

ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC – TỰ NHIÊN
KHOA ĐIỆN TỬ – VIỄN THÔNG
BỘ MÔN THỰC HÀNH LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG
-----o0o-----



ĐỒ ÁN: QUẢN LÝ KHÁCH SẠN

TP. HỒ CHÍ MINH, THÁNG 4 NĂM 2024

ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC – TỰ NHIÊN
KHOA ĐIỆN TỬ – VIỄN THÔNG
BỘ MÔN THỰC HÀNH LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

-----o0o-----



ĐỒ ÁN: QUẢN LÝ KHÁCH SẠN

GVHD: ThS. Cao Trần Bảo Thương

Nhóm: 4

1. Bùi Minh Nhật – 21207070 (Nhóm Trưởng)
2. Huỳnh Lê Trí Cường - 21207128
3. Hoàng Trung Hiếu - 21207030

TP. HỒ CHÍ MINH, THÁNG 4 NĂM 2024

LỜI MỞ ĐẦU

Chúng em xin dành những dòng lời này để bày tỏ lòng biết ơn sâu sắc đến các Thầy, những người đã dành thời gian và tâm huyết để hướng dẫn chúng em trong môn học Thực hành Lập trình Hướng Đối Tượng. Dưới sự chỉ dạy của các Thầy, chúng em đã được trang bị những kiến thức vô cùng quý báu về các nguyên lý cơ bản và ứng dụng thực tiễn của lập trình.

Với sự tận tâm và am hiểu sâu sắc về chủ đề, các Thầy đã không ngừng truyền đạt những kiến thức chuyên sâu cùng những kinh nghiệm thực tiễn giúp chúng em hiểu rõ hơn về các khái niệm phức tạp và ứng dụng chúng trong thế giới thực. Sự cam kết của các Thầy đến từ trái tim và lòng nhiệt thành trong việc chia sẻ kiến thức đã là nguồn động viên lớn lao cho chúng em trong quá trình học tập và nghiên cứu.

Dù chúng em có thể gặp khó khăn và thách thức trong quá trình tiếp thu kiến thức, nhưng sự hỗ trợ và sự khích lệ từ Thầy luôn là nguồn động viên quý báu, giúp chúng em vượt qua mọi khó khăn và phát triển trong hành trình học tập của mình.

Chúng em rất trân trọng những kiến thức mà các Thầy đã truyền đạt và dành riêng những lời cảm ơn sâu sắc nhất đến các Thầy vì sự hiểu biết, tận tâm và sự hướng dẫn tận tình của mình. Chúng em hân hoan chờ đợi những hướng dẫn và phản hồi của Thầy trong tương lai để có thể hoàn thiện bản thân và tiếp tục khám phá sâu hơn về lĩnh vực này.

Tp. Hồ Chí Minh, ngày 26 tháng 04 năm 2024

Trưởng nhóm



Bùi Minh Nhựt

MỤC LỤC

TỔNG QUAN	1
1. Mục Tiêu Và Ý Nghĩa.....	1
2. Cấu Trúc Đồ Án	2
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT VÀ PHẦN MỀM SỬ DỤNG.....	2
1.1 Phần Mềm Sử Dụng	3
1.2 Cơ Sở Lý Thuyết.....	3
1.3 Phân Tích Thiết Kế Cơ Sở Dữ Liệu.....	4
CHƯƠNG 2: THỰC HIỆN ĐỒ ÁN	8
2.1 Thiết Lập Kết Nối	8
2.2 Thiết Kế Giao Diện	9
CHƯƠNG 3: THAO TÁC TRÊN GIAO DIỆN	22
3.1 Một Số Hoạt Động Ở Giao Diện Chính	22
3.2 Chạy Thử Phần Mềm (QRCode + File)	25
DANH MỤC HÌNH ẢNH	26
DANH MỤC BẢNG	27
DANH MỤC TÀI LIỆU THAM KHẢO	28
PHỤ LỤC 1	29

TỔNG QUAN

1. Mục Tiêu Và Ý Nghĩa

Mục Tiêu:

- **Tự động hóa và tối ưu hóa quy trình quản lý:** Phần mềm nhằm mục đích tự động hóa các quy trình quản lý từ việc đặt phòng, check-in và check-out của khách, tới việc tính toán tài chính và bảo trì phòng ốc, nhằm giảm bớt công việc thủ công và tăng cường hiệu quả hoạt động.
- **Cung cấp thông tin tức thì:** Giao diện cho phép người quản lý có cái nhìn tổng quan về tình trạng các phòng (đã đặt, có khách, trống, đang bảo trì), từ đó có thể đưa ra quyết định kinh doanh phù hợp.
- **Quản lý khách hàng hiệu quả:** Lưu trữ và quản lý thông tin khách hàng, gồm họ và tên, số chứng minh nhân dân/căn cước công dân, ngày đến và mã phòng. Điều này giúp nhận diện khách hàng thường xuyên và cung cấp dịch vụ tốt hơn.
- **Tối ưu quản lý tài chính:** Tính năng thanh toán giúp theo dõi chi tiết các giao dịch và hỗ trợ quản lý tài chính một cách minh bạch và chính xác.
- **Tìm kiếm và truy xuất thông tin nhanh chóng:** Tích hợp chức năng tìm kiếm nhanh theo mã phòng hoặc theo tên khách hàng, giúp tiết kiệm thời gian và cải thiện trải nghiệm sử dụng cho nhân viên.

Ý Nghĩa:

- **Thực tiễn:** Phần mềm này giúp áp dụng các kiến thức lý thuyết vào việc giải quyết vấn đề thực tế trong ngành quản lý khách sạn.
- **Cải thiện dịch vụ khách hàng:** Bằng cách làm cho quá trình quản lý trở nên mượt mà và ít lỗi hơn, đề án có ý nghĩa trong việc nâng cao chất lượng dịch vụ khách hàng.
- **Đào tạo kỹ năng:** Đối với sinh viên việc thực hiện một đề án như vậy giúp chúng em phát triển kỹ năng giải quyết vấn đề và kỹ năng công nghệ thông tin.
- **Nghiên cứu và phát triển:** Nó có thể phục vụ như một nền tảng ban đầu cho việc nghiên cứu và phát triển thêm các tính năng nâng cao, ứng dụng AI để dự đoán nhu cầu, phân tích xu hướng và tối ưu hóa doanh thu.

2. Cấu Trúc Tổng Quan Của Dự Án

Cơ Sở Dữ Liệu (SQL Server):

- Khởi tạo Cơ sở dữ liệu: Xây dựng các bảng và mối quan hệ giữa chúng để lưu trữ *thông tin phòng, khách hàng, và hóa đơn*.
- Bảo mật: Áp dụng các chiến lược bảo mật để đảm bảo dữ liệu được an toàn và nguyên vẹn.
- Thiết kế một ứng dụng theo mô hình Windows Forms, kết hợp với cơ sở dữ liệu và xử lý các sự kiện cơ bản để tương tác với người dùng.

Ứng Dụng WinForms (Visual Studio):

- **Formmain.cs:** Là giao diện chính mà người dùng sẽ tương tác trực tiếp khi khởi chạy ứng dụng. Nó có thể bao gồm các chức năng cơ bản như nhập thông tin, hiển thị dữ liệu, và các thao tác cơ bản khác.
- **DataSet1.xsd:** Một tập tin dataset được sử dụng để quản lý dữ liệu trong mô hình nguồn dữ liệu, nó đóng vai trò là cầu nối giữa ứng dụng và cơ sở dữ liệu.
- **Tích hợp dữ liệu:** Dữ liệu từ SQL Server có thể được tích hợp thông qua ADO.NET, với các đối tượng như DataTables, DataSets, và DataAdapters được khai báo và sử dụng trong ứng dụng.
- **Thực hiện Kiểm thử:** Đảm bảo mọi chức năng hoạt động chính xác thông qua việc kiểm thử đơn vị và kiểm thử tích hợp.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT VÀ PHẦN MỀM SỬ DỤNG

1.1 Phần Mềm Sử Dụng



SQL Server Management Studio: SQL Server Management Studio (SSMS) là một môi trường tích hợp để quản lý bất kỳ cơ sở hạ tầng SQL nào, từ SQL Server đến Azure SQL Database. SSMS cung cấp các công cụ để cấu hình, giám sát và quản trị các phiên bản của SQL Server và cơ sở dữ liệu. Sử dụng SSMS để triển khai, giám sát và nâng cấp các thành phần dữ liệu được sử dụng bởi ứng dụng của bạn và xây dựng các truy vấn và script.



Visual Studio: Microsoft Visual Studio là một môi trường phát triển tích hợp từ Microsoft. Microsoft Visual Studio còn được gọi là "Trình soạn thảo mã nhiều người sử dụng nhất thế giới", được dùng để lập trình C++ và C# là chính.

1.2 Cơ Sở Lý Thuyết:

Encapsulation: Đóng gói là một khái niệm trong đó dữ liệu (thuộc tính) và mã (phương thức) được bọc trong một đơn vị duy nhất hoặc một đối tượng. các phương thức và thuộc tính được đóng gói trong lớp `Form1`, ví dụ như các biến `sql`, `chuoiketnoi`, `ketnoi`, `thuchien`, và `docdulieu` đều được giới hạn truy cập trong lớp và được quản lý thông qua các phương thức như `hienthi()`, `btntinhvien_Click()`, v.v.

Inheritance: Kế thừa cho phép một lớp (subclass) kế thừa các thuộc tính và phương thức của lớp khác (superclass). Mặc dù đoạn code không trực tiếp minh họa việc tạo lớp con kế thừa từ lớp cha, nhưng `Form1` kế thừa từ `Form`, một lớp của .NET Framework, cho phép sử dụng các thuộc tính và phương thức của `Form`.

Polymorphism: Đa hình cho phép các phương thức được gọi theo cách khác nhau tùy thuộc vào đối tượng hoặc dữ liệu đầu vào. Trong C#, đa hình thường được thể hiện qua các phương thức được ghi đè (override) hoặc các phương thức được nạp chồng (overload).

Abstraction: Trừu tượng hoá là việc ẩn đi các chi tiết phức tạp và chỉ hiển thị những thông tin cần thiết đối với người dùng. Trong Windows Forms, điều này được thực hiện thông qua việc tạo các controls (như buttons, text boxes) mà người dùng cuối không cần quan tâm đến cách chúng hoạt động bên trong, chỉ cần biết cách tương tác.

Composition: Composition là việc sử dụng các đối tượng hiện có như các thành phần của đối tượng mới, thay vì kế thừa từ chúng. các đối tượng như 'SqlConnection', 'SqlCommand', 'SqlDataReader' được sử dụng như các thành phần trong lớp 'Form1' để thực hiện kết nối và thao tác dữ liệu.

1.3 Mô Tả Design Table(Trong DataBase)

1. Khách Hàng (KHACHHANG1)

	Column Name	Data Type	Allow Nulls
►	MaKH	nchar(10)	<input checked="" type="checkbox"/>
	HoVaTen	nchar(50)	<input checked="" type="checkbox"/>
	SoCCCD	int	<input checked="" type="checkbox"/>
	NgayDen	date	<input checked="" type="checkbox"/>
	MaPhong	nchar(10)	<input checked="" type="checkbox"/>
	NgayLuuTru	int	<input checked="" type="checkbox"/>
	ThanhTien	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Bảng 1.1: Design Table (dbo.KHACHHANG)

	MaKH	HoVaTen	SoCCCD	NgayDen	MaPhong	NgayLuuTru	ThanhTien
►	KH01	Bui Minh Nhut ...	34522424	2024-04-14	Phong 01	NULL	720000
	KH02	Hoang Trung H...	23452342	2024-04-23	Phong 02	NULL	90000
	KH03	Huynh Le Tri C...	34134143	2024-04-21	Phong 03	NULL	400000
	KH04	Nguyen Xuan C...	34313243	2024-04-25	Phong 04	NULL	50000
	KH05	NULL	NULL	NULL	Phong 05	NULL	NULL
	KH06	NULL	NULL	NULL	Phong 06	NULL	NULL
	KH07	NULL	NULL	NULL	Phong 07	NULL	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Bảng 1.2: Dữ Liệu Của Bảng (dbo.KHACHHANG)

- MaKH: Mã của khách hàng, dùng để nhận diện duy nhất mỗi khách hàng.
- HoVaTen: Họ và tên của khách hàng.
- SoCCCD: Số căn cước công dân của khách hàng.

- NgayLuuTru: Ngày đến lưu trú của khách hàng.
- MaPhong: Mã phòng mà khách hàng đặt.
- ThanhTien: Tổng tiền khách đã sử dụng dịch vụ của Khách Sạn.

Thao tác trên thực thể Khách Hàng:

- Truy vấn thông tin khách hàng dựa trên mã phòng hoặc tên.
- Cập nhật danh sách khách hàng trên giao diện người dùng.

2. Phòng (PHONG)

	Column Name	Data Type	Allow Nulls
►	MaPhong	nchar(10)	<input checked="" type="checkbox"/>
	SoGiuong	nchar(10)	<input checked="" type="checkbox"/>
	DonGia	int	<input checked="" type="checkbox"/>
	TinhTrang	nchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Bảng 1.3: Design Table (dbo.PHONG)

	MaPhong	SoGiuong	DonGia	TinhTrang	
	Phong 01	2	60000	Có Khách	...
	Phong 02	1	30000	Trống	...
	Phong 03	4	80000	Trống	...
	Phong 04	3	50000	Đã Đặt	...
	Phòng 05	2	50000	Trống	...
	Phòng 06	1	30000	Bảo Trì	...
	Phòng 07	2	40000	Trống	...
►*	NULL	NULL	NULL	NULL	

Bảng 1.4: Dữ Liệu Của Bảng (dbo.PHONG)

- MaPhong: Mã của phòng, định danh duy nhất cho mỗi phòng.
- SoGiuong: Số Giường mà khách muốn thuê
- DonGia: Đơn giá thuê phòng.
- TinhTrang: Tình trạng hiện tại của phòng (ví dụ: đã đặt, trống).

Thao tác trên thực thể Phòng:

- Tính tiền dựa trên số ngày lưu trú và đơn giá phòng.
- Hiển thị và cập nhật tình trạng phòng.

3. Booking (Giả định có bảng này dựa trên chức năng đặt phòng)

	Column Name	Data Type	Allow Nulls
►	MaKH	nchar(10)	<input checked="" type="checkbox"/>
	HoVaTen	nchar(10)	<input checked="" type="checkbox"/>
	SoCCCD	nchar(10)	<input checked="" type="checkbox"/>
	NgayNhanPhong	date	<input checked="" type="checkbox"/>
	NgayTraPhong	date	<input checked="" type="checkbox"/>
	LoaiPhong	nchar(10)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Bảng 1.5: Design Table ([dbo.booking](#))

	MaKH	HoVaTen	SoCCCD	NgayNhanPho...	NgayTraPhong	LoaiPhong
	NULL	NULL	NULL	NULL	NULL	NULL

Bảng 1.6: Dữ Liệu Của Bảng ([dbo.booking](#))

- MaKH: Mã khách hàng.
- HoVaTen: Tên khách hàng.
- SoCCCD: Số căn cước công dân.
- NgayNhanPhong: Ngày nhận phòng.
- NgayTraPhong: Ngày trả phòng.
- LoaiPhong: Loại phòng đã đặt.

Thao tác trên thực thể Booking:

- Thêm thông tin đặt phòng vào cơ sở dữ liệu.
- Hiện thị thông tin đặt phòng trên giao diện người dùng.
- Giả sử khi có khách đặt phòng khi chúng ta nhập liệu thông tin sẽ được đưa lên dữ liệu của bảng ([dbo.booking](#))

4. Quản Lý Hóa Đơn

	Column Name	Data Type	Allow Nulls
	MaHD	nchar(10)	<input checked="" type="checkbox"/>
	NgayThanhToan	date	<input checked="" type="checkbox"/>
	TongTien	nchar(10)	<input checked="" type="checkbox"/>
	MaPhong	nchar(10)	<input checked="" type="checkbox"/>
	MaKH	nchar(10)	<input checked="" type="checkbox"/>
►			<input type="checkbox"/>

Bảng 1.7: Design Table ([dbo.QLHoaDon](#))

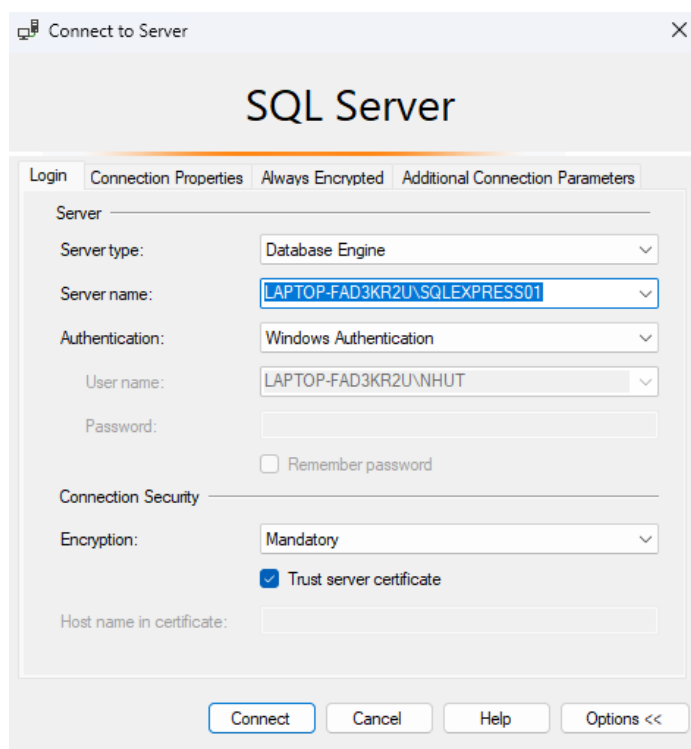
	MaHD	NgayThanhToan	TongTien	MaPhong	MaKH
	HD002	2024-04-26	90000	Phong 02	KH02
	HD003	2024-04-26	400000	Phong 03	KH03
	HD004	2024-04-26	50000	Phong 04	KH04
►*	NULL	NULL	NULL	NULL	NULL

Bảng 1.8: Dữ Liệu Của Bảng ([dbo.QLHoaDon](#))

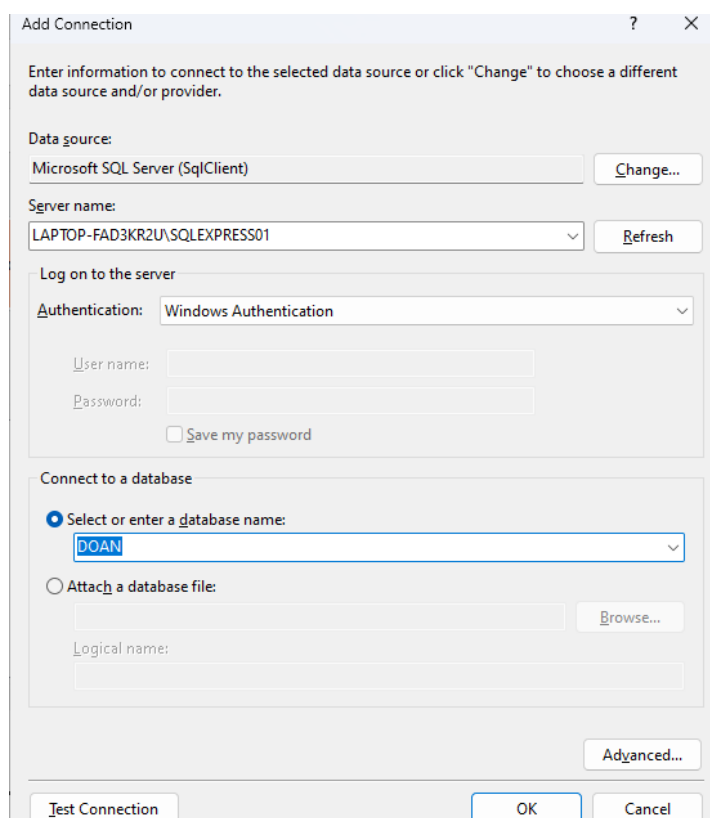
- MaHD: Mã hóa đơn.
- NgayThanhToan: Ngày khách kết thúc thuê phòng.
- TongTien: Số tiền mà khách thanh toán.
- MaPhong: Mã phòng.
- MaKH: Mã khách hàng.

CHƯƠNG 2: THỰC HIỆN ĐỒ ÁN

2.1 Thiết Lập Kết Nối

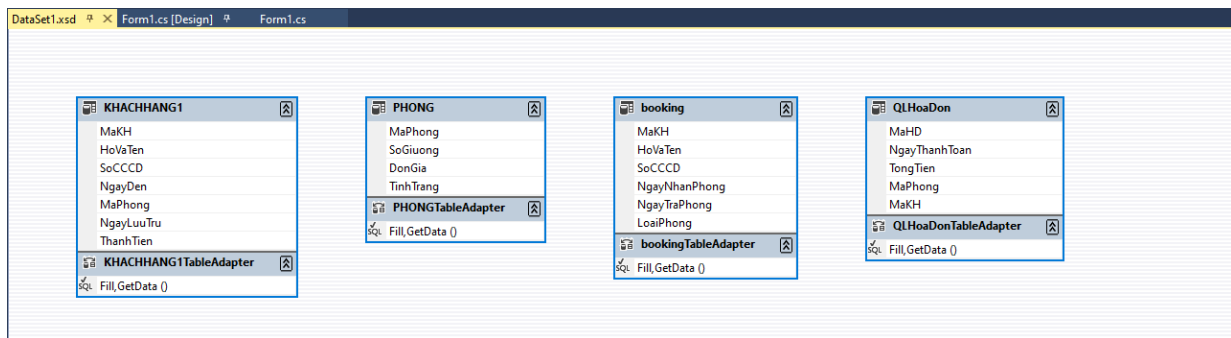


Hình 1.1: Kết Nối Đến Server



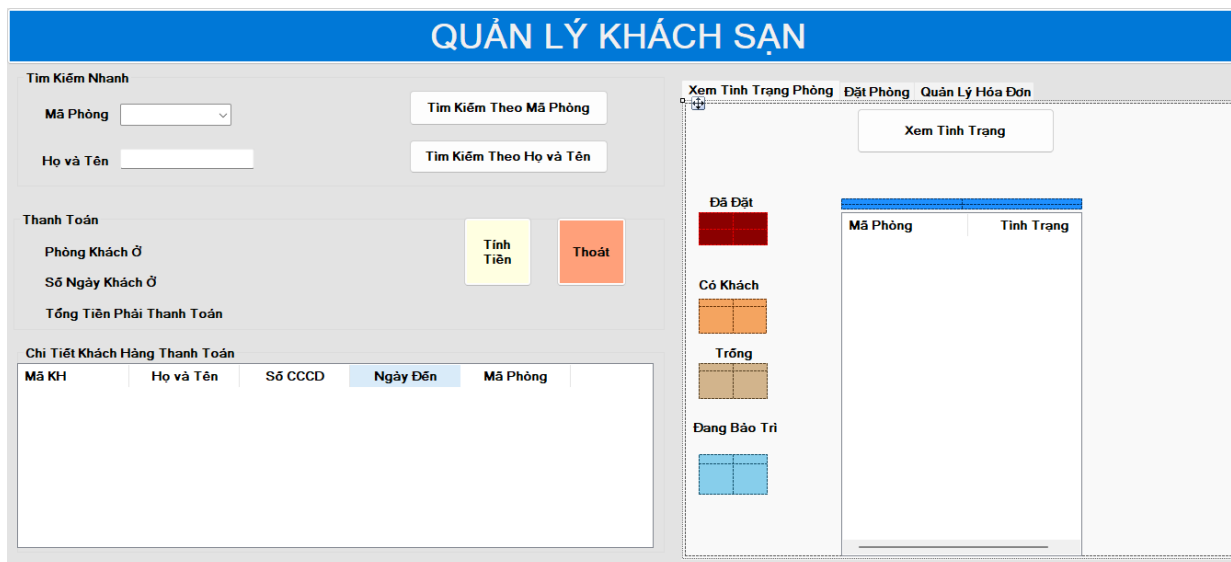
Hình 1.2: Thêm Kết Nối

Thêm DataSet từ Add New Items:



Hình 1.3: Các Bảng Đã Được Thêm Vào DataSet

2.2 Thiết Kế Giao Diện (Windows Form)



Hình 1.4: Giao Diện Của Phần Mềm

Giao diện sau khi được thiết kế bằng cách dùng các Toolbox (TextBox, Label, Button, ListView, TabControl...).

2.3 Code (FormMain.cs)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.SqlClient;
```

- System: Cung cấp các lớp cơ bản và kiểu dữ liệu.
- System.Collections.Generic: Hỗ trợ cấu trúc dữ liệu tổng quát.
- System.ComponentModel: Hỗ trợ thiết kế giao diện người dùng.
- System.Data: Cung cấp các lớp làm việc với dữ liệu và cơ sở dữ liệu.
- System.Drawing: Hỗ trợ xử lý đồ họa.
- System.Linq: Cho phép thực hiện truy vấn dữ liệu.
- System.Text: Cung cấp công cụ xử lý chuỗi.
- System.Threading.Tasks: Hỗ trợ lập trình bất đồng bộ và đa luồng.
- System.Windows.Forms: Là nền tảng chính cho giao diện người dùng đồ họa.
- System.Data.SqlClient: Để kết nối và truy vấn SQL Server.
- string sql;: Biến này được dùng để lưu trữ các câu truy vấn SQL.

```
public Form1()
{
    InitializeComponent();
}
string sql;
string chuoiKetnoi = @"Data Source=LAPTOP-FAD3KR2U\SQLEXPRESS01;Initial
Catalog=KHACHHANG;Integrated Security=True";
SqlConnection ketnoi;
SqlCommand thuchien;
SqlDataReader docdulieu;

int i = 0;
```

- string chuoiKetnoi;: Chuỗi kết nối đến cơ sở dữ liệu SQL Server. Nó chỉ ra nguồn dữ liệu (server và instance của SQL Server), cơ sở dữ liệu mục tiêu (KHACHHANG), và loại xác thực (Integrated Security=True dùng xác thực Windows).
- SqlConnection ketnoi;: Đối tượng này quản lý kết nối giữa ứng dụng C# và cơ sở dữ liệu SQL Server.

- SqlCommand thuchien;: Đối tượng này được dùng để thực thi câu lệnh SQL được định nghĩa trong biến sql.
- SqlDataReader docdulieu;: Đối tượng này đọc dữ liệu trả về từ cơ sở dữ liệu sau khi thực thi câu lệnh SQL bằng SqlCommand

Load dữ liệu vào DataSet:

```
private void Form1_Load(object sender, EventArgs e)
{
    this.pHONGTableAdapter.Fill(this.dataSet1.PHONG);
    ketnoi = new SqlConnection(chuoiKetnoi);
    hienthi();
}
```

- **this.pHONGTableAdapter.Fill(this.dataSet1.PHONG):** Dòng này sử dụng adapter pHONGTableAdapter để nạp dữ liệu từ bảng PHONG vào dataSet1.PHONG. Điều này đảm bảo rằng khi form được nạp, nó cũng sẽ hiển thị dữ liệu mới nhất từ bảng PHONG

Khởi tạo kết nối cơ sở dữ liệu:

- **ketnoi = new SqlConnection(chuoiKetnoi):** Khởi tạo đối tượng SqlConnection mới với chuỗi kết nối được định nghĩa trước (chuoiKetnoi).

Hiển thị dữ liệu:

- **hienthi():** Gọi phương thức hienthi() để tải và hiển thị dữ liệu.

```
public void hienthi()
{
    listView1.Items.Clear();
    ketnoi.Open();
    sql = @"Select MaKH, HoVaTen, SoCCCD,NgayDen,MaPhong From KHACHHANG1";
    thuchien = new SqlCommand(sql, ketnoi);
    docdulieu = thuchien.ExecuteReader();
    i = 0;
    while (docdulieu.Read())
    {
        listView1.Items.Add(docdulieu[0].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[1].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[2].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[3].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[4].ToString());
        i++;
    }
    ketnoi.Close();
}
```

```

        listView4.Items.Clear();
        ketnoi.Open();
        sql = "SELECT MaHD, NgayThanhToan, TongTien, MaPhong, MaKH FROM
dbo.QLHoaDon";
        thuchien = new SqlCommand(sql, ketnoi);
        docdulieu = thuchien.ExecuteReader();
        while (docdulieu.Read())
        {
            ListViewItem item = new
ListViewItem(docdulieu["MaHD"].ToString());
            item.SubItems.Add(docdulieu["NgayThanhToan"].ToString());
            item.SubItems.Add(docdulieu["TongTien"].ToString());
            item.SubItems.Add(docdulieu["MaPhong"].ToString());
            item.SubItems.Add(docdulieu["MaKH"].ToString());
            listView4.Items.Add(item);
        }
        ketnoi.Close();
    }
}

```

1. Xóa danh sách hiện tại:

- listView1.Items.Clear(): Xóa tất cả các mục hiện có trong listView1.
- listView4.Items.Clear(): Tương tự, xóa tất cả mục trong listView4.

2. Mở kết nối cơ sở dữ liệu:

- ketnoi.Open(): Mở kết nối đến cơ sở dữ liệu bằng đối tượng SqlConnection.

3. Thực hiện truy vấn và hiển thị dữ liệu cho listView1:

- sql = @"Select MaKH, HoVaTen, SoCCCD, NgayDen, MaPhong From KHACHHANG1": Đặt câu truy vấn SQL để lấy dữ liệu khách hàng.
- thuchien = new SqlCommand(sql, ketnoi): Tạo một SqlCommand với câu truy vấn đã định.
- docdulieu = thuchien.ExecuteReader(): Thực hiện câu truy vấn và lấy kết quả.
- Vòng lặp while (docdulieu.Read()) được sử dụng để đọc từng dòng dữ liệu trả về và thêm vào listView1 dưới dạng các mục và subitems.

4. Thực hiện truy vấn và hiển thị dữ liệu cho listView4:

- sql = "SELECT MaHD, NgayThanhToan, TongTien, MaPhong, MaKH FROM dbo.QLHoaDon": Đặt câu truy vấn SQL để lấy dữ liệu hóa đơn.

- thuchien = new SqlCommand(sql, ketnoi): Tạo một SqlCommand mới.
- docdulieu = thuchien.ExecuteReader(): Thực hiện câu truy vấn và lấy kết quả.
- Dữ liệu từ mỗi hàng được đọc và thêm vào listView4 với thông tin chi tiết của hóa đơn.

5. Đóng kết nối cơ sở dữ liệu:

- ketnoi.Close(): Đóng kết nối sau khi hoàn thành truy vấn và cập nhật giao diện người dùng.

❖ BUTTON Tìm Kiếm Mã Phòng

```
private void btntkmaphong_Click(object sender, EventArgs e)
{
    listView1.Items.Clear();
    ketnoi.Open();
    sql = "Select MaKH, HoVaTen, SoCCCD, NgayDen, MaPhong From KHACHHANG1 Where
(MaPhong like '%" + cbmaphong.Text + "%')";
    thuchien = new SqlCommand(sql, ketnoi);
    docdulieu = thuchien.ExecuteReader();
    i = 0;
    while (docdulieu.Read())
    {
        listView1.Items.Add(docdulieu[0].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[1].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[2].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[3].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[4].ToString());
        i++;
    }
    ketnoi.Close();
}
```

1. Xóa danh sách hiện có:

- listView1.Items.Clear(): Xóa tất cả các mục trong listView1 để chuẩn bị cho việc hiển thị dữ liệu mới.

2. Mở kết nối cơ sở dữ liệu:

- ketnoi.Open(): Mở kết nối đến cơ sở dữ liệu SQL Server.

3. Thực hiện truy vấn SQL:

- `sql = "Select MaKH, HoVaTen, SoCCCD, NgayDen, MaPhong From KHACHHANG1 Where (MaPhong like '%' + cbmaphong.Text + '%')"`: Tạo câu truy vấn SQL để lấy thông tin khách hàng có mã phòng phù hợp với giá trị được nhập trong ComboBox.
- `thuchien = new SqlCommand(sql, ketnoi)`: Tạo SqlCommand với câu truy vấn đã xác định.
- `docdulieu = thuchien.ExecuteReader()`: Thực hiện truy vấn và lấy kết quả dưới dạng SqlDataReader.

4. Đọc và hiển thị kết quả truy vấn:

- Dùng vòng lặp while (`docdulieu.Read()`) để đọc từng dòng dữ liệu trả về. Dữ liệu được đọc bao gồm các trường MaKH, HoVaTen, SoCCCD, NgayDen, và MaPhong.
- Thêm mỗi dòng dữ liệu vào `listView1` dưới dạng một mục mới với các subitems tương ứng.

5. Đóng kết nối cơ sở dữ liệu:

- `ketnoi.Close()`: Sau khi hoàn thành việc truy vấn và cập nhật giao diện người dùng, kết nối đến cơ sở dữ liệu được đóng lại.

❖ BUTTON Tìm Kiếm Họ Và Tên

```
private void btntkhovaten_Click(object sender, EventArgs e)
{
    listView1.Items.Clear();
    ketnoi.Open();
    sql = "Select MaKH, HoVaTen, SoCCCD, NgayDen, MaPhong From KHACHHANG1 Where (HoVaTen like '%' + txthovaten.Text + '%')";
    thuchien = new SqlCommand(sql, ketnoi);
    docdulieu = thuchien.ExecuteReader();
    i = 0;
    while (docdulieu.Read())
    {
        listView1.Items.Add(docdulieu[0].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[1].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[2].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[3].ToString());
        listView1.Items[i].SubItems.Add(docdulieu[4].ToString());
        i++;
    }
    ketnoi.Close();
}
```

Đoạn code này tương tự như phân tích đã được thực hiện ở đoạn trước, nhưng lần này là để xử lý sự kiện click cho nút btntkhovaten. Nó tìm kiếm và hiển thị thông tin khách hàng dựa trên tên khách hàng được nhập từ một control TextBox (txthovaten).

❖ BUTTON Tính Tiền

```
private void btntinhvien_Click(object sender, EventArgs e)
{
    ketnoi.Open();
    sql = @"Select PHONG.MaPhong, KHACHHANG1.HoVaTen,
Datediff(day,KHACHHANG1.NgayDen, GETDATE()), Datediff(day,KHACHHANG1.NgayDen, GETDATE())
* PHONG.DonGia
FROM      PHONG Inner Join KHACHHANG1
ON        PHONG.MaPhong = KHACHHANG1.MaPhong
Where     (KHACHHANG1.MaPhong = N'" + cbmaphong.Text + @"') and
(KHACHHANG1.HoVaTen = N'" + txthovaten.Text + @"')";
    thuchien = new SqlCommand(sql, ketnoi);
    docdulieu = thuchien.ExecuteReader();
    while (docdulieu.Read())
    {
        lbphong.Text = docdulieu[0].ToString();
        lbsongayotien.Text = docdulieu[2].ToString();
        lbtongtien.Text = docdulieu[3].ToString();
        i++;
    }
    ketnoi.Close();
}
```

1. Mở kết nối cơ sở dữ liệu:

- ketnoi.Open(): Mở kết nối với cơ sở dữ liệu SQL Server.

2. Thực hiện truy vấn SQL:

- sql = @"Select PHONG.MaPhong, KHACHHANG1.HoVaTen, Datediff(day,KHACHHANG1.NgayDen, GETDATE()), Datediff(day,KHACHHANG1.NgayDen, GETDATE()) * PHONG.DonGia ...": Truy vấn này tính toán số ngày lưu trú từ ngày đến đến ngày hiện tại (GETDATE()) và tổng tiền phải trả dựa trên đơn giá phòng. Truy vấn sử dụng kết nối (JOIN) giữa bảng PHONG và KHACHHANG1 dựa trên mã phòng.
- thuchien = new SqlCommand(sql, ketnoi): Khởi tạo đối tượng SqlCommand với câu truy vấn SQL đã tạo.
- docdulieu = thuchien.ExecuteReader(): Thực thi truy vấn và trả về dữ liệu dưới dạng SqlDataReader.

3. Đọc và hiển thị kết quả:

- Vòng lặp while (docdulieu.Read()) để đọc từng dòng dữ liệu trả về từ truy vấn.
- Cập nhật các controls trên form với dữ liệu trả về:
 - lbphong.Text: Hiển thị mã phòng.
 - lbsongayo.Text: Hiển thị số ngày lưu trú.
 - lbtongtien.Text: Hiển thị tổng tiền phải trả.

4. Đóng kết nối cơ sở dữ liệu:

- ketnoi.Close(): Đóng kết nối cơ sở dữ liệu sau khi hoàn tất truy vấn.

❖ BUTTON Thoát

```
private void btnthoat_Click(object sender, EventArgs e)
{
    // Hiển thị MessageBox với các nút Yes và No
    var result = MessageBox.Show("Bạn có chắc chắn muốn thoát không?",
    "Thoát Ứng Dụng", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

    // Kiểm tra xem người dùng đã nhấn Yes hay chưa
    if (result == DialogResult.Yes)
    {
        this.Close(); // Đóng form hiện tại và thoát ứng dụng nếu đây là
form chính
    }
}
```

- **Hiển thị MessageBox:** Khi người dùng nhấn vào nút thoát, một MessageBox được hiển thị với câu hỏi xác nhận.
- **Kiểm tra lựa chọn của người dùng:** Nếu người dùng chọn "Yes", ứng dụng sẽ đóng form hiện tại và thoát. Nếu là form chính, ứng dụng sẽ được đóng hoàn toàn.

❖ ListView1

```
private void listView1_Click(object sender, EventArgs e)
{
    cbmaphong.Text = listView1.SelectedItems[0].SubItems[4].Text;
    txthovaten.Text = listView1.SelectedItems[0].SubItems[1].Text;
}
```

- **Cập nhật ComboBox cbmaphong:** Thiết lập giá trị của ComboBox cbmaphong bằng với giá trị của subitem thứ năm (chỉ số 4, bắt đầu từ 0) của mục được chọn trong listView1.

- **Cập nhật TextBox txtthovaten:** Thiết lập giá trị của TextBox txtthovaten bằng với giá trị của subitem thứ hai (chỉ số 1) của mục được chọn trong listView1.

❖ **BUTTON Xem Tình Trạng**

```
private void btnxemtinhttrang_Click(object sender, EventArgs e)
{
    listView2.Items.Clear();
    ketnoi.Open();
    sql = "SELECT MaPhong, TinhTrang FROM dbo.PHONG";
    thuchien = new SqlCommand(sql, ketnoi);
    docdulieu = thuchien.ExecuteReader();

    while (docdulieu.Read())
    {
        string maPhong = docdulieu["MaPhong"].ToString();
        string tinhTrang = docdulieu["TinhTrang"].ToString();

        ListViewItem item = new ListViewItem(maPhong);
        item.SubItems.Add(tinhTrang);
        listView2.Items.Add(item);
    }

    ketnoi.Close();
}
```

1. **Xóa danh sách hiện tại:** listView2.Items.Clear() được gọi để xóa các mục hiện có trong listView2, đảm bảo thông tin mới hiển thị là cập nhật và không trùng lặp.
2. **Mở kết nối cơ sở dữ liệu:** Sử dụng ketnoi.Open() để mở kết nối SQL.
3. **Thực hiện truy vấn SQL:**
 - Khởi tạo câu truy vấn: sql = "SELECT MaPhong, TinhTrang FROM dbo.PHONG" để lấy mã phòng và trạng thái từ bảng PHONG.
 - Tạo và thực thi lệnh SQL: thuchien = new SqlCommand(sql, ketnoi) và docdulieu = thuchien.ExecuteReader() để thực hiện truy vấn và lấy dữ liệu.
4. **Đọc và hiển thị dữ liệu:**
 - Sử dụng vòng lặp while (docdulieu.Read()) để duyệt qua từng hàng dữ liệu trả về.
 - Đối với mỗi hàng, lấy maPhong và tinhTrang từ đối tượng docdulieu.
 - Tạo một đối tượng ListViewItem mới và thêm vào listView2 để hiển thị.
5. **Đóng kết nối cơ sở dữ liệu:** Cuối cùng, ketnoi.Close() được gọi để đóng kết nối, giải phóng tài nguyên.

❖ BUTTON Đặt Phòng

```
private void btndatphong_Click_1(object sender, EventArgs e)
{
    // Lấy thông tin từ các controls
    string maKH = makh.Text; // Giả định rằng tên của TextBox là makh
    string hoTen = txthoten.Text; // Giả định rằng tên của TextBox là
txthoten
    string cccd = txtcccd.Text; // Giả định rằng tên của TextBox là txtcccd
    DateTime ngayNhanPhong = datengaynhanphong.Value; // Giả định rằng tên
của DateTimePicker là datengaynhanphong
    DateTime ngayTraPhong = datengaytraphong.Value; // Giả định rằng tên của
DateTimePicker là datengaytraphong
    string loaiPhong = cbloaiphong.Text; // Giả định rằng tên của ComboBox
là cbloaiphong

    // Thêm dữ liệu vào listView3
    var item = new ListViewItem(new[] {
        maKH,
        hoTen,
        cccd,
        ngayNhanPhong.ToString("yyyy-MM-dd"),
        ngayTraPhong.ToString("yyyy-MM-dd"),
        loaiPhong
    });

    listView3.Items.Add(item);

    // Kết nối và thêm dữ liệu vào cơ sở dữ liệu
    using (SqlConnection conn = new SqlConnection(chuoiiketnoi))
    {
        try
        {
            conn.Open();
            // Tạo SQL command để insert dữ liệu
            using (SqlCommand cmd = new SqlCommand("INSERT INTO dbo.booking
(MaKH, HoVaTen, SoCCCD, NgayNhanPhong, NgayTraPhong, LoaiPhong) VALUES (@MaKH, @HoVaTen,
@SoCCCD, @NgayNhanPhong, @NgayTraPhong, @LoaiPhong)", conn))
            {
                // Thêm các parameters vào command
                cmd.Parameters.AddWithValue("@MaKH", maKH);
                cmd.Parameters.AddWithValue("@HoVaTen", hoTen);
                cmd.Parameters.AddWithValue("@SoCCCD", cccd);
                cmd.Parameters.AddWithValue("@NgayNhanPhong",
ngayNhanPhong);
                cmd.Parameters.AddWithValue("@NgayTraPhong", ngayTraPhong);
                cmd.Parameters.AddWithValue("@LoaiPhong", loaiPhong);

                // Thực thi command
                cmd.ExecuteNonQuery();
                MessageBox.Show("Đặt phòng thành công và thông tin đã được
thêm vào danh sách!");
            }
        }
    }
}
```

1. Thu thập Dữ liệu Người Dùng:

- Lấy thông tin khách hàng từ các controls trên giao diện như TextBoxes và DateTimePickers, bao gồm mã khách hàng, họ tên, số CCCD, ngày nhận phòng, ngày trả phòng, và loại phòng.

2. Cập nhật Giao Diện Người Dùng:

- Tạo và thêm một ListViewItem vào listView3, hiển thị thông tin vừa nhập để người dùng có thể xem thông tin đặt phòng trực tiếp trên giao diện.

3. Lưu Trữ Dữ liệu vào Cơ sở dữ liệu:

- Mở kết nối đến cơ sở dữ liệu sử dụng SqlConnection.
- Sử dụng SqlCommand để tạo câu lệnh SQL INSERT, thêm thông tin đặt phòng vào bảng booking.
- Đặt các giá trị từ các controls vào các tham số của câu lệnh SQL và thực thi câu lệnh.

4. Xử lý Ngoại lệ:

- Bắt và xử lý các ngoại lệ SQL bằng cách hiển thị thông báo lỗi và xóa thông tin từ listView3 nếu có lỗi xảy ra trong quá trình thêm dữ liệu vào cơ sở dữ liệu.

5. Đóng kết nối:

- Đảm bảo kết nối đến cơ sở dữ liệu được đóng sau khi hoàn thành quá trình lưu trữ hoặc nếu có lỗi xảy ra.

❖ BUTTON Thêm Hóa Đơn

```
private void btnthemhoadon_Click(object sender, EventArgs e)
{
    // Lấy dữ liệu từ các control trên form
    string maHD = txtMaHoaDon.Text; // Lấy Mã Hóa Đơn từ TextBox
    string maKH = txtMaKH.Text; // Lấy Mã Khách Hàng từ TextBox
    DateTime ngayThanhToan = dateTimePickerNgayThanhToan.Value; // Lấy Ngày
Thanh Toán từ DateTimePicker
    string maPhong = comboBoxMaPhong.Text; // Lấy Mã Phòng từ ComboBox
    int tongTien; // Khởi tạo biến để chứa giá trị Tổng Tiền

    // Kiểm tra định dạng và chuyển đổi giá trị Tổng Tiền từ TextBox
    if (!int.TryParse(txtTongTien.Text, out tongTien))
    {
        MessageBox.Show("Tổng tiền phải là một số nguyên.");
        return;
    }

    // Tiến hành kết nối và thêm hóa đơn mới vào cơ sở dữ liệu
    using (SqlConnection conn = new SqlConnection(chuoiKetnoi))
    {
        try
        {
            conn.Open();
            string query = "INSERT INTO dbo.QLHoaDon (MaHD, NgayThanhToan,
TongTien, MaPhong, MaKH) VALUES (@MaHD, @NgayThanhToan, @TongTien, @MaPhong, @MaKH)";
            using (SqlCommand cmd = new SqlCommand(query, conn))
            {
                cmd.Parameters.AddWithValue("@MaHD", maHD);
                cmd.Parameters.AddWithValue("@NgayThanhToan",
ngayThanhToan);

                cmd.Parameters.AddWithValue("@TongTien", tongTien);
                cmd.Parameters.AddWithValue("@MaPhong", maPhong);
                cmd.Parameters.AddWithValue("@MaKH", maKH);
                int result = cmd.ExecuteNonQuery();
                if (result > 0)
                {
                    MessageBox.Show("Hóa đơn đã được thêm thành công.");
                }
                else
                {
                    MessageBox.Show("Không thêm được hóa đơn.");
                }
            }
        }
        catch (SqlException ex)
        {
            MessageBox.Show("Có lỗi khi thêm hóa đơn: " + ex.Message);
        }
        finally
        {
            conn.Close();
        }
    }

    // Gọi hàm hiển thị hóa đơn mới trên listView4
    hienthi();
}
```


- **INSERT INTO:** Từ khóa này bắt đầu một câu lệnh chèn mới, chỉ định rằng dữ liệu sẽ được thêm vào bảng.
- **dbo.QLHoaDon:** Là tên của bảng mà dữ liệu sẽ được chèn vào. dbo là schema mặc định trong SQL Server, và QLHoaDon là tên bảng.
- **(MaHD, NgayThanhToan, TongTien, MaPhong, MaKH):** Đây là danh sách các cột trong bảng QLHoaDon mà dữ liệu sẽ được chèn vào. Mỗi cột tương ứng với một phần thông tin của hóa đơn:
 - **MaHD:** Mã hóa đơn
 - **NgayThanhToan:** Ngày thanh toán của hóa đơn
 - **TongTien:** Tổng tiền phải thanh toán
 - **MaPhong:** Mã phòng liên quan đến hóa đơn
 - **MaKH:** Mã khách hàng phát sinh hóa đơn
- **VALUES (@MaHD, @NgayThanhToan, @TongTien, @MaPhong, @MaKH):**
Phần này của câu lệnh xác định các giá trị cụ thể sẽ được chèn vào các cột tương ứng. Các giá trị này được truyền dưới dạng tham số để tránh SQL injection và lỗi định dạng dữ liệu:
 - **@MaHD:** Tham số cho mã hóa đơn
 - **@NgayThanhToan:** Tham số cho ngày thanh toán
 - **@TongTien:** Tham số cho tổng tiền
 - **@MaPhong:** Tham số cho mã phòng
 - **@MaKH:** Tham số cho mã khách hàng

CHƯƠNG 3: THAO TÁC TRÊN GIAO DIỆN

3.1 Một Số Hoạt Động Ở Giao Diện Chính

- **Tìm Kiểm Theo Mã Phòng:**

The screenshot shows the 'QUẢN LÝ KHÁCH SẠN' (Hotel Management) interface. The 'Tìm Kiểm Nhanh' (Quick Search) section has 'Mã Phòng' (Room Code) set to 'Phong 02' and 'Họ và Tên' (Last Name and First Name) empty. The 'Thanh Toán' (Payment) section shows 'Phòng Khách Ở' (Guest Room) and 'Số Ngày Khách Ở' (Number of Days Guest Stayed). The 'Chi Tiết Khách Hàng Thanh Toán' (Guest Payment Details) table shows one record for room 'Phong 02'. The 'Xem Tình Trạng Phòng' (Room Status) section shows a list of rooms with status 'Đã Đặt' (Booked).

Mã KH	Họ và Tên	Số CCCD	Ngày Đến	Mã Phòng
KH02	Hoang Trung ...	23452342	4/23/2024 12:...	Phong 02

Hình 1.5: Tìm Kiểm Theo Mã Phòng

- **Tìm Kiểm Theo Họ Và Tên:**

The screenshot shows the 'QUẢN LÝ KHÁCH SẠN' (Hotel Management) interface. The 'Tìm Kiểm Nhanh' (Quick Search) section has 'Mã Phòng' (Room Code) set to 'Phong 01' and 'Họ và Tên' (Last Name and First Name) set to 'Bui Minh Nhu'. The 'Thanh Toán' (Payment) section shows 'Phòng Khách Ở' (Guest Room) and 'Số Ngày Khách Ở' (Number of Days Guest Stayed). The 'Chi Tiết Khách Hàng Thanh Toán' (Guest Payment Details) table shows one record for room 'Phong 01'. The 'Xem Tình Trạng Phòng' (Room Status) section shows a list of rooms with status 'Đã Đặt' (Booked).

Mã KH	Họ và Tên	Số CCCD	Ngày Đến	Mã Phòng
KH01	Bui Minh Nhu...	34522424	4/14/2024 12:...	Phong 01

Hình 1.6: Tìm Kiểm Theo Họ Và Tên

- **Tính Tiền Theo Đơn Giá Phòng:**

QUẢN LÝ KHÁCH SẠN

Tìm Kiếm Nhanh

Mã Phòng Phòng 01

Họ và Tên Bui Minh Nhat

Tìm Kiếm Theo Mã Phòng

Tìm Kiếm Theo Họ và Tên

Thanh Toán

Phòng Khách Ở Phòng 01

Tính Tiền Thoát

Số Ngày Khách Ở 12

Tổng Tiền Phải Thanh T 720000

Chi Tiết Khách Hàng Thanh Toán

Mã KH	Họ và Tên	Số CCCD	Ngày Đến	Mã Phòng
KH01	Bui Minh Nhat...	34522424	4/14/2024 12:...	Phòng 01

Xem Tình Trạng Phòng Đặt Phòng Quản Lý Hóa Đơn

Xem Tình Trạng

Đã Đặt

Có Khách

Trống

Đang Bảo Trì

Mã Phòng	Tình Trạng

Hình 1.7: Tính Tiền Theo Đơn Giá Phòng

- **Giao diện TabControl để xem Tình Trạng Phòng:**

Xem Tình Trạng Phòng
Đặt Phòng
Quản Lý Hóa Đơn

Xem Tình Trạng

Đã Đặt

Có Khách

Trống

Đang Bảo Trì

Mã Phòng	Tình Trạng
Phòng 01	Có Khách
Phòng 02	Trống
Phòng 03	Trống
Phòng 04	Đã Đặt
Phòng 05	Trống
Phòng 06	Bảo Trì
Phòng 07	Trống

Hình 1.8: Giao Diện TabControl Để Xem Tình Trạng Phòng

- Thay Đổi hoặc Xóa Thông Tin Khách Đặt Phòng

Xem Tình Trạng Phòng

Đặt Phòng

Quản Lý Hóa Đơn

Mã KH

KH08

Họ và Tên

Gia Bao

Số CCCD

3453424

Ngày Nhận Phòng

Friday . April 26, 2024

Ngày Trả Phòng

Saturday . April 27, 2024

Loại Phòng

2

Đặt Phòng

Xóa

Mã KH	Họ và Tên	Số CCCD	Ngày Nh...	Ngày Trả...	Loại Phòng
KH08	Gia Bao	3453424	2024-04-26	2024-04-27	2

Đặt phòng thành công và thông tin đã được thêm vào danh sách!

OK

Hình 1.9: Thay Đổi Thông Tin Khách Đặt Phòng

Xem Tình Trạng Phòng

Đặt Phòng

Quản Lý Hóa Đơn

Mã KH

KH08

Họ và Tên

Gia Bao

Số CCCD

3453424

Ngày Nhận Phòng

Friday . April 26, 2024

Ngày Trả Phòng

Saturday . April 27, 2024

Loại Phòng

2

Đặt Phòng

Xóa

Mã KH	Họ và Tên	Số CCCD	Ngày Nh...	Ngày Trả...	Loại Phòng
KH08	Gia Bao	3453424	2024-04-26	2024-04-27	2

Xác nhận xóa

?

Bạn có chắc chắn muốn xóa thông tin của khách hàng này?

Yes

No

Hình 1.10: Xóa Thông Tin Khách Đặt Phòng

- Quản Lý Hóa Đơn

Xem Tình Trạng PhòngĐặt PhòngQuản Lý Hóa Đơn

Mã Hóa Đơn

HD003

Mã KH

KH01

Ngày Thanh Toán

Friday . April 26, 2024

Mã Phòng

Phòng 01

Tổng Tiền

720000

Thêm Hóa Đơn

Xóa

Mã Hóa Đơn	Ngày Thanh Toán	Tổng Tiền	Mã Phòng	Mã KH
HD003	4/26/2024 12:00:0...	720000	Phòng 01	KH01

Hình 1.11: Quản Lý Hóa Đơn Khách Hàng

3.2 Chạy Thử Phần Mềm (QR_Code + File)



DANH MỤC HÌNH ẢNH

Hình 1.1: Kết Nối Đến Server	8
Hình 1.2: Thêm Kết Nối	8
Hình 1.3: Các Bảng Đã Được Thêm Vào DataSet	9
Hình 1.4: Giao Diện Của Phần Mềm	9
Hình 1.5: Tìm Kiếm Theo Mã Phòng	22
Hình 1.6: Tìm Kiếm Theo Họ Và Tên.....	22
Hình 1.7: Tính Tiền Theo Đơn Giá Phòng	23
Hình 1.8: Giao Diện TabControl Để Xem Tình Trạng Phòng	23
Hình 1.9: Thay Đổi Thông Tin Khách Đặt Phòng	24
Hình 1.10: Xóa Thông Tin Khách Đặt Phòng	24

DANH MỤC BẢNG

Bảng 1.1: Design Table(dbo.KHACHHANG).....	4
Bảng 1.2: Dữ Liệu Của Bảng(dbo.KHACHHANG).....	4
Bảng 1.3: Design Table(dbo.PHONG).....	5
Bảng 1.4: Dữ Liệu Của Bảng(dbo.PHONG).....	5
Bảng 1.5: Design Table(dbo.booking).	6
Bảng 1.6: Dữ Liệu Của Bảng(dbo.booking).....	6
Bảng 1.7: Design Table(dbo.QLHoaDon).....	7
Bảng 1.8: Dữ Liệu Của Bảng(dbo.QLHoaDon).....	7

DANH MỤC TÀI LIỆU THAM KHẢO

1. Nguyen, P. T., & Ho, T. M. (2023). *Design and Implementation of Hotel Room Management System Based on C*. Truy cập ngày 23 tháng 4 năm 2024, từ: https://www.researchgate.net/publication/366458460_Design_and_Implementati_on_of_Hotel_Room_Management_System_Based_on_C
2. UserA (2018). Room management program in C#. [Online forum comment]. Stack Overflow. Truy cập ngày 23 tháng 4 năm 2024, từ: <https://stackoverflow.com/questions/50075283/room-management-program-in-c-sharp>
3. Thanh, P. T. (n.d.). Hotel Management [Source code]. GitHub. Truy cập ngày 23 tháng 4 năm 2024, từ: <https://github.com/thanhpt1110/hotel-management>
4. QSProg (2022). C# Tutorial for Beginners: Hotel Management. QS Programming Blog. Truy cập ngày 23 tháng 4 năm 2024, từ: <https://qsprog.blogspot.com/2022/04/c-tutorial-for-beginners-hotel.html>
5. ASP.NET Buying Guide (n.d.). Hotel Management System Source Code (ASP.NET, C#). Truy cập ngày 23 tháng 4 năm 2024, từ: <https://buyasp.net/hotel-management-system-source%20code-dot-net-asp-net-c-sharp>

PHỤ LỤC 1

DANH SÁCH NHÓM 4

STT	Họ và tên	MSSV	Nhiệm vụ	Hoàn thành
01	Bùi Minh Nhật	21207070	Thiết Kế Giao Diện	100%
02	Huỳnh Lê Trí Cường	21207128	Tìm Tài Liệu Về Đồ Án	100%
03	Hoàng Trung Hiếu	21207030	Viết Báo Cáo	100%