TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

======\*\*\*======



BÁO CÁO ĐỒ ÁN TỐT NGHIỆP

NGÀNH CÔNG NGHỆ THÔNG TIN

**ĐỀ TÀI: XÂY DỰNG WEBSITE QUẢN LÝ KHÁCH SẠN VÀ ĐẶT PHÒNG TRỰC TUYẾN CHO CÔNG TY BÌNH MINH PALACE**

|  |  |
| --- | --- |
| GVHD | : TS. Phạm Văn Hiệp |
| Sinh viên | : Vũ Đức Doanh |
| Mã sinh viên | : 2021603155 |
| Mã lớp  Khóa | : 20242IT6129001  :16 |

Hà Nội, Năm 2025

LỜI CẢM ƠN

Trong thời gian làm đồ án tốt nghiệp, em đã nhận được nhiều sự giúp đỡ, đóng góp ý kiến và chỉ bảo nhiệt tình của thầy cô, gia đình và bạn bè đã giúp em hoàn thiện đồ án tốt nghiệp.

Trước tiên em xin gửi tới các thầy cô khoa Công nghệ thông tin trường Đại học Công Nghiệp Hà Nội lời chúc sức khỏe và lời cảm ơn chân thành. Với sự quan tâm, chỉ bảo của thầy cô đã tạo điều kiện cho em trong suốt quá trình học tập và hoàn thành đồ án tốt nghiệp lần này.

Đặc biệt em xin gửi lời cảm ơn chân thành nhất đến thầy Phạm Văn Hiệp đã trực tiếp tận tình hướng dẫn cũng như nhận xét và giúp đỡ em trong suốt quá trình hoàn thiện đồ án. Những gì chúng em nhận được không chỉ dừng lại ở kiến thức môn học mà nhiều hơn thế đó là những lời khuyên, chia sẻ thực tế từ thầy. Với khoảng thời gian của học phần, em đã cố gắng để hoàn thành thật tốt đề tài.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của một sinh viên, đồ án này không tránh khỏi những thiếu sót. Em rất mong nhận được sự giúp đỡ của thầy cô và các bạn để hoàn thiện hơn đồ án của mình.

Em xin chân thành cảm ơn! Em xin kính chúc các thầy, cô thật nhiều sức khoẻ để cống hiến nhiều hơn trong sự nghiệp giảng dạy.

|  |  |
| --- | --- |
|  | Hà Nội, ngày tháng năm 2025  Sinh viên thực hiện  Vũ Đức Doanh |

Mục lục

[LỜI CẢM ƠN 2](#_Toc195385499)

[DANH MỤC HÌNH ẢNH 7](#_Toc195385500)

[DANH MỤC BẢNG BIỂU 9](#_Toc195385501)

[MỞ ĐẦU 10](#_Toc195385502)

[1. Lý do chọn đề tài 10](#_Toc195385503)

[2. Mục đích chọn đề tài 10](#_Toc195385504)

[3. Đối tượng và phạm vi nghiên cứu 11](#_Toc195385505)

[4. Ý nghĩa khoa học và thực tiễn của đề tài 11](#_Toc195385506)

[Chương 1. TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU 12](#_Toc195385507)

[1.1 Giới thiệu chung 12](#_Toc195385508)

[1.2 Nghiên cứu hiện trạng và giải quyết vấn đề 12](#_Toc195385509)

[1.2.1 Khảo sát sơ bộ hệ thống 12](#_Toc195385510)

[1.2.2 Khảo sát chi tiết hệ thống 15](#_Toc195385511)

[1.3 Khảo sát các mô hình thương mại điện tử từ truyền thống đến hiện đại 17](#_Toc195385512)

[1.3.1 Mô hình thường mại điện tử truyền thống 17](#_Toc195385513)

[1.3.2 Mô hình thường mại điện tử hiện đại 18](#_Toc195385514)

[1.4 Giới thiệu về ngôn ngữ và công nghệ 19](#_Toc195385515)

[1.4.1 ReactJS 19](#_Toc195385516)

[1.4.2 Spring Boot 20](#_Toc195385517)

[1.4.3 MySQL 22](#_Toc195385518)

[Chương 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 24](#_Toc195385519)

[2.1 Xác định các tác nhân, ca sử dụng 24](#_Toc195385520)

[2.1.1 Tác nhân 24](#_Toc195385521)

[2.1.2 Use Case 24](#_Toc195385522)

[2.2 Các biểu đồ 26](#_Toc195385523)

[2.2.1 Biểu đồ use case chính 26](#_Toc195385524)

[2.2.2 Biều đồ use case phân rã 26](#_Toc195385525)

[2.3 Mô tả chi tiết các use case 28](#_Toc195385526)

[2.3.1 Mô tả use case Dang Nhap 28](#_Toc195385527)

[2.3.2 Mô tả use case Dang Ky 28](#_Toc195385528)

[2.3.3 Mô tả use case Thong Tin Ca Nhan 30](#_Toc195385529)

[2.3.4 Mô tả use case Xem Chi Tiet Phong 30](#_Toc195385530)

[2.3.5 Mô tả use case Danh Gia Phong 31](#_Toc195385531)

[2.3.6 Mô tả use case Dat Phong 31](#_Toc195385532)

[2.3.7 Mô tả use case Quan Ly Loai Phong 32](#_Toc195385533)

[2.3.8 Mô tả use case Quan Ly Phong 34](#_Toc195385534)

[2.3.9 Mô tả use case Quan Ly Dat Phong 36](#_Toc195385535)

[2.3.10 Mô tả use case Quan Ly Thanh Toan 37](#_Toc195385536)

[2.3.11 Mô tả use case Quan Ly Nguoi Dung 38](#_Toc195385537)

[2.3.12 Mô tả use case Thong Ke 38](#_Toc195385538)

[2.4 Biểu đồ trình tự 39](#_Toc195385539)

[2.4.1 Use case Danh Nhap 39](#_Toc195385540)

[2.4.2 Use case Dang Ky 40](#_Toc195385541)

[2.4.3 Use case Thong Tin Ca Nhan 41](#_Toc195385542)

[2.4.4 Use case Xem Chi Tiet Phong 42](#_Toc195385543)

[2.4.5 Use case Dat Phong 43](#_Toc195385544)

[2.4.6 Use case Quan Ly Phong 45](#_Toc195385545)

[2.4.7 Use case Quan Ly Dat Phong 47](#_Toc195385546)

[2.4.8 Use case Quan Ly Thanh Toan 48](#_Toc195385547)

[2.4.9 Use case Quan Ly Nguoi Dung (Khach Hang) 50](#_Toc195385548)

[2.4.10 Use case Thong Ke 52](#_Toc195385549)

[2.5 Biểu đồ lớp chi tiết 53](#_Toc195385550)

[2.5.1 Use case Use case Dang Nhap 53](#_Toc195385551)

[2.5.2 Use case Dang Ky 53](#_Toc195385552)

[2.5.3 Usecase Thong Tin Ca Nhan 54](#_Toc195385553)

[2.5.4 Use case Xem Chi Tiet Phong 54](#_Toc195385554)

[2.5.5 Use case Dat Phong 55](#_Toc195385555)

[2.5.6 Use case Quan Ly Phong 55](#_Toc195385556)

[2.5.7 Use case Quan Ly Dat Phong 56](#_Toc195385557)

[2.5.8 Use case Quan Ly Thanh Toan 57](#_Toc195385558)

[2.5.9 Use case Quan Ly Nguoi Dung (Khach Hang) 57](#_Toc195385559)

[2.5.10 Use case Thong Ke 58](#_Toc195385560)

[2.6 Thiết kế Cơ sở dữ liệu 58](#_Toc195385561)

[2.6.1 Mô hình hóa dữ liệu 58](#_Toc195385562)

[2.6.2 Sơ đồ dữ liệu quan hệ 61](#_Toc195385563)

[2.6.3 Chi tiết các bảng dữ liệu 61](#_Toc195385564)

[2.7 Thiết kế giao diện hệ thống 64](#_Toc195385565)

[2.7.1 Use case Dang Nhap 64](#_Toc195385566)

[2.7.2 Use case Dang Ky 65](#_Toc195385567)

[2.7.3 Use case Thong Tin Ca Nhan 65](#_Toc195385568)

[2.7.4 Use case Xem Chi Tiet Phong 65](#_Toc195385569)

[2.7.5 Use case Dat Phong 66](#_Toc195385570)

[2.7.6 Use case Quan Ly Phong 66](#_Toc195385571)

[2.7.7 Use case Quan Ly Dat Phong 67](#_Toc195385572)

[2.7.8 Use case Quan Ly Thanh Toan 67](#_Toc195385573)

[2.7.9 Use case Qaun Ly Nguoi Dung (Khach Hang) 68](#_Toc195385574)

[2.7.10 Use case Quan Ly Thong Ke 68](#_Toc195385575)

# DANH MỤC HÌNH ẢNH

[Hình 2.1 Biểu đồ use casse chính 28](#_Toc184995189)

[Hình 2.2 Biểu đồ use casse của khách hàng 28](#_Toc184995190)

[Hình 2.3 Biểu đồ use case của nhân viên 29](#_Toc184995191)

[Hình 2.4 Biểu đồ use case của người quản trị và người quản lý 29](#_Toc184995192)

[Hình 2.5 PTTK Use case Đăng nhập 42](#_Toc184995193)

[Hình 2.6 PTTK Use case Đăng ký 43](#_Toc184995194)

[Hình 2.7 PTTK Use case Thông tin cá nhân 44](#_Toc184995195)

[Hình 2.8 PTTK Use case Xem chi tiết phòng 45](#_Toc184995196)

[Hình 2.9 PTTK Use case Đặt phòng 46](#_Toc184995197)

[Hình 2.10 PTTK Use case Quản lý phòng 48](#_Toc184995198)

[Hình 2.11 PTTK Use case Quản lý đặt phòng 50](#_Toc184995199)

[Hình 2.12 PTTK Use case Quản lý thanh toán 51](#_Toc184995200)

[Hình 2.13 PTTK Use case Quản lý người dùng 53](#_Toc184995201)

[Hình 2.14 PTTK Use case Thống kê 54](#_Toc184995202)

[Hình 2.15 Biểu đồ lớp Đăng nhập 55](#_Toc184995203)

[Hình 2.16 Biểu đồ lớp Đăng ký 55](#_Toc184995204)

[Hình 2.17 Biểu đồ lớp Thông tin cá nhân 56](#_Toc184995205)

[Hình 2.18 Biểu đồ lớp Xem chi tiết phòng 56](#_Toc184995206)

[Hình 2.19 Biểu đồ lớp Đặt phòng 57](#_Toc184995207)

[Hình 2.20 Biểu đồ lớp Quản lý phòng 57](#_Toc184995208)

[Hình 2.21 Biểu đồ lớp Quản đặt phòng 58](#_Toc184995209)

[Hình 2.22 Biểu đồ lớp Quản lý thanh toán 59](#_Toc184995210)

[Hình 2.23 Biểu đồ lớp Quản lý người dùng 59](#_Toc184995211)

[Hình 2.24 Biểu đồ lớp Thống kê 60](#_Toc184995212)

[Hình 2.25 Sơ đồ dữ liệu quan hệ 63](#_Toc184995213)

[Hình 2.26 Các bảng trong CSDL 66](#_Toc184995214)

[Hình 2.27 Hình hình dung màn hình đăng nhập 66](#_Toc184995215)

[Hình 2.28 Hình hình dung màn hình đăng ký 67](#_Toc184995216)

[Hình 2.29 Hình hình dung màn hình thông tin cá nhân 67](#_Toc184995217)

[Hình 2.30 Hình hình dung màn hình xem chi tiết phòng 67](#_Toc184995218)

[Hình 2.31 Hình hình dung màn hình đặt phòng 68](#_Toc184995219)

[Hình 2.32 Hình hình dung màn hình quản lý phòng 68](#_Toc184995220)

[Hình 2.33 Hình hình dung màn hình quản lý đặt phòng 69](#_Toc184995221)

[Hình 2.34 Hình hình dung màn hình quản lý thanh toán 69](#_Toc184995222)

[Hình 2.35 Hình hình dung màn hình quản lý người dùng 70](#_Toc184995223)

[Hình 2.36 Hình hình dung màn hình thống kê 70](#_Toc184995224)

# DANH MỤC BẢNG BIỂU

[Bảng 1.1 Bảng Kế hoạch phỏng vấn 14](#_Toc170038846)

[Bảng 1.2 Bảng phỏng vấn 16](#_Toc170038847)

# MỞ ĐẦU

## Lý do chọn đề tài

Ngày nay, Công nghệ thông tin có vai trò hết sức quan trọng, là động lực thúc đẩy tăng trưởng kinh tế và kích thích đổi mới nền kinh tế toàn cầu nói chung và nền kinh tế nước ta nói riêng. Bước vào thời đại công nghiệp 4.0, công nghệ thông tin đã có các cải tiến vượt bậc. Ngày nay, các thiết bị điện tử đã không còn là những phương tiện quý hiếm, mà ngược lại đã trở thành phương tiện giải trí, công cụ làm việc tiện lợi của con người mọi lúc, mọi nơi. Việc phổ thông hóa các thiết bị điện tử đã mở ra cho nhân loại một kỷ nguyên mới – kỷ nguyên công nghệ thông tin.

Tại khách sạn vấn đề quản lý phòng là cần thiết và cấp bách. Một khách sạn muốn mang lại trải nghiệmtốt nhất cho khách hàng thì cần nắm rõ được thời gian cũng như mong muốn của khách hàng về phòng ở, nghỉ dưỡng. Quản lý khách sạn thủ công chỉ áp dụng đối với các khách sạn nhỏ. Đối với các khách sạn lớn, người quản lý cần thay đổi phương pháp và cách thức quản lý của khách sạn hiệu quả. Đây là lý do tại sao các nhà quản lý khách sạn cần đến sự hỗ trợ đắc lực của phần mềm quản lý khách sạn.

Với tư cách là một sinh viên của khoa Công nghệ thông tin việc áp dụng công nghệ vào đặt phòng và quản lý khách sạn để giúp cho thấy được nhiều lợi ích nhờ việc áp dụng công nghệ thông tin vào thực tế. Chính vì vậy, em đã quyết định lựa chọn đề tài “**Xây dựng website quản lý khách sạn và đặt phòng trực tuyến**” để thực hiện đồ án của mình với mục đích nghiên cứu hệ thống website đặt phòng, giúp cho việc đặt phòng được thuận tiện hơn nhờ việc áp dụng công nghệ thông tin.

## Mục đích chọn đề tài

Tạo ra một nền tảng trực tuyến quản lý khách sạn và đặt phòng, cho phép khách hàng tìm kiếm thông tin phòng và đặt chỗ một cách nhanh chóng và tiện lợi. Cung cấp công cụ quản lý hiệu quả cho các khách sạn. Hỗ trợ các chủ khách sạn trong việc quản lý phòng, cập nhật tình trạng và xử lý yêu cầu đặt phòng nhanh chóng, từ đó nâng cao chất lượng phục vụ khách hàng.

## Đối tượng và phạm vi nghiên cứu

Đối tượng chính của hệ thống quản lý khách sạn và đặt phòng trực tuyến là khách hàng có nhu cầu tìm kiếm và đặt phòng, cùng với các khách sạn cần một công cụ quản lý đặt phòng hiệu quả. Phạm vi nghiên cứu tập trung vào các khách sạn trong khu vực nội địa, giúp xây dựng một cộng đồng lưu trú trực tuyến tiện lợi và an toàn.

## Ý nghĩa khoa học và thực tiễn của đề tài

Ý nghĩa khoa học: Xây dựng hệ thống quản lý khách sạn và đặt phòng trực tuyến cung cấp kiến thức về phát triển và quản lý dịch vụ trực tuyến trong ngành du lịch. Đề tài cũng giúp phân tích xu hướng đặt phòng và đưa ra các biện pháp quảng bá hiệu quả, góp phần vào sự phát triển của ngành.

Ý nghĩa thực tiễn: Trang web tạo ra một kênh đặt phòng tin cậy và tiện lợi cho khách hàng, đồng thời hỗ trợ khách sạn trong việc quảng bá dịch vụ và tối ưu hóa hoạt động kinh doanh, từ đó nâng cao sự hài lòng của khách hàng và thúc đẩy phát triển du lịch địa phương.

# TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU

## Giới thiệu chung

Hiện nay, với sự phát triển mạnh mẽ của ngành du lịch và nhu cầu ngày càng cao về việc đặt phòng khách sạn trực tuyến, việc quản lý khách sạn hiệu quả trở thành một yếu tố quan trọng để đảm bảo chất lượng dịch vụ và thu hút khách hàng. Em nhận thấy rằng việc cung cấp một nền tảng trực tuyến tiện ích, giúp khách hàng dễ dàng tìm kiếm, đặt phòng và quản lý các dịch vụ khách sạn là rất cần thiết. Vì vậy, Em tập trung vào việc xây dựng website quản lý khách sạn và đặt phòng trực tuyến, nơi khách hàng có thể tìm kiếm thông tin về các khách sạn, lựa chọn phòng theo nhu cầu và thực hiện thanh toán trực tuyến một cách thuận tiện.

## Nghiên cứu hiện trạng và giải quyết vấn đề

### Khảo sát sơ bộ hệ thống

#### Mục tiêu

Tìm hiểu hoạt động quản lý phòng, dịch vụ của các khách sạn hiện nay.

Nghiên cứu các hệ thống quản lý khách sạn đã có, ví dụ như hệ thống quản lý của các khách sạn lớn hoặc các công ty cung cấp dịch vụ đặt phòng trực tuyến (marriott.com…).

Đánh giá lý do cần xây dựng một hệ thống website quản lý khách sạn và đặt phòng trực tuyến để đáp ứng nhu cầu của khách hàng. Hệ thống này sẽ giúp khách hàng tìm kiếm, lựa chọn và đặt phòng dễ dàng, trong khi người quản lý khách sạn có thể theo dõi tình trạng phòng, đặt phòng, và thống kê doanh thu.

#### Phương pháp

|  |  |
| --- | --- |
| **Kế hoạch phỏng vấn** | |
| Người được hỏi: Nguyễn Thị Lan | Người phỏng vấn: Nguyễn Tử Trung |
| Địa chỉ: Xã Thượng Trưng, huyện Vĩnh Tường, tỉnh Vĩnh Phúc. | Thời gian hẹn: Ngày 1/03/2025  Thời gian bắt đầu: 14h  Thời gian kết thúc: 15h |
| Đối tượng:  Người quản lý tại Bình Minh Palace.  Khảo sát kỹ lưỡng, không bỏ sót thông tin, Các thông tin thu thập được phải chính xác, không gây ác cảm với người được phỏng vấn | Các yêu cầu đòi hỏi:  Vai trò: Quản lý khách sạn  Vị trí: Quản lý  Kinh nghiệm: > 3 năm kinh nghiệm |
| Chương trình:  Giới thiệu: buổi phỏng vấn nhằm thu thập thông tin chi tiết để xây dựng website  Tổng quan về dự án: thu thập và phân tích các thông tin cần thiết để xây dựng trang web  Tổng quan về phỏng vấn: nội dung các câu hỏi phỏng vấn sẽ xoay quanh các chi tiết và đặc điểm của cửa hàng cũng như các yêu cầu đặt ra.  Chủ đề 1: Câu hỏi và trả lời  Chủ đề 2: Câu hỏi và trả lời  Tập hợp các nội dung chính  Ý kiến người được hỏi  Kết thúc (thỏa thuận) | Ước lượng thời gian  1 phút    2 phút    1 phút    10 phút  10 phút  2 phút  5 phút  2 phút |
|  | Dự kiến: tổng cộng 33 phút |

Bảng . Bảng Kế hoạch phỏng vấn

|  |  |
| --- | --- |
| **Phiếu phỏng vấn** | |
| Tên dự án: Xây dựng Website | |
| Tên tiểu dự án: Xây dựng website quản lý khách sạn và đặt phòng | |
| Người được hỏi: Nguyễn Thị Lan | Người hỏi: Nguyễn Tử Trung |
| **Câu hỏi**  Câu 1: Khách sạn của bạn cung cấp những loại dịch vụ và phòng nào?  Câu 2: Anh/chị muốn phân loại các loại phòng trong khách sạn như thế nào (theo giá, loại phòng, hoặc số lượng giường)?  Câu 3: Anh/chị có muốn cho phép khách hàng đánh giá và phản hồi về các dịch vụ, phòng ở của khách sạn sau khi sử dụng dịch vụ không?  Câu 4: Khách sạn có cho phép khách hàng thanh toán trực tuyến qua website không? | **Ghi chú**  Câu 1:  Khách sạn cung cấp các loại phòng như phòng đơn, phòng đôi, phòng gia đình và phòng cao cấp, cùng với dịch vụ ăn sáng, spa, và đưa đón sân bay.  Câu 2:  Chúng tôi phân loại phòng theo giá, theo các phòng được sale và sắp sếp theo giường  Câu 3:  Khách hàng có thể đánh giá dịch vụ và phòng ở sau khi sử dụng, giúp chúng tôi cải thiện chất lượng dịch vụ.  Câu 4:  Khách sạn hỗ trợ thanh toán trực tuyến qua website bằng thẻ tín dụng hoặc các phương thức thanh toán điện tử khác. |
| Đánh giá chung:  Người được phỏng vấn trả lời một cách chính xác và đầy đủ các câu hỏi.  Thông qua câu trả lời của người được phỏng vấn, có thể tổng hợp được một số thông tin quan trọng để xây dựng website. | |

Bảng . Bảng phỏng vấn

#### Kết quả sơ bộ

Về giao diện của trang web:

Hiển thị trang chủ đẹp, bắt mắt, màu sắc phù hợp, thu hút người dùng.

Các chức năng sử dụng được hiển thị công khai, dễ thấy.

Về chức năng:

* Quản trị viên:

Quản lý hoạt động của trang web, quản lý 1 số thông tin cơ bản của trang web.

Quản lý loại phòng, thông tin phòng,đặt phòng, thanh toán, theo dõi doanh thu, mã giảm giá, đánh giá, liên hệ đối với người dùng và người dùng.

* Khách hàng:

Có thể đăng ký, đăng nhập, đăng xuất, quên mật khẩu, đổi mật khẩu, xem thông tin về các phòng, đặt phòng và thanh toán online bằng VNPay, áp dụng mã giảm giá, xem thông tin đặt phòng, lấy mã giảm giá từ chatbot tự động

### Khảo sát chi tiết hệ thống

#### Hoạt động của hệ thống

Hệ thống quản lý khách sạn và đặt phòng trực tuyến cung cấp nền tảng cho khách hàng để đăng ký tài khoản, xem thông tin các loại phòng, thực hiện đặt phòng và thanh toán trực tuyến, đồng thời hỗ trợ tư vấn trực tuyến để đảm bảo khách hàng có trải nghiệm đặt phòng thuận tiện và chính xác. Các hoạt động bao gồm:

Tạo tài khoản: Khách hàng có thể đăng ký tài khoản miễn phí trên hệ thống. Để thực hiện, họ cần cung cấp thông tin cá nhân như tên, địa chỉ, số điện thoại và email qua một biểu mẫu trực tuyến. Sau khi đăng ký, khách hàng sẽ có tài khoản riêng để quản lý đặt phòng.

Đặt phòng, tìm kiếm phòng trống: Khách hàng có thể tìm kiếm phòng trống và xem thông tin chi tiết các phòng. Sau khi tìm kiếm phòng trống người dùng có thể chọn phòng phù hợp và thực hiện thanh toán trực tuyến bằng các phương thức thanh toán hỗ trợ.

#### Các yêu cầu chức năng và dữ liệu

Người dùng có thể vào website và thực hiện:

* Xem thông tin chi tiết các phòng
* Tìm kiếm phòng theo loại phòng, giá cả hoặc các tiêu chí khác.
* Đăng ký tài khoản người dùng.
* Đăng nhập, quản lý thông tin cá nhân, đổi mật khẩu và xem lịch sử đặt phòng.
* Thực hiện thanh toán online cho phòng đặt.
* Nhận mã giảm giá trực tiếp từ chatbot

Hệ thống phía quản trị cần hỗ trợ các chức năng về:

* Đăng nhập và đăng ký cho quản trị viên.
* Quản lý danh mục phòng, thông tin phòng.
* Quản lý đơn đặt phòng, thông tin thanh toán
* Quản lý mã giảm giá, đánh giá của người dùng
* Quản lý người dùng (Vô hiệu hóa người dùng khi cần thiết)
* Thống kê dữ liệu về đặt phòng và doanh thu.

#### Các yêu cầu phi chức năng

Giao diện dễ sử dụng: Hệ thống cần có giao diện người dùng thân thiện, dễ sử dụng giúp khách hàng dễ dàng tìm kiếm và đặt phòng.

Tích hợp dễ dàng: Hệ thống cần dễ dàng tích hợp với các phương thức thanh toán trực tuyến, và chatbot

Bảo mật dữ liệu: Đảm bảo bảo mật cho thông tin cá nhân của người dùng, bao gồm thông tin tài khoản và lịch sử đặt phòng và mã hóa password của người dùng.

Tương thích đa nền tảng: Trang web cần hoạt động tốt trên các thiết bị khác nhau như máy tính, điện thoại di động và máy tính bảng.

## Khảo sát các mô hình thương mại điện tử từ truyền thống đến hiện đại

### Mô hình thường mại điện tử truyền thống

Các mô hình thương mại điện tử truyền thống chủ yếu phát triển từ những ngày đầu của internet và tập trung vào việc mua bán sản phẩm thông qua các kênh online như website nhưng hạn chế về trải nghiệm người dùng và tính tương tác.

**Mô hình B2C (Business-to-Consumer)**

Đây là mô hình thương mại điện tử phổ biến, trong đó doanh nghiệp bán sản phẩm hoặc dịch vụ trực tiếp cho người tiêu dùng. Các ví dụ nổi bật là Amazon, eBay, hoặc các cửa hàng trực tuyến của các công ty bán lẻ. Các giao dịch chủ yếu là mua hàng và thanh toán qua các phương thức truyền thống như thẻ tín dụng, chuyển khoản ngân hàng.

**Mô hình B2B (Business-to-Business)**

Doanh nghiệp bán hàng hóa hoặc dịch vụ đang kinh doanh cho các doanh nghiệp khác. Đây là mô hình thương mại điện tử giữa các tổ chức, thường là các đơn hàng với số lượng lớn và giá trị cao.

**Mô hình C2C (Consumer-to-Consumer)**

Các cá nhân bán sản phẩm hoặc dịch vụ cho những người tiêu dùng khác, thường thông qua nền tảng trung gian như eBay, ..

### Mô hình thường mại điện tử hiện đại

Thương mại điện tử hiện đại ngày nay đã phát triển vượt bậc nhờ sự tích hợp của các công nghệ mới như trí tuệ nhân tạo (AI), dữ liệu lớn (Big Data), thanh toán điện tử, và các mô hình kinh doanh đa kênh. Các mô hình hiện đại chủ yếu tập trung vào việc tối ưu hóa trải nghiệm người dùng và sử dụng công nghệ để nâng cao hiệu quả hoạt động.

**Mô hình D2C (Direct-to-Consumer)**

Doanh nghiệp bán sản phẩm trực tiếp cho người tiêu dùng mà không qua các kênh trung gian. Ví dụ: Nike, Warby Parker.

Các công ty có thể kiểm soát trải nghiệm khách hàng từ đầu đến cuối, từ việc phát triển sản phẩm đến việc giao hàng.

Thanh toán qua các phương thức hiện đại như ví điện tử (PayPal, Apple Pay), thanh toán qua điện thoại di động hoặc thanh toán qua blockchain.

**Mô hình B2B2C (Business-to-Business-to-Consumer)**

Mô hình này kết hợp các doanh nghiệp và các doanh nghiệp nhỏ hơn với mục tiêu cuối cùng là phục vụ người tiêu dùng. Các ví dụ như Amazon B2B2C, Alibaba cung cấp nền tảng để các doanh nghiệp nhỏ bán hàng trực tiếp đến người tiêu dùng.

Các doanh nghiệp lớn tạo ra nền tảng cho các nhà cung cấp bán hàng trực tiếp tới người tiêu dùng cuối cùng.

Giúp các doanh nghiệp nhỏ tiếp cận thị trường rộng lớn mà không cần phải tự xây dựng nền tảng của riêng mình.

**Mô hình Social Commerce (Thương mại qua mạng xã hội)**

Các nền tảng mạng xã hội như Facebook, Instagram, và TikTok đã phát triển các tính năng cho phép người tiêu dùng mua hàng trực tiếp từ các bài đăng, quảng cáo, hoặc livestream của các thương hiệu.

Hệ thống website quản lý khách sạn và đặt phòng trực tuyến của em đang hướng theo mô hình thương mại điện tử hiện đại được thiết kế để mang đến sự tiện lợi cho khách hàng. Hệ thống hỗ trợ thanh toán online qua VNPAY, giúp bạn dễ dàng thanh toán trực tiếp khi đặt phòng mà không cần sử dụng tiền mặt, phù hợp với xu hướng tiêu dùng hiện nay.

Ưu điểm của hệ thống là thanh toán nhanh chóng, an toàn và dễ dàng tích hợp trên các thiết bị di động. Bên cạnh đó, chatbot sẽ giúp hỗ trợ trong quá trình tìm kiếm thông tin và giải đáp thắc mắc một cách nhanh chóng.

Tuy nhiên, hệ thống có thể phụ thuộc vào chất lượng kết nối internet và khả năng sử dụng công nghệ của người dùng. Nhưng với những tính năng này, hy vọng sẽ mang lại trải nghiệm tốt nhất cho khách hàng.

## Giới thiệu về ngôn ngữ và công nghệ

### ReactJS

ReactJS được phát triển bởi Facebook và được giới thiệu lần đầu tiên vào năm 2011. Ban đầu, ReactJS được phát triển để xây dựng giao diện người dùng trên trang web Facebook, nhằm cải thiện tốc độ và hiệu suất của ứng dụng web.

Tuy nhiên, ReactJS không được công bố cho cộng đồng phát triển cho đến năm 2013, khi Facebook công bố mã nguồn mở của nó và giới thiệu cho cộng đồng lập trình viên. Từ đó, ReactJS nhanh chóng trở thành một trong những thư viện phổ biến nhất để phát triển các ứng dụng web động.

Sau đó, vào năm 2015, Facebook giới thiệu phiên bản React Native, một framework phát triển ứng dụng di động sử dụng ReactJS. React Native cho phép các nhà phát triển xây dựng các ứng dụng di động cho cả iOS và Android sử dụng cùng một mã nguồn, tương tự như ReactJS trên web.

Hiện nay, ReactJS đã trở thành một trong những thư viện phát triển web phổ biến nhất, được sử dụng rộng rãi bởi các công ty lớn và nhỏ trên toàn thế giới. Facebook cũng tiếp tục đầu tư phát triển và nâng cấp ReactJS để đáp ứng nhu cầu của cộng đồng phát triển .

Ưu điểm:

Hiệu suất cao: Sử dụng Virtual DOM, ReactJS giúp cập nhật trang nhanh chóng và hiệu quả.

Tái sử dụng mã nguồn: Các thành phần UI có thể tái sử dụng nhiều lần, giảm thời gian và chi phí phát triển.

Tốt cho SEO: Hỗ trợ tối ưu SEO nhờ các thư viện như React Helmet để quản lý meta và title.

Đa nền tảng: Có thể phát triển ứng dụng web và di động (iOS, Android) với React Native từ cùng một mã nguồn.

Nhược điểm:

Khó học cho người mới bắt đầu: Cần hiểu nhiều khái niệm như JSX, Hooks, và quản lý trạng thái.

Quản lý ứng dụng phức tạp: Khi ứng dụng phát triển lớn, việc quản lý trạng thái và dữ liệu giữa các component có thể trở nên phức tạp, đòi hỏi sử dụng các thư viện bổ trợ như Redux hay Context API

### Spring Boot

Spring Boot là một framework mạnh mẽ, đơn giản hóa quá trình phát triển các ứng dụng Java bằng cách tự động hóa nhiều cấu hình trong Spring Framework. Đặc biệt với việc xây dựng các dịch vụ web RESTful, Spring Boot cung cấp các công cụ và tính năng hỗ trợ lập trình viên dễ dàng tạo các API nhanh chóng và linh hoạt mà không cần phải cấu hình phức tạp.

Trong Spring Boot, các API RESTful thường được phát triển theo mô hình kiến trúc REST (Representational State Transfer), giúp giao tiếp dễ dàng với các hệ thống khác nhau thông qua giao thức HTTP, sử dụng các phương thức như GET, POST, PUT, và DELETE.

**Các Thành Phần Chính trong RESTful API với Spring Boot:**

**Controller:**

Controller là thành phần chính trong việc xử lý các yêu cầu từ người dùng (client). Trong Spring Boot, các lớp Controller thường được đánh dấu bằng annotation @RestController, giúp phân biệt các API RESTful.

Mỗi phương thức trong Controller tương ứng với một hành động HTTP cụ thể, được định nghĩa bằng các annotation như @GetMapping, @PostMapping, @PutMapping, và @DeleteMapping, cho phép xác định các endpoint dễ dàng và nhanh chóng.

**Service:**

Service là lớp trung gian giữa Controller và tầng dữ liệu (repository), chịu trách nhiệm xử lý logic nghiệp vụ của ứng dụng.

Bằng cách tạo lớp Service, Spring Boot hỗ trợ tính năng Dependency Injection giúp Service dễ dàng tích hợp và quản lý.

Service cũng hỗ trợ kiểm thử đơn vị và kiểm thử tích hợp, giúp đảm bảo tính chính xác và ổn định của ứng dụng.

**Repository:**

Repository là lớp giúp quản lý và tương tác với cơ sở dữ liệu, thường được đánh dấu bằng annotation @Repository hoặc @JpaRepository (khi sử dụng JPA).

Repository cung cấp các phương thức xử lý CRUD (Create, Read, Update, Delete) với cơ sở dữ liệu và có thể mở rộng để tùy chỉnh truy vấn theo nhu cầu cụ thể của ứng dụng.

**Ưu Điểm của Spring Boot cho RESTful API:**

Cấu hình đơn giản: Với Spring Boot, các thiết lập cấu hình phức tạp được tự động hóa nhờ tính năng Spring Boot Auto-Configuration và các starter như spring-boot-starter-web giúp tiết kiệm thời gian khi thiết lập dự án RESTful API.

Hỗ trợ mạnh mẽ cho RESTful: Spring Boot cung cấp các annotation và công cụ tối ưu cho RESTful API như @RestController, @RequestMapping, @GetMapping, @PostMapping,… giúp dễ dàng phát triển các dịch vụ RESTful.

Tích hợp dễ dàng với các công nghệ khác: Spring Boot dễ dàng tích hợp với các thành phần khác của Spring như Spring Security để bảo mật API, Spring Data JPA để quản lý dữ liệu, và Spring Cloud để xây dựng các hệ thống phân tán.

Đơn giản hóa quản lý lỗi: Cung cấp khả năng xử lý và quản lý lỗi thông qua @ExceptionHandler và @ControllerAdvice, giúp định nghĩa rõ ràng các phản hồi lỗi đến client.

**Nhược Điểm của Spring Boot:**

Hiệu suất: So với một số framework nhẹ hơn, Spring Boot có thể chậm hơn vì tính năng và tính linh hoạt cao.

Phụ thuộc vào hệ sinh thái Spring: Sử dụng Spring Boot đồng nghĩa với việc gắn bó sâu sắc với hệ sinh thái Spring, điều này có thể gây khó khăn khi cần chuyển đổi sang các công nghệ khác

### MySQL

MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở phổ biến được phát triển và duy trì bởi Oracle Corporation. Nó được thiết kế để quản lý các cơ sở dữ liệu quan hệ, giúp lưu trữ và truy vấn dữ liệu hiệu quả. MySQL được sử dụng rộng rãi trong các ứng dụng web, dịch vụ trực tuyến và nhiều ứng dụng doanh nghiệp nhờ vào khả năng mạnh mẽ và tính linh hoạt của nó.

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS), trong đó dữ liệu được tổ chức thành các bảng (tables). Mỗi bảng gồm các hàng (rows) và các cột (columns), và mỗi dữ liệu được thêm vào bảng phải tuân theo cấu trúc đã quy định của bảng đó. MySQL sử dụng ngôn ngữ truy vấn có cấu trúc (SQL) để thao tác với dữ liệu, bao gồm việc thêm, sửa, xóa và truy vấn dữ liệu.

Ưu điểm

* Tránh rùng lặp dữ liệu, dữ liệu không đồng bộ, không an toàn, lọc và tìm kiếm thông tin không dễ dàng
* Dễ dàng truy cập database và lấy thông tin khi cần.
* Thông tin được cập nhật trên cơ sở dữ liệu thì người dùng lấy dữ liệu, dữ liệu sẽ được đảm bảo đồng bộ hoàn toàn.
* Thông tin được quản lý ở một nơi duy nhất, đảm bảo việc bảo mật dữ liệu, tránh rò rỉ thông tin. Ngoài ra có thể phân quyền cho người dùng, ai có thể truy cập dữ liệu
* Việc tìm kiếm thông tin cũng vô cùng dễ dàng khi dữ liệu đã được quản lsy bởi DBMS, chúng có thể tìm kiếm, gom nhóm các thông tin lại với nhau bằng cách dùng câu truy vấn.

Nhược điểm

* Việc sử dụng cơ sở dữ liệu cũng có một số nhược điểm sau:
* Việc sử dụng chung cơ sở dữ liệu khi xảy ra sự cố, toàn bộ dữ liệu sẽ bị mất hoàn toàn.
* Việc dùng chung cơ sở dữ liệu khi có quá nhiều truy vấn sẽ gây nghẽn, sập hệ thống.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Xác định các tác nhân, ca sử dụng

### Tác nhân

KhachHang: Khách hàng

+ Khách hàng có thể đăng nhập,tìm kiếm các phòng trống, xem chi tiết phòng,đặt phòng , thêm thông tin cá nhân, đánh gia sau khi đã sử dụng phòng v.v.

NguoiQuanTri: Người quản trị

+ Xem thống kê

+ Người quản trị có khả năng thêm sửa xóa các thông tin như loại phòng, phòng, đặt phòng, thanh toán, mã giảm giá, đánh giá, liên hệ, người dùng v.v.

NhanVien: Nhân viên

+ Nhân viên có chức năng xem hiệu xuất của cá nhân, quản lý đặt phòng, quản lý thanh toán, quản lý đánh giá, quản lý liên hệ

NguoiQuanTri: Người quản lý

+ Xem thống kê

+ Người quản lý có khả năng thêm sửa xóa các thông tin như loại phòng, phòng, đặt phòng, thanh toán, đánh giá, liên hệ, người dùng v.v.

### Use Case

* Dang Nhap: Đăng nhập
* Dang Ky: Đăng ký
* Quen Mat Khau: Quên Mật Khẩu
* Doi Mat Khau: Đổi mật khẩu
* Tim Phong Trong: Tìm phòng trống
* Thong Tin Dat Phong: Thông tin đặt phòng
* Chi Tiet Phong: Chi tiết phòng
* Thong Tin Ca Nhan: Thông tin cá nhân
* Dat Phong: Đặt phòng
* Danh Gia: Đánh giá
* Thong Ke: Thống kê
* Quan Ly Danh Muc: Quản lý danh mục
* Quan Ly Phong: Quản lý phòng
* Quan Ly Dat Phong: Quản lý đặt phòng
* Quan Ly Thanh Toan: Quản lý thanh toán
* Quan Ly Ma Giam Gia: Quản lý mã giảm giá
* Quan Ly Danh Gia: Quản lý đánh giá
* Quan Ly Lien he: Quản lý liên hệ
* Quan Ly Nguoi Dung: Quản lý người dùng

## Các biểu đồ

### Biểu đồ use case chính



Hình . Biểu đồ use casse chính

### Biều đồ use case phân rã



Hình .2 Biểu đồ use casse của khách hàng



Hình .3 Biểu đồ use case của nhân viên



Hình .4 Biểu đồ use case của người quản trị và người quản lý

## Mô tả chi tiết các use case

### Mô tả use case Dang Nhap

* Tên Use Case: Đăng nhập
* Mô tả vắn tắt: Use case này cho phép người dùng đăng nhập vào hệ thống.
* Luồng cơ bản:

1. Người dùng nhập thông tin tài khoản (tên người dùng và mật khẩu) vào form đăng nhập và nhấn nút "Đăng nhập".
2. Hệ thống xác thực thông tin đăng nhập với cơ sở dữ liệu.
3. Nếu thông tin hợp lệ, hệ thống chuyển hướng người dùng đến trang chủ

* Luồng rẽ nhánh:

1. Nếu thông tin đăng nhập không hợp lệ, hệ thống hiển thị thông báo "Thông tin đăng nhập không chính xác" và yêu cầu người dùng nhập lại.
2. Nếu không kết nối được với cơ sở dữ liệu, hệ thống hiển thị thông báo lỗi "Lỗi kết nối" và kết thúc use case.

* Các yêu cầu đặc biệt: Không có.
* Tiền điều kiện: Người dùng đã đăng ký tài khoản
* Hậu điều kiện: Không có.
* Điểm mở rộng: Không có.

### Mô tả use case Dang Ky

* Tên Use Case: Đăng ký
* Mô tả vắn tắt: Use case này cho phép người dùng tạo tài khoản trong hệ thống.
* Luồng các sự kiện
* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng truy cập trang đăng ký của hệ thống.
2. Người dùng nhập thông tin cá nhân vào các trường như UserName (Tên đăng nhập), Email, Password (Mật khẩu).
3. Người dùng kích vào nút "Đăng ký" để gửi thông tin đăng ký lên hệ thống.
4. Hệ thống kiểm tra tính hợp lệ của thông tin đăng ký và xác minh tính duy nhất của địa chỉ email.
5. Nếu thông tin đăng ký hợp lệ và địa chỉ email chưa tồn tại trong hệ thống, hệ thống tạo một tài khoản mới cho người dùng và lưu thông tin tài khoản vào bảng USERS trong cơ sở dữ liệu.

* Luồng rẽ nhánh:

1. Nếu người dùng nhập thông tin đăng ký không hợp lệ, hệ thống sẽ hiển thị thông báo lỗi “Nhập thông tin lỗi” yêu cầu nhập lại thông tin.
2. Nếu địa chỉ email mà người dùng đã nhập đã tồn tại trong hệ thống, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng sử dụng một địa chỉ email khác hoặc đăng nhập nếu họ đã có tài khoản trước đó.

* Các yêu cầu đặc biệt

Use case này có thể được thực hiện bởi tất cả người dùng chưa có tài khoản trong hệ thống.

* Tiền điều kiện: Không có.
* Hậu điều kiện

Nếu thông tin đăng ký hợp lệ và địa chỉ email chưa tồn tại, người dùng sẽ có một tài khoản mới và có thể đăng nhập vào hệ thống. Nếu thông tin đăng ký không hợp lệ hoặc địa chỉ email đã tồn tại, hệ thống sẽ hiển thị thông báo lỗi và người dùng cần sửa lại thông tin đăng ký để tiếp tục.

Điểm mở rộng: Không có

### Mô tả use case Thong Tin Ca Nhan

* Tên Use Case: Xem thông tin cá nhân.
* Mô tả vắn tắt: Use case này cho phép người dùng xem thông tin cá nhân của mình.
* Luồng các sự kiện
* Luồng cơ bản:

1. Người dùng đăng nhập và chọn tùy chọn "Thông tin cá nhân".
2. Hệ thống hiển thị thông tin cá nhân của người dùng (họ tên, email, số điện thoại, địa chỉ, v.v.).

* Luồng rẽ nhánh:

1. Nếu không kết nối được với cơ sở dữ liệu, hệ thống hiển thị thông báo "Lỗi kết nối".

* Các yêu cầu đặc biệt: Không có
* Tiền điều kiện

Người dùng cần đăng nhập trước khi có thể thực hiện use case.

* Hậu điều kiện: Không có.
* Điểm mở rộng: Không có.

### Mô tả use case Xem Chi Tiet Phong

* Tên Use Case: Xem chi tiết phòng
* Mô tả vắn tắt: Use case này cho phép khách hàng xem thông tin chi tiết về phòng.
  + Luồng cơ bản:
    - Use case này bắt đầu khi khách hàng nhấp vào “Chi tiết phòng” từ danh sách phòng hoặc kết quả tìm kiếm. Hệ thống sẽ lấy thông tin phòng được chọn từ bảng ROOMS và hiển thị ra màn hình thông tin chi tiết của phòng.
  + Luồng rẽ nhánh:
    - Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu, hệ thống sẽ hiển thị một thông báo lỗi "Lỗi kết nối" và use case kết thúc.
* Các yêu cầu đặc biệt: Không có.
* Tiền điều kiện: Không có.
* Hậu điều kiện: Không có.
* Điểm mở rộng: Đặt phòng

### Mô tả use case Danh Gia Phong

* Tên Use Case: Đánh giá phòng
* Mô tả vắn tắt: Use case này cho phép người dùng đã sử dụng phòng để đánh giá chất lượng phòng.
* Luồng các sự kiện
* Luồng cơ bản:

1. Người dùng đăng nhập và chọn tùy chọn “Thông tin đặt phòng” và chọn tùy chọn đánh giá.
2. Người dùng nhập nội dung đánh giá và sao đánh giá.

* Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu, hệ thống sẽ hiển thị một thông báo lỗi "Lỗi kết nối" và use case kết thúc

* Các yêu cầu đặc biệt: Không có.
* Tiền điều kiện: Người dùng đã sử dụng phòng và muốn đánh giá
* Hậu điều kiện: Đánh giá được lưu vào cơ sở dữ liệu.
* Điểm mở rộng: Không có.

### Mô tả use case Dat Phong

* Tên Use Case: Đặt phòng.
* Mô tả vắn tắt: Use case này cho phép dùng đặt phòng trên website.
* Luồng các sự kiện
* Luồng cơ bản:

1. Người dùng chọn phòng từ danh sách phòng hoặc có thể tìm kiếm phòng trống.
2. Người dùng click vào “Xem chi tiết”.
3. Người dùng nhập thông tin (Ngày đến, ngày đi, số người lớn và trẻ em) hệ thống kiểm tra nếu còn phòng và valiade dữ liệu thành công thì chuyển sang trang chi tiết đặt phòng
4. Người dùng nhập thêm thông tin (Tên, SĐT, Email…) và nhấn đặt phòng
5. Hệ thống chuyển sang trang thanh toàn với VNPay
6. Khi thanh toán thành công sẽ hiển thị ra thông tin chi tiết đặt phòng

* Luồng rẽ nhánh:

1. Nếu phòng không còn trống, hệ thống sẽ hiển thị thông báo "Phòng không còn trống".
2. Nếu không kết nối được với cơ sở dữ liệu, hệ thống hiển thị thông báo "Lỗi kết nối" và kết thúc use case

* Các yêu cầu đặc biệt: Không có
* Tiền điều kiện

Người dùng cần đăng nhập trước khi có thể thực hiện use case.

* Hậu điều kiện:

Nếu người dùng đặt phòng thành công thì dữ liệu sẽ nhập xuống CSDL.

* Điểm mở rộng: Không có.

### Mô tả use case Quan Ly Loai Phong

* Tên Use Case: Quản lý loại phòng.
* Mô tả vắn tắt: Người quản trị và người quản lý có thể thêm, sửa, xóa các danh mục liên quan đến loại phòng.
* Luồng các sự kiện
* Luồng cơ bản:

1. Use case bắt đầu khi người quản trị kích vào nút "Danh mục" trên menu quản trị. Hệ thống lấy tất cả thông tin về loại phòng từ bảng CATEGORY trong cơ sở dữ liệu và hiển thị danh sách loại phòng lên màn hình.
2. Thêm thông tin loại phòng:
3. Người quản trị kích vào nút "Thêm mới" trên cửa sổ danh mục. Hệ thống hiển thị màn hình yêu cầu nhập thông tin về loại phòng.
4. Người quản trị nhập các thông tin và kích vào nút "Tạo". Hệ thống sẽ tạo một ID mới, cập nhật bảng CATEGORY và hiển thị danh sách loại phòng đã được cập nhật.
5. Sửa thông tin loại phòng:
6. Người quản trị kích vào nút "Sửa" trên một hàng loại phòng. Hệ thống lấy tất cả thông tin cũ của loại phòng được chọn.
7. Người quản trị nhập đầy đủ thông tin mới cho loại phòng đó sau đó kích vào nút "Cập nhật". Hệ thống sẽ cập nhật thông tin của loại phòng được chọn trong bảng Category và hiển thị danh sách loại phòng đã được cập nhật.
8. Xóa thông tin loại phòng:
9. Người quản trị kích vào nút "Xóa" trên một dòng loại phòng. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa.
10. Người quản trị kích vào nút "Đồng ý". Hệ thống sẽ xóa loại phòng được chọn khỏi bảng Category và hiển thị danh sách loại phòng đã được cập nhật.

* Luồng rẽ nhánh:

1. Tại bước 4b trong luồng cơ bản, nếu người quản trị kích vào nút "Hủy", hệ thống sẽ bỏ qua thao tác xóa và hiển thị danh sách loại hàng trong bảng CATEGORY.
2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu, hệ thống sẽ hiển thị một thông báo lỗi "Lỗi kết nối" và use case kết thúc.

* Các yêu cầu đặc biệt

Use case này chỉ cho phép người quản trị và người quản lý thực hiện.

* Tiền điều kiện

Người quản trị cần đăng nhập với vai trò quản trị và người quản lý hệ thống trước khi có thể thực hiện use case.

* Hậu điều kiện

Nếu use case kết thúc thành công, thông tin về loại phòng sẽ được câp nhật vào CSDL.

* Điểm mở rộng: Không có.

### Mô tả use case Quan Ly Phong

* Tên Use Case: Quản lý phòng.
* Mô tả vắn tắt: Use case này cho phép người quản trị và người quản lý có thể thêm, sửa, xóa thông tin phòng trong hệ thống.
* Luồng các sự kiện
* Luồng cơ bản:

1. Use case bắt đầu khi người quản trị kích vào nút "Phòng" trên menu quản trị. Hệ thống lấy tất cả thông tin về các phòng từ bảng ROOMS trong cơ sở dữ liệu và hiển thị danh sách các phòng lên màn hình.
2. Thêm thông tin phòng:
3. Người quản trị kích vào nút "Thêm mới" trên cửa sổ phòng. Hệ thống hiển thị màn hình yêu cầu nhập thông tin cho phòng, bao gồm tên phòng, số phòng, loại phòng…
4. Người quản trị nhập các thông tin và kích vào nút "Tạo". Hệ thống sẽ tạo một ID mới, cập nhật bảng ROOMS và hiển thị danh sách phòng đã được cập nhật.
5. Sửa thông tin phòng:
6. Người quản trị kích vào nút "Sửa" trên một dòng phòng Hệ thống lấy tất cả thông tin cũ của phòng được chọn.
7. Người quản trị nhập đầy đủ thông tin mới cho phòng đó sau đó kích vào nút "Cập nhật". Hệ thống sẽ cập nhật thông tin của khách hàng được chọn trong bảng ROOMS và hiển thị danh sách phòng đã được cập nhật.
8. Xóa thông tin phòng:
9. Người quản trị kích vào nút "Xóa" trên một dòng phòng. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa.
10. Người quản trị kích vào nút "Đồng ý". Hệ thống sẽ xóa phòng được chọn khỏi bảng ROOMS và hiển thị danh sách các phòng đã được cập nhật.

* Luồng rẽ nhánh:

1. Tại bước 4b trong luồng cơ bản, nếu người quản trị kích vào nút "Hủy", hệ thống sẽ bỏ qua thao tác xóa và hiển thị danh sách các phòng trong bảng ROOMS
2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu, hệ thống sẽ hiển thị một thông báo lỗi "Lỗi kết nối" và use case kết thúc.

* Các yêu cầu đặc biệt

Use case này chỉ cho phép người quản trị và người quản lý thực hiện.

* Tiền điều kiện

Người quản trị và người quản lý cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case.

* Hậu điều kiện

Nếu use case kết thúc thành công, thông tin về phòng sẽ được cập nhật trong cơ sở dữ liệu.

* Điểm mở rộng: Không có.

### Mô tả use case Quan Ly Dat Phong

* Tên Use Case: Quản lý đặt phòng.
* Mô tả vắn tắt: Use case này cho phép người quản trị, quản lý, nhân viên có thể xem, xác nhận, và hủy các đơn đặt phòng.
* Luồng các sự kiện
* Luồng cơ bản:

1. Admin chọn tùy chọn "Đặt Phòng" trên menu hệ thống.
2. Admin xem danh sách đơn đặt phòng và thực hiện các thao tác xác nhận hoàn thành hoặc hủy.

* Luồng rẽ nhánh:

1. Nếu có lỗi trong quá trình xác nhận/hủy/xóa, hệ thống hiển thị thông báo lỗi.
2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu, hệ thống sẽ hiển thị một thông báo lỗi "Lỗi kết nối" và use case kết thúc.

* Các yêu cầu đặc biệt

Use case này chỉ cho phép người quản trị, nhân viên, quản lý thực hiện.

* Tiền điều kiện

Người quản trị,nhân viên, quản lý cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case.

* Hậu điều kiện: Nếu use case kết thúc thành công, thông tin về đặt phòng sẽ được cập nhật trong cơ sở dữ liệu.
* Điểm mở rộng: Không có.

### Mô tả use case Quan Ly Thanh Toan

* Tên Use Case: Quản lý thanh toán
* Mô tả vắn tắt: Use case này cho phép người quản trị có thể xem, các phương thức thanh toán của người dùng.
* Luồng các sự kiện
* Luồng cơ bản:

1. Admin chọn tùy chọn "Thanh Toán" trên menu hệ thống.
2. Hệ thống sẽ lấy ra thông tin thanh toán trong bảng PAYMENTS trong cơ sở dữ liệu.

* Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu, hệ thống sẽ hiển thị thông báo lỗi "Lỗi kết nối" và use case kết thúc.

* Các yêu cầu đặc biệt:

Use case này chỉ cho phép người quản trị, quản lý, nhân viên thực hiện.

* Tiền điều kiện: Người quản trị,quản lý, nhân viên cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case..
* Hậu điều kiện: Không có
* Điểm mở rộng: Không có.

### Mô tả use case Quan Ly Nguoi Dung

* Tên Use Case: Quản lý người dùng
* Mô tả vắn tắt: Use case này cho phép người quản trị,quản lý có thể thêm tài khoản người dùng
* Luồng cơ bản:

1. Admin chọn tùy chọn "Quản lý người dùng" trên menu hệ thống.
2. Người dùng thực hiện thêm, cập nhật quyền, hoặc vô hiệu hóa tài khoản người dùng khi cần thiết.

* Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu, hệ thống sẽ hiển thị một thông báo lỗi "Lỗi kết nối" và use case kết thúc.

* Các yêu cầu đặc biệt

Use case này chỉ cho phép người quản trị, quản lý thực hiện.

* Tiền điều kiện

Người quản trị cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case.

* Hậu điều kiện: Nếu use case kết thúc thành công, thông tin về người dùng sẽ được cập nhật trong cơ sở dữ liệu.
* Điểm mở rộng: Không có.

### Mô tả use case Thong Ke

* Tên Use Case: Xem thống kê.
* Mô tả vắn tắt: Admin có thể xem các báo cáo thống kê về hoạt động của hệ thống.
* Luồng các sự kiện
* Luồng cơ bản:

1. Use case bắt đầu khi người quản trị kích vào nút "Thống kê" trên menu quản trị. Hệ thống hiển thị thống kê về số lượt đặt phòng, doanh thu, và các báo cáo khác.

* Luồng rẽ nhánh:

1. Nếu không có dữ liệu, hệ thống hiển thị thông báo "Không có dữ liệu.

* Các yêu cầu đặc biệt

Use case này chỉ cho phép người quản trị, quản lý thực hiện.

* Tiền điều kiện

Người quản trị, quản lý cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case.

* Hậu điều kiện: Không có
* Điểm mở rộng: Không có.

## Biểu đồ trình tự

### Use case Danh Nhap

#### Luồng cơ bản



#### Luồng rẽ nhánh



Hình .5 PTTK Use case Đăng nhập

### Use case Dang Ky

#### Luồng cơ bản



#### Luồng rẽ nhánh



Hình .6 PTTK Use case Đăng ký

### Use case Thong Tin Ca Nhan

#### Luồng cơ bản



#### Luồng rẽ nhánh



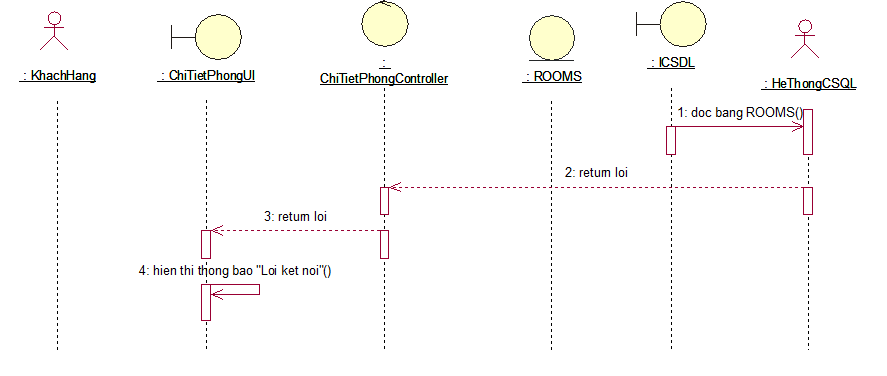
Hình .7 PTTK Use case Thông tin cá nhân

### Use case Xem Chi Tiet Phong

#### Luồng cơ bản



#### Luồng rẽ nhánh



Hình .8 PTTK Use case Xem chi tiết phòng

### Use case Dat Phong

#### Luồng cơ bản



1. Luồng rẽ nhánh



Hình .9 PTTK Use case Đặt phòng

### Use case Quan Ly Phong

#### Luồng cơ bản



#### Luồng rẽ nhánh



Hình .10 PTTK Use case Quản lý phòng

### Use case Quan Ly Dat Phong

#### Luồng cơ bản



#### Luồng rẽ nhánh



Hình .11 PTTK Use case Quản lý đặt phòng

### Use case Quan Ly Thanh Toan

#### Luồng cơ bản



#### Luồng rẽ nhánh



Hình .12 PTTK Use case Quản lý thanh toán

### Use case Quan Ly Nguoi Dung (Khach Hang)

#### Luồng cơ bản



#### Luồng rẽ nhánh



Hình .13 PTTK Use case Quản lý người dùng

### Use case Thong Ke

#### Luồng cơ bản



#### Luồng rẽ nhánh



Hình .14 PTTK Use case Thống kê

## Biểu đồ lớp chi tiết

### Use case Use case Dang Nhap



Hình .15 Biểu đồ lớp Đăng nhập

### Use case Dang Ky



Hình .16 Biểu đồ lớp Đăng ký

### Usecase Thong Tin Ca Nhan



Hình .17 Biểu đồ lớp Thông tin cá nhân

### Use case Xem Chi Tiet Phong



Hình .18 Biểu đồ lớp Xem chi tiết phòng

### Use case Dat Phong



Hình .19 Biểu đồ lớp Đặt phòng

### Use case Quan Ly Phong



Hình .20 Biểu đồ lớp Quản lý phòng

### Use case Quan Ly Dat Phong



Hình .21 Biểu đồ lớp Quản đặt phòng

### Use case Quan Ly Thanh Toan



Hình .22 Biểu đồ lớp Quản lý thanh toán

### Use case Quan Ly Nguoi Dung (Khach Hang)



Hình .23 Biểu đồ lớp Quản lý người dùng

### Use case Thong Ke



Hình . Biểu đồ lớp Thống kê

## Thiết kế Cơ sở dữ liệu

### Mô hình hóa dữ liệu

Để quản lý tốt thông tin của khách hàng, đặc biệt là khách hàng có nhu cầu đặt phòng trực tuyến, hệ thống yêu cầu khách hàng cung cấp một số thông tin cá nhân cơ bản để dễ dàng liên hệ và quản lý. Ví dụ, trường “Full Name” giúp nhận diện khách hàng trên hệ thống, đồng thời phục vụ cho việc tạo tài khoản. Trường “Address” được sử dụng để liên hệ khách hàng trong trường hợp có vấn đề phát sinh.

Khi một khách hàng có nhu cầu đặt phòng thường xuyên hoặc muốn nhận ưu đãi, họ có thể đăng ký tài khoản. Thông tin tài khoản sẽ được lưu trữ trong bảng “users”, trong đó trường “id” sẽ dùng để ánh xạ tới hồ sơ khách hàng sở hữu tài khoản. Ngoài ra, bảng còn bao gồm các trường như “phone\_number” và “email” để xác minh danh tính và liên hệ.

Một trong những đối tượng quan trọng nhất của trang web là các phòng khách sạn. Để lưu trữ thông tin phòng trong cơ sở dữ liệu, chúng ta sử dụng các bảng như “rooms” và “room\_categories”. Bảng “rooms” lưu các thông tin chi tiết về phòng như tên, giá, mô tả, trong khi bảng “room\_categories” lưu các thông tin về loại phòng như tên loại phòng và mô tả.

**Các bảng và trường trong cơ sở dữ liệu**

**Bảng Hotels**

**hotels**: Lưu trữ thông tin khách sạn, gồm các trường id, address, email, name, và phone\_number.

**Bảng Rooms**

**rooms**: Lưu trữ thông tin chi tiết từng phòng, gồm các trường id, hotel\_id, capacity, description, discounted\_price, discount, price, room\_img, room\_img\_public\_id, room\_number, size, view, và category\_id.

**Bảng Room Categories**

**room\_categories**: Lưu thông tin về loại phòng, gồm các trường id, description, và name.

**Bảng Users**

**users**: Lưu trữ thông tin khách hàng, gồm các trường id, create\_at, update\_at, address, avatar, avatar\_public\_id, email, enable, gender, name, password, phone\_number, username, và role\_id.

**Bảng Roles**

**roles**: Định nghĩa các vai trò người dùng, gồm các trường id và name.

**Bảng Bookings**

**bookings**: Quản lý các đặt phòng với các trường id, create\_at, update\_at, booking\_email, booking\_phone, check\_in\_date, check\_out\_date, guest, status, total\_amount, room\_id, user\_id, booking\_status, và is\_rated.

**Bảng Payments**

**payments**: Lưu trữ thông tin thanh toán, gồm các trường id, amount, payment\_date, payment\_method, và booking\_id.

**Bảng Coupon**

**coupon**: Lưu trữ thông tin mã giảm giá, gồm các trường id, code, discount\_percentage, và expiry\_date.

**Bảng Reviews**

**reviews**: Quản lý các đánh giá của người dùng, gồm các trường id, avatar, comment, create\_at, rating, booking\_id, và user\_id.

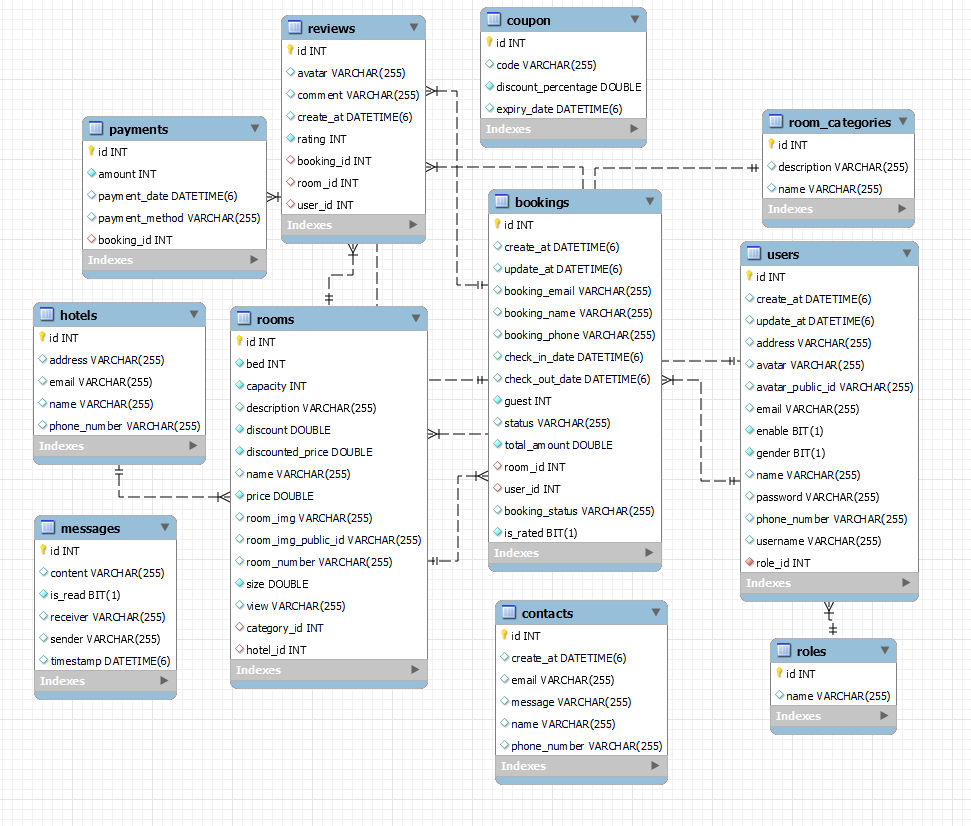
**Bảng Messages**

**messages**: Quản lý tin nhắn giữa các người dùng, gồm các trường id, content, is\_read, receiver, sender, và timestamp.

**Bảng Contacts**

**contacts**: Lưu trữ các yêu cầu liên hệ từ người dùng, gồm các trường id, create\_at, email, message, name, và phone\_number

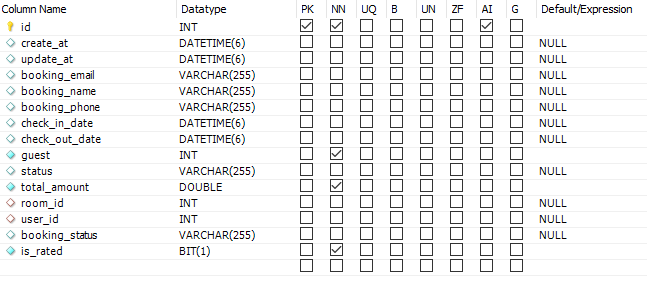
### Sơ đồ dữ liệu quan hệ



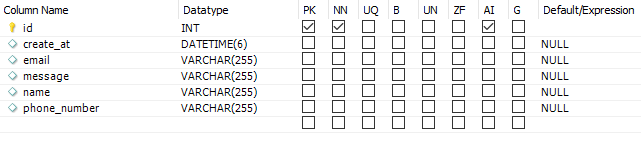
Hình .25 Sơ đồ dữ liệu quan hệ

### Chi tiết các bảng dữ liệu

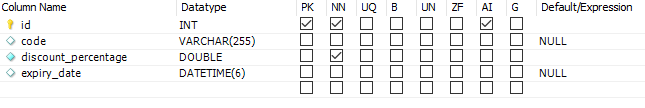
*Bảng booking*

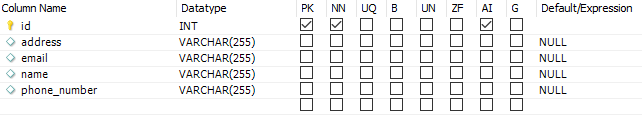


*Bảng contact*

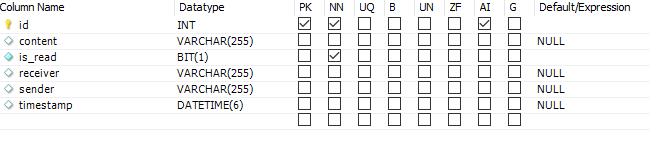


*Bảng Coupon*

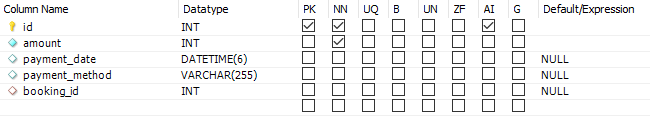
****

*Bảng Hotel*

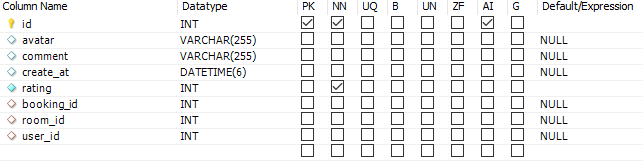
*Bảng Messages*



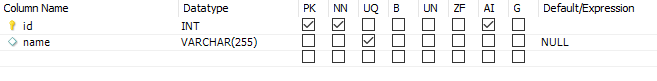
*Bảng Payments*

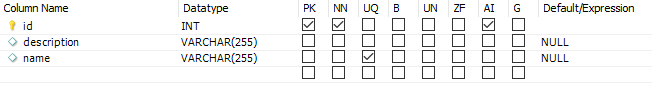


*Bảng Review*

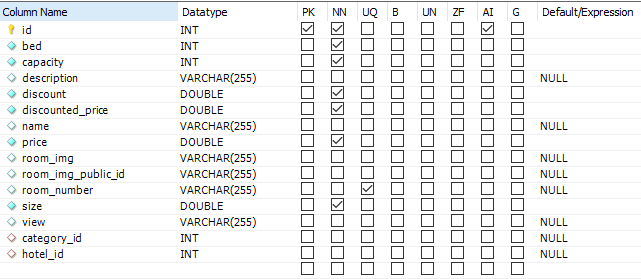
**

*Bảng Roles*

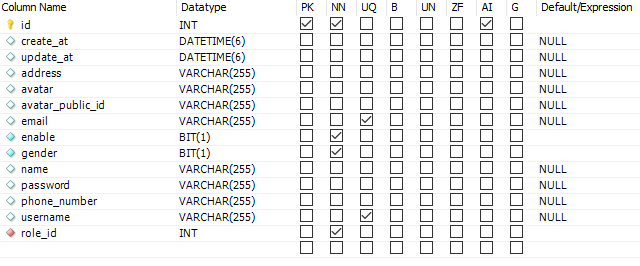


*Bảng Room Category*

*Bảng Room*



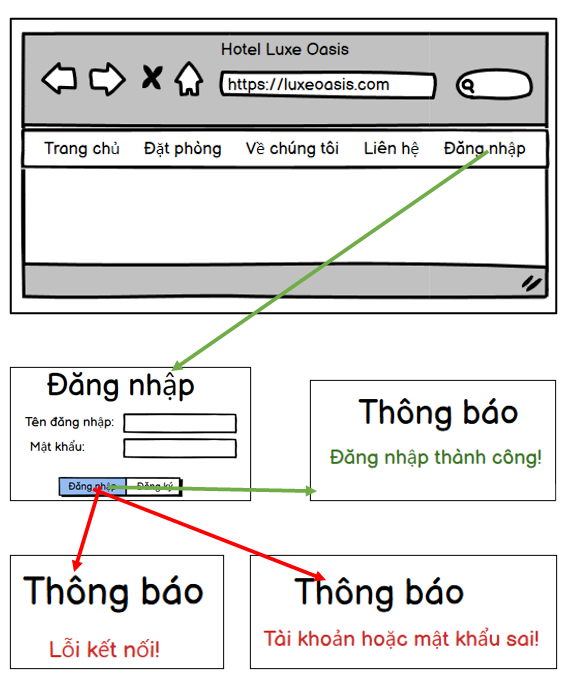
*Bảng User*



Hình .26 Các bảng trong CSDL

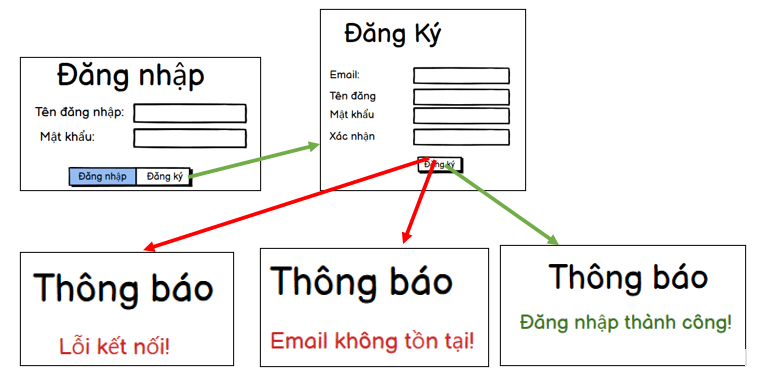
## Thiết kế giao diện hệ thống

### Use case Dang Nhap



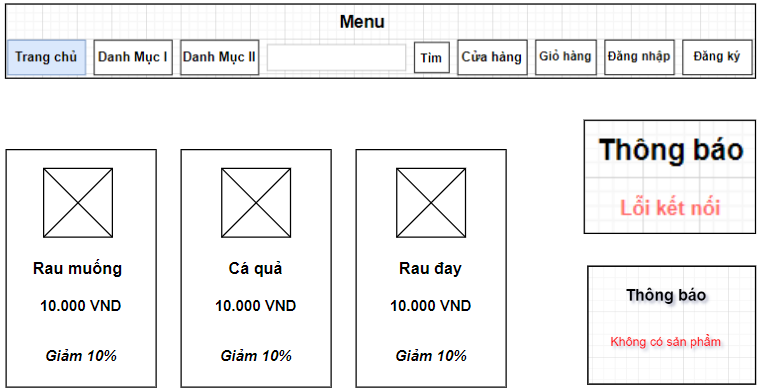
Hình .27 Hình hình dung màn hình đăng nhập

### Use case Dang Ky



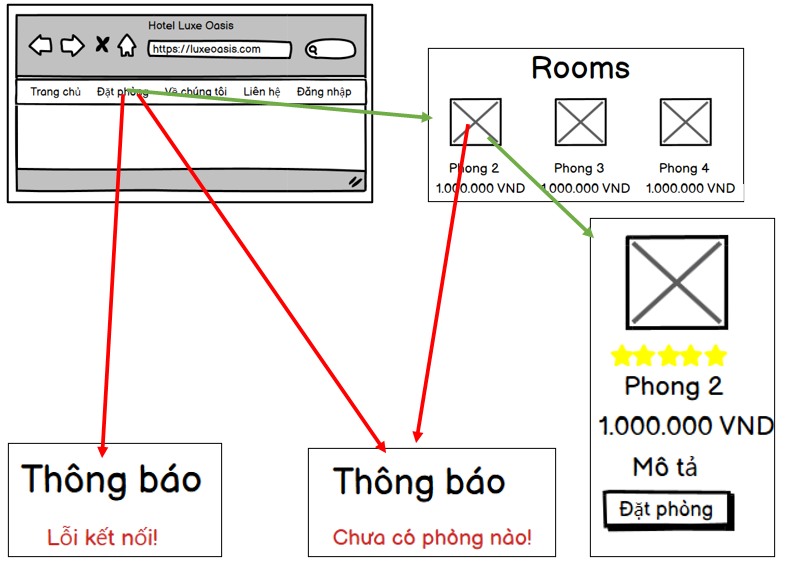
Hình .28 Hình hình dung màn hình đăng ký

### Use case Thong Tin Ca Nhan



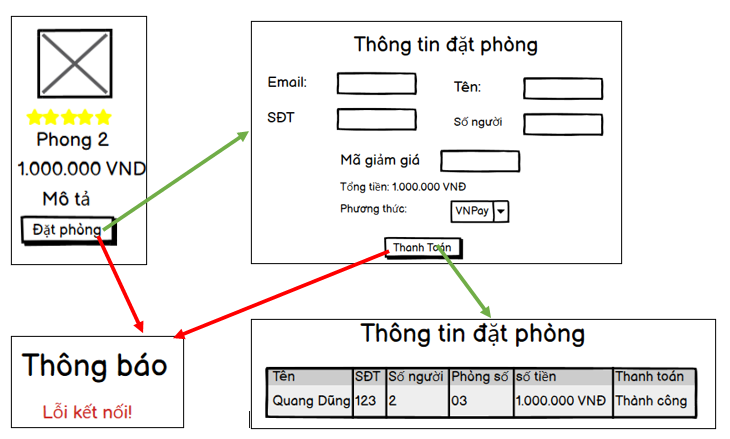
Hình .29 Hình hình dung màn hình thông tin cá nhân

### Use case Xem Chi Tiet Phong



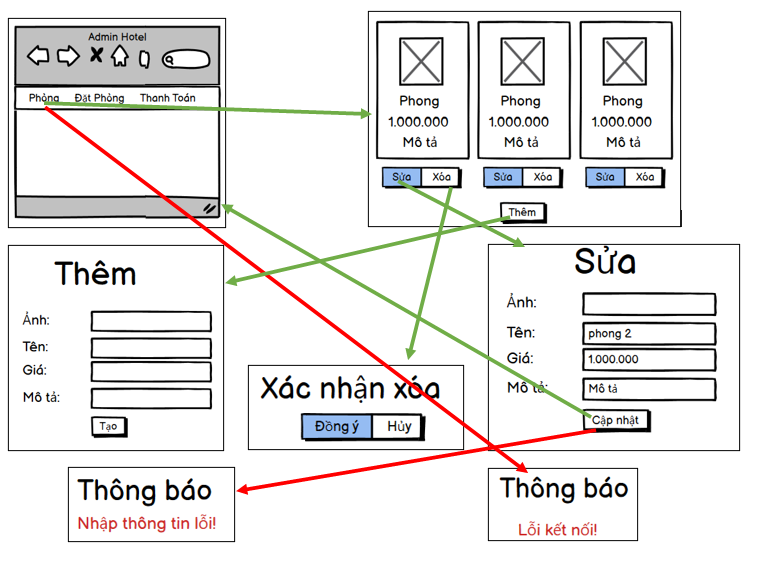
Hình .30 Hình hình dung màn hình xem chi tiết phòng

### Use case Dat Phong



Hình .31 Hình hình dung màn hình đặt phòng

### Use case Quan Ly Phong



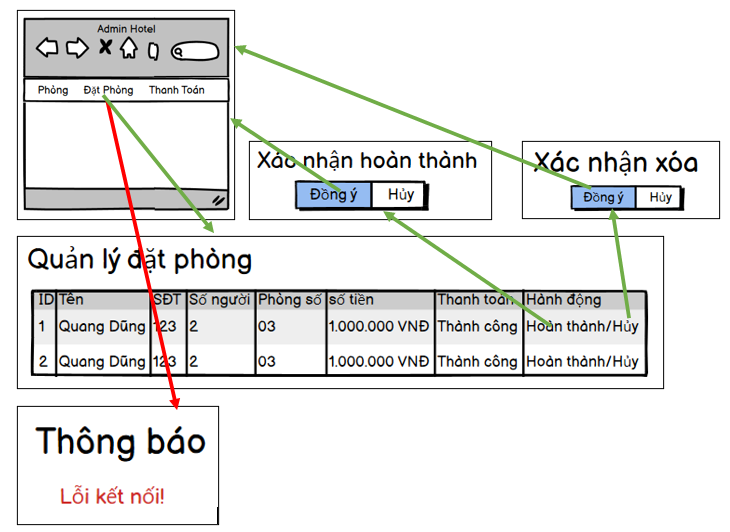
Hình .32 Hình hình dung màn hình quản lý phòng

### Use case Quan Ly Dat Phong



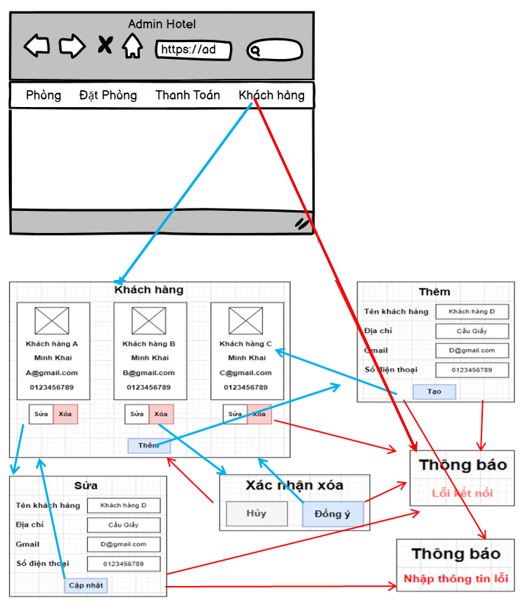
Hình .33 Hình hình dung màn hình quản lý đặt phòng

### Use case Quan Ly Thanh Toan



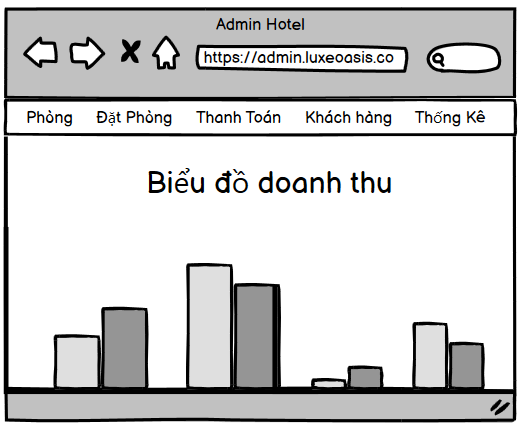
Hình .34 Hình hình dung màn hình quản lý thanh toán

### Use case Qaun Ly Nguoi Dung (Khach Hang)



Hình .35 Hình hình dung màn hình quản lý người dùng

### Use case Quan Ly Thong Ke



Hình .36 Hình hình dung màn hình thống kê