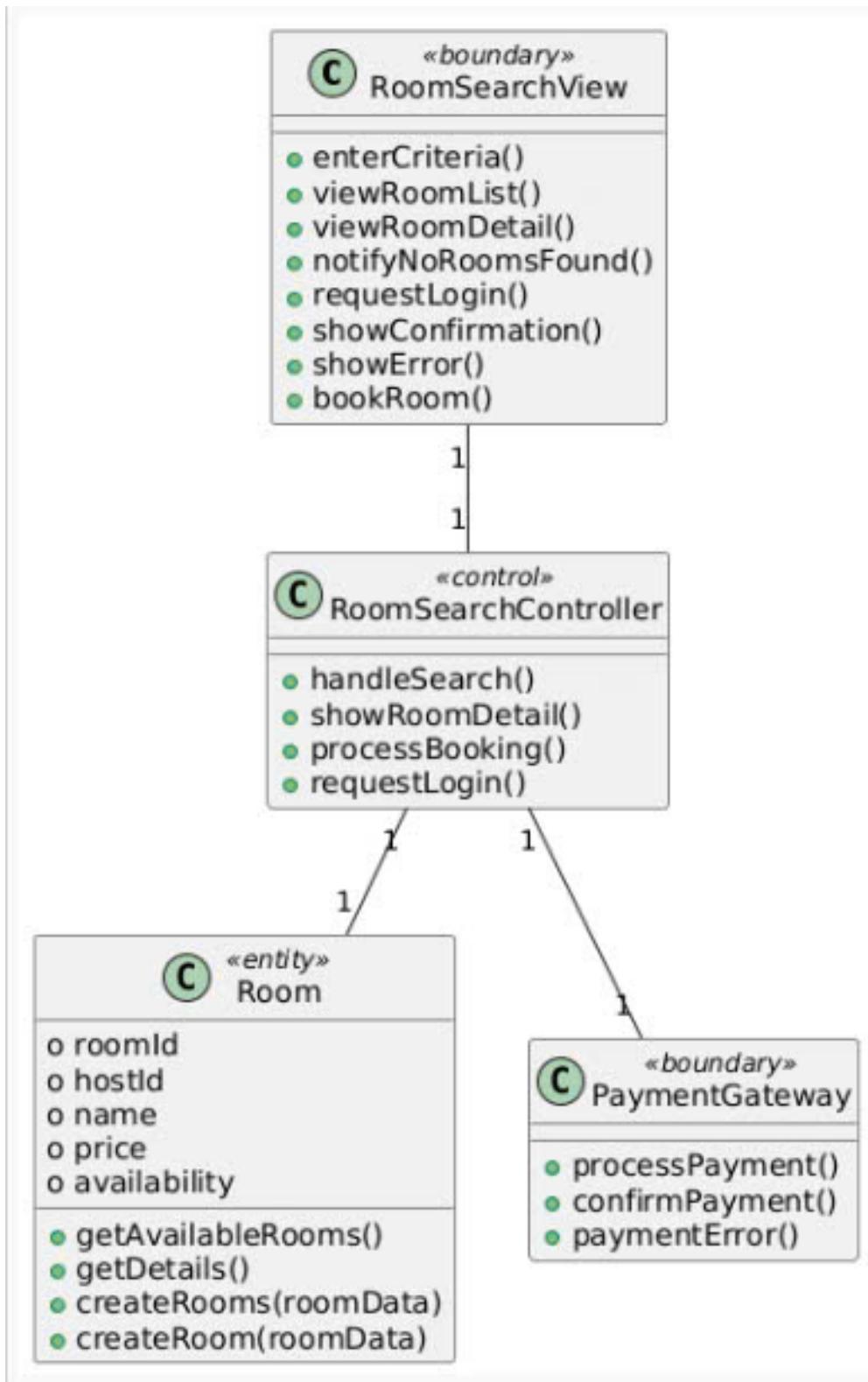
2.1.14 Lựa chọn ngôn ngữ



Hình 2.1.19: Biểu đồ tuần tự ca sử dụng Chọn ngôn ngữ

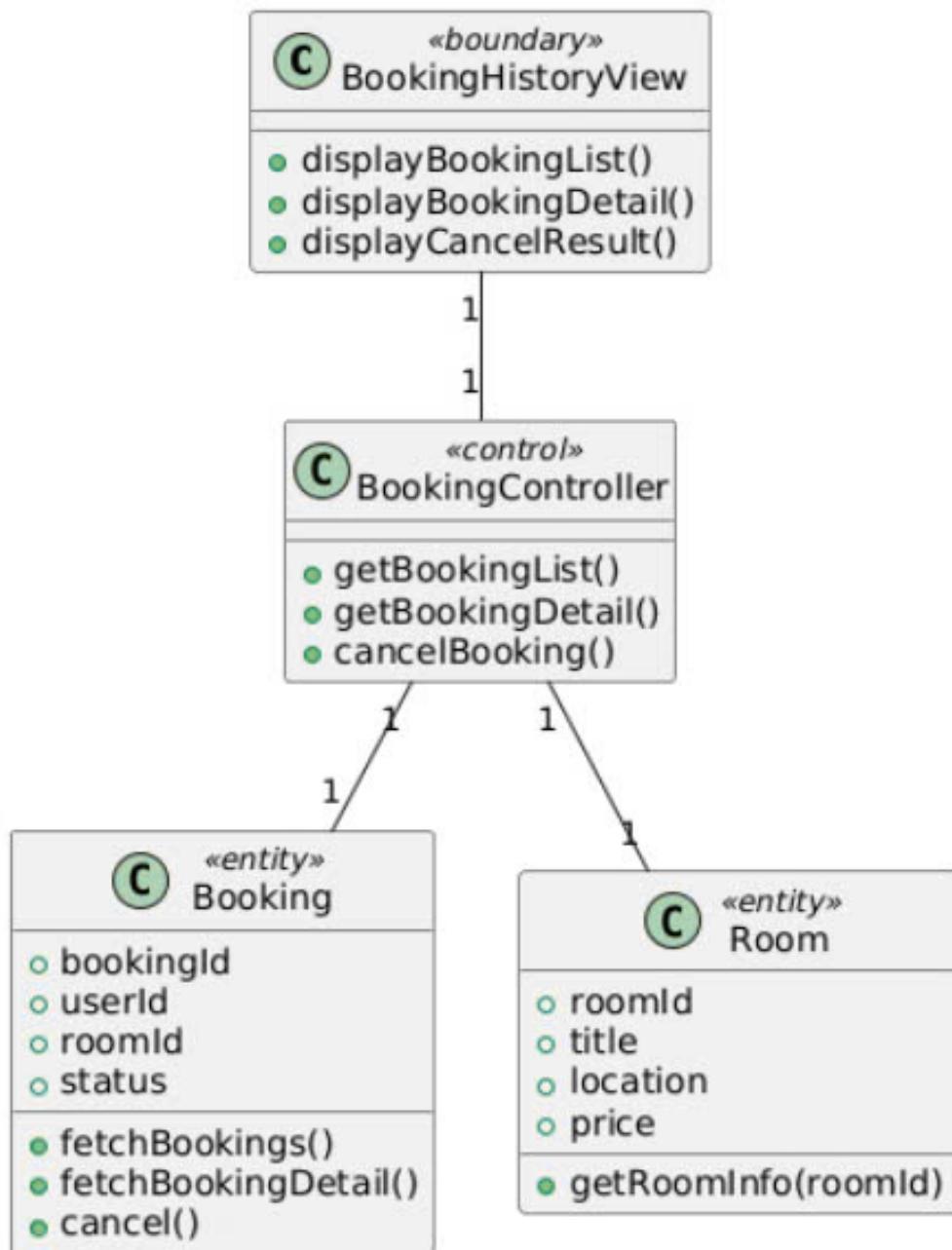
2.2 Biểu đồ lớp pha phân tích

2.2.1 Đặt phòng khách sạn



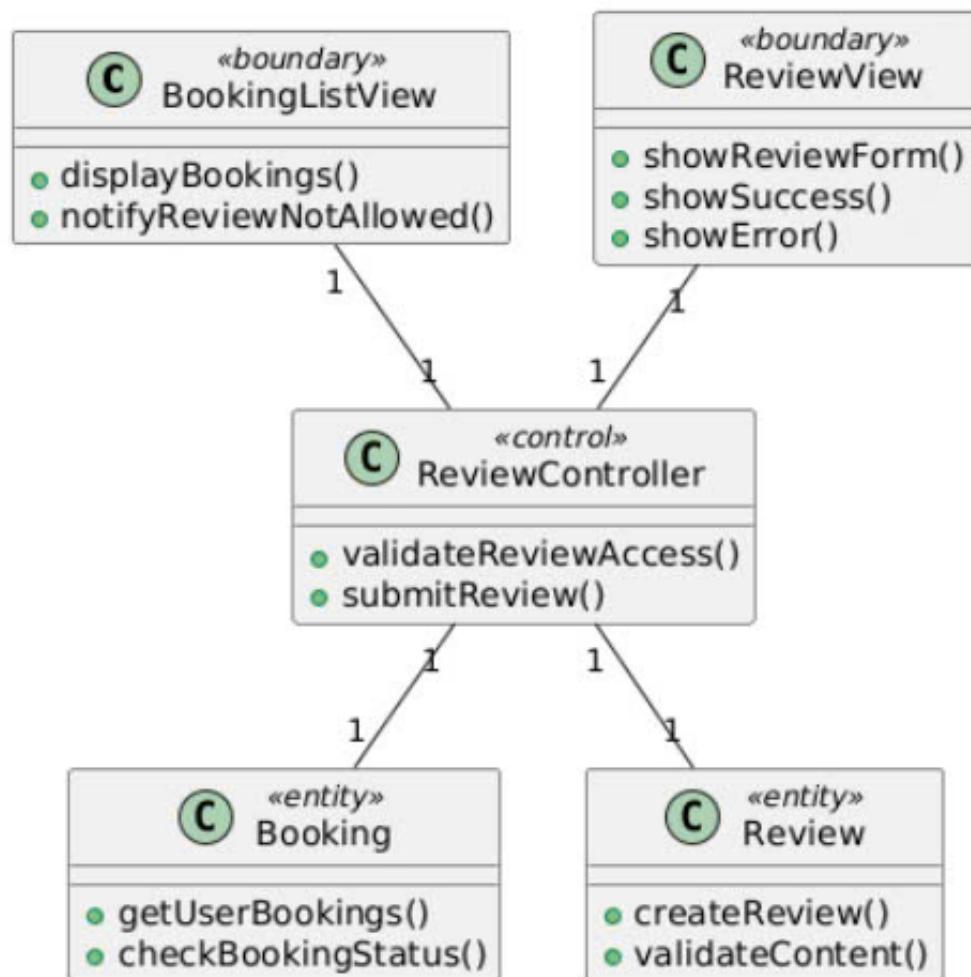
Hình 2.2.1: Biểu đồ lớp cho ca sử dụng Đặt phòng khách sạn

2.2.2 Theo dõi phòng đã đặt



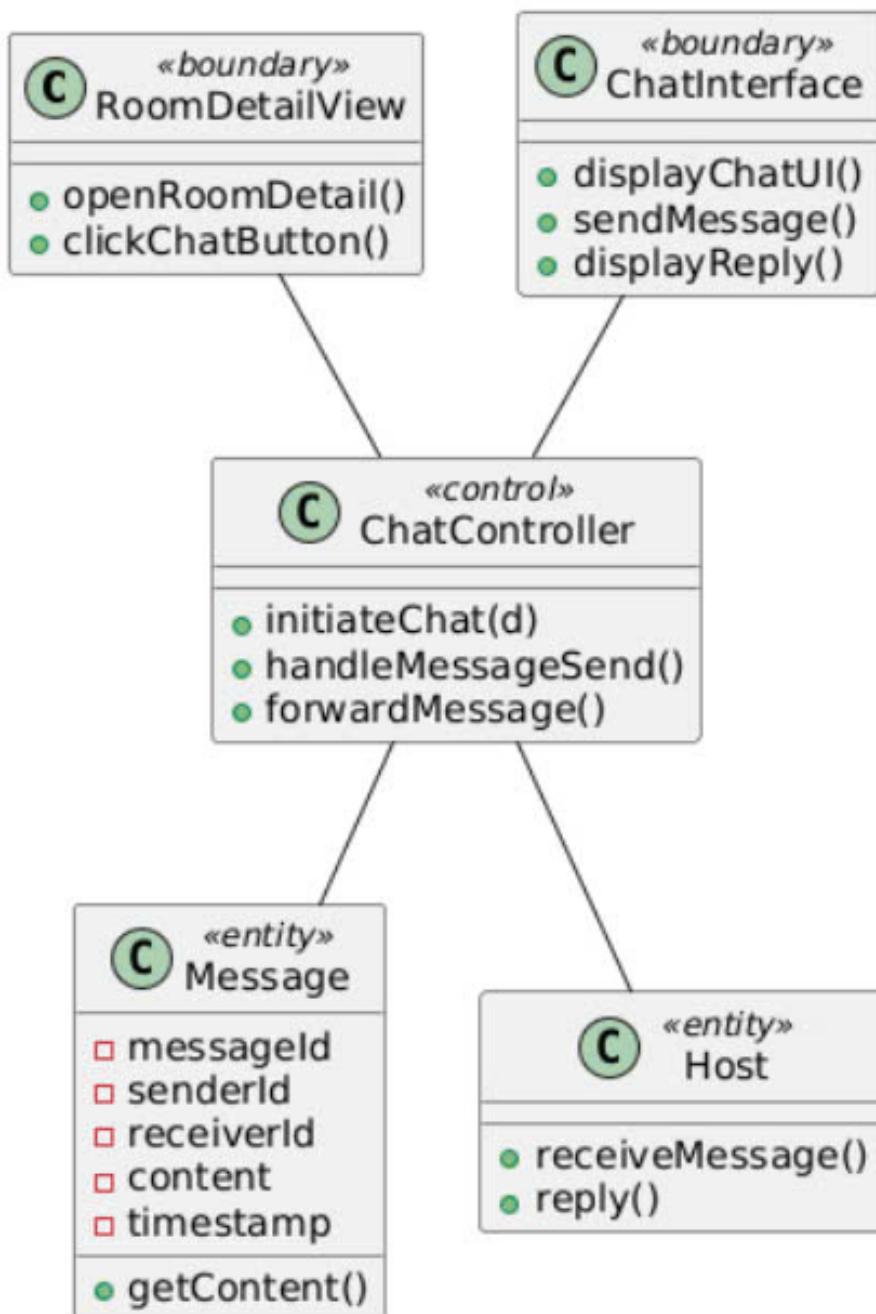
Hình 2.2.2: Biểu đồ lớp ca sử dụng Theo dõi phòng đã đặt

2.2.3 Đánh giá phòng đã thuê



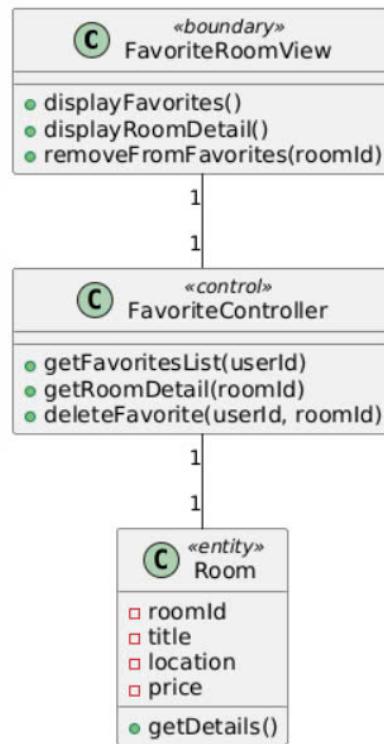
Hình 2.2.3: Biểu đồ lớp ca sử dụng Đánh giá phòng đã thuê

2.2.4 Liên hệ với chủ khách sạn



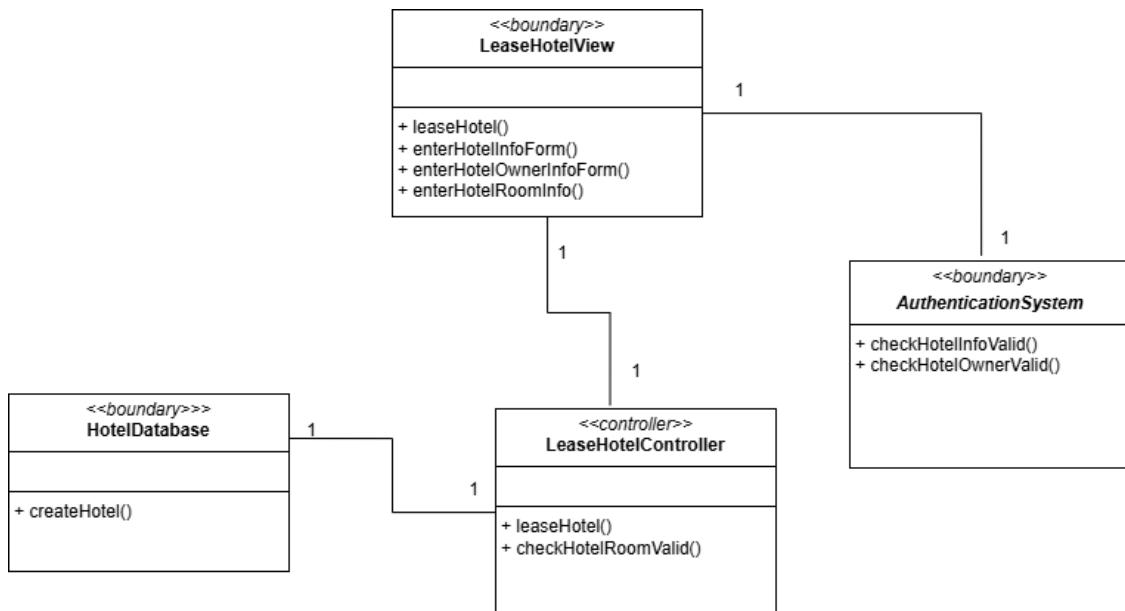
Hình 2.2.4: Biểu đồ lớp ca sử dụng Liên hệ với chủ khách sạn

2.2.5 Xem danh sách phòng yêu thích



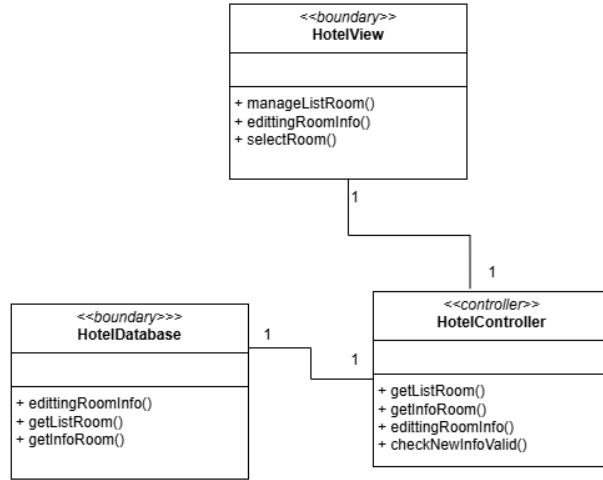
Hình 2.2.5: Biểu đồ lớp ca sử dụng Xem danh sách phòng yêu thích

2.2.6 Cho thuê phòng



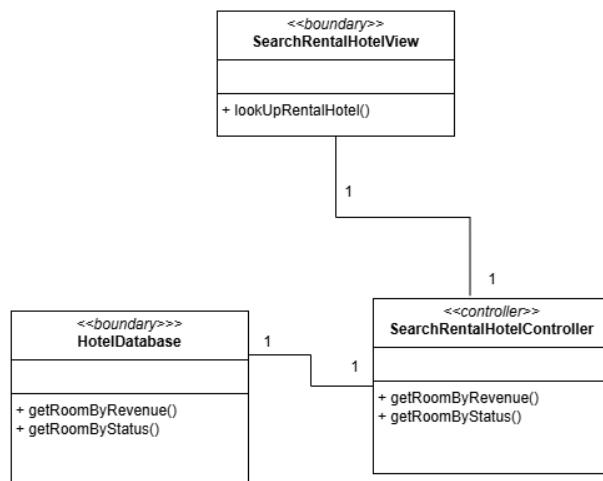
Hình 2.2.6: Biểu đồ lớp ca sử dụng Cho thuê phòng

2.2.7 Chỉnh sửa thông tin phòng khách sạn



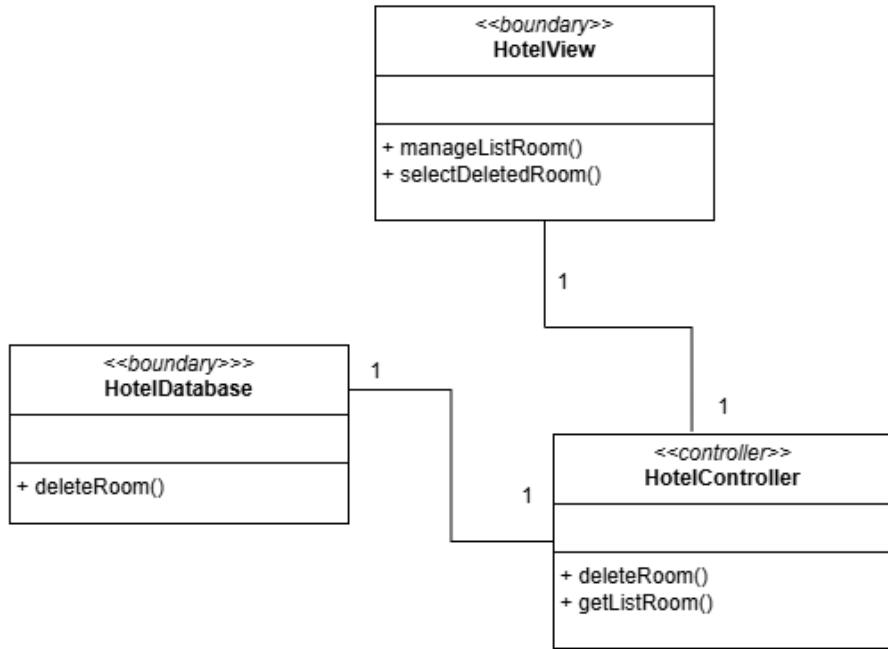
Hình 2.2.7: Biểu đồ lớp ca sử dụng Chỉnh sửa thông tin phòng khách sạn

2.2.8 Tra cứu thông tin phòng cho thuê



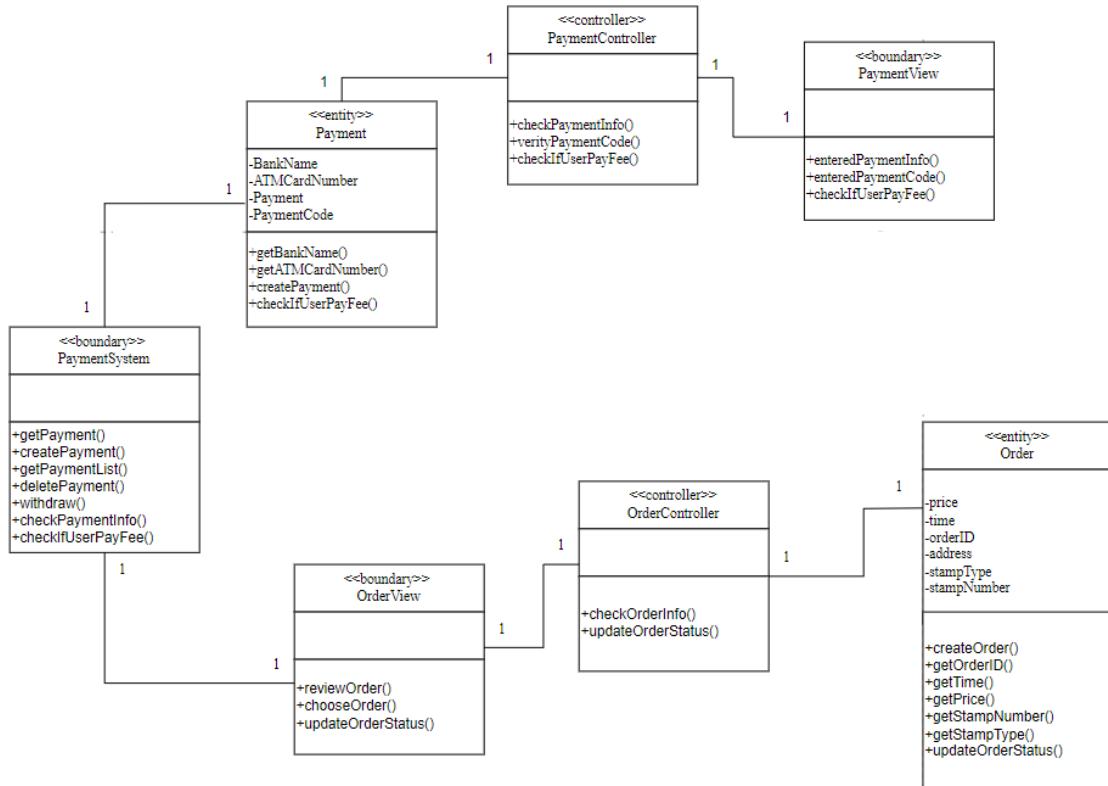
Hình 2.2.8: Biểu đồ lớp ca sử dụng Tra cứu phòng cho thuê

2.2.9 Xóa phòng khách sạn



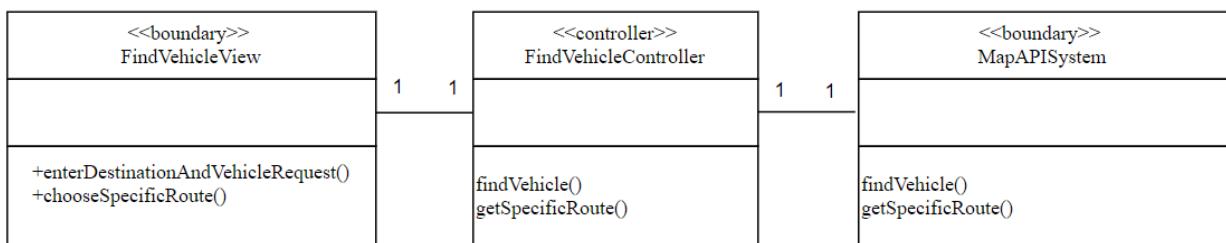
Hình 2.2.9: Biểu đồ lớp ca sử dụng Xóa phòng khách sạn

2.2.10 Thanh toán



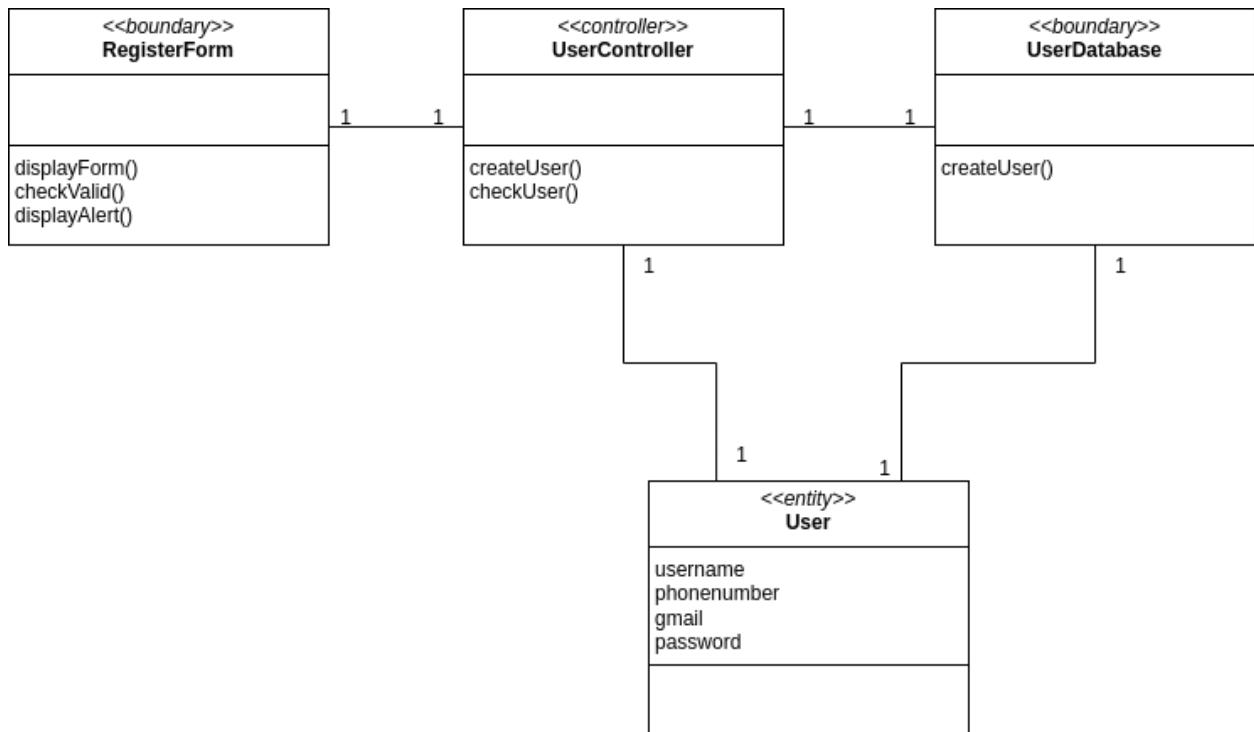
Hình 2.2.10: Biểu đồ lớp ca sử dụng Thanh toán

2.2.11 Tra cứu phương tiện



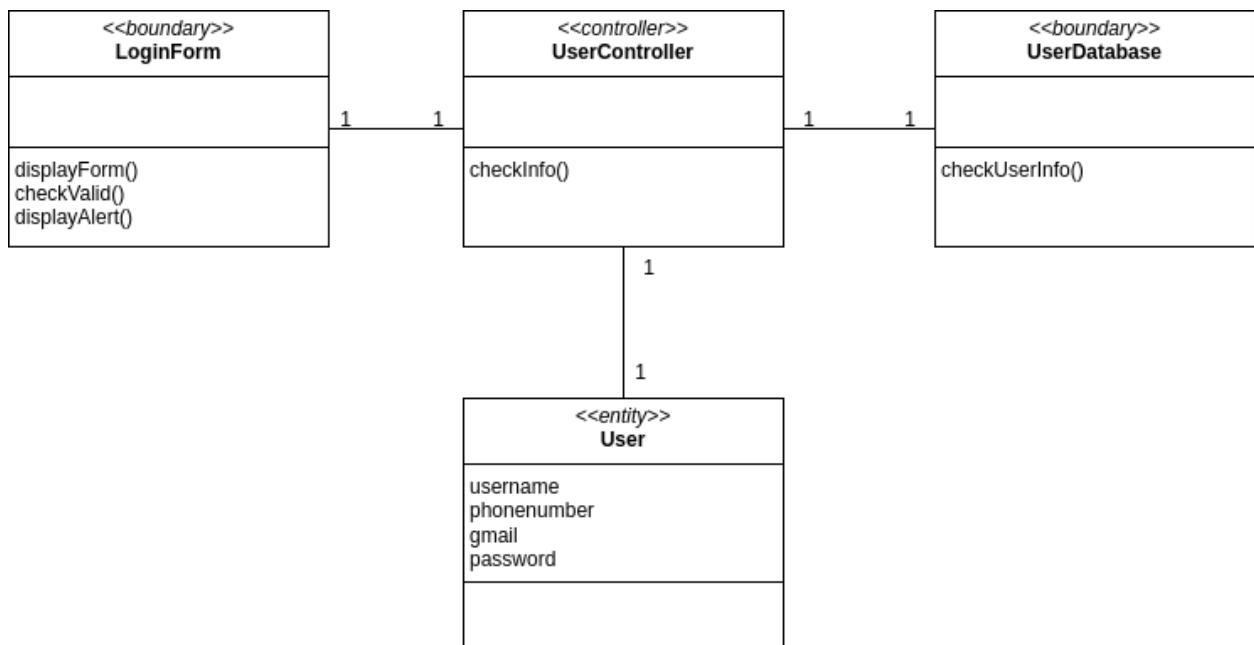
Hình 2.2.11: Biểu đồ lớp ca sử dụng Tra cứu phương tiện

2.2.12 Đăng ký tài khoản

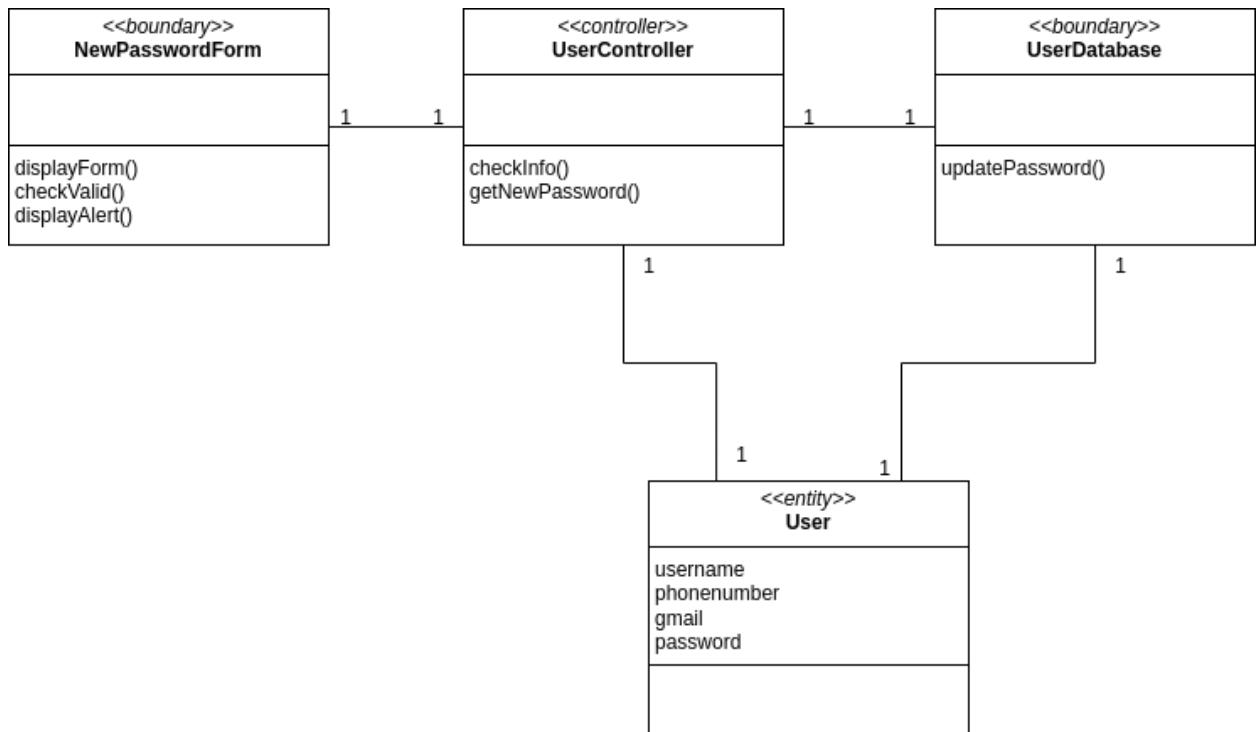


Hình 2.2.12: Biểu đồ lớp ca sử dụng Đăng ký tài khoản

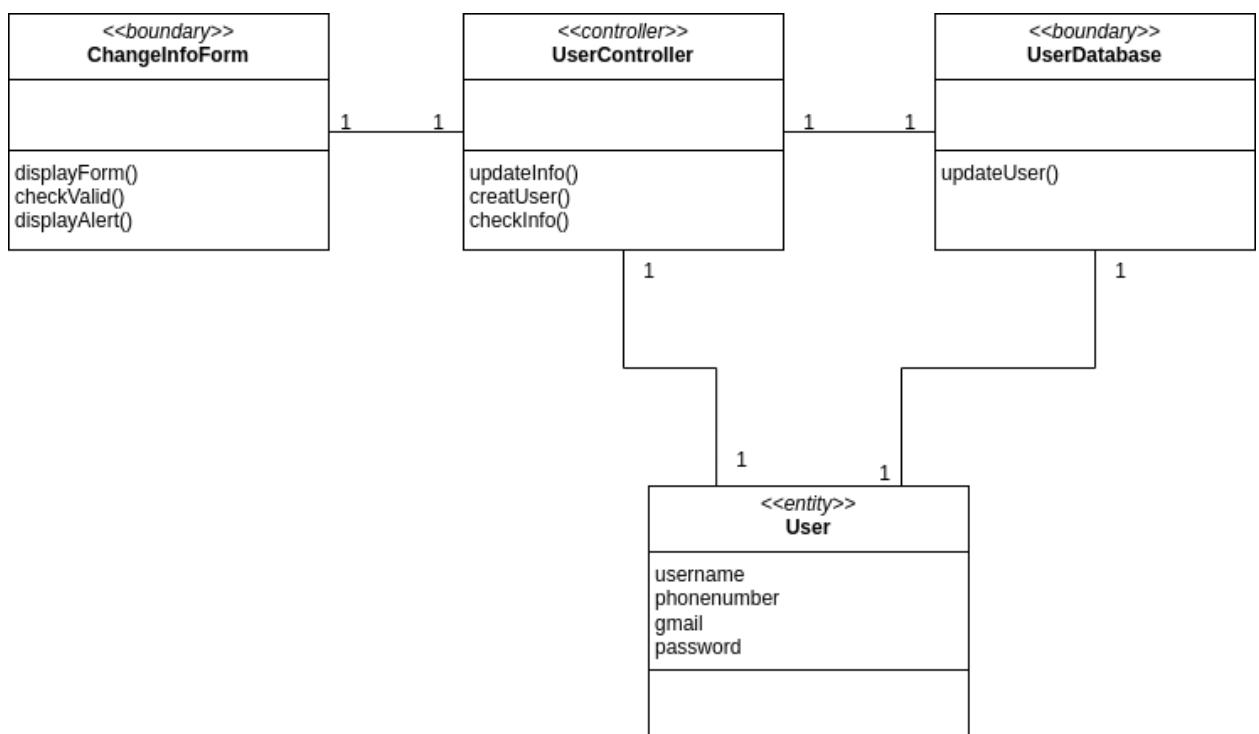
2.2.13 Đăng nhập



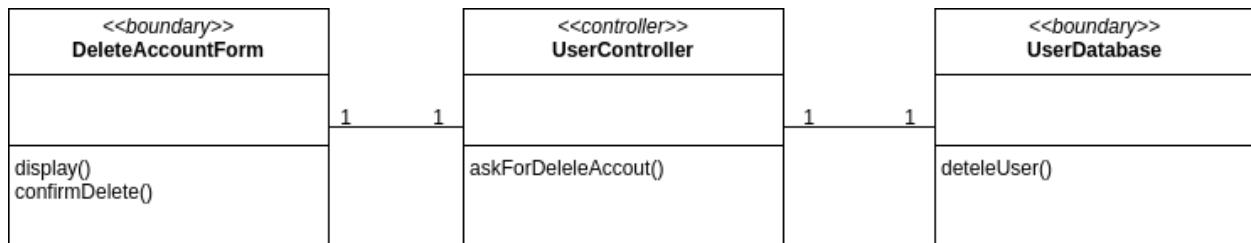
Hình 2.2.13: Biểu đồ lớp ca sử dụng Đăng nhập



Hình 2.2.14: Biểu đồ lớp ca sử dụng Quên mật khẩu

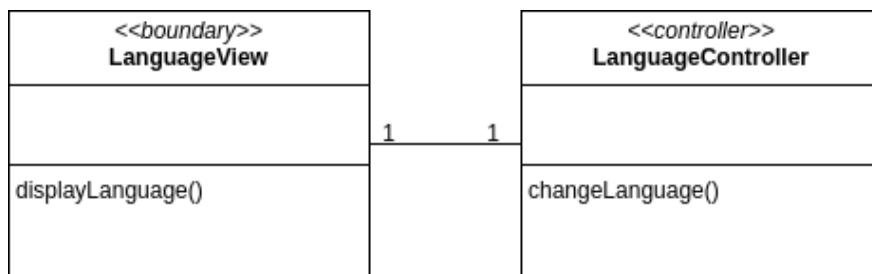


Hình 2.2.15: Biểu đồ lớp ca sử dụng Thay đổi thông tin tài khoản



Hình 2.2.16: Biểu đồ lớp ca sử dụng Xóa tài khoản

2.2.14 Lựa chọn ngôn ngữ



Hình 2.2.17: Biểu đồ lớp ca sử dụng Chọn ngôn ngữ

2.3 Ánh xạ từ lớp phân tích tới cơ chế phân tích

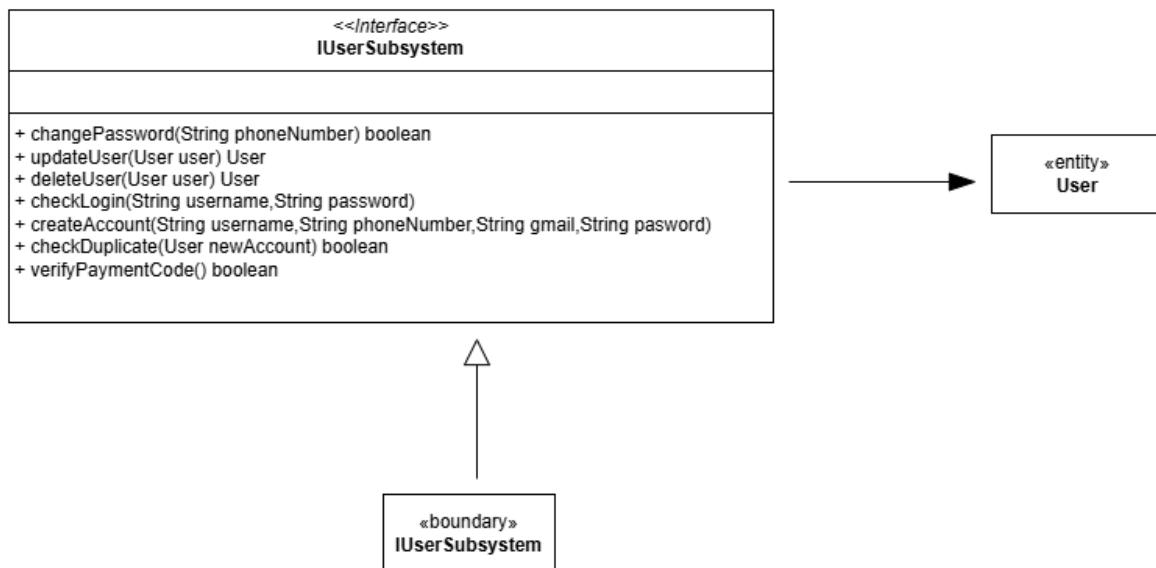
Analysis Class	Analysis Mechanism(s)
RoomSearchView	None
RoomSearchController	Distribution
Room	Security, Persistency
PaymentGateway	Security, Error detection /handling /reporting
NotificationSystem	None
BookingHistoryView	None
BookingController	Distribution, Error detection /handling /reporting
Booking	Security, Persistency
BookingListView	None
ReviewView	None
ReviewController	Distribution
Review	Persistency, Security
RoomDetailView	None
ChatController	None
ChatInterface	None
Message	Persistency, Security
Host	Persistency, Security
FavoriteRoomView	None
FavoriteController	Distribution
Room	Persistency, Security
UserController	Distribution

LeaseHotelView	None
AuthenticationSystem	None
LeaseHotelController	Distribution
HotelDatabase	Persistency, Security
HotelView	None
HotelController	Distribution
SearchRentalHotelView	None
SearchRentalHotelController	Distribution
PaymentView	None
PaymentController	Distribution, Error detection /handling /reporting
Payment	Persistency, Security
PaymentSystem	Security, Error detection /handling /reporting
OrderView	None
OrderController	Distribution
Order	Persistency, Security
FindVehicleView	None
FindVehicleController	Distribution
LeaseHotelController	Distribution
MapAPISystem	None
RegisterForm	None
UserDatabase	Persistency, Security
User	Persistency, Security
LoginForm	None
NewPasswordForm	None
ChangeInfoForm	None
DeleteAccountForm	None
LanguageView	None
LanguageController	None

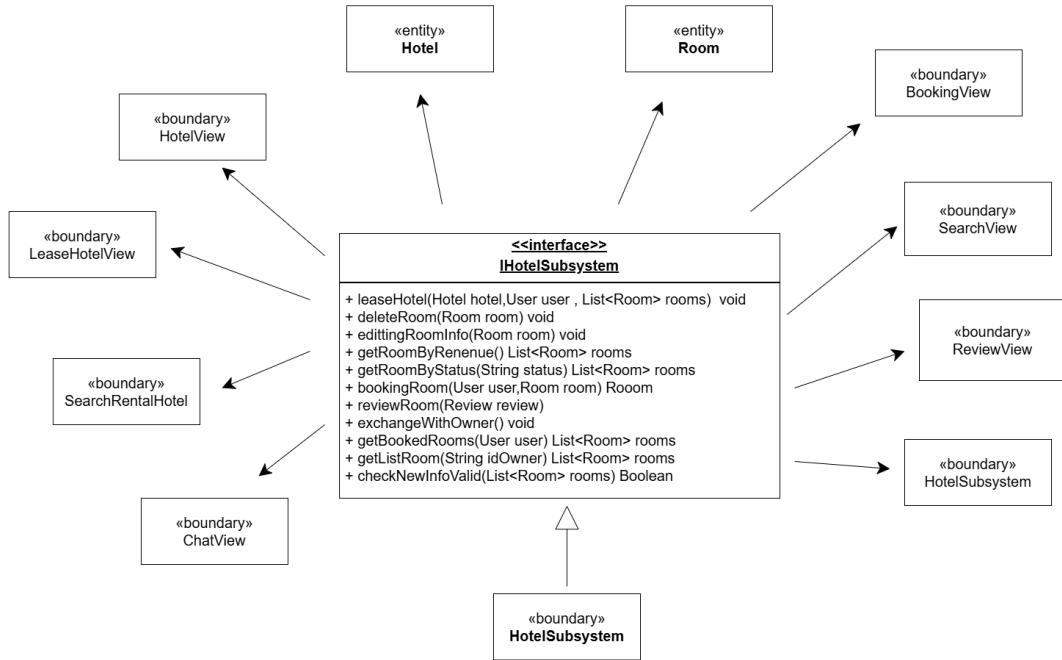
III Thiết kế hệ thống

1 Xác định các thành phần cần thiết kế

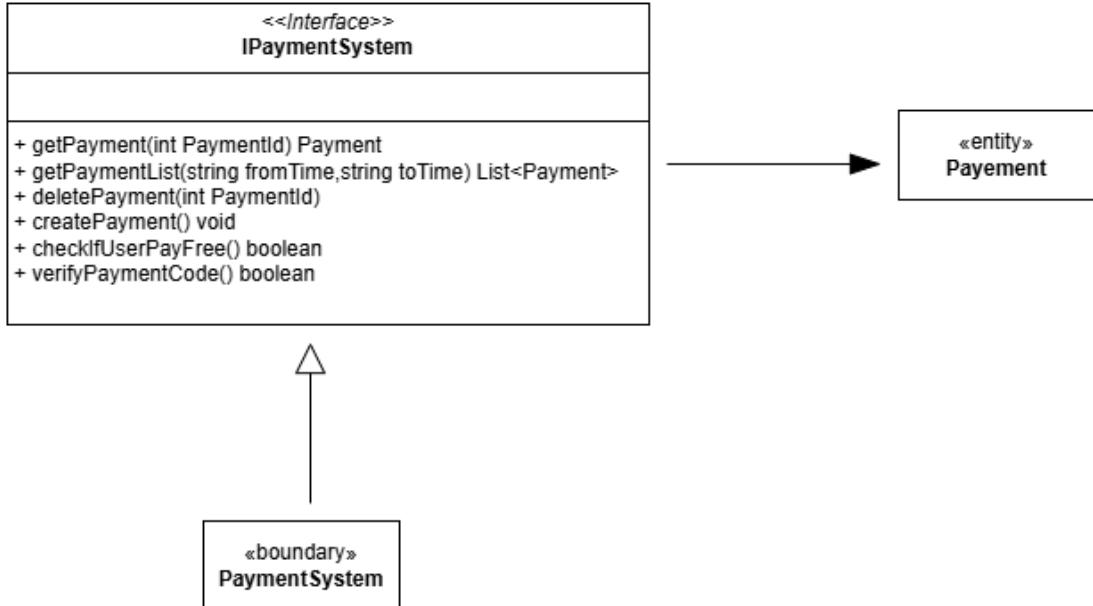
1.1 Subsystem Context



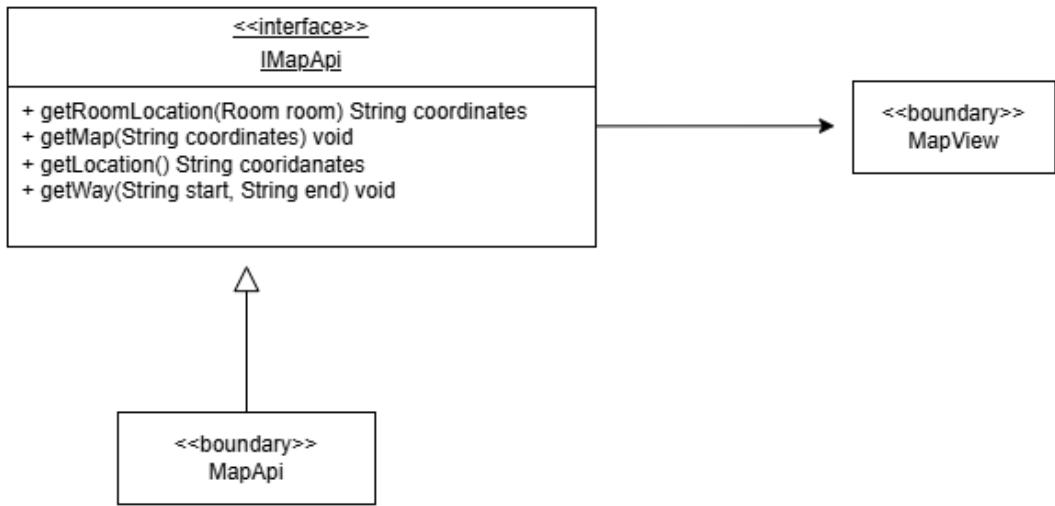
Hình 1.1.1: User Subsystem Context



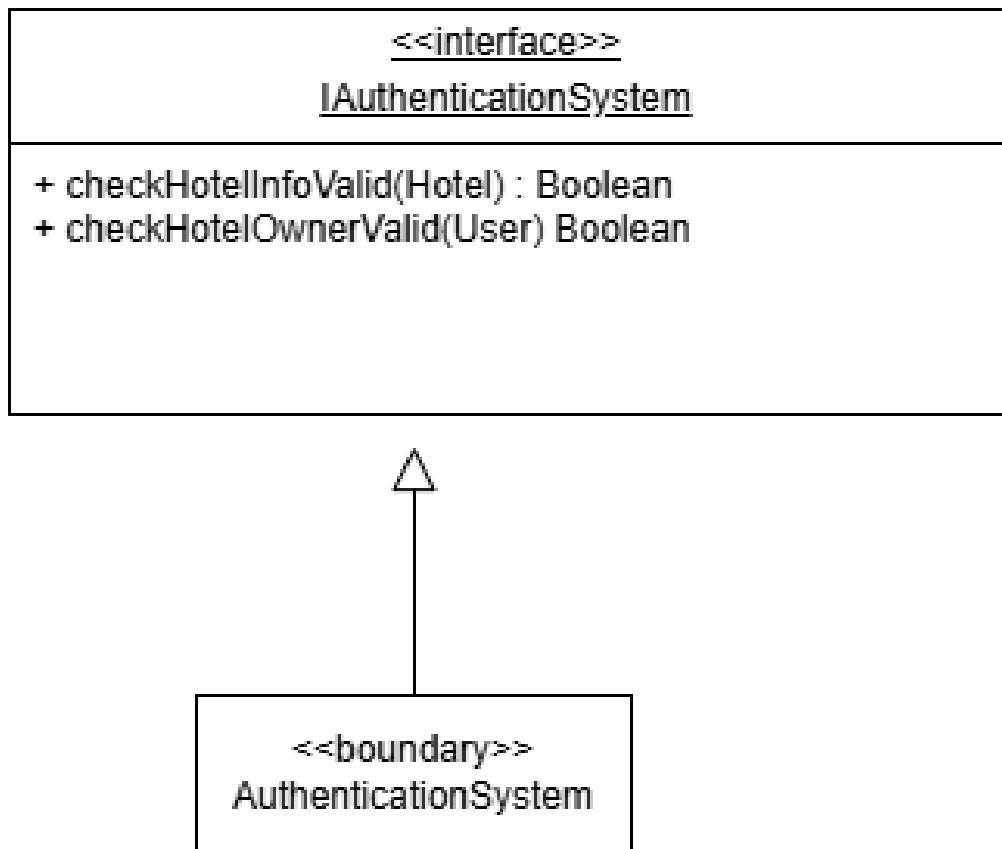
Hình 1.1.2: Hotel Subsystem Context



Hình 1.1.3: Payment Subsystem Context



Hình 1.1.4: MapAPI System Context



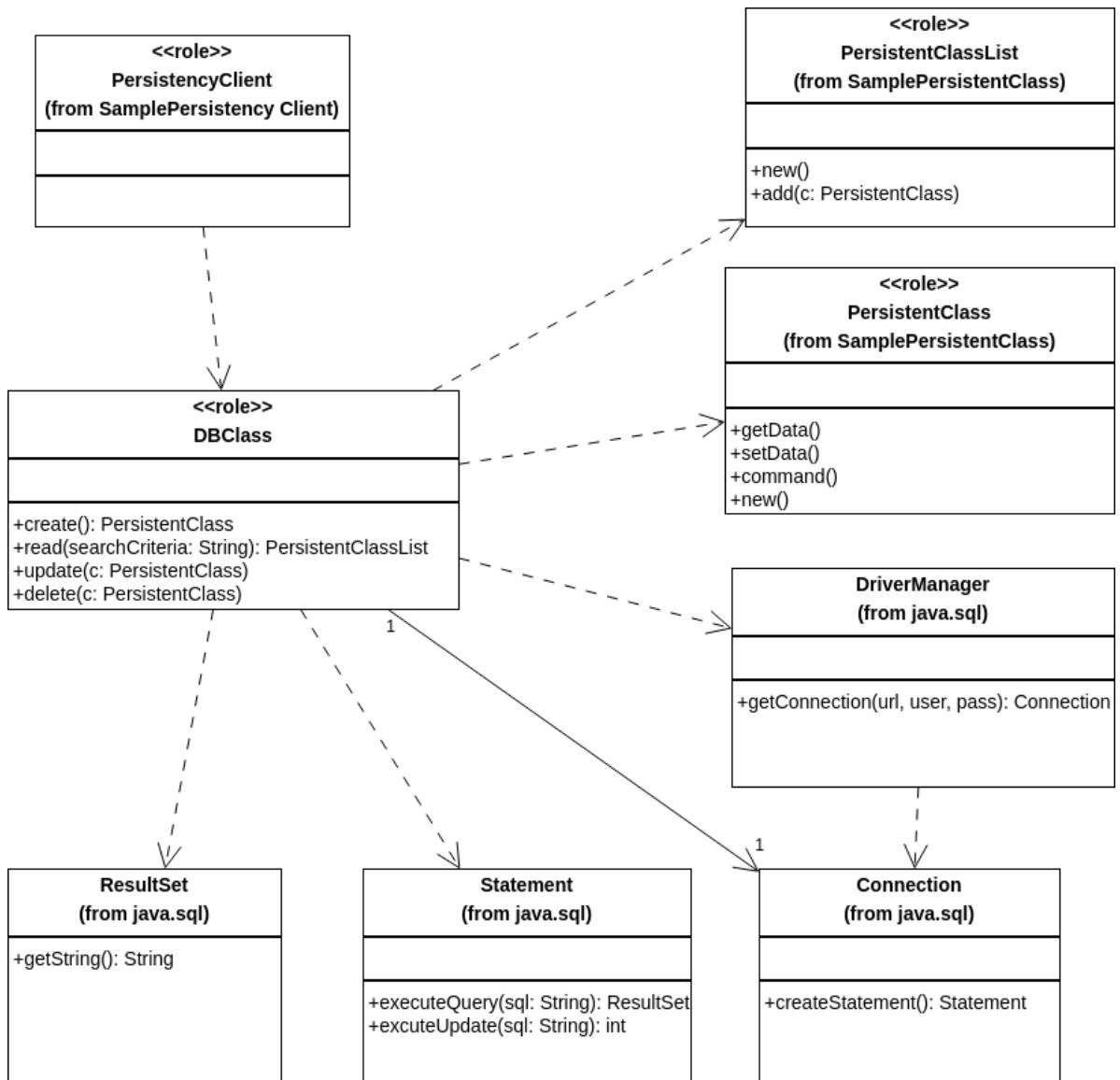
Hình 1.1.5: Authentication System Context

1.2 Analysis-to-Design-to-Implementation Mechanisms Map

Cơ chế phân tích	Cơ chế thiết kế	Cơ chế cài đặt
Persistency	OODBMS (new data)	ObjectStore
Persistency	RDBMS (data from legacy database)	JDBC to Ingres
Distribution	Remote Method Invocation (RMI)	Java
Security		Reverse Engineered Secure.java and UserContextRemoteObject components
Error detection/handling/reporting		

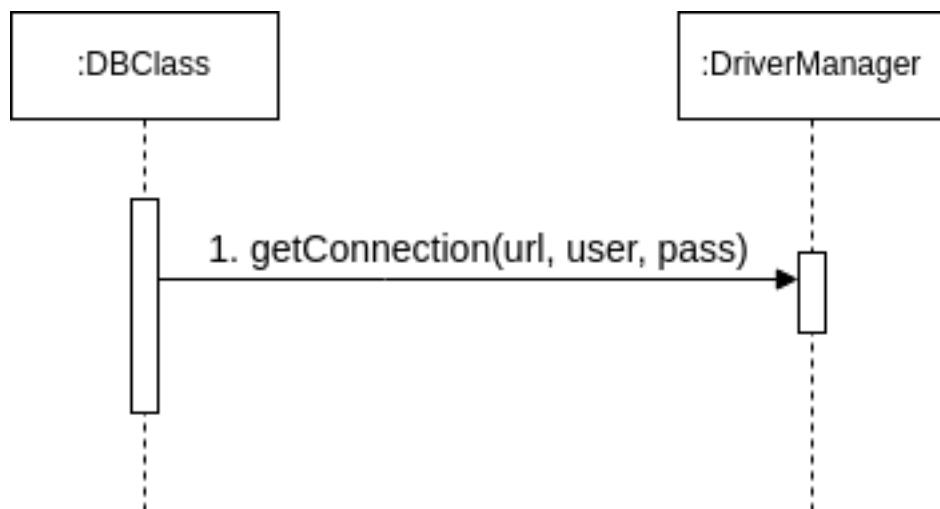
1.2.1 Cơ chế Persistency - ObjectStore OODBMS

Static View



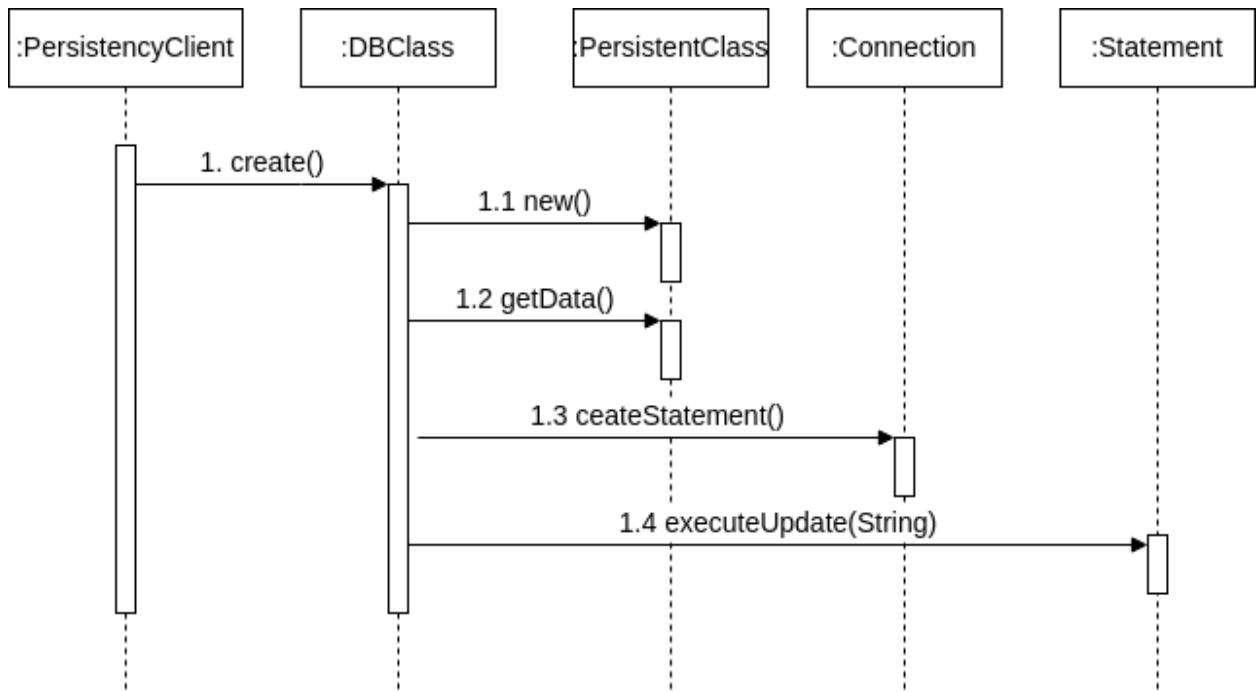
- Trong JDBC, một máy khách sẽ làm việc với **DBClass** để đọc và ghi dữ liệu. DBClass chịu trách nhiệm truy cập cơ sở dữ liệu JDBC bằng lớp **DriverManager**. Khi một Kết nối cơ sở dữ liệu được mở, DBClass sau đó có thể tạo các câu lệnh SQL sẽ được gửi đến RDBMS cơ bản và được thực thi bằng lớp **Statement**. Lớp Statement là lớp "nói chuyện" với cơ sở dữ liệu. Kết quả của truy vấn SQL được trả về trong một đối tượng **ResultSet**.
- Lớp **DBClass** chịu trách nhiệm tạo ra instance của một lớp. Nó có cơ chế ánh xạ OO-to-RDBMS và có hành vi để giao tiếp với RDBMS. DBClass làm phẳng đối tượng, ghi nó vào RDBMS và đọc dữ liệu đối tượng từ RDBMS và xây dựng đối tượng đó. Mỗi lớp persistency sẽ có một DBClass tương ứng.
- PersistentclassList** được sử dụng để trả về một tập hợp các đối tượng persistency là kết quả của truy vấn cơ sở dữ liệu.

Dynamic View



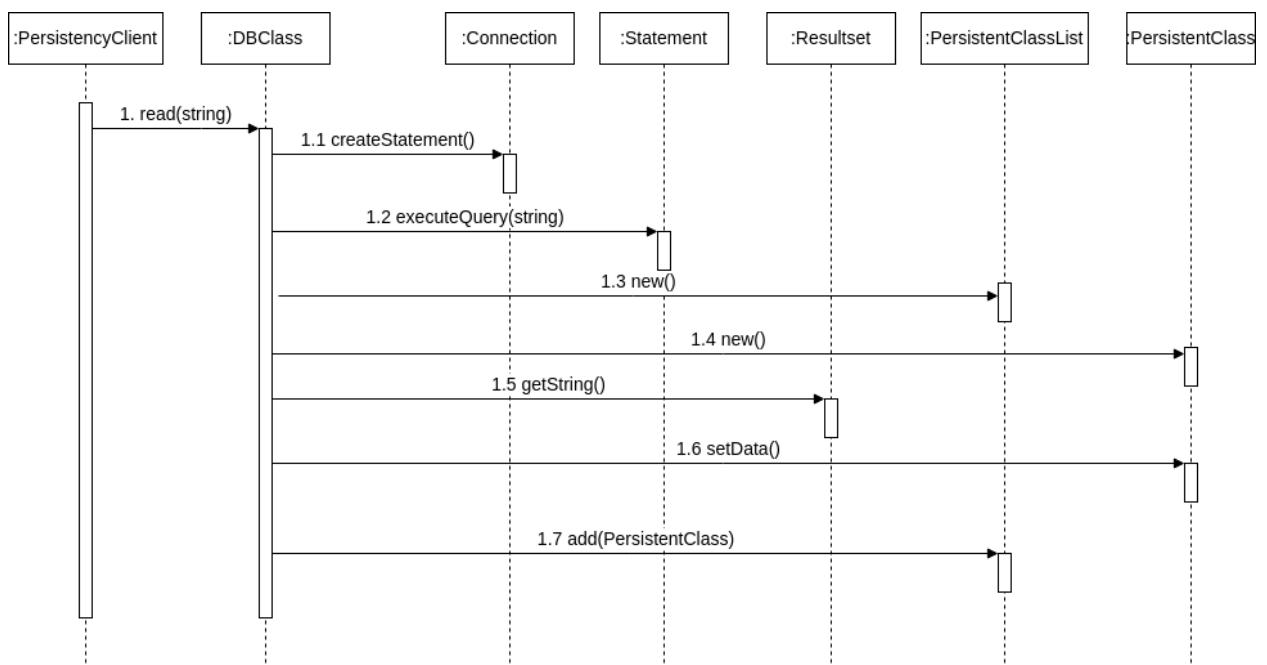
Hình 1.2.1: Initialize

Bước khởi tạo phải có trước khi các lớp khác có thể kết nối. Để khởi tạo kết nối tới cơ sở dữ liệu, DBClass phải gọi hàm `getConnection()`. Và phương thức `getConnection()` sẽ tạo một kết nối với URL đã nhập sau đó trả về kết nối này.



Hình 1.2.2: Create

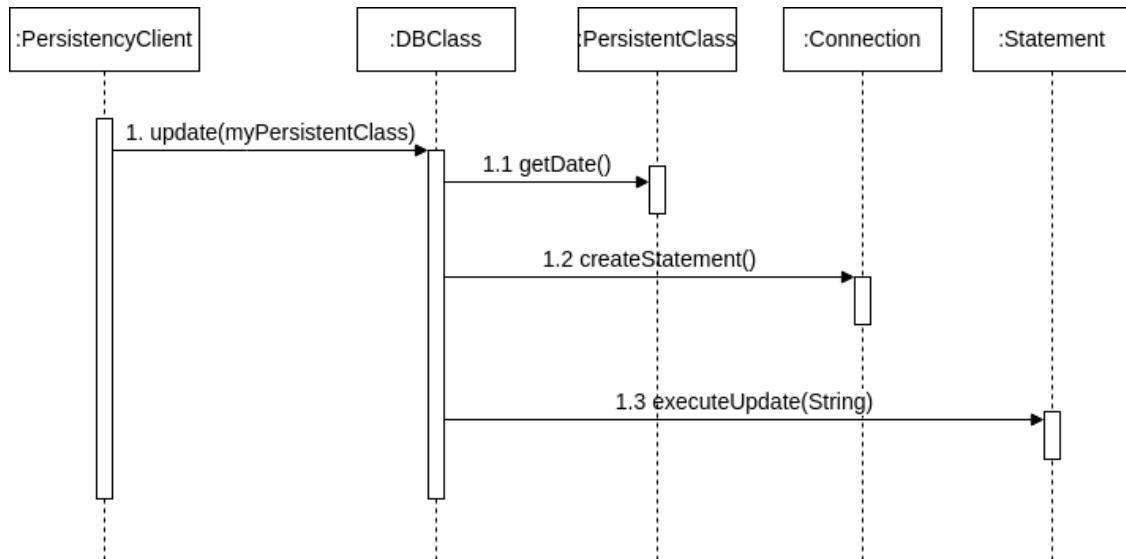
Để tạo một lớp mới, DBClass tạo một thể hiện mới của PersistentClass với các giá trị mặc định. Sau đó, DBClass tạo một Statement mới bằng cách sử dụng phương thức createStatement() của lớp Connection. Statement được thực thi và dữ liệu được chèn vào cơ sở dữ liệu.



Hình 1.2.3: Read

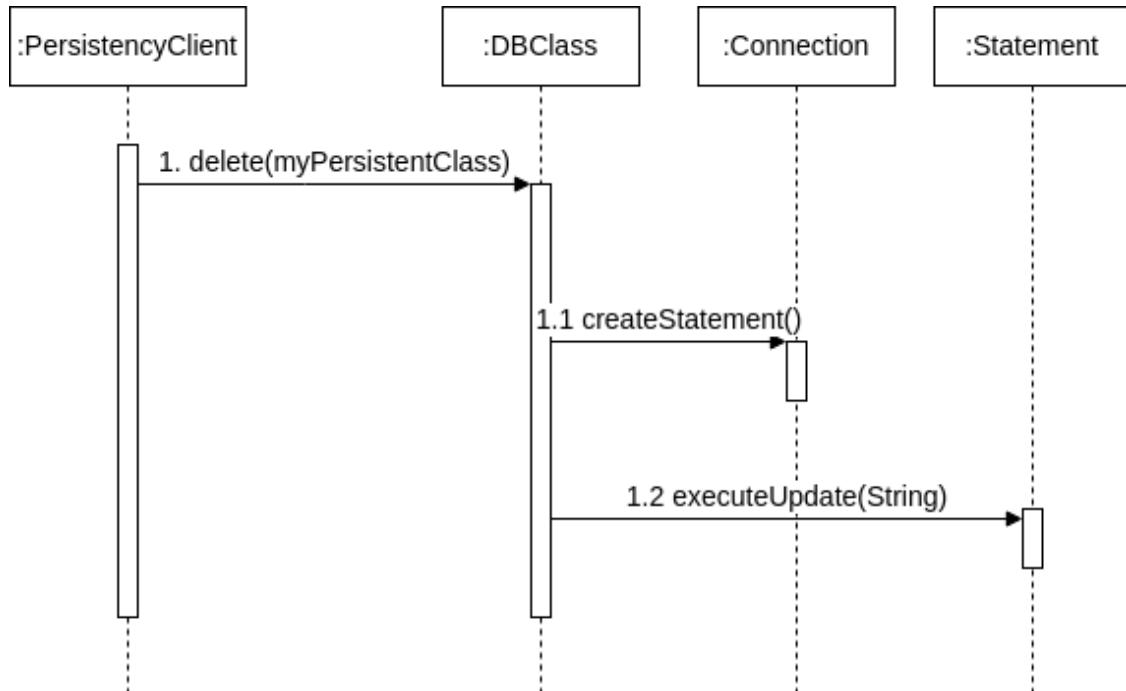
Để đọc dữ liệu, DBClass tạo một Statement mới bằng cách sử dụng createStatement() của lớp Connection. Statement được thực thi và dữ liệu được trả về trong một đối tượng

ResultSet. Sau đó, DBClass tạo một thể hiện mới của PersistentClass và điền vào đó dữ liệu đã truy xuất. Dữ liệu được trả về trong một đối tượng collection, một thể hiện của lớp PersistentClassList.



Hình 1.2.4: Update

Để cập nhật dữ liệu, DBClass lấy dữ liệu từ đối tượng PersistentClass đã cho và tạo một Statement mới bằng cách sử dụng phương thức createStatement() của lớp Connection. Sau khi Statement được xây dựng, bản cập nhật được thực thi và cơ sở dữ liệu được cập nhật với dữ liệu mới.

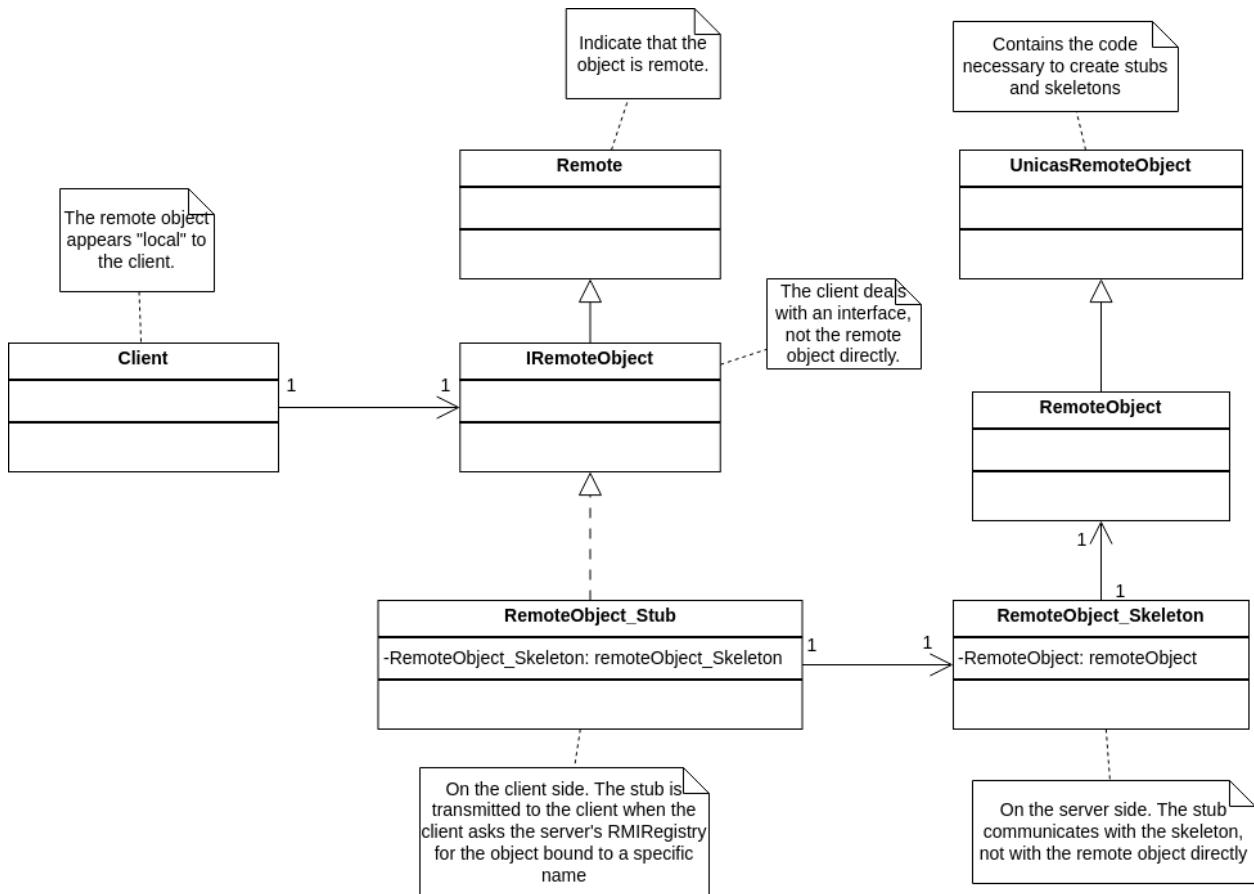


Hình 1.2.5: Delete

Để xóa dữ liệu, DBClass tạo một Statement mới bằng cách sử dụng phương thức createStatement() của lớp Connection. Statement được thực thi và dữ liệu được xóa khỏi cơ sở dữ liệu.

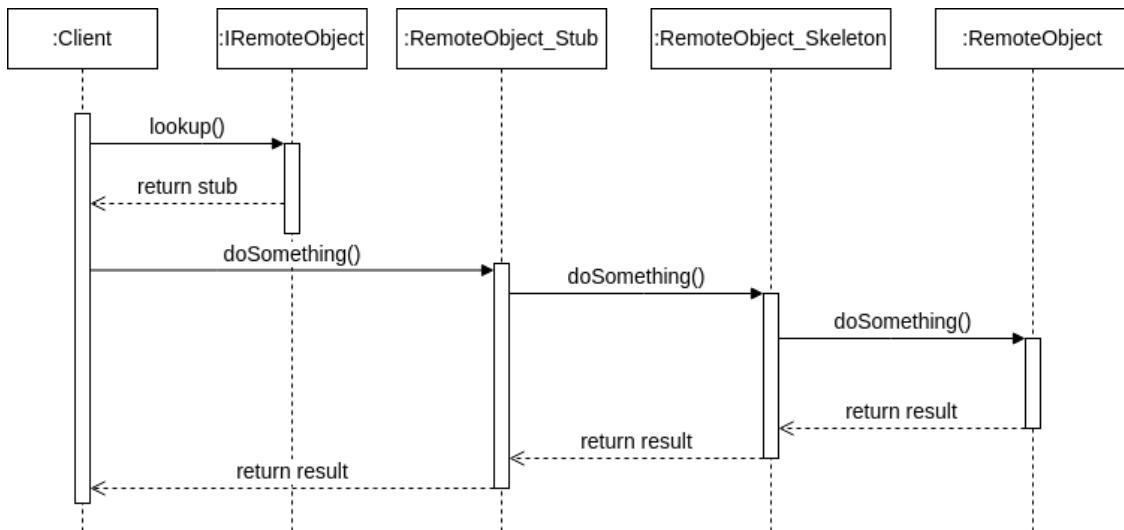
1.2.2 Cơ chế Distribution

Static View



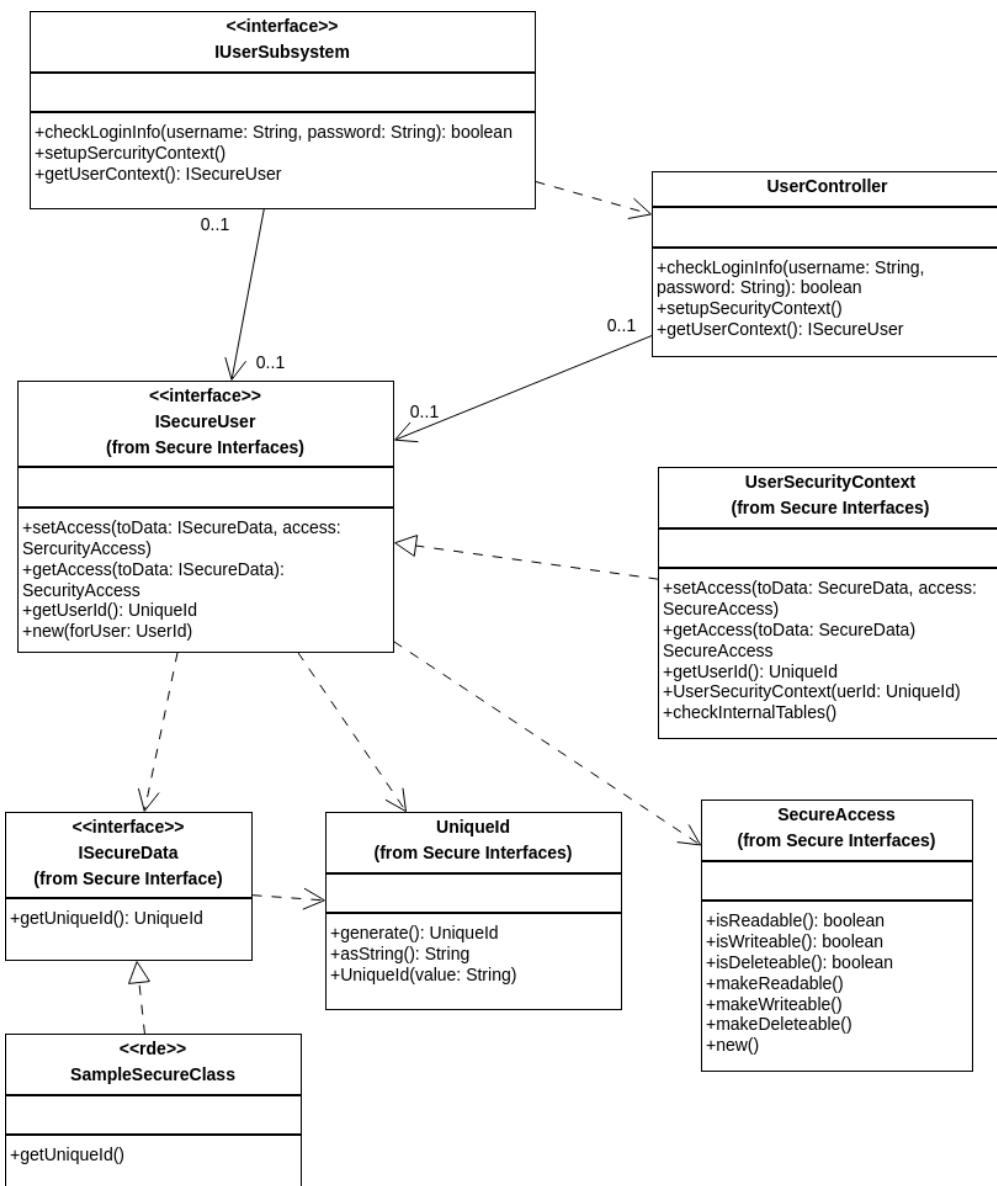
- Cả client và server đều là hệ thống hướng đối tượng. Tương tác giữa server và client được đóng gói trong một hoặc nhiều đối tượng có thể truy cập từ xa. Các đối tượng này được khởi tạo trên server, nhưng client xử lý đối tượng như thể chúng là các khởi tạo cục bộ.
- Các stub và skeleton là các adapter, đầu tiên giúp client thích ứng với Internet và sau đó là Internet thích ứng với đối tượng của server.
- Bất kỳ đối tượng có thể tuần tự hóa nào đều có thể được truyền đến máy chủ bằng cách sử dụng tham số đầu vào nếu là phương thức của giao diện IRemoteObject.
- Bất kỳ phương thức nào mà client chạy trên đối tượng từ xa đều thực sự chạy trên server.
- Bất kỳ phương thức nào của đối tượng không phải UnicastRemoteObject có thể tuần tự hóa được đặt cho client đều chạy trên client.
- Lớp server phải chạy để RMIREGISTRY có thể tải các stub ban đầu để chuẩn bị truyền chúng tới client.

Dynamic View



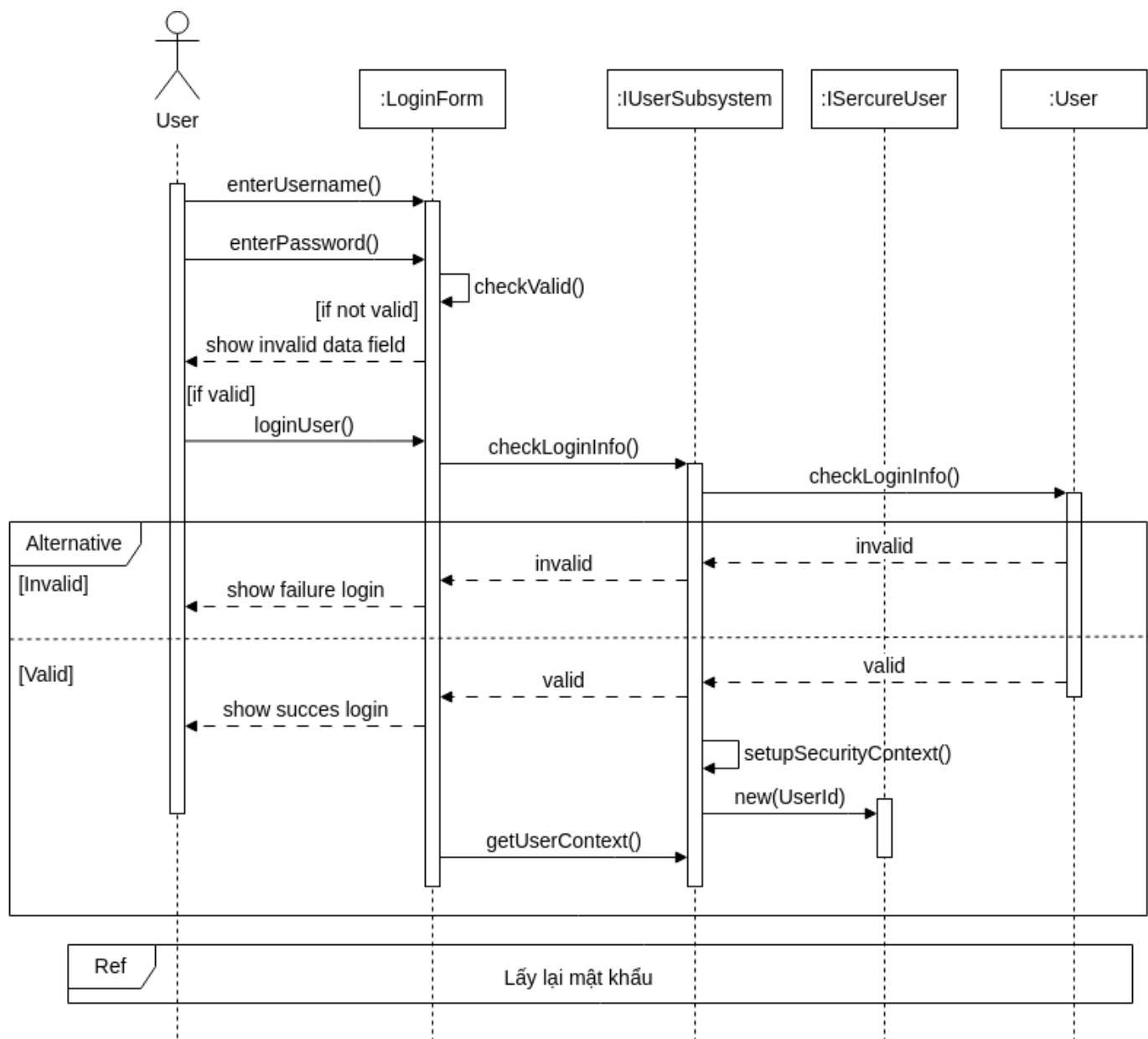
1.2.3 Cơ chế Security

Static View



- ISecureData: cơ chế phân tích: security
- SecurityAccess: cơ chế phân tích: security
- UserSecurityContext: cơ chế phân tích: security
- UniqueId: cơ chế phân tích: security
- IUserSubsystem: cơ chế phân tích: security
- ISecureUser: cơ chế phân tích: security
- UserController: cơ chế phân tích: security

Dynamic View



1.3 Analysis-Class-to-Design-Element Map

Analysis Class	Design Elements
RoomSearchView	RoomSearchForm
RoomSearchController	RoomService
Room	Room
PaymentGateway	PaymentService
NotificationSystem	NotificationService
BookingHistoryView	BookingHistoryView
BookingController	BookingController, BookingRepository
Booking	Booking
BookingListView	BookingListView
ReviewView	ReviewView
ReviewController	ReviewController, ReviewRepository
Review	Review
RoomDetailView	RoomDetailView
ChatController	ChatController, ChatRepository
ChatInterface	ChatInterface
Message	Message, MessageService
Host	Host
FavoriteRoomView	FavoriteRoomView
FavoriteController	FavoriteController, FavoriteRepository
Room	Room
UserController	UserController
LeaseHotelView	LeaseHotelView
AuthenticationSystem	AuthenticationSystem
LeaseHotelController	LeaseHotelController
HotelDatabase	HotelDatabase
HotelView	HotelView
HotelController	HotelController, IHotelSubSystem
SearchRentalHotelView	SearchRentalHotelView
SearchRentalHotelController	SearchRentalHotelController
PaymentView	PaymentView
PaymentController	PaymentController
Payment	Payment
PaymentSystem	IPaymentSystem
OrderView	OrderView
OrderController	OrderController, IOrderSystem
Order	Order
FindVehicleView	FindVehicleView
FindVehicleController	FindVehicleController
LeaseHotelController	LeaseHotelController
AuthenticationSystem	IAuthentifacationSubSystem
MapAPISystem	ImapAPI
RegisterForm	RegisterForm
UserDatabase	UserDatabase
User	User, IUserSubsystem
LoginForm	LoginForm

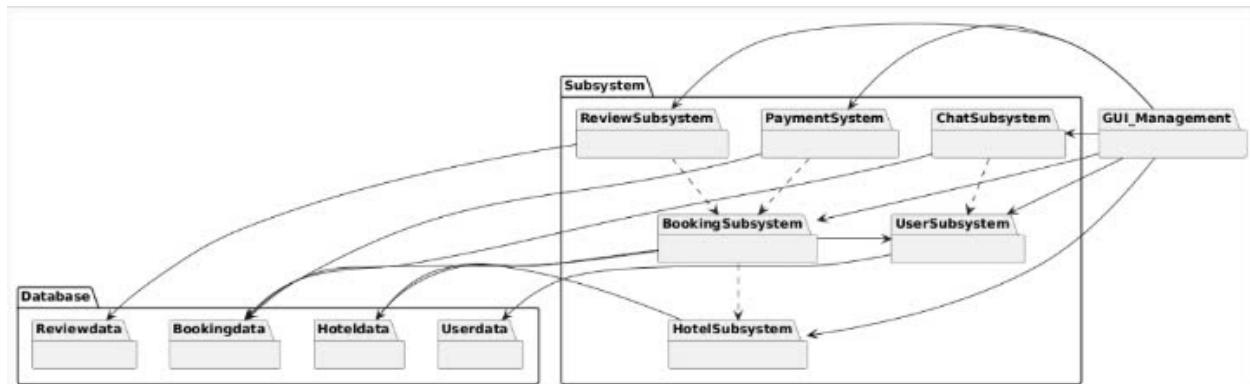
NewPasswordForm	NewPasswordForm
ChangeInfoForm	ChangeInfoForm
DeleteAccountForm	DeleteAccountForm
LanguageView	LanguageView
LanguageController	LanguageController

1.4 Design-Element-to-Owning-Package Map

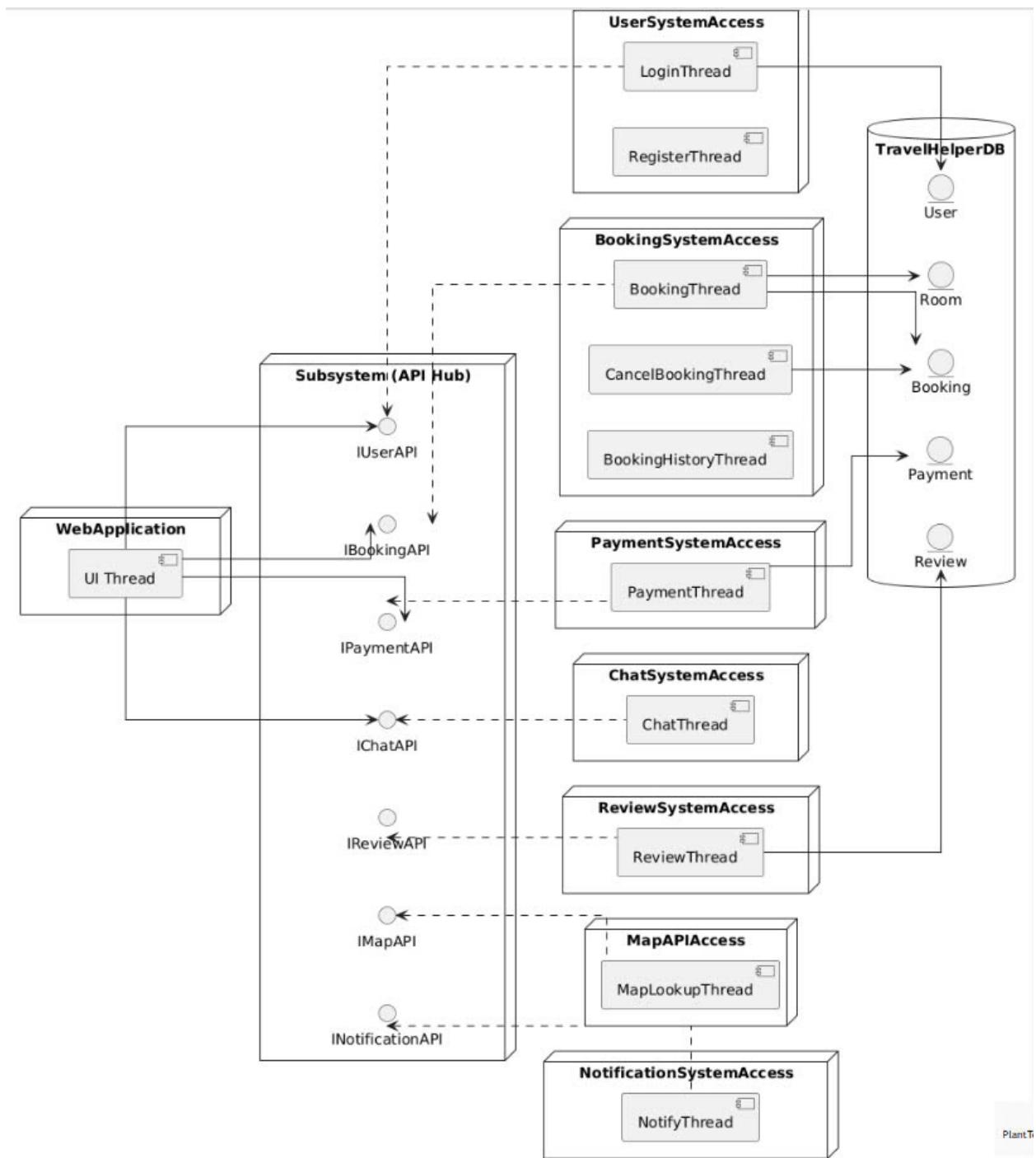
Design Elements Owni	Owning package
RoomSearchForm	GUI_Management
RoomService	Subsystem.HotelSubsystem
Room	Database.Hotelinfo
PaymentService	Subsystem.PaymentSystem
NotificationService	Subsystem
BookingHistoryView	GUI_Management
BookingController	Subsystem.BookingSubsystem
BookingRepository	Subsystem.BookingSubsystem
Booking	Database.Bookingdata
BookingListView	GUI_Management
ReviewView	GUI_Management
ReviewController	Database.ReviewSubsystem
ReviewRepository	Subsystem.ReviewSubsystem
Review	Database.Reviewdata
RoomDetailView	GUI_Management
ChatController	Subsystem.ChatSubsystem
ChatRepository	Subsystem.ChatSubsystem
ChatInterface	Subsystem.ChatSubsystem
Message	Database.Bookingdata
MessageService	Subsystem.ChatSubsystem
Host	Database.Userdata
FavoriteRoomView	GUI_Management
FavoriteController	Subsystem.BookingSubsystem
FavorieRepository	Subsystem.BookingSubsystem
Room	Database.hotldata
UserController	Subsystem.UserSubsystem
LeaseHotelView	GUI_Management
AuthenticationSystem	Subsystem
LeaseHotelController	Subsystem.BookingSubsystem
HotelDatabase	Database.Hotldata
HotelView	GUI_Management
HotelController	Subsystem.HotelSubsystem
IHotelSubSystem	Subsystem.HotelSubsystem
SearchRentalHotelView	GUI_Management
SearchRentalHotelController	Subsystem.BookingSubsystem
PaymentView	GUI_Management
PaymentController	Subsystem.PaymentSystem

Payment	Database.Bookingdata
PaymentSystem	Subsystem.PaymentSystem
OrderView	GUI_Management
OrderController	Subsystem.BookingSubsystem
IOrderSystem	Subsystem.BookingSubsystem
Order	Database.Bookingdata
FindVehicleView	GUI_Management
FindVehicleController	Subsystem.BookingSubsystem
LeaseHotelController	Subsystem.BookingSubsystem
AuthenticationSystem	Subsystem
MapAPISystem	GUI_Management
RegisterForm	GUI_Management
UserDatabase	Database.Userdata
User	Database.Userdata
IUserSubsystem	Subsystem.UserSubsystem
LoginForm	GUI_Management
NewPasswordForm	GUI_Management
ChangeInfoForm	GUI_Management
DeleteAccountForm	GUI_Management
LanguageView	GUI_Management
LanguageController	Subsystem

1.5 Packages and Their Dependency



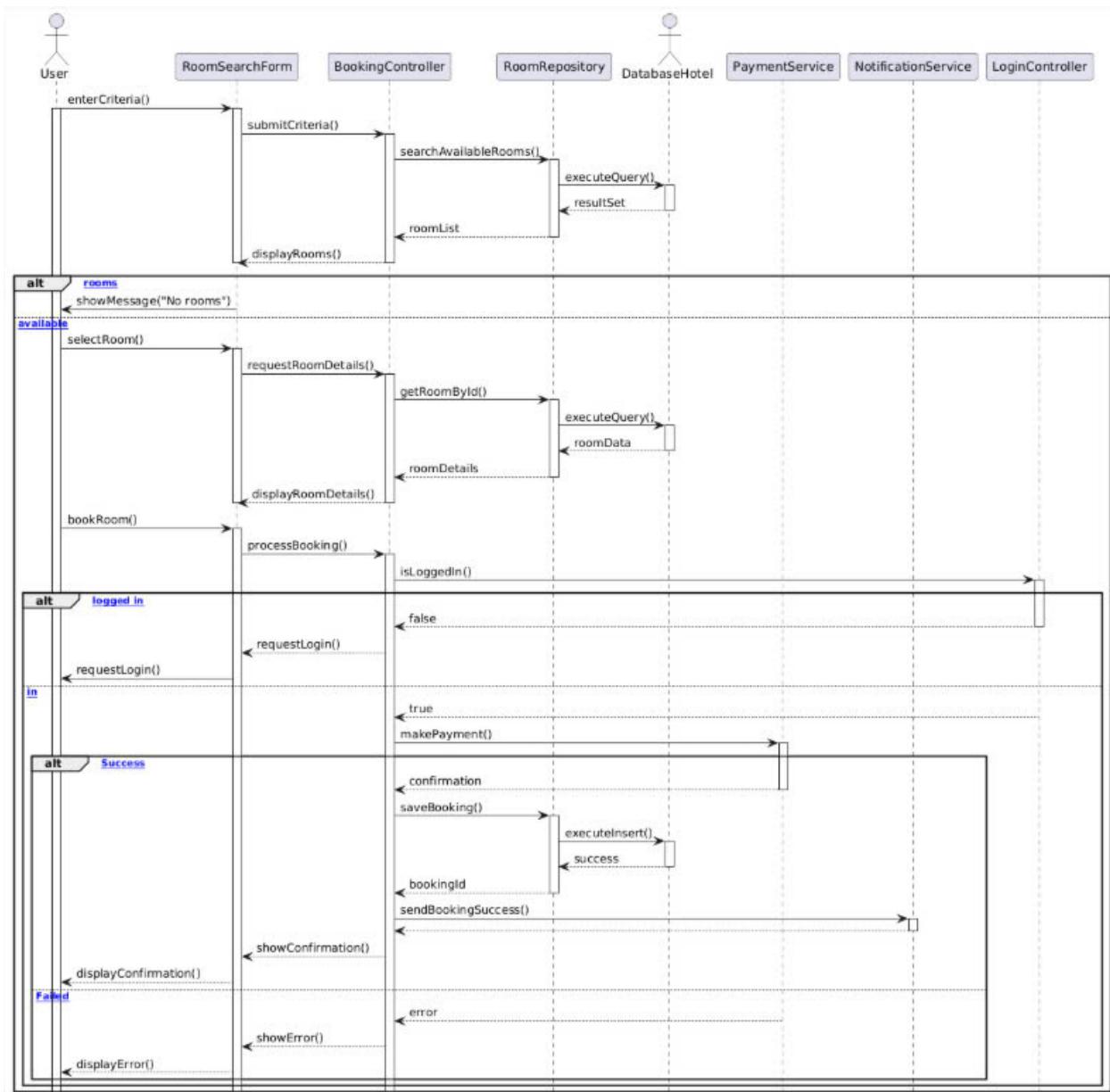
2 Mô tả kiến trúc thực thi



3 Thiết kế Use Case

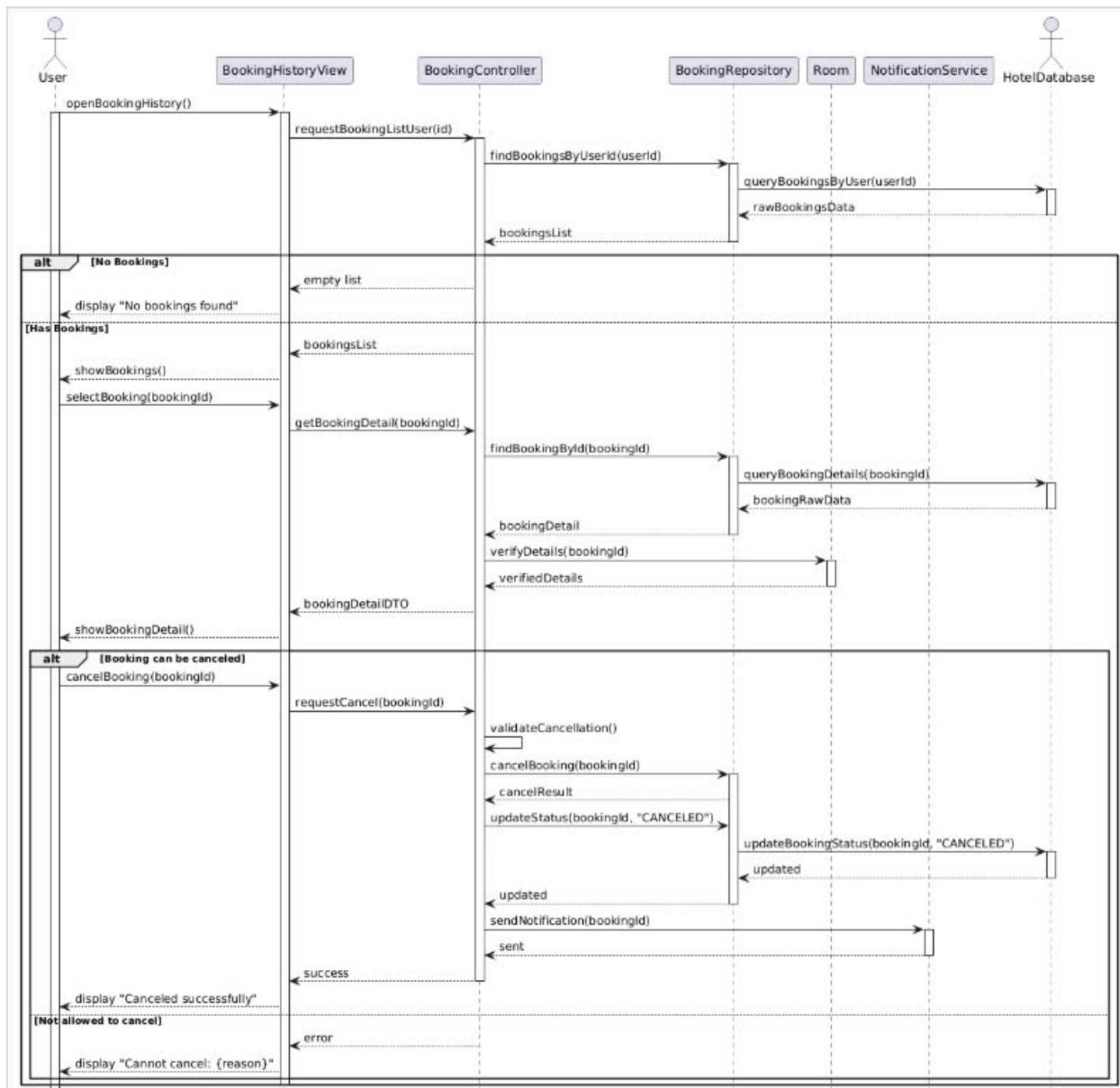
3.1 Thiết kế biểu đồ tuần tự

3.1.1 Đặt phòng khách sạn



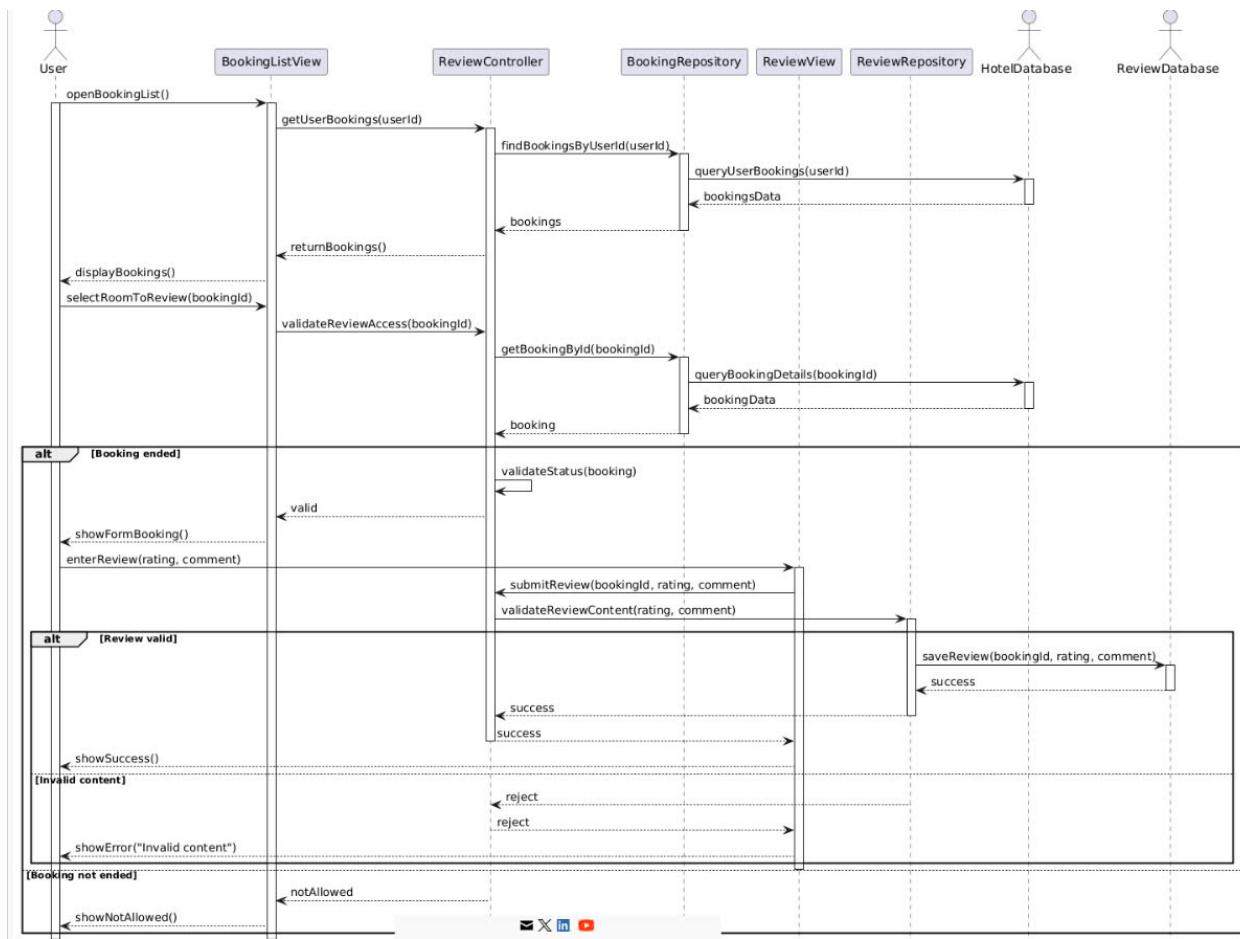
Hình 3.1.1: Biểu đồ tuần tự Đặt phòng khách sạn

3.1.2 Theo dõi phòng đã đặt



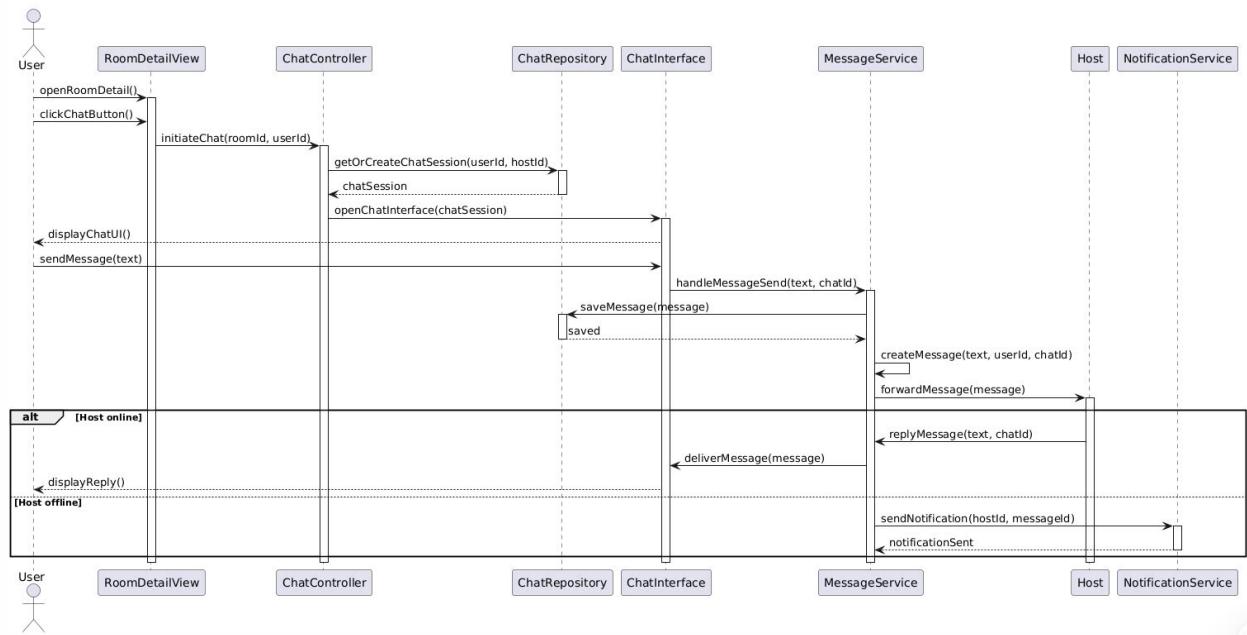
Hình 3.1.2: Biểu đồ tuần tự Theo dõi phòng đã đặt

3.1.3 Đánh giá phòng đã thuê



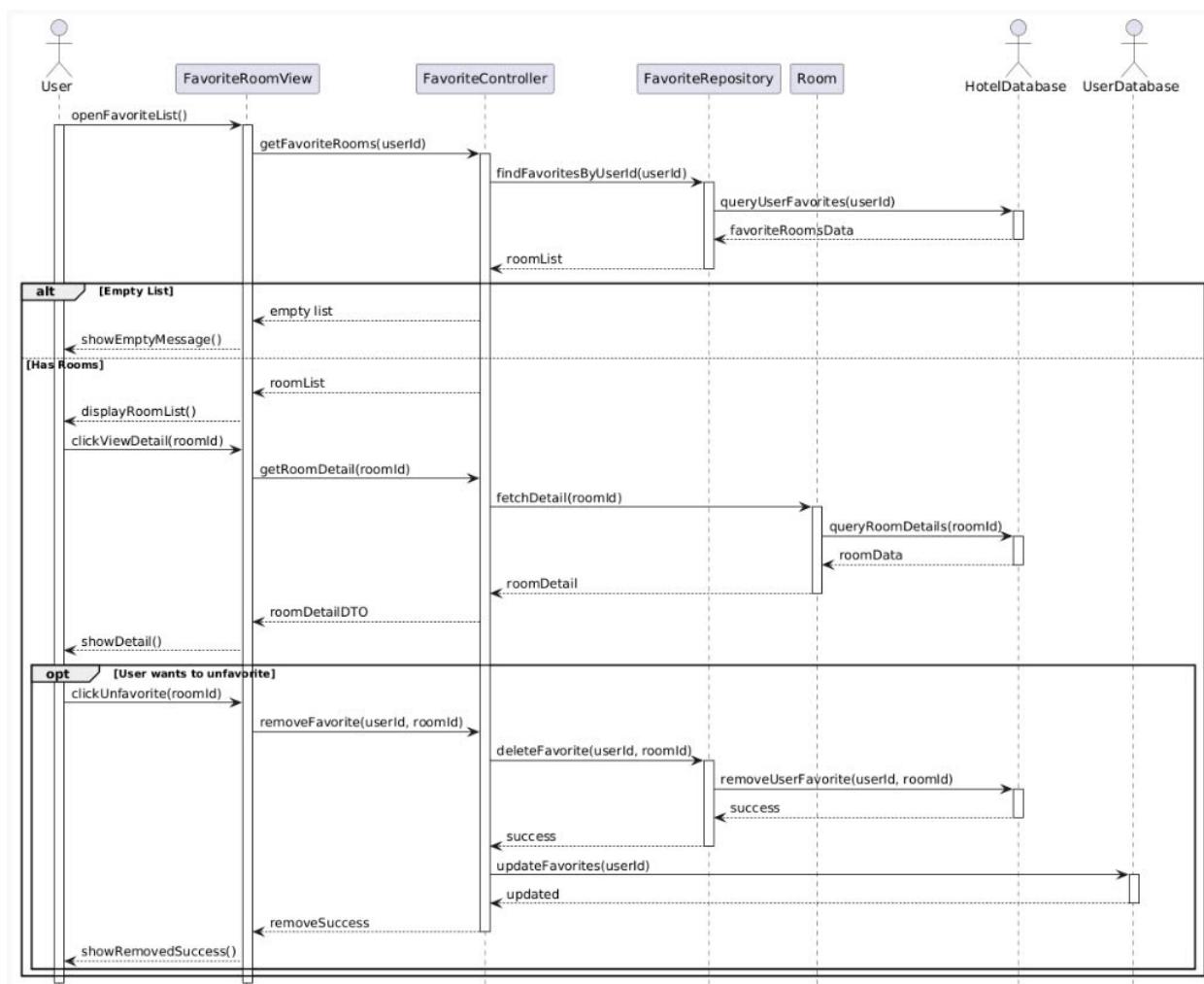
Hình 3.1.3: Biểu đồ tuần tự Đánh giá phòng đã thuê

3.1.4 Liên hệ với chủ khách sạn



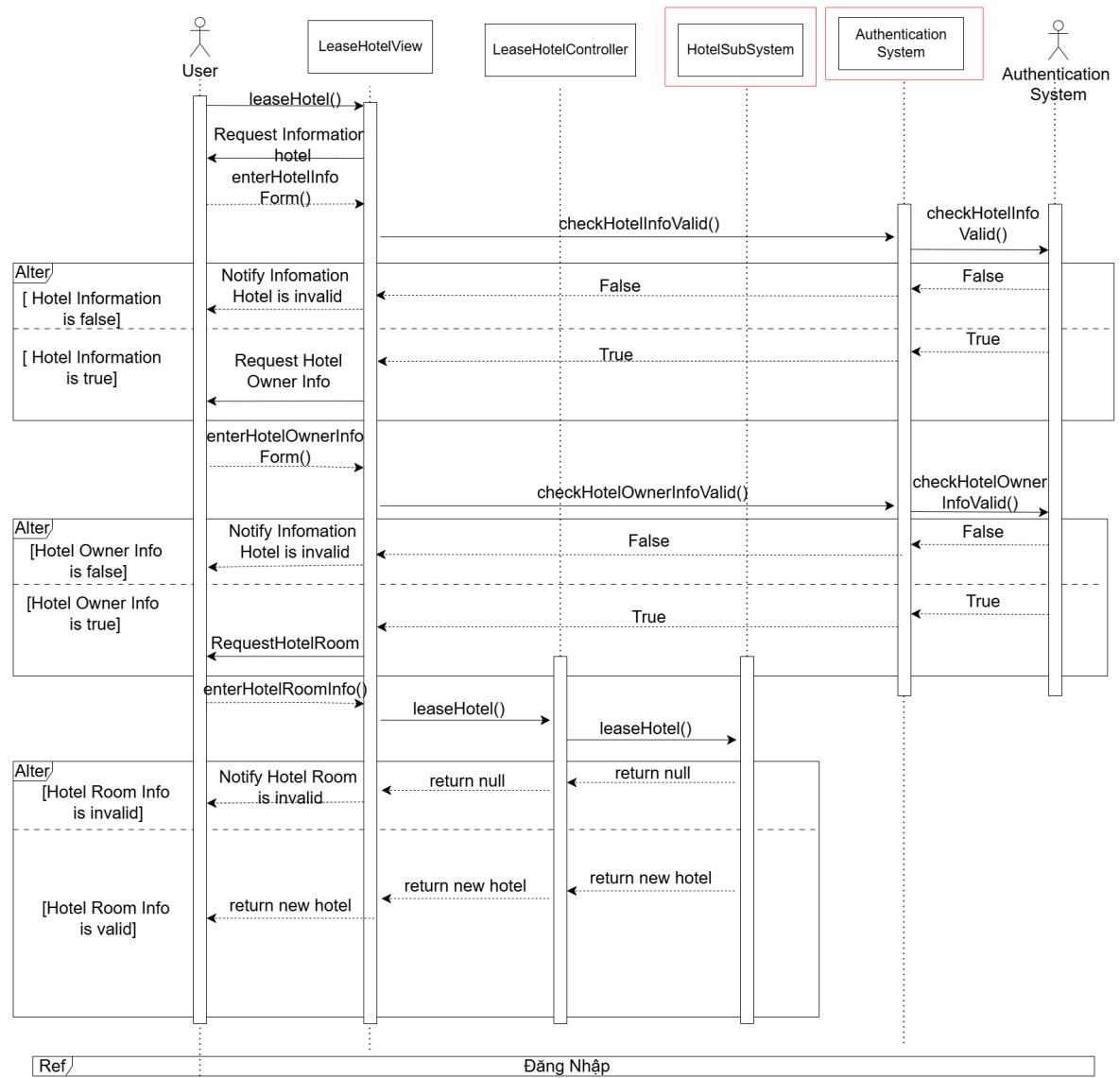
Hình 3.1.4: Biểu đồ tuần tự Liên hệ với chủ khách sạn

3.1.5 Xem danh sách phòng yêu thích



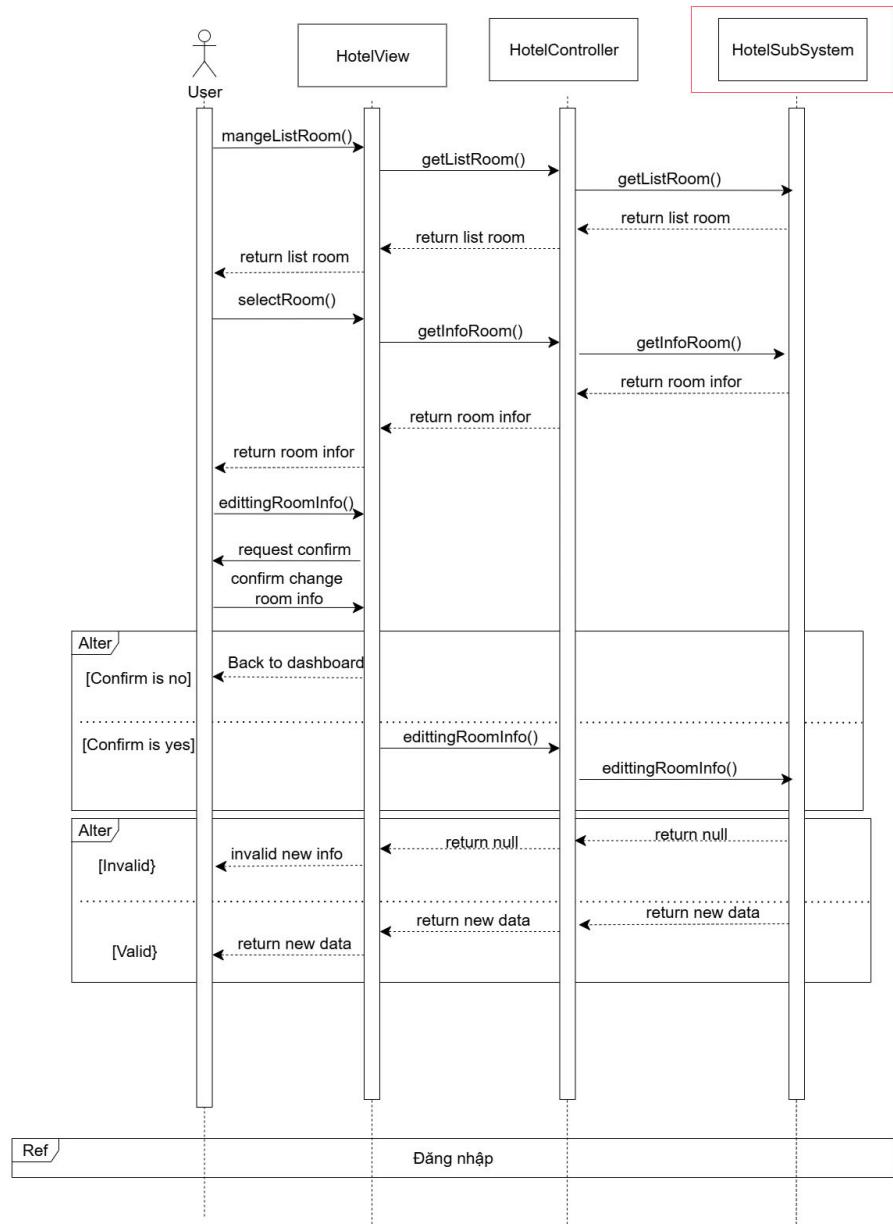
Hình 3.1.5: Biểu đồ tuần tự Xem danh sách phòng yêu thích

3.1.6 Cho thuê phòng



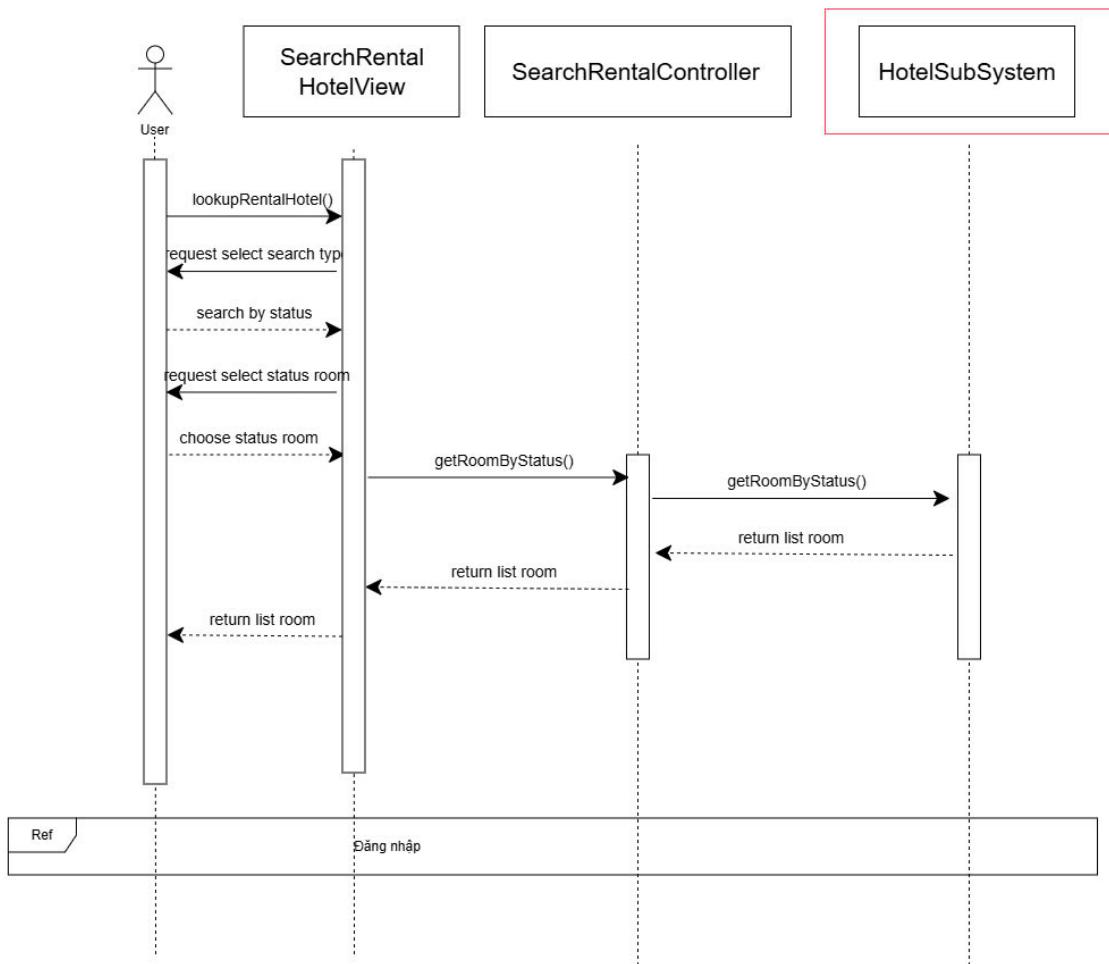
Hình 3.1.6: Biểu đồ tuần tự Cho thuê phòng

3.1.7 Sửa thông tin phòng

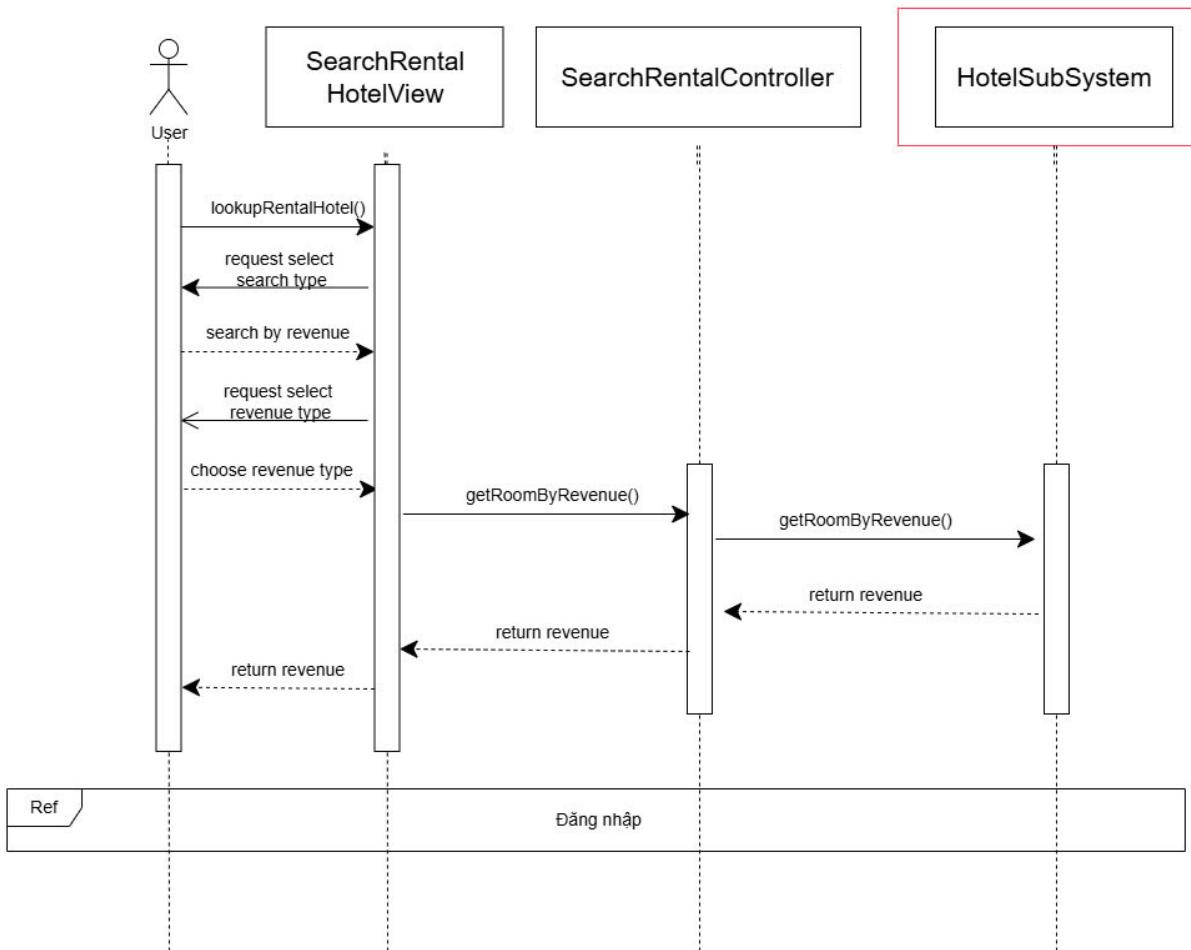


Hình 3.1.7: Biểu đồ tuần tự Sửa thông tin phòng

3.1.8 Tra cứu thông tin phòng đang cho thuê

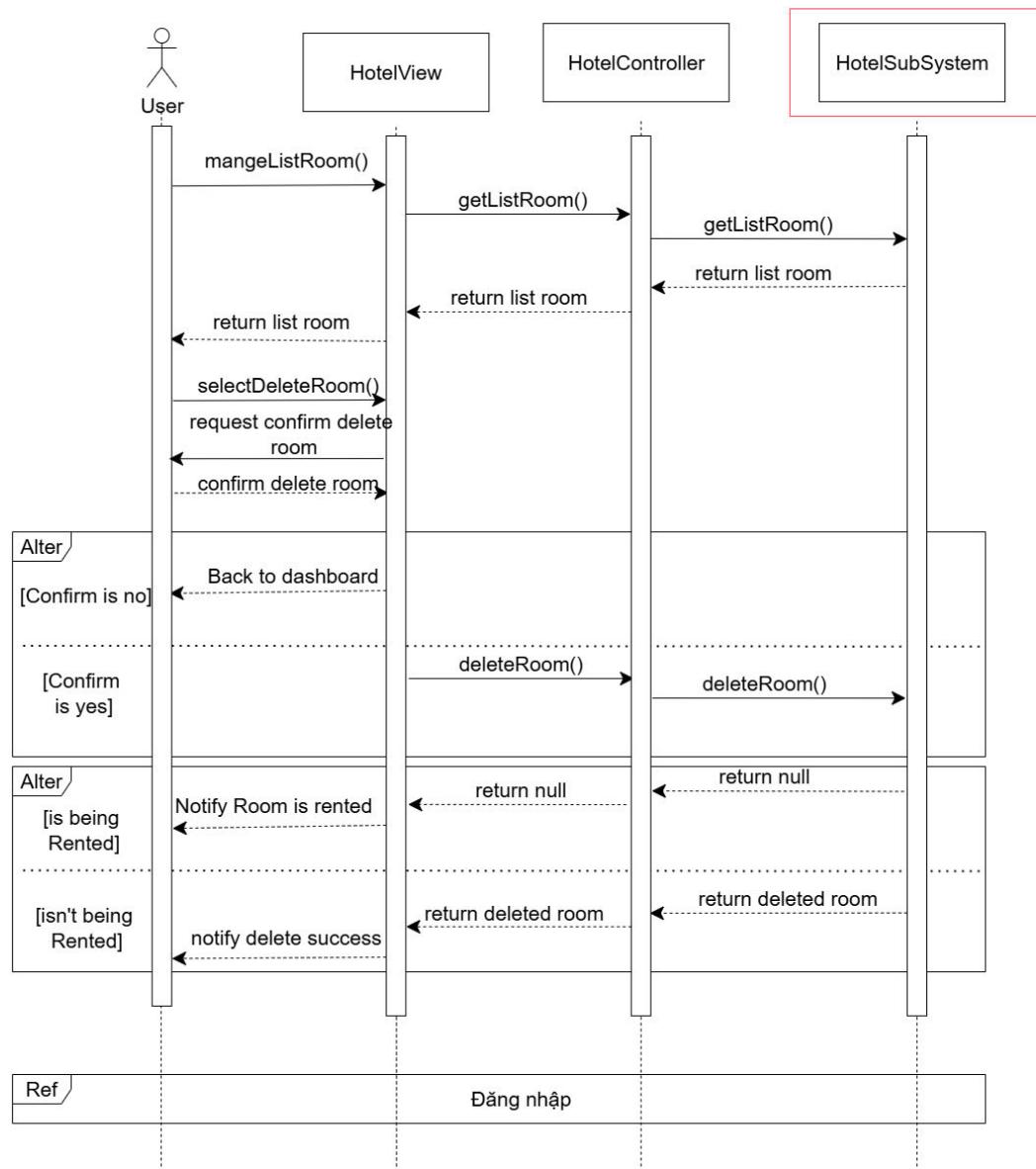


Hình 3.1.8: Biểu đồ tuần tự Tra cứu phòng theo trạng thái



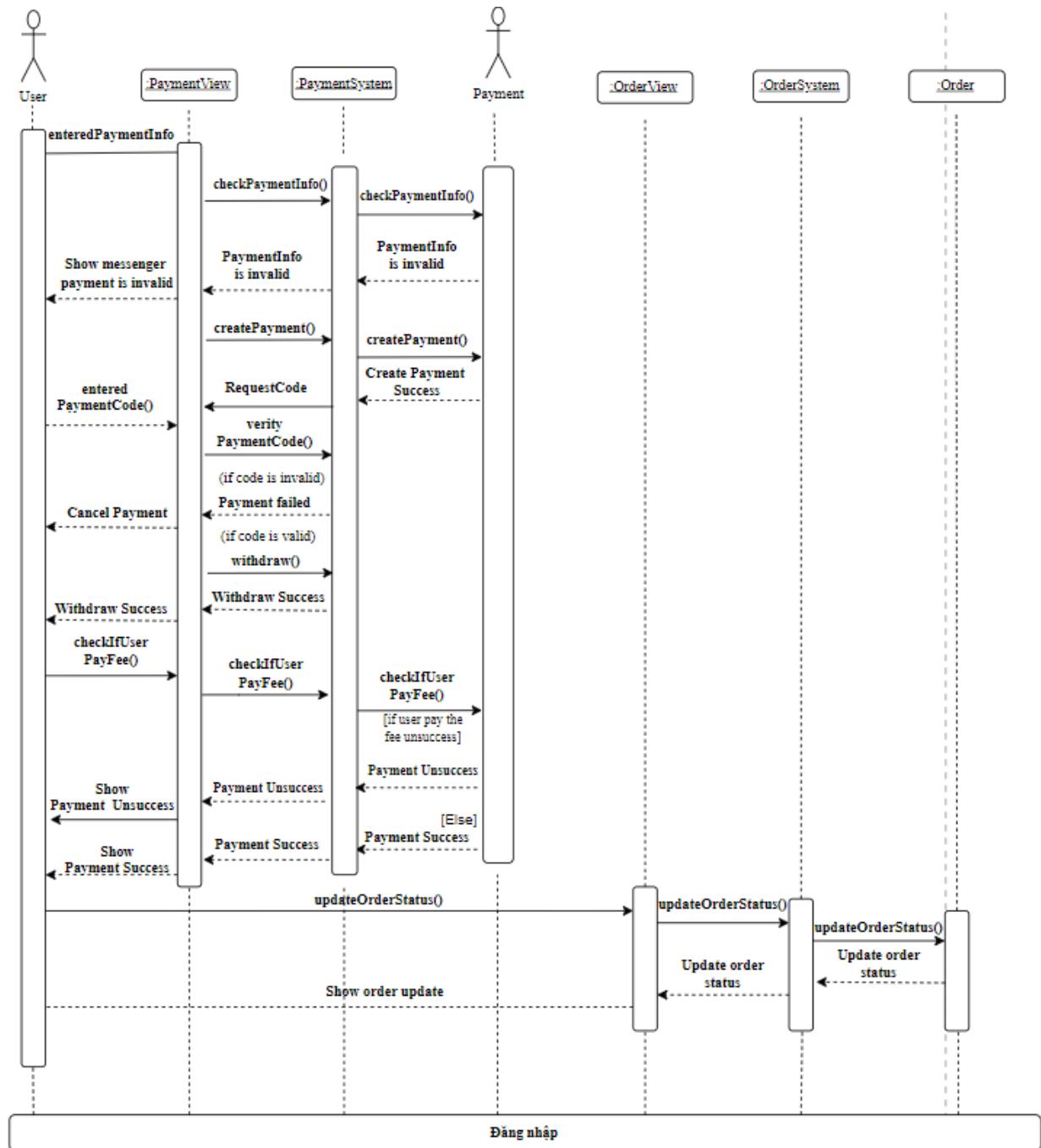
Hình 3.1.9: Biểu đồ tuần tự Tra cứu doanh thu phòng

3.1.9 Xóa phòng khách sạn

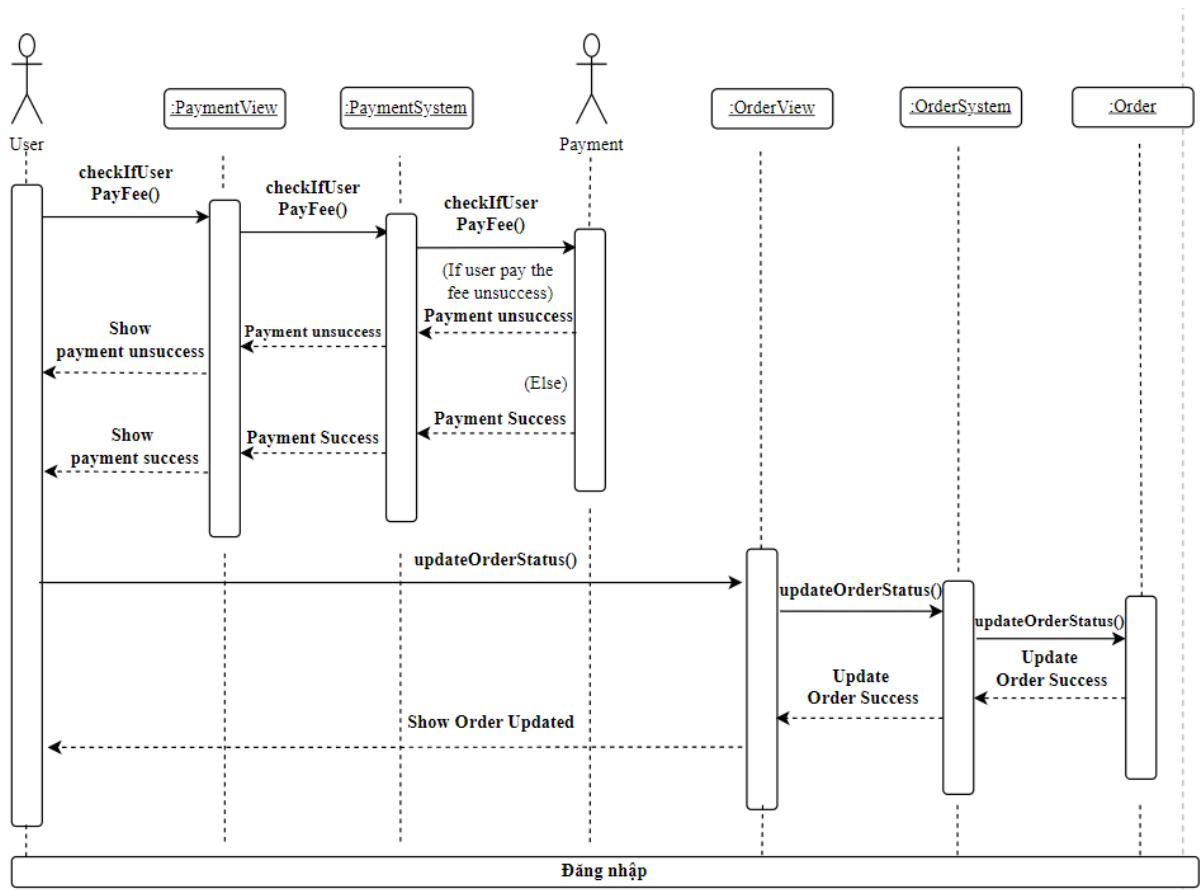


Hình 3.1.10: Biểu đồ tuần tự Xóa phòng khách sạn

3.1.10 Thanh toán

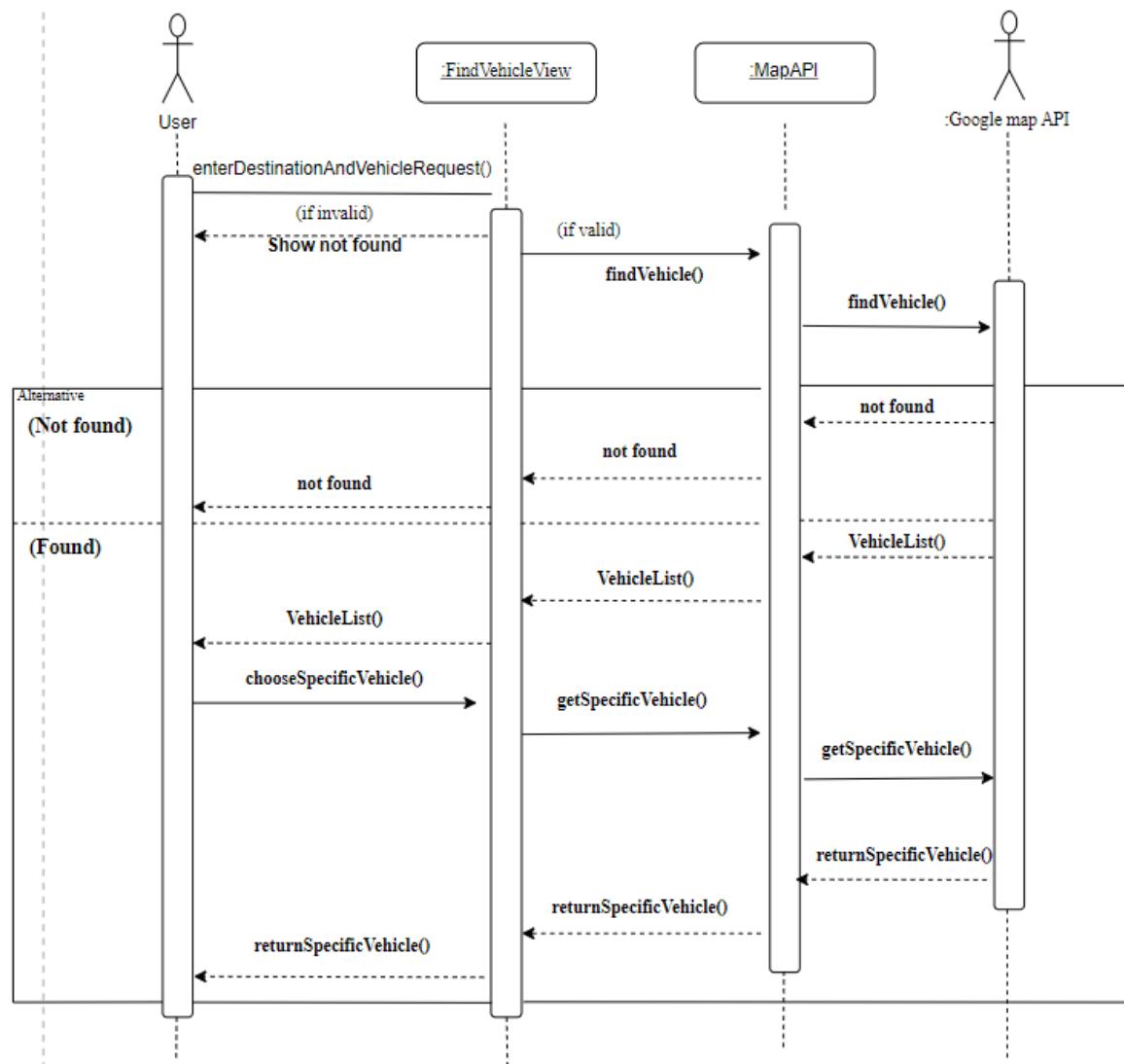


Hình 3.1.11: Biểu đồ tuần tự Thanh toán trực tuyến



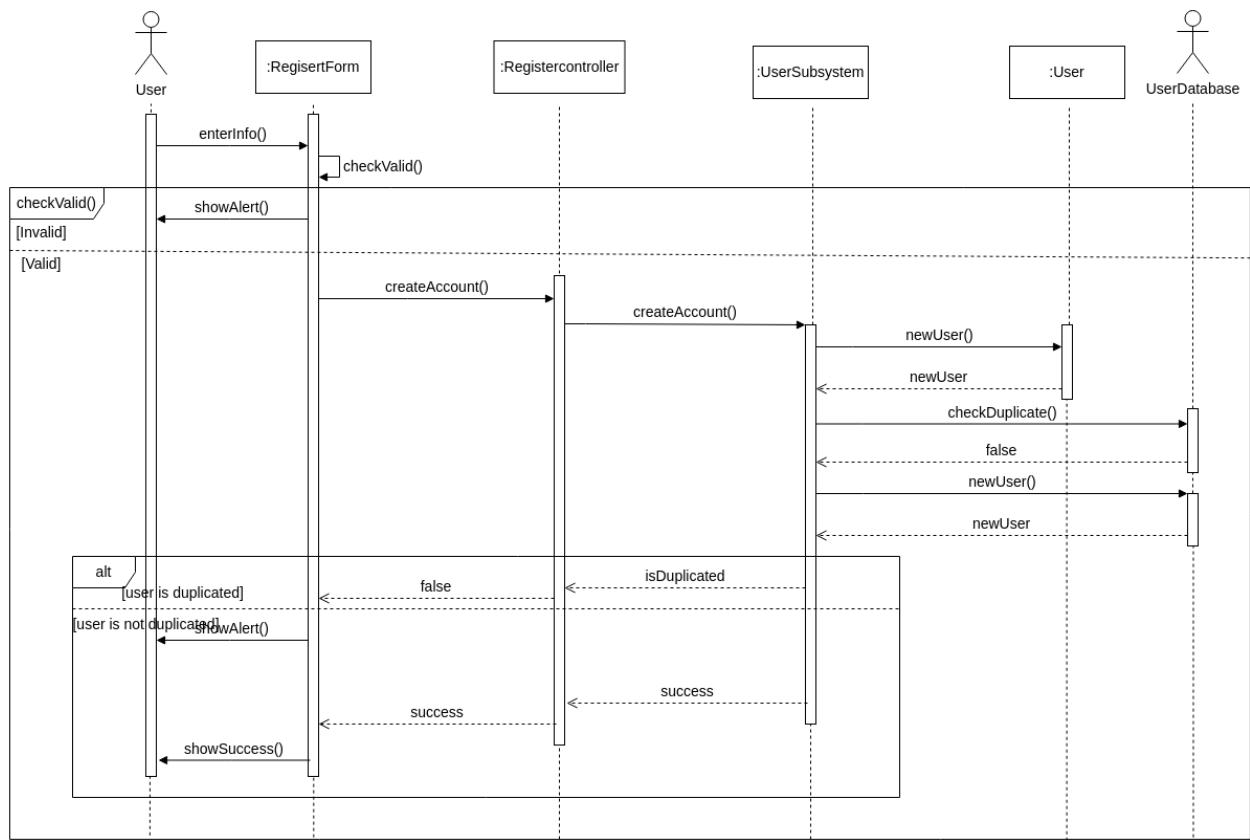
Hình 3.1.12: Biểu đồ tuần tự Thanh toán trực tiếp

3.1.11 Tra cứu phương tiện



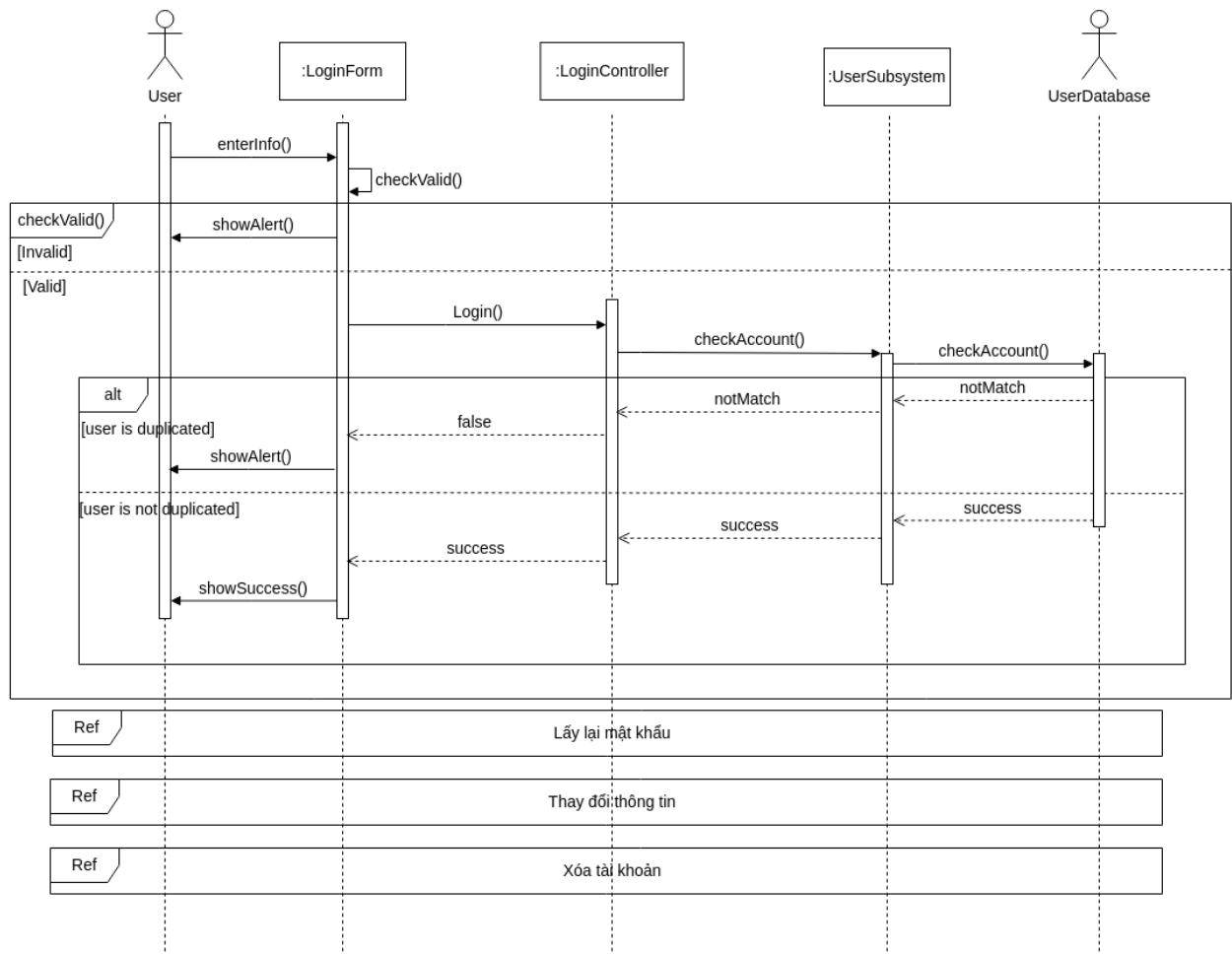
Hình 3.1.13: Biểu đồ tuần tự Tra cứu phương tiện

3.1.12 Đăng ký tài khoản

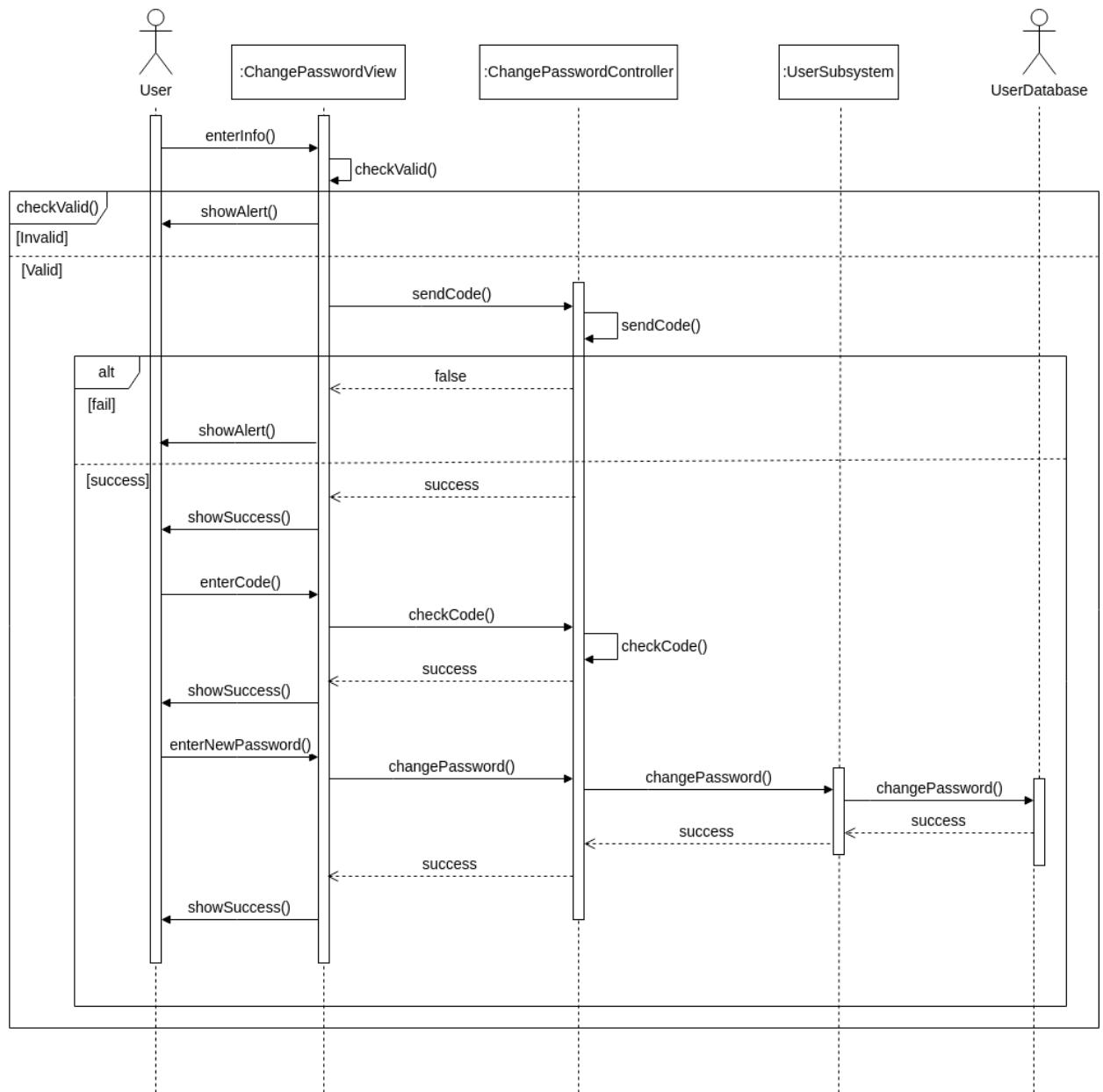


Hình 3.1.14: Biểu đồ tuần tự Đăng ký tài khoản

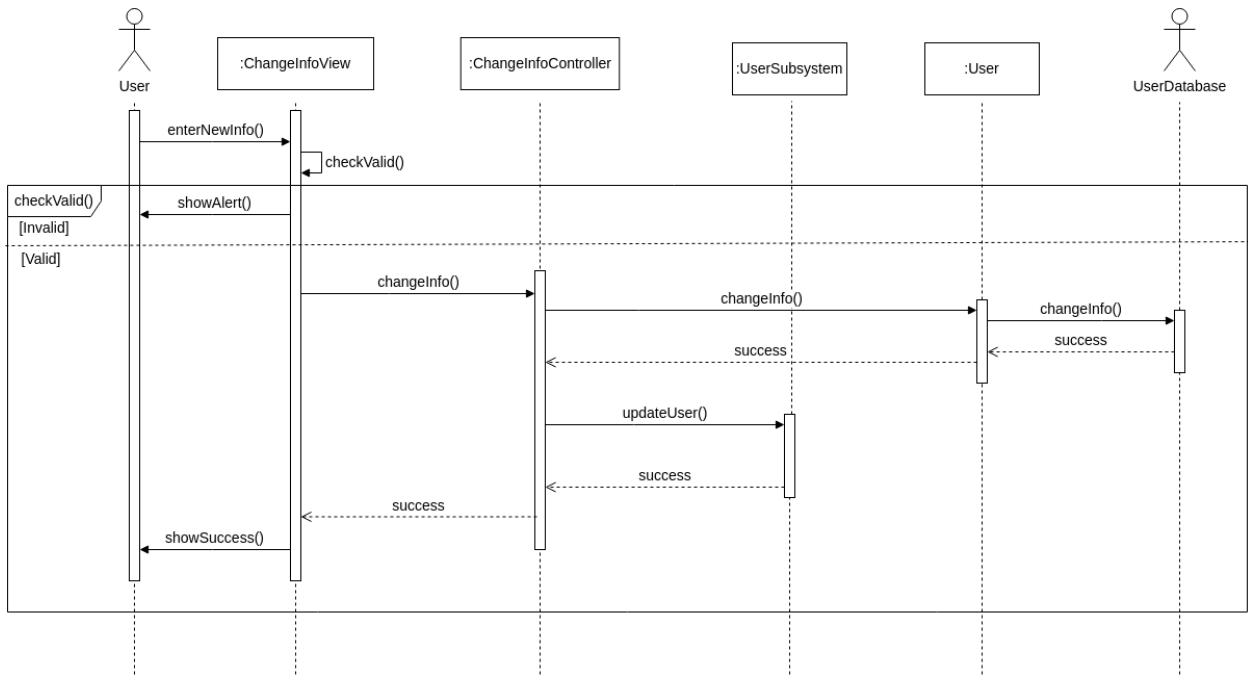
3.1.13 Đăng nhập



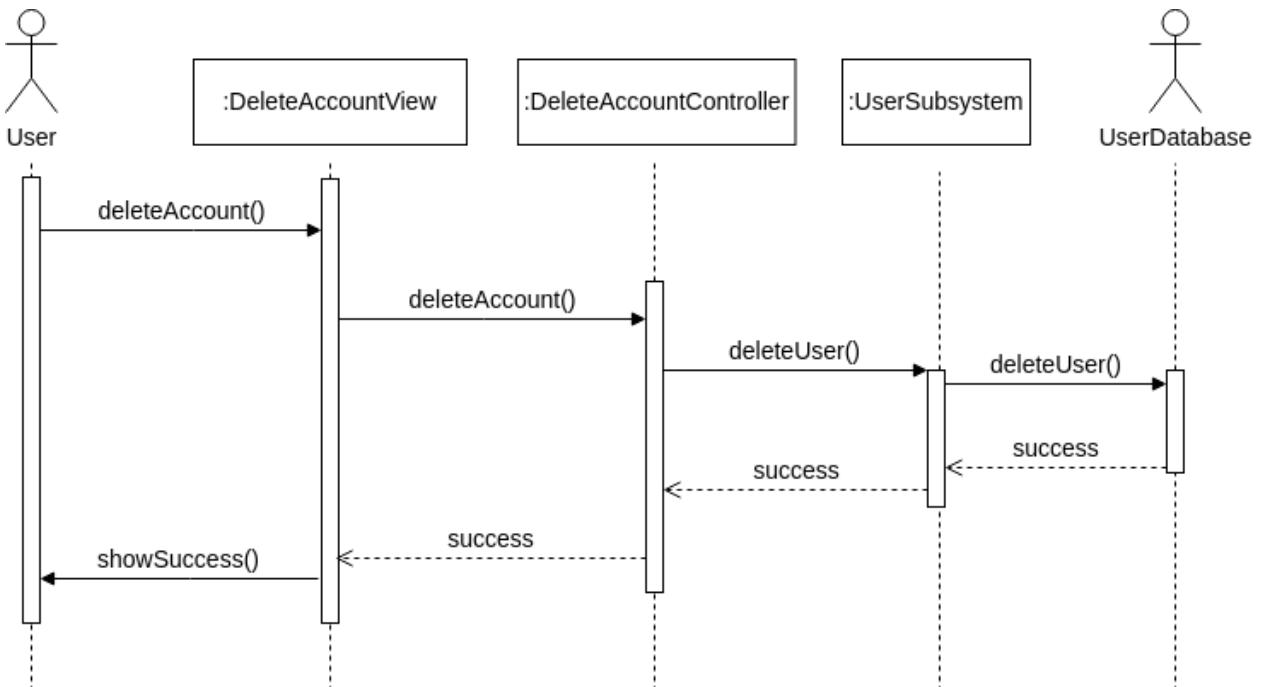
Hình 3.1.15: Biểu đồ tuần tự Đăng nhập



Hình 3.1.16: Biểu đồ tuần tự Quên mật khẩu

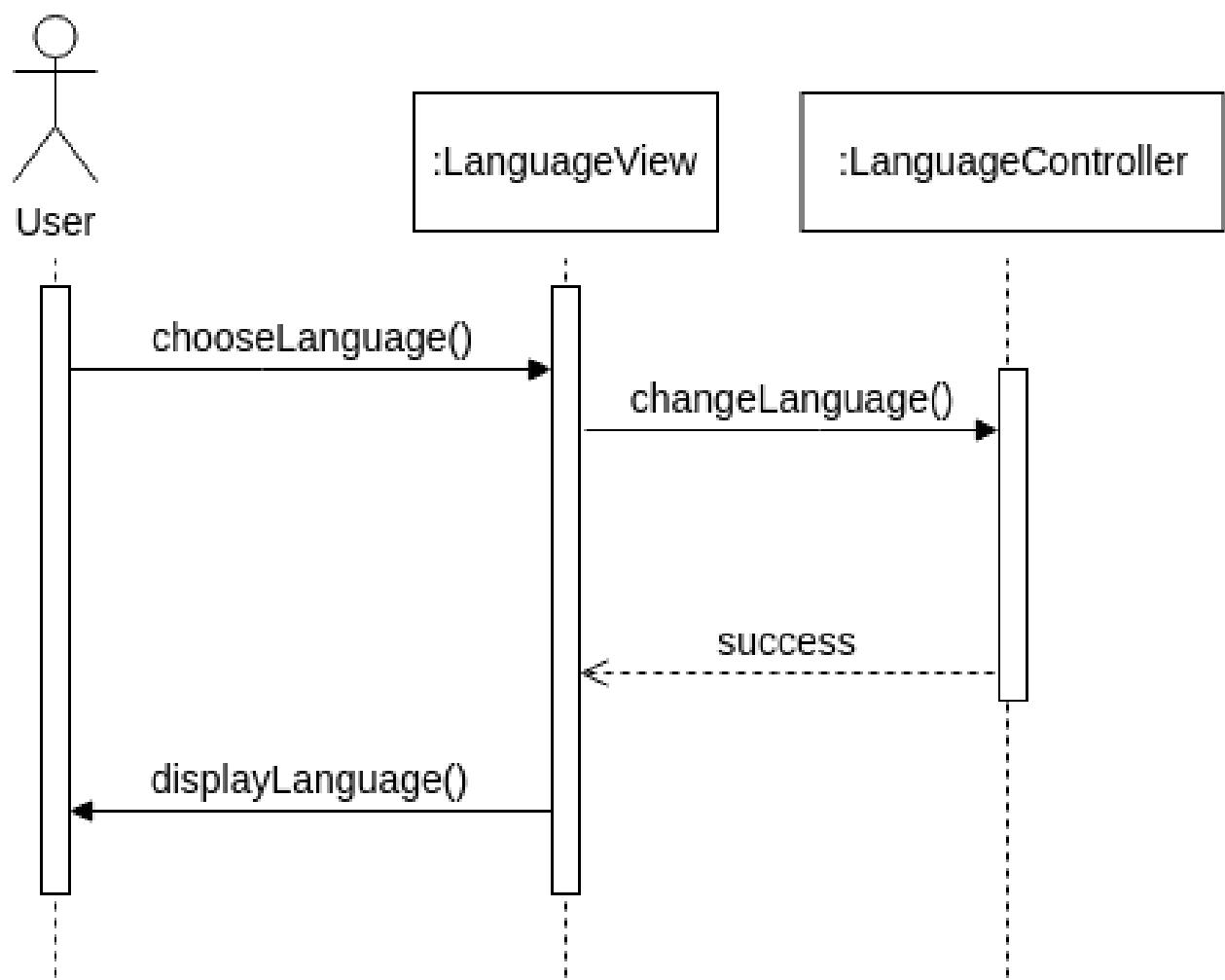


Hình 3.1.17: Biểu đồ tuần tự Thay đổi thông tin tài khoản



Hình 3.1.18: Biểu đồ tuần tự Xóa tài khoản

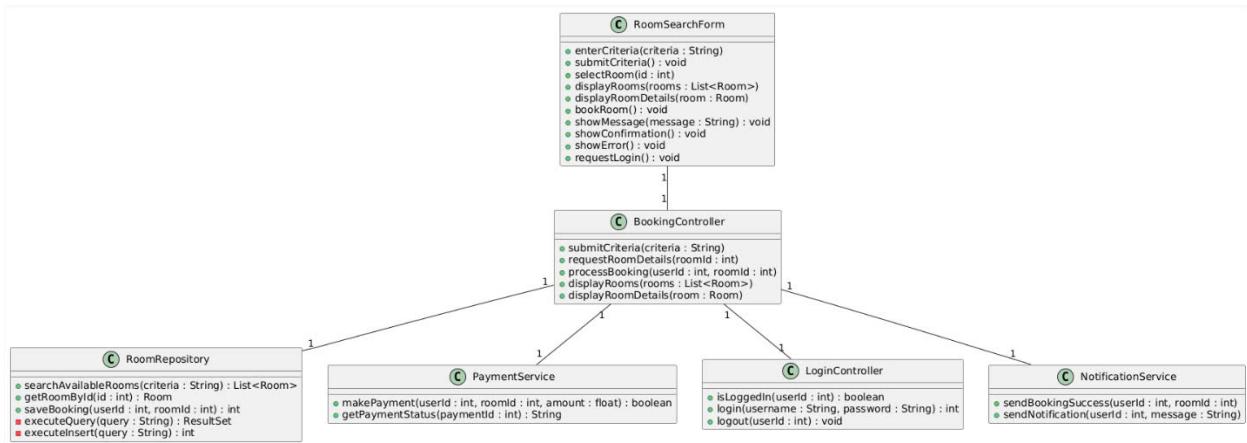
3.1.14 Lựa chọn ngôn ngữ



Hình 3.1.19: Biểu đồ tuần tự Lựa chọn ngôn ngữ

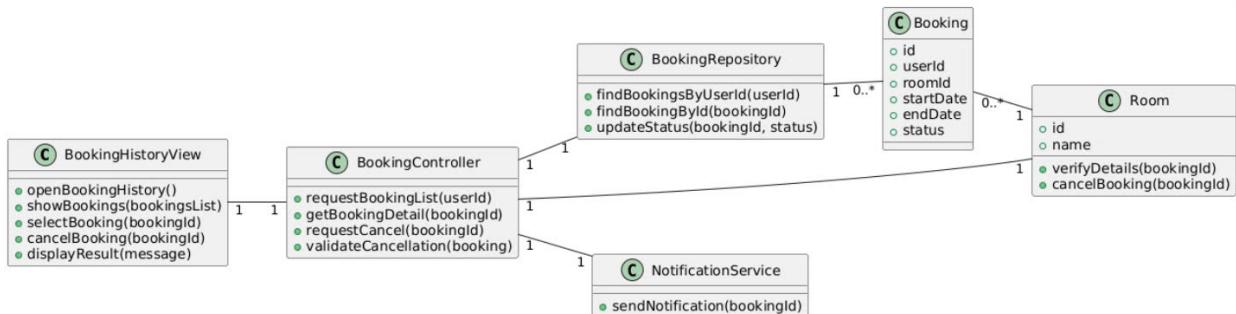
3.2 Thiết kế biểu đồ lớp

3.2.1 Đặt phòng khách sạn



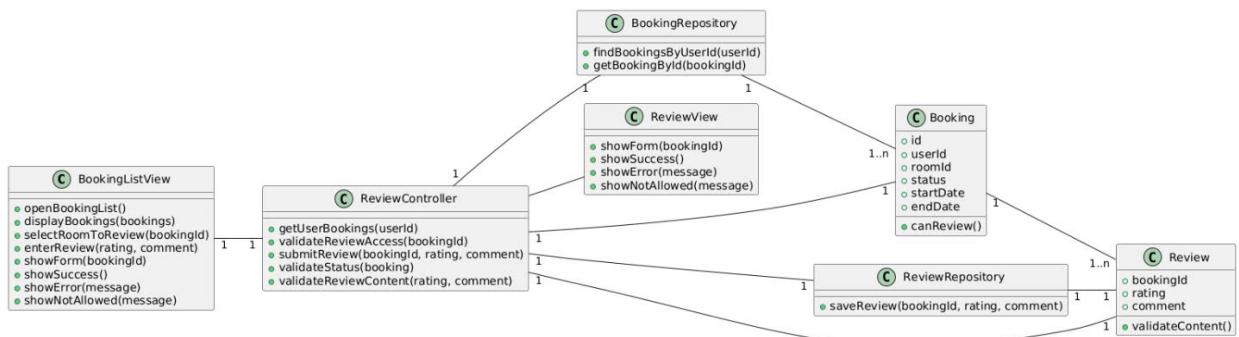
Hình 3.2.1: Biểu đồ lớp Đặt phòng khách sạn

3.2.2 Theo dõi phòng đã đặt



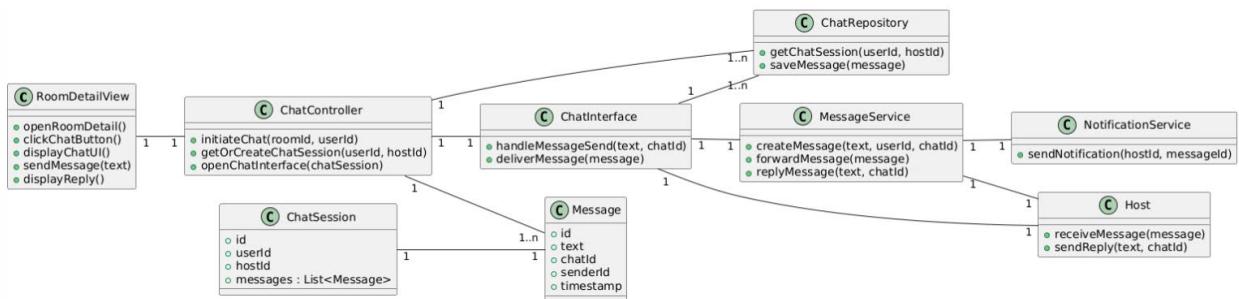
Hình 3.2.2: Biểu đồ lớp Theo dõi phòng đã đặt

3.2.3 Đánh giá phòng đã thuê



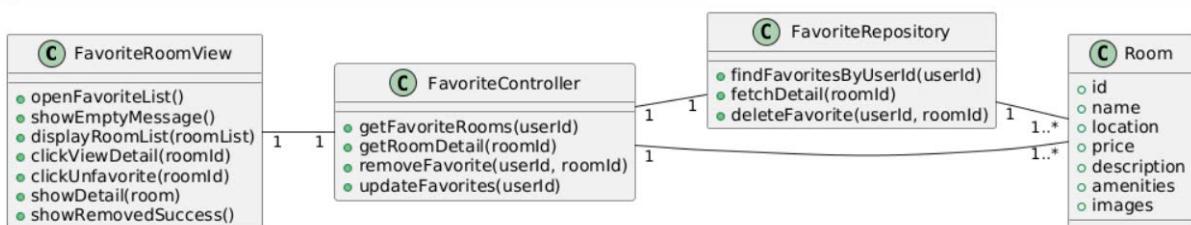
Hình 3.2.3: Biểu đồ lớp Đánh giá phòng đã thuê

3.2.4 Liên hệ với chủ khách sạn



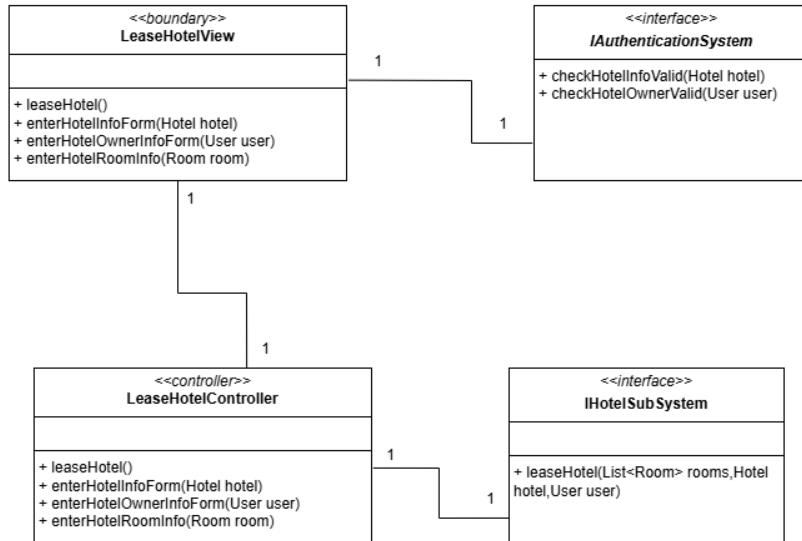
Hình 3.2.4: Biểu đồ lớp Liên hệ với chủ khách sạn

3.2.5 Xem danh sách phòng yêu thích



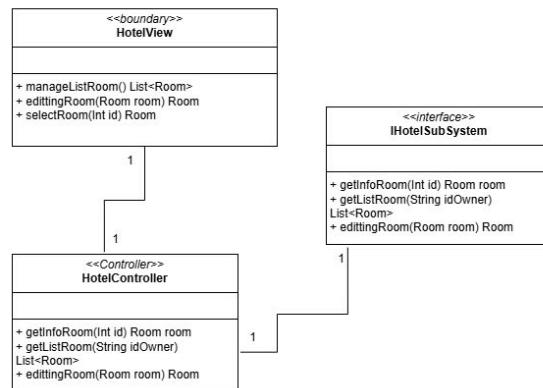
Hình 3.2.5: Biểu đồ lớp Xem danh sách phòng yêu thích

3.2.6 Cho thuê phòng



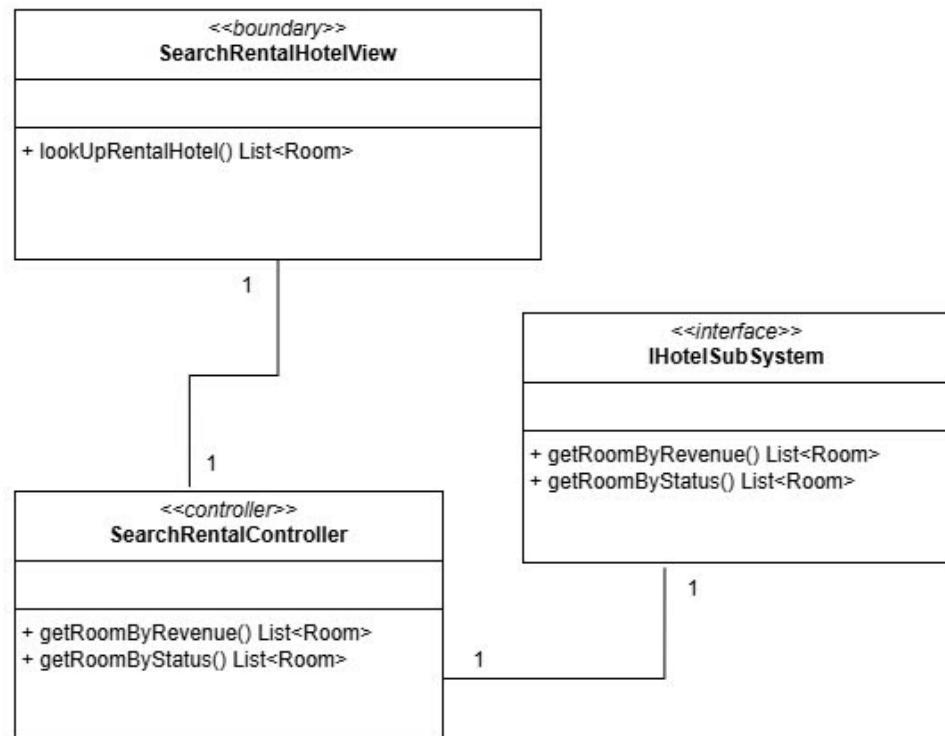
Hình 3.2.6: Biểu đồ lớp Cho thuê phòng

3.2.7 Sửa thông tin phòng



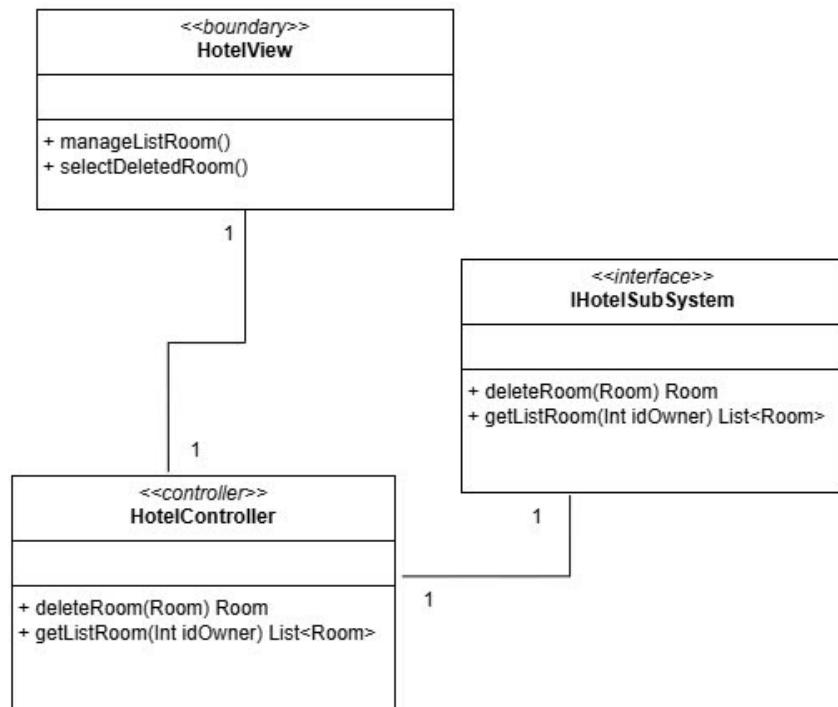
Hình 3.2.7: Biểu đồ lớp Sửa thông tin phòng

3.2.8 Tra cứu thông tin phòng đang cho thuê



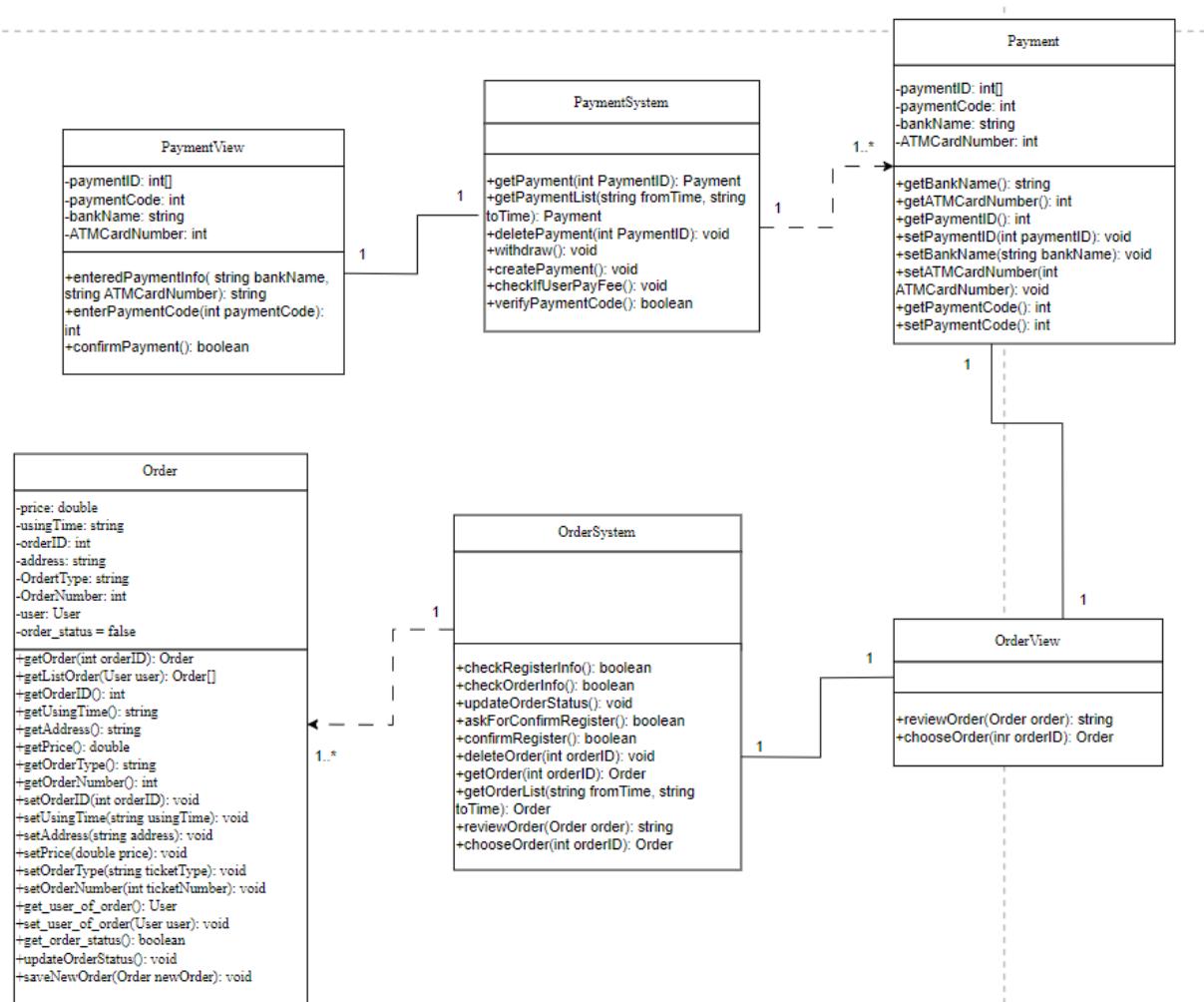
Hình 3.2.8: Biểu đồ lớp Tra cứu thông tin phòng đang cho thuê

3.2.9 Xóa phòng khách sạn

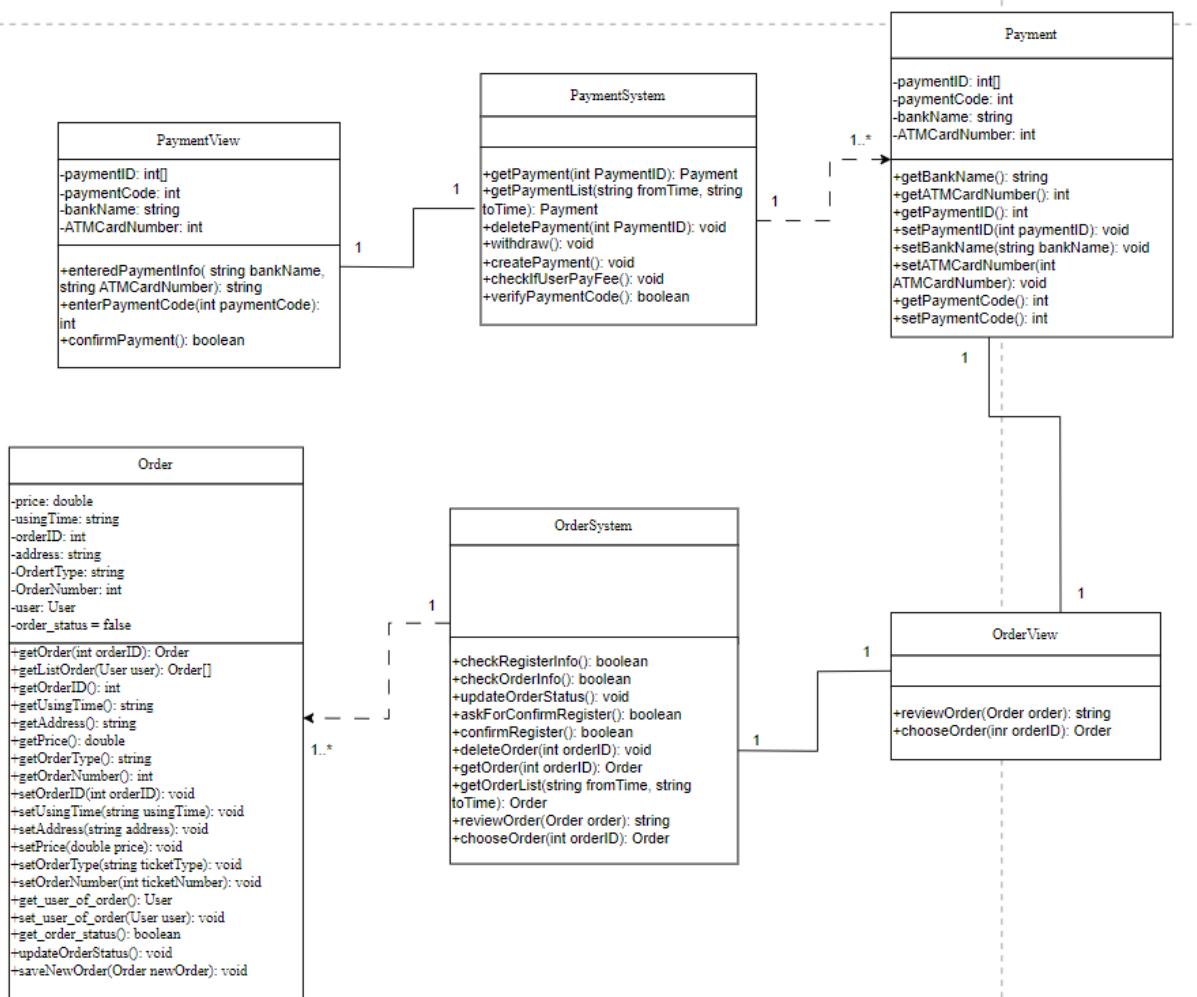


Hình 3.2.9: Biểu đồ lớp Xóa phòng khách sạn

3.2.10 Thanh toán

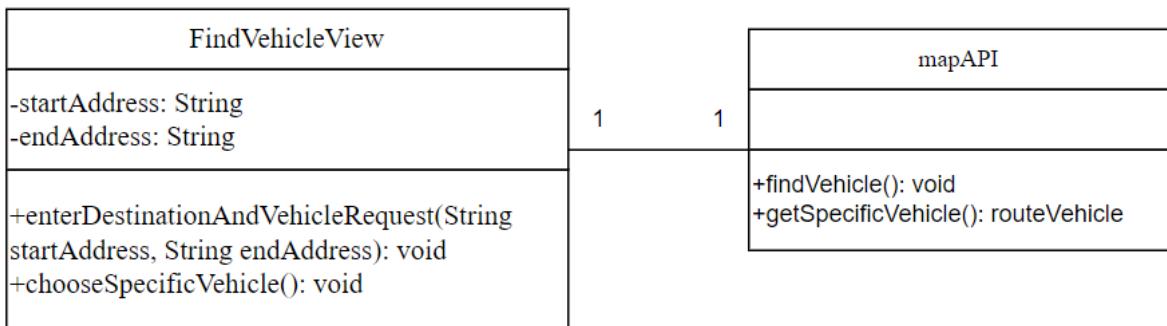


Hình 3.2.10: Biểu đồ lớp Thanh toán trực tuyến



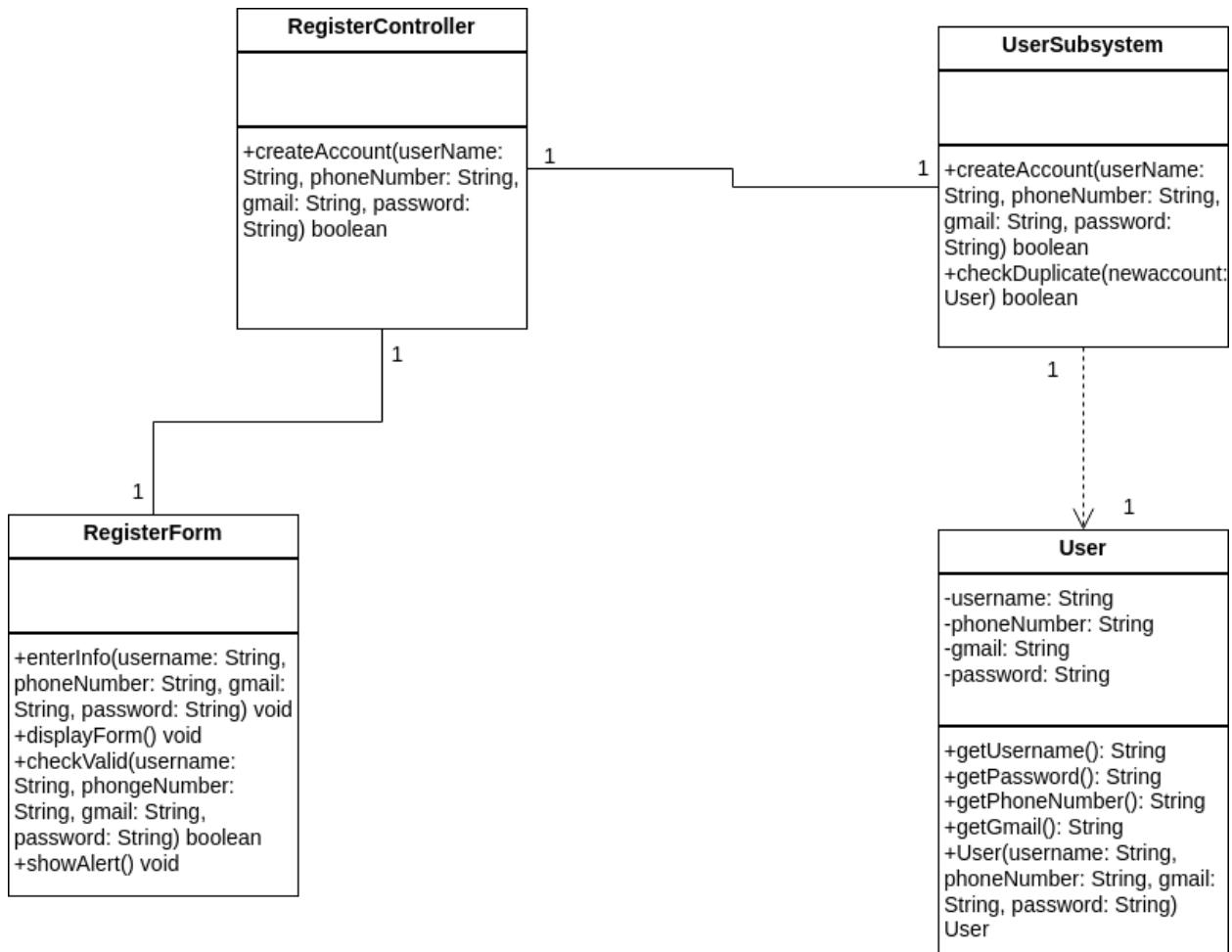
Hình 3.2.11: Biểu đồ lớp Thanh toán trực tiếp

3.2.11 Tra cứu phương tiện



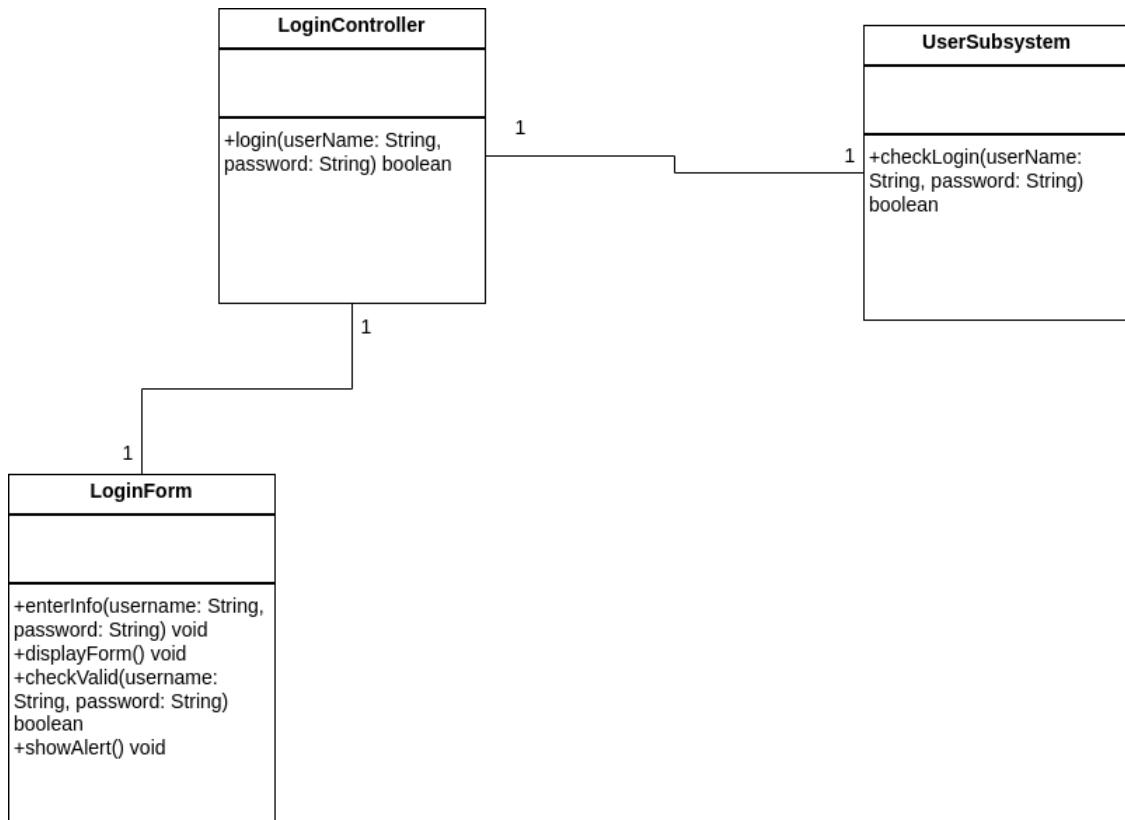
Hình 3.2.12: Biểu đồ lớp Tra cứu phương tiện

3.2.12 Đăng ký tài khoản

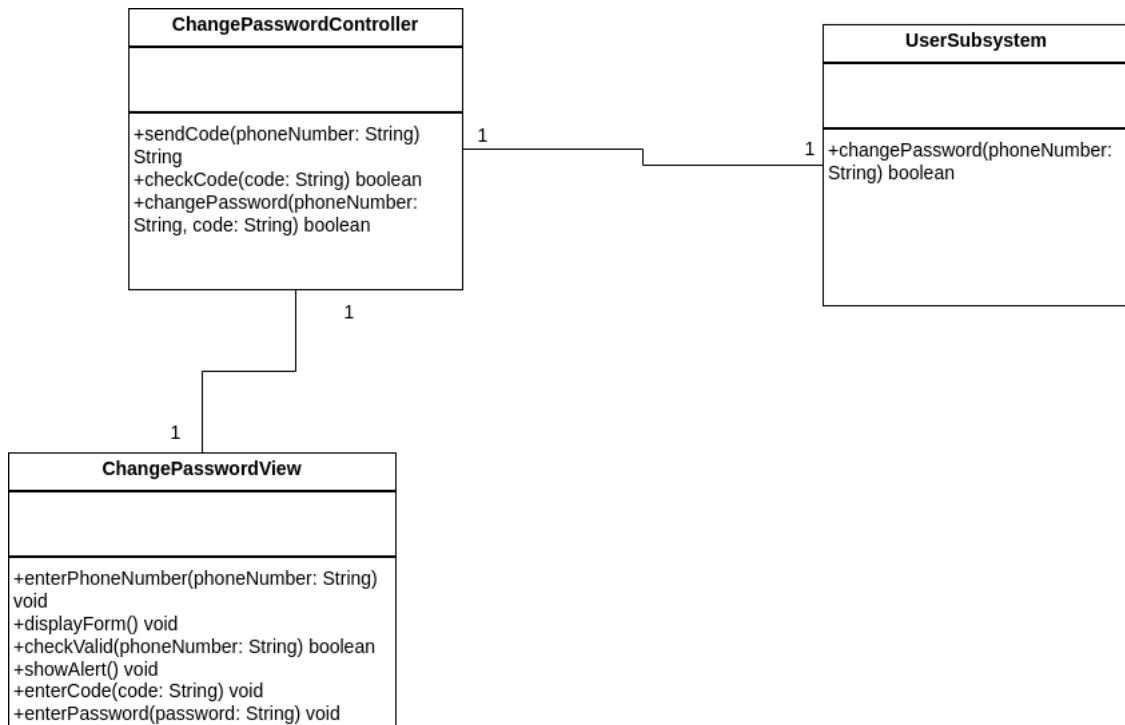


Hình 3.2.13: Biểu đồ lớp Đăng ký tài khoản

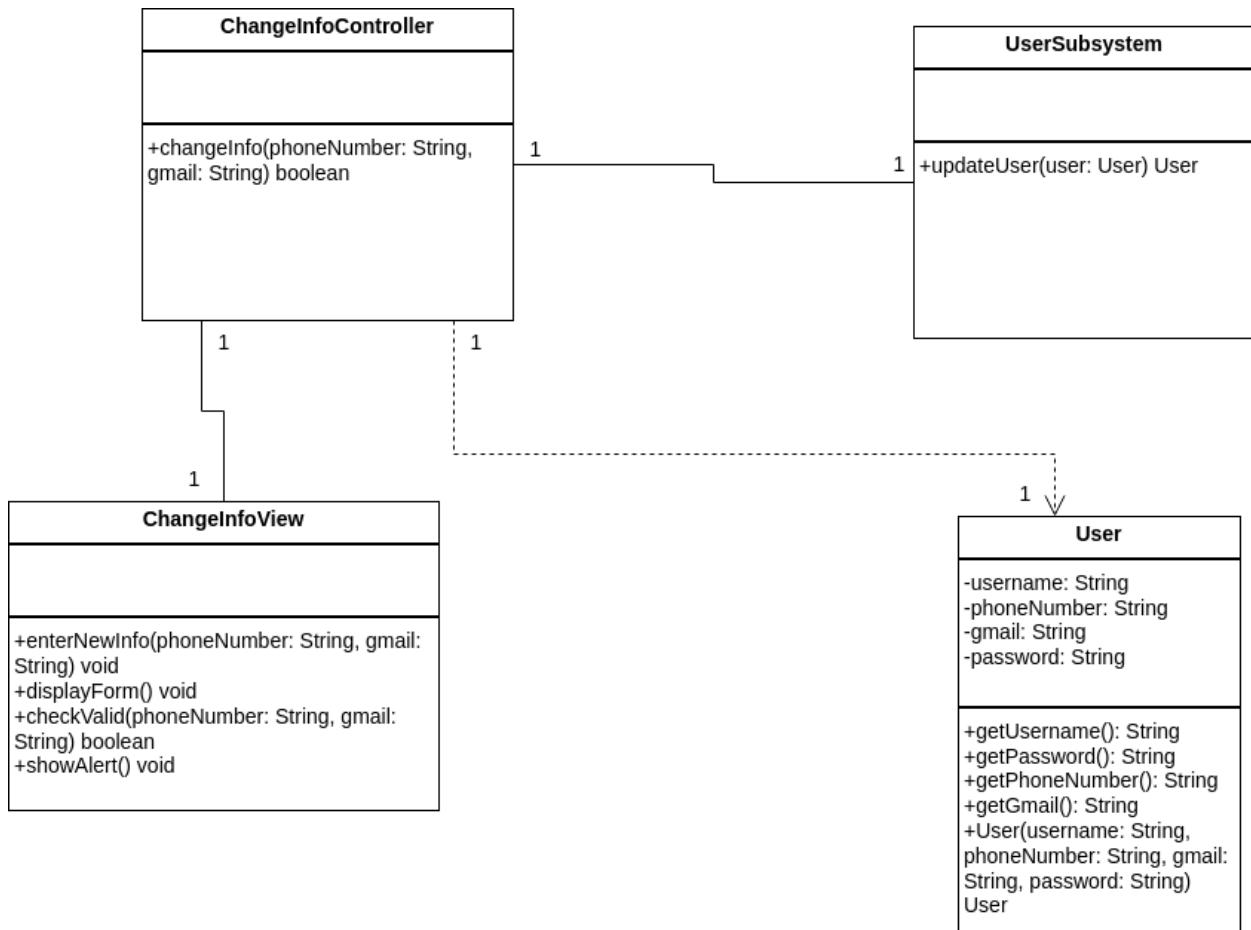
3.2.13 Đăng nhập



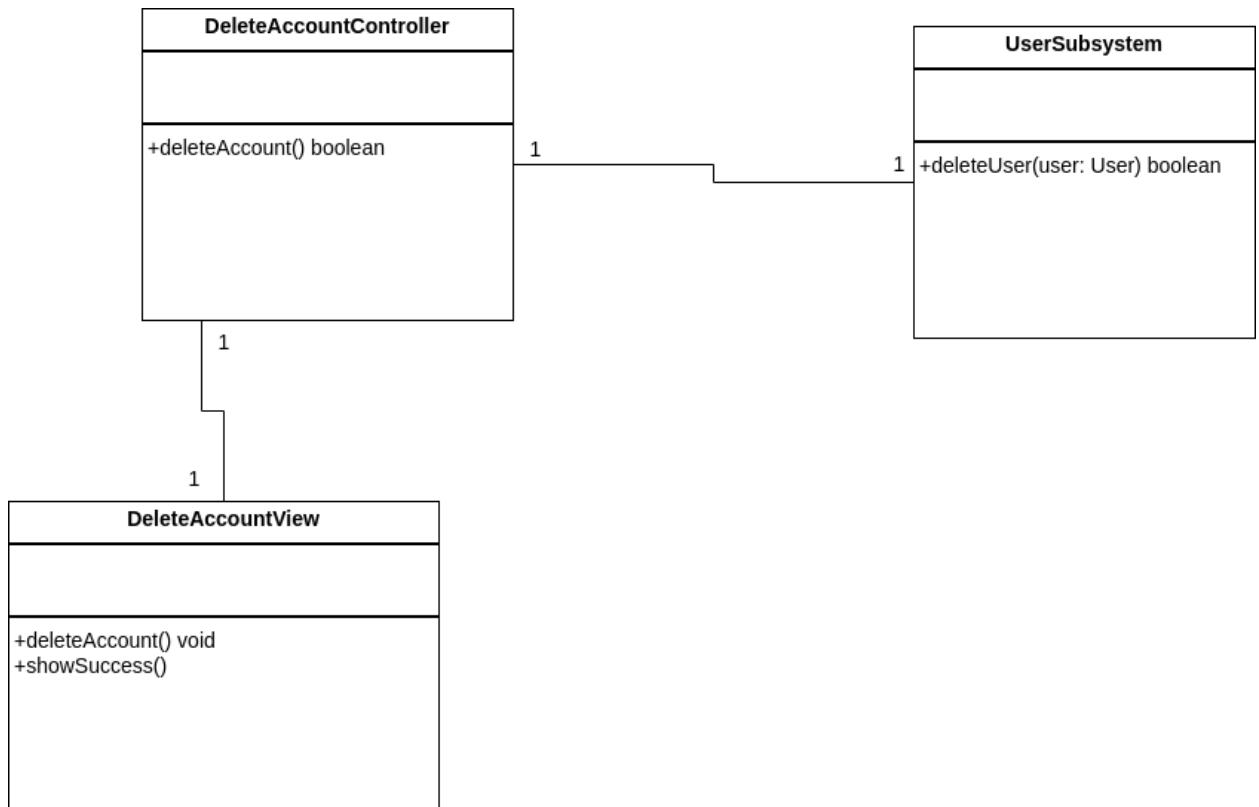
Hình 3.2.14: Biểu đồ lớp Đăng nhập



Hình 3.2.15: Biểu đồ lớp Quên mật khẩu

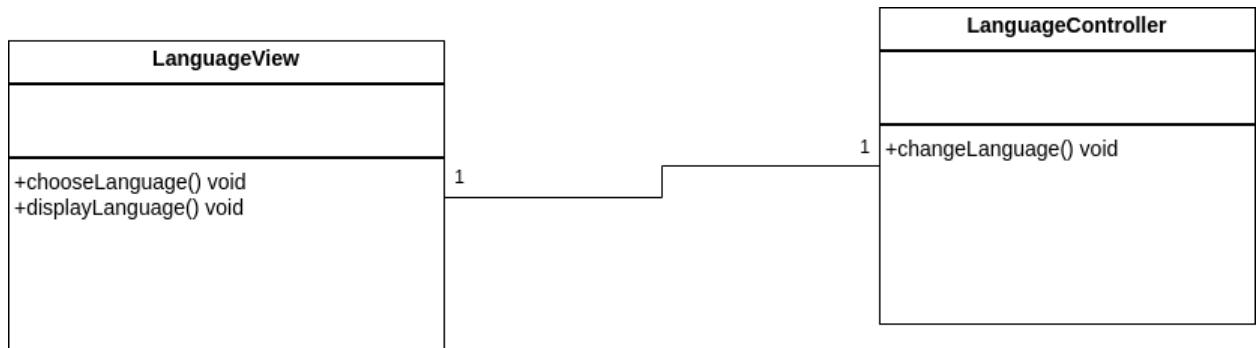


Hình 3.2.16: Biểu đồ lớp Thay đổi thông tin tài khoản



Hình 3.2.17: Biểu đồ lớp Xóa tài khoản

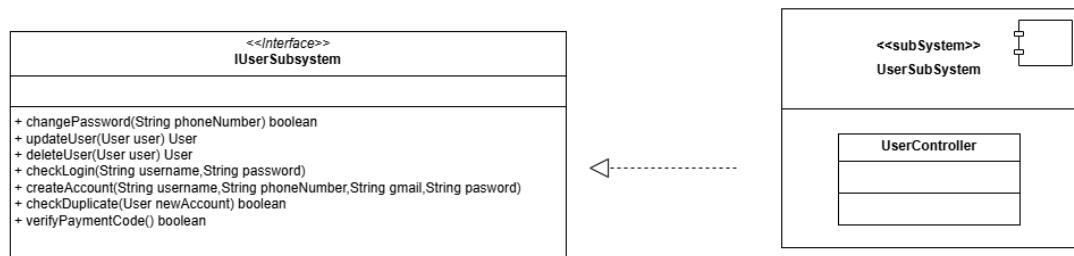
3.2.14 Lựa chọn ngôn ngữ



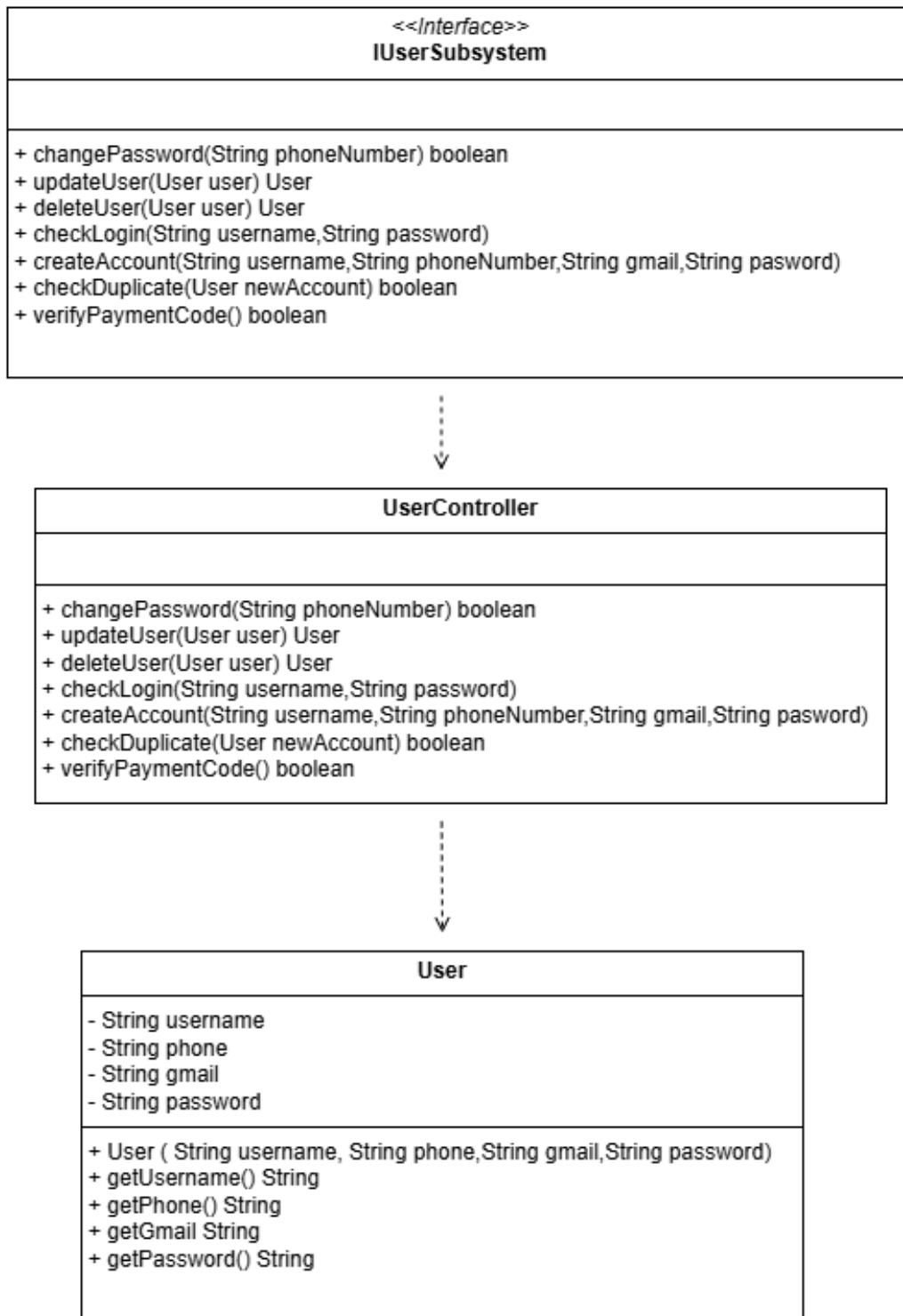
Hình 3.2.18: Biểu đồ lớp Lựa chọn ngôn ngữ

4 Thiết kế Hệ thống con

4.1 User Subsystem



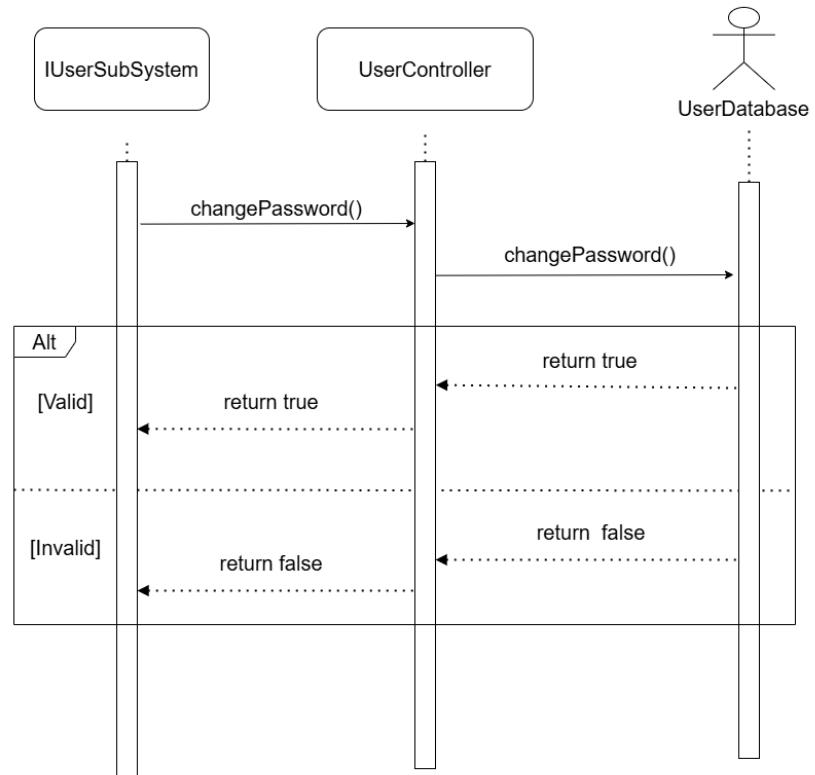
Hình 4.1.1: Biểu đồ cấu trúc UserSubsystem



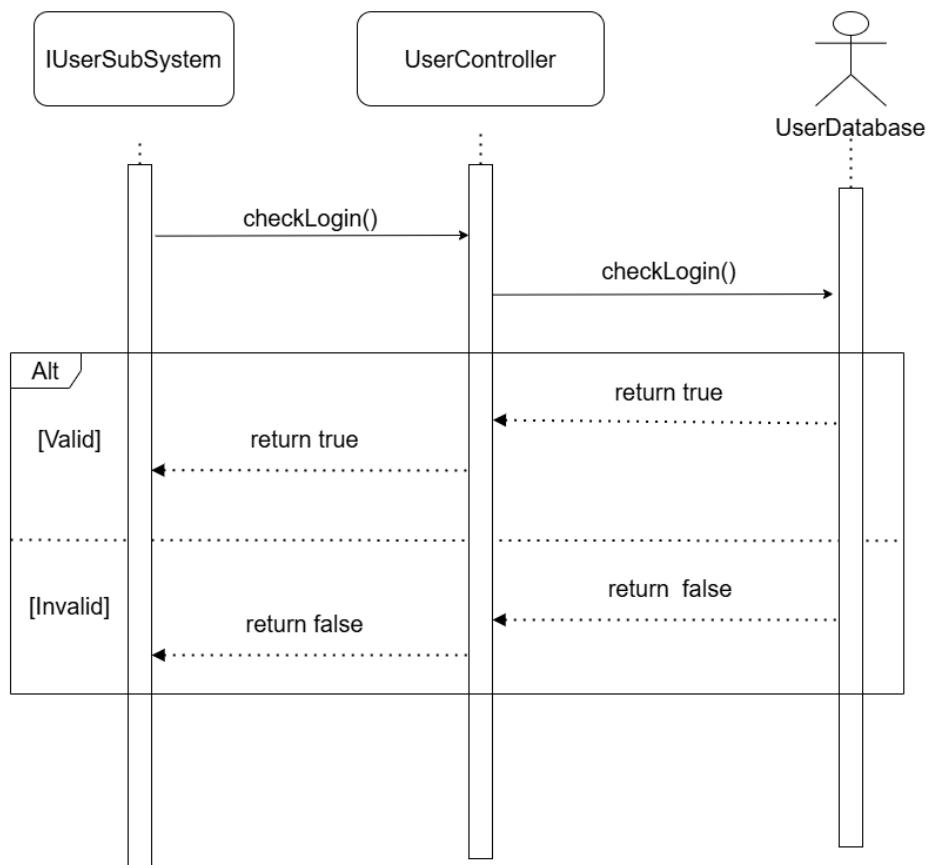
Hình 4.1.2: Biểu đồ quan hệ UserSubsystem

Interface IUserSubsystem

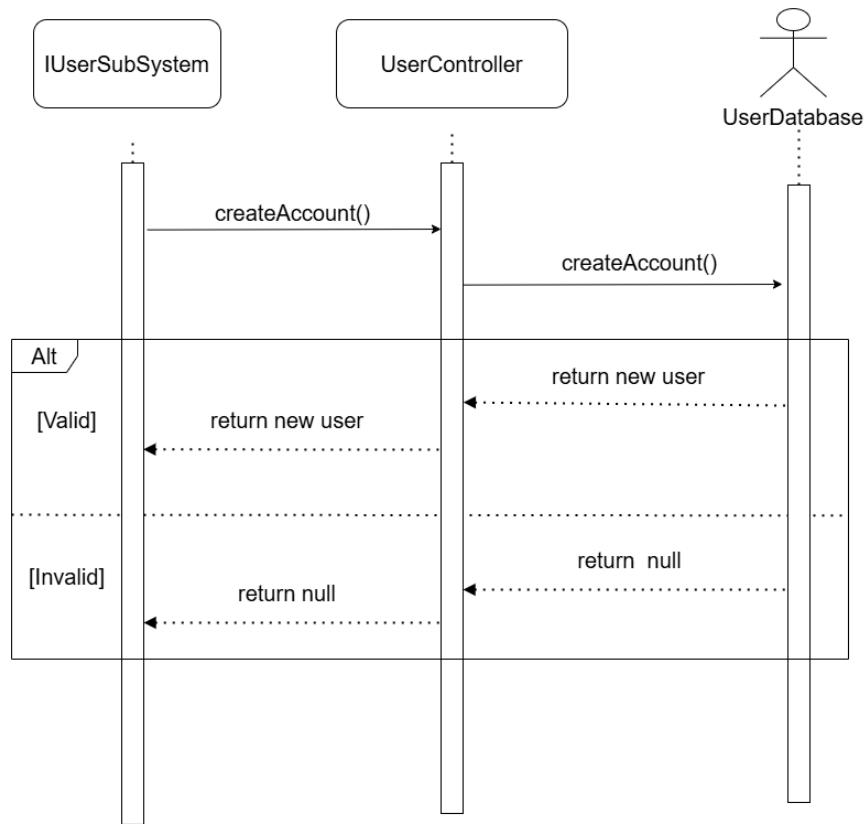
1. Hàm `changePassword()`



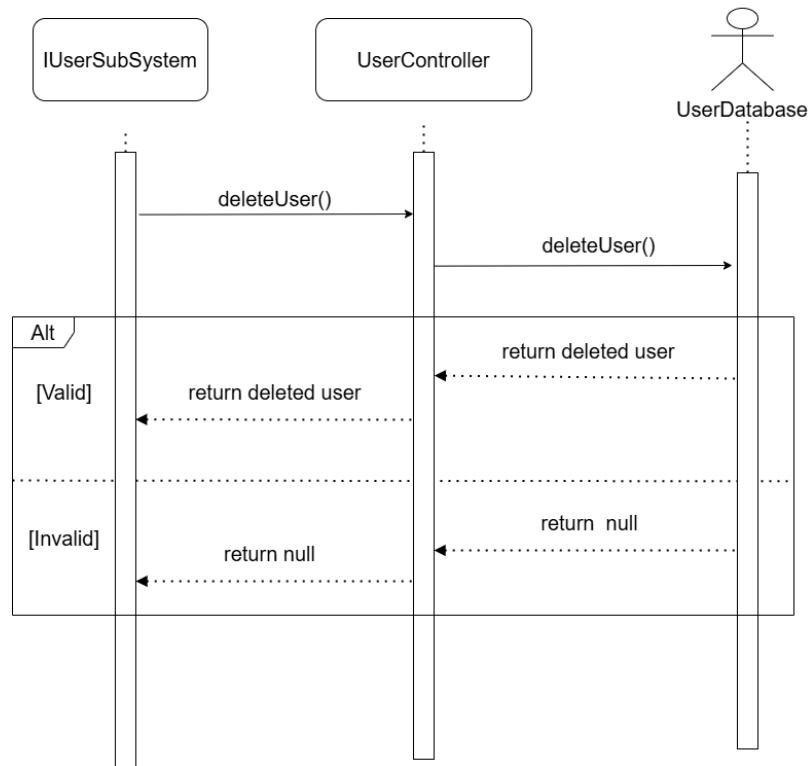
2. Hàm checkLogin()



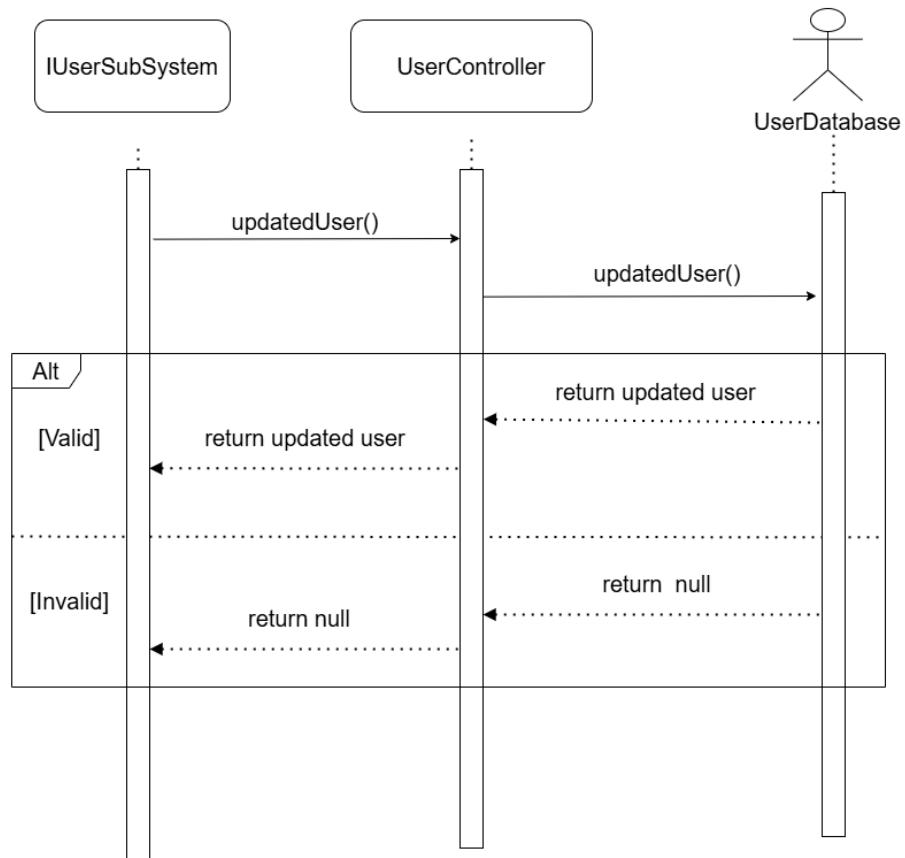
3. Hàm createAccount()



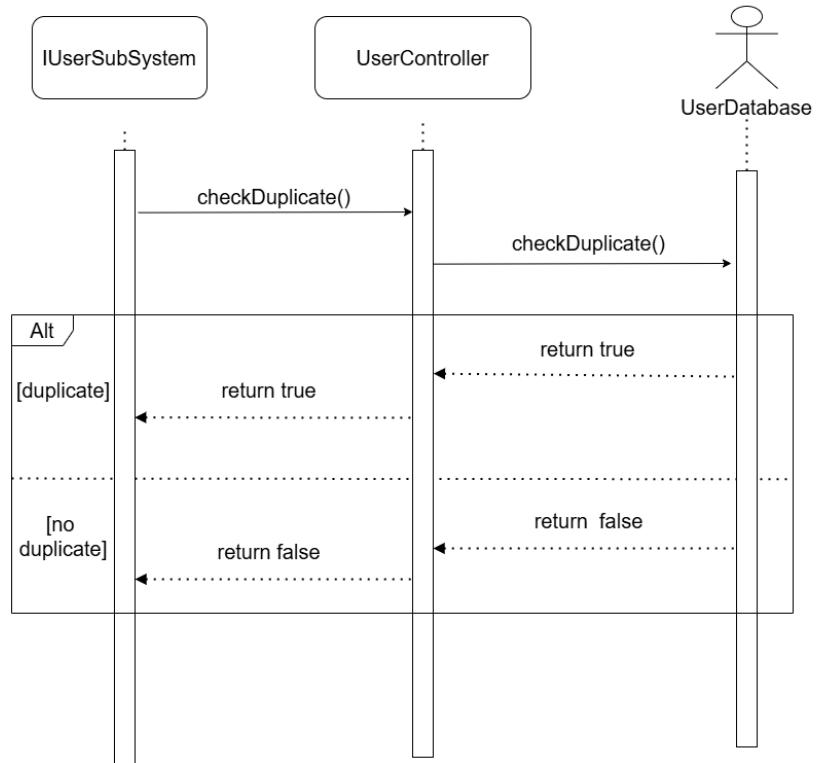
4. Hàm deleteUser()



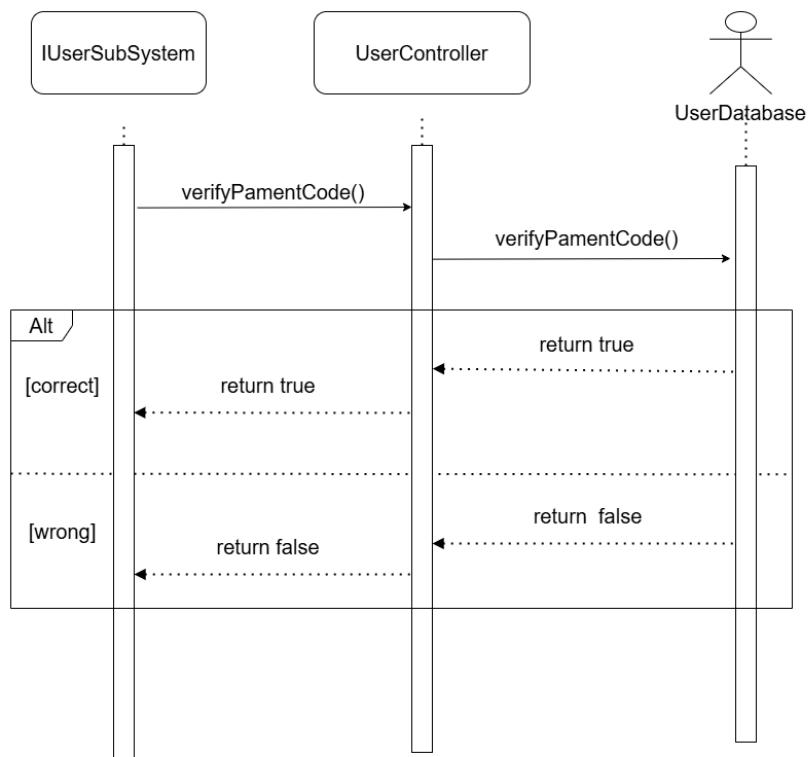
5. Hàm updateUser()



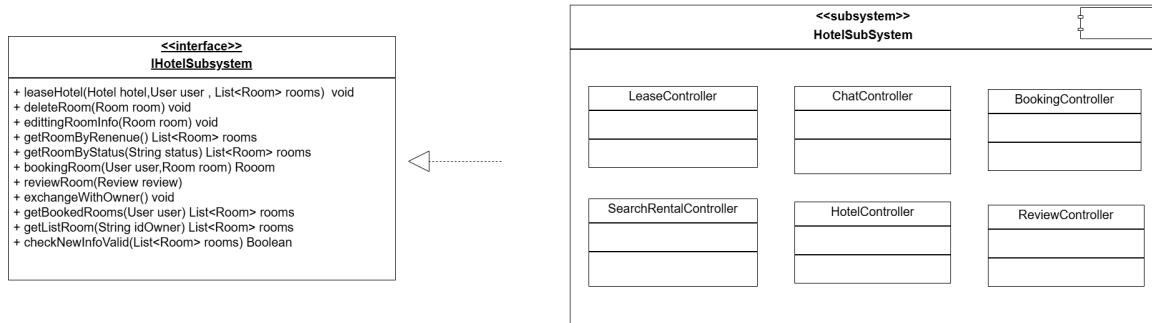
6. Hàm checkDuplicate()



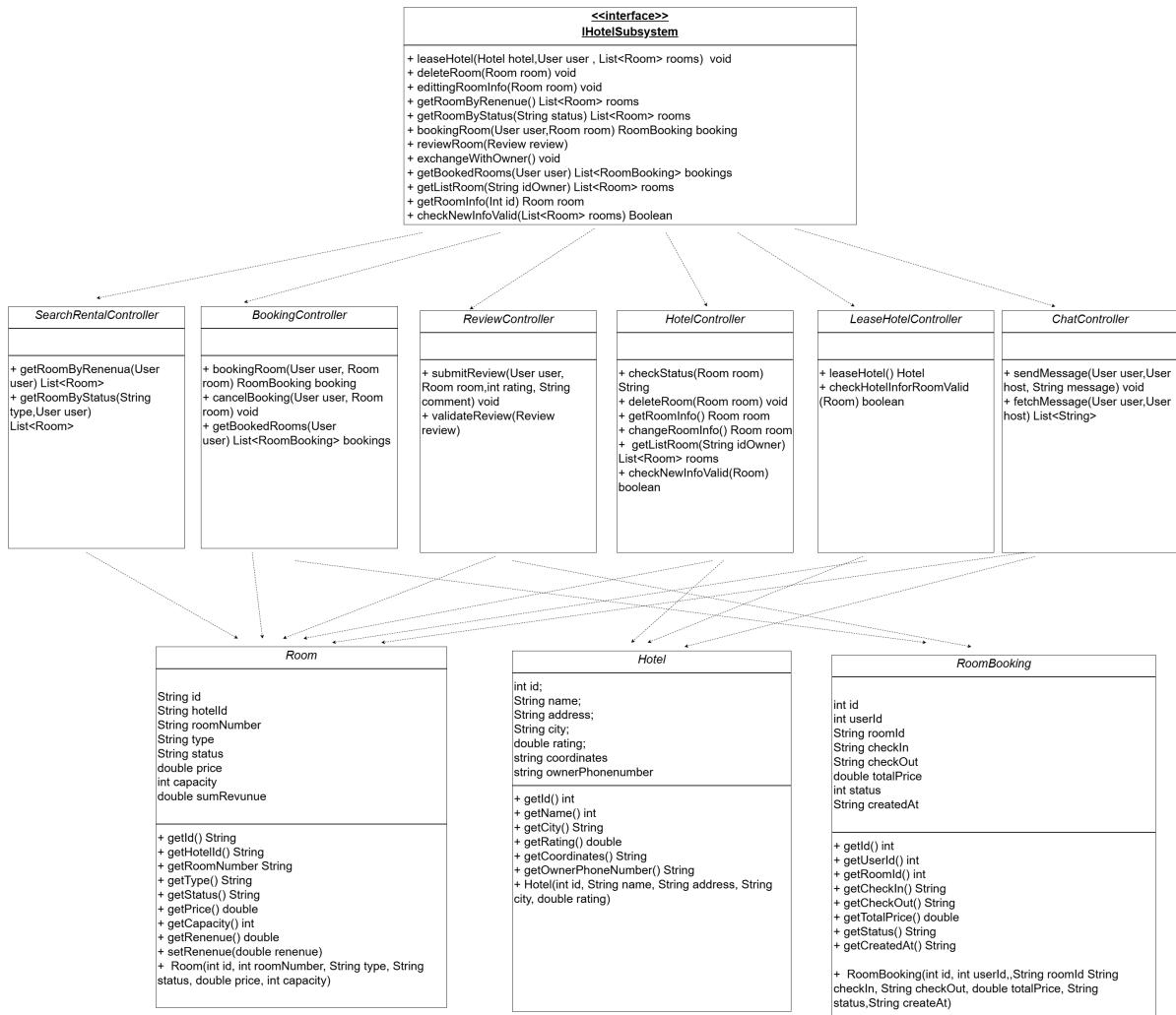
7. Hàm verifyPaymentCode()



4.2 Hotel Subsystem



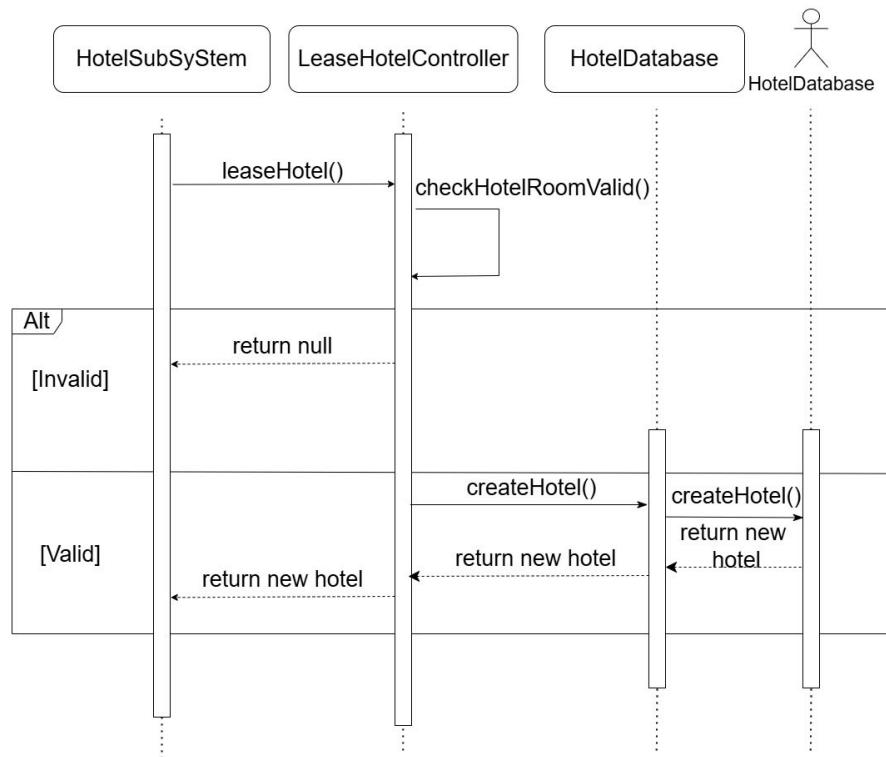
Hình 4.2.1: Biểu đồ cấu trúc HotelSubSystem



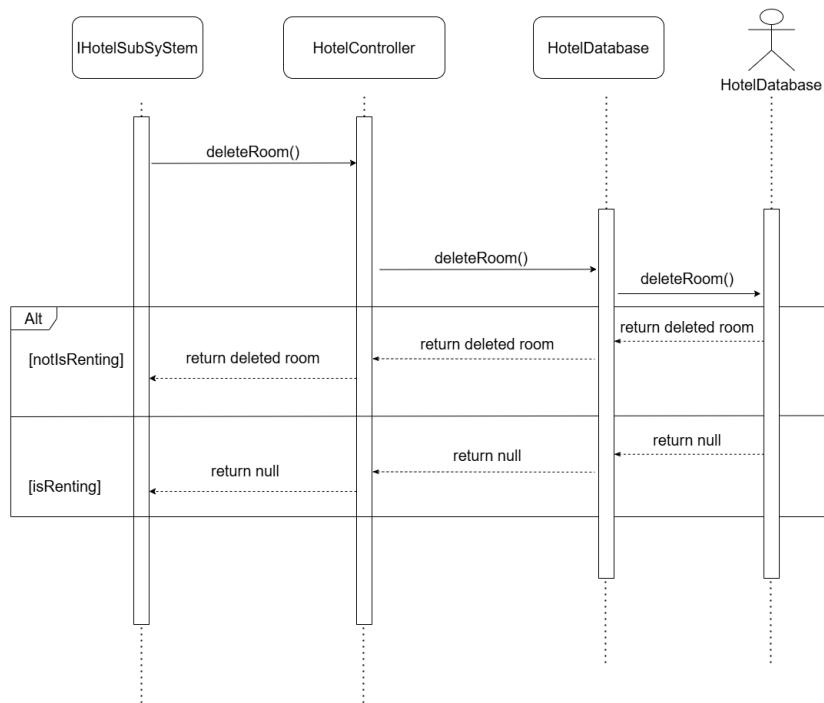
Hình 4.2.2: Biểu đồ quan hệ HotelSubSystem

Interface IHotelSubsystem

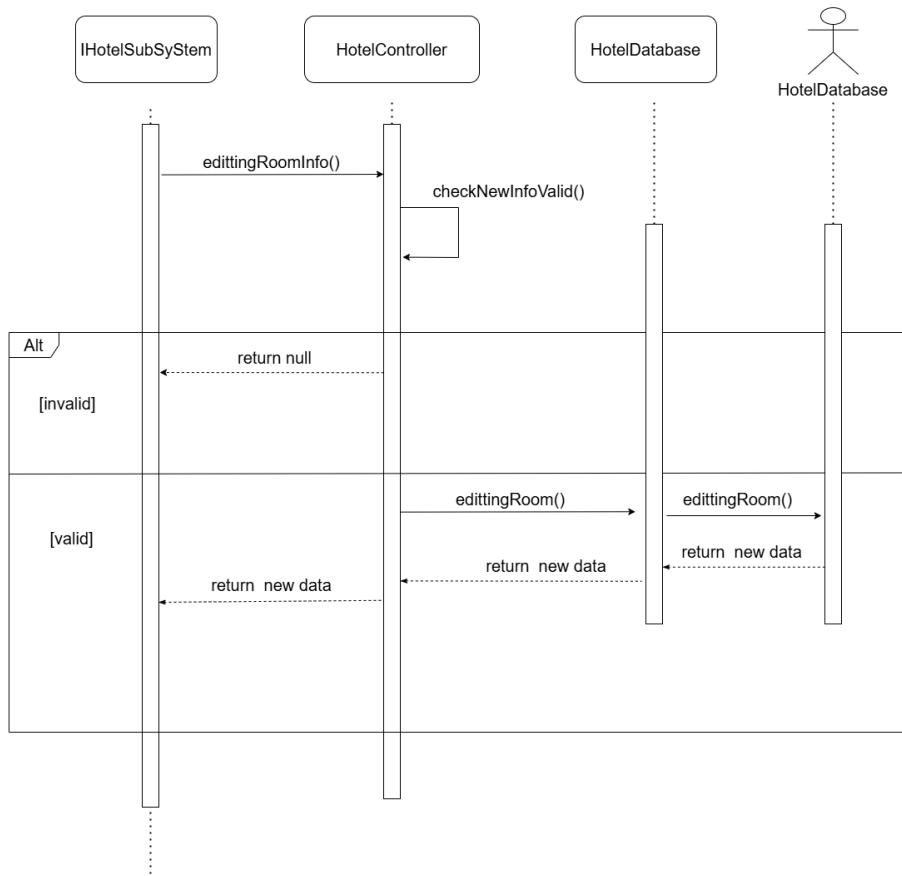
1. Hàm leaseHotel()



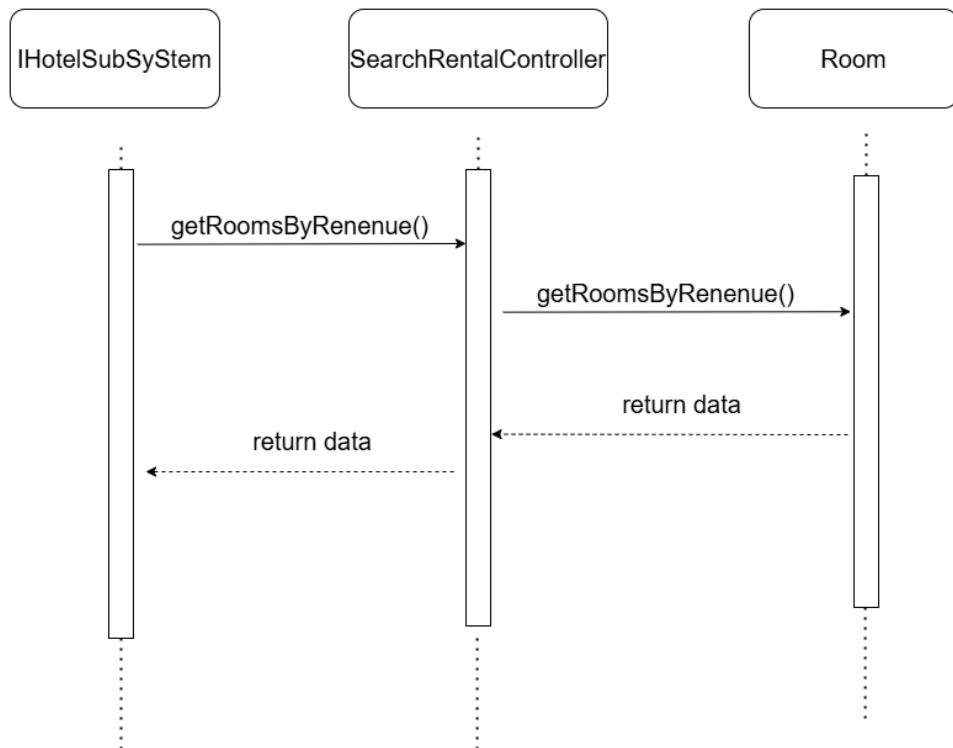
2. Hàm deleteRoom()



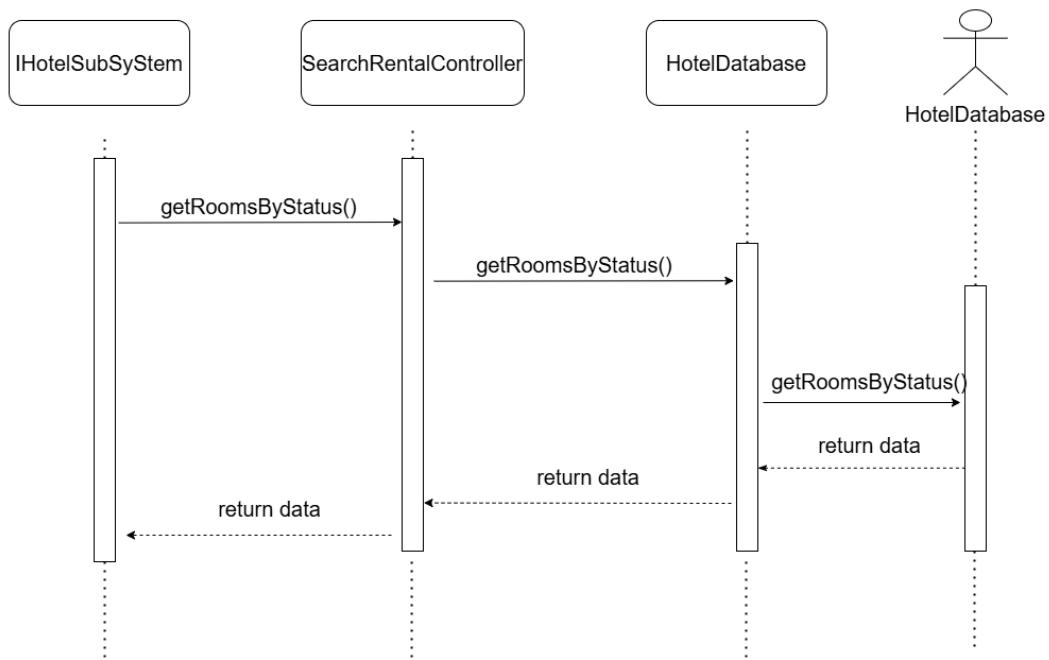
3. Hàm edittingRoomInfo()



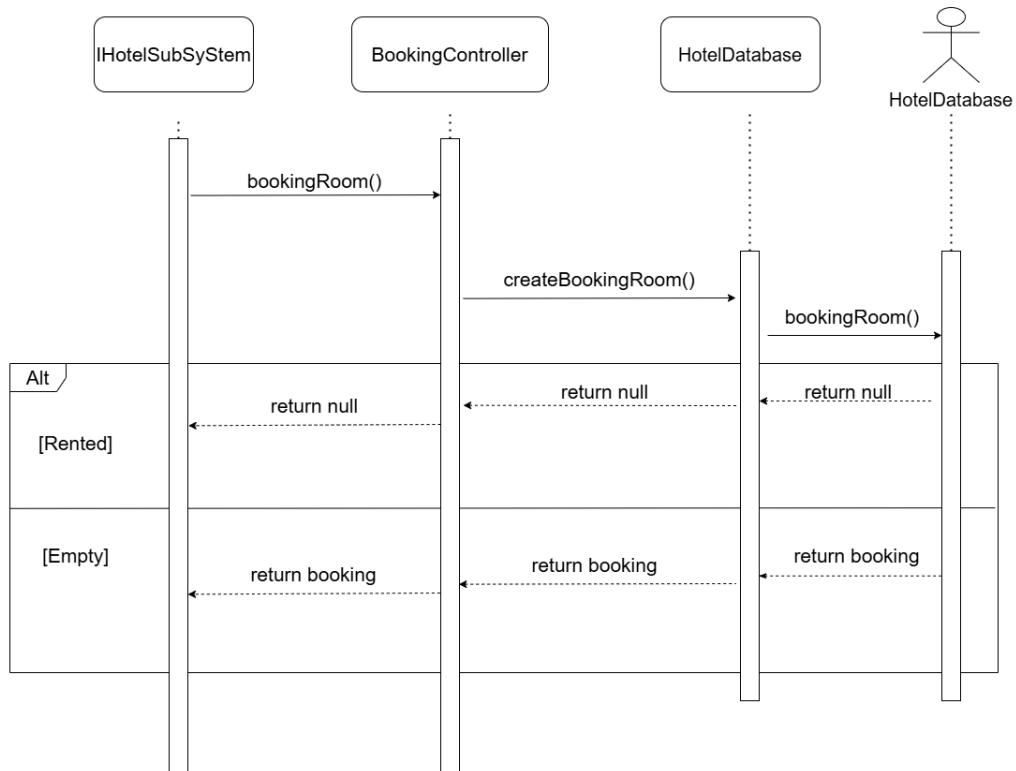
4. Hàm getRoomsByRenenuer()



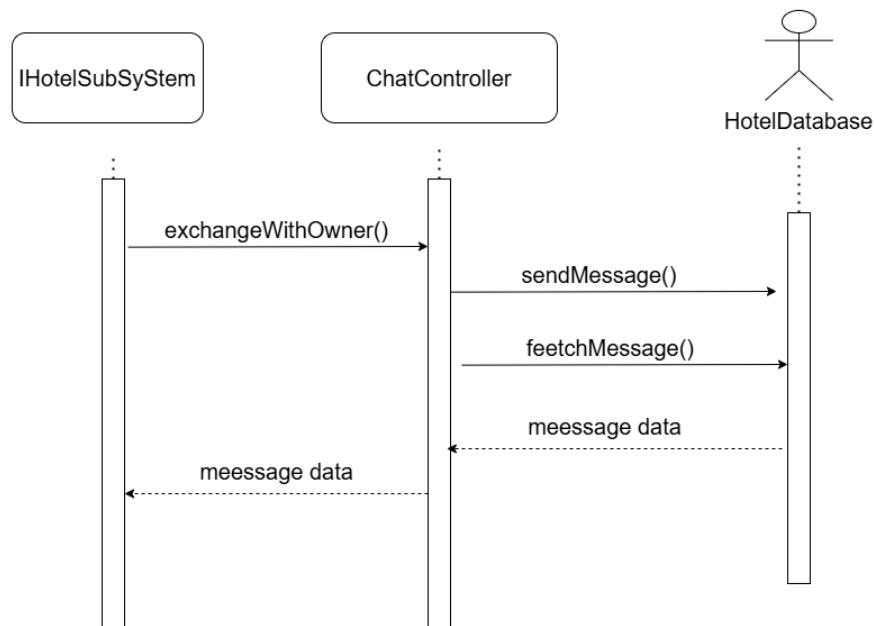
5. Hàm getRoomsByStatus()



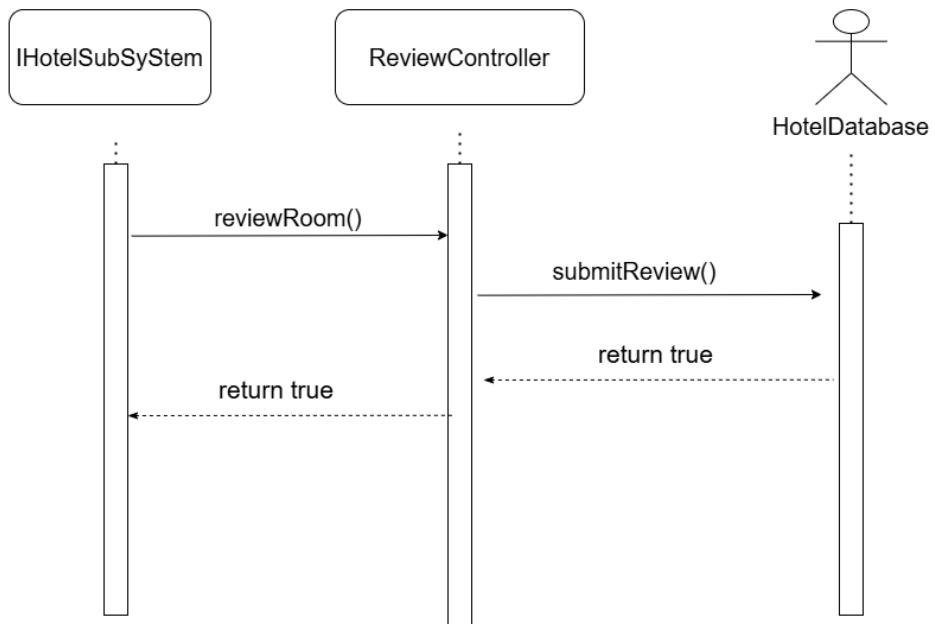
6. Hàm bookingRoom()



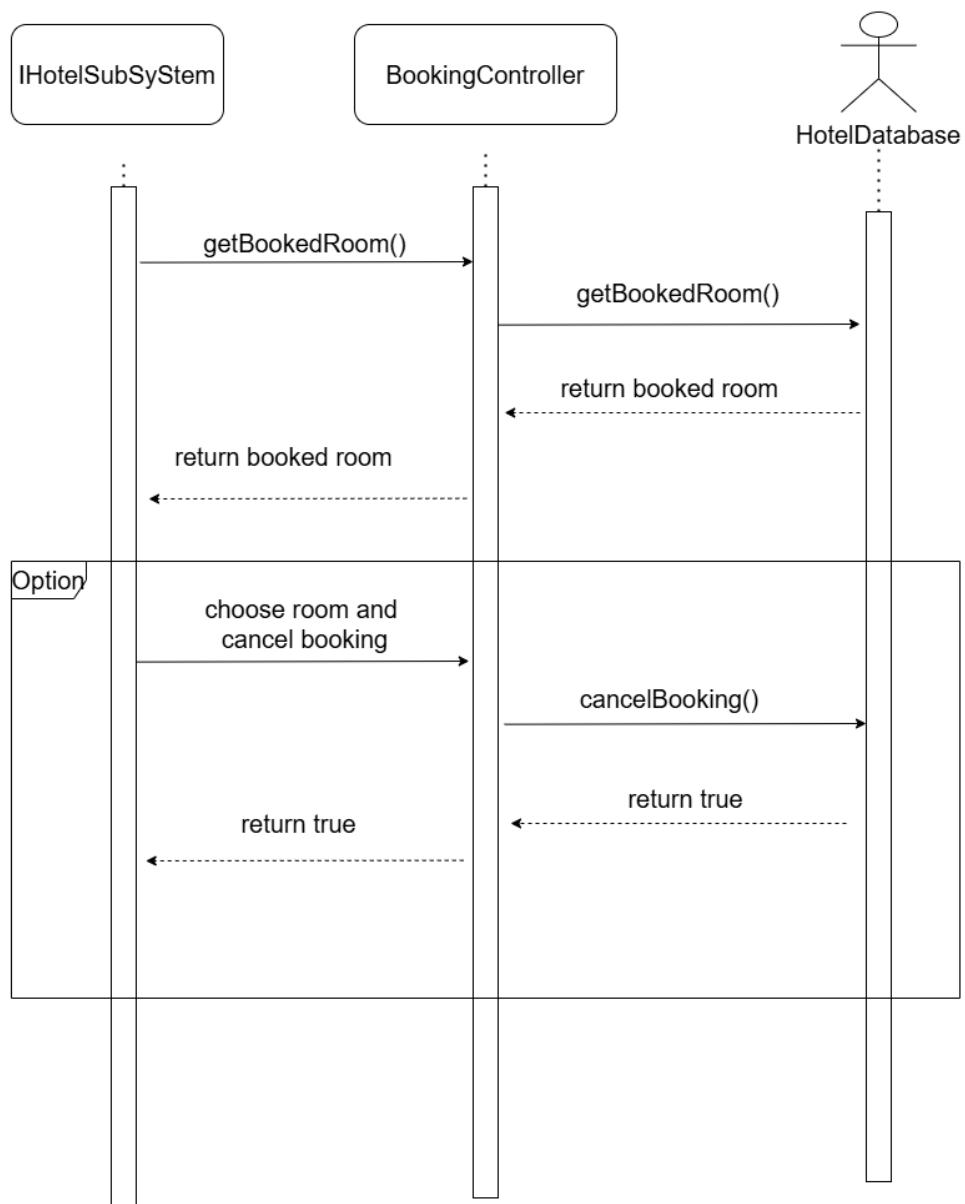
7. Hàm exchangeWithOwner()



8. Hàm reviewRoom()



9. Hàm getBookedRoom()



5 Thiết kế lớp

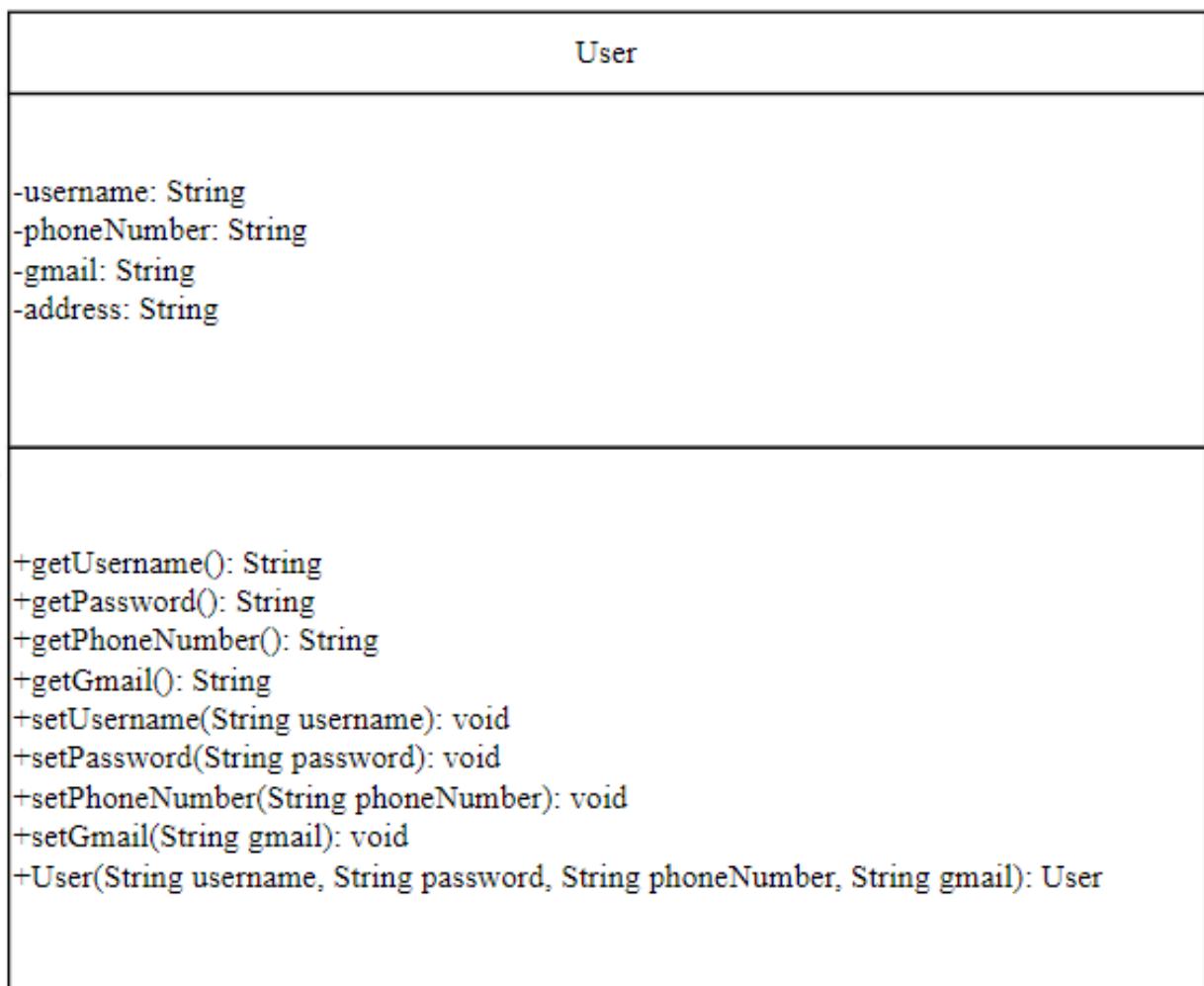
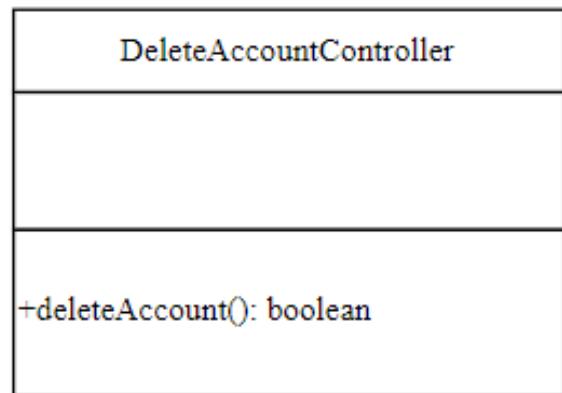
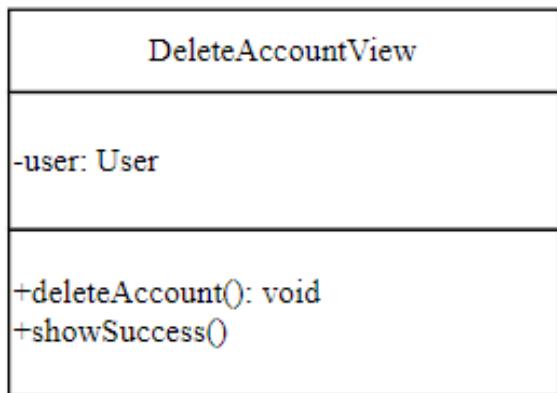
RegisterForm	RegisterController
<pre>-username: String -phoneNumber: String -gmail: String -password: String</pre>	<pre>-username: String -phoneNumber: String -gmail: String -password: String</pre>
<pre>+setInfo(String username, String phoneNumber, String gmail, String password): void +displayForm(): void +checkValid(String username, String phoneNumber, String gmail, String password): boolean +showAlert(): void</pre>	<pre>+createAccount(String username, String phoneNumber, String gmail, String password): boolean</pre>
ChangeInfoView	ChangeInfoView
<pre>-phoneNumber: String -gmail: String</pre>	<pre>-phoneNumber: String -gmail: String</pre>
<pre>+setNewInfo(String phoneNumber, String gmail): void +displayForm(): void +checkValid(String phoneNumber, String gmail): boolean +showAlert(): void</pre>	<pre>+changeInfo(String phoneNumber, String gmail): boolean</pre>

LoginForm
<pre>-username: String -password: String</pre>
<pre>+setInfo(String username, String password): void +displayForm(): void +checkValid(String username, String password): void +showAlert(): void</pre>

LoginController
<pre>-username: String -password: String</pre>
<pre>+login(String username, String password): boolean</pre>

ChangePasswordView
<pre>-phoneNumber: String -password: String</pre>
<pre>+setPhoneNumber(String phoneNumber): void +displayForm(): void +checkValid(String phoneNumber): boolean +showAlert(): void +setCode(String code): void +setPassword(String password): void</pre>

ChangePasswordController
<pre>-phoneNumber: String -password: String</pre>
<pre>+sendCode(String phoneNumber): String +checkCode(String code): boolean +changePassword(String phoneNumber, String code): boolean</pre>



UserSubsystem

-username: String
-phoneNumber: String
-gmail: String
-address: String
-newaccount: User
-user: User

+createAccount(String username, String phoneNumber, String gmail, String password): boolean
+checkDuplicate(User newaccount): boolean
+checkLogin(String username, String password): boolean
+changePassword(String phoneNumber): boolean
+updateUser(User user): User
+deleteUser(User user): boolean

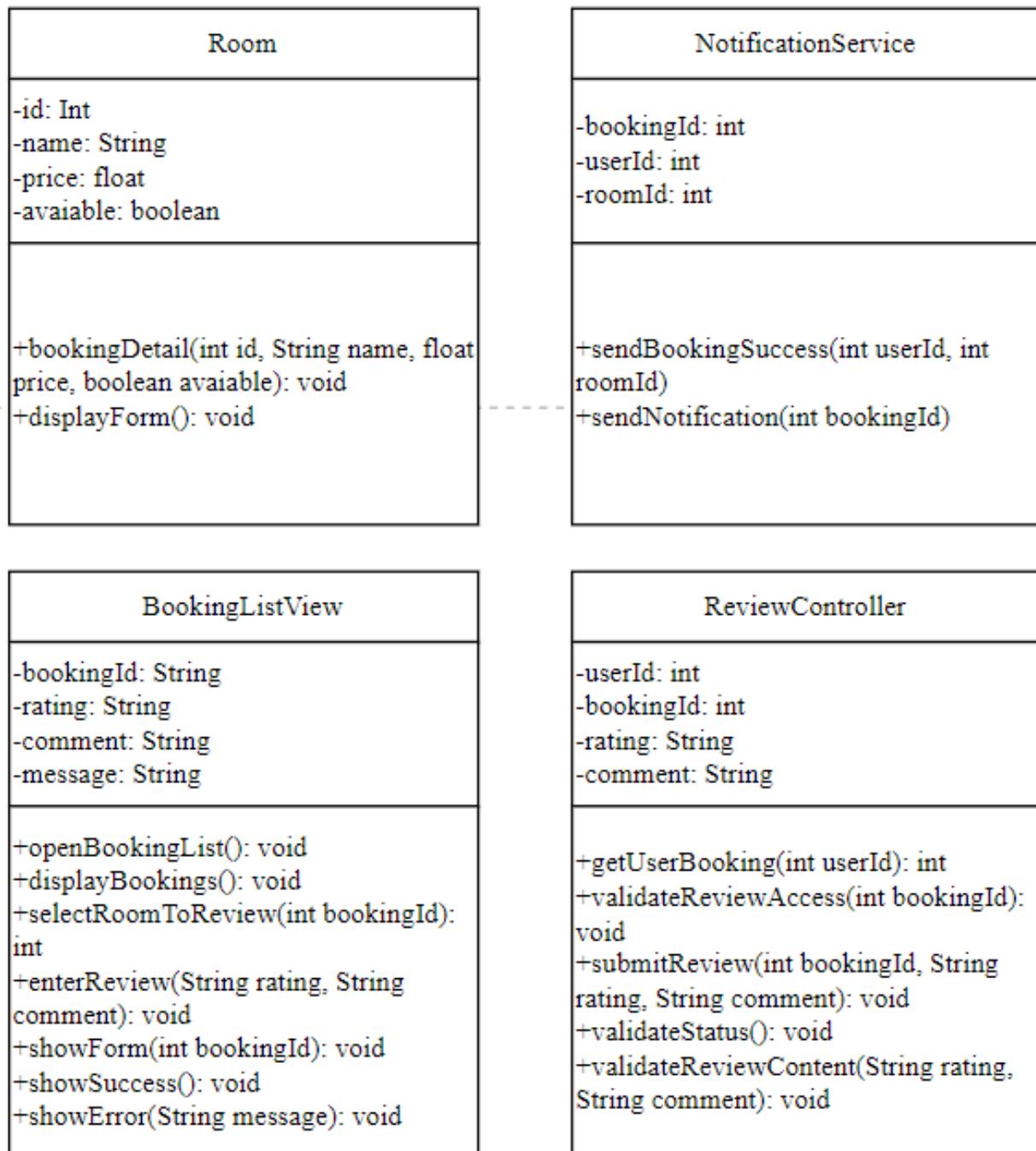
LanguageView

+chooseLanguage(): void
+displayLanguage(): void

LanguageController

+changeLanguage(): void

<p>RoomSearchForm</p> <p>-criteria: String -id: Int -room: Room</p> <p>+setCriteria(String criteria): void +selectRoom(int id): void +bookRoom(): void +displayRoom(): void +displayRoomDetails(Room room): void</p>	<p>BookingHistoryView</p> <p>-bookingId: Int -message: String -bookingsList: String</p> <p>+openBookingHistory(): void +showBookings(String bookingsList): void +selectBooking(int bookingId): void +cancelBooking(int bookingId): void +displayResult(String message)</p>
<p>BookingController</p> <p>-criteria: String -roomId: Int -userId: Int -bookingId: Int</p> <p>+submitCriteria(String criteria): void +requestRoomDetails(int roomId): void +processBooking(int userId, int roomId): boolean +requestBookingList(int userId): void +setBookingDetails(int booking Id): void +requestCancel(int bookingId): void</p>	



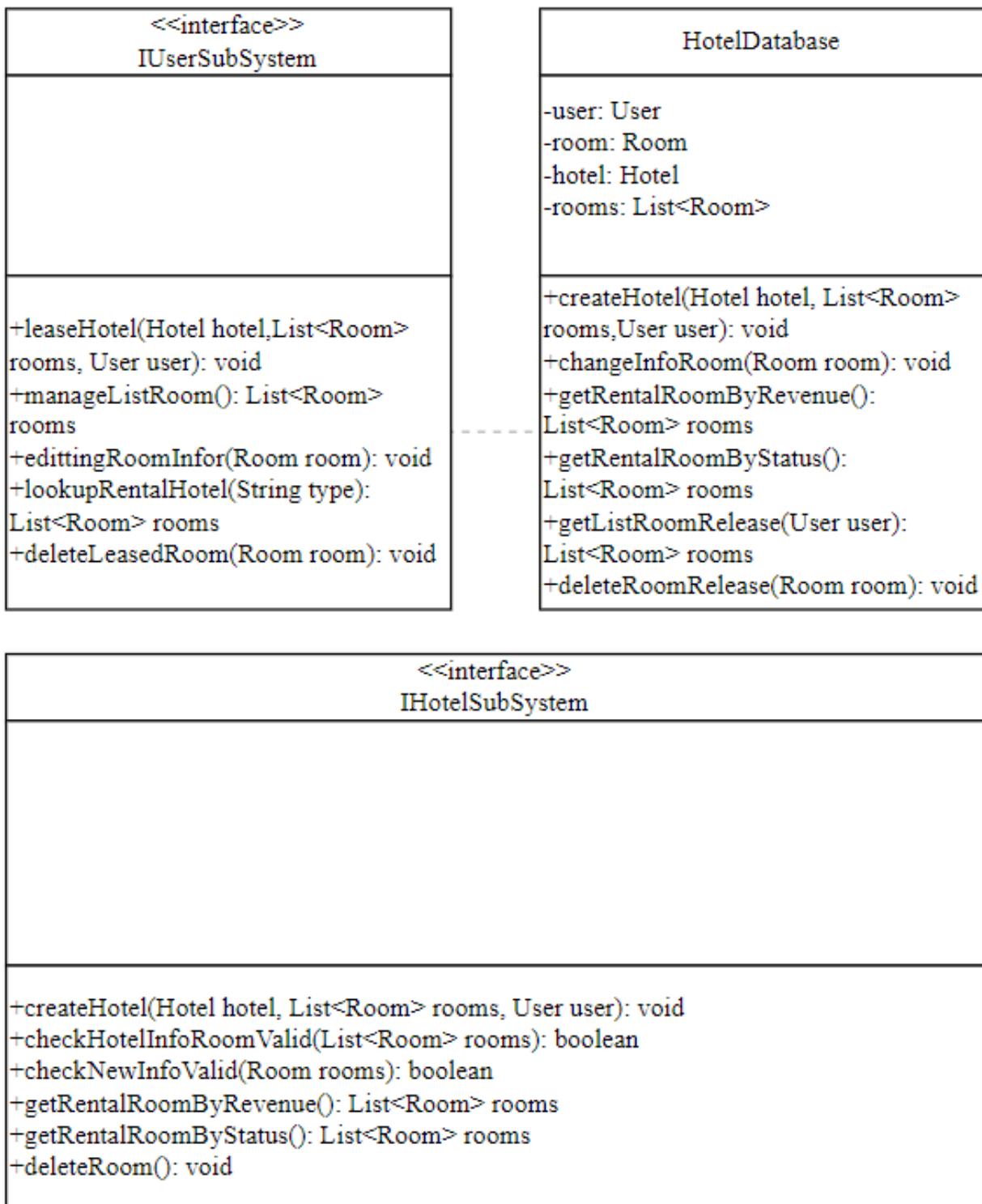
<p>RoomDetailView</p> <hr/> <p>-text: String</p> <hr/> <p>+openRoomDetail(): void +clickChatBotton(): void +displayChatUI(): void +sendMessage(string text): void +displayReply(): void</p>	<p>ChatController</p> <hr/> <p>-roomId: Int -userId: Int -hostId: Int -chatSession: String</p> <hr/> <p>+initiateChat(int roomId, int userId): void +getOrCreateChatSession(int userId, int hostId): int +openChatInterface(String chatSession): void</p>
<p>FavouriteRoomView</p> <hr/> <p>-roomId: int -room: String</p> <hr/> <p>+openFavouriteList(): void +showEmptyMessage() void +displayForm(): void +clickViewDetail(int roomId): void +clickUnfavorite(int roomId): void +showDetail(string room): void +showRemoveSuccess(): void</p>	<p>FavouriteController</p> <hr/> <p>-userId: int -roomId: int</p> <hr/> <p>+getFavouriteRooms(int userId): int +getRoomDetail(int roomId): int +removeFavourite(int userId, int roomId): void +updateFavourites(int userId): void</p>

LeaseHotelView
-hotel: Hotel
-user: User
-room: Room
+enterHotelInfor(Hotel hotel): void
+enterHotelOwnerInfor(User user): void
+enterHotelRoomInfo(List<Room> room): void

<<interface>>
IAuthenticationSubsystem
+checkHotelInforValid(): boolean
+checkHotelOwnerInforValid(): boolean

HotelView
-room: Room
-id: Int
+selectEdittingRoom(int id): Room room
+selectDeleteRoom(Room): void
+notifyStatus(): boolean
+notifyDeleteRoom(): void
+confirmEdittingRoom(): boolean
+confirmDeleteRoom(): boolean

SearchRentalHotelView
-rooms: List<Room>
+chooseTypeRentalHotel(): String
+displayInfoLookingUp(List<Room> rooms): void



PaymentView

```
-paymentID: int[]  
-paymentCode: int  
-bankName: string  
-ATMCardNumber: int
```

```
+enteredPaymentInfo( string bankName, string ATMCardNumber): string  
+enterPaymentCode(int paymentCode): int  
+confirmPayment(): boolean
```

Payment

```
-paymentID: int[]  
-paymentCode: int  
-bankName: string  
-ATMCardNumber: int
```

```
+getBankName(): string  
+getATMCardNumber(): int  
+getPaymentID(): int  
+setPaymentID(int paymentID): void  
+setBankName(string bankName): void  
+setATMCardNumber(int ATMCardNumber): void  
+getPaymentCode(): int  
+setPaymentCode(): int
```

<pre><<interface>> IPaymentSystem</pre>
<pre>+getPayment(int PaymentID): Payment +getPaymentList(string fromTime, string toTime): Payment +deletePayment(int PaymentID): void +withdraw(): void +createPayment(): void +checkIfUserPayFee(): void +verifyPaymentCode(): boolean</pre>

<pre>Order</pre>
<pre>-price: double -usingTime: string -orderID: int -address: string -OrdertType: string -OrderNumber: int -user: User -order_status = false +getOrder(int orderID): Order +getListOrder(User user): Order[] +getOrderID(): int +getUsingTime(): string +getAddress(): string +getPrice(): double +getOrderType(): string +getOrderNumber(): int +setOrderID(int orderID): void +setUsingTime(string usingTime): void +setAddress(string address): void +setPrice(double price): void +setOrderType(string ticketType): void +setOrderNumber(int ticketNumber): void +get_user_of_order(): User +set_user_of_order(User user): void +get_order_status(): boolean +updateOrderStatus(): void +saveNewOrder(Order newOrder): void</pre>

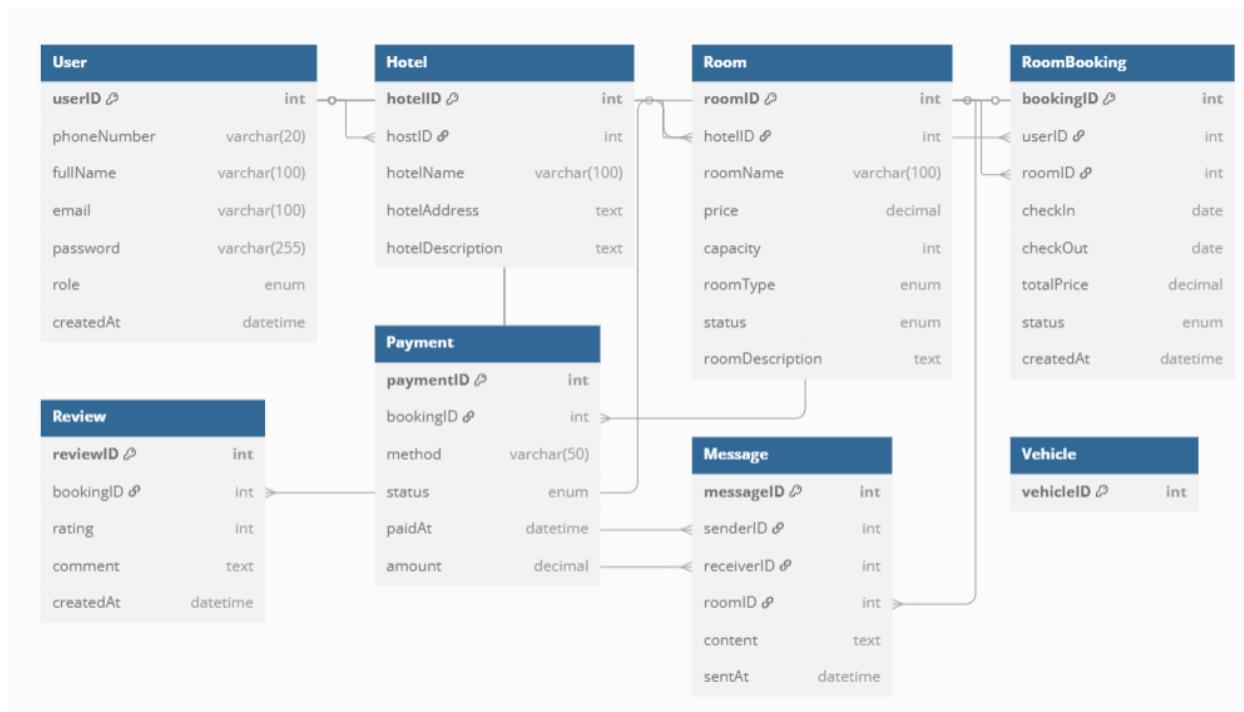
OrderView
<p>-order: Order -orderID: Int</p>
<p>+reviewOrder(Order order): string +chooseOrder(int orderID): Order</p>

<p><<interface>> IOrderSystem</p>
<p>+checkRegisterInfo(): boolean +checkOrderInfo(): boolean +updateOrderStatus(): void +askForConfirmRegister(): boolean +confirmRegister(): boolean +deleteOrder(int orderID): void +getOrder(int orderID): Order +getOrderList(string fromTime, string toTime): Order +reviewOrder(Order order): string +chooseOrder(int orderID): Order</p>

FindVehicleView
<p>-startAddress: String -endAddress: String</p>
<p>+enterDestinationAndVehicleRequest (String startAddress, String endAddress): void +chooseSpecificVehicle(): void</p>

<p><<interface>> IMapAPI</p>
<p>+findVehicle(): void +getSpecificVehicle(): routeVehicle</p>

6 Thiết kế Cơ sở dữ liệu



Bảng phân chia công việc

	Phạm Đức Thiện	Phạm Tuấn Việt	Đoàn Văn Tuyển	Phạm Văn Minh
Đặc tả	+Đặt vấn đề +Bảng thuật ngữ +Đặc tả bổ sung +Sơ đồ Usecase +Đặc tả Đăng ký tài khoản +Đặc tả Đăng nhập +Đặc tả Lựa chọn ngôn ngữ	+Đặc tả Cho thuê phòng +Đặc tả Sửa thông tin phòng +Đặc tả Tra cứu thông tin phòng đang cho thuê +Đặc tả Xóa phòng khách sạn	+Đặc tả Đặt phòng khách sạn +Đặc tả Theo dõi phòng đã đặt +Đặc tả Đánh giá phòng đã thuê +Đặc tả Liên hệ với chủ khách sạn +Đặc tả Xem danh sách phòng yêu thích	+Đặc tả Thanh toán +Đặc tả Tra cứu phương tiện

Phân tích	<ul style="list-style-type: none"> +Phân tích kiến trúc +Biểu đồ lớp, tuần tự Đăng ký tài khoản +Biểu đồ lớp, tuần tự Đăng nhập +Biểu đồ lớp, tuần tự Lựa chọn ngôn ngữ +Ánh xạ từ lớp phân tích tới cơ chế phân tích 	<ul style="list-style-type: none"> +Biểu đồ lớp, tuần tự Cho thuê phòng +Biểu đồ lớp, tuần tự Chính sửa thông tin phòng khách sạn +Biểu đồ lớp, tuần tự Tra cứu thông tin phòng cho thuê +Biểu đồ lớp, tuần tự Xóa phòng khách sạn 	<ul style="list-style-type: none"> +Biểu đồ lớp, tuần tự Đặt phòng khách sạn +Biểu đồ lớp, tuần tự Theo dõi phòng đã đặt +Biểu đồ lớp, tuần tự Đánh giá phòng đã thuê +Biểu đồ lớp, tuần tự Liên hệ với chủ khách sạn +Biểu đồ lớp, tuần tự Xem danh sách phòng yêu thích 	<ul style="list-style-type: none"> +Biểu đồ lớp, tuần tự Thanh toán +Biểu đồ lớp, tuần tự Tra cứu phương tiện
Thiết kế	<ul style="list-style-type: none"> +Analysis-to-Design-to-Implementation Mechanisms Map +Analysis-Class-to-Design-Element Map +Design-Element-to-Owning-Package Map +Biểu đồ lớp, tuần tự Đăng ký tài khoản +Biểu đồ lớp, tuần tự Đăng nhập +Biểu đồ lớp, tuần tự Lựa chọn ngôn ngữ 	<ul style="list-style-type: none"> +Subsystem Context +Biểu đồ lớp, tuần tự Cho thuê phòng +Biểu đồ lớp, tuần tự Chính sửa thông tin phòng khách sạn +Biểu đồ lớp, tuần tự Tra cứu thông tin phòng cho thuê +Biểu đồ lớp, tuần tự Xóa phòng khách sạn +Thiết kế Hệ thống con 	<ul style="list-style-type: none"> +Packages and Their Dependency +Mô tả kiến trúc thực thi +Biểu đồ lớp, tuần tự Đặt phòng khách sạn +Biểu đồ lớp, tuần tự Theo dõi phòng đã đặt +Biểu đồ lớp, tuần tự Đánh giá phòng đã thuê +Biểu đồ lớp, tuần tự Liên hệ với chủ khách sạn +Biểu đồ lớp, tuần tự Xem danh sách phòng yêu thích 	<ul style="list-style-type: none"> +Biểu đồ lớp, tuần tự Thanh toán +Biểu đồ lớp, tuần tự Tra cứu phương tiện +Thiết kế lớp +Thiết kế Cơ sở dữ liệu
Hệ số	25%	25%	25%	25%