|  |  |
| --- | --- |
| **logo moi** | BỘ GIÁO DỤC VÀ ĐÀO TẠO  **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM** |

**ĐỒ ÁN TỔNG HỢP**

**WEBSITE QUẢN LÝ KHÁCH SẠN**

Ngành: **CÔNG NGHỆ THÔNG TIN**

Chuyên Ngành: **CÔNG NGHỆ PHẦN MỀM**

Giáo viên hướng dẫn:TRƯƠNG THỊ MINH CHÂU

Lớp: 16DTHB3

Sinh viên thực hiện:

1. Nguyễn Đức Hiếu MSSV: 1611060735
2. Dương Nguyễn Hoàng Thanh MSSV: 1611060646

TP. Hồ Chí Minh, 2020

**LỜI CẢM ƠN**

Sau thời gian thực hiện, đề tài nghiên cứu “Xây dựng Website quản lý khách sạn” đã hoàn thành. Ngoài sự cố gắng của chúng em còn nhận được sự giúp đỡ nhiệt tình của thầy cô, bạn bè, các anh, chị.

Chúng em xin cám ơn cô Trương Thị Minh Châu đã truyền đạt cho chúng em những kiến thức, hiểu biết quý báu cùng niềm đam mê để chúng em có thể hoàn thành tốt đề tài này.

Do chưa có nhiều kinh nghiệm nghiên cứu, thực hành nên đề tài thực hiện còn nhiều thiếu sót, chúng em mong nhận được đóng góp ý kiến của thầy để đề tài được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

Sinh viên thực hiện:

Nguyễn Đức Hiếu

Dương Nguyễn Hoàng Thanh

**LỜI MỞ ĐẦU**

Nghành du lịch là một nghành kinh doanh có từ rất lâu đời trên thế giới. Tại các nước đang phát triển, nghành du lịch phát triển hết sức mạnh mẽ và có lợi nhuận cao với sự hỗ trợ hết sức đắc lực của công nghệ thông tin trong tổ chức và hoạt động. Tại nước ta nghành du lịch có thể được xem là một nghành lâu đời. Nhưng trong những năm gần đây, nghành du lịch mới thực sự phát triển về quy mô cũng như ảnh hưởng của nó trong lĩnh vực kinh tế xã hội ngày các được mở rộng. Tuy nhiên, việc ứng dụng công nghệ thông tin cho nghành du lịch nước ta còn hạn chế.  
Trong những năm gần đây, với sự phát triển mạnh mẽ của CNTT, việc ứng dụng CNTT vào việc giải quyết các vấn đề trong cuộc sống không còn xa lạ với chúng ta. Do đó việc lập trình đã được hỗ trợ tối đa bằng nhiểu ngôn ngữ lập trình khác nhau. Việc lập trình đã trở nên đơn giản hơn và không còn là vấn đề quan trọng. Vấn đề mà các nhà tin học hiện nay quan tâm là làm thế nào để triển khai những dự án tin học hoá một cách rõ ràng, logic, khoa học và đầy tính mở. Như vậy cần phải mô hình hoá được toàn bộ hệ thống một cánh đầy đủ và chính xác trên cơ sở đó để phát triển được phần mềm có tính tương thích cao, đáp ứng được nhu cầu của người sử dụng.  
Xuất phát từ nhu cầu thực tế ứng dụng của CNTT vào quản lý khách sạn, chúng em chọn thực hiện đề tài Quản Lý Khách Sạn làm đề tài nghiên cứu.

**Mục lục**

[**CHƯƠNG I. TỔNG QUAN** 10](#_Toc42118342)

[**1.1 Tổng quan về vấn đề nghiên cứu** 10](#_Toc42118343)

[1.1.1 Khảo sát thực trạng 10](#_Toc42118344)

[1.1.2 Những tồn tại, khó khăn 10](#_Toc42118345)

[**1.2 Nhiệm vụ đồ án** 11](#_Toc42118346)

[1.2.1 Đối tượng và phạm vi 11](#_Toc42118347)

[**1.3. Cơ sở tạo dựng đề tài** 12](#_Toc42118348)

[1.3.1 Phương pháp thiết kế và phân tích hướng đối tượng UML 12](#_Toc42118349)

[1.3.2 Kiến trúc 3 tầng 14](#_Toc42118350)

[**1.4 Cấu trúc đồ án** 15](#_Toc42118352)

[1.4.1 Yêu cầu nghiệp vụ 15](#_Toc42118353)

[1.4.2 Yêu cầu lưu trữ 15](#_Toc42118354)

[1.4.3 Các yêu cầu phi chức năng 16](#_Toc42118355)

[**CHƯƠNG II. CƠ SỞ LÝ THUYẾT** 17](#_Toc42118356)

[**2.1 Các khái niệm và cơ chế hoạt động** 17](#_Toc42118357)

[**2.2 Tổng quan về công nghệ** 17](#_Toc42118358)

[2.2.1 Ngôn ngữ C# 17](#_Toc42118359)

[2.2.2 ASP.NET framework 18](#_Toc42118360)

[2.2.3 Hệ quản trị cơ sở dữ liệu SQL Server 19](#_Toc42118361)

[2.2.4 Tổng quan về ASP.NET MVC5 20](#_Toc42118362)

[2.2.5 Môi trường lập trình Visual studio 2019 22](#_Toc42118364)

[**CHƯƠNG III. PHÂN TÍCH THIẾT KẾ HỆ THỐNG** 24](#_Toc42118366)

[**3.1 Xác định yêu cầu hệ thống** 24](#_Toc42118367)

[3.1.1 Yêu cầu hệ thống 24](#_Toc42118368)

[3.1.2 Yêu cầu chức năng 24](#_Toc42118369)

[**3.2 Các ràng buộc, yêu cầu sau khi hoàn thành** 24](#_Toc42118370)

[**3.3 Các sơ đồ phân tích thiết kế hệ thống** 25](#_Toc42118371)

[3.3.1 Usecase diagram 25](#_Toc42118372)

[*3.3.2* *Sequence diagram* 27](#_Toc42118376)

[**3.4 Class diagram** 32](#_Toc42118383)

[Hình 3.4 Biểu đồ lớp 32](#_Toc42118384)

[**3.5 Sơ đồ chức năng** 33](#_Toc42118385)

[Hình 3.5 Sơ đồ chức năng hệ thống 33](#_Toc42118386)

[**3.6 Đặc tả use case** 34](#_Toc42118387)

[3.6.1 Use case đăng nhập 34](#_Toc42118388)

[3.6.2 Use case đăng xuất 35](#_Toc42118389)

[3.6.3 Use case Đặt phòng 36](#_Toc42118390)

[3.6.4 Use case Tra cứu phòng 38](#_Toc42118391)

[3.6.5 Use case tìm kiếm dịch vụ 39](#_Toc42118392)

[3.6.6 Use case lập hóa đơn 40](#_Toc42118393)

[**3.7 Mô hình thực thể ERD và Diagram của hệ thống** 41](#_Toc42118394)

[**3.8 Bảng cơ sở dữ liệu chi tiết thuộc tính** 43](#_Toc42118397)

[**3.9 Mô tả chức năng hệ thống** 47](#_Toc42118409)

[*3.9.1 Chức năng người dùng:* 47](#_Toc42118410)

[3.9.2 Chức năng của admin, quản trị: 47](#_Toc42118411)

[**CHƯƠNG IV KẾT QUẢ THỰC NGHIỆM** 48](#_Toc42118412)

[**4.1 Giao diện của người dùng** 48](#_Toc42118413)

[4.1.1 Giao diện đăng ký, đăng nhập: 48](#_Toc42118414)

[4.1.2 Giao diện trang chủ: 49](#_Toc42118417)

[4.1.3 Giao diện danh mục phòng, khách sạn: 50](#_Toc42118419)

[4.1.4 Giao diện đặt phòng, đánh giá: 50](#_Toc42118421)

[4.1.5 Giao diện thanh toán: 51](#_Toc42118423)

[**4.2 Giao diện chức năng của admin, quản trị** 52](#_Toc42118425)

[4.2.1 Giao diện quản lý đặt phòng: 52](#_Toc42118426)

[4.2.2 Giao diện quản lý đơn đặt phòng: 53](#_Toc42118428)

[4.2.3 Giao diện quản lý khách hàng: 53](#_Toc42118430)

[**CHƯƠNG V. KẾT LUẬN VÀ KIẾN NGHỊ** 54](#_Toc42118432)

[**5.1 Những vấn đề đã giải quyết** 54](#_Toc42118433)

[**5.2 Những vấn đề chưa giải quyết và thiếu sót** 54](#_Toc42118434)

[**5.3 Hướng phát triển mở rộng chương trình** 55](#_Toc42118435)

[**TÀI LIỆU THAM KHẢO** 56](#_Toc42118436)

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

……………………………………………………………….………………………………..

*TP.HCM, ngày…tháng…năm 2020*

**NGƯỜI HƯỚNG DẪN**

**DANH MỤC CÁC BẢNG**

[Bảng khách sạn 43](#_Toc42117652)

[Bảng khách hàng 43](#_Toc42117653)

[Bảng phòng 44](#_Toc42117654)

[Bảng loại phòng 44](#_Toc42117655)

[Bảng khu vực 44](#_Toc42117656)

[Bảng tiện nghi 44](#_Toc42117657)

[Bảng hình ảnh 45](#_Toc42117658)

[Bảng điểm đánh giá 45](#_Toc42117659)

[Bảng đơn đặt phòng 46](#_Toc42117660)

[Bảng chi tiết đặt phòng 46](#_Toc42117661)

[Bảng tài khoản 46](#_Toc42117662)

**DANH MỤC CÁC HÌNH ẢNH**

[Hình 1.3.2 Mô hình kiến trúc 3 lớp 14](#_Toc42117256)

[Hình 2.2.4 Mô hình hoạt động MVC 21](#_Toc42117257)

[Hình 2.2.5 Hoạt động của SQL Server 23](#_Toc42117258)

[Hình 3.3.1a Usecase khách hàng đặt phòng 25](#_Toc42117259)

[Hình 3.3.1b Usecase Lễ tân 26](#_Toc42117260)

[Hình 3.3.1c. Usecase Quản trị 26](#_Toc42117261)

[Hình 3.3.2a Sequence QL thuê phòng 27](#_Toc42117262)

[Hình 3.3.2b Sequence QL khách hàng 28](#_Toc42117263)

[Hình 3.3.2c Sequence QL Tài khoản 28](#_Toc42117264)

[Hình 3.3.2d Sequence QL Nhân Viên 29](#_Toc42117265)

[Hình 3.3.2e Sequence Lập hóa đơn 30](#_Toc42117266)

[Hình 3.3.2f Sequence Thuê dịch vụ 31](#_Toc42117267)

[Hình 3.4 Biểu đồ lớp 32](#_Toc42117268)

[Hình 3.5 Sơ đồ chức năng hệ thống 33](#_Toc42117269)

[Hình 3.7.1a Mô hình ERD 41](#_Toc42117270)

[Hình 3.7.1b Diagram của hệ thống 42](#_Toc42117271)

[Hình 4.1.1a Giao diện đăng ký 48](#_Toc42117272)

[Hình 4.1.1b Giao diện đăng nhập 48](#_Toc42117273)

[Hình 4.1.2 Giao diện trang chủ 49](#_Toc42117274)

[Hình 4.1.3 Giao diện phòng, khách sạn 50](#_Toc42117275)

[Hình 4.1.4 Giao diện đặt phòng, đánh giá 51](#_Toc42117276)

[Hình 4.1.5 Giao diện thanh toán 51](#_Toc42117277)

[Hình 4.2.1 Giao diện admin quản lý phòng 52](#_Toc42117278)

[Hình 4.2.2 Giao diện admin quản lý đơn đặt phòng 53](#_Toc42117279)

[Hình 4.2.3 Giao diện admin quản lý khách hàng 53](#_Toc42117280)

# **CHƯƠNG I. TỔNG QUAN**

## **1.1 Tổng quan về vấn đề nghiên cứu**

### 1.1.1 Khảo sát thực trạng

Ngày nay, trong thời đại Công nghệ thông tin bùng nổ trên toàn cầu thì các quốc gia trên thế giới dù là phát triển hay đang phát triển đều cố gắng áp dụng tin học vào mọi mặt của đời sống xã hội. Tin học luôn có mặt mọi lúc, mọi nơi nhằm hỗ trợ cho các ngành nghiên cứu đạt được những thành tựu to lớn cũng như để hiện đại hóa quy trình quản lý sản xuất kinh doanh nhằm đạt hiệu quả kinh tế cao nhất. Chính vì vậy, trong thời gian gần đây mọi ngành nghề đã chú trọng đến việc xây dựng các phần mềm ứng dụng tin học trong công tác quản lý.

Với những lợi ích hiển nhiên do Công nghệ thông tin mang lại, các nhà quản lý đã kịp thời đưa những ứng dụng tin học vào phục vụ cho công tác kinh doanh, đặc biệt là quản lý khách sạn.

Tùy thuộc vào quy mô, mục đích thị trường, mức độ phục vụ, quyền sở hữu mà ta phân tích thiết kế sao cho người quản lý nắm được nhanh chóng chính xác đồng thời giảm được các chi phí, các thao tác thủ công và đạt hiệu quả kinh tế cao nhất. Qua đó nó cũng giúp người quản lý nắm bắt được số lượng khách, tình trạng phòng, thời gian lưu trú của khách… để làm cơ sở cho việc thanh toán sau này.

Chúng em quyết định xây dựng hệ thống website “Quản lý đặt phòng khách sạn” nhằm giải quyết và đáp ứng một cách hiệu quả các nhu cầu về mặt quản lý thông tin trong khách sạn trong điều kiện cơ sở vật chất và cơ sở hạ tầng ở Việt Nam còn nhiều khoảng cách so với các nước phát triển trên thế giới.

### 1.1.2 Những tồn tại, khó khăn

Những xử lý thủ tục đặt phòng, nhận phòng, trả phòng mất thời gian và vẫn còn thực hiện thủ công.

Khó kiểm tra phòng hư, kiểm soát tình trạng phòng (phòng trống, phòng đang sử dụng…) do đó làm giảm chất lượng dịch vụ của khách sạn.

Khó khăn trong việc quản lý nhân sự, quản lý các thông tin khách hàng, thông tin hóa đơn, thông tin phiều đăng ký, đăng ký dịch vụ.

## **1.2 Nhiệm vụ đồ án**

### 1.2.1 Đối tượng và phạm vi

* Đối tượng

Hệ thống quản lý website được xây dựng hướng đến các khách hàng đi du lịch có nhu cầu đặt phòng nghỉ ngơi và sử dụng dịch vụ.

* Phạm vi

Phần mềm được thực hiện và áp dụng trên toàn quốc.

* Phát biểu vấn đề:

Với số lượng khách hàng cần sử dụng dịch vụ hoặc đặt phòng tìm kiếm một website an toàn về việc đặt phòng online với một khách sạn đẹp, sang trọng, đầy đủ tiện nghi. Sự can thiệp của website “Quản lý đặt phòng khách sạn” sẽ mang lại hiệu quả hoạt động cao hơn.

* Lợi ích mang lại:
* Tạo sự tiện dụng, nhanh chóng cho người sử dụng.
* Tự động hóa công tác quản lý.
* Tiết kiệm được nhiều thời gian.
* Mô tả phương án tổng quan
* Phương án lưu trữ
* Cơ sở dữ liệu tập trung là phương án đưa dữ liệu về một nơi.
* Giúp quản lý chặt chẽ hơn, tăng tính bảo mật vì mọi thao tác trên dữ liệu chỉ được thực hiện ở một nơi.
* Tốc độ thao tác dữ liệu hạn chế do nhiều thao tác cùng một lúc vào một dữ liệu ở một nơi.
* Cơ sở dữ liệu phân tán.
* Tốc độ thao tác dữ liệu nhanh hơn cơ sở dữ liệu tập trung.
* Thiết kế dữ liệu tương đối khó khan, không chặt chẽ, có thể bị lỗi không cập nhật cho tất cả nơi lưu trữ.
* Chỉ phù hợp cho cơ sở dữ liệu lớn, có khoảng cách địa lý.
* Các hệ quản trị cơ sở dữ liệu: Microsoft Acces, Microsoft SQL Server, MySQL, Oracel…
* Phương án khả thi
* Với các mô hình dữ liệu trên, mô hình dữ liệu được áp dụng cho hệ thống là mô hình dữ liệu tập trung vì những lợi ích sau.
* Với sự phát triển của công nghệ hiện nay thì tốc độ đường truyền, dung lượng bộ nhớ không là vấn đề lớn, cơ sở dữ liệu tập trung giúp chúng ta dễ dàng sao lưu, phục hồi, bảo đảm an toàn dữ liệu.
* Về mặt phần cứng thì chi phí đầu tư cho mô hình này không cao. Về mặt bảo mật dữ liệu, cần phân quyền đối với người sử dụng hệ thống, mặt khác giúp việc quản lí được chặt chẽ hơn.

## **1.3. Cơ sở tạo dựng đề tài**

### 1.3.1 Phương pháp thiết kế và phân tích hướng đối tượng UML

“Phương pháp phân tích hướng đối tượng(OOAD)” là thuật ngữ thông dụng hiện thời của ngành công nghiệp phần mềm. Các công ty đang nhanh chóng tìm cách áp dụng và tích hợp công nghệ mới này vào các ứng dụng của họ. Thật sự là đa phần các ứng dụng hiện thời đều mang tính hướng đối tượng. Lối tiếp cận hướng đối tượng là một lối tư duy về vấn đề theo lối ánh xạ các thành phần trong bài toán vào các đối tượng ngoài đời thực. Với lối tiếp cận này, chúng ta chia ứng dụng thành các thành phần nhỏ, gọi là các đối tượng, chúng tương đối độc lập với nhau.

Sau đó ta có thể xây dựng ứng dụng bằng cách chắp các đối tượng đó lại với nhau. Hãy nghĩ đến trò chơi xây lâu đài bằng các mẫu gỗ. Bước đầu tiên là tạo hay mua một vài loại mẫu gỗ căn bản, từ đó tạo nên các khối xây dựng căn bản của mình. Một khi đã có các khối xây dựng đó, bạn có thể chắp ráp chúng lại với nhau để tạo lâu đài. Tương tự như vậy một khi đã xây dựng một số đối tượng căn bản trong thế giới máy tính, bạn có thể chắp chúng lại với nhau để tạo ứng dụng của mình.

Các ưu điểm của phương pháp phân tích hướng đối tượng:

Tính tái sử dụng: Phương pháp phân tích và thiết kế hướng đối tượng thực hiện theo các thuật ngữ và khái niệm của phạm vi lĩnh vực ứng dụng (tức là của doanh nghiệp hay đơn vị mà hệ thống tương lai cần phục vụ), nên nó tạo sự tiếp cận tương ứng giữa hệ thống và vấn đề thực ngoài đời.

Các giai đoạn của chu trình phát triển phần mềm với mô hình hướng đối tượng

+ Phân tích hướng đối tượng (Object Oriented Analysis- OOA):

Là giai đọan phát triển một mô hình chính xác và súc tích của vấn đề, có thành phần là các đối tượng và khái niệm đời thực, dễ hiểu đối với người sử dụng. Trong giai đoạn OOA, vấn đề được trình bày bằng các thuật ngữ tương ứng với các đối tượng có thực. Thêm vào đó, hệ thống cần phải được định nghĩa sao cho người không chuyên Tin học có thể dễ dàng hiểu được.

+ Thiết kế hướng đối tượng (Object Oriented Design- OOD):

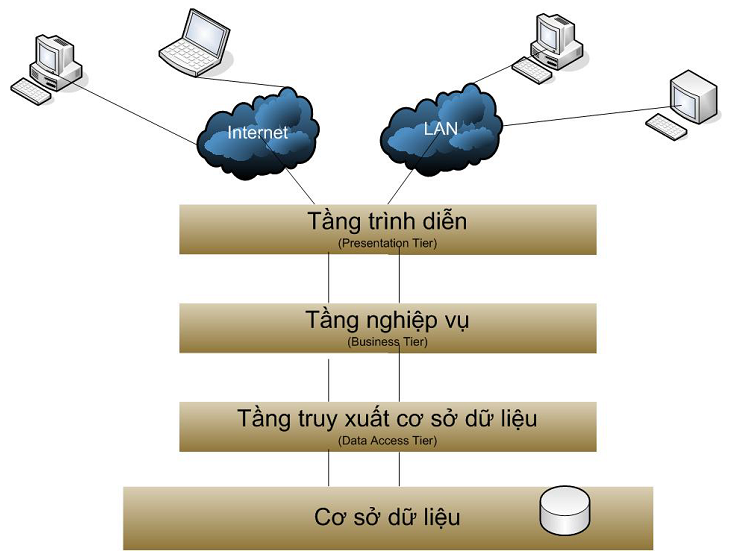
Là giai đoạn tổ chức chương trình thành các tập hợp đối tượng cộng tác, mỗi đối tượng trong đó là thực thể của một lớp. Các lớp là thành viên của một cây cấu trúc với mối quan hệ thừa kế. OOD tập trung vào việc cải thiện kết quả của OOA, tối ưu hóa giải pháp đã được cung cấp trong khi vẫn đảm bảo thoả mãn tất cả các yêu cầu đã được xác lập. Trong giai đoạn OOD, nhà thiết kế định nghĩa các chức năng, thủ tục (operations), thuộc tính (attributes) cũng như mối quan hệ của một hay nhiều lớp (class) và quyết định chúng cần phải được điều chỉnh sao cho phù hợp với môi trường phát triển. Đây cũng là giai đoạn để thiết kế ngân hàng dữ liệu và áp dụng các kỹ thuật tiêu chuẩn hóa.

+ Lập trình hướng đối tượng (Object Oriented Progamming- OOP):

Giai đoạn xây dựng phần mềm có thể được thực hiện sử dụng kỹ thuật lập trình hướng đối tượng. Đó là phương thức thực hiện thiết kế hướng đối tượng qua việc sử dụng một ngôn ngữ lập trình có hỗ trợ các tính năng hướng đối tượng. Kết quả chung cuộc của giai đoạn này là một loạt các code chạy được, nó chỉ được đưa vào sử dụng sau khi đã trải qua nhiều vòng quay của nhiều bước thử nghiệm khác nhau.

### 1.3.2 Kiến trúc 3 tầng

Hệ thống website quản lý đồ án thực tập được thiết kế theo mô hình kiến trúc 3 lớp.



Hình 1.3.2 Mô hình kiến trúc 3 lớp

Website quản lý đồ án tham chiếu đến các dịch vụ web (tầng nghiệp vụ) mà không tham chiếu trực tiếp đến tầng truy xuất cơ sở dữ liệu. Tầng trình diễn là giao tiếp giữa người dùng và với dịch vụ web. Mọi thao tác của người dùng với hệ thống đều thông qua tầng trình diễn. Tầng trình diễn sau khi nhận yêu cầu người dùng, cung cấp thông tin cho tầng nghiệp vụ xử lý.

Tầng nghiệp vụ (Business Tier)

Tầng nghiệp vụ được tổ chức theo các dịch vụ web. Tầng này có thể triển khai phân tán, các dịch vụ web có thể chạy trên các máy chủ khác nhau nhằm tăng hiệu năng của ứng dụng web. Với tầng nghiệp vụ thì không phụ thuộc giao diện ứng dụng cũng như không cần biết cách thức truy cập, xử lý cơ sở dữ liệu ra sao.

Tầng này đơn thuần thể hiện nghiệp vụ của hệ thống nên bất kỳ thay đổi nào về nghiệp vụ hầu như chỉ thay đổi trong tầng này mà không phải sửa lại nhiều 2 tầng còn lại. Đây cũng là một trong những ưu điểm của kiến trúc 3 tầng. Lớp dịch vụ web chỉ triệu gọi các phương thức truy xuất cơ sở dữ liệu từ tầng DAO mà không cần biết truy xuất như thế nào. Mọi nghiệp vụ của hệ thống đều được thể hiện trong tầng nghiệp vụ.

Tầng truy cập cơ sở dữ liệu (Data Access Tier)

Tầng truy xuất cơ sở dữ là tầng trực tiếp liên kết với csdl của hệ thống. Mọi chức năng yêu cầu giao tiếp với cơ sở dữ liệu đều được thực hiện ở đây. Tầng này được tổ chức thành các gói (package), mỗi gói là 1 thư viện dùng để truy xuất và thao tác trên các thực thể tương ứng.

## **1.4 Cấu trúc đồ án**

### 1.4.1 Yêu cầu nghiệp vụ

* Quản lý người dùng.
* Quản lý đăng ký thuê phòng.
* Quản lý nhận trả phòng.
* Quản lý hóa đơn.
* Quản lý khách hàng.
* Quản lý dịch vụ.
* Quản lý phòng.
* Quản lý thiết bị trong phòng.
* Quản lý tình trạng phòng.

### 1.4.2 Yêu cầu lưu trữ

* Lưu trữ thông tin người dùng.
* Lưu trữ thông tin khách hàng thuê phòng.
* Lưu trữ thông tin khách hàng nhận, trả phòng.
* Lưu trữ thông tin phòng.
* Lưu trữ thông tin tình trạng phòng.
* Lưu trữ thông tin loại phòng.
* Lưu trữ thông tin thiết bị.
* Lưu trữ thông tin dịch vụ.
* Lưu trữ thông tin sử dụng dịch vụ của khách hàng.
* Lưu trữ thông tin loại dịch vụ.
* Lưu trữ thông tin đơn vị tính.
* Lưu trữ qui định khách sạn.
* Lưu trữ chính sách trả phòng.
* Lưu trữ thông tin phiếu đăng ký thuê phòng.
* Lưu trữ thông tin phiếu nhận phòng.
* Lưu trữ thông tin hóa đơn.

### 1.4.3 Các yêu cầu phi chức năng

* Giao diện hệ thống phải dễ sử dụng, trực quan, thân thiện với mọi người dùng.
* Cho phép truy cập dữ liệu đa người dùng.
* Tính bảo mật và độ an toàn cao.
* Tốc độ xử lý của hệ thống phải nhanh chóng và chính xác.

# **CHƯƠNG II. CƠ SỞ LÝ THUYẾT**

## **2.1 Các khái niệm và cơ chế hoạt động**

Một website là tập hợp của 2 thành phần cơ bản, bao gồm front-end và backend. Front-end là phần giao diện mà người sử dụng website sẽ trực tiếp nhìn thấy trên trình duyệt và thao tác với nó. Back-end là phần được thực thi trên máy chủ server, tiếp nhận các yêu cầu (request) từ người dùng đang tương tác với giao diện front-end, tiến hành xử lý, tương tác với cơ sở dữ liệu (database) để truy xuất hoặc chỉnh sửa các thông tin, sau đó trả về cho người dùng và hiển thị trên phần front-end.

Một website muốn hoạt động được trên môi trường trực tuyến cần phải có một máy chủ để thực thi. Server là nơi chứa website, các máy người dùng sẽ đưa ra các yêu cầu và gửi lên máy server này để xử lý. Tuy nhiên, đầu tư một máy server sẽ tốn kém, do đó người ta sử dụng dịch vụ hosting. - Đồng thời, website cần có một domain (tên miền) để dễ nhớ và cũng để quảng bá cho hình ảnh của công ty, tổ chức đang sở hữu website đó.

Có nhiều ngôn ngữ và công nghệ, framework hiện có được dùng để viết website. Em chọn công nghệ ASP.NET MVC để thực hiện đồ án này, vì phù hợp với yêu cầu công việc và nhận thấy rằng ASP.NET MVC hiện tại cũng đang là một công nghệ đang “hot” đối với các nhà phát triển web.

## **2.2 Tổng quan về công nghệ**

### 2.2.1 Ngôn ngữ C#

C# là một ngôn ngữ lập trình đơn giản, hiện đại, mục đích tổng quát, hướng đối tượng được phát triển bởi Microsoft và được phê chuẩn bởi European Computer Manufacturers Association (ECMA) và International Standards Organization (ISO). - C# được phát triển bởi Anders Hejlsberg và team của ông trong khi phát triển .NET Framework.

C# được thiết kế cho Common Language Infrastructure (CLI), mà gồm Executable Code và Runtime Environment, cho phép chúng ta sử dụng các ngôn ngữ high-level đa dạng trên các nền tảng và cấu trúc máy tính khác nhau. - Cấu trúc C# khá gần với các ngôn ngữ high-level truyền thống, C và C++, và là một ngôn ngữ lập trình hướng đối tượng. Nó có sự giống nhau mạnh mẽ với Java, nó có nhiều đặc điểm lập trình mạnh mẽ mà làm cho nó trở nên ưa thích với các lập trình viên trên toàn thế giới.

Dưới đây là các đặc điểm quan trọng của C#:

* Điều kiện Boolean.
* Tự động dọn rác bởi Garbage-Collector (GC).
* Thư viện chuẩn (Standard Library).
* Assembly.
* Property và sự kiện (Event).
* Delegate Quản lý sự kiện.
* Dễ dàng để sử dụng Generic.
* Indexer.
* Biên dịch có điều kiện (Conditional Compilation).
* Đa luồng dễ dàng (Multithreading).
* LINQ và Lambda Expression.
* Tích hợp với Windows.

### 2.2.2 ASP.NET framework

ASP.NET framework là một nền tảng lập trình và cũng là một nền tảng thực thi ứng dụng chủ yếu trên hệ điều hành Microsoft Windows được phát triển bởi Microsoft. Các chương trình được viết trên nền tảng .NET framework sẽ được triển khai trong môi trường phần mềm (ngược lại với môi trường phần cứng) được biết đến với tên Common Language Runtime (CLR). Môi trường phần mềm này là một máy ảo trong đó cung cấp các dịch vụ như an ninh phần mềm (security), quản lý bộ nhớ (memory management), và các xử lý lỗi ngoại lệ (exception handling).

ASP.NET framework bao gồm tập các thư viện lập trình lớn, và những thư viện này hỗ trợ việc xây dựng các chương trình phần mềm như lập trình giao diện; truy cập, kết nối cơ sở dữ liệu; ứng dụng web; các giải thuật, cấu trúc dữ liệu; giao tiếp mạng... CLR cùng với bộ thư viện này là 2 thành phần chính của .NET framework.

ASP.NET framework đơn giản hóa việc viết ứng dụng bằng cách cung cấp nhiều thành phần được thiết kế sẵn, người lập trình chỉ cần học cách sử dụng và tùy theo sự sáng tạo mà gắn kết các thành phần đó lại với nhau. Nhiều công cụ được tạo ra để hỗ trợ xây dựng ứng dụng .NET, và IDE (Integrated Developement Environment) được phát triển và hỗ trợ bởi chính Microsoft là Visual Studio.

ASP.NET Framework là thành quả tối ưu của sự kết hợp công sức và trí tuệ của Microsoft, nhằm tạo ra một nền tảng cho việc xây dựng và triển khai nhanh chóng các dịch vụ và ứng dụng Web XML. Tầm nhìn của nền tảng ASP .NET Framework kết hợp một mô hình lập trình đơn giản, dễ sử dụng với các giao thức mở và biến đổi được của Internet. Để đạt được tầm nhìn này, việc thiết kế ASP .NET Framework nhằm một số mục đích:

Khả nǎng biến đổi được thông qua một kiến trúc "ghép nối lỏng": Đa số các hệ thống lớn, biến đổi được trên thế giới được xây dựng trên những kiến trúc không đồng bộ dựa trên nền thông điệp (message-based). Nhưng công việc xây dựng các ứng dụng trên một kiến trúc như vậy thường phức tạp và có ít các công cụ hơn so với những môi trường phát triển ứng dụng N lớp (N-tier) "ghép nối chặt".  ASP.NET Framework được xây dựng để đem lại những lợi thế về nǎng suất của kiến trúc "ghép nối chặt" với khả nǎng biến đổi được và vận hành với nhau của kiến trúc "ghép nối lỏng".

### 2.2.3 Hệ quản trị cơ sở dữ liệu SQL Server

Hệ quản trị cơ sở dữ liệu (Database Management System - DBMS), là phần mềm hay hệ thống được thiết kế để quản trị một cơ sở dữ liệu. Cụ thể, các chương trình thuộc loại này hỗ trợ khả năng lưu trữ, sửa chữa, xóa và tìm kiếm thông tin trong một cơ sở dữ liệu (CSDL). Có rất nhiều loại hệ quản trị CSDL khác nhau: từ phần mềm nhỏ chạy trên máy tính cá nhân cho đến những hệquản trị phức tạp chạy trên một hoặc nhiều siêu máy tính. - SQL Server là hệ thống quản trị cơ sở dữ liệu quan hệ (Relational DataBase Management System- RDBMS) sử dụng các lệnh chuyển Transaction-SQL để trao đổi dữ liệu giữa Client Computer và Server Computer.

- SQL Server có một số đặc tính sau:

Cho phép quản trị một hệ CSDL lớn (lên đến vài tega byte), có tốc độ xử lý dữ liệu nhanh đáp ứng yêu cầu về thời gian.

Cho phép nhiều người cùng khai thác trong một thời điểm đối với một CSDL và toàn bộ quản trị CSDL (lên đến vài chục ngàn user).

Có hệ thống phân quyền bảo mật tương thích với hệ thống bảo mật của công nghệ NT (Network Technology), tích hợp với hệ thống bảo mật của Windows NT hoặc sử dụng hệ thống bảo vệ độc lập của SQL Server.

Hỗ trợ trong việc triển khai CSDL phân tán và phát triển ứng dụng trên Internet.

Cho phép lập trình kết nối với nhiều ngôn ngữ lập trình khác dùng xây dựng các ứng dụng đặc thù (Visual Basic, C, C++, ASP, ASP. NET, XML...).

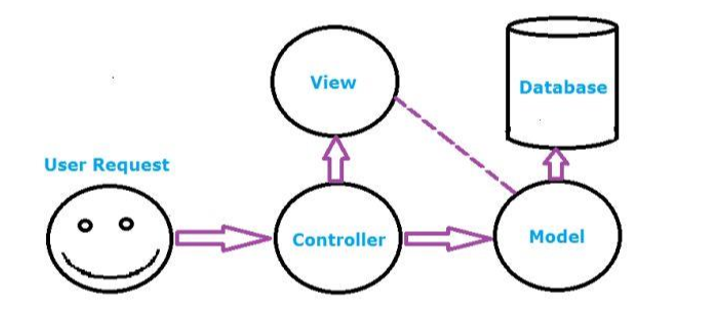
Sử dụng câu lệnh truy vấn dữ liệu Transaction-SQL (Access là SQL, Oracle là PL/SQL).

SQL Server là hệ quản trị CSDL hoạt động trên mạng, có thể thực hiện trao đổi dữ liệu theo nhiều mô hình mạng khác nhau, nhiều giao thức và phương thức truyền tin khác nhau.

### 2.2.4 Tổng quan về ASP.NET MVC5

Mẫu kiến trúc Model – View – Controller được sử dụng nhằm chia ứng dụng thành ba thành phần chính: Model, View và Controller. Nền tảng ASP.NET MVC giúp cho chúng ta có thể tạo được các ứng dụng web áp dụng mô hình MVC thay vì tạo ứng dụng theo mẫu ASP.NET Web Forms. Nền tảng ASP.NET MVC có đặc điểm nổi bật là nhẹ (lighweigt), dễ kiểm thử phần giao diện (so với ứng dụng Web Forms), tích hợp các tính năng có sẵn của ASP.NET. Nền tảng ASP.NET MVC được định nghĩa trong namespace System Web.Mvc và là một phần của namespace System.Web.

MVC là một mẫu thiết kế (Design Pattern) chuẩn mà nhiều lập trình viên đã quen thuộc. Một số loại ứng dụng web sẽ thích hợp với kiến trúc MVC. Số khác vẫn thích hợp với ASP.NET Web Forms và cơ chế postbacks. Đôi khi có những ứng dụng kết hợp cả hai kiến trúc trên. - Nền tảng MVC bao gồm các thành phần dưới đây:

****

Hình 2.2.4 Mô hình hoạt động MVC

**-** Models**:** Các đối tượng Models là một phần của ứng dụng, các đối tượng này thiết lập logic của phần dữ liệu của ứng dụng. Thông thường, các đối tượng model lấy và lưu trạng thái của model trong CSDL. Ví dụ như, một đối tượng Product (sản phẩm) sẽ lấy dữ liệu từ CSDL, thao tác trên dữ liệu và sẽ cập nhật dữ liệu trở lại vào bảng Products ở SQL Server. Trong các ứng dụng nhỏ, model thường là chỉ là một khái niệm nhằm phân biệt hơn là được cài đặt thực thụ, ví dụ, nếu ứng dụng chỉ đọc dữ liệu từ CSDL và gởi chúng đến view, ứng dụng không cần phải có tầng model và các lớp liên quan. Trong trường hợp này, dữ liệu được lấy như là một đối tượng model (hơn là tầng model).

**-** Views**:** Views là các thành phần dùng để hiển thị giao diện người dùng (UI). Thông thường, view được tạo dựa vào thông tin dữ liệu model. Ví dụ như, view dùng để cập nhật bảng Products sẽ hiển thị các hộp văn bản, dropdown list, và các check box dựa trên trạng thái hiện tại của một đối tượng Product.

**-** Controllers**:** Controller là các thành phần dùng để quản lý tương tác người dùng, làm việc với model và chọn view để hiển thị giao diện người dùng. Trong một ứng dụng MVC, view chỉ được dùng để hiển thị thông tin, controller chịu trách nhiệm quản lý và đáp trả nội dung người dùng nhập và tương tác với người dùng. Ví dụ, controller sẽ quản lý các dữ liệu người dùng gởi lên (query-string values) và gởi các giá trị đó đến model, model sẽ lấy dữ liệu từ CSDL nhờ vào các giá trị này.

Mẫu MVC giúp chúng ta tạo được các ứng dụng mà chúng phân tách rạch ròi các khía cạnh của ứng dụng (logic về nhập liệu, logic xử lý tác vụ và logic về giao diện). Mẫu MVC chỉ ra mỗi loại logic kể trên nên được thiếp lập ở đâu trên ứng dụng. Logic giao diện (UI logic) thuộc về views. Logic nhập liệu (input logic) thuộc về controller. Và logic tác vụ (Business logic – là logic xử lý thông tin, mục đích chính của ứng dụng) thuộc về model. Sự phân chia này giúp bạn giảm bớt được sự phức tạp của ứng dụng và chỉ tập trung vào mỗi khía cạnh cần được cài đặt ở mỗi thời điểm.

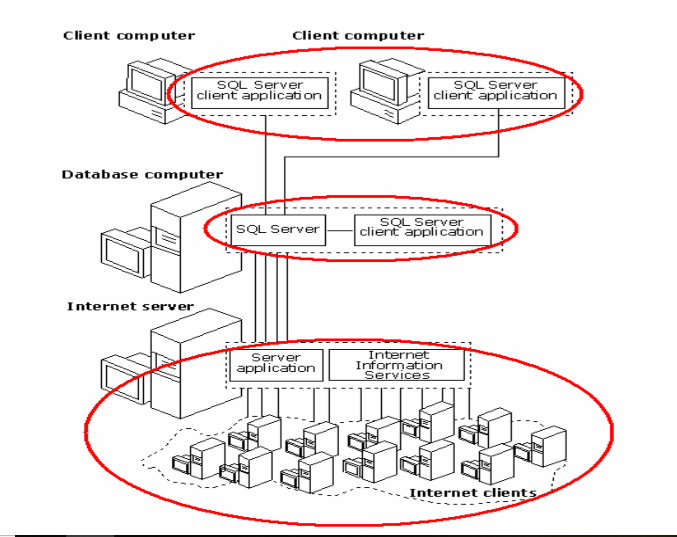
### 2.2.5 Môi trường lập trình Visual studio

Microsoft Visual Studio là một môi trường phát triển tích hợp từ Microsoft. Nó được sử dụng để phát triển chương trình máy tính cho Microsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như Windows API, Windows Forms, Windows Presentation Foundation, Windows Store và Microsoft Silverlight. Nó có thể sản xuất cả hai ngôn ngữ máy và mã số quản lý.

Visual Studio bao gồm một trình soạn thảo mã hỗ trợ IntelliSense cũng như cải tiến mã nguồn. Trình gỡ lỗi tích hợp hoạt động cả về trình gỡ lỗi mức độ mã nguồn và gỡ lỗi mức độ máy. Công cụ tích hợp khác bao gồm một mẫu thiết kế các hình thức xây dựng giao diện ứng dụng, thiết kế web, thiết kế lớp và thiết kế giản đồ cơ sở dữ liệu. Nó chấp nhận các plug-in nâng cao các chức năng ở hầu hết các cấp bao gồm thêm hỗ trợ cho các [hệ thống quản lý phiên bản](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BB%91ng_qu%E1%BA%A3n_l%C3%BD_phi%C3%AAn_b%E1%BA%A3n) (như [Subversion](https://vi.wikipedia.org/wiki/Subversion)) và bổ sung thêm bộ công cụ mới như biên tập và thiết kế trực quan cho các [miền ngôn ngữ cụ thể](https://vi.wikipedia.org/w/index.php?title=Mi%E1%BB%81n_ng%C3%B4n_ng%E1%BB%AF_c%E1%BB%A5_th%E1%BB%83&action=edit&redlink=1) hoặc bộ công cụ dành cho các khía cạnh khác trong [quy trình phát triển phần mềm](https://vi.wikipedia.org/wiki/Quy_tr%C3%ACnh_ph%C3%A1t_tri%E1%BB%83n_ph%E1%BA%A7n_m%E1%BB%81m).

Giống như bất kỳ [IDE](https://vi.wikipedia.org/wiki/IDE) khác, nó bao gồm một [trình soạn thảo mã](https://vi.wikipedia.org/w/index.php?title=Tr%C3%ACnh_so%E1%BA%A1n_th%E1%BA%A3o_m%C3%A3&action=edit&redlink=1) hỗ trợ [tô sáng cú pháp](https://vi.wikipedia.org/w/index.php?title=T%C3%B4_s%C3%A1ng_c%C3%BA_ph%C3%A1p&action=edit&redlink=1) và [hoàn thiện mã](https://vi.wikipedia.org/w/index.php?title=Ho%C3%A0n_thi%E1%BB%87n_m%C3%A3&action=edit&redlink=1) bằng cách sử dụng [IntelliSense](https://vi.wikipedia.org/w/index.php?title=IntelliSense&action=edit&redlink=1) không chỉ cho các [biến](https://vi.wikipedia.org/wiki/Bi%E1%BA%BFn_(l%E1%BA%ADp_tr%C3%ACnh)), [hàm](https://vi.wikipedia.org/wiki/Ch%C6%B0%C6%A1ng_tr%C3%ACnh_con) và các [phương pháp](https://vi.wikipedia.org/wiki/Ph%C6%B0%C6%A1ng_th%E1%BB%A9c_l%E1%BA%ADp_tr%C3%ACnh) mà còn các cấu trúc ngôn ngữ như [vòng điều khiển](https://vi.wikipedia.org/w/index.php?title=Ki%E1%BB%83m_so%C3%A1t_l%C6%B0u_l%C6%B0%E1%BB%A3ng&action=edit&redlink=1) hoặc [truy vấn](https://vi.wikipedia.org/w/index.php?title=Language_Integrated_Query&action=edit&redlink=1). IntelliSense được hỗ trợ kèm theo cho các ngôn ngữ như [XML](https://vi.wikipedia.org/wiki/XML), [Cascading Style Sheets](https://vi.wikipedia.org/wiki/CSS) và [JavaScript](https://vi.wikipedia.org/wiki/JavaScript) khi phát triển các [trang web](https://vi.wikipedia.org/wiki/Trang_web) và các [ứng dụng web](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web). Các đề xuất tự động hoàn chỉnh được xuất hiện trong một hộp danh sách phủ lên trên đỉnh của trình biên tập mã.

Các trình biên tập mã Visual Studio cũng hỗ trợ cài đặt dấu trang trong mã để điều hướng nhanh chóng. Hỗ trợ điều hướng khác bao gồm [thu hẹp các khối mã lệnh](https://vi.wikipedia.org/w/index.php?title=Thu_h%E1%BA%B9p_c%C3%A1c_kh%E1%BB%91i_m%C3%A3_l%E1%BB%87nh&action=edit&redlink=1) và [tìm kiếm gia tăng](https://vi.wikipedia.org/w/index.php?title=T%C3%ACm_ki%E1%BA%BFm_gia_t%C4%83ng&action=edit&redlink=1), ngoài việc tìm kiếm văn bản thông thường và tìm kiếm [Biểu thức chính quy](https://vi.wikipedia.org/wiki/Bi%E1%BB%83u_th%E1%BB%A9c_ch%C3%ADnh_quy). Các trình biên tập mã cũng bao gồm một bìa kẹp đa mục và một danh sách công việc. Các trình biên tập mã hỗ trợ lưu lại các đoạn mã được lặp đi lặp lại nhằm để chèn vào mã nguồn sử dụng về sau. Một công cụ quản lý cho đoạn mã được xây dựng là tốt. Những công cụ này nổi lên như các cửa sổ trôi nổi có thể được thiết lập để tự động ẩn khi không sử dụng hoặc neo đậu đến các cạnh của màn hình.

****

Hình 2.2.5 Hoạt động của SQL Server

Visual Studio hỗ trợ nhiều ngôn ngữ lập trình khác nhau và cho phép trình biên tập mã và gỡ lỗi để hỗ trợ (mức độ khác nhau) hầu như mọi ngôn ngữ lập trình. Các ngôn ngữ tích hợp gồm có C, C++ và C++/CLI (thông qua Visual C++), VB.NET (thông qua Visual Basic.NET), C# (thông qua Visual C#) và F# (như của Visual Studio 2010). Hỗ trợ cho các ngôn ngữ khác như J++/J#, Python và Ruby thông qua dịch vụ cài đặt riêng rẽ. Nó cũng hỗ trợ XML/XSLT, HTML/XHTML, JavaScript và CSS.

# **CHƯƠNG III. PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

## **3.1 Xác định yêu cầu hệ thống**

### 3.1.1 Yêu cầu hệ thống

Máy chủ có khả năng tính toán nhanh, chính xác, lưu trữ lâu dài, bảo mật.

Hệ thống mạng đáp ứng khả năng truy cập lớn.

Cập nhật dữ liệu chính xác và nhanh nhất.

Thông tin có tính đồng bộ, phân quyền quản lý chặt chẽ.

Bảo mật tốt cho người quản trị hệ thống.

Thông báo, tin tức cũng như các tin tức cập nhật khác…

### 3.1.2 Yêu cầu chức năng

Hệ thống phải cập nhật, lưu trữ, tìm kiếm được tất cả các thông tin chi tiết về khách hàng, phòng trống, phòng đang sử dụng, các dịch vụ.

Cập nhật theo danh mục : Khách hàng, nhân viên, thông tin về các phòng, dịch vụ khác…

Thanh toán hóa đơn, sử dụng các dịch vụ, quản lý thống kê chi tiết.

Chức năng dành cho bộ phận lễ tân: Có thể tra cứu thông tin phòng, khách hàng, các dịch vụ của khách sạn.

Chức năng khách hàng: Tìm kiếm thông tin phòng theo tên hoặc số phòng, đặt phòng trước, tham khảo về các dịch vụ của khách sạn, bình luận, đánh giá.

Chức năng của admin: Luôn cập nhật thông tin và quản lý chung hoạt động của hệ thống (quản lý tin tức, quản lý góp ý, quản lý tài khoản người dùng) không ảnh hưởng đến cơ cấu tổ chức và các hệ thống khác của quản lý.

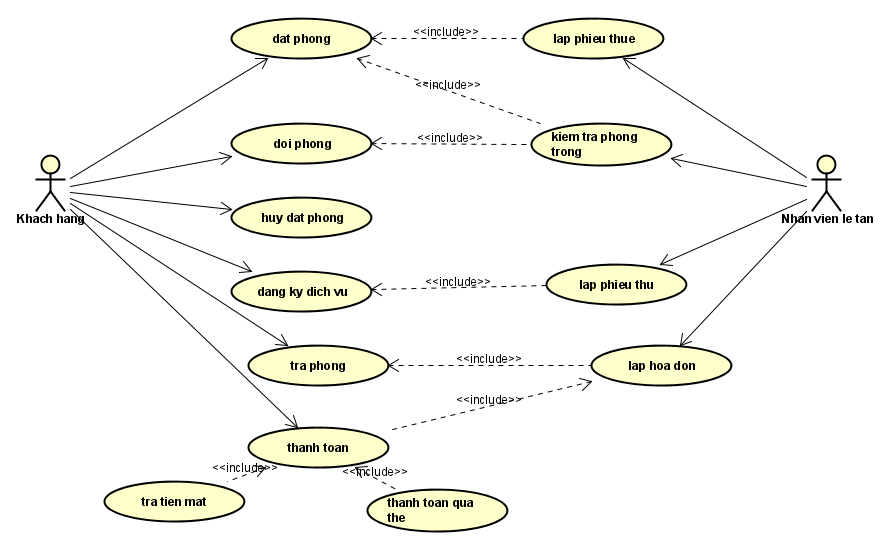
## **3.2 Các ràng buộc, yêu cầu sau khi hoàn thành**

Phần mềm sau khi triển khai phải đáp ứng được nhu cầu tự động 50% số lượng công việc liên quan.

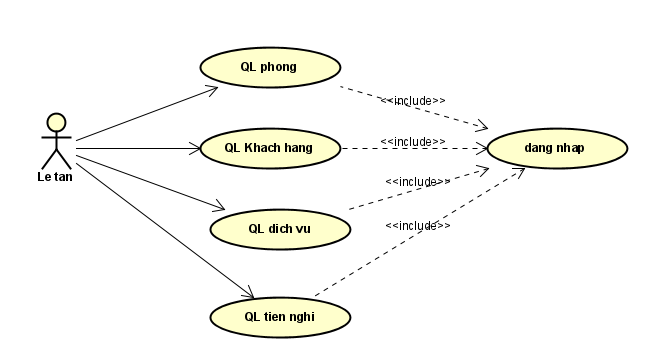
Dữ liệu phải đúng với thực tế và phải cập nhật thường xuyên.

## **3.3 Các sơ đồ phân tích thiết kế hệ thống**

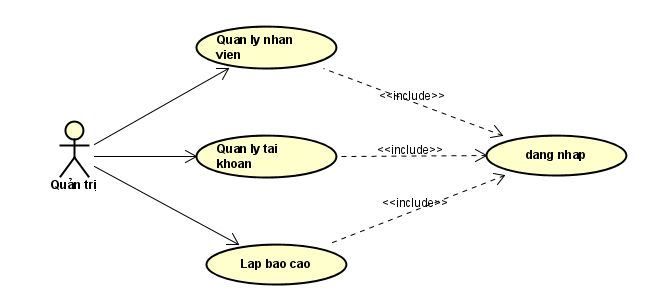
### 3.3.1 Usecase diagram



Hình 3.3.1a Usecase khách hàng đặt phòng

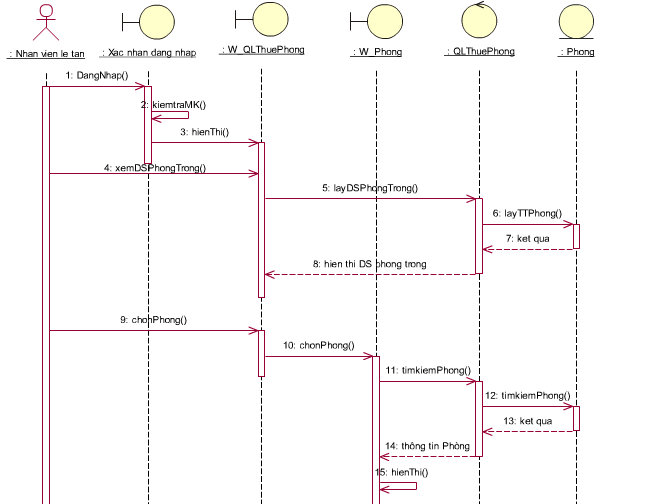


Hình 3.3.1b Usecase Lễ tân

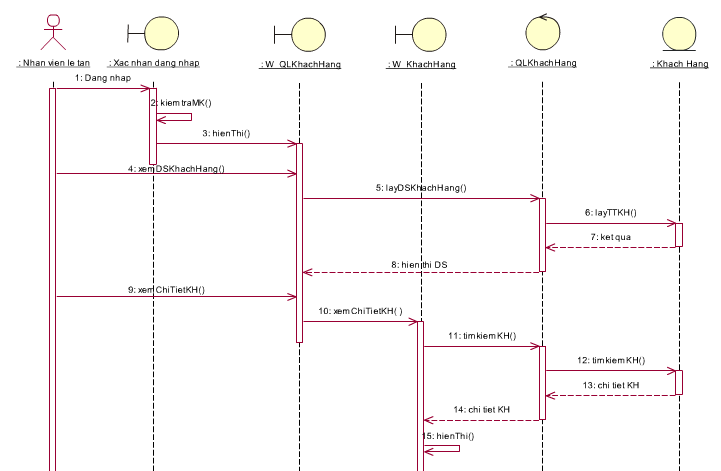


Hình 3.3.1c. Usecase Quản trị

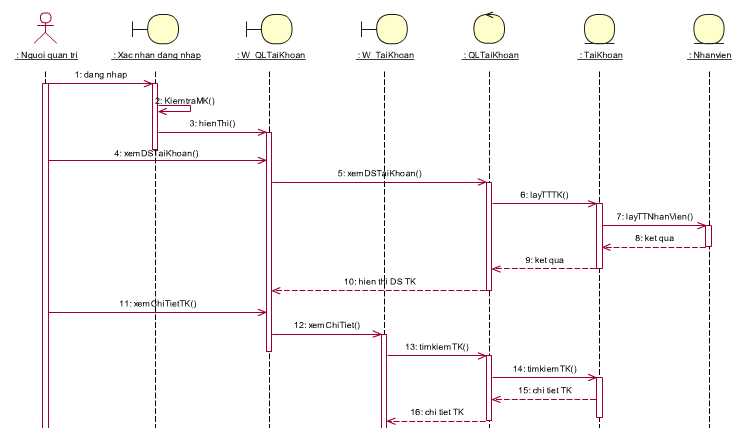
* + 1. *Sequence diagram*



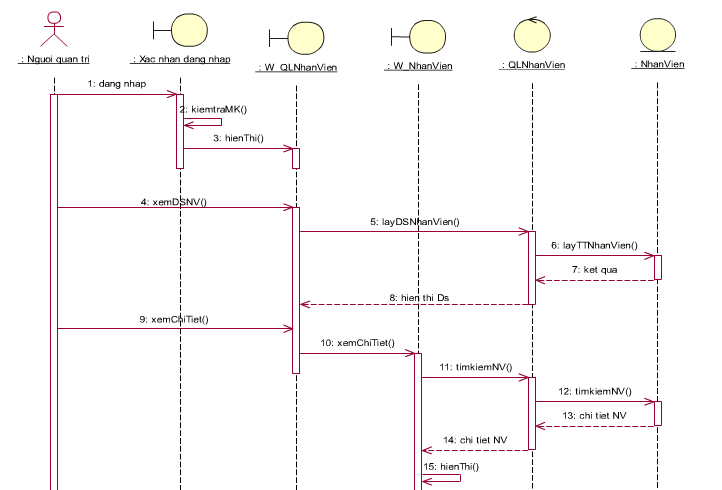
Hình 3.3.2a Sequence QL thuê phòng



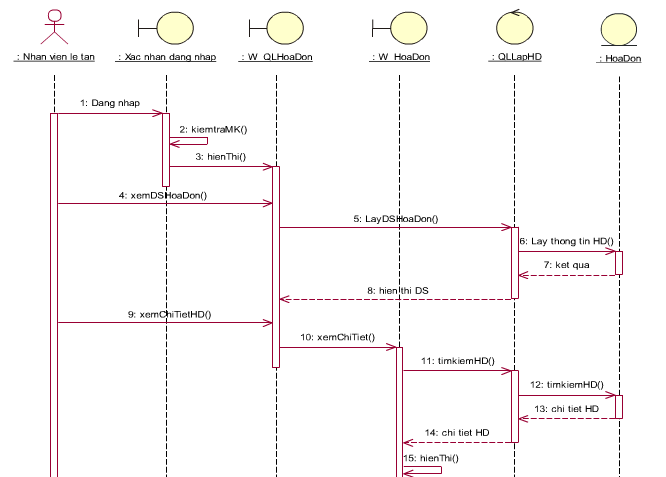
Hình 3.3.2b Sequence QL khách hàng



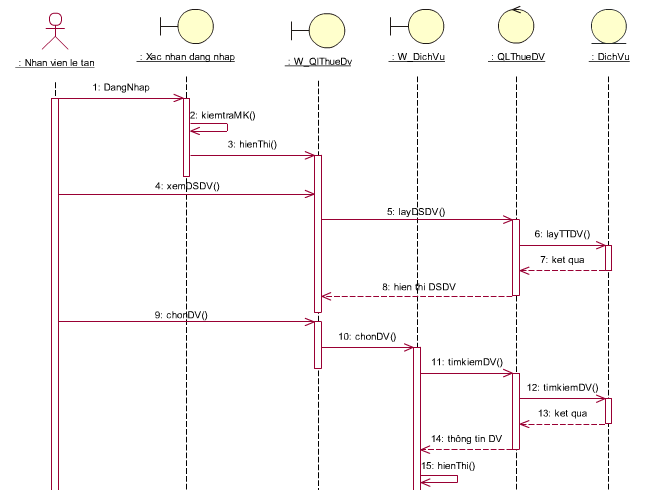
Hình 3.3.2c Sequence QL Tài khoản



Hình 3.3.2d Sequence QL Nhân Viên

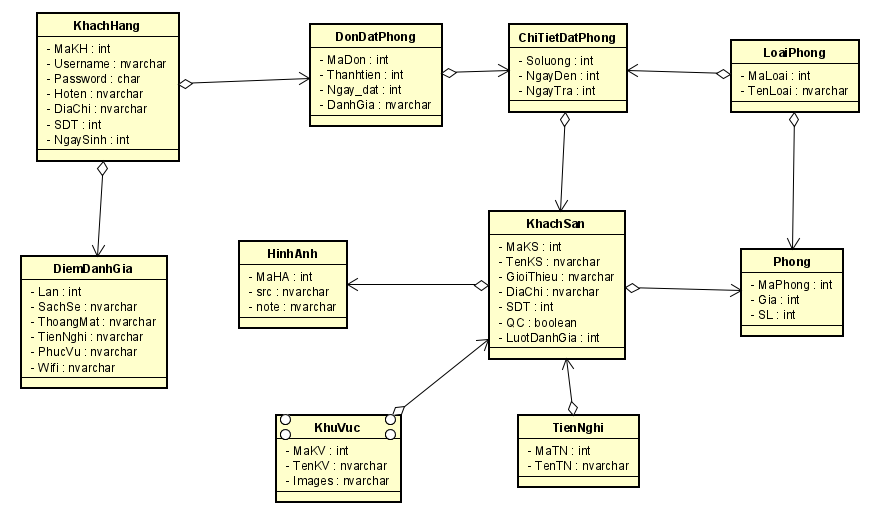


Hình 3.3.2e Sequence Lập hóa đơn



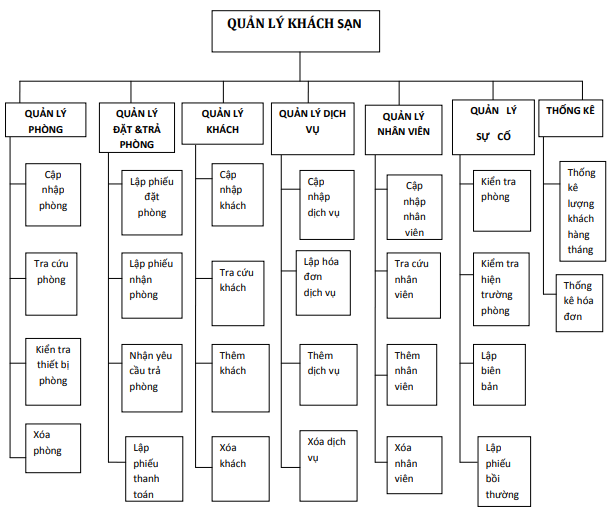
Hình 3.3.2f Sequence Thuê dịch vụ

## **3.4 Class diagram**



Hình 3.4 Biểu đồ lớp

## **3.5 Sơ đồ chức năng**



Hình 3.5 Sơ đồ chức năng hệ thống

## **3.6 Đặc tả use case**

### 3.6.1 Use case đăng nhập

|  |  |
| --- | --- |
| Use case | Đăng nhập |
| Actor | Lễ tân, nhân viên kinh doanh, nhân viên nhân sự, nhân viên kế toán, nhân viên dịch vụ. |
| Athor |  |
| Brief Description | Use case này mô tả các bước đăng nhập của actor vào hệ thống. |
| Pre-conditions | Không có |
| Basic Flows | 1. Hệ thống yêu cầu actor nhập tên đăng nhập, mật khẩu.  2. Actor nhập tên đăng nhập, mật khẩu của mình và nhấn nút đăng nhập  3. Hệ thống kiểm tra thông tin đăng nhập 4. Hệ thống thông báo thành công và cho actor đăng nhập vào hệ thống, đồng thời phân quyền tùy theo loại nhân viên. |
| Alternative Flows | 5. Tên đăng nhập hoặc mật khẩu không đúng, hệ thống hiện thông báo cho người dung và yêu cầu đăng nhập lại. |
| Post-conditions | Cho phép actor đăng nhập vào hệ thống nếu đăng |
| Special Requirements | Không có |

### 

### 3.6.2 Use case đăng xuất

|  |  |
| --- | --- |
| Use case | Đăng xuất |
| Actor | Lễ tân, nhân viên kinh doanh, nhân viên nhân sự, nhân viên kế toán, nhân viên dịch vụ. |
| Athor |  |
| Brief Description | Use case này môt tả việc đăng xuất khỏi hệ thống |
| Pre-conditions | Actor phải đăng nhập thành công |
| Basic Flows | 1. Actor chọn chức năng đăng xuất khỏi hệ thống.  2. Actor nhập tên đăng nhập, mật khẩu của mình và nhấn nút đăng nhập  3. Hệ thống hiển thị yêu cầu xác nhận từ actor.  4. Actor dùng xác nhận đăng xuất.  5. Hệ thống đăng actor khỏi hệ thống. |
| Alternative Flows | 3.1. Actor không xác nhận đăng xuất thì hệ thống sẽ giữ nguyên hiện trạng. |
| Post-conditions | Cho phép actor đăng nhập vào hệ thống nếu đăng |
| Special Requirements | Không có |

### 

### 3.6.3 Use case Đặt phòng

|  |  |
| --- | --- |
| Use case | Đặt phòng |
| Actor | Lễ tân |
| Athor |  |
| Brief Description | Use case này cho phép bộ phận lễ tân tiếp nhận việc đặt phòng trước của khách hàng. |
| Pre-conditions | Bộ phận lễ tân phải đăng nhập vào hệ thống. |
| Basic Flows | Use case được thực hiện khi khi khách hàng muốn đặt phòng trước bằng cách gọi điện thoại cho bộ phận lễ tân hoặc đến trực tiếp đến khách sạn để đăng ký. 1. Bộ phận lễ tân chọn chức năng đặt phòng cho khách hàng.  2. Hệ thống hiện thị form yêu cầu nhập thông tin khách hàng và ngày nhận phòng. Thông tin khách hàng bao gồm: + Số CMND; + Họ tên; + Địa chỉ; + Số điện thoại.  3. Bộ phận lễ tân nhập thông tin khách hàng và ngày nhận phòng của khách hàng. 4. Hệ thống kiểm tra thông tin phòng của ngày mà khách hàng yêu cầu đặt phòng đồng thời lấy ra danh sách loại phòng và các phòng tương ứng mà khách hàng có thể thuê vào ngày đó.  5. Lễ tân chọn phòng theo yêu cầu của khách hàng đã đặt.  6. Lễ tân chọn nút “Đăng ký” để hoàn tất việc đăng ký phòng cho khách hàng.  7. Hệ thống kiểm tra dữ liệu lễ tân vừa nhập.  8. Hệ thống lưu lại thông tin đặt phòng của khách hàng.  9. Kết thúc Use case |
| Alternative Flows | 3.1. Loại phòng mà khách hàng yêu cầu đã hết phòng trống.  3.1.1. Hệ thống thông báo hết phòng với loại phòng đã chọn và yêu cầu chọn loại phòng khác.  3.1.2. Lễ tân thông báo cho khách hàng và yêu cầu khách hàng chọn loại phòng khác. Lễ tân sẽ tiếp tục tìm phòng với loại phòng khác mà khách hàng yêu cầu hoặc hủy phiếu đăng ký nếu khác hàng từ chối tiếp tục đặt phòng.  3.2. Hết phòng  3.2.1. Lễ tân thông báo hết phòng đến khách hàng và yêu cầu khách hàng chọn ngày khác hoặc hủy việc đăng ký nếu khách hàng từ chối tiếp tục đặt phòng.  7.1. Dữ liệu nhập không hợp lệ  7.1.1. Hệ thống thông báo và yêu cầu thực hiện lại.  8.1. Nếu thông tin khách hàng đã tồn tại trong hệ thống thì sẽ không lưu thông tin khách hàng lại nữa mà chỉ lưu thông tin đặt phòng. |
| Post-conditions | Lưu thông tin đăng ký đặt phòng vào hệ thống nếu use case thực hiện thành công. |
| Special Requirements | Không có |

### 3.6.4 Use case Tra cứu phòng

|  |  |
| --- | --- |
| Use case | Tra cứu phòng |
| Actor | Lễ tân |
| Athor |  |
| Brief Description | Use case này cho phép bộ phận lễ tân kiểm tra phòng nào đó có trống hay không tại một thời điểm cụ thể để thực hiện đặt phòng hoặc thuê phòng cho khách hàng. |
| Pre-conditions | Bộ phận lễ tân phải đăng nhập vào hệ thống. |
| Basic Flows | Use case thực hiện khi lễ tân chọn chức năng “Tra cứu phòng”  3. Hệ thống sẽ kích hoạt Use case “Kiểm tra tình trạng phòng” với thông tin cung cấp là mã phòng để xem tình trạng phòng của phòng mà lễ tân cung cấp (đang ở, đã được đặt trước hay còn trống).  4. Kết thúc Use case |
| Alternative Flows | Không có |
| Post-conditions | Lưu thông tin đăng ký đặt phòng vào hệ thống nếu use case thực hiện thành công. |
| Special Requirements | Không có |

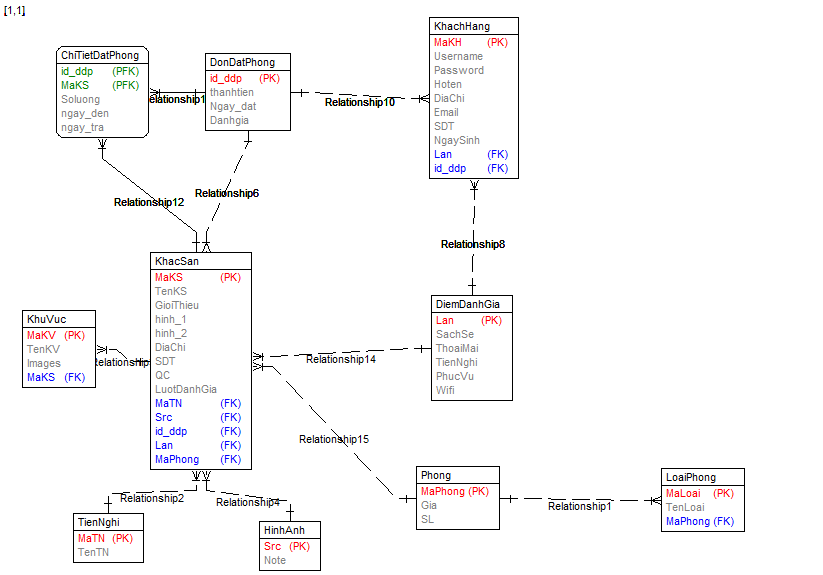
### 3.6.5 Use case tìm kiếm dịch vụ

|  |  |
| --- | --- |
| Use case | Tìm kiếm dịch vụ |
| Actor | Nhân viên dịch vụ |
| Brief Description | Use case này cho phép nhân viên dịch vụ tìm kiếm thông tin dịch vụ. |
| Pre-conditions | Nhân viên dịch vụ phải đăng nhập thành công vào hệ thống. |
| Basic Flows | 1. Nhân viên dịch vụ chọn nút tìm kiếm dịch vụ.  2. Hệ thống hiển thị form yêu cầu nhập mã dịch vụ.  3. Nhân viên dịch vụ nhập thông tin vào form và nhấn nút tìm.  4.Hệ thống hiển thị chi tiết thông tin dịch vụ |
| Alternative Flows | Không có |
| Post-conditions | Trả về thông tin dịch vụ cần tìm |
| Special Requirements | Không có |

### 3.6.6 Use case lập hóa đơn

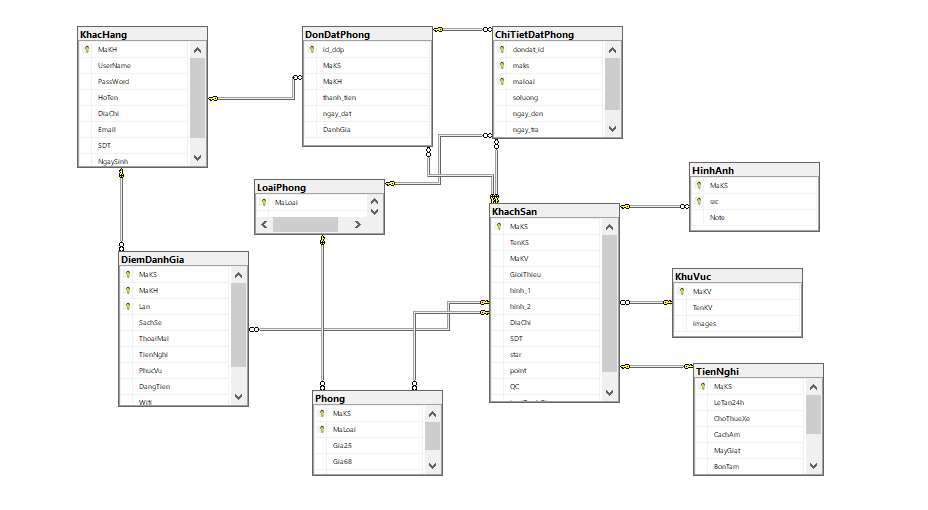
|  |  |
| --- | --- |
| Use case | Lập hóa đơn |
| Actor | Lễ tân |
| Author |  |
| Brief Description | Use case này cho phép bộ phận lễ lập hóa đơn tính tiền khi khách hàng trả phòng |
| Pre-conditions | Bộ phận lễ tân phải đăng nhập vào hệ thống. |
| Basic Flows | Use case được thực hiện khi khách hàng có yêu cầu trả phòng  1. Lễ tân sẽ ghi lại thông tin về mã phòng, CMND của khách hàng.  2. Dựa vào số CMND của khách hàng, hệ thống sẽ tìm ra các thông tin ở và sử dụng dịch vụ khách sạn của khách hàng để tiến hành tính ra số tiền mà khách hàng phải trả.  3. Lễ tân click “Thanh toán” để tiến hành thanh toán tiền cho khách hàng. 4. Hệ thống tự động in ra hóa đơn cho khách hàng  5. Hệ thống lưu thông tin hóa đơn, chi tiết hóa đơn xuống CSDL. 6. Kết thúc Use case. |
| Alternative Flows | Không có |
| Post-conditions | Thêm thông tin hóa đơn, chi tiết hóa đơn xuống CSDL |
| Special Requirements | Không có |

## **3.7 Mô hình thực thể ERD và Diagram của hệ thống**



Hình 3.7.1a Mô hình ERD

* Diagram của hệ thống:



Hình 3.7.1b Diagram của hệ thống

## **3.8 Bảng cơ sở dữ liệu chi tiết thuộc tính**

* Bảng khách sạn

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| MaKS | smallint |  | Khóa chính |
| TenKS | Nvarchar | 50 |  |
| GioiThieu | Nvarchar | 500 |  |
| hinh\_1 | Nvarchar | 50 |  |
| hinh\_2 | Nvarchar | 50 |  |
| DiaChi | Nvarchar | 50 |  |
| SDT | varchar | 10 |  |
| star | smallint |  |  |
| point | decimal(2, 1) |  |  |
| QC | bit |  |  |
| LuotDanhGia | int |  |  |

* Bảng khách hàng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| MaKH | smallint |  | Khóa chính |
| UserName | Nvarchar | 50 |  |
| PassWord | Nvarchar | 50 |  |
| HoTen | Nvarchar | 50 |  |
| DiaChi | Nvarchar | 50 |  |
| SDT | Varchar | 11 |  |
| NgaySinh | date |  |  |
| Email | Nvarchar | 50 |  |

* Bảng phòng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| MaKS | smallint |  | Khóa chính |
| MaLoai | tinyint |  | Khóa chính |
| Gia25 | decimal(18, 0) |  |  |
| Gia68 | decimal(18, 0) |  |  |
| SL | smallint |  |  |

* Bảng loại phòng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| MaLoai | tinyint |  | Khóa chính |
| TenLoai | Nvarchar | 50 |  |

* Bảng khu vực

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| MaKV | int |  | Khóa chính |
| TenKV | Nvarchar | 50 |  |
| images | Nvarchar | 30 |  |

* Bảng tiện nghi

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| MaKS | smallint |  | Khóa chính |
| LeTan24h | bit |  |  |
| ChoThueXe | bit |  |  |
| CachAm | bit |  |  |
| MayGiat | bit |  |  |
| BonTam | bit |  |  |
| Tivi | bit |  |  |

* Bảng hình ảnh

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| MaKS | smallint |  | Khóa chính |
| src | Nvarchar | 30 | Khóa chính |
| Note | Nvarchar | 50 |  |

* Bảng điểm đánh giá

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| MaKS | smallint |  | Khóa chính |
| MaKH | smallint |  | Khóa chính |
| Lan | smallint |  | Khóa chính |
| SachSe | smallint |  |  |
| ThoaiMai | smallint |  |  |
| TienNghi | smallint |  |  |
| PhucVu | smallint |  |  |
| DangTien | smallint |  |  |
| Wifi | smallint |  |  |
| DiaDiem | smallint |  |  |

* Bảng đơn đặt phòng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| id\_ddp | int |  | Khóa chính |
| MaKS | smallint |  |  |
| MaKH | smallint |  |  |
| thanh\_tien | decimal(15, 0) |  |  |
| ngay\_dat | datetime |  |  |
| DanhGia | bit |  |  |

* Bảng chi tiết đặt phòng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| dondat\_id | int |  | Khóa chính |
| maks | smallint |  | Khóa chính |
| maloai | tinyint |  | Khóa chính |
| soluong | int |  |  |
| ngay\_den | date |  |  |
| ngay\_tra | date |  |  |

* Bảng tài khoản

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Kiểu | Độ dài | Ghi chú |
| MaTK | int |  | Khóa chính |
| TenTK | char | 50 |  |
| MatKhau | char | 50 |  |
| VaiTro | char | 10 |  |

**3.9 Mô tả chức năng hệ thống**

*3.9.1 Chức năng người dùng:*

* Chức năng đăng ký, đăng nhập: Chức năng tạo tài khoản cho người dùng, đăng nhập vào hệ thống để sử dụng và đặt phòng, đặt tour trên tài khoản của mình.
* Chức năng đặt phòng: Người dùng lựa chọn phòng hoặc tour theo giá, khu vực, số lượng người ở, số lượng phòng, giường đơn đôi, ngày đến và ngày trả theo nhu cầu của người dùng.
* Chức năng thanh toán: Sau khi khách trả phòng liên hệ trực tiếp lễ tân, hoặc thanh toán online paypal qua hóa đơn của mình.
* Chức năng đánh giá: Chức năng cho phép người dùng thông qua website đánh giá chất lượng phòng ở, trang thiết bị, độ hài lòng của người dùng theo đánh giá sao.

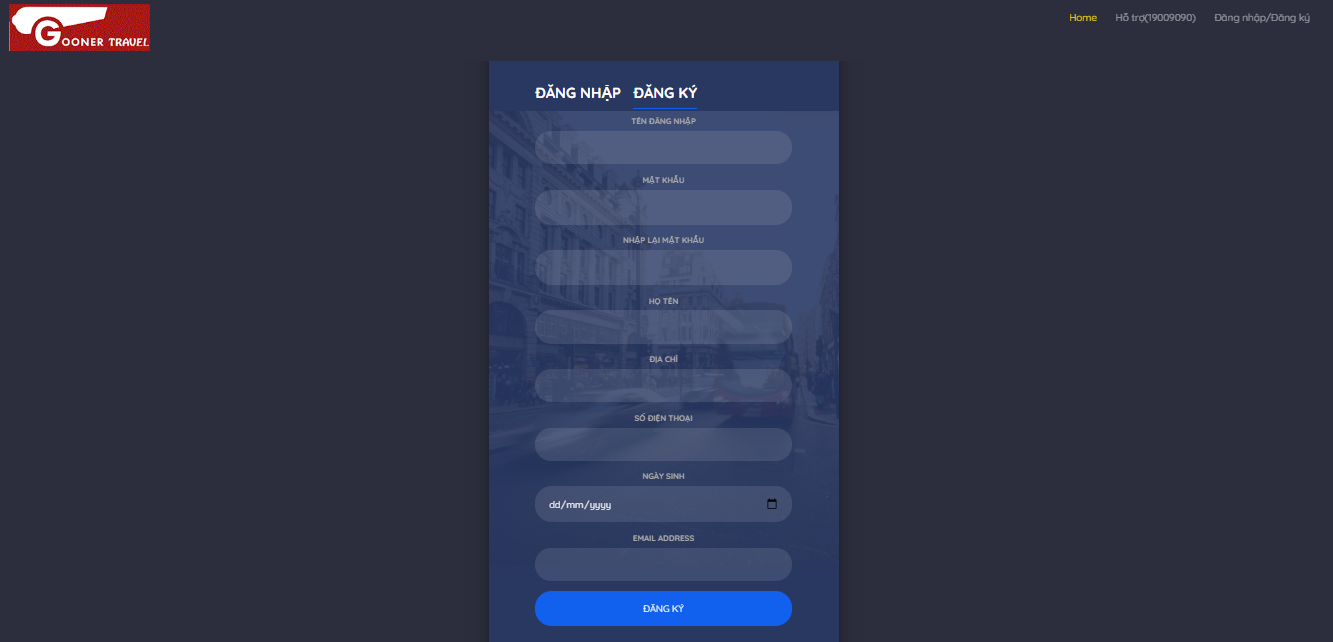
### 3.9.2 Chức năng của admin, quản trị:

* Chức năng quản lý đặt phòng: Quản trị viên được phép quản lý thêm, xóa, sửa, cập nhật phòng tour.
* Chức năng quản lý thanh toán: Quản trị viên quản lý những khách sạn phòng, tour đã thanh toán, ngày giờ cụ thể, giá phòng.
* Chức năng quản lý người dùng: Quản trị viên sẽ quản lý thông tin của người dùng sau khi đã đăng ký tài khoản của hệ thống.

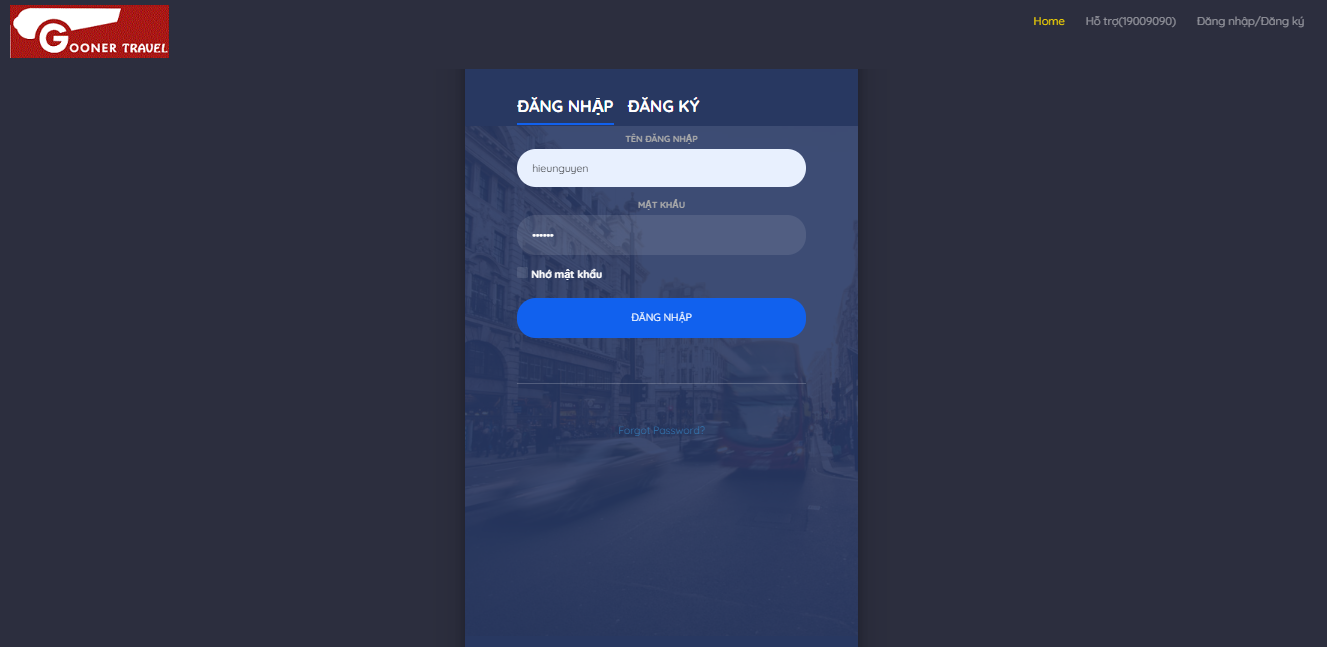
# **CHƯƠNG IV KẾT QUẢ THỰC NGHIỆM**

## **4.1 Giao diện của người dùng**

### 4.1.1 Giao diện đăng ký, đăng nhập:

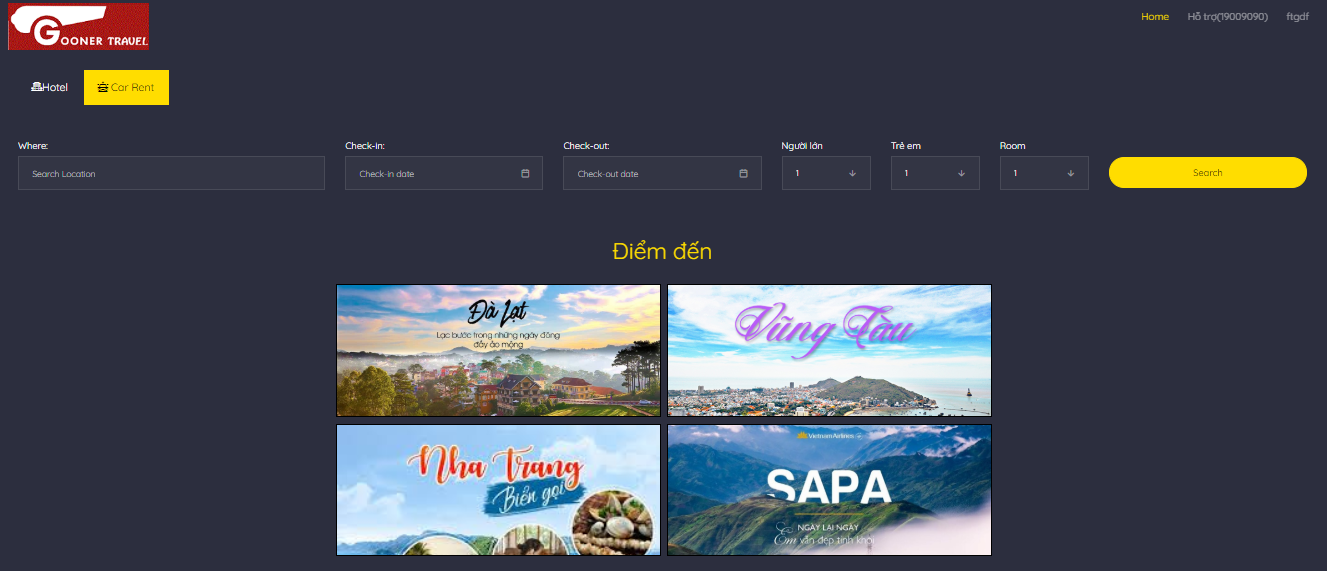


Hình 4.1.1a Giao diện đăng ký



Hình 4.1.1b Giao diện đăng nhập

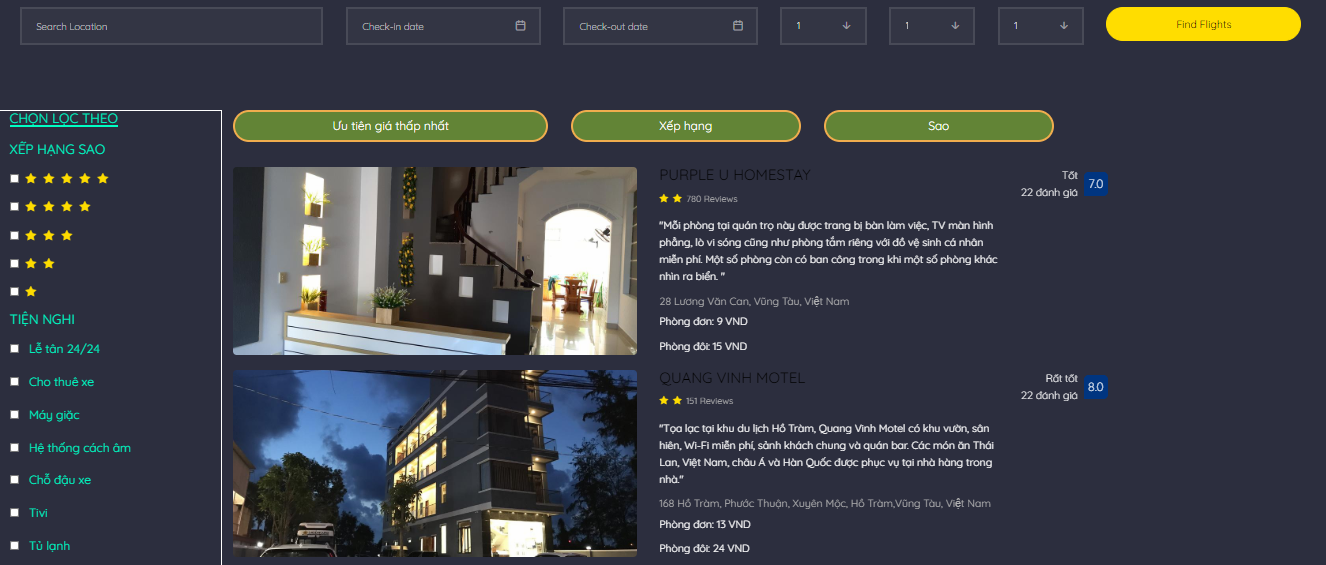
### 4.1.2 Giao diện trang chủ:





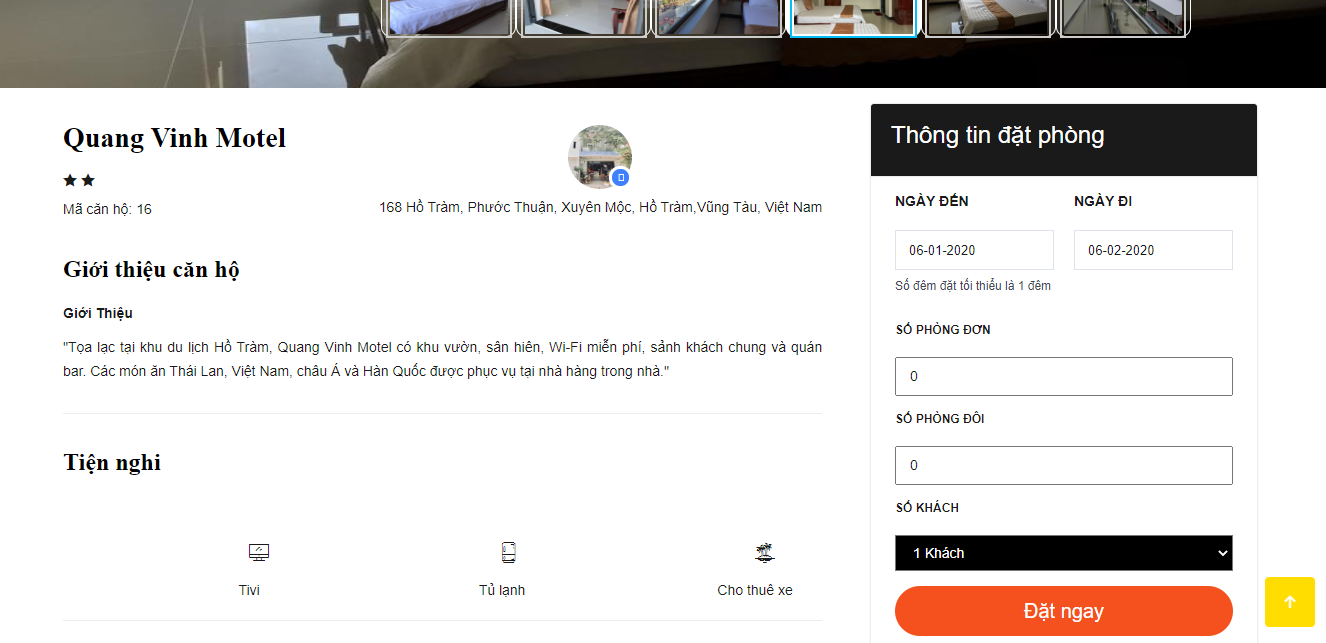
Hình 4.1.2 Giao diện trang chủ

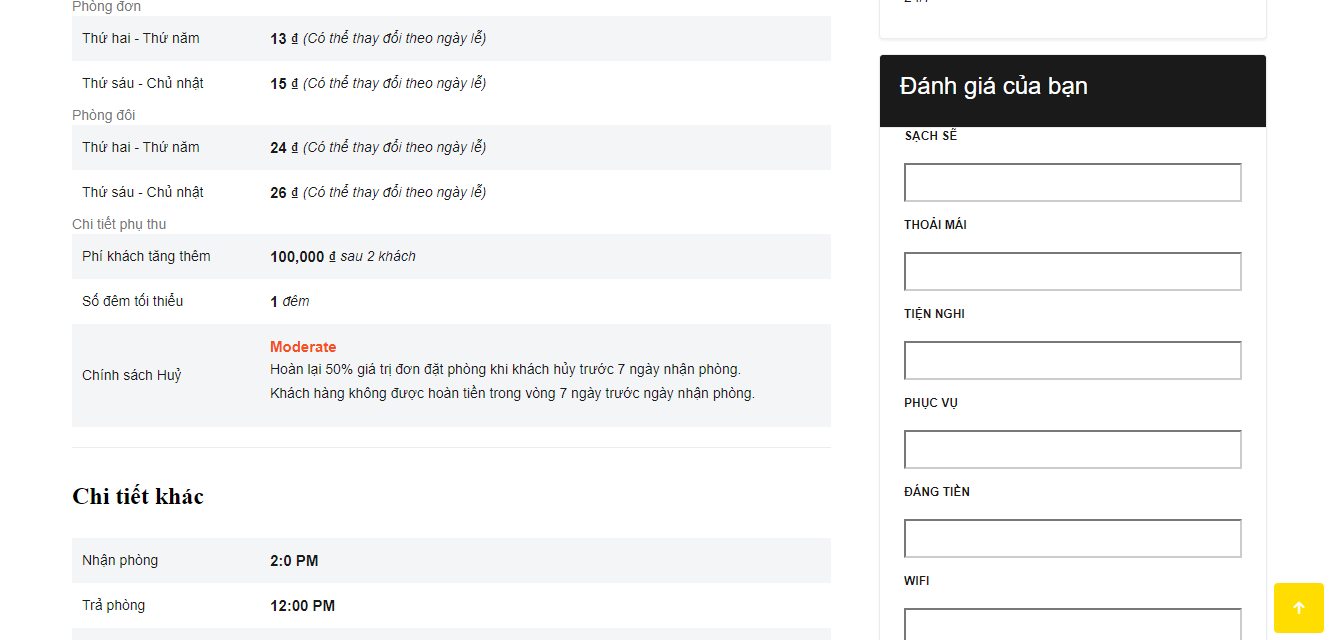
### 4.1.3 Giao diện danh mục phòng, khách sạn:



Hình 4.1.3 Giao diện phòng, khách sạn

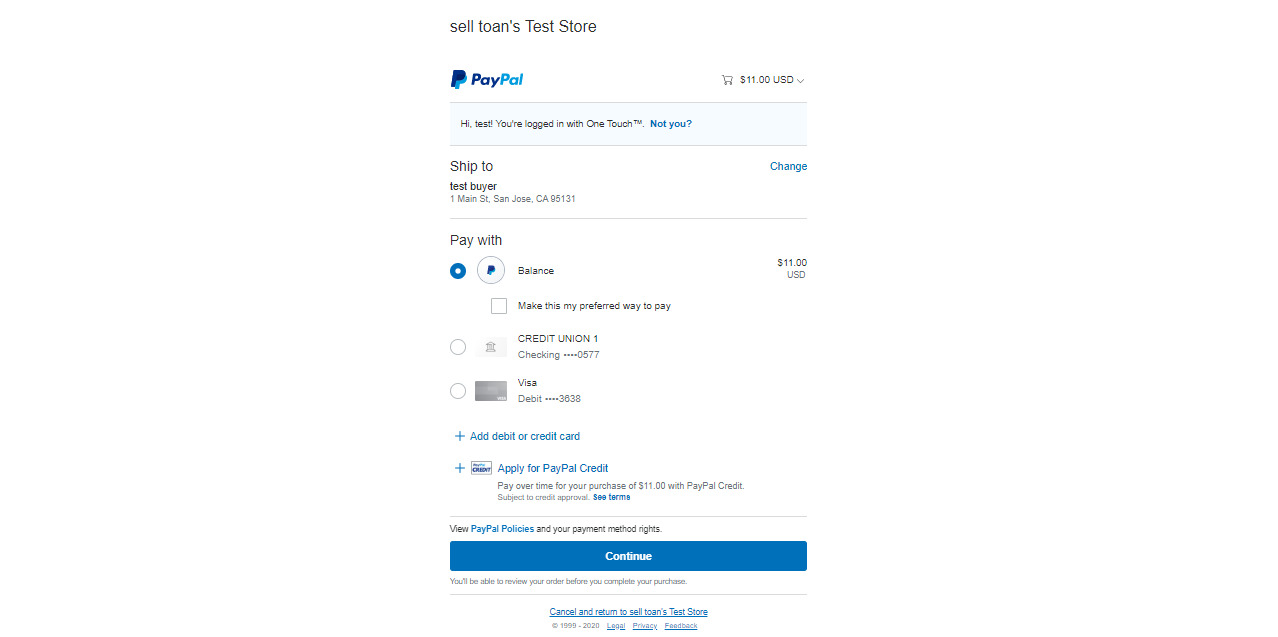
### 4.1.4 Giao diện đặt phòng, đánh giá:





Hình 4.1.4 Giao diện đặt phòng, đánh giá

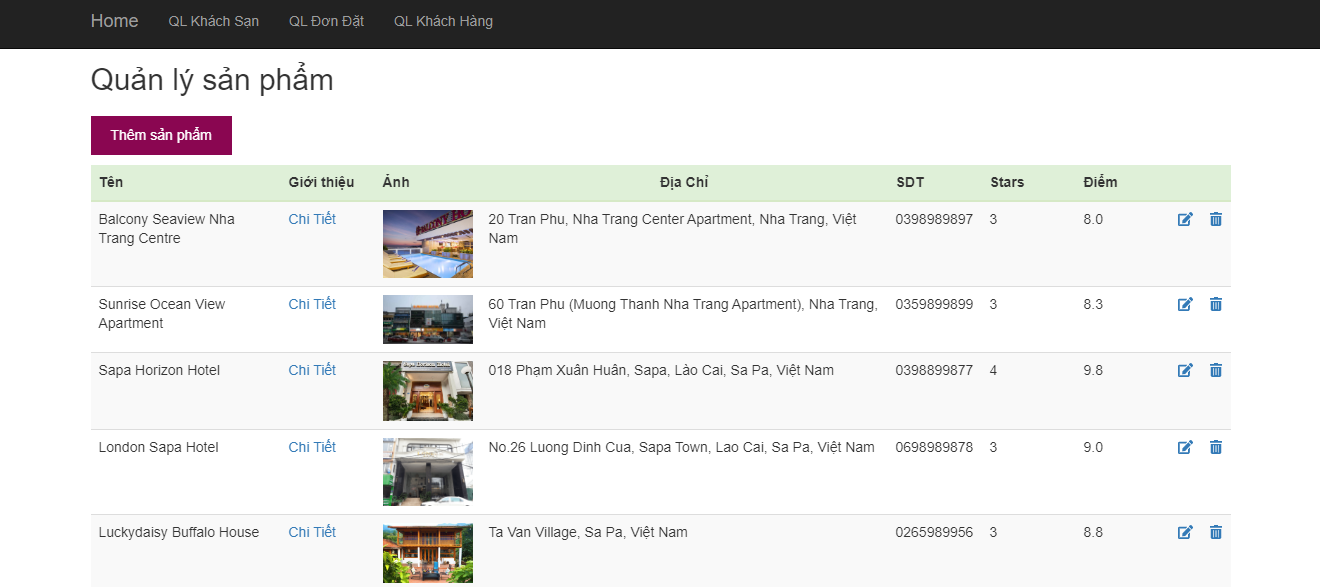
### 4.1.5 Giao diện thanh toán:

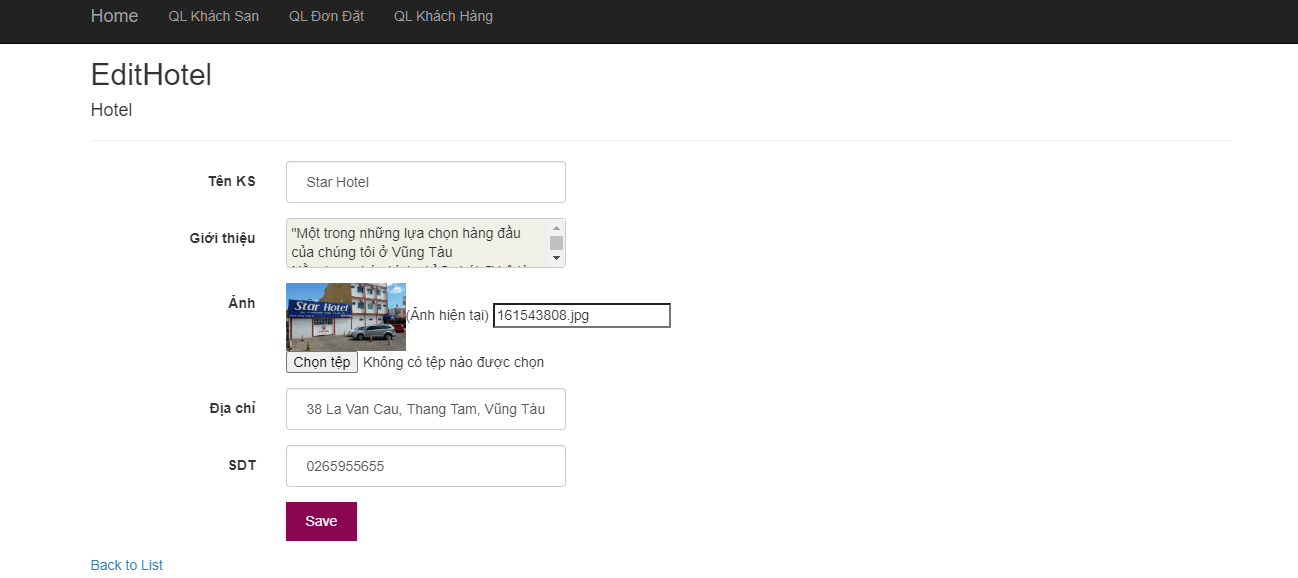


Hình 4.1.5 Giao diện thanh toán

## **4.2 Giao diện chức năng của admin, quản trị**

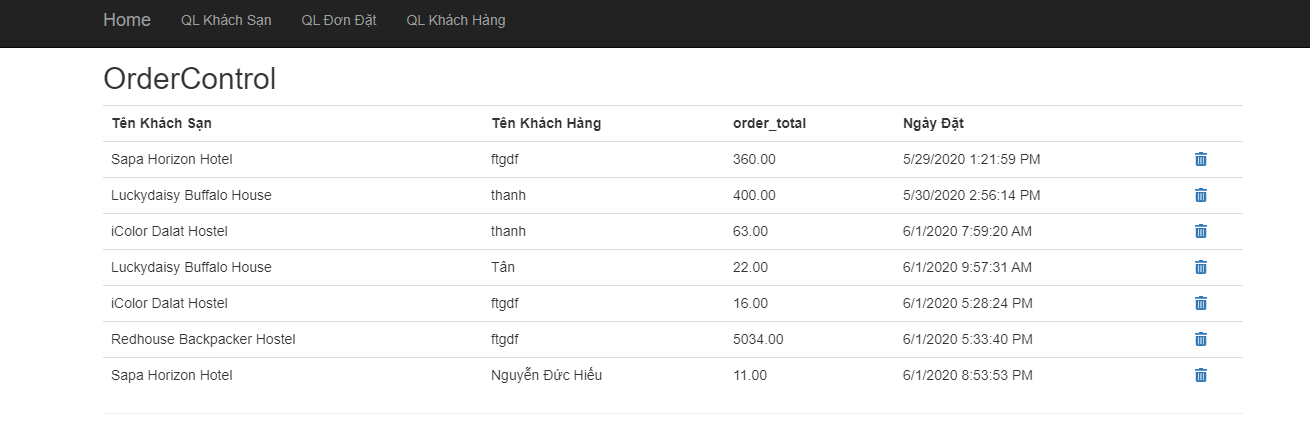
### 4.2.1 Giao diện quản lý đặt phòng:





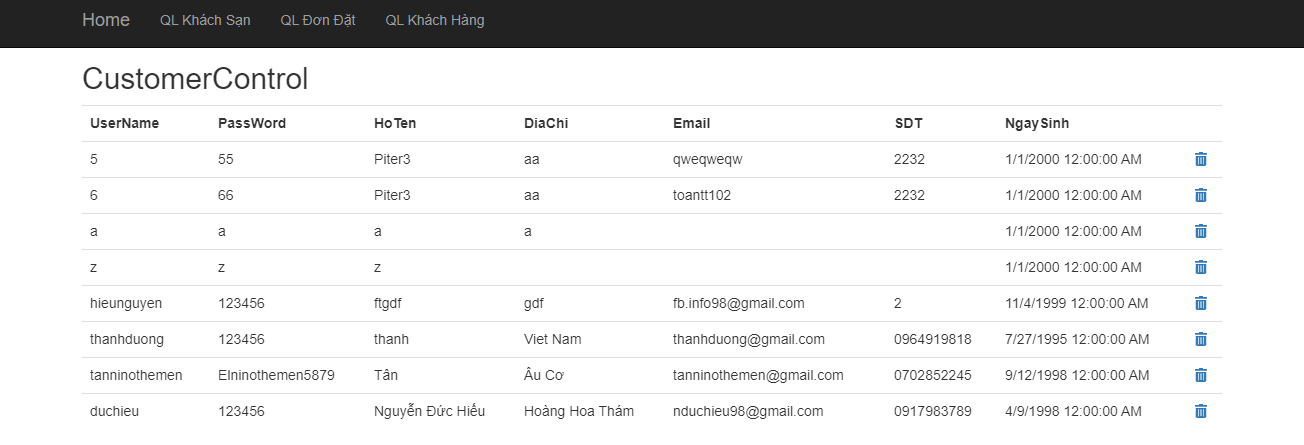
Hình 4.2.1 Giao diện admin quản lý phòng

### 4.2.2 Giao diện quản lý đơn đặt phòng:



Hình 4.2.2 Giao diện admin quản lý đơn đặt phòng

### 4.2.3 Giao diện quản lý khách hàng:



Hình 4.2.3 Giao diện admin quản lý khách hàng

# **CHƯƠNG V. KẾT LUẬN VÀ KIẾN NGHỊ**

**5.1 Những vấn đề đã giải quyết**

Xây dựng thành công website quản lý đồ án với các chức năng cơ bản mà một website cần có.

Có phần giao diện dành cho quản trị viên.

Các cách tìm kiếm, tra cứu theo các chủ đề khác nhau, kết hợp nhiều chủ đề theo ý khách hàng.

Xem chi tiết thông tin sản phẩm thương mại và đầy đủ các thông tin về Website.

Cho phép khách hàng đăng ký thông tin để thực hiện việc mua dịch vụ. Kiểm tra tính hợp lệ của khách hàng đăng nhập hệ thống.

Tạo đơn đặt hàng.

Có thể chạy trên nhiều hệ điều hành.

Xem thông tin của các đề mục cần quản lý như: Khách hàng, sản phẩm, đơn đặt hàng…

Đưa ra các form để nhập dữ liệu mới của các loại tư liệu. Có thể sửa đổi, cập nhật các dữ liệu trên.

Trong việc thêm mới sản phẩm chương trình tự động sinh mã sản phẩm, điều này tiện lợi cho người quản trị và dữ liệu luôn đồng nhất.

Các chức năng này được thực hiện thông qua giao diện web. Có thể chạy trên nhiều hệ điều hành.

## **5.2 Những vấn đề chưa giải quyết và thiếu sót**

Tính năng phần mềm chưa thực sự nhiều, một số tính năng còn gặp lỗi.

Chức năng phần quyền chưa thật sự sâu, còn nhiều hạn chế.

Tính bảo mật chưa thực sự tốt, một số thành phần chương trình có thời gian xử lý lâu do thuật toán sử dụng chưa tối ưu.

## **5.3 Hướng phát triển mở rộng chương trình**

Vì website này có tính ứng dụng cao trong thực tế, giúp em có thêm hiểu biết, kinh nghiệm nên em sẽ phát triển, hoàn thiện hơn sản phầm này:

Xử lý nguồn dữ liệu lớn, khi nhiều người truy cập.

Bảo mật tốt hơn, phân quyền tốt hơn.

Nâng cấp, phát triển và dựa vào ý tưởng này để phát triển trên các nền tảng khác (ví dụ: ứng dụng quản lý đồ án trên các thiết bị di động…).

Quản lý kho (kiểm tra lượng sản phẩm tồn kho tự động), thanh toán điện tử có sử dụng Edit card.

Bổ sung thêm một số chức năng kiểm tra dữ liệu nhập, thay đổi mật mã truy nhập của khách hàng cũng như nhà quản trị.

Qua đồ án tổng hợp này, từ những kiến thức, kỹ năng được thầy hỗ trợ, hướng dẫn, cộng với việc tự tìm tòi, nghiên cứu các vấn đề phát sinh trong quá trình xây dựng website, em tự tin rằng mình có thể phát triển khả năng hơn và hoàn toàn tạo ra được các ứng dụng web đáp ứng được các nhu cầu thực tế.

Mặc dù đã cố gắng hoàn chỉnh các yêu cầu, nhưng bài báo cáo còn rất nhiều thiếu sót mong nhận được sự chỉ bảo hướng dẫn của các thầy cô khác giúp đỡ xem xét, đề xuất thêm các ý kiến cũng như bổ sung các vấn đề phục vụ cho việc xây dựng Website để em có thể hoàn chỉnh hơn. Chúng em xin cảm ơn các quý thầy cô.

# **TÀI LIỆU THAM KHẢO**

* Tài liệu giao diện mẫu:

[1] <https://travel.com/vi/>

[2] <https://www.agoda.com/vi-vn/>

* Tài liệu thông tin:

[3] <https://vi.wikipedia.org/wiki/ASP.NET>

[4] <https://vi.wikipedia.org/wiki/Microsoft_SQL_Server>

[5]<https://monamedia.co/mvc-la-gi-ung-dung-cua-mo-hinh-mvc-trong-lap-trinh/>

[6] <https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model>

* Tài liệu tham khảo code, hướng dẫn:

[7] <https://www.w3schools.com/>

[8] <https://khoapham.vn/>

[9]<https://www.youtube.com/watch?v=M0jdFS4ZyEk&list=PLRhlTlpDUWsyK1TIsewrQ7WwC7QkCSCPD>

[10] <https://www.youtube.com/watch?v=_73okoRv30Y>