**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**TRƯỜNG CÔNG NGHỆ TT & TRUYỀN THÔNG**

**BỘ MÔN HỆ THỐNG THÔNG TIN**



**TIỂU LUẬN TỐT NGHIỆP NGÀNH**

**HỆ THỐNG THÔNG TIN**

**ĐỀ TÀI**

XÂY DỰNG VÀ PHÁT TRIỂN PHẦM MỀM

QUẢN LÝ KHÁCH SẠN VỪA VÀ NHỎ

|  |  |
| --- | --- |
| **Người hướng dẫn**  TS. Trương Quốc Định | **Sinh viên thực hiện**  Phan Nguyên Tâm  B1704692, Khóa: 43 |

**Cần Thơ 5/2023**

**LỜI CẢM ƠN**

Lời đầu tiên, em xin gửi lời cảm ơn chân thành đến TS.Trương Quốc Định. Trong quá trình học tập và tìm hiểu Tiểu luận , em đã nhận được sự quan tâm giúp đỡ, hướng dẫn tận tình, chi tiết của Thầy.

Do chưa có nhiều kinh nghiệm làm đề tài cũng như một số vấn đề phát sinh trong quá trình thực hiện nên bài tiểu luận này chắc chắn sẽ không thoát khỏi những thiếu sót. Rất mong nhận được sự nhận xét, ý kiến đóng góp, phê bình từ phía Thầy .

Lời cuối cùng, em xin kính chúc Thầy có nhiều sức khỏe, thành công và hạnh phúc.

Em xin chân thành cảm ơn!

**NHẬN XÉT**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_nb\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**MỤC LỤC**

**[DANH MỤC BẢNG 5](#_Toc22030)**

**[DANH MỤC HÌNH ẢNH 6](#_Toc26345)**

**[TÓM TẮT 7](#_Toc15711)**

**[CHƯƠNG I: TỔNG QUAN 8](#_Toc4056)**

**[1. ĐẶT VẤN ĐỀ 8](#_Toc1042)**

**[2. MỤC TIÊU 8](#_Toc23786)**

[2.1 Mục tiêu chuyên môn 8](#_Toc2176)

[2.2 Mục tiêu về sản phẩm 8](#_Toc8019)

**[3. PHẠM VI VÀ KẾT QUẢ CẦN ĐẠT ĐƯỢC 9](#_Toc6729)**

[3.1 Phạm vi 9](#_Toc1672)

[3.2 Kết quả cần đạt được 9](#_Toc6172)

**[4. KẾ HOẠCH THỰC HIỆN 9](#_Toc22365)**

**[CHƯƠNG II: CƠ SỞ LÝ THUYẾT 10](#_Toc6354)**

**[1. NGÔN NGỮ C# 10](#_Toc1337)**

[1.1 Đặt trưng của ngôn ngữ C# 11](#_Toc25187)

[1.2 Các phiên bản C# 12](#_Toc1462)

[1.3 Nền tảng .Net 13](#_Toc17787)

**[3. SQL SERVER 14](#_Toc14946)**

**[CHƯƠNG III: NỘI DUNG VÀ KẾT QUẢ THỰC HIỆN 15](#_Toc27320)**

**[1. ĐẶT TẢ VẤN ĐỀ 15](#_Toc27015)**

[1.1 Đặt tả yêu cầu 15](#_Toc18136)

[1.2 Sơ đồ USE CASE 16](#_Toc17947)

[1.3 Đặc tả ràng buộc toàn vẹn 21](#_Toc21858)

**[2. THIẾT KẾ LƯỢC ĐỒ CƠ SỞ DỮ LIỆU 42](#_Toc18504)**

[2.1 Đặt tả lược đồ cở sở dữ liệu quan hệ 42](#_Toc30098)

[2.2 Biểu diễn các ràng buộc toàn vẹn 43](#_Toc25395)

[2.3 Mô hình mức vật lý 43](#_Toc25477)

**[3. THIẾT KẾ THÀNH PHẦN XỬ LÝ 50](#_Toc18913)**

**[4. THIẾT KẾ GIAO DIỆN 53](#_Toc22530)**

**[CHƯƠNG IV: ĐÁNH GIÁ KIỂM THỬ 62](#_Toc27473)**

**DANH MỤC BẢNG**

**Error! No table of figures entries found.**

**DANH MỤC HÌNH ẢNH**

[HÌnh 1 . Use Case Lễ tân 16](#_Toc20056)

[HÌnh 2 . Use Case Admin 17](#_Toc31840)

[HÌnh 3 . Use Case admin chức năng quản lý hóa đơn 17](#_Toc31142)

[HÌnh 4 . Use Case Admin chức năng quản lý khách hàng 18](#_Toc10866)

[HÌnh 5 . Use Case Admn chức năng quản lý phòng 18](#_Toc31801)

[HÌnh 6 . Use Case Admin chức năng quản lý dịch vụ 19](#_Toc1559)

[HÌnh 7 . Use Case Admin chức năng quản lý nhân sự 20](#_Toc26150)

[HÌnh 8 . Giao diện đăng nhập 53](#_Toc2719)

[HÌnh 9 . Giao diện đổi mật khẩu 53](#_Toc19682)

[HÌnh 10 . Giao diện chính 54](#_Toc13784)

[HÌnh 11 . Giao diện đặt phòng trực tiếp 54](#_Toc15948)

[HÌnh 12 . Thông báo thành công 55](#_Toc13106)

[HÌnh 13 . Giao diện phiếu dịch vụ 55](#_Toc25838)

[HÌnh 14 . Form thanh toán 56](#_Toc16929)

[HÌnh 15 . Thông báo thanh toán thành công 56](#_Toc16768)

[HÌnh 16 . Giao diện đặt phòng trước 57](#_Toc23491)

[HÌnh 17 . Giao diện đặt phòng trước 57](#_Toc7598)

[HÌnh 18 . Giao diện quản lý nhân viên 58](#_Toc31283)

[HÌnh 19 . Giao diện quản lý dịch vụ 58](#_Toc22627)

[HÌnh 20 . GIao diện quản lý thiết bị 59](#_Toc26768)

[HÌnh 21 . Giao diện quản lý loại phòng 59](#_Toc27426)

[HÌnh 22 . Giao diện quản lý hóa đơn 60](#_Toc2998)

[HÌnh 23 . Giao diện quản lý đặt phòng 60](#_Toc6757)

[HÌnh 24 . Giao diện quản lý khách hàng 61](#_Toc3127)

[HÌnh 25 . Giao diện quản lý phòng 61](#_Toc25688)

**TÓM TẮT**

Hiện nay ngành du lịch đang là hướng phát triển của nhiều vùng trên cả nước và với sự phát triển đó và đáp ứng nhu cầu đi, nghỉ của du khách nhiều khách sạn vừa và nhỏ thi nhau xuất hiện và dần mở rộng quy mô. Tuy nhiên phần lớn các khách sạn đấy thường chọn việc quản lý thủ công việc tốn thời gian nhưng không đảm bảo sự chính xác.

Nhằm vấn đề đó Phần mềm quản lý khách sạn xuất hiện nhằm giải quyết các vấn đề quản lý trở nên dễ dàng, nhanh chóng và chính xác hơn. Và về cơ bản đây sẽ là sản phẩm đảm bảo các công việc cơ bản của quản lý khách sạn.

CHƯƠNG I: TỔNG QUAN

1. ĐẶT VẤN ĐỀ

Ngày nay với sự phát triển của công nghệ hiện đại nhu cầu của con người về vật chất và tinh thần được nâng cao thêm, kéo theo đó là sự tăng trưởng của các loại hình dịch vụ cũng được nâng cao. Thông qua Internet và các mạng xã hội mọi người dần tìm đến sự trải nghiệm và mới mẻ ở vùng du lịch thông qua các tour du lịch hay tự mình đến nên số lượng du khách ngày càng tăng dẫn đến hàng loạt khách sạn, nhà nghỉ vừa và nhỏ liên tục xuất hiện và dần mở rộng quy mô. Nhưng nếu quản lý khách sạn một cách thủ công thì sẽ không đảm bảo được sự chính xác, nhanh chóng và chất lượng dịch vụ. Trong thời đại công nghệ thông tin phát triển như hiện nay thì sự chuyển dời từ quản lý thủ công sang quản lý bằng công nghệ hiện đại là cần thiết dẫn đến các phần mềm quản lý khách sạn dần xuất hiện nhiều hơn.

So với quản lý thủ công, một phần mềm quản lý khách sạn sẽ giảm thao tác làm việc và tính toán giúp công việc trở nên nhanh chóng, chính xác hơn cũng như giúp nâng cao chất lượng dịch vụ tốt hơn cho khách hàng.

Qua vấn đề trên em xin trình bày đề tài “Xây dựng và phát triển phần mềm quản lý khách sạn”. Nhằm qua đó có thể nâng cao, hiểu biết thêm về các nghiệp vụ quản lý khách sạn nói riêng và các hệ thống quản lý khác nới chung.

1. MỤC TIÊU
2. Mục tiêu chuyên môn

* Phân tích thiết kế hệ thống quản lý khách sạn nhằm giúp cho việc: quản lý phòng, quản lý khách hàng, quản lý đơn đặt phòng,... được tốt hơn, đảm bảo chính xác và nhanh chóng.

1. Mục tiêu về sản phẩm

* Có chức năng quản lý.
* Giao diện trực quan.

1. PHẠM VI VÀ KẾT QUẢ CẦN ĐẠT ĐƯỢC
2. Phạm vi

* Tìm hiểu về quy trình hoạt động và quản lý của khách sạn.
* Tìm hiểu xây dựng hệ thống khách sạn.
* Xây dựng phần mềm quản lý khách sạn.
* Quản lý phòng, đặt phòng, loại phòng,...

1. Kết quả cần đạt được

* Tạo ra hệ thống có đầy đủ chức năng quản lý của khách sạn

1. KẾ HOẠCH THỰC HIỆN

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **CÔNG VIỆC** | **TUẦN** | | | | | | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** |
| 1 | Khảo sát hệ thống |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Phân tích yêu cầu |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Thiết kê mô hình |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Cài đặt chương trình |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Nhập liệu và kiểm thử |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Sửa lỗi chương trình |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Viết báo cáo |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

CHƯƠNG II: CƠ SỞ LÝ THUYẾT

1. NGÔN NGỮ C#

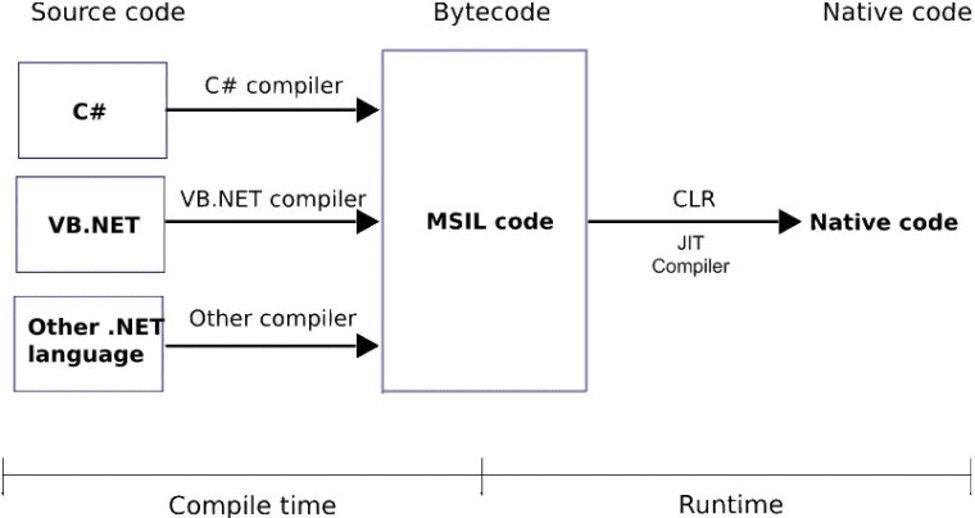


**C#** (**C Sharp**, đọc là *"xi-sáp"*) là một ngôn ngữ lập trình hướng đối tượng đa năng, mạnh mẽ được phát triển bởi Microsoft, C# là phần khởi đầu cho kế hoạch .NET của họ. Tên của ngôn ngữ bao gồm ký tự thăng theo Microsoft nhưng theo [ECMA](https://vi.wikipedia.org/w/index.php?title=ECMA&action=edit&redlink=1" \o "ECMA (trang không tồn tại)) là **C#**, chỉ bao gồm dấu số thường. Microsoft phát triển C# dựa trên C++ và Java. C# được miêu tả là ngôn ngữ có được sự cân bằng giữa C++, Visual Basic, Delphi và Java.

Trong các ứng dụng Windows truyền thống, mã nguồn chương trình được biên dịch trực tiếp thành các mã thực thi của hệ điều hành.

Trong các ứng dụng sử dụng .NET Framework, mã nguồn chương trình ((C#, VB.NET) được biên dịch thành mã ngôn ngữ trung gian MSIL (Microsoft intermediate language).

Sau đó mã nàu được biên dịch bởi Common Language Runtime (CLR) để trở thành mã thực thi của hệ điều hành.



1. Đặt trưng của ngôn ngữ C#

* C#là ngôn ngữ đơn giản

C# loại bỏ một vài sự phức tạp và rối rắm của những ngôn ngữ như Java và C++, bao gồm lạo bỏ những macro, những template, đa kế thừa, và lớp cơ sở ảo (virtual base class).

Ngôn ngữ C# đơn giản vì nó dưa trên nền tảng C và C++.

* C#là ngôn ngữ hiện đại

C# có những đặc tính như xử lý ngoại lệ, thu gom bộ nhớ tự động, những kiểu dữ liệu mở rộng, và bảo mật mã nguồn.

* C#là ngôn ngữ hướng đối tượng

Lập trình hướng đối tượng (OOP: Object-oriented programming) là phương pháp lập trình có 4 tính chất. Đó là tính trừu tượng (abstraction), tính đóng gói (encapsulation), tính đa hình (polymorphism) và tính kế thừa (inheritance). C# hỗ trợ tất cả những đặc tính trên.

* C#là ngôn ngữ ít từ khóa

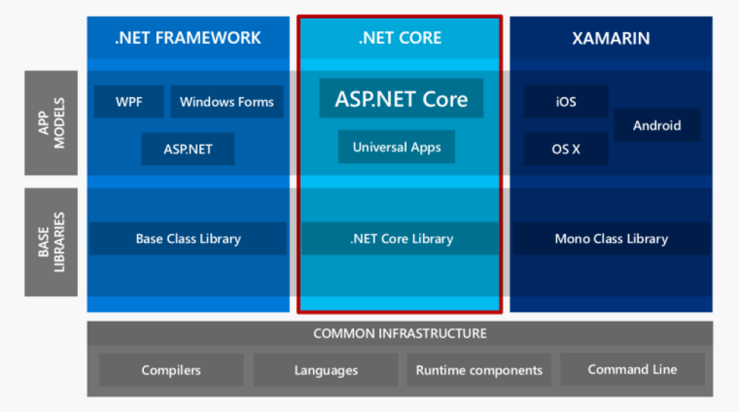
C# là ngôn ngữ sử dụng giới hạn những từ khóa. Phần lớn các từ khóa được sử dụng để mô tả thông tin.

1. Các phiên bản C#

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| VERSION | Language specification | | | DATE | .NET  Version | VISUAL  STUDIO |
| Ecma | ISO/IEC | Microsoft |
| VERSION  C# 1.0 | 12/  2002 | 04/2003 | 01/2002 | 01/  2002 | .NET  Framework  1.0 | Visual  Studio .NET  2002 |
| VERSION  C# 1.1  C# 1.2 | 10/2003 | 04/  2003 | .NET  Framework  1.1 | Visual  Studio .NET  2003 |
| VERSION  C# 2.0 | 06/  2006 | 09/2006 | 09/2005 | 11/  2005 | .NET  Framework  2.0  .NET  Framework  3.0 | Visual  Studio 2005  Visual  Studio 2008 |
| VERSION  C# 3.0 | None | | 08/2007 | 11/  2007 | .NET Framework 3.5 | Visual  Studio 2008 |
| VERSION  C# 4.0 | 04/2010 | 04/  2010 | .NET Framework 4.0 | Visual  Studio 2010 |
| VERSION  C# 5.0 | 12/  20117 | 12/2018 | 06/2013 | 08/  2012 | .NET Framework 4.5 | Visual  Studio 2012  Visual  Studio 2013 |
| VERSION  C# 6.0 | None | | Draft | 07/  2015 | .NET Framework 4.6  .NET Core 1.0  .NET Core 1.1 | Visual  Studio 2015 |
| VERSION  C# 7.0 | Specification proposal | 05/  2017 | .NET Framework 4.7 | Visual  Studio 2017  version 15.0 |
| VERSION  C# 7.1 | Specification proposal | 08/  2017 | .NET Core 2.0 | Visual  Studio 2017 version 15.3 |
| VERSION  C# 7.2 | Specification proposal | 11/  2017 |  | Visual  Studio 2017 version 15.5 |
| VERSION  C# 7.3 | Specification proposal | 05/  2018 | .NET Core 2.1  .NET Core 2.2  .NET Framework 4.8 | Visual  Studio 2017 version 15.7 |
| VERSION  C# 8 | Specification proposal | 09/  2019 | .NET Core 3.0 | Visual  Studio 2019 version 16.3 |

1. Nền tảng .Net

* .NET Framework được Microsoft đưa chính thức từ năm 2002. .NET Framework chỉ hoạt động trên Windows. Những nền tảng ứng dụng như WPF, Winforms, ASP .NET(1-4) hoạt động dựa trên .NET Framework.
* Mono là phiên bản cộng đồng nhằm mang .NET đến những nền tảng ngoài Windowns. Mono được phát triển chủ yếu nhằm xây dựng những ứng dụng với giao diện người dùng và được sử dụng rất rộng rãi: Unity Game, Xamarin...
* Cho đến năm 2013, Microsoft định hướng đi đa nền tảng và phát triển .NET core. .NET core hiện được sử dụng trong các ứng dụng Universal Windows platform và ASP .NET Cỏe. Từ đây, C# có thể được sử dụng để phát triển các loại ứng dụng đa nền tảng trên các hệ điều hành khác nhau (Windows, Linux, MacOS,...)



1. ADO .NET

****ADO.NET**** (ActiveX Data Object) là tập hợp các thư viện lớp qua đó cho phép ứng dụng tương tác (lấy về, cập nhật, xóa) với các nguồn dữ liệu (Như SQLServer, XML, MySQL, Oracle Database ...).

1. SQL SERVER

SQL server hay Microsoft SQL Server là phần mềm ứng dụng cho hệ thống quản trị cơ sở dữ liệu quan hệ(Relational Database Management System – RDBMS) được phát triển bởi Microsoft vào năm 1988. Nó được sử dụng để tạo, duy trì, quản lý và triển khai hệ thống RDBMS.

Phần mềm SQL Server được sử dụng khá rộng rãi vì nó được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn lên đến Tera – Byte cùng lúc phục vụ cho hàng ngàn user. Bên cạnh đó, ứng dụng này cung cấp đa dạng kiểu lập trình SQL từ ANSI SQL (SQL truyền thống) đến SQL và cả T-SQL (Transaction-SQL) được sử dụng cho cơ sở dữ liệu quan hệ nâng cao.

CHƯƠNG III: NỘI DUNG VÀ KẾT QUẢ THỰC HIỆN

1. ĐẶT TẢ VẤN ĐỀ
2. Đặt tả yêu cầu

Hệ thống quản lý khách sạn là hệ thống giúp các khách sạn quản lý một cách chuyên nghiệp, hiệu quả, cũng như kiểm soát chất lượng dịch vụ.

Phần mềm có 2 người dùng là quản lý và lễ tân. Lễ tân chỉ thực hiện các vai trò đặt phòng, lập hóa đơn. Trong khi đó, quản lý sẽ đóng vai trò chủ khách sạn bằng cách đăng nhập vào sẽ xuất hiện các danh mục quản lý: quản lý phòng, quản lý loại phòng, quản lý thiết bị, quản lý đặt phòng, quản lý dịch vụ, quản lý hóa đơn và quản lý nhân viên

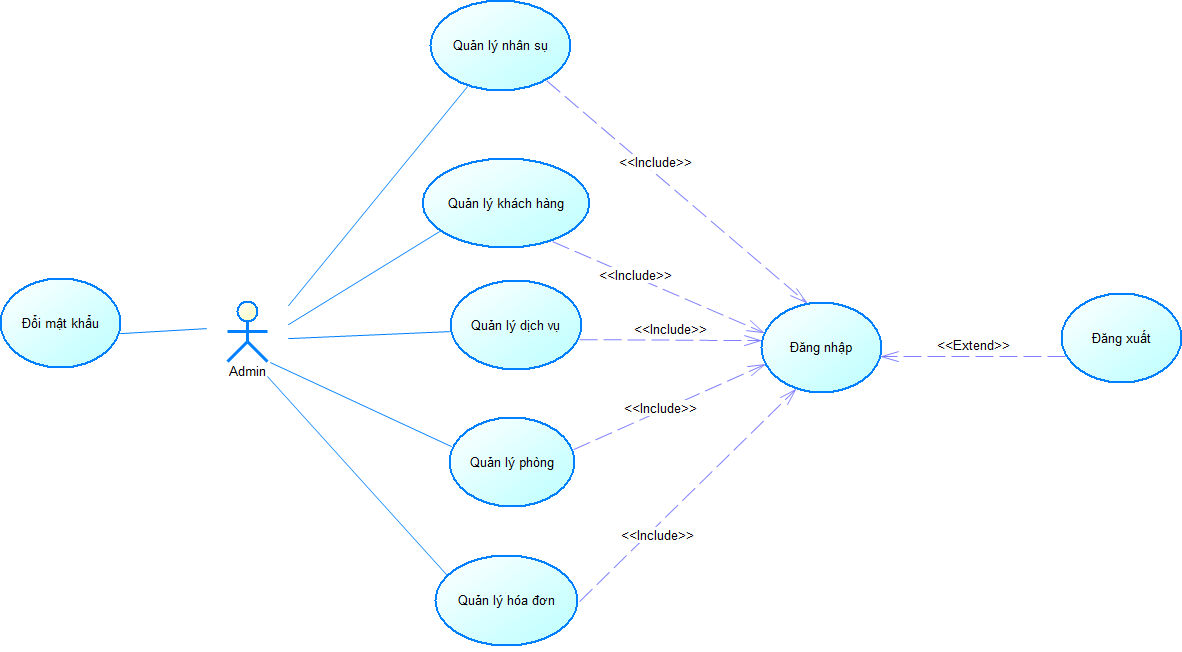
1. Sơ đồ USE CASE

Lễ tân:



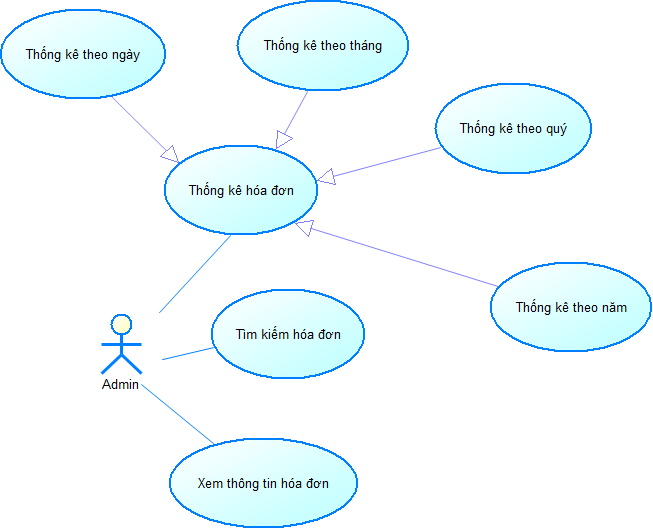
HÌnh 1. Use Case Lễ tân

Admin:



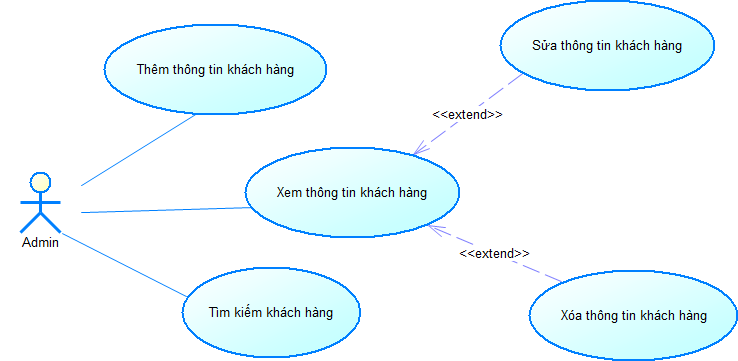
HÌnh 2. Use Case Admin

1. Chức năng quản lý hóa đơn



HÌnh 3. Use Case admin chức năng quản lý hóa đơn

1. Chức năng quản lý khách hàng



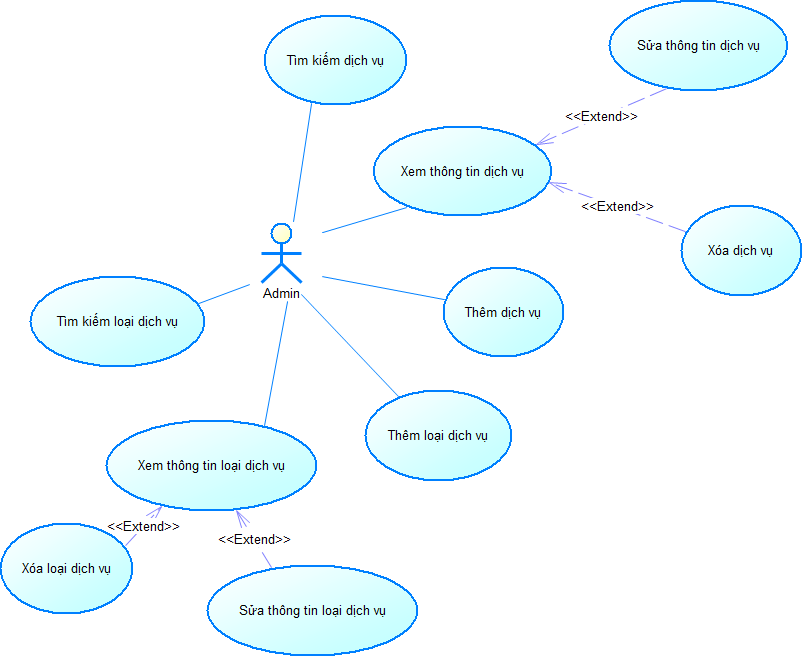
HÌnh 4. Use Case Admin chức năng quản lý khách hàng

1. Chức năng quản lý phòng



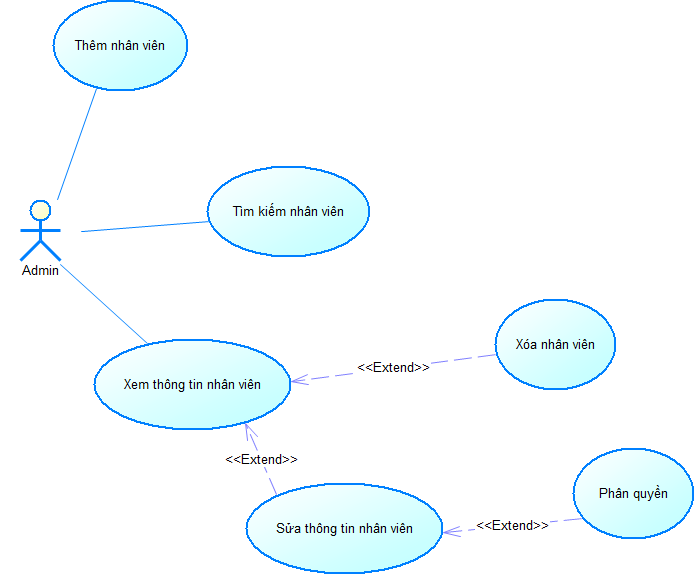
HÌnh 5. Use Case Admn chức năng quản lý phòng

1. Chức năng quản lý dịch vụ



HÌnh 6. Use Case Admin chức năng quản lý dịch vụ

1. Chức năng quản lý nhân sự



HÌnh 7. Use Case Admin chức năng quản lý nhân sự

1. Đặc tả ràng buộc toàn vẹn
2. Use case Đăng nhập

|  |  |
| --- | --- |
| **Use case** | Đăng nhập |
| **Actor** | Lễ tân, Admin |
| **Brief Description** | Use case mô tả các bước đăng nhập của Actor vào hệ thống. |
| **Pre-conditions** | Không có |
| **Basic Flow** | 1. Hệ thống yêu cầu actor nhập tên đăng nhập và mật khẩu; 2. Actor nhập tên đăng nhập và 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 phép actor đăng nhập |
| **Alternative Flow** | 1. Tên đăng nhập hoặc mật khẩu không đúng hệ thống sẽ báo cho người dùng và yêu cầu đăng nhập lại |
| **Exception Flows** | Không có |
| **Post-conditions** | Cho phép đăng nhập vào hệ thống nếu đăng nhập thành công |
| **Special Requirements** | Không có |

Table 1

1. Use case Đổi mật khẩu

|  |  |
| --- | --- |
| **Use case** | Đổi mật khẩu |
| **Actor** | Lễ tân, Admin |
| **Brief Description** | Use case mô tả việc đổi mật khẩu của Actor |
| **Pre-conditions** | Không có |
| **Basic Flow** | 1. Actor chọn chức năng đổi mật khẩu 2. Hệ thống hiện form thay đổi mật khẩu 3. Actor điền thông tin như:   +Tên tài khoản  +Mật khẩu mới  +Xác nhận lại mật khẩu mới   1. Actor nhấn nút đổi mật khẩu 2. Hệ thống thông báo đã đổi mật khẩu |
| **Alternative Flow** | 4.1 Actor không nhấn nút đổi mật khẩu mà thoát form |
| **Exception Flows** | Không có |
| **Post-conditions** | Đổi mật khẩu |
| **Special Requirements** | Không có |

Bảng 1. Bảng mô tả Use case Đổi mật khẩu

1. Use case Đăng xuất

|  |  |
| --- | --- |
| **Use case** | Đăng xuất |
| **Actor** | Lễ tân, admin |
| **Brief Description** | Use case mô tả việc đăng xuất của actor |
| **Pre-conditions** | Actor phải đăng nhập thành công |
| **Basic Flow** | 1. Actor chọn chức năng đăng xuất khỏi hệ thống 2. Hệ thống yêu cầu xác nhận từ actor 3. Actor chọn xác nhận 4. Hệ thống đăng xuất Actor |
| **Alternative Flow** | 3.1 Actor không chọn đăng xuất thì hệ thống sẽ giữ nguyên trạng |
| **Exception Flows** | Không có |
| **Post-conditions** | Đăng xuất actor khỏi hệ thống và bỏ quyền sử dụng hệ thống của actor |
| **Special Requirements** | Không có |

Bảng 2. Bảng mô tả Use case Đăng xuất

1. Use case Đặt phòng

|  |  |
| --- | --- |
| **Use case** | Đặt phòng |
| **Actor** | Lễ tân, admin |
| **Brief Description** | Use case cho phép actor nhận việc đặt phòng trước của khách |
| **Pre-conditions** | Actor phải đăng nhập vào hệ thống |
| **Basic Flow** | Use case được thực hiện khi khách hàng muốn đặt phòng trước bằng cách gọi điện cho actor hoặc đến trực tiếp khách sạn để đăng ký.  1. Bộ phận actor 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;  + Số điện thoại.  3. Bộ phận actor 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. actor chọn phòng theo yêu cầu của khách hàng đã đặt.  6. actor chọn nút “Đăng ký” để hoàn tất việc đăng ký phòng cho khách hàng.  7. Hệ thống kiểm tra dữ liệu actor vừa nhập.  8. Hệ thống lưu lại thông tin đặt phòng của khách hàng.  9. Kết thúc Use case |
| **Alternative Flow** | * 1. Loại phòng mà khách hàng yêu cầu đã hết phòng trống 3.1.2. actor thông báo cho khách hàng và yêu cầu khách hàng chọn loại phòng khác. actor 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. actor 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. |
| **Exception Flows** | Không có |
| **Post-conditions** | 3.1.2. actor thông báo cho khách hàng và yêu cầu khách hàng chọn loại phòng khác. actor 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. actor 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. |
| **Special Requirements** | Không có |

Bảng 3. Bảng mô tả Use case Đặt phòng

1. Use case Kiểm tra tình trạng phòng

|  |  |
| --- | --- |
| **Use case** | Kiểm tra tình trạng phòng |
| **Actor** | Lễ tân, admin |
| **Brief Description** | Use case này do hệ thống thực hiện để cung cấp cho actor biết tình trạng phòng của một phòng bất kỳ nào đó. |
| **Pre-conditions** | Actor phải đăng nhập vào hệ thống |
| **Basic Flow** | Use case thực hiện khi actor chọn chức năng “Đặt phòng” hoặc “Thuê phòng” với một phòng.  1. Hệ thống sẽ tìm kiếm thông tin phòng dựa vào mã phòng và phản hồi lại tình trạng hiện tại của phòng (đang ở, đã được đặt trước hay còn trống).  2. Kết thúc Use case |
| **Alternative Flow** | Không có |
| **Exception Flows** | Không có |
| **Post-conditions** | Cho biết tình trạng của phòng |
| **Special Requirements** | Không có |

Bảng 4. Bảng mô tả Use case Kiểm tra tình trạng phòng

1. Use case Tra cứu phòng

|  |  |
| --- | --- |
| **Use case** | Tra cứu phòng |
| **Actor** | Lễ tân, admin |
| **Brief Description** | Use case này cho phép actor 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** | Actor phải đăng nhập vào hệ thống |
| **Basic Flow** | Use case thực hiện khi actor 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 actor cung cấp (đang ở, đã được đặt trước hay còn trống).  4. Kết thúc Use case |
| **Alternative Flow** | Không có |
| **Exception 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 |

Bảng 5. Bảng mô tả Use case Tra cứu phòng

1. Use case Thuê phòng đặt trước

|  |  |
| --- | --- |
| **Use case** | Thuê phòng đặt trước |
| **Actor** | Admin, Lễ tân |
| **Brief Description** | Use case này cho phép actor thực hiện đăng ký thuê phòng cho khách đã có đăng ký trước. |
| **Pre-conditions** | Actor phải đăng nhập vào hệ thống |
| **Basic Flow** | Use case được thực hiện khi khi khách hàng đến thuê phòng trực tiếp tại quầy actor trong khách sạn.  1. Actor chọn chức năng thuê phòng để tiến hành thủ tục thuê phòng cho khách hàng.  2. Actor chọn tùy chọn “đặt phòng trước”  3. Hệ thống hiện thị form yêu cầu nhập thông tin khách hàng đặt phòng trước (CMND, số điện thoại).  4. Actor nhập vào thông tin và click chọn “Tìm” để tiến hành tìm thông tin đặt phòng của khách.  5. Hệ thống kích hoạt Use case “Tìm thông tin đặt phòng” và lấy thông tin tìm được hiển thị vào phiếu thuê phòng để actor thực hiện thuê phòng cho khách.  6. Actor đối chiếu giấy tờ tùy thân và thực hiện cập nhật lại thông tin khách hàng nếu có sai sót và click chọn “Thuê phòng” để hoàn tất thủ tục thuê phòng cho khách.  7. Hệ thống lưu lại thông tin thuê phòng của khách và cập nhật lại thông tin khách hàng nếu có sự thay đổi đồng thời hiển thị thông báo việc hoàn tất thủ thục thuê phòng.  8. Kết thúc Use case |
| **Alternative Flow** | 5.1. Không tìm thấy thông tin đặt phòng của khách hàng  5.1.1. Hệ thống thông báo cho actor và kích thay đổi tùy chọn thành “Đăng ký mới”.  5.1.2. Use case đăng ký mới được kích hoạt. |
| **Exception Flows** | Không có |
| **Post-conditions** | Lưu thông tin nhận phòng (trực tiếp) của khách vào hệ thống nếu use case thực hiện thành công. |
| **Special Requirements** | Không có |

Bảng 6. Thuê phòng đặt trước

1. Use case Thuê phòng trực tiếp

|  |  |
| --- | --- |
| **Use case** | Thuê phòng trực tiếp |
| **Actor** | Lễ tân, Actor |
| **Brief Description** | Use case này cho phép actor thực hiện đăng ký thuê phòng cho khách. |
| **Pre-conditions** | Actor phải đăng nhập vào hệ thống |
| **Basic Flow** | Use case được thực hiện khi khi khách hàng đến thuê phòng trực tiếp tại quầy actor trong khách sạn.  1. Actor chọn chức năng thuê phòng để tiến hành thủ tục thuê phòng cho khách hàng.  2. Hệ thống hiện thị form yêu cầu nhập thông tin khách hàng đặt phòng trước (CMND, số điện thoại) cùng các thông tin thuê phòng với tùy chọn mặc định là “Đăng ký trực tiếp”.  3. Actor nhập thông tin khách hàng (dựa vào giấy tờ tùy thân).  4. Hệ thống hiển thị các phòng có thể thuê hiện tại.  5. Actor chọn phòng theo các tiêu chí khách hàng cung cấp và click “Thuê phòng” để hoàn tất thủ thục thuê phòng cho khách hàng.  6. Hệ thống lưu lại thông tin khách hàng và thông tin thuê phòng của khách hàng.  7. Kết thúc Use case |
| **Alternative Flow** | 4.1. Loại phòng mà khách hàng yêu cầu đã hết phòng trống  4.1.1. Hệ thống thông báo hết phòng với loại phòng đã chọn và yêu cầu chọn loại phòng khác.  4.1.2. Actor thông báo cho khách hàng và yêu cầu khách hàng chọn loại phòng khác. Actor 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.  4.2. Hết phòng  4.2.1. Actor 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.  5.1. Dữ liệu nhập không hợp lệ  5.1.1. Hệ thống thông báo và yêu cầu thực hiện lại.  6.1. 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. |
| **Exception Flows** | Không có |
| **Post-conditions** | Lưu thông tin thuê phòng của khách vào hệ thống nếu use case thực hiện thành công. |
| **Special Requirements** | Không có |

Bảng 7. Bảng mô tả Use case Thuê phòng đặt trước

1. Use case Tìm thông tin đặt phòng

|  |  |
| --- | --- |
| **Use case** | Tìm thông tin đặt phòng |
| **Actor** | Hệ thống |
| **Brief Description** | Use case này phép lấy thông tin đặt phòng của một khách hàng khách hàng đến nhận phòng mà đã đặt phòng trước đó. |
| **Pre-conditions** | Thực hiện chức năng đăng ký phòng trước |
| **Basic Flow** | Use case thực hiện khi lễ tân chọn chức năng “Tìm thông tin đặt phòng”  1. Lễ tân nhập số CMND của khách hàng để tiến hành tìm thông tin đặt phòng.  2. Hệ thống tìm kiếm thông tin đặt phòng của khách hàng và trả về kết quả.  3. Kết thúc Use case |
| **Alternative Flow** | Không có |
| **Exception Flows** | Không có |
| **Post-conditions** | Thông tin đặt phòng của khách hàng được trả về |
| **Special Requirements** | KHông có |

Bảng 8. Bảng mô tả Use case Tìm thông tin đặt phòng

1. Use case Lập phiếu dịch vụ

|  |  |
| --- | --- |
| **Use case** | Lập phiếu dịch vụ |
| **Actor** | Admin, Lễ tân |
| **Brief Description** | Use case này cho phép actor tiếp nhận yêu cầu và lập phiếu sử dụng dịch vụ của khách hàng. |
| **Pre-conditions** | Actor phải đăng nhập vào hệ thống |
| **Basic Flow** | 1. Actor chọn chức năng lập phiếu dịch vụ 2. Hệ thống sẽ tạo ra phiếu dịch vụ ứng với thông tin nhận phòng tương ứng và hiện hiện thị thông tin ra actor xem đồng thời yêu cầu actor chọn các dịch mà khách hàng yêu cầu.  3. Nhân viên click chọn “Thêm phiếu dịch vụ”.  4. Hệ thống lưu lại phiếu sử dụng dịch vụ, đồng thời lưu thông tin chi tiết xuống “Chi tiết phiếu dịch vụ”.  5. Kết thúc Use case |
| **Alternative Flow** | Không có |
| **Exception Flows** | Không cso |
| **Post-conditions** | Lưu thông tin phiếu sử dụng dịch vụ của khách vào hệ thống nếu use case thực hiện thành công. |
| **Special Requirements** | Không có |

Bảng 9. Bảng mô tả Use case Lập phiếu dịch vụ

1. Use case Lập hóa đơn

|  |  |
| --- | --- |
| **Use case** | Lập hóa đơn |
| **Actor** | Lễ tân, admin |
| **Brief Description** | Use case này cho phép bộ phận lễ lập hóa đơn tính tiền khi khách hàng trả phòng |
| **Pre-conditions** | Actor phải đăng nhập vào hệ th |
| **Basic Flow** | 1. Actor 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. Actor 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 Flow** | Không có |
| **Exception Flows** | Không có |
| **Post-conditions** | Thêm thông tin hóa đơn, chi tiết hóa đơn xuống CSDL |
| **Special Requirements** | Không có |

Bảng 10. Bảng mô tả Use case Lập hóa đơn

1. Use case Tìm kiếm khách hàng

|  |  |
| --- | --- |
| **Use case** | Tìm kiếm khách hàng |
| **Actor** | Admin |
| **Brief Description** | Use case này cho admin tìm kiếm chính xác thông tin khách hàng một cách nhanh chóng. |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống |
| **Basic Flow** | 1.Admin chọn chức năng tìm kiếm khách hàng.  2.Hệ thống hiển thị form yêu cầu nhập thông tin số điện thoại của khách hàng  3.Admin nhập thông tin vào form và nhấn nút tìm.  4. Hệ thống hiển thị chi tiết thông tin khách hàng |
| **Alternative Flow** | Không có |
| **Exception Flows** | Không có |
| **Post-conditions** | 1.1 Không có thông tin khách hàng trong CSDL  4.1.1.Hệ thống hiển thị “không tìm thấy khách hàng” và kết thúc use case |
| **Special Requirements** | Không có |

Bảng 11. Bảng mô tả Use case Tìm kiếm khách hàng

1. Use case Thêm khách hàng

|  |  |
| --- | --- |
| **Use case** | Thêm khách hàng |
| **Actor** | Admin |
| **Brief Description** | Use case này cho admin thêm khách hàng mới vào danh sách khách hàng |
| **Pre-conditions** | Admin phải đăng nhập vào hệ thống |
| **Basic Flow** | 1.Admin chọn nút ”thêm”.  2.Hệ thống hiển thị form yêu cầu nhập thông tin:  + Tên khách hàng  + Email  + Số điện thoại khách hàng  3.Admin chọn nút “ok”  4. Hệ thống sẽ thêm khách hàng và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Admin chọn “Hủy”  3.1.1 Hệ thống về trạng thái ban đầu |
| **Post-conditions** | Khách hàng mới được thêm vào |
| **Special Requirements** | Không có |

Bảng 12. Bảng mô tả Use case Thêm khách hàng

1. Use case Sửa thông tin khách hàng

|  |  |
| --- | --- |
| **Use case** | Sửa thông tin khách hàng |
| **Actor** | Admin |
| **Brief Description** | Use case này cho admin sửa thông tin khách hàng |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác khách hàng cần sửa |
| **Basic Flow** | 1.Admin chọn nút ”sửa”.  2.Hệ thống hiển thị form cho phép sửa thông tin khách hàng.  3.Admin nhập thông tin mới vào những chổ cần sửa trong form và nhấn nút “lưu”.  4.Hệ thống sẽ cập nhật lại thông tin |
| **Alternative Flow** | Không |
| **Exception Flows** | 3.1 Admin chọn “Hủy”  3.1.1 Hệ thống về trạng thái ban đầu |
| **Post-conditions** | Thông tin khách hàng được cập nhật lại |
| **Special Requirements** | Không có |

Bảng 13. Bảng mô tả Use case Sửa thông tin khách hàng

1. Use case Xóa khách hàng

|  |  |
| --- | --- |
| **Use case** | Xóa khách hàng |
| **Actor** | Admin |
| **Brief Description** | Use case này cho admin xóa thông tin khách hàng |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác khách hàng cần xóa |
| **Basic Flow** | 1.Admin chọn nút ”xóa”.  2.Hệ thống hiển thị “Bạn có chắc muốn xóa khách hàng không”  3.Admin chọn nút “ok”  4. Hệ thống sẽ xóa khách hàng và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin khách hàng bị xóa |
| **Special Requirements** | Không |

Bảng 14. Xóa thông tin khách hàng

1. Use case Tìm kiếm phòng

|  |  |
| --- | --- |
| **Use case** | Tìm kiếm phòng |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin tìm kiếm chính xác thông tin phòng một cách nhanh chóng. |
| **Pre-conditions** | Admin phải đăng nhập vào hệ thống |
| **Basic Flow** | 1.Admin chọn chức năng tìm kiếm phòng.  2.Hệ thống hiển thị form yêu cầu nhập tên phòng  3.Admin nhập thông tin vào form và nhấn nút tìm.  4. Hệ thống hiển thị chi tiết thông tin phòng |
| **Alternative Flow** | Không |
| **Exception Flows** | Không |
| **Post-conditions** | 1.1 Không có thông tin phòng trong CSDL  4.1.1.Hệ thống hiển thị “không tìm thấy phòng” và kết thúc use case |
| **Special Requirements** | Không có |

Bảng 15. Bảng mô tả Use case Tìm kiếm phòng

1. Use case Thêm phòng

|  |  |
| --- | --- |
| **Use case** | Thêm phòng |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin thêm |
| **Pre-conditions** | Admin phải đăng nhập vào hệ thống |
| **Basic Flow** | 1.Admin chọn nút ”thêm”.  2.Hệ thống hiển thị form yêu cầu nhập thông tin:  + Tên phòng  + Tầng  + Loại phòng  3.Admin chọn nút “ok”  4. Hệ thống sẽ thêm phòng và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Admin chọn “Hủy”  3.1.1 Hệ thống về trạng thái ban đầu |
| **Post-conditions** | Phòng mới được thêm vào CSDL |
| **Special Requirements** | Không có |

Bảng 16. Bảng mô tả Use case Thêm phòng

1. Use case Sửa thông tin phòng

|  |  |
| --- | --- |
| **Use case** | Sửa thông tin phòng |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin sửa thông tin phòng |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác phòng cần sửa |
| **Basic Flow** | 1.Admin chọn nút ”sửa”.  2.Hệ thống hiển thị form cho phép sửa thông tin phòng.  3.Admin nhập thông tin mới vào những chổ cần sửa trong form và nhấn nút “lưu”.  4.Hệ thống sẽ cập nhật lại thông tin |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Admin chọn “Hủy”  3.1.1 Hệ thống về trạng thái ban đầu |
| **Post-conditions** | Thông tin phòng được cập nhật lại |
| **Special Requirements** | Không có |

Bảng 17. Bảng mô tả Use case Sửa thông tin phòng

1. Use case Xóa phòng

|  |  |
| --- | --- |
| **Use case** | Xóa phòng |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin xóa thông tin phòng |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác phòng cần xóa |
| **Basic Flow** | 1.Admin chọn nút ”xóa”.  2.Hệ thống hiển thị “Bạn có chắc muốn xóa phòng này không”  3.Admin chọn nút “ok”  4. Hệ thống sẽ xóa khách hàng và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin phòng bị xóa |
| **Special Requirements** | Không có |

Bảng 18. Bảng mô tả Use case Xóa phòng

1. Use case Tìm loại phòng

|  |  |
| --- | --- |
| **Use case** | Tìm loại phòng |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin tìm kiếm chính xác thông tin loại phòng một cách nhanh chóng. |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống |
| **Basic Flow** | 1.Admin chọn chức năng tìm kiếm loại phòng.  2.Hệ thống hiển thị form yêu cầu nhập tên loại phòng  3.Admin nhập thông tin vào form và nhấn nút tìm.  4. Hệ thống hiển thị chi tiết thông tin loại |
| **Alternative Flow** | Không có |
| **Exception Flows** | Không có |
| **Post-conditions** | 1.1 Không có thông tin khách hàng trong CSDL  4.1.1.Hệ thống hiển thị “không tìm thấy loại phòng” và kết thúc use case |
| **Special Requirements** | Không có |

Bảng 19. Bảng mô tả Use case Tìm loại phòng

1. Use case Thêm loại phòng

|  |  |
| --- | --- |
| **Use case** | Thêm loại phòng |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin thêm loại phòng mới vào danh sách loại phòng |
| **Pre-conditions** | Admin phải đăng nhập vào hệ thống |
| **Basic Flow** | 1.Admin chọn nút ”thêm”.  2.Hệ thống hiển thị form yêu cầu nhập thông tin:  + Tên loại phòng  + Số người  + Diện tích  + Mô tả  3.Admin chọn nút “ok”  4. Hệ thống sẽ thêm dịch vụ và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Loại phòng mới được thêm vào |
| **Special Requirements** | Không có |

Bảng 20. Bảng mô tả Use case Thêm loại phòng

1. Use case Sửa thông tin loại phòng

|  |  |
| --- | --- |
| **Use case** | Sửa thông tin loại phòng |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin sửa thông tin loại phòng |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác loại phòng cần sửa |
| **Basic Flow** | 1.Admin chọn nút ”sửa”.  2.Hệ thống hiển thị form cho phép sửa thông tin loại phòng.  3.Admin nhập thông tin mới vào những chổ cần sửa trong form và nhấn nút “lưu”.  4.Hệ thống sẽ cập nhật lại thông tin |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin loại phòng được cập nhật lại |
| **Special Requirements** | Không có |

Bảng 21. Bảng mô tả Use case Sửa thông tin phòng

1. Use case Xóa loại phòng

|  |  |
| --- | --- |
| **Use case** | Xóa loại phòng |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin xóa thông tin loại phòng |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác loại phòng cần xóa |
| **Basic Flow** | 1.Admin chọn nút ”xóa”.  2.Hệ thống hiển thị “Bạn có chắc muốn xóa loại phòng không”  3.Admin chọn nút “ok”  4. Hệ thống sẽ xóa loại phòng và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin loại phòng bị xóa |
| **Special Requirements** | Không có |

Bảng 22. Bảng mô tả Use case Xóa loại phòng

1. Use case Tìm thiết bị

|  |  |
| --- | --- |
| **Use case** | Tìm thiết bị |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin tìm kiếm chính xác thông tin thiết bị một cách nhanh chóng. |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống |
| **Basic Flow** | 1.Admin chọn chức năng tìm kiếm thiết bị.  2.Hệ thống hiển thị form yêu cầu nhập thông tin tên thiết bị.  3.Admin nhập thông tin vào form và nhấn nút tìm.  4. Hệ thống hiển thị chi tiết thông tin thiệt bị. |
| **Alternative Flow** | Không có |
| **Exception Flows** | Không có |
| **Post-conditions** | 1.1 Không có thông tin thiết bị trong CSDL  4.1.1.Hệ thống hiển thị “không tìm thấy thiết bị” và kết thúc use case |
| **Special Requirements** | Không có |

Bảng 23. Bảng 23. Bảng mô tả Use case Tìm thiết bị

1. Use case Thêm thiết bị

|  |  |
| --- | --- |
| **Use case** | Thêm thiết bị |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin thêm thiết bị mới vào danh sách thiết bị |
| **Pre-conditions** | Admin phải đăng nhập vào hệ thống |
| **Basic Flow** | 1.Admin chọn nút ”thêm”.  2.Hệ thống hiển thị form yêu cầu nhập thông tin:  + Tên thiết bị  + Giá  + Đơn vị  3.Admin chọn nút “ok”  4. Hệ thống sẽ thêm thiết bị và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thiết bị mới được thêm vào |
| **Special Requirements** | Không có |

Bảng 24. Bảng mô tả Use case Thêm thiết bị

1. Use case Sửa thông tin thiết bị

|  |  |
| --- | --- |
| **Use case** | Sửa thông tin thiết bị |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin sửa thông tin thiết bị |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác thiết bị cần sửa |
| **Basic Flow** | 1.Admin chọn nút ”sửa”.  2.Hệ thống hiển thị form cho phép sửa thông tin thiết bị.  3.Admin nhập thông tin mới vào những chổ cần sửa trong form và nhấn nút “lưu”.  4.Hệ thống sẽ cập nhật lại thông tin |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin thiết bị được cập nhật lại |
| **Special Requirements** | Không có |

Bảng 25. Bảng mô tả Use case Sửa thông tin thiết bị

1. Use case Xóa thiết bị

|  |  |
| --- | --- |
| **Use case** | Xóa thiết bị |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin xóa thông tin thiết bị |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác thiết bị cần xóa |
| **Basic Flow** | 1.Admin chọn nút ”xóa”.  2.Hệ thống hiển thị “Bạn có chắc muốn xóa thiết bị không”  3.Admin chọn nút “ok”  4. Hệ thống sẽ xóa thiết bị và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin thiết bị bị xóa |
| **Special Requirements** | Không có |

Bảng 26. Bảng mô tả Use case Xóa thiết bị

1. Use case Tìm dịch vụ

|  |  |
| --- | --- |
| **Use case** | Tìm dịch vụ |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin tìm kiếm chính xác thông tin dịch vụ một cách nhanh chóng. |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống |
| **Basic Flow** | 1.Admin chọn chức năng tìm kiếm dịch vụ.  2.Hệ thống hiển thị form yêu cầu nhập thông tin tên dịch vụ hoặc chọn lại dịch vụ của dịch vụ  3.Admin nhập thông tin vào form và nhấn nút tìm.  4. Hệ thống hiển thị chi tiết thông tin dịch vụ |
| **Alternative Flow** | Không có |
| **Exception Flows** | Không có |
| **Post-conditions** | 1.1 Không có thông tin khách hàng trong CSDL  4.1.1.Hệ thống hiển thị “không tìm thấy dịch vụ” và kết thúc use case |
| **Special Requirements** | Không có |

Bảng 27.Bảng mô tả Use case Tìm dịch vụ

1. Use case Thêm dịch vụ

|  |  |
| --- | --- |
| **Use case** | Thêm dịch vụ |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin thêm dịch vụ mới vào danh sách khách hàng |
| **Pre-conditions** | Admin phải đăng nhập vào hệ thống |
| **Basic Flow** | 1.Admin chọn nút ”thêm”.  2.Hệ thống hiển thị form yêu cầu nhập thông tin:  + Tên dịch vụ  + Loại dịch vụ  + Giá  + Đơn vị  3.Admin chọn nút “ok”  4. Hệ thống sẽ thêm dịch vụ và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Dịch vụ mới được thêm vào |
| **Special Requirements** | Không có |

Bảng 28. Bảng mô tả Use case Thêm dịch vụ

1. Use case Sửa thông tin dịch vụ

|  |  |
| --- | --- |
| **Use case** | Sửa thông tin dịch vụ |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin sửa thông tin dịch vụ |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác dịch vụ cần sửa |
| **Basic Flow** | 1.Admin chọn nút ”sửa”.  2.Hệ thống hiển thị form cho phép sửa thông tin dịch vụ.  3.Admin nhập thông tin mới vào những chổ cần sửa trong form và nhấn nút “lưu”.  4.Hệ thống sẽ cập nhật lại thông tin |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin dịch vụ được cập nhật lại |
| **Special Requirements** | Không có |

Bảng 29. Bảng mô tả Use case Sửa thông tin dịch vụ

1. Use case Xóa dịch vụ

|  |  |
| --- | --- |
| **Use case** | Xóa dịch vụ |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin xóa thông tin dịch vụ |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác dịch vụ cần xóa |
| **Basic Flow** | 1.Admin chọn nút ”xóa”.  2.Hệ thống hiển thị “Bạn có chắc muốn xóa dịch vụ không”  3.Admin chọn nút “ok”  4. Hệ thống sẽ xóa dịch vụ và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin dịch vụ bị xóa |
| **Special Requirements** | Không có |

Bảng 30. Bảng mô tả Use case Xóa dịch vụ

1. Use case Tìm nhân viên

|  |  |
| --- | --- |
| **Use case** | Tìm nhân viên |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin tìm kiếm chính xác thông tin nhân viên một cách nhanh chóng. |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống |
| **Basic Flow** | 1.Admin chọn chức năng tìm kiếm nhân viên.  2.Hệ thống hiển thị form yêu cầu nhập thông tin số điện thoại của nhân viên  3.Admin nhập thông tin vào form và nhấn nút tìm.  4. Hệ thống hiển thị chi tiết thông tin nhân viên |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | 1.1 Không có thông tin khách hàng trong CSDL  4.1.1.Hệ thống hiển thị “không tìm thấy nhân viên” và kết thúc use case |
| **Special Requirements** | Không có |

Bảng 31. Bảng mô tả Use case Tìm nhân viên

1. Use case Thêm nhân viên

|  |  |
| --- | --- |
| **Use case** | Thêm nhân viên |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin thêm nhân viên mới vào danh sách nhân viên |
| **Pre-conditions** | Admin phải đăng nhập vào hệ thống |
| **Basic Flow** | 1.Admin chọn nút ”thêm”.  2.Hệ thống hiển thị form yêu cầu nhập thông tin:  + Tên nhân viên  + Ngày sinh  + Giới tính  + Địa chỉ  + Số điện thoại  + Tài khoản  + Qyền hạn  3.Admin chọn nút “ok”  4. Hệ thống sẽ thêm dịch vụ và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Nhân viên mới được thêm vào |
| **Special Requirements** | Không có |

Bảng 32. Bảng mô tả Use case Thêm nhân viên

1. Use case Sửa thông tin nhân viên

|  |  |
| --- | --- |
| **Use case** | Sửa thông tin nhân viên |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin sửa thông tin nhân viên |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác nhân viên cần sửa |
| **Basic Flow** | 1.Admin chọn nút ”sửa”.  2.Hệ thống hiển thị form cho phép sửa thông tin nhân viên.  3.Admin nhập thông tin mới vào những chổ cần sửa trong form và nhấn nút “lưu”.  4.Hệ thống sẽ cập nhật lại thông tin |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin nhân viên được cập nhật lại |
| **Special Requirements** | Không có |

Bảng 33. Bảng mô tả Use case Sửa thông tin nhân viên

1. Use case Xóa nhân viên

|  |  |
| --- | --- |
| **Use case** | Xóa nhân viên |
| **Actor** | admin |
| **Brief Description** | Use case này cho admin xóa thông tin nhân viên |
| **Pre-conditions** | Admin phải đăng nhập thành công vào hệ thống và chọn chính xác nhân viên cần xóa |
| **Basic Flow** | 1.Admin chọn nút ”xóa”.  2.Hệ thống hiển thị “Bạn có chắc muốn xóa nhân viên không”  3.Admin chọn nút “ok”  4. Hệ thống sẽ xóa nhân viên và cập nhật lại danh sách. |
| **Alternative Flow** | Không có |
| **Exception Flows** | 3.1 Nhân viên quản lý chọn ”hủy”  3.1.1 Hệ thống trở về hiện trạng ban đầu |
| **Post-conditions** | Thông tin nhân viên bị xóa |
| **Special Requirements** | Không có |

Bảng 34. Bảng mô tả Use case Xóa nhân viên

1. THIẾT KẾ LƯỢC ĐỒ CƠ SỞ DỮ LIỆU
2. Đặt tả lược đồ cở sở dữ liệu quan hệ
3. **Tbl\_loaiPhong** (idLP, tenLP, moTaLP, soNguoi, dienTich, statusLP)
4. **Tbl\_thietBi** (idTB, tenTB, donViTinh, statusTB)
5. **Tbl\_trangBi** (#idThietBi, #idLoaiPhong, soLuong)
6. **Tbl\_thoiGian** (idTG, tenTG)
7. **Tbl\_donGia** (#idLP, #idG, gia)
8. **Tbl\_tang** (idTang, tenTang)
9. **Tbl\_trangThaiPhong** (idTTP, nameTTP)
10. **Tbl\_phong** (idPhong, #idTang, #idTTP, #idLP, tenPhong)
11. **Tbl\_quocTich** (idQT, nameQT)
12. **Tbl\_khachHang** (idKH, #idQT, tenKH, email, diaChi, CCCD, SDTKH, statusKH, sexKH)
13. **Tbl\_trangThaiBook** (idTTDT, tenTTDT)
14. **Tbl\_datPhong** (idDP, i#dKH, #idTTDT, slNguoi, trangThai, ngayDat)
15. **Tbl\_dangKiPhong** (#idPhong, #idDP, ngayDen, ngayDi, datTruoc)
16. **Tbl\_loaiDichVu** (idLDV, tenLDV, statusLDV)
17. **Tbl\_dichVu** (idDV, #idLDV, tenDV, doVi, giaDV, statusDV)
18. **Tbl\_phieuSuDungDV**(idPhieu, #idDP)
19. **Tbl\_SDDV**(#idPhieu, #idDV, solDV, idSD)
20. **Tbl\_nhanVien**(idNV, tenNV, ngaySinh, diaChiNV, emailNV, SDTNV, sexNV)
21. **Tbl\_taiKhoan** (tenTK, #idNV, matKhau, statusAcc)
22. **Tbl\_phanQuyen** (idPQ, tenPQ)
23. **Tbl\_thongTinPQ** (idTT, #idPQ, tenTT, maTT)
24. **Tbl\_TK\_PER** (#idPQ, #tenTK)
25. **Tbl\_hoaDon** (#idDP, idHD, #idNV, ngayLap, tongTien, tienPhong, tienDV, phatSinh, thue, pay, payment)
26. Biểu diễn các ràng buộc toàn vẹn

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Bảng con(Reference table)** |  | **Bảng cha (Primary table)** |
| 1 | **Tbl\_trangBi** | → | **Tbl\_thietBi** (idTB) |
| 2 | **Tbl\_trangBi** | → | **Tbl\_loaiPhong** (idLP) |
| 3 | **Tbl\_donGia** | → | **Tbl\_thoiGian** (idTG) |
| 4 | **Tbl\_donGia** | → | **Tbl\_loaiPhong** (idLP) |
| 5 | **Tbl\_phong** | → | **Tbl\_tang** (idTang) |
| 6 | **Tbl\_phong** | → | **Tbl\_loaiPhong** (idLP) |
| 7 | **Tbl\_phong** | → | **Tbl\_trangThaiPhong** (idTTP) |
| 8 | **Tbl\_khachHang** | → | **Tbl\_quocTich** (idQT) |
| 9 | **Tbl\_datPhong** | → | **Tbl\_khachHang** (idKH) |
| 10 | **Tbl\_datPhong** | → | **Tbl\_trangThaiBook** (idTTDT) |
| 11 | **Tbl\_dangKiPhong** | → | **Tbl\_phong** (idPhong) |
| 12 | **Tbl\_dangKiPhong** | → | **Tbl\_datPhong** (idDP) |
| 13 | **Tbl\_dichVu** | → | **Tbl\_loaiDichVu** (idLDV) |
| 14 | **Tbl\_phieuSuDungDV** | → | **Tbl\_datPhong** (idDP) |
| 15 | **Tbl\_SDDV** | → | **Tbl\_dichVu** (idDV) |
| 16 | **Tbl\_SDDV** | → | **Tbl\_phieuSuDungDV** (idPhieu) |
| 17 | **Tbl\_taiKhoan** | → | **Tbl\_nhanVien** (idNV) |
| 18 | **Tbl\_thongTinPQ** | → | **Tbl\_phanQuyen** (idPQ) |
| 19 | **Tbl\_TK\_PER** | → | **Tbl\_phanQuyen** (idPQ) |
| 20 | **Tbl\_TK\_PER** | → | **Tbl\_taiKhoan** (tenTK) |
| 21 | **Tbl\_hoaDon** | → | **Tbl\_nhanVien** (idNV) |
| 22 | **Tbl\_hoaDon** | → | **Tbl\_datPhong** (idDP) |

Bảng 35. Bảng biểu diễn ràng buộc toàn vẹn

1. Mô hình mức vật lý

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idLP | Int | ✓ |  |  | ✓ | ✓ | Mã loại phòng |
| tenLP | nvarchar(150) |  |  |  |  | ✓ | Tên loại phòng |
| moTaLP | nvarchar(250) |  |  | Chưa có mô tả |  | ✓ | Mô tả |
| soNguoi | Int |  |  |  |  | ✓ | Số người ở |
| dienTich | Int |  |  |  |  | ✓ | Diện tích phòng |
| statusLP | bit |  |  | 1 |  | ✓ | Trạng thái |

Bảng 36. Bảng loại phòng

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idTB | Int | ✓ |  |  | ✓ | ✓ | Mã thiết bị |
| tenTB | nvarchar(50) |  |  |  |  | ✓ | Tên thiết bị |
| giaTB | money |  |  | 0 |  | ✓ | Giá thiết bị |
| dvTinh | Nvarchar (10) |  |  | cái |  | ✓ | Đơn vị tính |
| statusTB | bit |  |  | 1 |  | ✓ | Trạng thái |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idLP | Int |  | ✓ |  |  | ✓ | Mã loại phòng |
| idTB | int |  | ✓ |  |  | ✓ | Mã thiết bị |
| soLuong | int |  |  | 1 |  | ✓ | Số lượng |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idTG | Int | ✓ |  |  | ✓ | ✓ | Mã thời gian |
| tenTG | nvarchar(20) |  |  |  |  | ✓ | Tên thời gian |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idLP | Int |  | ✓ |  |  | ✓ | Mã loại phòng |
| idTG | int |  | ✓ |  |  | ✓ | Mã thời gian |
| gia | money |  |  | 0 |  | ✓ | Giá loại phòng |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idTang | Int | ✓ |  |  | ✓ | ✓ | Mã tầng |
| tenTang | nvarchar(20) |  |  |  |  | ✓ | Tên tầng |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idTTP | Int | ✓ |  |  | ✓ | ✓ | Mã trạng thái phòng |
| nameTTP | nvarchar(50) |  |  |  |  | ✓ | Tên trạng thái phòng |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idPhong | int | ✓ |  |  | ✓ | ✓ | Mã phòng |
| idLP | Int |  | ✓ |  |  | ✓ | Mã loại phòng |
| idTTP | Int |  | ✓ |  |  | ✓ | Mã trạng thái phòng |
| tenPhong | nvarchar(10) |  |  |  |  | ✓ | Tên phòng |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idQT | int | ✓ |  |  | ✓ | ✓ | Mã quốc tịch |
| nameQT | Nvarchar(50) |  |  |  |  | ✓ | Tên quốc tịch |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idTTDP | int | ✓ |  |  | ✓ | ✓ | Mã trạng thái đặt phòng |
| tenTTDP | Nvarchar(50) |  |  |  |  | ✓ | Tên trạng thái đặt phòng |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idKH | int | ✓ |  |  | ✓ | ✓ | Mã khách hàng |
| idQT | Int |  | ✓ |  |  | ✓ | Mã quốc tịch |
| tenKH | nvarchar(50) |  |  |  |  | ✓ | Tên khách hàng |
| Email | Nvarchar(50) |  |  | Chưa nhập |  | ✓ | Email khách hàng |
| diaChi | Nvarchar(150) |  |  | Chưa nhập |  | ✓ | Địa chỉ khách hàng |
| CCCD | Nvarchar(50) |  |  | Chưa nhập |  | ✓ | Chính minh thư |
| SDTKH | Nvarchar(20) |  |  |  |  | ✓ | Số điện thoại khách hàng |
| gtKH | bit |  |  | 0 |  | ✓ | Giới tính khách hàng |
| statusKH | Bit |  |  | 1 |  | ✓ | Trạng thái |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idDP | Int | ✓ |  |  | ✓ | ✓ | Mã đặt phòng |
| idKH | int |  | ✓ |  |  | ✓ | Mã khách hàng |
| idTTDT | Int |  | ✓ |  |  | ✓ | Mã tình trạng đặt phòng |
| slNguoi | Int |  |  |  |  | ✓ | Số lượng người |
| ngayDat | dateTime |  |  |  |  | ✓ | Ngày đặt |
| hinhThuc | Nvarchar(50) |  |  |  |  | ✓ | Hình thức |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idLDV | Int | ✓ |  |  | ✓ | ✓ | Mã loại dịch vụ |
| tenLDV | Nvarchar(50) |  |  |  |  | ✓ | Tên loại dịch vụ |
| statusLDV | bit |  |  | 1 |  | ✓ | Tình trạng loại dịch vụ |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idDV | Int | ✓ |  |  | ✓ | ✓ | Mã dịch vụ |
| idLDV | int |  | ✓ |  |  | ✓ | Mã loại dịch vụ |
| tenDV | Nvarchar(50) |  |  |  |  | ✓ | Tên dịch vụ |
| donVi | Nvarchar(10) |  |  | Lần |  | ✓ | Đơn vị |
| giaDV | int |  |  | 0 |  | ✓ | Giá dịch vụ |
| statusDV | bit |  |  | 1 |  | ✓ | Tình trạng |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idPhieu | Int | ✓ |  |  | ✓ | ✓ | Mã phiếu sử dụng dịch vụ |
| idDP | int |  | ✓ |  |  | ✓ | Mã đặt phòng |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idSD | Int |  |  |  | ✓ | ✓ | Mã sử dụng |
| idDV | int |  | ✓ |  |  | ✓ | Mã dịch vụ |
| idPhieu | Int |  | ✓ |  |  | ✓ | Mã phiếu |
| slDV | Int |  |  | 1 |  | ✓ | Số lượng |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idNV | Int | ✓ |  |  | ✓ | ✓ | Mã nhân viên |
| tenNV | Nvarchar(50) |  |  |  |  | ✓ | Tên nhân viên |
| ngaySinh | date |  |  |  |  | ✓ | Ngày sinh |
| diaChiNV | Nvarchar(250) |  |  |  |  | ✓ | Địa chỉ |
| emailNV | Nvarchar(50) |  |  |  |  | ✓ | email |
| SDTNV | Nvarchar(20) |  |  | Chưa nhập |  | ✓ | Số diện thoại |
| sexNV | bit |  |  |  |  |  | Giới tính |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| tenTK | Nvarchar(20) | ✓ |  |  |  | ✓ | Tên tài khoản |
| idNV | int |  | ✓ |  |  | ✓ | Mã nhân viên |
| matKhau | Nvarchar(20) |  |  |  |  | ✓ | Mật khẩu |
| statusACC | bit |  |  |  |  | ✓ | Tình trạng tài khoản |

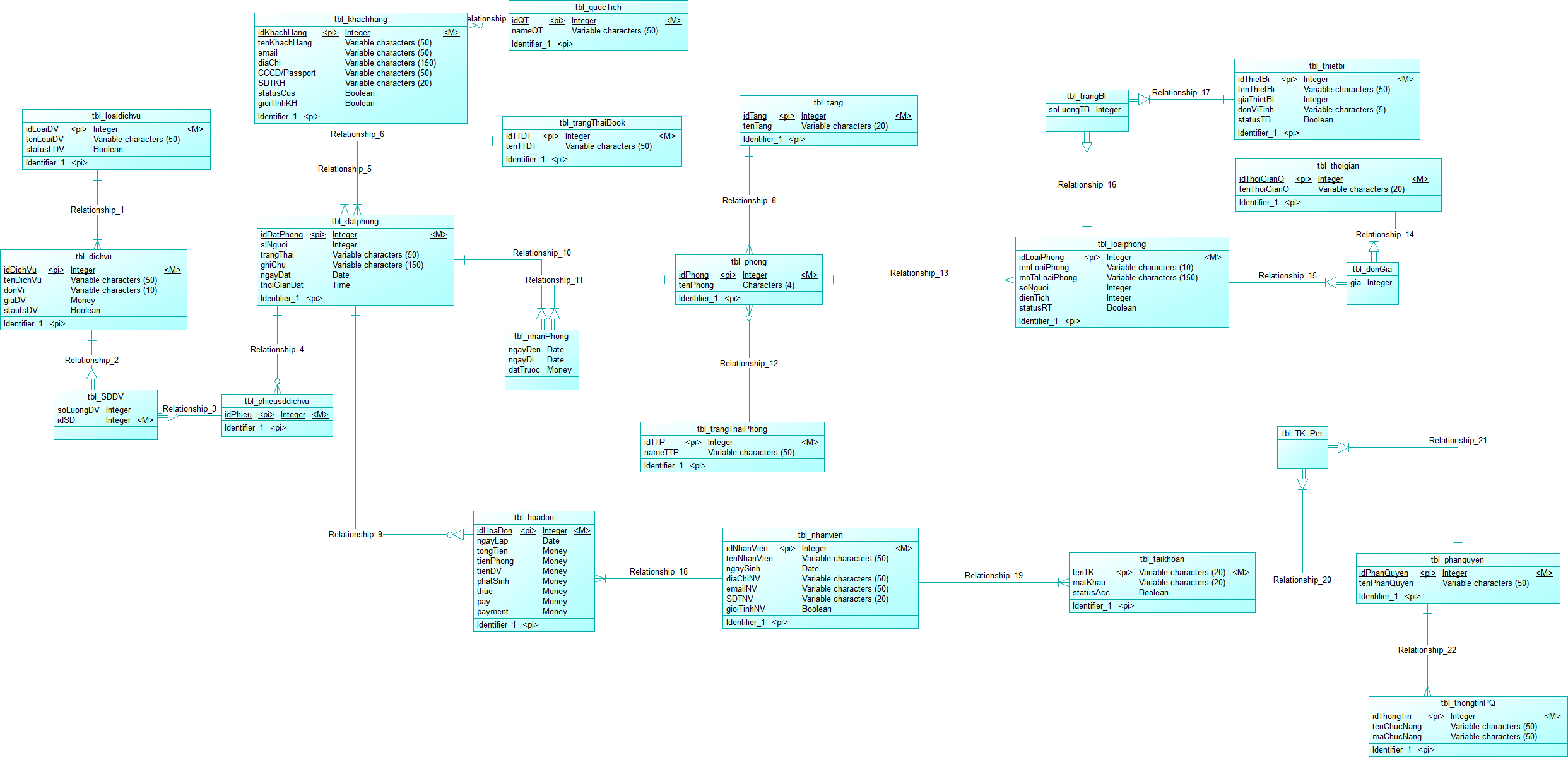
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idPQ | Int | ✓ |  |  | ✓ | ✓ | Mã phân quyền |
| tenPQ | Nvarchar(50) |  |  |  |  | ✓ | Tên phân quyền |

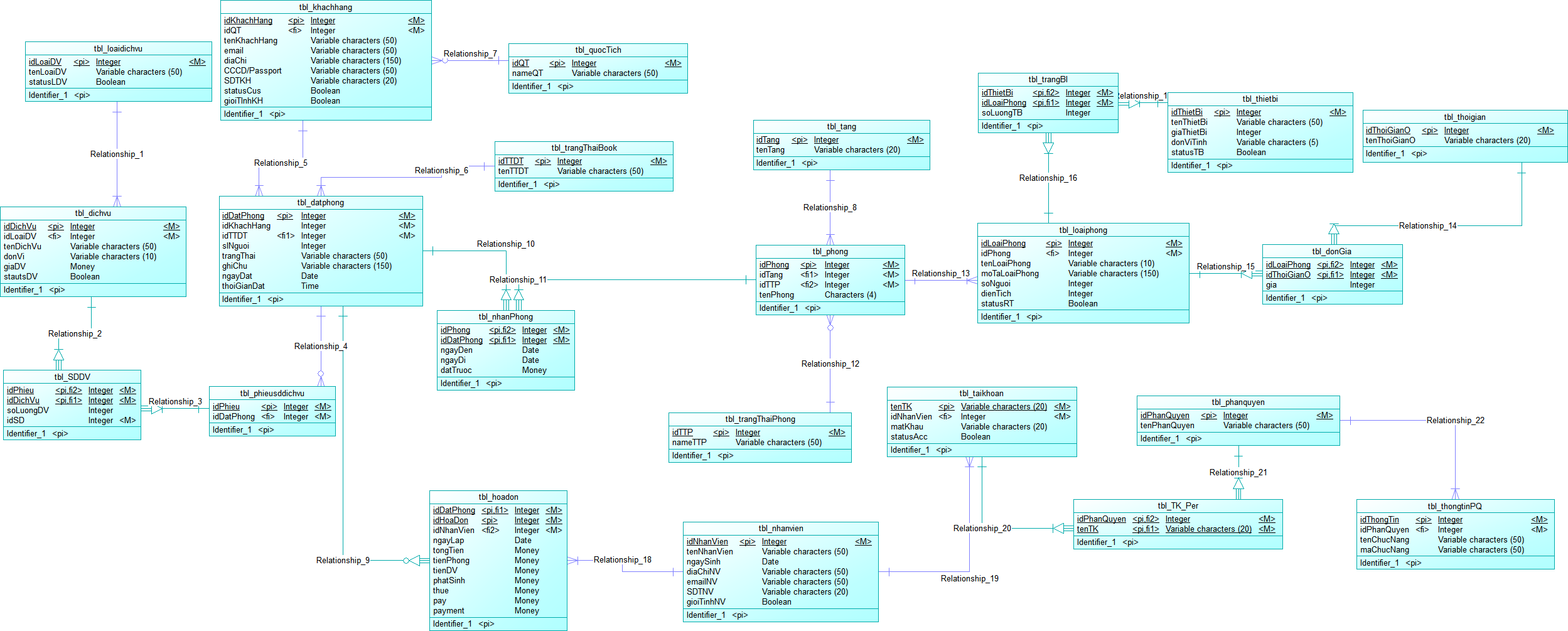
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idTT | Int | ✓ |  |  | ✓ | ✓ | Mã thông tin |
| idPQ | int |  | ✓ |  |  | ✓ | Mã phân quyền |
| tenTT | Int |  |  |  |  | ✓ | Tên chức năng |
| maTT | Int |  |  |  |  | ✓ | Mã chức năng |

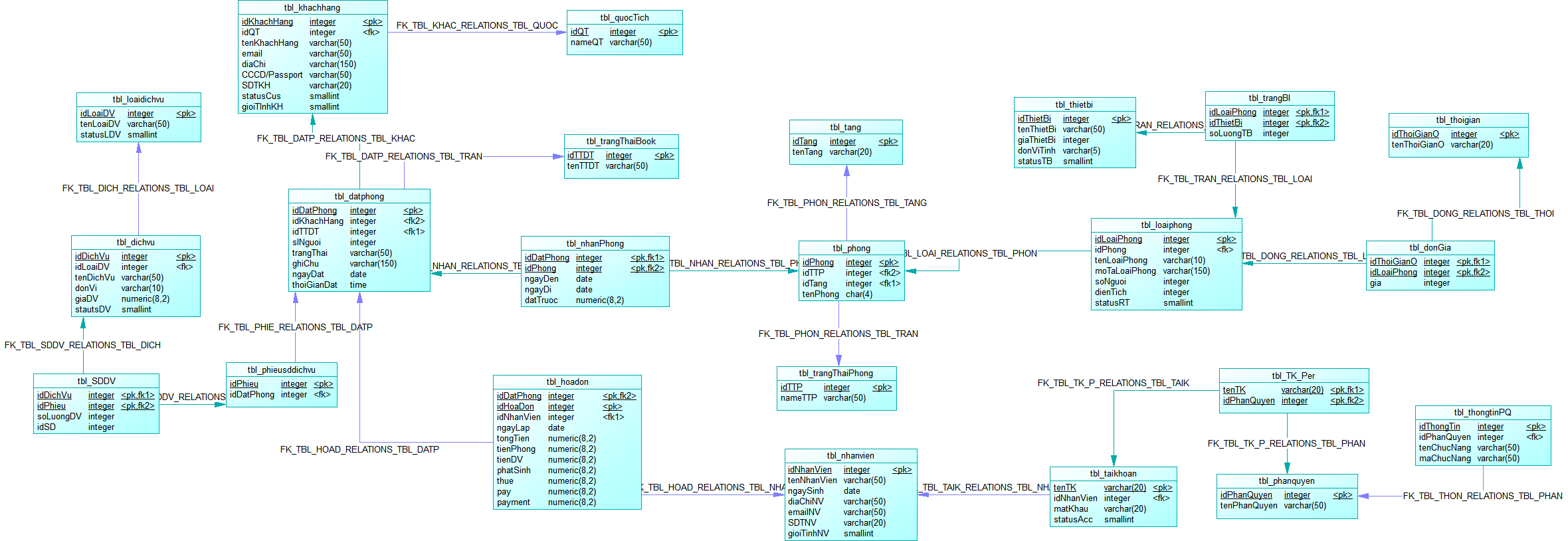
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idPQ | Int |  | ✓ |  |  | ✓ | Mã phân quyền |
| tenTK | Nvarchar(20) |  | ✓ |  |  | ✓ | Tên tài khoản |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tên thuộc tính | Kiểu  dữ  liệu | Khóa  chính | Khóa  ngoại | Giá trị  mặc định | Tự  tăng | NN | Diễn  giải |
| idHD | Int | ✓ |  |  | ✓ | ✓ | Mã hóa đơn |
| idDP | int |  | ✓ |  |  | ✓ | Mã đặt phòng |
| idNV | Int |  | ✓ |  |  | ✓ | Mã nhân viên |
| ngayLap | DateTime |  |  |  |  | ✓ | Ngày lập |
| tongTien | money |  |  |  |  | ✓ | Tổng tiền |
| tienPhong | money |  |  |  |  | ✓ | Tiền phòng |
| tienDV | money |  |  |  |  | ✓ | Phí dịch vụ |
| phatSinh | money |  |  |  |  | ✓ | Phát sinh |
| pay | money |  |  |  |  | ✓ | Khách trả |
| payment | money |  |  |  |  | ✓ | Tiền thồi |

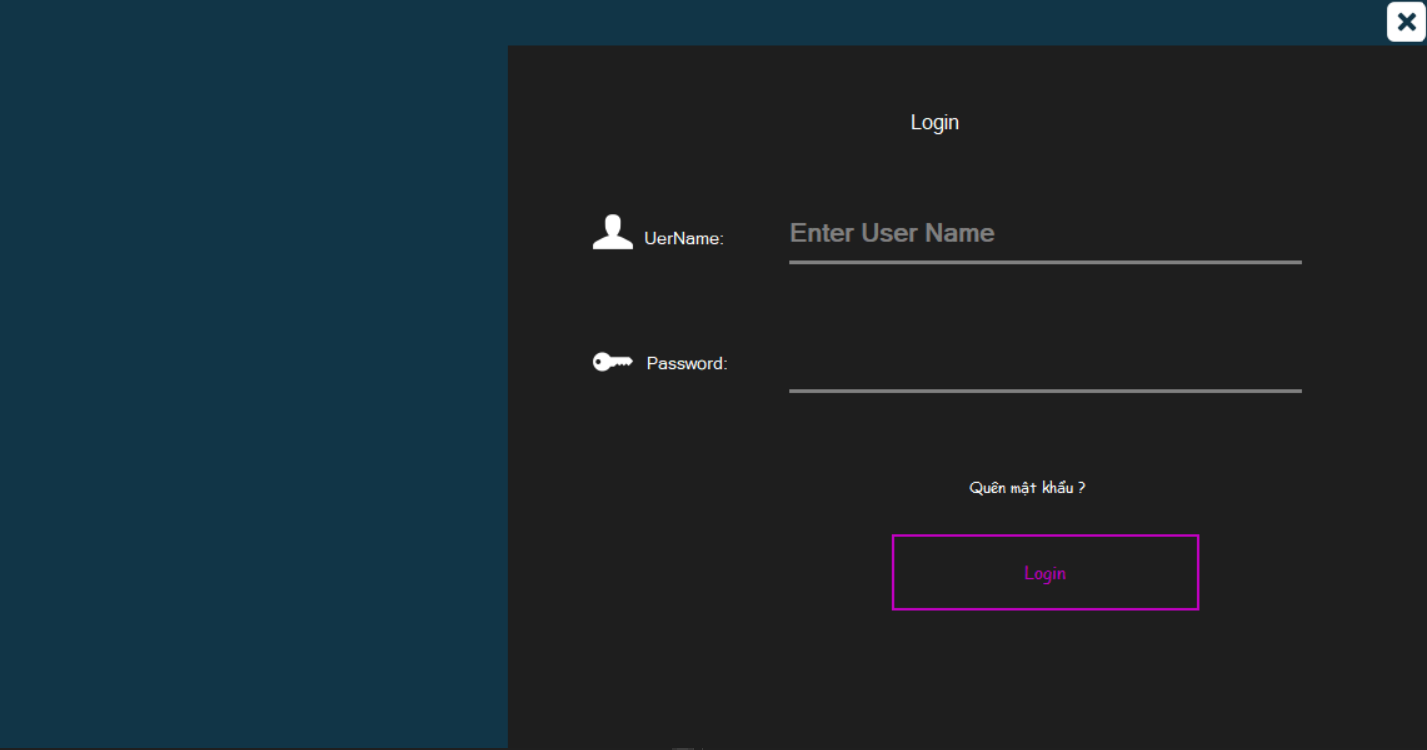
1. THIẾT KẾ THÀNH PHẦN XỬ LÝ
2. Các lưu đồ - giải thuật



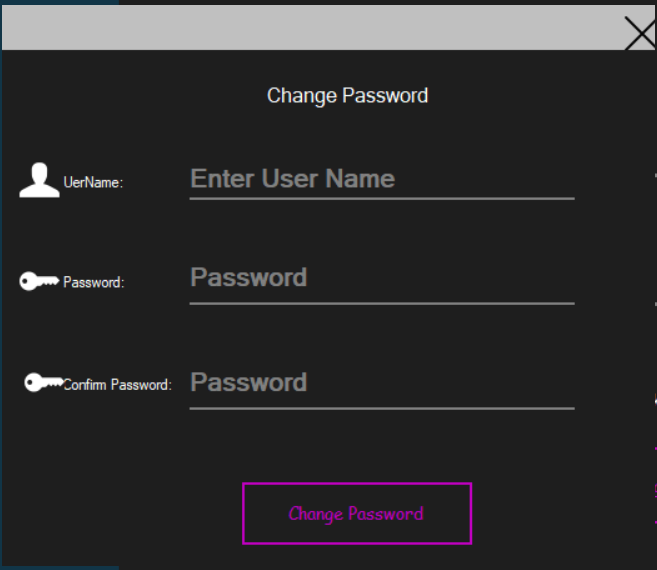




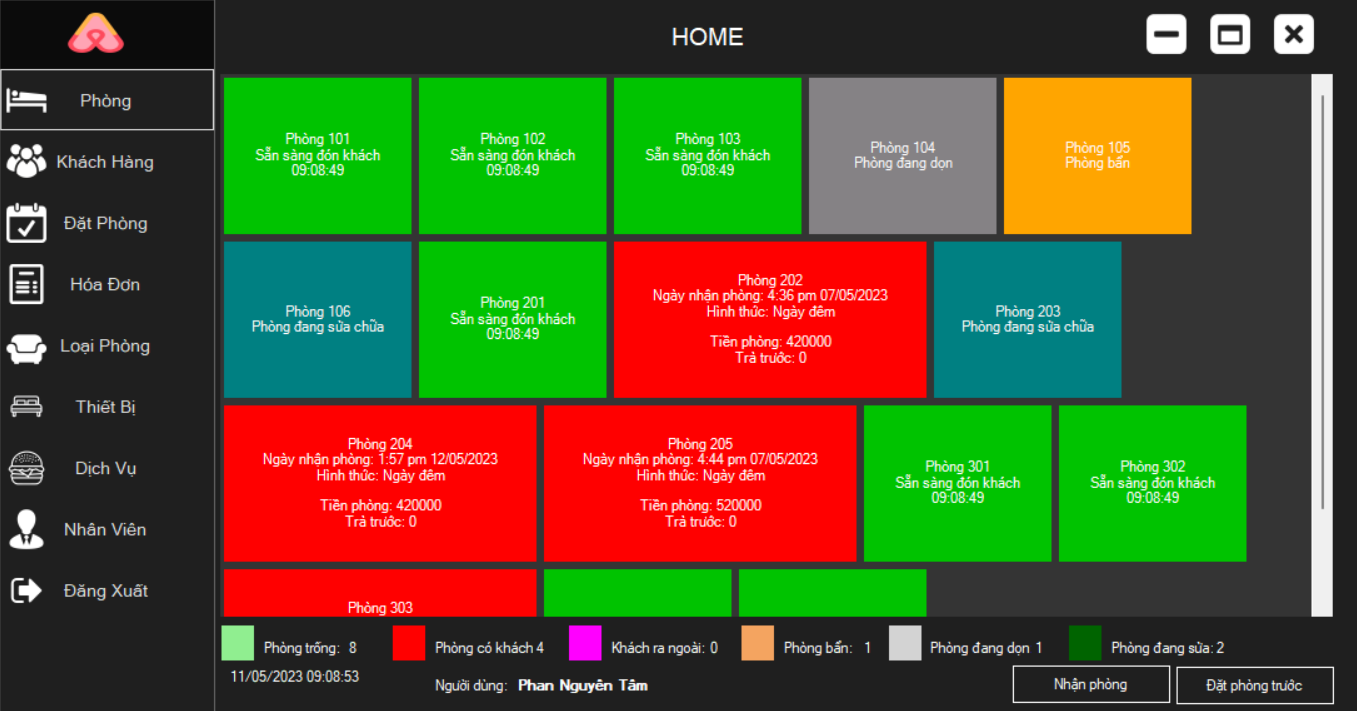
1. THIẾT KẾ GIAO DIỆN



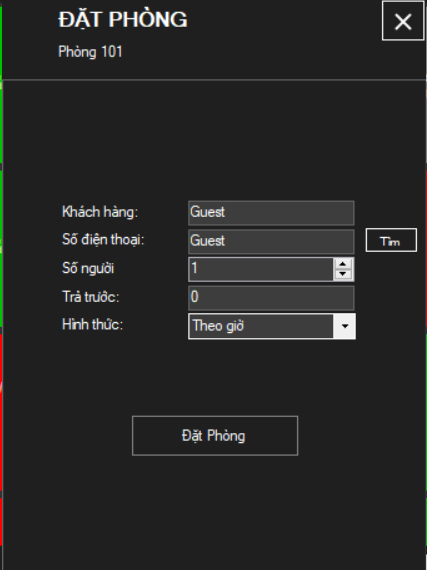
HÌnh 8. Giao diện đăng nhập



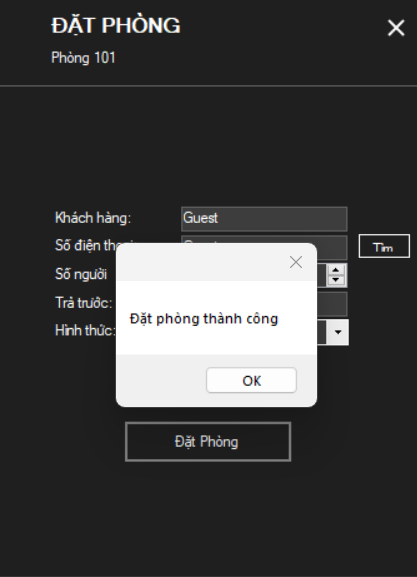
HÌnh 9. Giao diện đổi mật khẩu



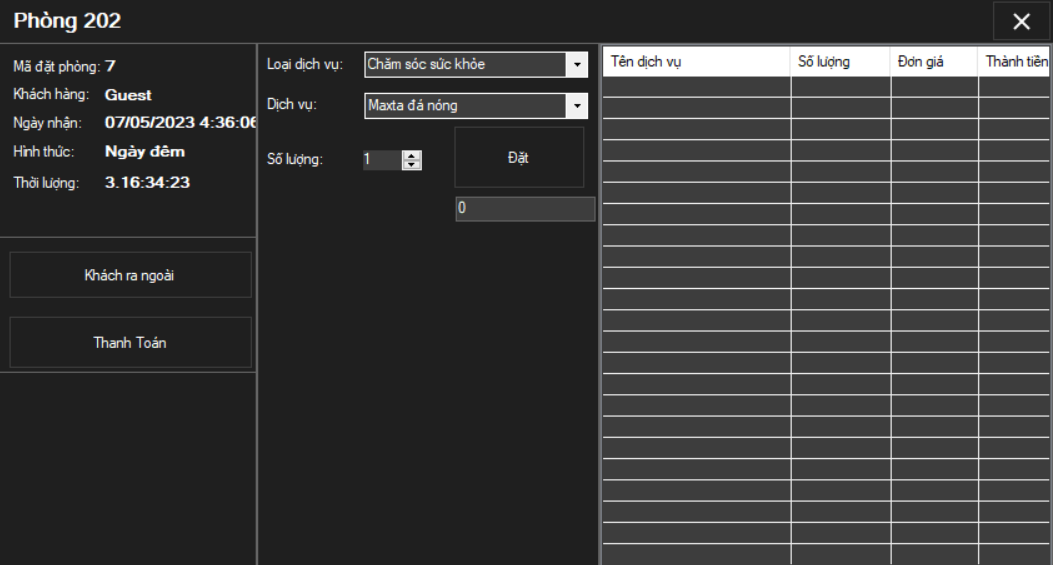
HÌnh 10. Giao diện chính



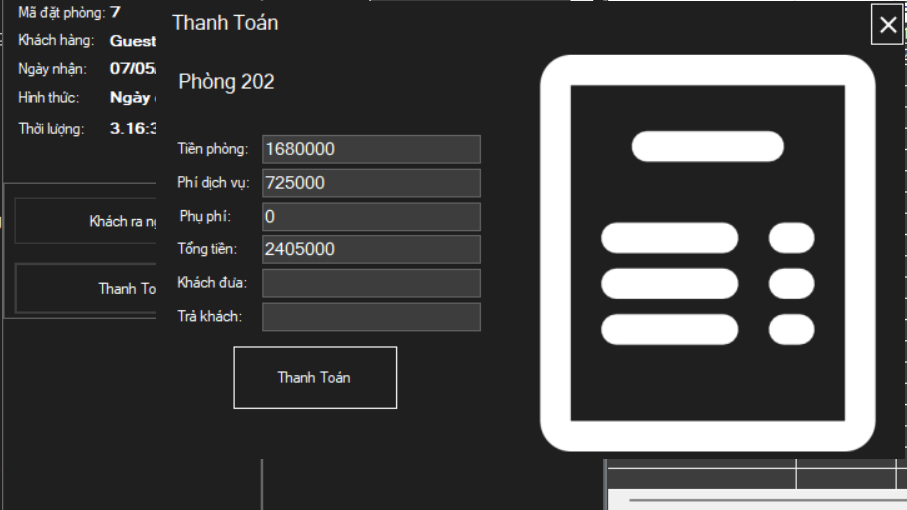
HÌnh 11. Giao diện đặt phòng trực tiếp



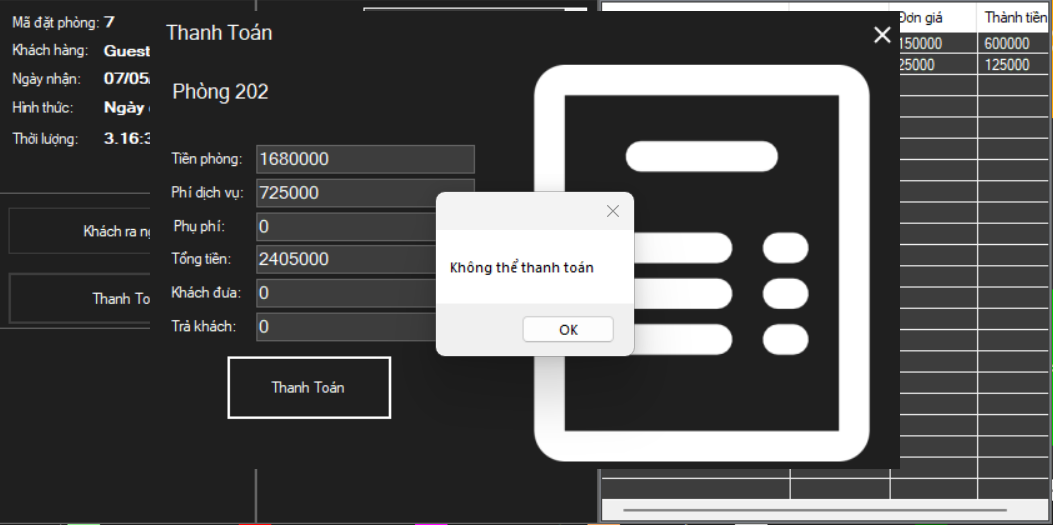
HÌnh 12. Thông báo thành công



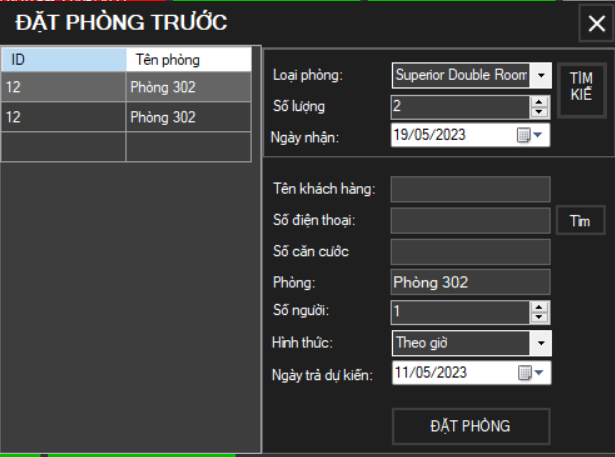
HÌnh 13. Giao diện phiếu dịch vụ



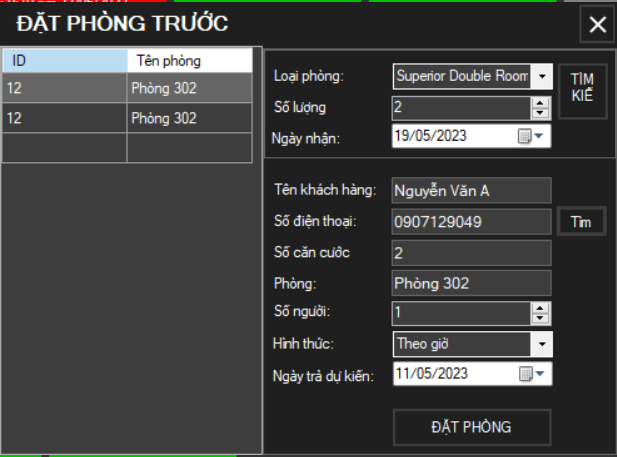
HÌnh 14. Form thanh toán



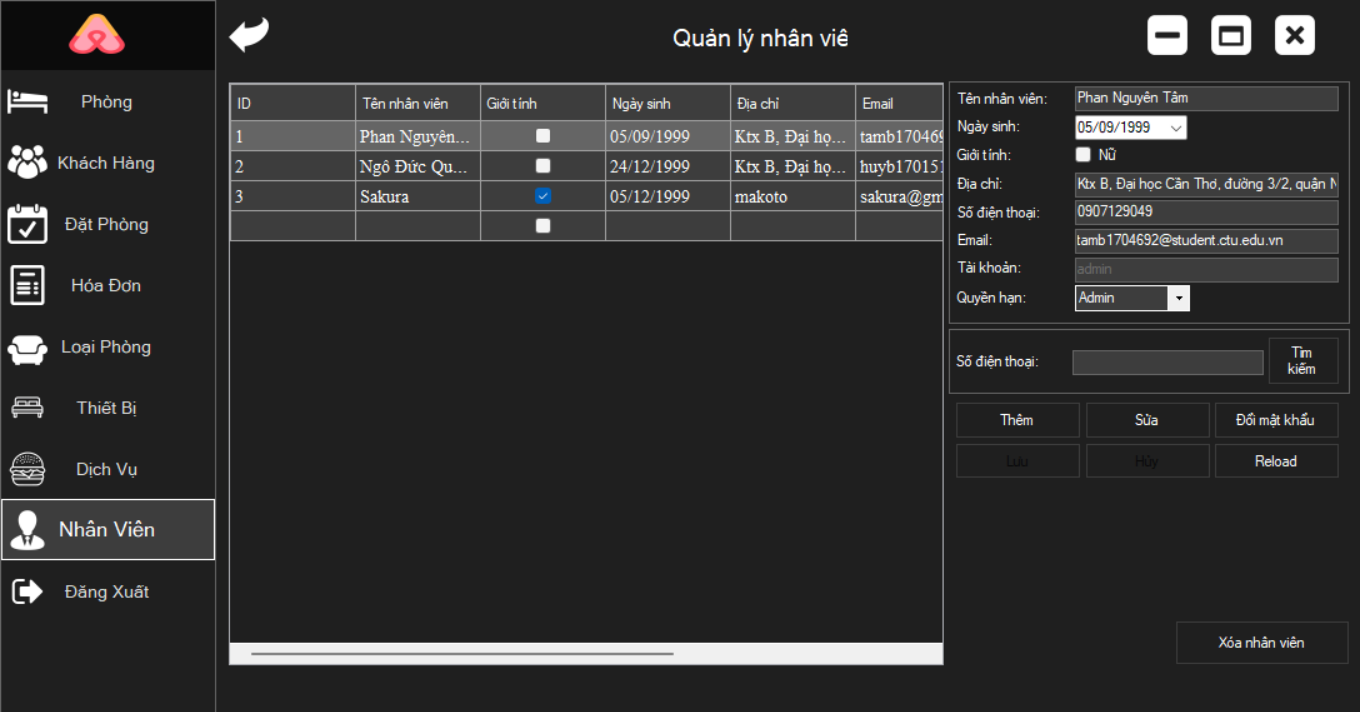
HÌnh 15. Thông báo thanh toán thành công



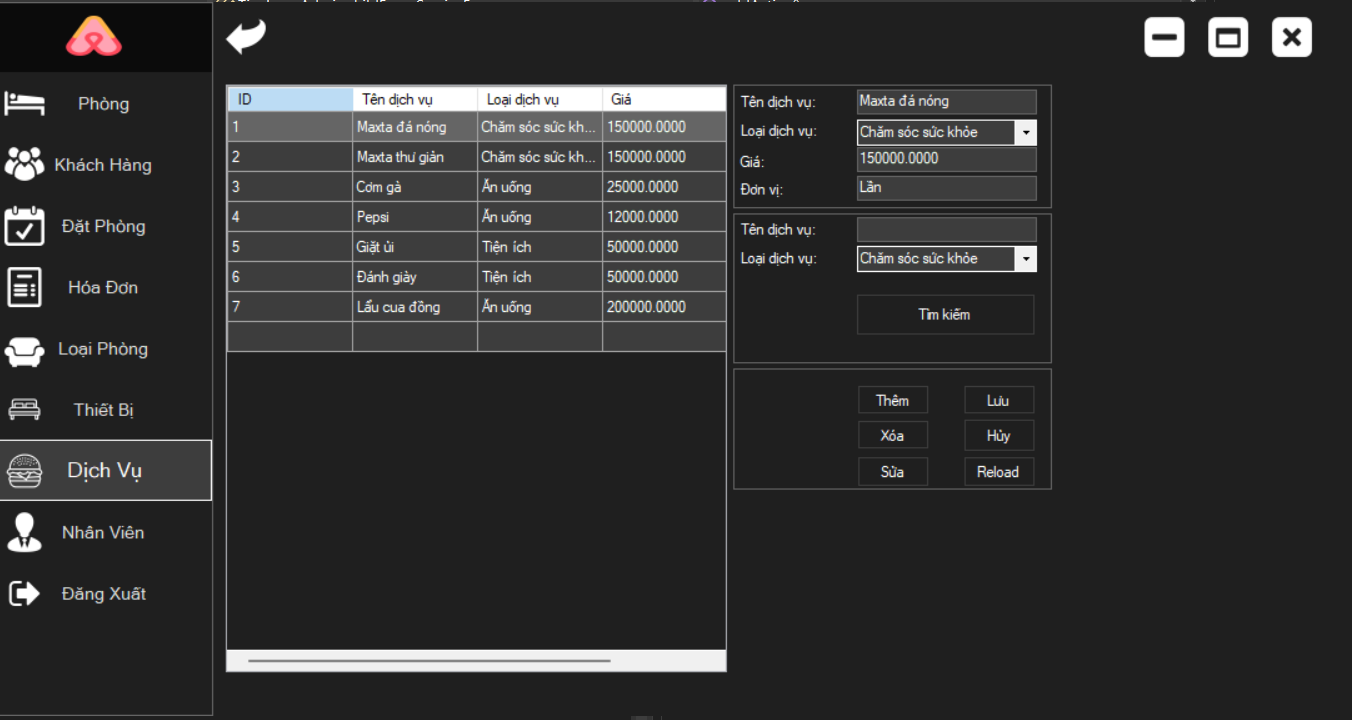
HÌnh 16. Giao diện đặt phòng trước



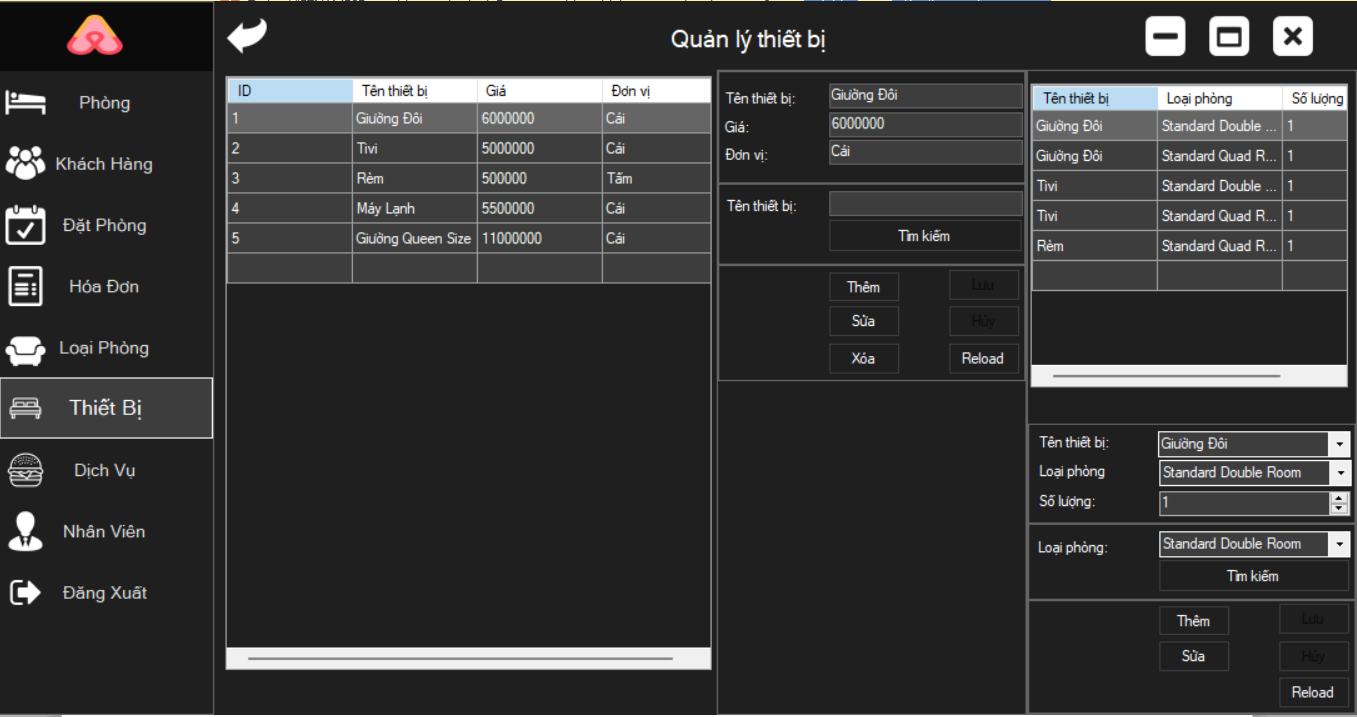
HÌnh 17. Giao diện đặt phòng trước



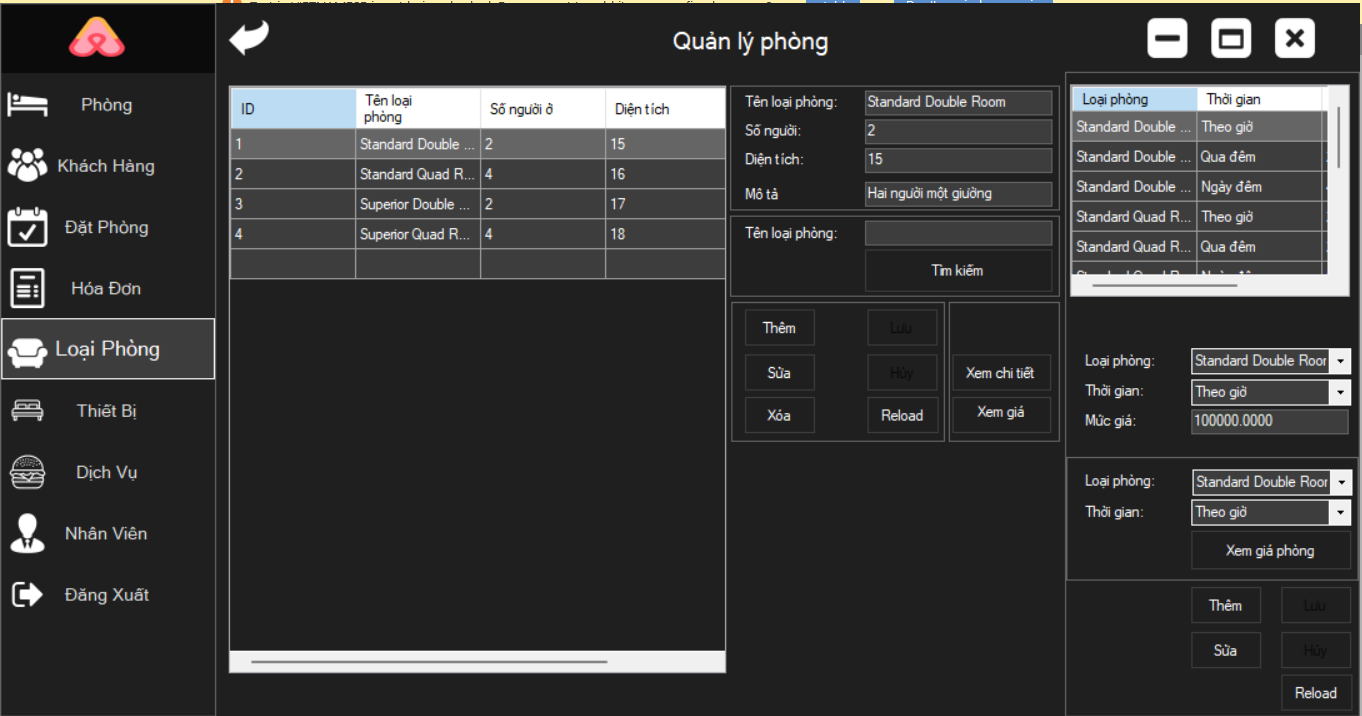
HÌnh 18. Giao diện quản lý nhân viên



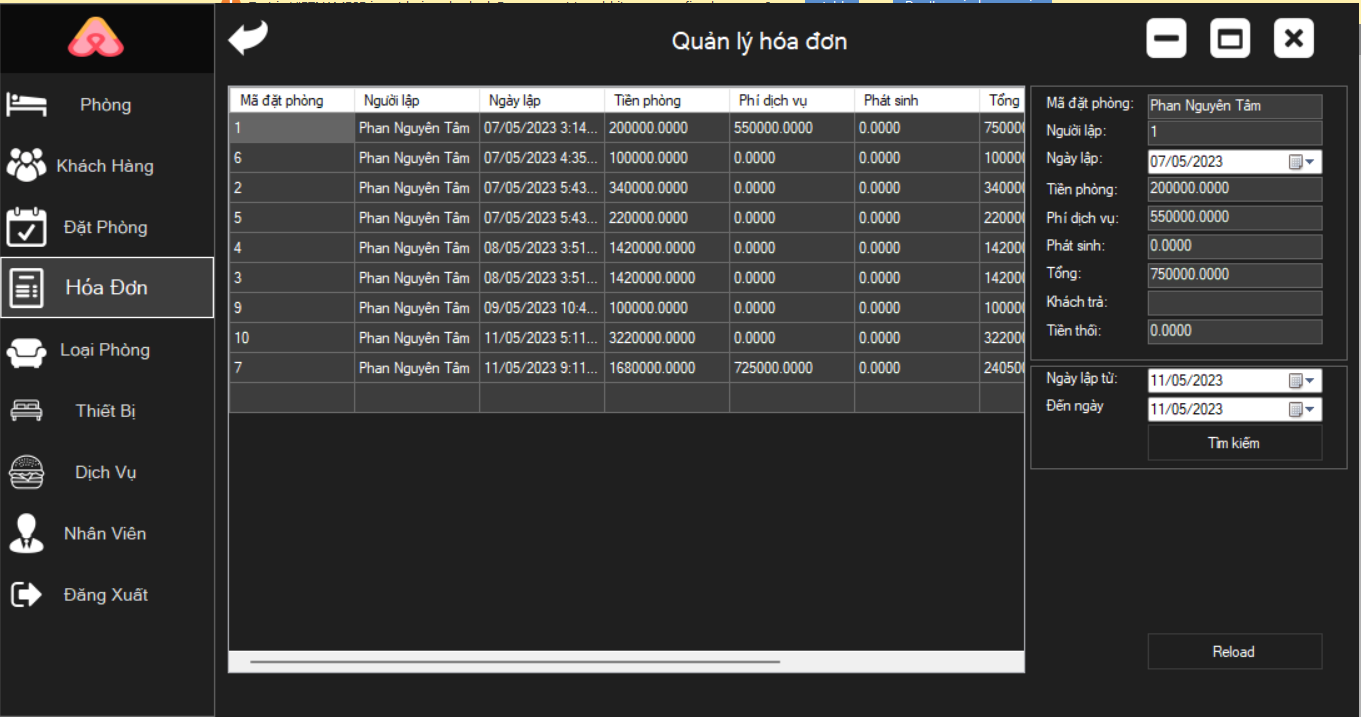
HÌnh 19. Giao diện quản lý dịch vụ



HÌnh 20. GIao diện quản lý thiết bị



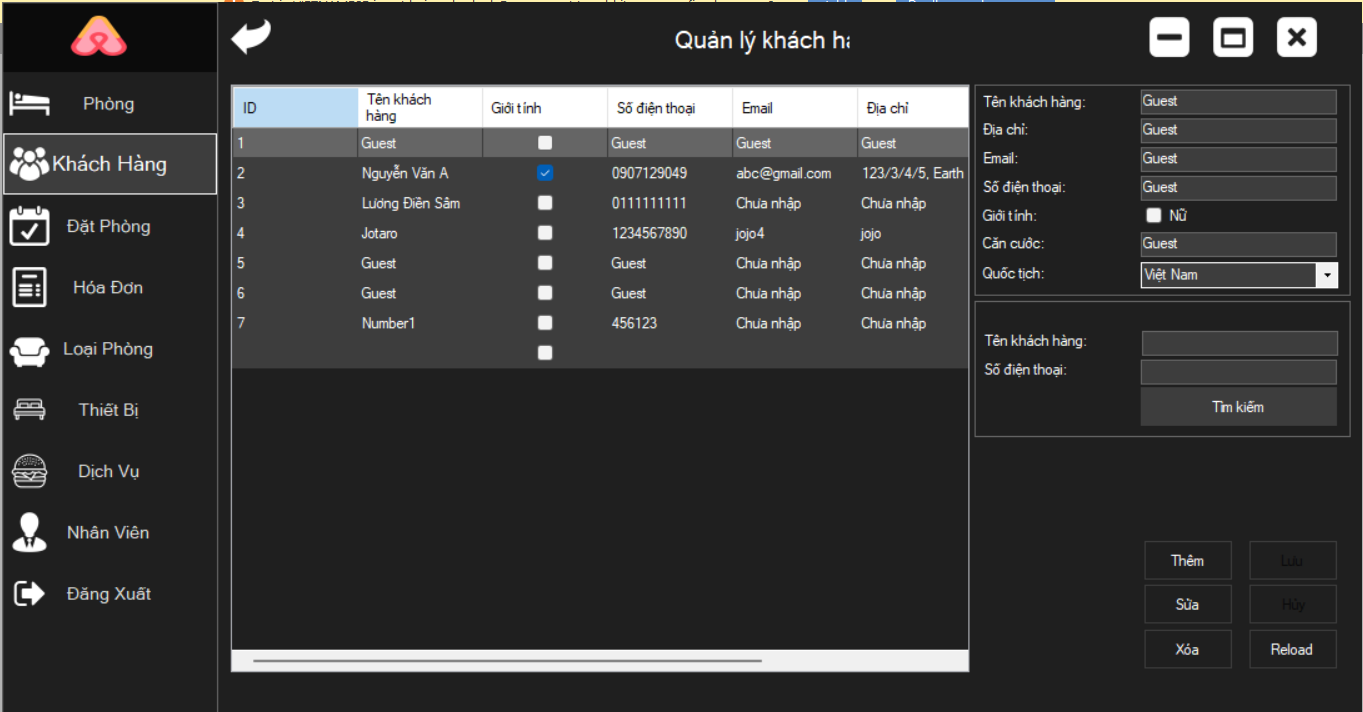
HÌnh 21. Giao diện quản lý loại phòng



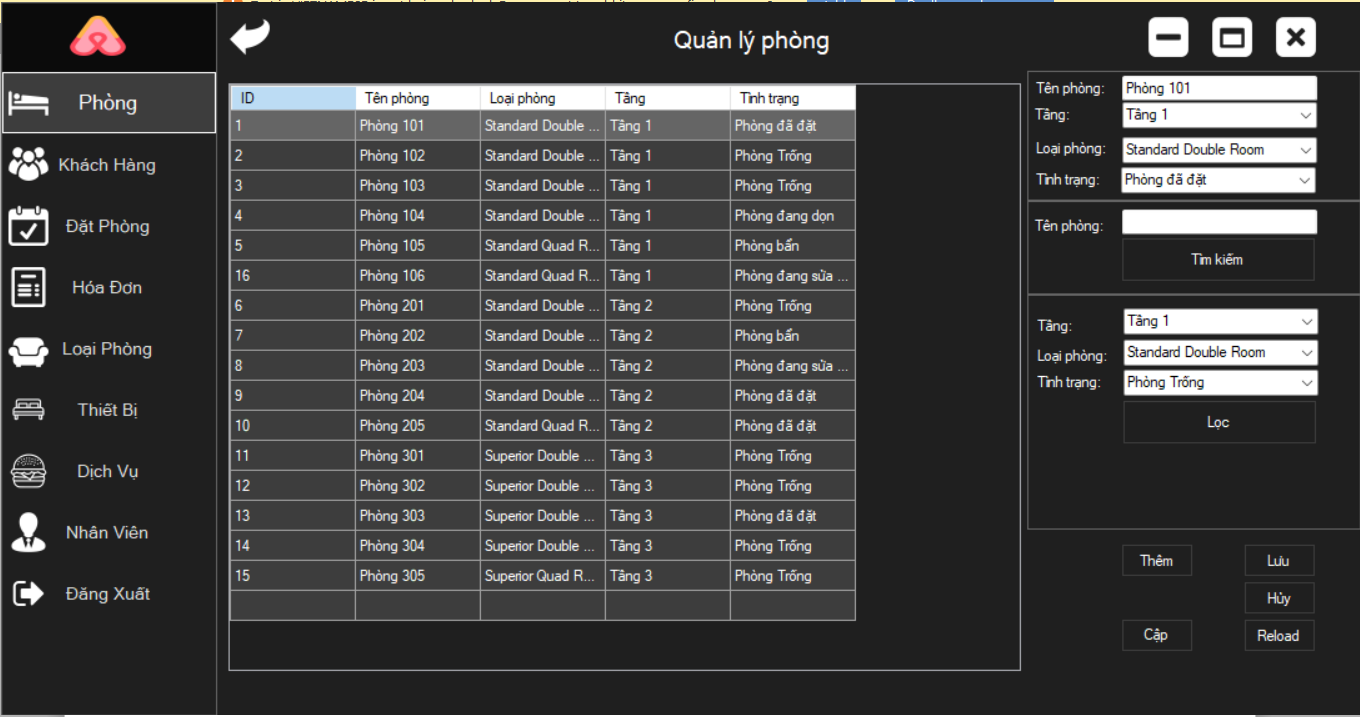
HÌnh 22. Giao diện quản lý hóa đơn



HÌnh 23. Giao diện quản lý đặt phòng



HÌnh 24. Giao diện quản lý khách hàng



HÌnh 25. Giao diện quản lý phòng

CHƯƠNG IV: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

**1. KẾT QUẢ ĐẠT ĐƯỢC**

- Biết được các yêu cầu cần thiết của một phần mềm quản lý khách sạn ,các chức năng cần thiết để phục vụ cho khách hàng

- Hệ thống tuy đáp ứng được nhu cầu cơ bản nhất nhưng vẫn còn vài chức năng chưa hoàn thiện cũng như một vài chưa hoạt động được.

**2. HẠN CHẾ**

- Giao diện chưa thân thiện với người dùng

- Chưa hỗ trợ xuất file exel và in đơn hàng PDF

- Chức năng về tính bảo mật độ an toàn chưa cao

**3. HƯỚNG PHÁT TRIỂN**

- Cải thiện giao diện người dùng

- Hoàn thành công cụ đọc và xuất file exel, tạo in file PDF

- Cải thiện và phát triển thê, các yêu cầu phi chức năng để tăng gia bảo mật và hiệu

suất