|  |
| --- |
| TRƯỜNG ĐẠI HỌC MỞ HÀ NỘI  **KHOA CÔNG NGHỆ THÔNG TIN**  ----------------------  Description: Macintosh HD:Users:xuantrinh:Dropbox:NCKH SV:BANG ROLL:logo.jpg  **BÁO CÁO BÀI TẬP LỚN**  **MÔN: LẬP TRÌNH HƯỚNG SỰ KIỆN**  **ĐỀ TÀI :** **Quản lý thuê phòng tại 1 khách sạn**  **Thành viên :**  Nguyễn Văn Đại - CHDN519  Tăng Thị Hoa - CNDN519  Nguyễn Hữu Khuê - CNDN519  Nguyễn Thị Yến Nhi - CNDN519    **Hà Nội – 2024** |

|  |  |  |
| --- | --- | --- |
| **Họ và tên** | **Nhiệm vụ** | **Thời Gian** |
| Nguyễn Văn Đại  Nguyễn Hữu Khuê | -Xây dựng chức năng Quản lý khách hàng  -Xây dựng chức năng Quản lý nhân viên | 05/08 - 09/08/2024 |
| Nguyễn Hữu Khuê | -Xây dựng chức năng Quản lý phòng  -Xây dựng cơ sở dữ liệu | 10/08 - 16/08/2024 |
| Tăng Thị Hoa | -Xây dựng chức năng thống kê báo cáo  -Viết báo cáo | 05/08 - 09/08/2024 |
| Nguyễn Thị Yến Nhi  Tăng Thị Hoa | -Xây dựng chức năng đăng ký đặt phòng  -Xây dựng chức năng Quản lý hóa đơn | 05/08 - 16/08/2024 |
| Nguyễn Văn Đại  Nguyễn Thị Yến Nhi | -Xây dựng chức năng thống kê báo cáo ,tìm kiếm  -Xây dựng chức năng Quản lý dịch vụ | 16/08 - 23/08/2024 |
| Cả Nhóm | -Test sản phẩm kiểm thử  -Viết báo cáo | 23/08 - 29/08/2024 |

**PHÂN CÔNG NHIỆM VỤ** QUẢN LÝ THUÊ PHÒNG TẠI 1 KHÁCH SẠN.

# MỞ ĐẦU

## Tổng Quan Về Đề Tài

### Giới thiệu chung khách sạn

**The Reverie Saigon Hotel** mới xây dựng theo tiêu chuẩn 3 sao, cạnh đường chính vào trung tâm thành phố. Chỉ vài phút đi bộ, du khách có thể đến các trung tâm triển lãm, khu mua sắm, và giải trí gần khách sạn. Là khách sạn duy nhất có bãi đáp trực thăng tại tầng thượng đủ điều kiện đáp ứng nhu cầu di chuyển bảo đảm tính riêng tư cho khách quan trọng, thương gia cao cấp.

**The Reverie Saigon Hotel** có tổng cộng 58 phòng ngủ với không gian ấm cúng, theo khuynh hướng cổ điển nhẹ nhàng và sang trọng, một số phòng còn có khăn tắm, dép, tivi màn hình phẳng, gương, bộ kim chỉ.    
Các dịch vụ khách sạn gồm wifi tất cả các phòng, quầy lễ tân 24 giờ, dịch vụ phòng 24 giờ, tiện nghi cho người khuyết tật, nhận/trả phòng nhanh, các hoạt động giải trí như bồn tắm nước nóng, phòng thể dục, hồ bơi ngoài trời, khu vực trị liệu, xoa bóp chăm sóc sắc đẹp theo phong cách Thái.  
Khách sạngồm các nhà hàng có sức chứa từ 50 đến 100 chỗ, với bầu không gian ấm cúng, đầu bếp có kinh nghiệm, đội ngũ nhân viên thanh lịch, luôn sẵn sàng phục vụ nhiều món ăn tinh hoa bốn phương, ăn sáng buffet, các món ăn Âu - Á và đặc biệt các món ăn mang đậm hương vị Việt Nam, giàu giá trị dinh dưỡng thỏa mãn gu ẩm thực tinh tế của khách.

### Khảo sát hiện trạng

Hiện tại khách sạn chưa có một chương trình chuyên quản lý việc đặt phòng và thanh toán mà chỉ dùng phần mềm Excel để quản lý, vậy nên đã gặp một số khó khăn :

* Các thủ tục đặt phòng cần xử lý sẽ mất nhiều thời gian.
* Gây khó khăn trong việc quản lý thông tin khách hàng, hóa đơn, phiếu đăng ký phòng cũng như các dịch vụ khác trong khách sạn.
* Khó có thể kiểm tra trạng thái phòng, các phòng cần được bảo trì.

Nhằm đáp ứng nhu cầu ứng dụng công nghệ thông tin cho các hoạt động kinh doanh và để giảm bớt thời gian tìm kiếm, theo dõi hoạt động của khách sạn trong việc quản lý. Chương trình “ Quản lý thuê phòng tại 1 khách sạn” ra đời nhằm đáp ứng các nhu cầu và vấn đề trên.

### Yêu cầu chức năng

Chương trình quản lý thuê phòng tại 1 khách sạn có các chức năng chính như sau:

* Quản lý dịch vụ
* Quản lý nhân viên
* Quản lý thuê phòng
* Quản lý chi tiết hóa đơn
* Quản lý khách sạn
* Tìm kiếm thông tin
* Báo cáo thống kê
* Quản lý thông tin phòng

### Mô tả bài toán:

Khách hàng có thể liên lạc đặt phòng tại khách sạn. Khi đặt phòng thông tin về khách

hàng phải được lưu lại trong bảng tblKhach. Đồng thời thông tin về đặt phòng cũng

phải được lưu lại trong bảng đăng ký ( khách hàng chỉ được đăng ký phòng trống). Khi

đến nhận, sử dụng phòng khách hàng có thể sử dụng thêm các dịch vụ khác của khách

sạn. Thông tin về sử dụng dịch vụ theo từng ngày sẽ phải được lưu lại. Khi trả phòng

tiền thanh toán sẽ phải được tính thêm phí sử dụng các dịch vụ này và các chi phí phát

sinh khác.

# CƠ SỞ LÝ THUYẾT

## I. Ngôn ngữ lập trình C#

### Giới thiệu

Ngôn ngữ C# là một ngôn ngữ được dẫn xuất từ C và C++, nhưng nó được tạo từ nền tảng phát triển hơn. Microsoft bắt đầu với công việc trong C và C++ và thêm vào những đặc tính mới để làm cho ngôn ngữ này dễ sử dụng hơn. Nhiều trong số những đặc tính này khá giống với những đặc tính có trong ngôn ngữ Java. Không dừng lại ở đó, Microsoft đưa ra một số mục đích khi xây dựng ngôn ngữ này. Những mục đích này được được tóm tắt như sau:

* C# là ngôn ngữ đơn giản
* C# là ngôn ngữ hiện đại
* C# là ngôn ngữ hướng đối tượng
* C# là ngôn ngữ mạnh mẽ và mềm dẻo
* C# là ngôn ngữ có ít từ khóa
* C# là ngôn ngữ hướng module
* C# sẽ trở nên phổ biến

### Các kiểu dữ liệu

Tương tự như C++ hay Java, C# chia thành hai tập hợp kiểu dữ liệu chính: Kiểu xây dựng sẵn (built- in) mà ngôn ngữ cung cấp cho người lập trình và kiểu được người dùng định nghĩa (user-defined) do người lập trình tạo ra

C# phân tập hợp kiểu dữ liệu này thành hai loại: Kiểu dữ liệu giá trị (value) và kiểu dữ liệu tham chiếu (reference). Việc phân chi này do sự khác nhau khi lưu kiểu dữ liệu giá trị và kiểu dữ liệu tham chiếu trong bộ nhớ. Đối với một kiểu dữ liệu giá trị thì sẽ được lưu giữ kích thước thật trong bộ nhớ đã cấp phát là stack. Trong khi đó kiểu dữ liệu tham chiếu như các đối tượng được cấp phát trên heap. Khi một đối tượng được cấp phát trên heap thì địa chỉ của nó được trả về, và địa chỉ này được gắn đến một tham chiếu.

**Bảng 1-1 Kiểu dữ liệu xây dựng sẵn**

|  |  |  |  |
| --- | --- | --- | --- |
| **Kiểu C#** | **Số byte** | **Kiểu .NET** | **Mô tả** |
| byte | 1 | Byte | Số nguyên dương không dấu từ 0-255 |
| char | 2 | Char | Kí tự Unicode |
| bool | 1 | Boolean | Giá trị logic true/ false |
| sbyte | 1 | Sbyte | Số nguyên có dấu ( từ -128 đến 127) |
| short | 2 | Int16 | Số nguyên có dấu giá trị từ -32768 đến 32767 |
| ushort | 2 | Int16 | Số nguyên không dấu 0 – 65.535 |
| int | 4 | Int32 | Số nguyên có dấu –2.147.483.647 và  2.147.483.647 |
| uint | 4 | Uint32 | Số nguyên không dấu 0 – 4.294.967.295 |
| float | 4 | Single | Kiểu dấu chấm động, giá trị xấp xỉ từ 3,4E-  38 đến 3,4E+38, với 7 chữ số có nghĩa.. |
| double | 8 | Double | Kiểu dấu chấm động có độ chính xác gấp  đôi, giá trị xấp xỉ từ 1,7E-308 đến 1,7E+308,  với 15,16 chữ số có nghĩa |
| decimal | 8 | Decimal | Có độ chính xác đến 28 con số và giá trị thập  phân, được dùng trong tính toán tài chính,  kiểu này đòi hỏi phải có hậu tố “m” hay “M”  theo sau giá trị. |
| long | 8 | Int64 | Kiểu số nguyên có dấu có giá trị trong  khoảng :  -9.223.370.036.854.775.808 đến  9.223.372.036.854.775.807 |
| ulong | 8 | Uint64 | Số nguyên không dấu từ 0 đến  0xffffffffffffffff |

**Chọn một kiểu định sẵn**

Tuỳ vào từng giá trị muốn lưu trữ mà ta chọn kiểu cho phù hợp. Nếu chọn kiểu quá lớn so với các giá trị cần lưu sẽ làm cho chương trình đòi hỏi nhiều bộ nhớ và chạy chậm. Trong khi nếu giá trị cần lưu lớn hơn kiểu thực lưu sẽ làm cho giá trị các biến bị sai và chương trình cho kết quả sai.

Kiểu char biểu diễn một ký tự Unicode. Ví dụ “\u0041” là ký tự “A” trên bảng Unicode. Một số ký tự đặc biệt được biểu diễn bằng dấu “\” trước một ký tự khác

**Kiểu dữ liệu do người dùng định nghĩa**

- Tất cả kiểu dữ liệu do người dùng định nghĩa ngoài trừ kiểu cấu trúc điều là kiểu dữ liệu tham chiếu

Một sô kiểu dữ liệu do người dùng định nghĩa gồm

* **object**: đây là kiểu dữ liệu cơ sở chứa tất cả các kiểu dữ liệu khác trong C#.
* **string**: kiểu dữ liệu chuỗi ký tự.
* **class**: kiểu dữ liệu class.
* **delegate**: kiểu dữ liệu chuyển giao.
* **interface**: kiểu dữ liệu giáo tiếp.
* **array**: kiểu dữ liệu mảng.

### Các toán tử

Toán tử được kí hiệu bằng một biểu tượng dùng để thực hiện một hành động. Các kiểu dữ liệu cơ bản của C# như kiểu nguyên hỗ trợ rất nhiều các toán tử như toán tử gán, toán tử toán học, logic....

Các phép toán +, -, \*, / là một ví dụ về toán tử. Áp dụng các toán tử này lên các biến kiểu số ta có kết quả như việc thực hiện các phép toán thông thường.

int a = 10;

int b = 20;

int c = a + b; // c = 10 + 20 = 30

C# cung cấp cấp nhiều loại toán tử khác nhau để thao tác trên các kiểu biến dữ liệu, được liệt kê trong bảng sau theo từng nhóm ngữ nghĩa.

**Bảng 1.2 Các nhóm toán tử trong C#**

|  |  |  |
| --- | --- | --- |
| **Nhóm toán tử** | **Toán tử** | **Ý nghĩa** |
| Toán học | + - \* / % | Cộng, trừ, nhân, chia, chia lấy phần dư |
| Logic | & | ^ ! ~ && true false | Phép toàn logic và thao tác trên bit |
| Ghép chuỗi | + | Ghép nối hai chuỗi |
| Tăng, giảm | ++, -- | Tăng/giảm toán hạng lên/xuống 1. Đứng trước hoặc sau toán hạng |
| Dịch bit | << >> | Dịch trái, dịch phải |
| Quan hệ | == != < > <= >= | Bằng, khác, nhỏ hơn, lớn hơn, nhỏ hơn hoặc bằng, lớn hơn hoạc bằng |
| Gắn | = += -= \*= /= %= &= != ^ = <<= >>= | Phép gắn |
| Chỉ số | [] | Cách truy xuất phần tử của mảng |
| Ép kiểu | () |  |
| Indirecti0on và Address | \* -> [] & | Dùng cho con trỏ |

**3.1 Toán tử gán (=)**

Toán tử gán (=) Toán tử này cho phép thay đổi các giá trị của biến bên phải toán tử bằng giá trị bên trái toán tử.

**3.2 Nhóm toán tử toán học**

C# dùng các toàn tử số học với ý nghĩa theo đúng tên của chúng như: + (cộng), – (trừ) \* (nhân) và / (chia). Tùy theo kiểu của hai toán hạng mà toán tử trả về kiểu tương ứng. Ngoài ra, còn có toán tử % (lấy phần dư) được sử dụng trong các kiểu số nguyên.

**3.3 Các toán tử tăng và giảm**

C# cũng kế thừa từ C++ và Java các toán tử: +=,-=, \*=, /= , %= nhằm làm đơn giản hoá. Nó còn kế thừa các toán tử tiền tố và hậu tố (như biến++, hay ++biến) để giảm bớt sự cồng kềnh trong các toán tử cổ điển.

**3.4 Các toán tử quan hệ**

Các toán tử quan hệ được dùng để so sánh hai giá trị với nhau và kết quả trả về có kiểu Boolean. Toán tử quan hệ gồm có: == (so sánh bằng), != (so sánh khác), > (so sánh lớn hơn), >= (lớn hơn hay bằng), < (so sánh nhỏ hơn), <= (nhỏ hơn hay bằng).

**3.5 Các toán tử logic**

Các toán tử logic gồm có: && (và), || (hoặc), ! (phủ định). Các toán tử này được dùng trong các biểu thức điều kiện để kết hợp các toán tử quan hệ theo một ý nghĩa nhất định.

**3.6 Thứ tự các toán tử**

Đối với các biểu thức toán, thứ tự ưu tiên là thứ tự được qui định trong toán học. Còn thứ tự ưu tiên thực hiện của các nhóm toán tử được liệt kê theo bảng dưới đây

**Bảng 1-3 Thứ tự ưu tiên của các nhóm toán tử (chiều ưu tiên từ trên xuống)**

|  |  |  |
| --- | --- | --- |
| **Nhóm toán tử** | **Toán tử** | **Ý nghĩa** |
| Primary(chính) | {x} x.y f(x) a[x] x++ x-- |  |
| Unary | + - ! ~ ++x –x (T)x |  |
| Nhân | \* / % | Nhân, chia, chia lấy phần dư |
| Cộng | + - | công, trừ |
| Dịch bit | << >> | Dịch trái dịch phải |
| Bằng | == != | Bằng, khác |
| Logic trên bit AND | & | Và trên bit |
| XOR | ^ | Xor trên bit |
| OR | | | Hoặc trên bit |
| Điều kiện AND | && | Và trên biểu thức điều kiện |
| Điều kiện OR | || | Hoặc trên biểu thức điều kiện |
| Điều kiện | ?: | Điều kiện tượng tự if |
| Asignment | = \*= /= %= += -= <<= =>> &= ^= |= |  |

**3.7 Toán tử tam phân**

Cú pháp:

<biểu thức điều kiện>? <biểu thức 1>: <biểu thức 2>;

Ý nghĩa:

* Nếu biểu thức điều kiện đúng thì thực hiện biểu thức 1.
* Nếu sai thì thực hiện biểu thức 2.

### Cấu trúc rẽ nhánh

a. Câu lệnh if… else …

Cú pháp:

if ( biểu thức logic )

khối lệnh;

hoặc

if ( biểu thức logic )

khối lệnh 1;

else

khối lệnh 2;

*Ghi chú: Khối lệnh là một tập các câu lện trong cặp dấu “{…}”. Bất kỳ**nơi đâu có câu lệnh thì ở đó có thể viết bằng một khối lệnh.*

Biểu thức logic là biểu thức cho giá trị dúng hoặc sai (***true*** hoặc ***false***). Nếu “*biểu thức logic*” cho giá trị đúng thì “*khối lệnh*” hay “*khối lệnh 1*” sẽ được thực thi,ngược lạ**i** “*khối lệnh 2*” sẽ thực thi. Một điểm khác biệt với C++ là biểu thức trongcâu lệnh if phải là biểu thức logic, không thể là biểu thức số.

b. Câu lệnh switch case

Cú pháp:

switch ( biểu\_thức\_lựa\_chọn )

{

case biểu\_thức\_hằng :

khối lệnh;

lệnh nhảy;

[ default :

khối lệnh;

lệnh nhảy; ]

}

Biểu thức lựa chọn là biểu thức sinh ra trị nguyên hay chuỗi. Switch sẽ so sánh biểu\_thức\_lựa\_chọn với các biểu\_thức\_hằng để biết phải thực hiện với khối lệnh nào. Lệnh nhảy như break, goto…để thoát khỏi câu switch và bắt buộc phải có.

int nQuyen = 0;

switch ( sQuyenTruyCap )

{

case “Administrator”:

nQuyen = 1;

break;

case “Admin”:

goto case “Administrator”;

default:

nQuyen = 2;

break;

}

c. Câu lệnh for

Cú pháp:

for ( [khởi\_tạo\_biến\_đếm]; [biểu\_thức]; [gia\_tăng\_biến\_đếm] )

khối lệnh;

Ví dụ 3-4 Tính tổng các số nguyên từ a đến b

int a = 10; int b = 100; int nTong = 0;

d. Câu lệnh while

Cú pháp:

while ( biểu\_thức\_logic )

khối\_lệnh;

Khối\_lệnh sẽ được thực hiện cho đến khi nào biểu thức còn đúng. Nếu ngay từ đầu biểu thức sai, khối lệnh sẽ không được thực thi.

e. Câu lệnh do …while

Cú pháp:

do

khối\_lệnh

while ( biếu\_thức\_logic )

Khác với while khối lệnh sẽ được thực hiện trước, sau đó biệu thức được kiểm tra. Nếu biểu thức đúng khối lệnh lại được thực hiện.

## SQL Server

### Định nghĩa SQL sever là gì?

SQL server hay còn được gọi là Microsoft SQL Server, nó từ viết tắt của MS SQL Server. Đây chính là một loại phần mềm đã được phát triển bởi Microsoft và nó được sử dụng để có thể dễ dàng lưu trữ cho những dữ liệu dựa theo tiêu chuẩn RDBMS.

Từ đó, người ta sẽ lưu trữ dữ liệu dựa vào tiêu chuẩn RDBMS và nó cũng là một  trong những hệ quản trị cơ sở dữ liệu dạng quan hệ đối tượng.



*SQL Sever là gì?*

SQL Server có khả năng cung cấp đầy đủ các công cụ cho việc quản lý từ giao diện GUI đến sử dụng ngôn ngữ cho việc truy vấn SQL. Điểm mạnh của SQL điểm mạnh của nó là có nhiều nền tảng được kết hợp cùng như: ASP.NET, C# để xây dựng Winform cũng chính nó có khả năng hoạt động độc lập. Tuy nhiên, SQL Server thường đi kèm với việc thực hiện riêng các ngôn ngữ SQL, T-SQL,...

Cụ thể như sau:

* T-SQL là một trong những loại ngôn ngữ thuộc quyền sở hữu của Microsoft và được gọi với cái tên Transact-SQL. Nó thường cung cấp thêm rất nhiều cho các  khả năng khai báo biến, thủ tục lưu trữ và xử lý ngoại lệ,...
* SQL Server Management Studio là một loại công cụ giao diện chính cho máy chủ cơ sở của chính dữ liệu SQL, thông thường thì nó hỗ trợ cho cả môi trường 64 bit và 32 bit.

### Những thành phần của SQL Server là gì?

SQL Server đã trải qua hơn 20 năm phát triển và đã đề ra những version cụ thể khác nhau. Các mô hình Client - Server được chia làm 2 thành phần chính bao gồm:

* Workstation: Nó được cài đặt trên các thiết bị vận hành để trở thành phần mềm tương tác với hệ thống máy chủ Server.
* Server: Được cài đặt ở trên máy chủ chính, nó có thể là: SQL server, SQL Server Agent, SSIS, SSAS,...

Ngoài ra, bạn hoàn toàn có thể cài đặt nhiều phiên bản của SQL Server trên cùng một máy chủ và điều này sẽ giúp tiết kiệm được các chi phí mua Server để hoạt động và cần nhiều phiên bản khác nhau. Nó bảo mật và cũng được tách biệt hoàn toàn giúp cho hệ thống được an toàn hơn.

### Mục đích khi sử dụng SQL Server là gì?

SQL Server thông thường được sử dụng cho mục đích lưu trữ dữ liệu. Ngoài ra, nó còn mang lại những tính năng làm việc giúp người dùng làm việc hiệu quả hơn như sau:

* Giúp người sử dụng có thể duy trì việc lưu trữ bền vững.
* Cho phép bạn tạo ra nhiều cơ sở dữ liệu hơn.
* Có khả năng phân tích dữ liệu bằng SSAS
* Nó có khả năng bảo mật cao
* Việc tạo ra được các báo cáo bằng SSRS — SQL Server Reporting Services sẽ được dễ dàng hơn.
* Các quá trình sẽ được thực hiện bằng SSIS — SQL Server Integration Services.

4.Một số khái niệm cơ bản về các xử lý

* Thông thường, các xử lý bên trong một ứng dụng thường có thể sẽ được chia thành 2 loại xử lý, bao gồm: xử lý trên máy chủ và xử lý trên trạm.
* Nó có khả năng đọc và cập nhật được các dữ liệu.
* Quá trình tính toán, hiển thị dữ liệu ở trên màn hình giao diện.
* Bạn có thể sử dụng được nhiều loại ngôn ngữ lập trình khác nhau như là: C#, Java,...
* Xử lý các tiêu chuẩn trên máy chủ Database Server
* Xử lý những yêu cầu liên quan đến ghi/đọc dữ liệu.
* Thực hiện quản lý đồng bộ mọi dữ liệu giữa những yêu cầu đọc ghi từ các máy trạm gửi tới.
* Thực hiện các dịch vụ quản trị dữ liệu tự động dựa theo định kỳ như backup/restore dữ liệu.



*SQL Server được sử dụng để lưu trữ dữ liệu*

Dưới đây là một số lý do cho phép người sử dụng ưa chuộng sử dụng SQL để thực hiện đa dạng các chức năng như sau:

* Nó cho phép người sử dụng có thể dùng để truy cập dữ liệu bên trong hệ thống quản lý cơ sở dữ liệu quan hệ.
* Người ta cho phép người dùng mô tả dữ liệu.
* Cho phép người dùng thực hiện xác định dữ liệu bên trong cơ sở dữ liệu và thực hiện các thao tác dữ liệu.
* Nó cho phép nhúng trong các ngôn ngữ khác có thể sử dụng mô-đun SQL, thư viện và thực hiện trình biên dịch trước.
* Nó sẽ cho phép người dùng tạo và thả cho các cơ sở dữ liệu cũng như bảng.
* SQL cho phép người sử dụng để thực hiện tạo ra các chế độ view, các thủ tục lưu trữ và chức năng trong cơ sở dữ liệu.
* Nó sẽ cho phép người dùng để thực hiện thiết lập quyền trên các bảng, view và thủ tục.

### Các phiên bản SQL Server là gì?

Trong  khoảng từ năm 1995 đến năm 2916 thì Microsoft đã cho phát hành nhiều phiên bản cơ sở dữ liệu SQL. Ngoài ra, Microsoft đã kết hợp được với nhiều  công cụ cho việc thực hiện quản lý dữ liệu cũng như phân tích được dữ liệu vào SQL Server. Một số chức năng và công nghệ mới được xuất hiện bao gồm: web, [điện toán đám mây](https://itnavi.com.vn/blog/cloud-computing-la-gi/" \t "_blank) và các thiết bị di động.

***Phiên bản 1: SQL Server 2012***

Bản 2012 được cung cấp các tính năng mới như chỉ mục cột, có thể được sử dụng để thực hiện các lưu trữ theo hướng định dạng trên cột dành cho các ứng dụng. Ngoài ra, việc phân tích dữ liệu luôn được sẵn sàng và trang bị công nghệ để có thể khắc phục các thảm họa.

****

*Phiên bản Sql Server 2012*

**Phiên bản 2: SQL Server 2014**SQL 2014 đã được thêm OLTP trong bộ nhớ nên người dùng có thể thực hiện chạy cho các ứng dụng xử lý giao dịch trực tuyến. Quá trình thực hiện sẽ dựa trên dữ liệu lưu trữ bên trong những bảng đã được tối ưu hóa cho bộ nhớ thay vì phải sử dụng các tệp dựa trên bộ đĩa tiêu chuẩn.

Một trong những tính năng khác của bản 2014 sở hữu phần mở rộng nhóm bộ đệm và được tích hợp thêm bộ nhớ ở vùng đệm cho máy chủ SQL nhờ vào ổ đĩa có trạng thái rắn cũng như được thiết kế với thông lượng I/O nhờ việc giảm tải đi các dữ liệu từ những đĩa cứng có dạng thông thường khác.

Bạn đọc tham khảo thêm : [Tìm việc làm IT](https://itnavi.com.vn/job" \t "_blank) lương cao chế độ hấp dẫn nhất tại ITNAVI

**Phiên bản 3: Microsoft SQL Server 2016**Microsoft của máy chủ SQL 2016 nó đã có sẵn từ tháng 6/2016. Nó được phát triển như một phần của các chiến lược công nghệ đầu tiên đối với một thiết bị di động đầu tiên trên nền tảng đám mây (nó được Microsoft áp dụng vào khoảng 2 năm trước đó).

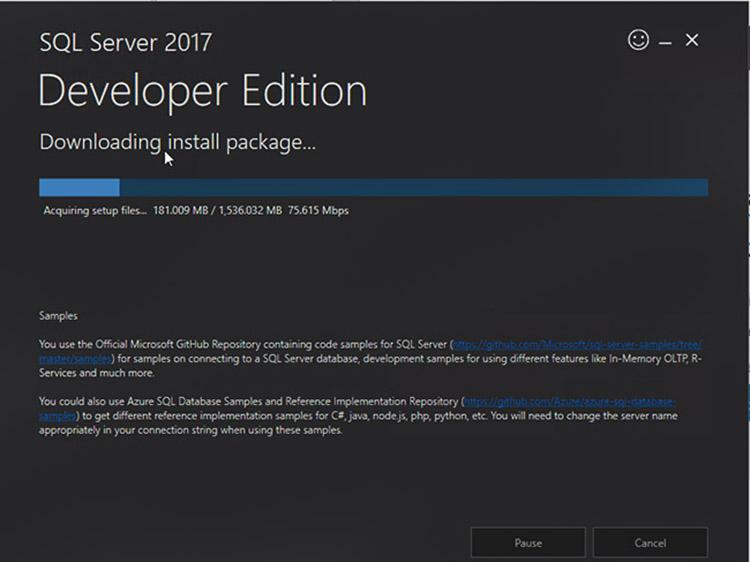
Phiên bản này cũng có thêm nhiều tính năng bao gồm sự điều chỉnh hiệu suất, phân tích hoạt động cho thời gian thực, sự hỗ trợ của đám mây.

Từ đó, cho phép DBA chạy dựa trên cơ sở dữ liệu được kết hợp hệ thống tại chỗ và dịch vụ đám mây có khả năng giảm thiểu được các chi phí CNTT. SQL Server 2016 có khả năng tăng hỗ trợ cho việc phân tích luồng dữ liệu lớn và ứng dụng nhiều trong việc phân tích các ứng dụng nâng cao khác thông qua hệ thống máy chủ cơ sở dữ liệu SQL R Services.

Nó còn cho phép DBMS chạy ứng dụng phân tích được viết bằng loại ngôn ngữ lập trình R nguồn mở và polyBase. Công nghệ này cho phép người sử dụng máy chủ SQL truy cập dữ liệu trong cụm Hadoop hoặc lưu trữ  blob để thực hiện phân tích.

**Phiên bản 4: SQL Server 2017 :** Bản cập nhật chính thức và đổi mới được phát hành vào tháng 10 năm 2017. Việc hỗ trợ cho máy chủ SQL trên Linux đã chuyển nền tảng cho cơ sở dữ liệu lên một hệ điều hành nguồn mở thường thấy ở trong các doanh nghiệp.

Từ đó, nó mang lại tiềm năng cho Microsoft với những khách hàng không sử dụng Windows hoặc ở trong môi trường máy chủ hỗn hợp.



*Phiên bản SQL Server 2017*

SQL Server 2017 đã được mở rộng để hỗ trợ Docker được thêm các hệ thống Windows dựa trên phiên bản trước để bao gồm các thùng chứa dựa trên Linux. SQL Server 2017 hỗ trợ lập trình Python, vì là ngôn ngữ mở nguồn và được sử dụng tương đối rộng rãi trong các ứng dụng phân tích.

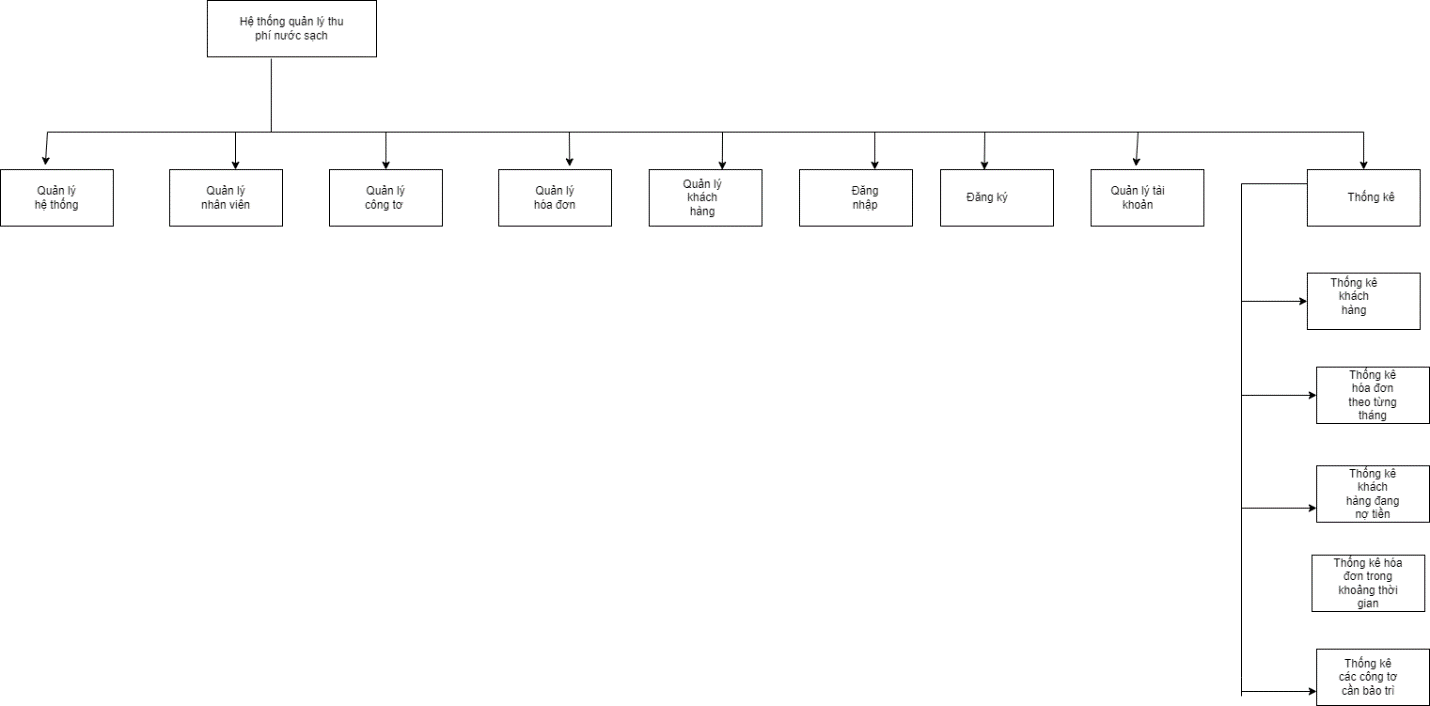
SQL Server R Services được đổi tên thành Machine Learning Services và được mở rộng để thực hiện chạy cho cả ứng dụng R và Python. Ban đầu thì bộ công cụ máy và một loạt tính năng khác chỉ có trong các phiên bản Windows của phần mềm cơ sở dữ liệu với các tính năng hạn chế hơn được hỗ trợ trên Linux.

**Phiên bản 5: SQL Server 2019**

Bản 2019 cho phép người được sử dụng để tham gia vào các thùng chứa SQL Server, HDFS và Spark cùng nhau bằng nhiều tính năng. Ngoài ra, nó còn giới thiệu cho việc xây dựng chỉ mục cột, xây dựng lại cũng như che giấu đi dữ liệu tĩnh. Từ đó, phục hồi dữ liệu tăng tốc mới và thực hiện, hoàn tác các giai đoạn làm lại số thứ tự nhật ký trang.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Sơ đồ phân cấp chức năng



## Thiết Kế Csdl Mức Logic

### Xác định các kiểu thực thể và thuộc tính

1. **Thực thể: tblDangky (Đăng ký)**
   * **Thuộc tính:**
     + MaDangKy (Khóa chính)
     + MaKhachHang
     + MaNhanVien
     + MaPhong
     + DonGia
     + NgayDangKy
     + NgayNhanPhong
     + NgayTraPhong
     + TongTien
     + Trangthai
2. **Thực thể: tblDichvu (Dịch vụ)**
   * **Thuộc tính:**
     + MaDichVu (Khóa chính)
     + TenDichVu
     + GiaDichVu
3. **Thực thể: tblHoadonchitiet (Hóa đơn chi tiết)**
   * **Thuộc tính:**
     + MaDangKy (Khóa ngoại tham chiếu đến tblDangky)
     + MaDichVu (Khóa ngoại tham chiếu đến tblDichvu)
     + NgaySuDung
     + GiaDichVu
4. **Thực thể: tblKhach (Khách hàng)**
   * **Thuộc tính:**
     + MaKhachHang (Khóa chính)
     + CMND
     + HoTen
     + NgaySinh
     + GioiTinh
     + SoDienThoai
5. **Thực thể: tblNhanvien (Nhân viên)**
   * **Thuộc tính:**
     + MaNhanVien (Khóa chính)
     + HoTen
     + GioiTinh
     + ChucVu
6. **Thực thể: tblPhong (Phòng)**
   * **Thuộc tính:**
     + SoPhong (Khóa chính)
     + SucChua
     + DonGia
     + TrangThai

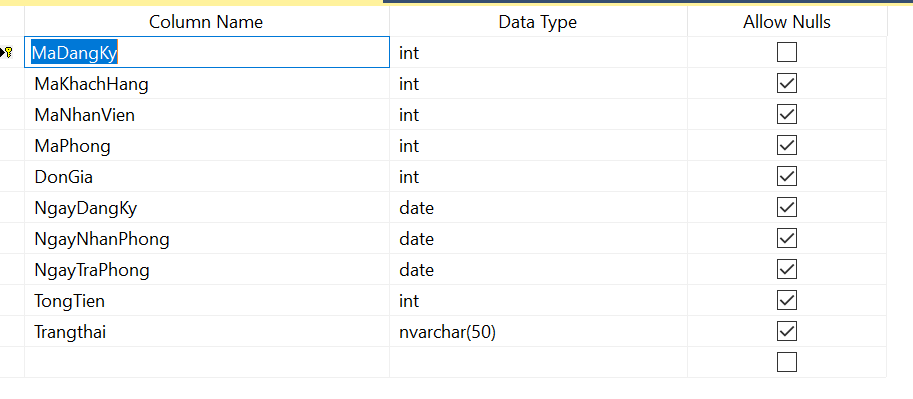
### Xác định liên kết giữa các kiểu thực thể:

1. Liên kết 1-N:
   * Bảng tblDangky có cột MaKhachHang (Mã Khách Hàng) liên kết với cột MaKhachHang trong bảng tblKhach thông qua khóa ngoại. Mối quan hệ 1-N này thể hiện mỗi lượt đăng ký thuê phòng có thể thuộc về một khách hàng duy nhất, trong khi một khách hàng có thể có nhiều lần đăng ký thuê phòng.
   * Bảng tblDangky cũng có cột MaNhanVien (Mã Nhân Viên) liên kết với cột MaNhanVien trong bảng tblNhanvien qua khóa ngoại. Mối quan hệ này biểu diễn mỗi lần đăng ký thuê phòng được thực hiện bởi một nhân viên, nhưng một nhân viên có thể phụ trách nhiều lần đăng ký thuê phòng cho khách hàng.
   * Bảng tblDangky có cột MaPhong (Mã Phòng) liên kết với cột SoPhong trong bảng tblPhong qua khóa ngoại. Mối quan hệ này thể hiện mỗi đăng ký đặt phòng chỉ thuộc về một phòng cụ thể, trong khi một phòng có thể có nhiều lượt đăng ký.
2. Liên kết N-N:
   * Bảng tblHoadonchitiet có cột MaDangKy liên kết với cột MaDangKy trong bảng tblDangky qua khóa ngoại. Đây là mối quan hệ N-N vì mỗi hóa đơn chi tiết có thể liên kết với nhiều lần đăng ký và mỗi lần đăng ký có thể có nhiều hóa đơn chi tiết.
   * Bảng tblHoadonchitiet cũng có cột MaDichVu liên kết với cột MaDichVu trong bảng tblDichvu qua khóa ngoại. Mối quan hệ này biểu diễn mỗi hóa đơn chi tiết liên quan đến một dịch vụ cụ thể và mỗi dịch vụ có thể xuất hiện trong nhiều hóa đơn chi tiết.
3. Liên kết N-1:
   * Bảng tblHoadonchitiet có cột MaDangKy liên kết với cột MaDangKy trong bảng tblDangky qua khóa ngoại. Mối quan hệ này thể hiện mỗi hóa đơn chi tiết chỉ thuộc về một đăng ký cụ thể, nhưng một đăng ký có thể có nhiều hóa đơn chi tiết.
   * Bảng tblDangky có cột MaKhachHang, MaNhanVien, và MaPhong liên kết với các cột tương ứng trong các bảng tblKhach, tblNhanvien, và tblPhong qua khóa ngoại. Mối quan hệ này thể hiện mỗi đăng ký liên quan đến một khách hàng, một nhân viên, và một phòng cụ thể, nhưng mỗi khách hàng, nhân viên, và phòng có thể xuất hiện trong nhiều đăng ký.

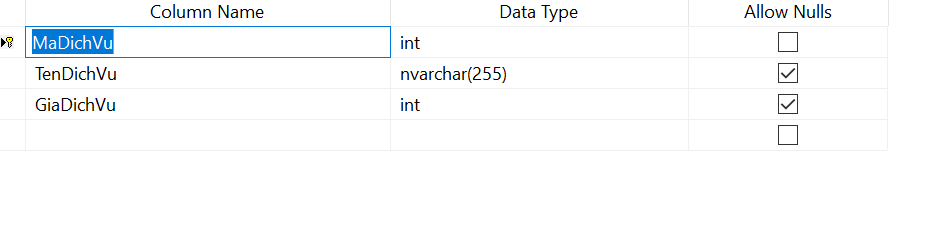
## Mô Hình Cơ Sở Dữ Liệu Quan Hệ

### Các bảng trong cơ sở dữ liệu

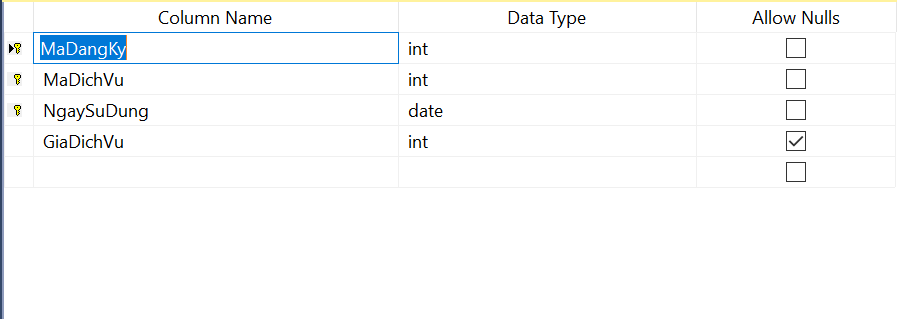
1. tblDangky



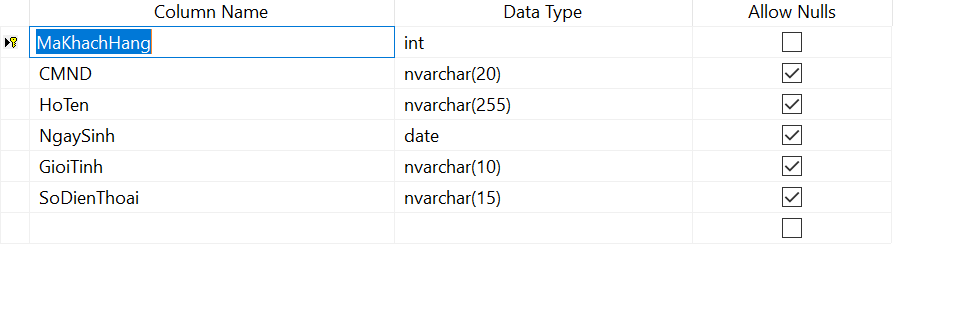
1. tblDichvu



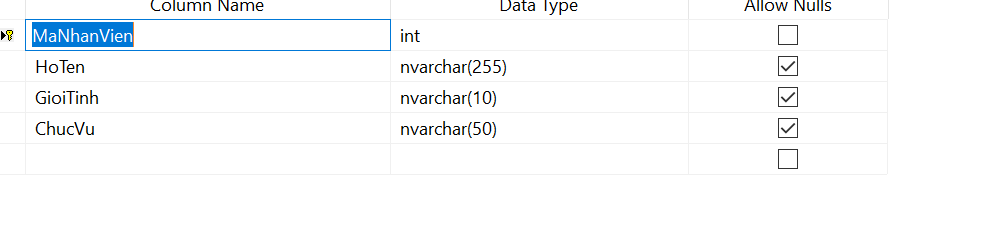
1. tblHoadonchitiet



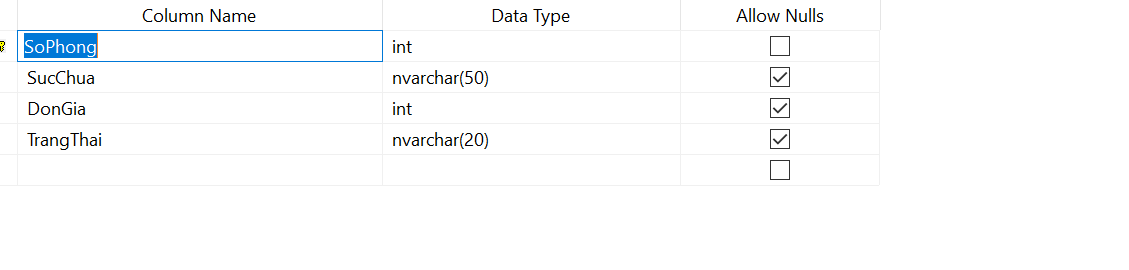
1. tblKhach



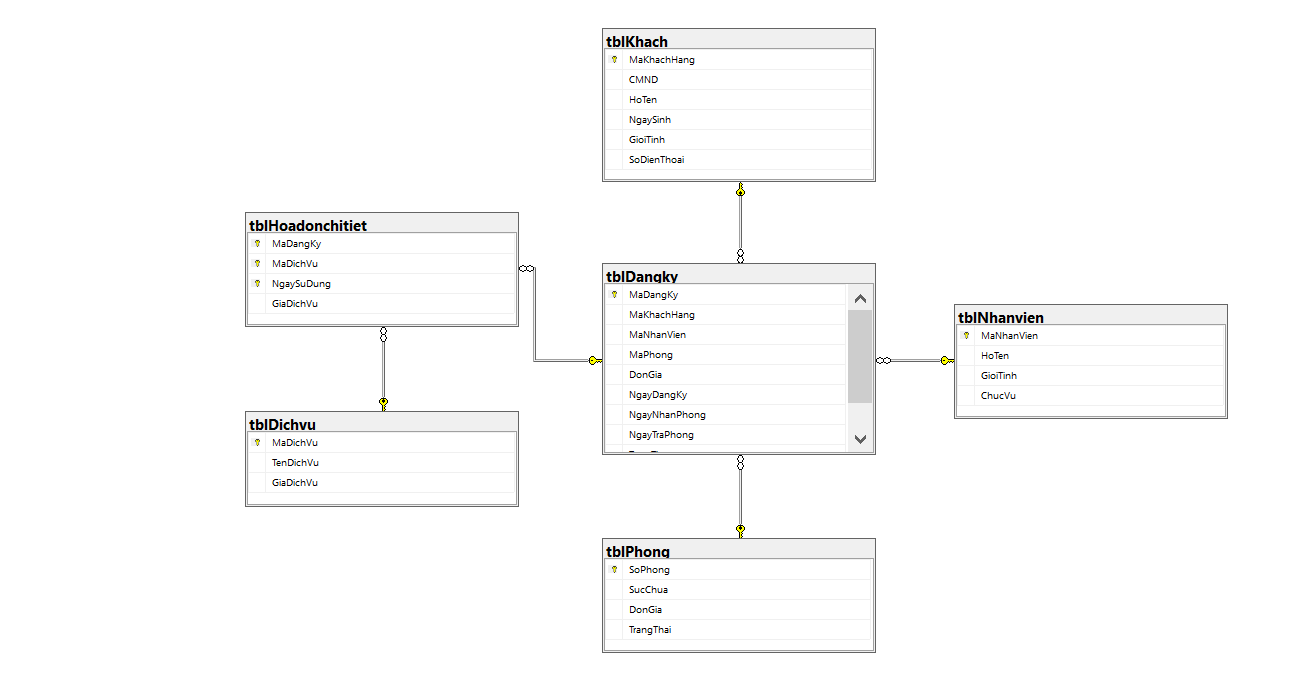
1. tblNhanvien



1. tblPhong

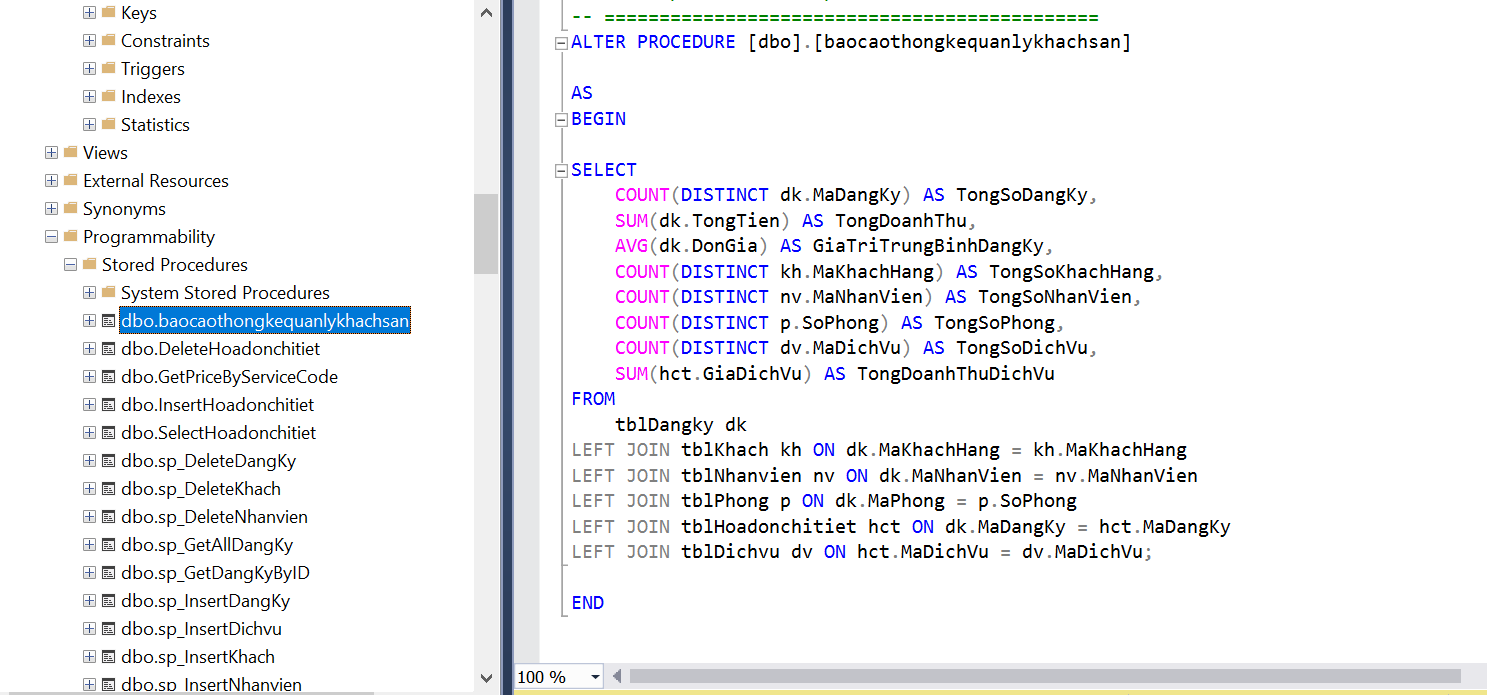


### Sơ đồ liên kết giữa các bảng:

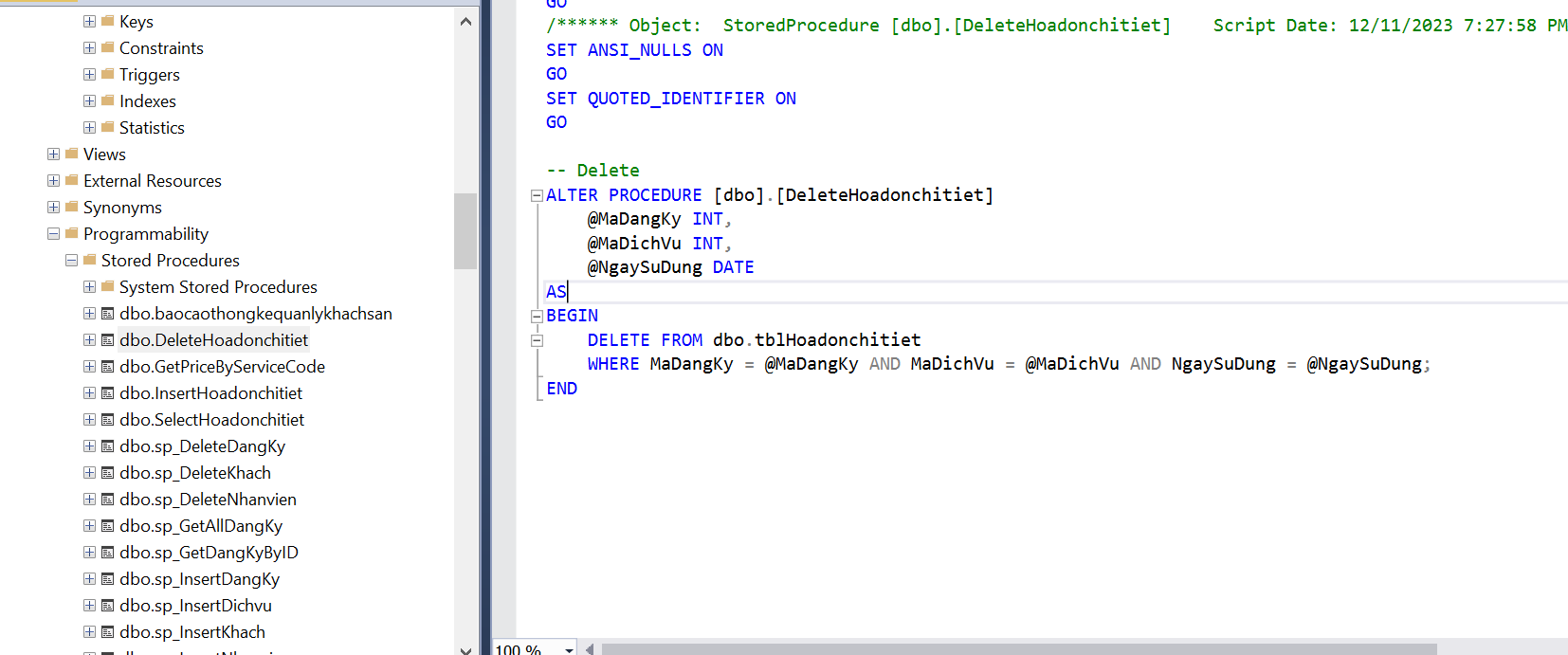


### Các Stored Procedure lưu trữ trong csdl

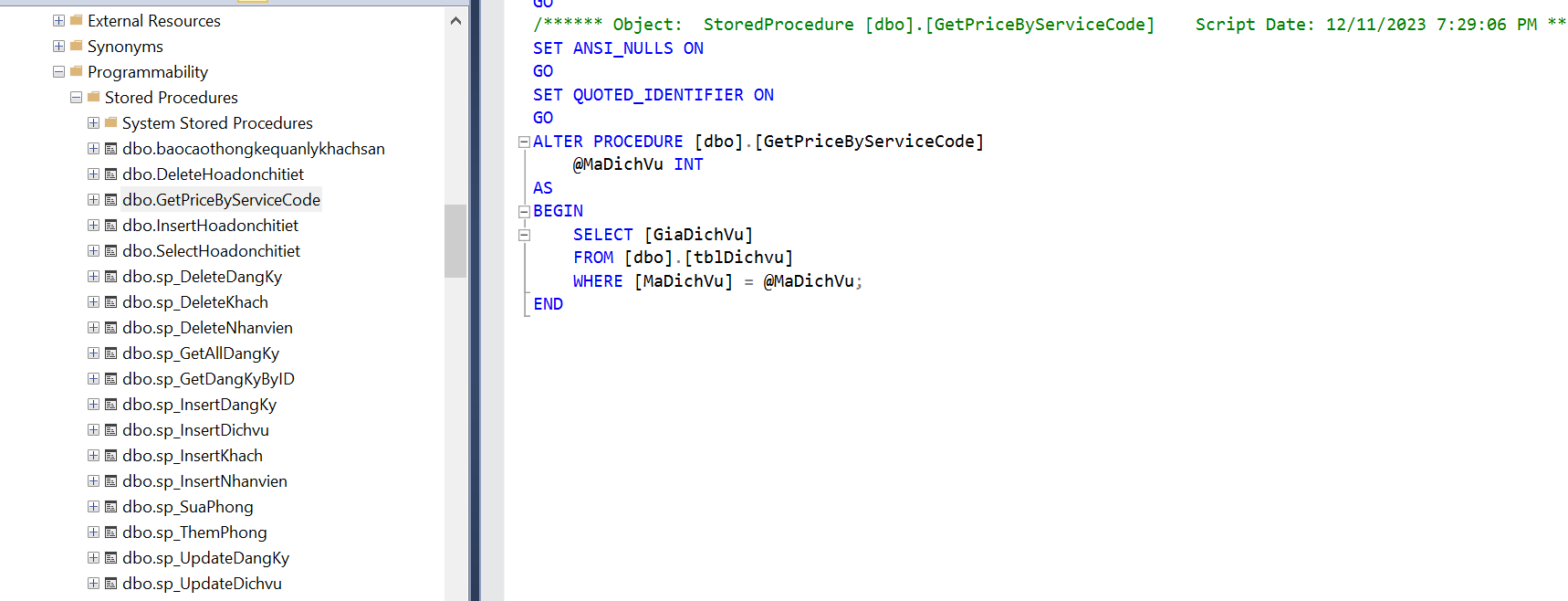
1. Baocaothongkequanlykhachsan



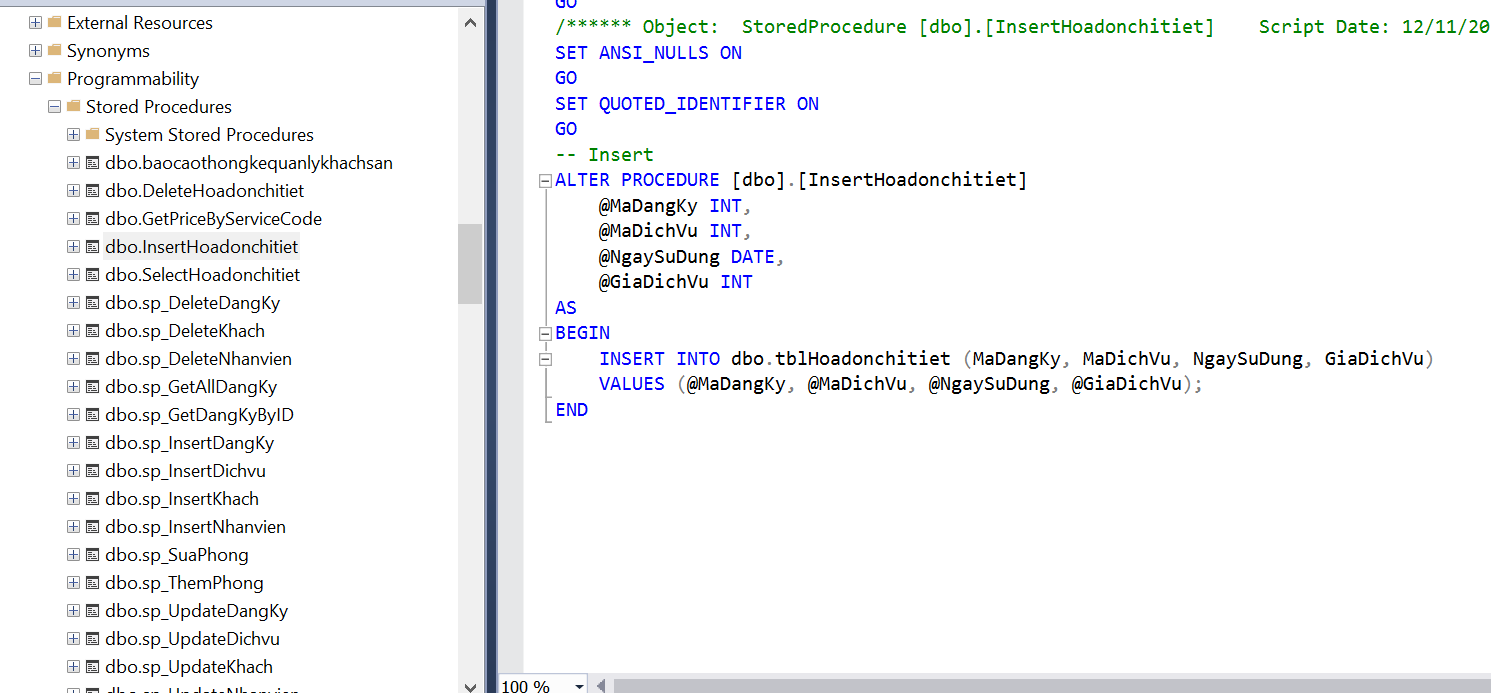
1. DeleteHoadonchitiet



1. GetPriceByServiceCode



1. InsertHoadonchitiet

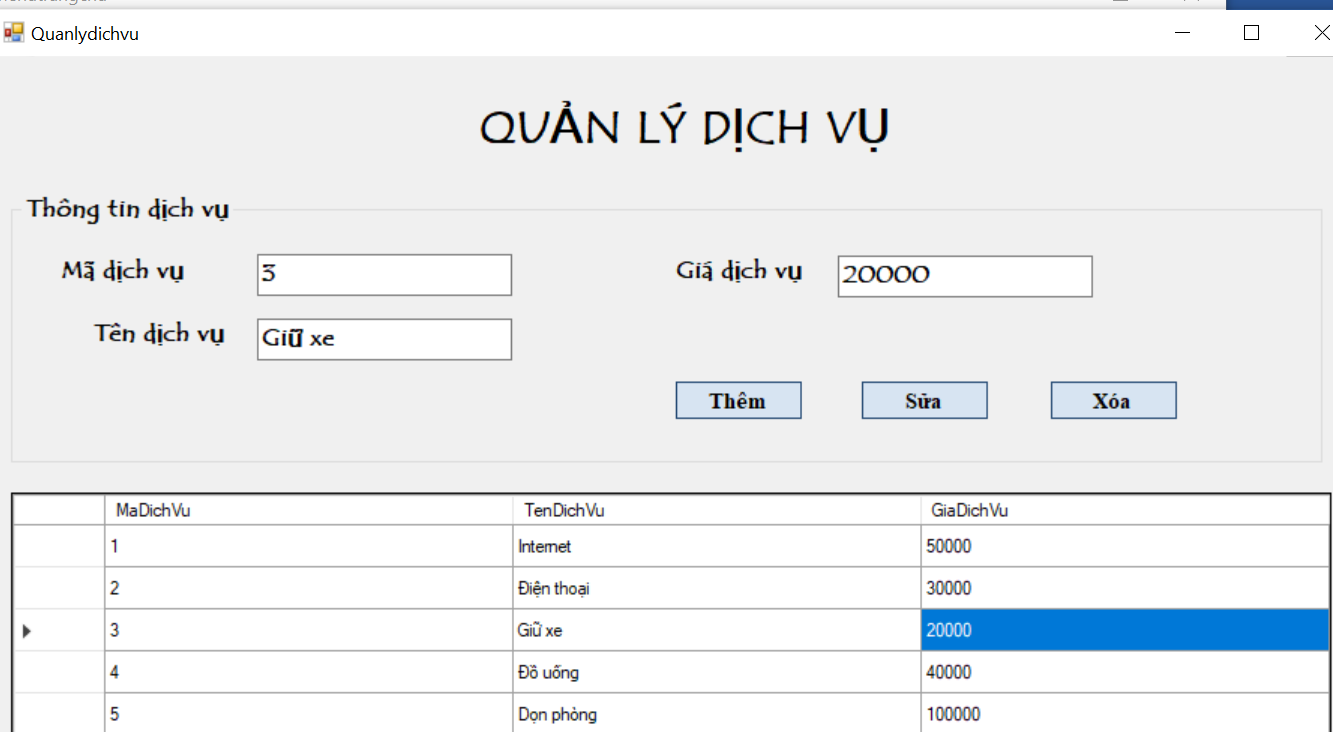


# THIẾT KẾ CHƯƠNG TRÌNH

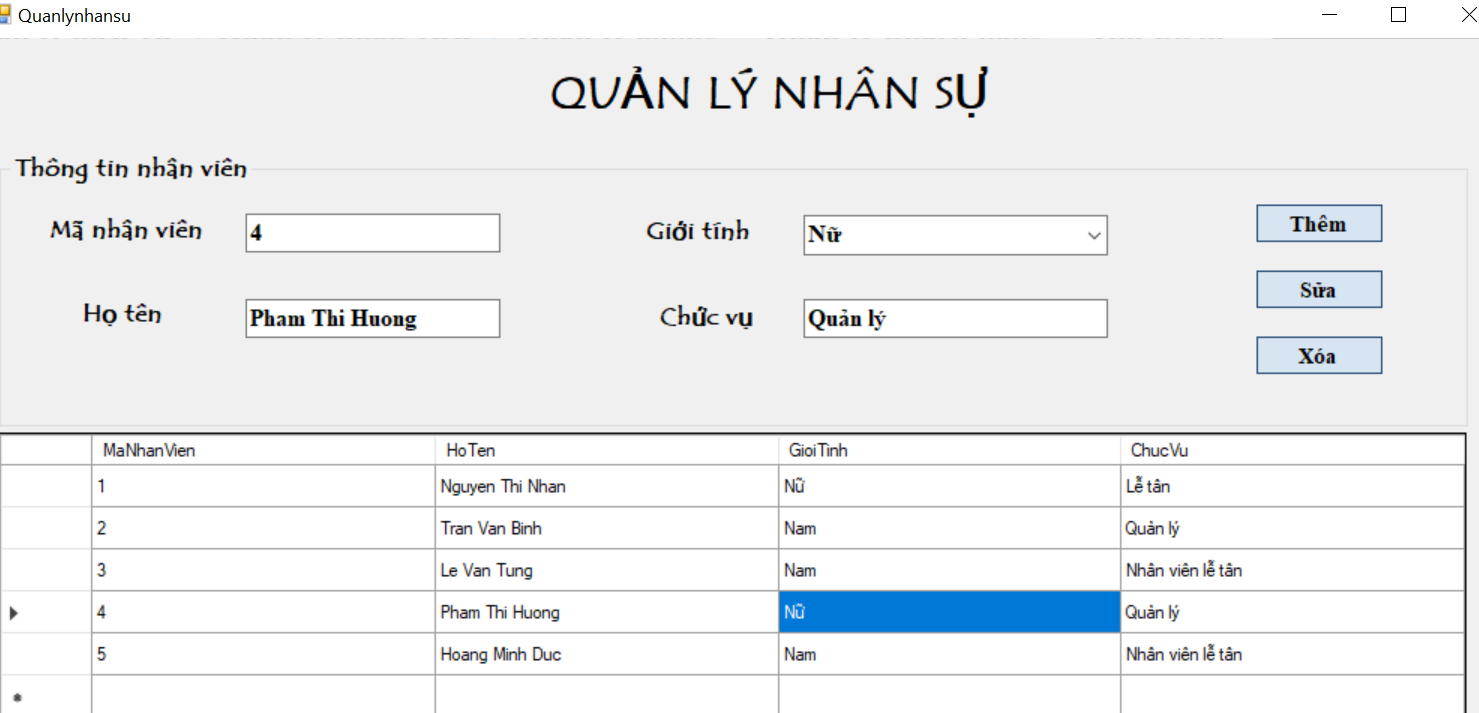
## Trang chủ



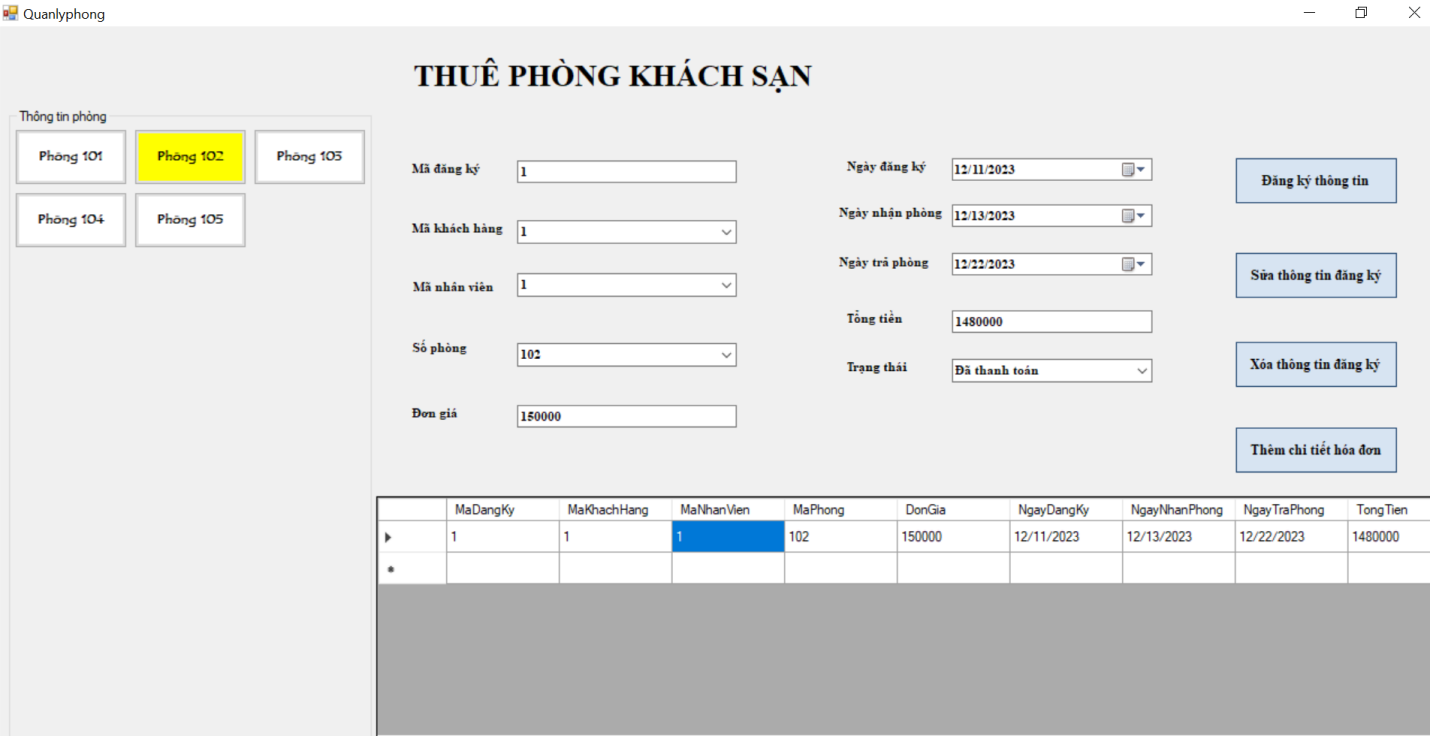
## Quản lý dịch vụ

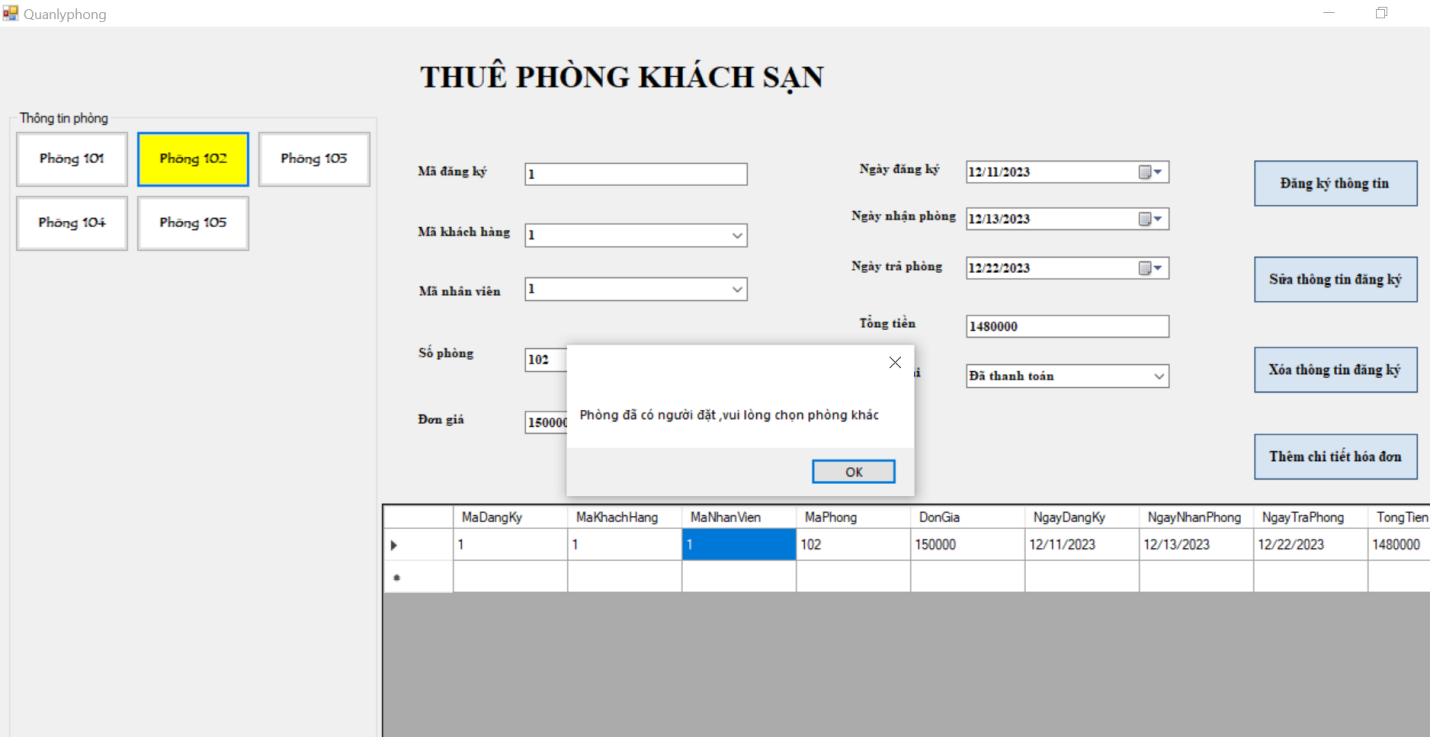


## Quản lý nhân viên

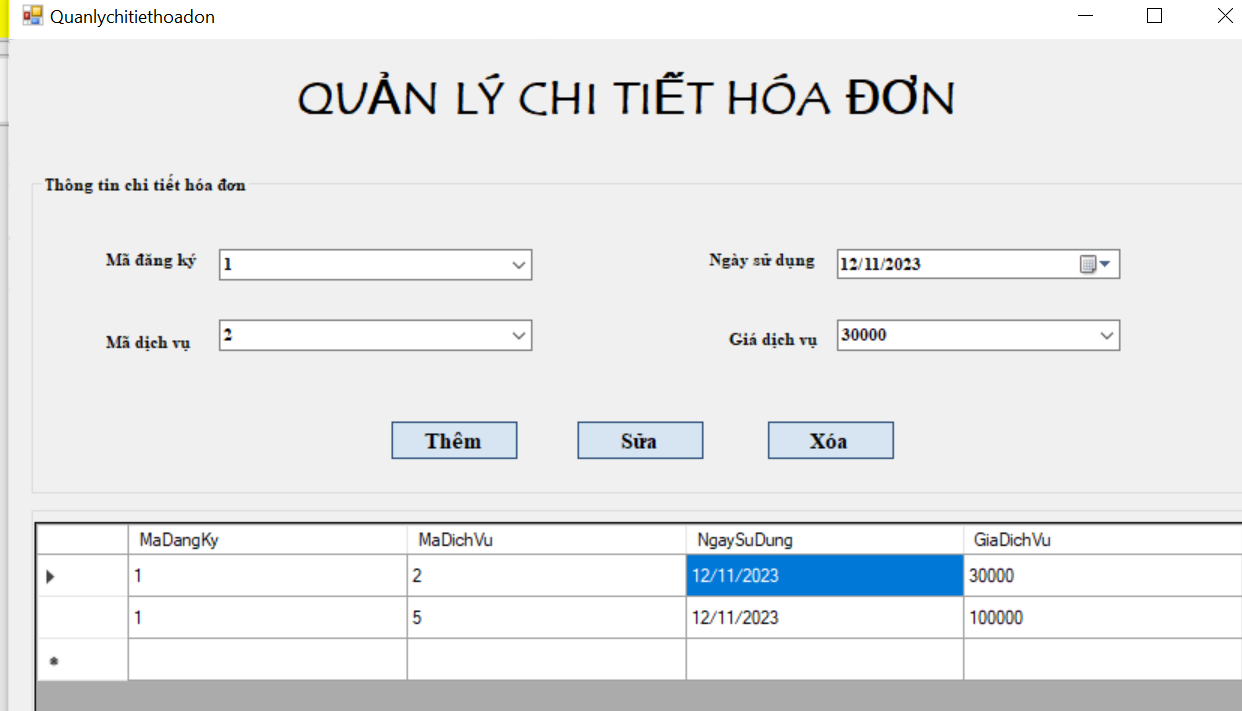


## Quản lý thuê phòng

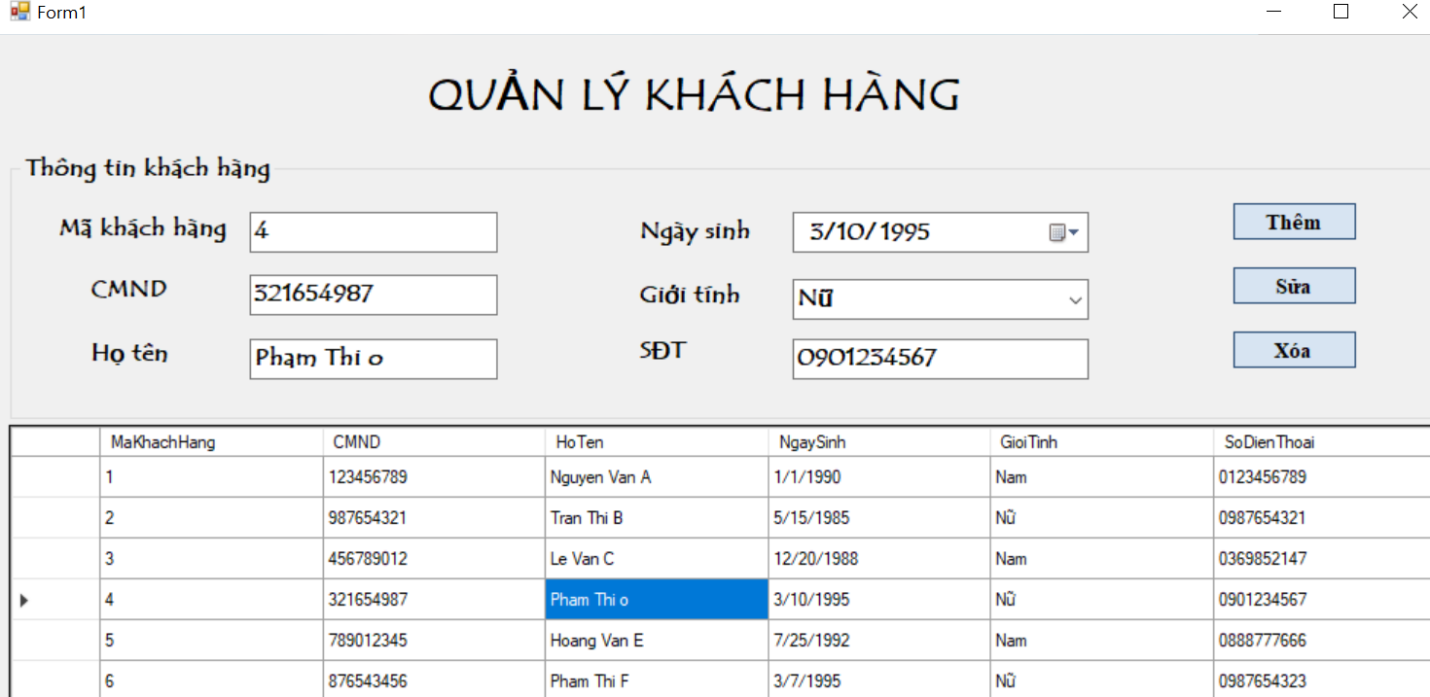




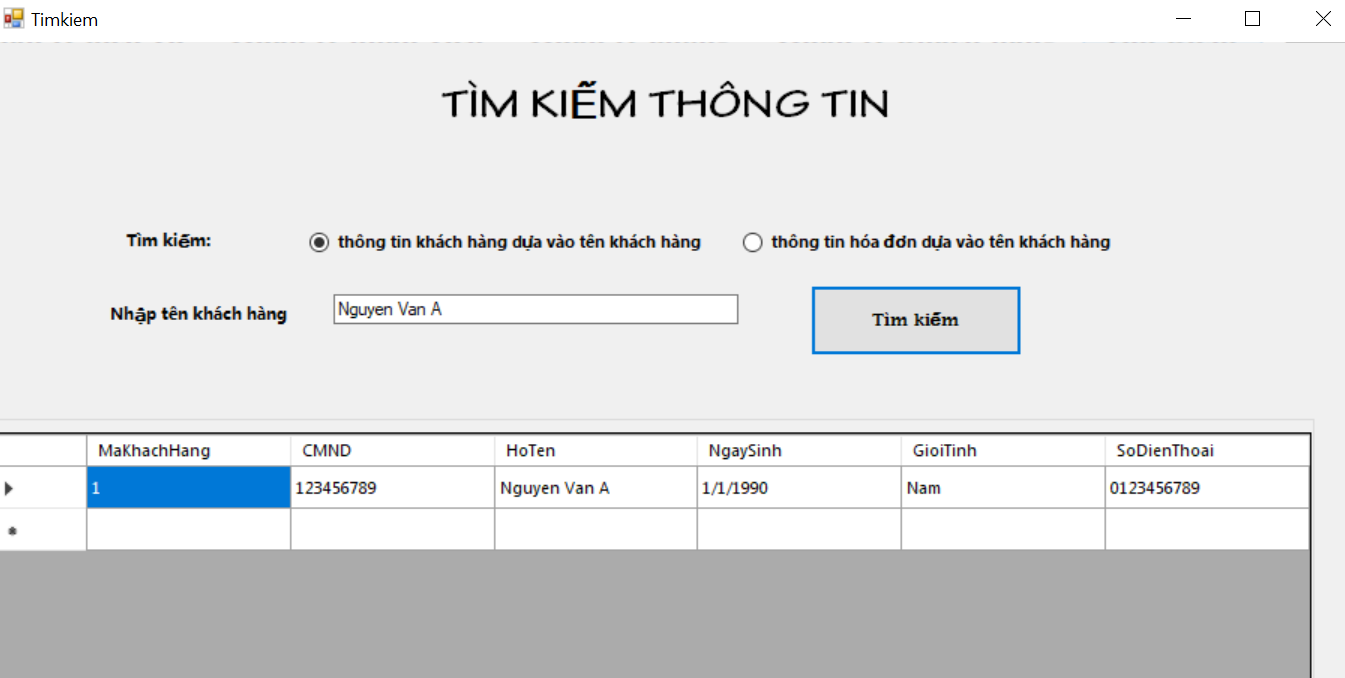
## Quản lý chi tiết hóa đơn

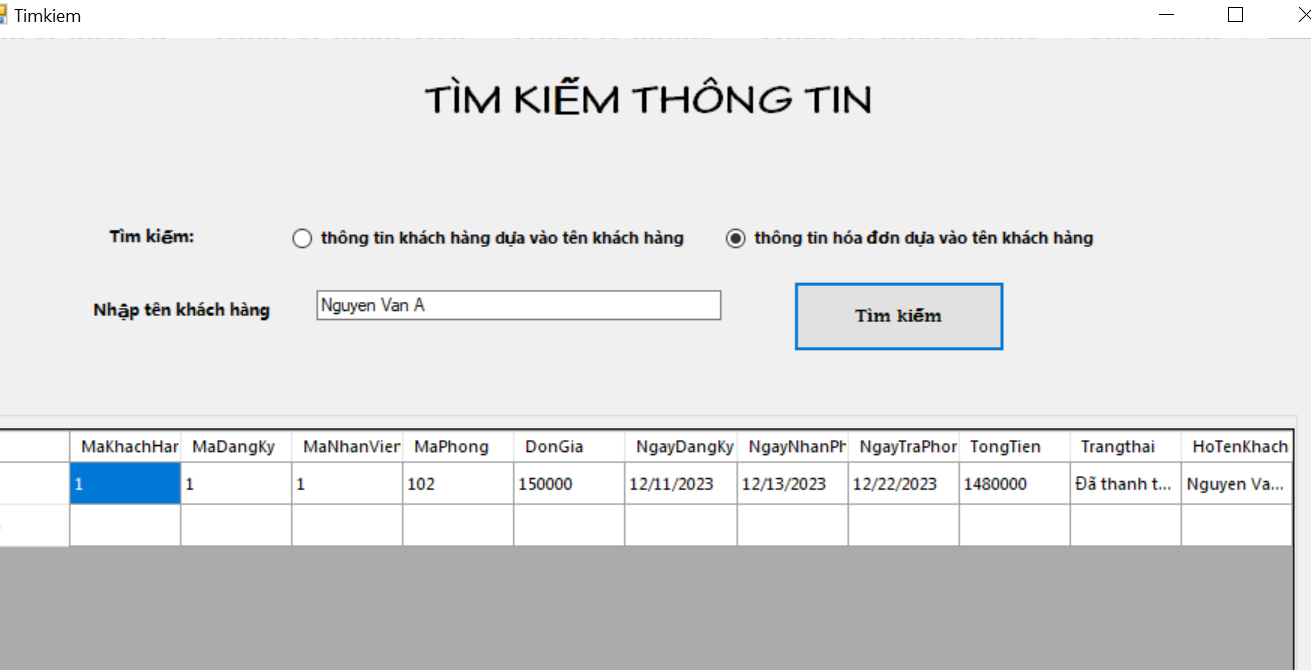


## Quản lý khách hàng

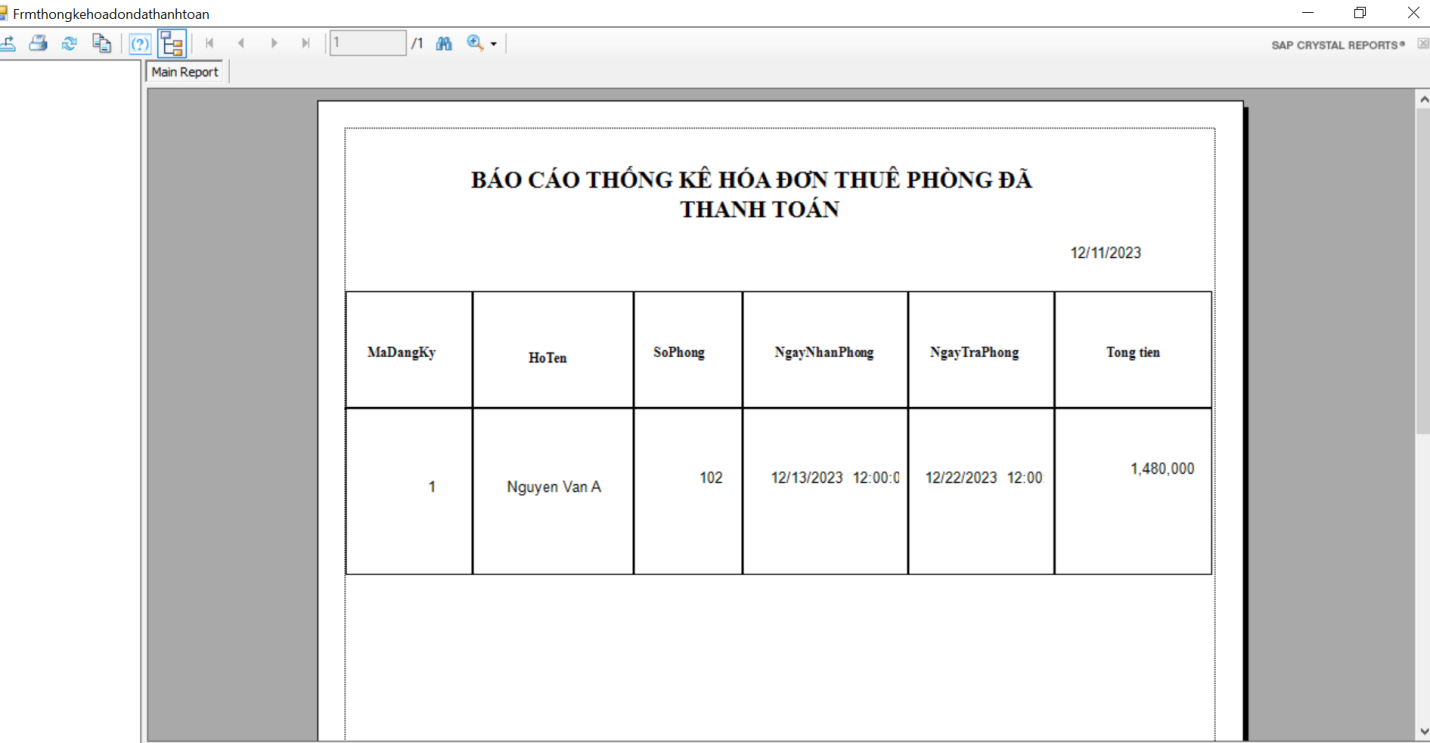


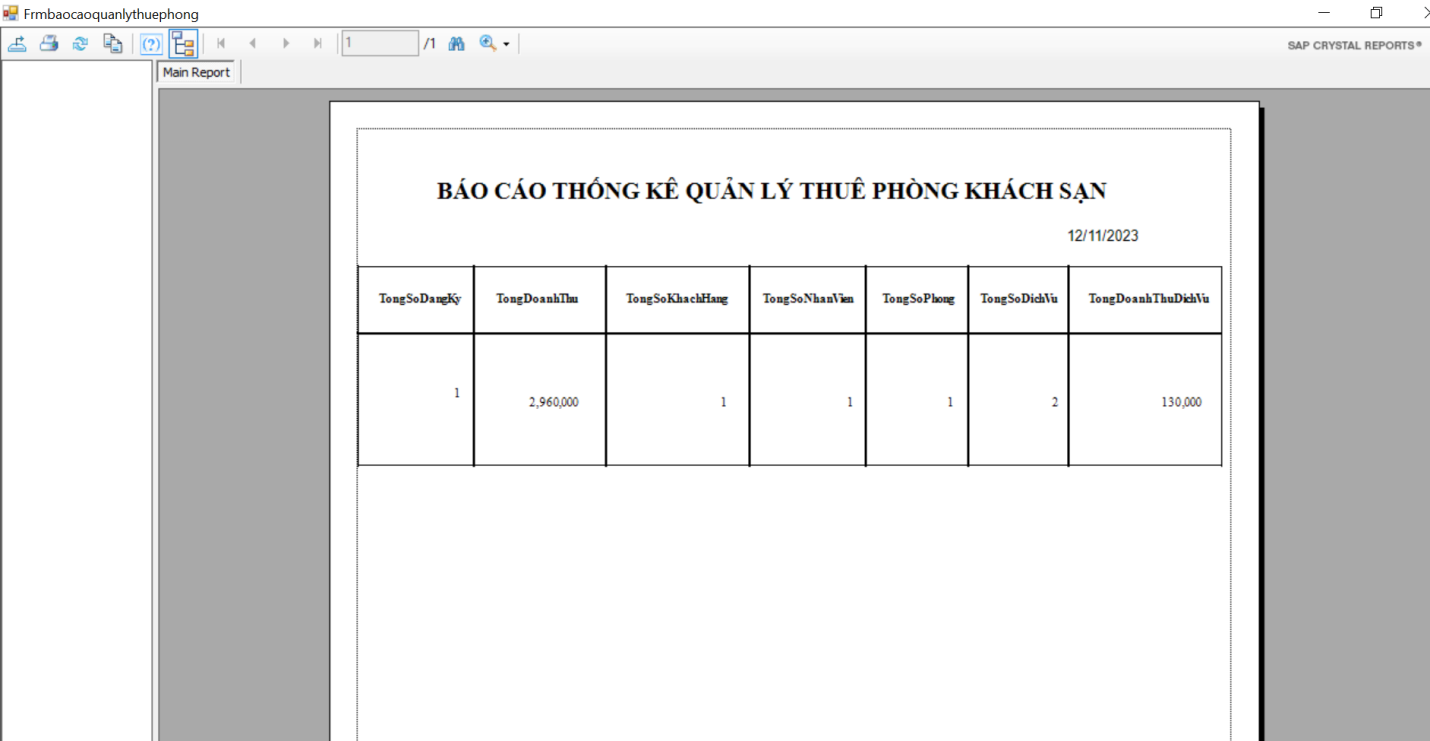
## Tìm kiếm thông tin





## Báo cáo thống kê





## Quản lý thông tin phòng

