

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
KHOA CÔNG NGHỆ THÔNG TIN



BÀI TẬP LỚN
HỌC PHẦN : LẬP TRÌNH TRỰC QUAN
Đề tài : XÂY DỰNG HỆ THỐNG QUẢN LÝ
HOMESTAY

Giảng viên hướng dẫn : Cô Cao Thị Luyện

Nhóm sinh viên thực hiện : Nhóm 10 – CNTT4 K64

1	Nguyễn Hoàng Tùng	231230950
2	Trần Công Việt	231230960
3	Hoàng Tiến Đạt	231230737
4	Phạm Minh Hiếu	231220778

Mục Lục

Mục Lục	1
Danh Mục Bảng Và Sơ Đồ	3
Chương 1. Giới thiệu về đề tài	4
1.1. Giới thiệu về đề tài	4
1.1.1. Lý do chọn đề tài	4
1.1.2. Mục tiêu của đề tài	4
1.1.3. Đối tượng và phạm vi ứng dụng	5
1.1.4. Ý nghĩa của đề tài	6
1.2 Kế hoạch làm đề tài	6
1.2.1. Mục tiêu tổng quát	6
1.2.2. Các giai đoạn thực hiện	6
1.2.3. Phân công công việc	7
1.3 Các công nghệ sử dụng	8
1.3.1. Ngôn ngữ lập trình	8
1.3.2. Giao diện người dùng – WinForms	8
1.3.3. Cơ sở dữ liệu	9
1.3.4. Công cụ và môi trường phát triển	12
1.3.5. Một số thư viện hỗ trợ	12
Chương 2. Mô hình hệ thống	14
2.1. Các chức năng của hệ thống	14
2.1.1. Chức năng Báo cáo Doanh thu theo tháng/năm	14
2.1.2. Chức năng Quản lý Đặt phòng	14
2.1.3. Chức năng Đổi mật khẩu	15
2.1.4. Chức năng Quản lý Hóa đơn	15

2.1.5. Chức năng Quản lý Khách hàng	16
2.1.6. Chức năng Quản lý Loại Phòng.....	17
2.1.7. Chức năng Đăng nhập (FrmLogin).....	17
2.1.8.Chức năng Chính (FrmMain)	18
2.1.9.Chức năng Quản lý Nhân viên.....	18
2.1.10.Chức năng Quản lý Phòng.....	19
2.2 Các giao diện có trong hệ thống.	20
2.3 Mô hình lớp.	44
2.4. Mô hình cơ sở dữ liệu.	46
2.5 Các tiện ích.	47
Chương 3. Phát triển hệ thống.	50
3.1. Lớp GUI	50
3.2 Lớp BUS.....	56
3.3. Lớp DAL.	61
3.4 Các Procedured.....	71
3.5 Lớp DTO	74
3.6 Kết chương.	79
Kết luận.....	80
Hạn chế và hướng phát triển của đề tài.....	81
Danh mục sách tham khảo	82

Danh Mục Bảng Và Sơ Đồ

Bảng 1.1 : Bảng Giai Đoạn của Dự Án	6
Bảng 2.1 :Giao Diện FrmMain (Form chính)	20
Bảng 2.2: FrmLogin (Form đăng nhập).....	22
Bảng 2.3: FrmPhong (Form quản lý phòng).....	24
Bảng 2.4: FrmLoaiPhong (Form quản lý loại phòng).....	27
Bảng 2.5: FrmKhachHang (Form quản lý khách hàng)	28
Bảng 2.6: FrmNhanVien (Form quản lý nhân viên)	32
Bảng 2.7: FrmDatPhong (Form đặt phòng)	34
Bảng 2.8: FrmHoaDon (Form quản lý hóa đơn).....	37
Bảng 2.9: FrmBaoCaoDoanhThu (Form báo cáo doanh thu)	40
Bảng 2.10: Giao Diện FrmDoiMatKhau (Form đổi mật khẩu)	42
Bảng 2.2 : Bảng Mô Hình Lớp.....	44
Sơ đồ 1.1 : Sơ Đồ RM của DataBase	46

Chương 1. Giới thiệu về đề tài

1.1. Giới thiệu về đề tài.

1.1.1. Lý do chọn đề tài

Trong bối cảnh xã hội hiện đại, ngành du lịch và dịch vụ **homestay** đang ngày càng phát triển mạnh mẽ, đặc biệt tại các thành phố lớn và khu du lịch ở Việt Nam. Cùng với sự gia tăng về số lượng khách lưu trú, các cơ sở **homestay** phải đối mặt với khối lượng công việc quản lý ngày càng lớn, bao gồm việc đặt phòng, thanh toán, quản lý khách hàng, nhân viên, dịch vụ, hóa đơn,...

Nếu việc quản lý này được thực hiện thủ công bằng sổ sách hoặc Excel, dễ dẫn đến sai sót, trùng lặp dữ liệu và mất nhiều thời gian xử lý.

Chính vì vậy, việc xây dựng phần mềm quản lý **homestay** là hết sức cần thiết. Phần mềm này giúp tự động hóa các công việc quản lý, giảm thiểu sai sót, tăng hiệu suất làm việc và giúp chủ homestay có thể theo dõi hoạt động kinh doanh dễ dàng, nhanh chóng và chính xác hơn.

1.1.2. Mục tiêu của đề tài

Mục tiêu chính của đề tài là xây dựng **phần mềm quản lý homestay** trên nền tảng **Windows Forms (C#)** kết hợp với **cơ sở dữ liệu SQL Server**, giúp quản lý thống nhất thông tin phòng, khách hàng, nhân viên, đặt phòng và hóa đơn một cách hiệu quả, nhanh chóng và chính xác.

Cụ thể, hệ thống cần đáp ứng các yêu cầu sau:

Quản lý thông tin phòng:

Lưu trữ và quản lý thông tin các phòng bao gồm mã phòng, tên phòng, loại phòng, giá phòng và trạng thái sử dụng.

Quản lý loại phòng:

Quản lý danh mục loại phòng (ví dụ: đơn, đôi, VIP), mô tả và giá tương ứng cho từng loại phòng.

Quản lý khách hàng:

Quản lý thông tin khách hàng như họ tên, giới tính, số CMND/CCCD, số điện thoại,

giúp tra cứu lịch sử đặt phòng của từng khách.

Quản lý nhân viên và tài khoản đăng nhập:

Quản lý thông tin nhân viên (mã, tên, giới tính, ngày sinh, địa chỉ) và tài khoản đăng nhập tương ứng (tên đăng nhập, mật khẩu, quyền truy cập) để đảm bảo phân quyền trong hệ thống.

Quản lý đặt phòng:

Lưu thông tin đặt phòng của khách hàng bao gồm ngày đặt, ngày đến, ngày đi, nhân viên thực hiện và chi tiết các phòng được đặt.

Quản lý hóa đơn:

Tạo hóa đơn tương ứng cho từng phiếu đặt phòng, ghi nhận ngày lập và tổng tiền cần thanh toán.

Báo cáo và thống kê:

Hỗ trợ thống kê doanh thu dựa trên dữ liệu hóa đơn theo ngày, tháng hoặc năm, giúp người quản lý dễ dàng theo dõi tình hình kinh doanh.

Tìm kiếm và tra cứu nhanh:

Cho phép tìm kiếm thông tin phòng, khách hàng, hóa đơn, nhân viên hoặc phiếu đặt phòng nhanh chóng, phục vụ cho công tác quản lý và tra cứu.

1.1.3. Đối tượng và phạm vi ứng dụng

Phần mềm được thiết kế hướng đến:

Các **homestay vừa và nhỏ**, mong muốn tin học hóa quy trình quản lý.

Các **chủ homestay** muốn dễ dàng kiểm soát hoạt động kinh doanh.

Nhân viên lễ tân, kế toán, quản lý phòng và bộ phận kỹ thuật có thể thao tác nhanh chóng và chính xác.

Phạm vi ứng dụng:

Phần mềm hoạt động trên **máy tính cá nhân chạy Windows**, không yêu cầu cấu hình cao, phù hợp cho môi trường **làm việc nội bộ trong homestay**

1.1.4. Ý nghĩa của đề tài

- **Về mặt thực tiễn:**

Giúp tin học hóa quy trình quản lý, tiết kiệm thời gian, nâng cao năng suất làm việc và độ chính xác trong công việc.

- **Về mặt học tập:**

Giúp người thực hiện củng cố kiến thức lập trình C#, kỹ năng thiết kế cơ sở dữ liệu, mô hình hóa bài toán thực tế, và phát triển phần mềm hoàn chỉnh theo quy trình.

- **Về mặt kỹ thuật:**

Là cơ hội để áp dụng mô hình lập trình hướng đối tượng (OOP), kết nối cơ sở dữ liệu (ADO.NET), xử lý sự kiện, giao diện đồ họa, và tư duy thiết kế hệ thống phần mềm.

1.2 Kế hoạch làm đề tài.

1.2.1. Mục tiêu tổng quát

Xây dựng phần mềm “Quản lý homestay” có đầy đủ các chức năng cơ bản, hoạt động ổn định, giao diện trực quan, dễ thao tác, và có khả năng mở rộng về sau.

1.2.2. Các giai đoạn thực hiện

Bảng 1.1 : Bảng Giai Đoạn của Dự Án

Giai đoạn	Thời gian dự kiến	Nội dung công việc
Giai đoạn 1: Khảo sát và phân tích yêu cầu	Tuần 1 – Tuần 2	<ul style="list-style-type: none">• Tìm hiểu quy trình quản lý homestay thực tế.• Phân tích yêu cầu chức năng và phi chức năng.• Xây dựng các sơ đồ Use Case, DFD và ERD.
Giai đoạn 2: Thiết kế hệ thống	Tuần 3 – Tuần 4	<ul style="list-style-type: none">• Thiết kế cơ sở dữ liệu.• Thiết kế giao diện người

		dùng (UI). • Thiết kế các lớp và mối quan hệ giữa chúng theo hướng đối tượng (OOP).
Giai đoạn 3: Cài đặt và lập trình	Tuần 5 – Tuần 7	• Xây dựng các module: Quản lý phòng, khách hàng, nhân viên, hóa đơn, đặt phòng. • Kết nối cơ sở dữ liệu với C# thông qua ADO.NET. • Xử lý sự kiện và các chức năng trên giao diện.
Giai đoạn 4: Kiểm thử và hoàn thiện	Tuần 8 – Tuần 9	• Kiểm tra tính đúng đắn của các chức năng. • Hiệu chỉnh giao diện, tối ưu hiệu suất. • Ghi nhận và khắc phục lỗi trong quá trình vận hành thử.
Giai đoạn 5: Báo cáo và nộp đề tài	Tuần 10	• Viết tài liệu hướng dẫn sử dụng và báo cáo kết quả thực hiện. • Nộp sản phẩm hoàn chỉnh và trình bày trước hội đồng.

1.2.3. Phân công công việc

Nhóm em phân công công việc cụ thể như sau:

- **Thành viên 1 - Đạt:** Phân tích yêu cầu, thiết kế cơ sở dữ liệu ,làm báo cáo chương 1 và 3, làm slide thuyết trình , làm giao diện và xử lý sự kiện frm Hóa Đơn và frm Báo Cáo , Xuất Excel.
- **Thành viên 2 – Hiếu:** Làm giao diện và xử lý sự kiện frm Nhân Viên và frm Khách hàng, làm báo cáo chương 2
- **Thành viên 3 – Việt:** Kết nối cơ sở dữ liệu Sql, làm giao diện và xử lý sự kiện frm Main và frm Login.
- **Thành viên 4 – Tùng :** Chia việc cho thành viên , phân luồng app, design giao diện và xử lý sự kiện frm Phòng, frm Đặt phòng, frm Loại phòng, validate, kiểm thử , làm slide thuyết trình, phân tách code theo mô hình 3 lớp, thuyết trình.

1.3 Các công nghệ sử dụng.

1.3.1. Ngôn ngữ lập trình

Đề tài sử dụng C# (C Sharp) – một ngôn ngữ lập trình hướng đối tượng mạnh mẽ của Microsoft, chạy trên nền .NET Framework.

Ưu điểm:

Cú pháp rõ ràng, dễ đọc.

Hỗ trợ lập trình hướng đối tượng đầy đủ.

Tích hợp tốt với Windows Forms.

Hỗ trợ thư viện phong phú cho giao diện và cơ sở dữ liệu.

1.3.2. Giao diện người dùng – WinForms

Windows Forms là công nghệ xây dựng ứng dụng desktop truyền thống của Microsoft.

Ưu điểm:

Dễ thiết kế giao diện bằng kéo thả (Drag & Drop).

Tương thích tốt với hệ điều hành Windows.

Dễ bảo trì và mở rộng.

Hỗ trợ các control phong phú: TextBox, ComboBox, DataGridView, Button,...

Ứng dụng trong đề tài Quản lý homestay:

Trong phần mềm quản lý homestay, WinForms được sử dụng để thiết kế:

Form đăng nhập: cho phép người dùng nhập tài khoản, mật khẩu.

Form quản lý phòng: hiển thị danh sách các phòng, tình trạng, giá, loại phòng.

Form quản lý khách hàng: thêm, sửa, xóa, tìm kiếm thông tin khách hàng.

Form hóa đơn – thanh toán: tạo hóa đơn, tính tổng tiền, xuất hóa đơn.

Form thống kê doanh thu: Danh sách hóa đơn theo thời gian.

Nhờ WinForms, các form này đều có giao diện thân thiện, dễ thao tác, phù hợp với nhân viên homestay – kể cả những người không am hiểu sâu về công nghệ.

1.3.3. Cơ sở dữ liệu

Hệ thống **Quản lý homestay** sử dụng **SQL Server** để quản lý dữ liệu.

Ưu điểm:

- Lưu trữ dữ liệu an toàn, có khả năng mở rộng.
- Tích hợp tốt với C# thông qua ADO.NET.
- Dễ truy vấn, thống kê, và bảo mật thông tin khách hàng.

Mô hình cơ sở dữ liệu

Cơ sở dữ liệu được thiết kế theo **mô hình quan hệ (Relational Database)**.

Các bảng có mối liên hệ với nhau thông qua **khóa chính (Primary Key)** và **khóa ngoại (Foreign Key)**.

Các bảng chính bao gồm:

1. Bảng LoaiPhong

→ Quản lý thông tin về loại phòng trong homestay.

Thuộc tính:

- MaLoaiPhong (*Primary Key*)
- TenLoai
- GiaPhong
- MoTa

2. Bảng Phong

→ Lưu thông tin chi tiết các phòng trong homestay.

Thuộc tính:

- MaPhong (*Primary Key*)
- TenPhong
- MaLoaiPhong (*Foreign Key* → *LoaiPhong*)
- TrangThai (*Phòng trống, đã đặt, đang sử dụng...*)

3. Bảng KháchHang

→ Lưu thông tin khách hàng thuê phòng.

Thuộc tính:

- MaKH (*Primary Key*)
- TenKH
- GioiTinh
- SDT
- CMND

4. Bảng NhanVien

→ Lưu thông tin nhân viên homestay.

Thuộc tính:

- MaNV (*Primary Key*)
- TenNV
- GioiTinh
- NgaySinh
- DiaChi

5. Bảng TaiKhoan

→ Quản lý thông tin đăng nhập của nhân viên.

Thuộc tính:

- TenDangNhap (*Primary Key*)

- MatKhu
- MaNV (*Foreign Key* → NhanVien)
- Quyen (*phân quyền người dùng: Admin, Lễ tân...*)

6. Bảng DatPhong

→ Lưu thông tin các lần khách hàng đặt phòng.

Thuộc tính:

- MaDatPhong (*Primary Key*)
- MaKH (*Foreign Key* → KhachHang)
- MaNV (*Foreign Key* → NhanVien)
- MaPhong (*Foreign Key* → Phong)
- NgayDat
- NgayDen
- NgayDi

7. Bảng ChiTietDatPhong

→ Lưu thông tin chi tiết các phòng trong mỗi đơn đặt phòng.

Thuộc tính:

- MaDatPhong (*Foreign Key* → DatPhong)
- MaPhong (*Foreign Key* → Phong)
- DonGia

8. Bảng HoaDon

→ Lưu thông tin thanh toán cho các đơn đặt phòng.

Thuộc tính:

- MaHD (*Primary Key*)
- MaDatPhong (*Foreign Key* → DatPhong)
- NgayLap
- TongTien

Mối quan hệ giữa các bảng:

- **LoaiPhong – Phong:** 1 loại phòng có nhiều phòng $\rightarrow (1-N)$
- **Phong – ChiTietDatPhong:** 1 phòng có thể nằm trong nhiều chi tiết đặt phòng $\rightarrow (1-N)$
- **KhachHang – DatPhong:** 1 khách hàng có thể đặt nhiều phòng $\rightarrow (1-N)$
- **NhanVien – DatPhong:** 1 nhân viên có thể lập nhiều phiếu đặt phòng $\rightarrow (1-N)$
- **DatPhong – ChiTietDatPhong:** 1 đặt phòng có thể chứa nhiều chi tiết $\rightarrow (1-N)$
- **DatPhong – HoaDon:** 1 đặt phòng có thể có 1 hóa đơn thanh toán $\rightarrow (1-1)$
- **NhanVien – TaiKhoan:** 1 nhân viên có 1 tài khoản đăng nhập $\rightarrow (1-1)$

Vai trò của cơ sở dữ liệu

Cơ sở dữ liệu là trung tâm lưu trữ và xử lý dữ liệu của hệ thống.

Tất cả các chức năng như đặt phòng, thêm khách hàng, tạo hóa đơn, thống kê doanh thu... đều được thực hiện thông qua các truy vấn tới **SQL Server**.

Hệ thống đảm bảo:

- **Dữ liệu toàn vẹn, nhất quán và không trùng lặp.**
- **Tốc độ truy cập nhanh, dễ dàng mở rộng.**
- **Hỗ trợ bảo mật thông tin khách hàng và nhân viên.**

1.3.4. Công cụ và môi trường phát triển

Visual Studio 2022: IDE chính để viết, chạy và gỡ lỗi chương trình.

SQL Server Management Studio (SSMS): Quản lý cơ sở dữ liệu.

.NET Framework 4.7.2: Nền tảng chạy ứng dụng WinForms.

Crystal Reports hoặc RDLC: Dùng để tạo báo cáo, thống kê doanh thu.

1.3.5. Một số thư viện hỗ trợ

Trong quá trình phát triển phần mềm, hệ thống sử dụng nhiều thư viện của .NET Framework để hỗ trợ xử lý dữ liệu, tạo giao diện, xuất báo cáo và kết nối cơ sở dữ liệu, gồm:

System.Data.SqlClient: Kết nối và thao tác với cơ sở dữ liệu SQL Server (SELECT, INSERT, UPDATE, DELETE).

System.Drawing: Tùy chỉnh giao diện (màu sắc, hình ảnh, font, kích thước, biểu tượng...).

System.IO: Đọc/ghi file, hỗ trợ xuất – nhập dữ liệu.

ClosedXML / Microsoft.Office.Interop.Excel: Xuất – nhập dữ liệu Excel (ClosedXML không cần cài Office).

System.Windows.Forms: Thư viện chính của WinForms, cung cấp các control như Button, TextBox, DataGridView,...

System.ComponentModel: Hỗ trợ Data Binding, quản lý thuộc tính và sự kiện của đối tượng.

System.Linq: Truy vấn, lọc, sắp xếp dữ liệu nhanh bằng cú pháp LINQ.

System.Data: Cung cấp các đối tượng DataSet, DataTable, DataAdapter để xử lý dữ liệu tạm thời.

System.Configuration: Đọc/ghi chuỗi kết nối (Connection String) và thiết lập ứng dụng.

System.Globalization: Định dạng ngày tháng, tiền tệ theo chuẩn Việt Nam.

System.Threading.Tasks: Hỗ trợ xử lý bất đồng bộ, giúp ứng dụng mượt và không bị treo.

Chương 2. Mô hình hệ thống.

2.1. Các chức năng của hệ thống.

2.1.1. Chức năng Báo cáo Doanh thu theo tháng/năm

Mô tả: Thống kê doanh thu homestay từ hóa đơn trong tháng và năm được chọn. Hiện thị danh sách hóa đơn và tổng doanh thu.

Bao gồm:

- Chọn tháng, năm.
- Hiện thị các hóa đơn (Mã HĐ, Mã đặt phòng, Ngày lập, Tổng tiền).
- Tính tổng doanh thu.

Bảng liên quan: HoaDon

Chi tiết:

- **Mục đích:** Giúp quản lý theo dõi doanh thu để báo cáo và ra quyết định.
- **Người dùng:** Quản lý, kế toán.
- **Cách dùng:** Mở form, chọn tháng-năm, nhấn “Thống kê” để xem danh sách hóa đơn và tổng doanh thu.
- **Kết quả:** Hiện thị dữ liệu trực quan, cảnh báo nếu chưa chọn tháng hoặc năm.

2.1.2. Chức năng Quản lý Đặt phòng

Mô tả: Cho phép đặt, tìm kiếm, hủy (trả) phòng và lập hóa đơn cho khách hàng. Hiện thị danh sách phiếu đặt phòng và thông tin chi tiết liên quan.

Bao gồm:

- Tự động sinh mã đặt phòng.
- Chọn khách hàng, nhân viên, phòng (lọc theo phòng trống).
- Tự động hiện thị đơn giá phòng và tính tổng tiền theo số ngày.
- Chức năng thêm (đặt phòng), xóa (trả phòng), làm mới, tìm kiếm, lập hóa đơn.

Bảng liên quan: DatPhong, KhachHang, NhanVien, Phong, LoaiPhong, ChiTietDatPhong, HoaDon.

Chi tiết:

- **Mục đích:** Hỗ trợ nhân viên và quản lý thực hiện thao tác đặt phòng, theo dõi tình trạng phòng và tự động tạo hóa đơn.
- **Người dùng:** Nhân viên lễ tân, quản lý.
- **Cách dùng:** Mở form → chọn khách hàng, nhân viên, phòng, ngày đến/ngày đi → nhấn “Đặt phòng” hoặc các chức năng khác.
- **Kết quả:** Hệ thống hiển thị danh sách phiếu đặt phòng, tính tiền tự động, cập nhật trạng thái phòng và tạo hóa đơn tương ứng.

2.1.3. Chức năng Đổi mật khẩu

Mô tả: Cho phép nhân viên đổi mật khẩu đăng nhập của mình để bảo mật tài khoản.

Bao gồm:

- Nhập mật khẩu cũ.
- Nhập mật khẩu mới và xác nhận mật khẩu.
- Kiểm tra trùng khớp và lưu thay đổi.

Bảng liên quan: TaiKhoan

Chi tiết:

- **Mục đích:** Đảm bảo an toàn tài khoản và quyền truy cập của nhân viên.
- **Người dùng:** Nhân viên, quản lý.
- **Cách dùng:** Mở form → nhập mật khẩu cũ, mật khẩu mới và xác nhận → nhấn “Lưu”.
- **Kết quả:** Cập nhật mật khẩu mới, thông báo thành công hoặc lỗi nếu sai mật khẩu cũ/xác nhận không khớp.

2.1.4. Chức năng Quản lý Hóa đơn

Mô tả: Hiển thị, tìm kiếm, sửa và xóa thông tin hóa đơn. Admin có thể chỉnh sửa hoặc xóa hóa đơn; người dùng thường chỉ được xem và tìm kiếm.

Bao gồm:

- Hiện thị danh sách hóa đơn (Mã HĐ, Mã đặt phòng, Ngày lập, Tổng tiền).
- Chức năng **Tìm kiếm, Sửa, Xóa** hóa đơn.
- Giới hạn quyền thao tác theo vai trò người dùng (admin hoặc nhân viên).

Bảng liên quan: HoaDon

Chi tiết:

- **Mục đích:** Quản lý, tra cứu và chỉnh sửa thông tin hóa đơn thanh toán.
- **Người dùng:** Admin, nhân viên.
- **Cách dùng:** Mở form → xem danh sách hóa đơn → chọn hóa đơn để sửa/xóa hoặc nhập từ khóa để tìm kiếm.
- **Kết quả:** Hiện thị danh sách hóa đơn; admin có thể cập nhật hoặc xóa, nhân viên chỉ được xem và tìm kiếm.

2.1.5. Chức năng Quản lý Khách hàng

Mô tả: Quản lý thông tin khách hàng gồm thêm, sửa, xóa, tìm kiếm và hiện thị danh sách khách hàng. Hỗ trợ kiểm tra ràng buộc dữ liệu khi xóa (khách hàng đang được tham chiếu trong đặt phòng).

Bao gồm:

- Hiện thị danh sách khách hàng (Mã KH, Tên KH, Giới tính, SĐT, CMND/CCCD).
- Thêm, sửa, xóa, tìm kiếm khách hàng.
- Kiểm tra trùng mã, tự động gợi ý mã kế tiếp.
- Ràng buộc xóa khi khách hàng có dữ liệu liên quan trong bảng Đặt phòng.

Bảng liên quan: KhachHang, DatPhong

Chi tiết:

- Mục đích: Quản lý thông tin khách hàng để phục vụ tra cứu, đặt phòng và thống kê.
- Người dùng: Nhân viên lễ tân, quản lý.

- **Cách dùng:** Mở form → nhập thông tin khách hàng → chọn Thêm/Sửa/Xóa/Tìm kiếm → xem danh sách khách hàng cập nhật.
- **Kết quả:** Danh sách khách hàng hiển thị rõ ràng; thao tác cập nhật dữ liệu có kiểm tra ràng buộc và thông báo kết quả cụ thể.

2.1.6. Chức năng Quản lý Loại Phòng

Mô tả: Quản lý thông tin loại phòng bao gồm thêm, sửa, xóa, tìm kiếm và hiển thị danh sách loại phòng. Hỗ trợ kiểm tra dữ liệu hợp lệ và thông báo lỗi khi thao tác không hợp lệ.

Bao gồm:

- Hiển thị danh sách loại phòng (Mã loại, Tên loại, Giá cơ bản, Mô tả).
- Thêm, sửa, xóa loại phòng.
- Kiểm tra giá nhập hợp lệ.
- Cảnh báo khi xóa loại phòng có liên quan đến phòng trong homestay.

Bảng liên quan: LoaiPhong

Chi tiết:

- **Mục đích:** Quản lý thông tin các loại phòng để phục vụ đặt phòng và thống kê.
- **Người dùng:** Nhân viên lễ tân, quản lý.
- **Cách dùng:** Mở form → nhập thông tin loại phòng → chọn Thêm/Sửa/Xóa → xem danh sách cập nhật.
- **Kết quả:** Danh sách loại phòng hiển thị rõ ràng; thao tác thêm, sửa, xóa được kiểm tra dữ liệu và thông báo kết quả cụ thể.

2.1.7. Chức năng Đăng nhập (FrmLogin)

Mô tả: Cho phép người dùng đăng nhập vào hệ thống quản lý homestay bằng tên đăng nhập và mật khẩu. Hệ thống kiểm tra tính hợp lệ của dữ liệu và xác thực thông tin người dùng từ cơ sở dữ liệu.

Bao gồm:

- Nhập tên đăng nhập và mật khẩu.

- Kiểm tra dữ liệu nhập: thông báo nếu còn trống.
- Xác thực tài khoản với cơ sở dữ liệu.
- Thông báo đăng nhập thành công hoặc thất bại.
- Truyền thông tin người dùng (tên đăng nhập, quyền, mã nhân viên) sang FrmMain.
- Nút “Thoát” để đóng ứng dụng.

Bảng liên quan: TaiKhoan

Chi tiết:

- **Mục đích:** Bảo mật truy cập hệ thống, phân quyền người dùng.
- **Người dùng:** Nhân viên lễ tân, quản lý, admin.
- **Cách dùng:** Mở form → nhập tên đăng nhập và mật khẩu → nhấn Đăng nhập → nếu thành công, chuyển sang giao diện chính; nếu thất bại, hiện thông báo lỗi.
- **Kết quả:** Người dùng hợp lệ được vào FrmMain với quyền tương ứng; người dùng không hợp lệ được thông báo lỗi.

2.1.8.Chức năng Chính (FrmMain)

Mô tả: Giao diện chính cho phép truy cập các chức năng quản lý homestay như phòng, khách hàng, đặt phòng, hóa đơn, nhân viên, báo cáo và đổi mật khẩu. Phân quyền hiển thị theo vai trò người dùng.

Bao gồm:

- Hiển thị thông tin nhân viên đăng nhập.
- Các nút chức năng mở form tương ứng.
- Phân quyền ẩn/khóa nút theo vai trò.
- Đăng xuất và thoát chương trình với xác nhận.

Bảng liên quan: Không có.

Chi tiết:

- **Mục đích:** Trung tâm điều khiển hệ thống, phân quyền và bảo mật.
- **Người dùng:** Nhân viên, quản lý, admin.
- **Cách dùng:** Nhấn nút để mở form, đăng xuất hoặc thoát.
- **Kết quả:** Mở form chính xác, phân quyền đúng, thông báo xác nhận.

2.1.9.Chức năng Quản lý Nhân viên

Mô tả: Quản lý thông tin nhân viên gồm thêm, sửa, xóa, tìm kiếm và hiển thị danh sách nhân viên. Kiểm tra dữ liệu đầu vào và hiển thị thông báo khi thao tác không hợp lệ.

Bao gồm:

- Hiển thị danh sách nhân viên (Mã NV, Tên NV, Giới tính, Ngày sinh, Địa chỉ).
- Thêm mới, sửa, xóa nhân viên.
- Xử lý chọn dòng để hiển thị chi tiết lên form.
- Làm mới dữ liệu.

Bảng liên quan: NhanVien

Chi tiết:

- **Mục đích:** Quản lý thông tin nhân sự trong homestay.
- **Người dùng:** Nhân viên lễ tân, quản lý.
- **Cách dùng:** Mở form → nhập/sửa thông tin → chọn chức năng Thêm/Sửa/Xóa → xem kết quả trong bảng.
- **Kết quả:** Danh sách cập nhật theo thao tác, thông báo kết quả rõ ràng.

2.1.10.Chức năng Quản lý Phòng

Mô tả: Quản lý thông tin phòng gồm thêm, sửa, xóa, tìm kiếm và hiển thị danh sách phòng. Kiểm soát quyền người dùng (admin hoặc nhân viên) để cho phép hoặc giới hạn các thao tác.

Bao gồm:

- Hiển thị danh sách phòng (Mã phòng, Tên phòng, Loại phòng, Tình trạng).
- Thêm mới, sửa, xóa phòng với kiểm tra dữ liệu hợp lệ.
- Mở form quản lý loại phòng (dành cho admin).
- Quyền admin được toàn quyền, nhân viên chỉ sửa được tình trạng phòng.
- Tự động gợi ý mã phòng tiếp theo khi thêm.

Bảng liên quan: Phong, LoaiPhong

Chi tiết:

- **Mục đích:** Quản lý phòng trong homestay, phục vụ đặt phòng và báo cáo.
- **Người dùng:** Nhân viên lễ tân, quản lý (admin).
- **Cách dùng:** Mở form → nhập/sửa thông tin phòng → chọn Thêm/Sửa/Xóa → xem cập nhật trên bảng.
- **Kết quả:** Danh sách phòng hiển thị chính xác, thao tác được kiểm tra quyền và dữ liệu, thông báo rõ ràng.

2.2 Các giao diện có trong hệ thống.

Bảng 2.1 :Giao Diện FrmMain (Form chính)

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
1	FrmMain	Form	Form giao diện chính của hệ thống	- ClientSize: 638x411 - StartPosition: CenterScreen - BackgroundImageLayout: Stretch
2	lblNhanVien	Label	Hiển thị thông tin mã nhân viên đang đăng nhập	Font: Segoe UI, 10pt, Bold - Location: (15, 13) - BackColor: ActiveCaption - Text mặc định: "Mã nhân viên: (trống)"
3	btnPhong	Button	Nút mở chức năng quản lý phòng	- Size: 140x59 - Location: (38, 52) - BackColor: RGB(128, 128, 255) - Font: Verdana, 9.75pt, Bold - Text: "PHÒNG"
4	btnKhachHang	Button	Nút mở chức năng quản lý khách hàng	- Size: 140x60 - Location: (38, 115) - BackColor: RGB(128, 128, 255) - Font: Verdana, 9.75pt,

				Bold - Text: "KHÁCH HÀNG"
5	btnDatPhong	Button	Nút mở chức năng đặt phòng	- Size: 140x59 - Location: (182, 52) - BackColor: RGB(128, 128, 255) - Font: Verdana, 9.75pt, Bold - Text: "ĐẶT PHÒNG"
6	btnHoaDon	Button	Nút mở chức năng quản lý hóa đơn	Size: 140x60 - Location: (182, 115) - BackColor: RGB(128, 128, 255) - Font: Verdana, 9.75pt, Bold - Text: "HÓA ĐƠN"
7	btnNhanVien	Button	Nút mở chức năng quản lý nhân viên	- Size: 140x59 - Location: (38, 181) - BackColor: RGB(192, 255, 255) - Font: Verdana, 9.75pt, Bold - Text: "NHÂN VIÊN" - Chỉ Admin mới truy cập được
8	btnBaoCao	Button	Nút mở chức năng báo cáo thống kê	Size: 140x59 - Location: (182, 181) - BackColor: RGB(192, 255, 255) - Font: Verdana, 9.75pt, Bold

				- Text: "BÁO CÁO" - Chỉ Admin mới truy cập được
9	btnDoiMatKhau	Button	Nút mở chức năng đổi mật khẩu	- Size: 109x26 - Location: (425, 344) - BackColor: RGB(255, 128, 128) - Font: Segoe UI, 10pt - Text: "Đổi mật khẩu"
10	btnDangXuat	Button	Nút đăng xuất khỏi hệ thống	- Size: 109x26 - Location: (425, 374) - BackColor: RGB(255, 128, 128) - Font: Segoe UI, 10pt - Text: "Đăng xuất" - Quay về form đăng nhập
11	btnThoat	Button	Nút thoát ứng dụng	Size: 89x56 - Location: (538, 344) - BackColor: Red - Font: Verdana, 9.75pt, Bold - Text: "THOÁT" - Đóng toàn bộ ứng dụng

Bảng 2.2: FrmLogin (Form đăng nhập)

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
-----	---------------	------	---------	---------

1	FrmLogin	Form	Form đăng nhập hệ thống	<ul style="list-style-type: none"> - ClientSize: 463x311 - StartPosition: CenterScreen - FormBorderStyle: FixedDialog - MaximizeBox: false, MinimizeBox: false
2	lblTitle	Label	Tiêu đề "ĐĂNG NHẬP HỆ THỐNG"	<ul style="list-style-type: none"> - Font: Segoe UI, 16pt, Bold - Size: 296x38 - Location: (92, 9) - BackColor: RGB(128, 128, 255) - TextAlign: MiddleCenter
3	label1	Label	Tiêu đề "QUẢN LÝ HOMESTAY"	<ul style="list-style-type: none"> - Font: Segoe UI, 16pt, Bold - Size: 296x38 - Location: (92, 47) - BackColor: RGB(128, 128, 255) - TextAlign: MiddleCenter
4	lblUser	Label	Nhãn "Tên đăng nhập:"	<ul style="list-style-type: none"> - Location: (94, 159) - AutoSize: true - Text: "Tên đăng nhập:"
5	lblPass	Label	Nhãn "Mật khẩu:"	<ul style="list-style-type: none"> - Location: (94, 186) - AutoSize: true - Text: "Mật khẩu:"
6	txtUser	TextBox	Nhập tên đăng nhập	<ul style="list-style-type: none"> - Size: 214x20 - Location: (182, 156) - Nhận input username
7	txtPass	TextBox	Nhập mật khẩu	<ul style="list-style-type: none"> - Size: 214x20 - Location: (182, 183)

				- PasswordChar: '*' - Ẩn ký tự mật khẩu
8	btnLogin	Button	Nút đăng nhập	- Size: 103x33 - Location: (201, 222) - Text: "Đăng nhập" - Xác thực và đăng nhập vào hệ thống
9	btnExit	Button	Nút thoát	- Size: 88x33 - Location: (308, 222) - Text: "Thoát" - Đóng ứng dụng

Bảng 2.3: FrmPhong (Form quản lý phòng)

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
1	FrmPhong	Form	Form quản lý thông tin phòng	- ClientSize: 497x399 - StartPosition: CenterScreen - DoubleBuffered: true
2	lblMaPhong	Label	Nhãn "Mã phòng:"	- Font: Segoe UI, 9.75pt, Bold - Location: (26, 22) - AutoSize: true
3	txtMaPhong	TextBox	Nhập/hiển thị mã phòng	- Font: Segoe UI, 10pt - Size: 181x25 - Location: (111, 19)

4	lblTenPhong	Label	Nhãn "Tên phòng:"	Font: Segoe UI, 9.75pt, Bold - Location: (26, 52) - AutoSize: true
5	txtTenPhong	TextBox	Nhập/hiển thị tên phòng	- Font: Segoe UI, 10pt - Size: 181x25 - Location: (111, 49)
6	lblLoaiPhong	Label	Nhãn "Loại phòng:"	Font: Segoe UI, 9.75pt, Bold - Location: (26, 82) - AutoSize: true
7	cboLoaiPhong	ComboBox	Chọn loại phòng	Font: Segoe UI, 10pt - Size: 181x25 - Location: (111, 80) - DropDownStyle: DropDownList - Load dữ liệu từ bảng LoaiPhong
8	lblTinhTrang	Label	Nhãn "Tình trạng:"	- Font: Segoe UI, 9.75pt, Bold - Location: (26, 113) - AutoSize: true
9	cboTinhTrang	ComboBox	Chọn tình trạng phòng	- Font: Segoe UI, 10pt - Size: 181x25 - Location: (111, 110) - Items: "Trống", "Đang sử dụng", "Bảo trì"
10	dgvPhong	DataGridView	Hiển thị danh sách phòng	Font: Segoe UI, 10pt - Size: 81x26

				- Location: (26, 147) - Text: "Thêm"
11	btnThem	Button	Nút thêm phòng mới	- Font: Segoe UI, 10pt - Size: 98x26 - Location: (111, 147) - Text: "Sửa phòng"
12	btnSua	Button	Nút sửa thông tin phòng	- Font: Segoe UI, 10pt - Size: 77x26 - Location: (215, 147) - Text: "Xóa"

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
13	btnXoa	Button	Nút xóa phòng	Font: Segoe UI, 9.75pt, Bold - Size: 154x30 - Location: (317, 26) - Text: "Làm mới"
14	btnLamMoi	Button	Nút làm mới form	- Font: Segoe UI, 9.75pt, Bold - Size: 154x30 - Location: (317, 65) - Text: "Mở quản lý loại phòng"
15	btnMoLoaiPhong	Button	Nút mở form quản lý loại phòng	- Size: 473x208 - Location: (12, 179) - AutoSizeColumnsMode: Fill - SelectionMode: FullRowSelect


Bảng 2.4: FrmLoaiPhong (Form quản lý loại phòng)

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
1	FrmLoaiPhong	Form	Form quản lý loại phòng	- ClientSize: 600x338 - DoubleBuffered: true
2	lblMaLoaiPhong	Label	Nhãn "Mã loại phòng:"	- Font: Verdana, 12pt, Bold - Location: (11, 16) - AutoSize: true
3	txtMaLoaiPhong	TextBox	Nhập/hiển thị mã loại phòng	- Size: 146x20 - Location: (150, 14)
4	lblTenLoaiPhong	Label	Nhãn "Tên loại phòng:"	- Font: Verdana, 12pt, Bold - Location: (300, 16) - AutoSize: true
5	txtTenLoaiPhong	TextBox	Nhập/hiển thị tên loại phòng	- Size: 149x20 - Location: (447, 16)
6	lblMoTa	Label	Nhãn "Mô tả:"	Font: Verdana, 12pt, Bold - Location: (11, 44) - AutoSize: true
7	txtMoTa	TextBox	Nhập mô tả loại phòng	- Size: 218x40 - Location: (78, 43) - Multiline: true
8	lblGiaCoBan	Label	Nhãn "Giá cơ bản:"	- Font: Verdana, 12pt, Bold - Location: (300, 46) - AutoSize: true






9	txtGiaCoBan	TextBox	Nhập giá cơ bản	- Size: 149x20 - Location: (447, 44) - Nhập số tiền
10	dgvLoaiPhong	DataGridView	Hiển thị danh sách loại phòng	Size: 90x26 - Location: (22, 94) - Text: "Thêm"
11	btnThem	Button	Nút thêm loại phòng	- Size: 90x26 - Location: (128, 94) - Text: "Sửa"
12	btnSua	Button	Nút sửa loại phòng	Size: 90x26 - Location: (232, 94) - Text: "Xóa"
13	btnXoa	Button	Nút xóa loại phòng	- Size: 90x26 - Location: (338, 94) - Text: "Làm mới"
14	btnLamMoi	Button	Nút làm mới	Size: 555x182 - Location: (22, 136)

Bảng 2.5: FrmKhachHang (Form quản lý khách hàng)


STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
1	FrmKhachHang	Form	Form quản lý khách hàng	Size: 950x640; BackColor: WhiteSmoke; Font: Segoe UI 10pt

2	pnlHeader	Panel	Panel chứa tiêu đề	Size: 950x60; BackColor: RGB(0,123,167); Dock: Top
3	lblHeader	Label	Tiêu đề "  QUẢN LÝ KHÁCH HÀNG"	Font: Segoe UI 18pt Bold; Dock: Fill; TextAlign: MiddleCenter
4	pnlThongTin	Panel	Panel chứa thông tin khách hàng	Size: 870x180; Location: (40,80); BackColor: White; Anchor: Top, Left, Right
5	lblMaKH	Label	Nhãn "Mã khách hàng:"	Location: (20,50)
6	txtMaKH	TextBox	Nhập/hiển thị mã khách hàng	Size: 180x27; Location: (140,46)
7	lblTenKH	Label	Nhãn "Tên khách hàng:"	Location: (360,50)
8	txtTenKH	TextBox	Nhập/hiển thị tên khách hàng	Size: 180x27; Location: (480,46)
9	lblGioiTinh	Label	Nhãn "Giới tính:"	Location: (690,50); Anchor: Top, Right
10	cboGioiTinh	ComboBox	Chọn giới tính	DropDownStyle: DropDownList; Size: 100x27; Location:

				(760,46); Anchor: Top, Right
11	lblSDT	Label	Nhãn "Số điện thoại:"	Location: (20,95)
12	txtSDT	TextBox	Nhập số điện thoại	Size: 180x27; Location: (140,91)
13	lblCMND	Label	Nhãn "CMND/CCCD:"	Location: (360,95)
14	txtCMND	TextBox	Nhập CMND/CCCD	Size: 180x27; Location: (480,91)
15	lblTimKiem	Label	Nhãn "Tìm mã:"	Location: (680,95); Anchor: Top, Right
16	txtTimKiemMaKH	TextBox	Nhập mã tìm kiếm	Size: 90x27; Location: (770,92); Anchor: Top, Right
17	dgvKhachHang	DataGridView	Hiển thị danh sách khách hàng	Size: 870x220; Location: (40,340); SelectionMode: FullRowSelect; ReadOnly: true; Anchor: All
18	btnThem	Button	Nút thêm khách hàng	Text: "✚ Thêm"; BackColor: RGB(196,226,208); Font: Segoe UI Emoji 10pt Bold; Size: 120x40; Location: (100,280)

19	btnSua	Button	Nút sửa khách hàng	Text: "  Sửa"; BackColor: RGB(206,218,238); Size: 120x40; Location: (250,280)
20	btnXoa	Button	Nút xóa khách hàng	Text: "  Xóa"; BackColor: RGB(238,198,198); Size: 120x40; Location: (400,280)
21	btnLamMoi	Button	Nút làm mới	Text: "  Làm mới"; BackColor: RGB(219,207,236); Size: 120x40; Location: (550,280)
22	btnTimKiem	Button	Nút tìm kiếm	Text: "  Tìm"; BackColor: RGB(250,239,185); Size: 120x40; Location: (700,280)
23	btnThoat	Button	Nút thoát	Text: "  Thoát"; BackColor: RGB(204,204,204); Location: (820,580); Anchor: Bottom, Right

Bảng 2.6: FrmNhanVien (Form quản lý nhân viên)

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
1	FrmNhanVien	Form	Form quản lý nhân viên	Size: 880x640; BackColor: WhiteSmoke; Font: Segoe UI 10pt
2	headerPanel	Panel	Panel chứa tiêu đề	Size: 880x60; BackColor: RGB(0,123,255); Dock: Top
3	lblHeader	Label	Tiêu đề "  QUẢN LÝ NHÂN VIÊN"	Font: Segoe UI 18pt Bold; Dock: Fill; TextAlign: MiddleCenter
4	pnlThongTin	Panel	Panel thông tin nhân viên	Size: 800x200; Location: (40,90); BackColor: White; Anchor: Top, Left, Right
5	lblTenNV	Label	Nhãn "Tên nhân viên:"	Location: (20,50)
6	txtTenNV	TextBox	Nhập/hiển thị tên nhân viên	Size: 220x27; Location: (140,45)
7	lblGioiTinh	Label	Nhãn "Giới tính:"	Location: (400,50)
8	cboGioiTinh	ComboBox	Chọn giới tính	Items: "Nam", "Nữ"; DropDownStyle:

				DropDownList; Size: 140x27; Location: (470,45)
9	lblNgaySinh	Label	Nhãn "Ngày sinh:"	Location: (20,95)
10	dtpNgaySinh	DateTimePicker	Chọn ngày sinh	Format: Short; Size: 150x27; Location: (140,90)
11	lblDiaChi	Label	Nhãn "Địa chỉ:"	Location: (400,95)
12	txtDiaChi	TextBox	Nhập địa chỉ	Size: 280x27; Location: (470,90)
13	lblTimKiemNV	Label	Nhãn "Tìm kiếm (ID):"	Location: (20,140)
14	txtTimKiemNV	TextBox	Nhập ID tìm kiếm	Size: 220x27; Location: (140,135)
15	btnTimKiem	Button	Nút tìm kiếm	Text: "🔍 Tìm"; BackColor: RGB(222,239,230); Size: 90x32; Location: (370,133)
16	dgvNhanVien	DataGridView	Hiển thị danh sách nhân viên	Size: 800x250; Location: (40,350); BackColor: White; RowTemplate.Height: 38
17	btnThem	Button	Nút thêm nhân viên	Text: "➕ Thêm"; BackColor: RGB(222,239,230); Size:

				120x40; Location: (100,310)
18	btnSua	Button	Nút sửa nhân viên	Text: "✎ Sửa"; BackColor: RGB(230,236,245); Size: 120x40; Location: (250,310)
19	btnXoa	Button	Nút xóa nhân viên	Text: "🗑 Xóa"; BackColor: RGB(247,225,225); Size: 120x40; Location: (400,310)
20	btnLamMoi	Button	Nút làm mới	Text: "🔄 Làm mới"; BackColor: RGB(236,229,243); Size: 120x40; Location: (550,310)

Bảng 2.7: FrmDatPhong (Form đặt phòng)

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
1	FrmDatPhong	Form	Form quản lý đặt phòng	ClientSize: 860x600 - BackgroundImage: _684minhkhai3 - Text: "Quản lý đặt phòng"
2	lblTitle	Label	Tiêu đề "QUẢN LÝ"	- Font: Segoe UI, 20.25pt, Bold - Location: (286, 9)

			ĐẶT PHÒNG"	- BackColor: RGB(64, 64, 64) - ForeColor: RGB(224, 224, 224)
3	lblMaDP	Label	Nhãn "Mã đặt phòng:"	Font: Microsoft Sans Serif, 12pt - Location: (46, 78)
4	txtMaDatPhong	TextBox	Nhập/hiển thị mã đặt phòng	- Size: 200x20 - Location: (180, 77)
5	lblMaKH	Label	Nhãn "Khách hàng:"	Font: Microsoft Sans Serif, 12pt - Location: (46, 113)
6	cboMaKH	ComboBox	Chọn khách hàng	- Size: 200x21 - Location: (180, 112) - Load từ bảng KhachHang
7	lblMaNV	Label	Nhãn "Nhân viên:"	- Font: Microsoft Sans Serif, 12pt - Location: (46, 148)
8	cboMaNV	ComboBox	Chọn nhân viên	- Size: 200x21 - Location: (180, 147) - Load từ bảng NhanVien
9	lblMaPhong	Label	Nhãn "Phòng:"	- Font: Microsoft Sans Serif, 12pt - Location: (46, 182)
10	cboMaPhong	ComboBox	Chọn phòng	- Size: 200x21 - Location: (180, 182) - Load từ bảng Phong
11	lblDonGia	Label	Nhãn "Đơn giá:"	- Font: Microsoft Sans Serif, 12pt - Location: (46, 215)

12	txtDonGia	TextBox	Hiển thị đơn giá	<ul style="list-style-type: none"> - Size: 200x20 - Location: (180, 217) - ReadOnly: true - Auto fill khi chọn phòng
13	lblNgayDat	Label	Nhãn "Ngày đặt:"	<ul style="list-style-type: none"> - Font: Microsoft Sans Serif, 12pt - Location: (424, 78)
14	dtpNgayDat	DateTimePicker	Chọn ngày đặt	<ul style="list-style-type: none"> - Size: 200x20 - Location: (520, 77)
15	lblNgayDen	Label	Nhãn "Ngày đến:"	<ul style="list-style-type: none"> - Font: Microsoft Sans Serif, 12pt - Location: (424, 113)
16	dtpNgayDen	DateTimePicker	Chọn ngày đến	<ul style="list-style-type: none"> Size: 200x20 - Location: (520, 112) - Event: ValueChanged
17	lblNgayDi	Label	Nhãn "Ngày đi:"	<ul style="list-style-type: none"> - Font: Microsoft Sans Serif, 12pt - Location: (424, 148)
18	dtpNgayDi	DateTimePicker	Chọn ngày đi	<ul style="list-style-type: none"> - Size: 200x20 - Location: (520, 147) - Event: ValueChanged
19	lblTongTien	Label	Hiển thị tổng tiền	<ul style="list-style-type: none"> Font: Segoe UI, 20.25pt, Bold - Location: (484, 182) - ForeColor: DarkBlue - Text: "Tổng tiền: 0 VND" - Auto tính khi chọn ngày
20	dgvDatPhong	DataGridView	Hiển thị danh sách đặt phòng	<ul style="list-style-type: none"> - Size: 89x44 - Location: (50, 258) - Text: "Đặt phòng"

21	btnThem	Button	Nút đặt phòng	- Size: 91x44 - Location: (145, 258) - Text: "Trả phòng"
22	btnXoa	Button	Nút trả phòng	- Size: 98x44 - Location: (242, 258) - Text: "Làm mới"
23	btnLamMoi	Button	Nút làm mới	- Size: 95x44 - Location: (346, 258) - Text: "Hóa đơn"
24	btnLapHoaDon	Button	Nút lập hóa đơn	- Size: 184x20 - Location: (626, 271)
25	txtTimKiem	TextBox	Nhập từ khóa tìm kiếm	- Size: 100x44 - Location: (520, 258) - Text: "Tìm kiếm"
26	btnTimKiem	Button	Nút tìm kiếm	- Size: 760x250 - Location: (50, 320) - AutoSizeColumnsMode: Fill

Bảng 2.8: FrmHoaDon (Form quản lý hóa đơn)

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
1	FrmHoaDon	Form	Form quản lý hóa đơn	- ClientSize: 1000x600 - MinimumSize: 1000x600 - StartPosition: CenterScreen - BackColor: White

2	lblMaHD	Label	Nhãn "Mã hóa đơn"	Dock: Top - Size: 1000x70 - BackColor: RGB(52, 73, 94)
3	txtMaHD	TextBox	Nhập/hiển thị mã hóa đơn	- Dock: Fill - Font: Segoe UI, 20pt, Bold - ForeColor: White - TextAlign: MiddleCenter
4	lblMaDatPhong	Label	Nhãn "Mã đặt phòng"	- Dock: Top - Size: 1000x200 - BackColor: RGB(236, 240, 245) - Padding: (20, 15, 20, 10)
5	txtMaDatPhong	TextBox	Nhập/hiển thị mã đặt phòng	- Size: 520x175 - Location: (20, 15) - Font: Segoe UI, 10pt, Bold - Text: "Thông tin hóa đơn" - BackColor: White
6	lblNgayLap	Label	Nhãn "Ngày lập"	- Font: Segoe UI, 9.5pt - Location: (20, 35) - ForeColor: RGB(64, 64, 64)
7	dtpNgayLap	DateTimePicker	Chọn ngày lập hóa đơn	- Size: 330x25 - Location: (160, 32) - Font: Segoe UI, 9.5pt - BackColor: RGB(250, 250, 250)
8	lblTongTien	Label	Nhãn "Tổng tiền"	- Font: Segoe UI, 9.5pt - Location: (20, 70)

				- ForeColor: RGB(64, 64, 64)
9	txtTongTien	TextBox	Nhập/hiển thị tổng tiền	- Size: 330x25 - Location: (160, 67) - Font: Segoe UI, 9.5pt - BackColor: RGB(250, 250, 250)
10	dgvHoaDon	DataGridView	Hiển thị danh sách hóa đơn	- Font: Segoe UI, 9.5pt - Location: (20, 105) - ForeColor: RGB(64, 64, 64)
11	btnSua	Button	Nút sửa hóa đơn	- Size: 150x38 - Location: (25, 35) - BackColor: RGB(52, 152, 219) - ForeColor: White - Text: "■ Sửa" - FlatStyle: Flat - Cursor: Hand
12	btnXoa	Button	Nút xóa hóa đơn	- Size: 150x38 - Location: (25, 83) - BackColor: RGB(231, 76, 60) - ForeColor: White - Text: "🗑 Xóa" - FlatStyle: Flat - Cursor: Hand
13	txtTimKiem	TextBox	Nhập từ khóa tìm kiếm	- Size: 160x80 - Location: (20, 40) - Font: Segoe UI, 9.5pt

				<ul style="list-style-type: none"> - Multiline: true - BackColor: RGB(250, 250, 250) - BorderStyle: FixedSingle
14	btnTimKiem	Button	Nút tìm kiếm	<ul style="list-style-type: none"> - Size: 160x38 - Location: (20, 131) - BackColor: RGB(155, 89, 182) - ForeColor: White - Text: "🔍 Tìm kiếm" - FlatStyle: Flat - Cursor: Hand

Bảng 2.9: FrmBaoCaoDoanhThu (Form báo cáo doanh thu)

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
1	FrmBaoCaoDoanhThu	Form	Form báo cáo doanh thu theo tháng/năm	ClientSize: 900x580 - MinimumSize: 900x580 - StartPosition: CenterScreen - BackColor: White
2	label1	Label	Nhãn "Tháng:"	- Font: Segoe UI, 9.5pt - Location: (25, 35) - ForeColor: RGB(64, 64, 64)
3	cboThang	ComboBox	Chọn tháng thống kê	Size: 100x29 - Location: (85, 32) - Font: Segoe UI, 9.5pt - DropDownStyle:

				DropDownList - FlatStyle: Flat - BackColor: RGB(250, 250, 250) - Items: 1-12
4	label2	Label	Nhãn "Năm:"	- Font: Segoe UI, 9.5pt - Location: (210, 35) - ForeColor: RGB(64, 64, 64)
5	cboNam	ComboBox	Chọn năm thống kê	- Size: 120x29 - Location: (265, 32) - Font: Segoe UI, 9.5pt - DropDownStyle: DropDownList - FlatStyle: Flat - BackColor: RGB(250, 250, 250)
6	btnThongKe	Button	Nút xem thống kê	- Size: 140x38 - Location: (410, 28) - BackColor: RGB(46, 204, 113) - ForeColor: White - Text: "Thống kê" - Font: Segoe UI, 10pt, Bold - FlatStyle: Flat - Cursor: Hand
7	dgvDoanhThu	DataGridView	Hiển thị danh sách doanh thu	Dock: Fill - Size: 860x310 - BackgroundColor: White - BorderStyle: None - ColumnHeadersHeight: 40

				<ul style="list-style-type: none"> - RowTemplate.Height: 35 - SelectionMode: FullRowSelect - ReadOnly: true - AlternatingRowsBackColor: RGB(248, 249, 250) - Columns: MaHD, MaDatPhong, NgayLap, TongTien
8	label3	Label	Nhãn "Tổng doanh thu:"	<ul style="list-style-type: none"> - Font: Segoe UI, 14pt, Bold - Size: 230x40 - Location: (20, 12) - ForeColor: White - TextAlign: MiddleLeft
9	txtTongDoanhThu	TextBox	Hiển thị tổng doanh thu	<ul style="list-style-type: none"> - Size: 350x39 - Location: (260, 12) - Font: Segoe UI, 14pt, Bold - ForeColor: RGB(231, 76, 60) - BackColor: White - Text: "0 VNĐ" - TextAlign: Right - ReadOnly: true - BorderStyle: FixedSingle

Bảng 2.10: Giao Diện FrmDoiMatKhau (Form đổi mật khẩu)

STT	Tên đối tượng	Kiểu	Ý nghĩa	Ghi chú
-----	---------------	------	---------	---------

1	FrmDoiMatKhau	Form	Form đổi mật khẩu	ClientSize: 380x280 - StartPosition: CenterScreen - DoubleBuffered: true - BackgroundImage: userpassword_deusuari_787 - BackgroundImageLayout: Zoom
2	lblTitle	Label	Tiêu đề "ĐỔI MẬT KHẨU"	- Font: Segoe UI, 20.25pt, Bold - Location: (76, 9) - BackColor: RGB(255, 128, 128) - ForeColor: Black - AutoSize: true
3	lblOldPass	Label	Nhãn "Mật khẩu cũ:"	- Font: Microsoft Sans Serif, 12pt - Location: (29, 79) - AutoSize: true
4	txtOldPass	TextBox	Nhập mật khẩu cũ	- Size: 180x20 - Location: (188, 79) - PasswordChar: '*' - Ẩn ký tự nhập vào
5	lblNewPass	Label	Nhãn "Mật khẩu mới:"	- Size: 180x20 - Location: (188, 120) - PasswordChar: '*' - Ẩn ký tự nhập vào

2.3 Mô hình lớp.

Mô tả:

Hệ thống gồm 9 lớp chính thể hiện các thành phần nghiệp vụ của phần mềm quản lý homestay. Các lớp được tổ chức theo mô hình hướng đối tượng, trong đó có các quan hệ kế thừa và liên kết giữa các lớp.

Danh sách lớp và chức năng chính

Bảng 2.2 : Bảng Mô Hình Lớp

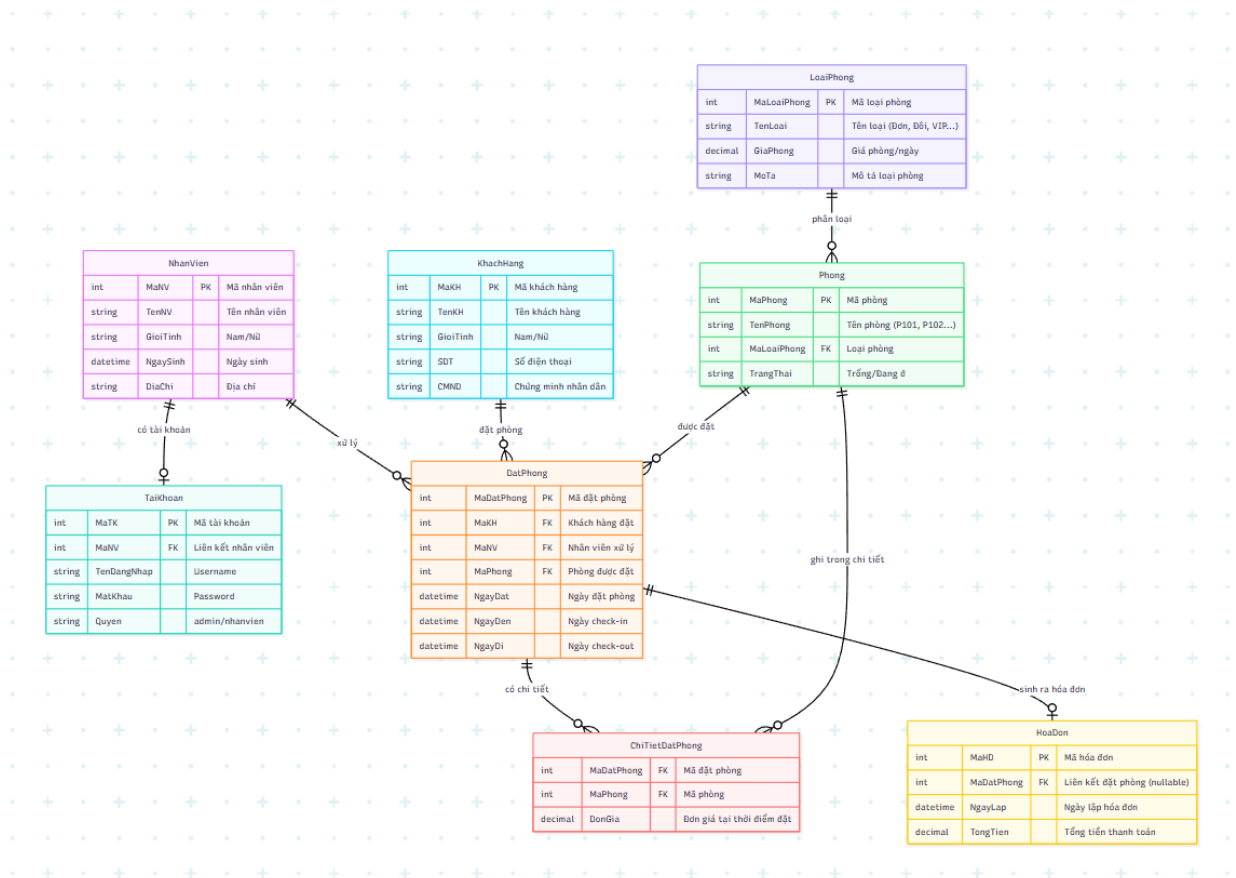
Tên lớp	Thuộc tính	Phương thức	Chức năng
KhachHang	MaKH, TenKH, GioiTinh, SDT, CMND	Them(), Sua(), Xoa(), TimKiem()	Quản lý thông tin khách hàng
NhanVien	MaNV, TenNV, ChucVu, SDT, TaiKhoan	Them(), Sua(), Xoa(), DangNhap()	Quản lý và xác thực nhân viên
Phong	MaPhong, TenPhong, MaLoai, TinhTrang	Them(), CapNhatTrangThai()	Quản lý trạng thái và thông tin phòng
LoaiPhong	MaLoai, TenLoai, GiaCoBan, MoTa	Them(), Sua(), Xoa()	Quản lý các loại phòng
DatPhong	MaDatPhong, MaKH, MaPhong, NgayDat, NgayTra	DatPhong(), HuyDatPhong()	Lưu thông tin đặt phòng

HoaDon	MaHD, MaDatPhong, NgayLap, TongTien	TinhTongTien(), XuatHoaDon()	Quản lý hóa đơn và thanh toán
TaiKhoan	TenDangNhap, MatKhau, VaiTro	DoiMatKhau(), KiemTraDangNhap()	Quản lý tài khoản đăng nhập
BaoCaoDoanhThu	Thang, Nam, TongDoanhThu	ThongKeTheoThang(), ThongKeTheoNam()	Thống kê doanh thu theo thời gian
MainForm (HeThong)	DanhSachForm, NguoiDungHienTai	MoForm(), DongForm()	Điều hướng các chức năng trong hệ thống

Quan hệ kế thừa và liên kết

- NhanVien **kế thừa** từ TaiKhoan (nhân viên có tài khoản đăng nhập).
- Phong **liên kết** với LoaiPhong (mỗi phòng thuộc một loại phòng).
- DatPhong **liên kết** với KhachHang và Phong.
- HoaDon **liên kết** với DatPhong.
- BaoCaoDoanhThu **tổng hợp dữ liệu** từ HoaDon.
- MainForm **liên kết** với tất cả các form chức năng khác.

2.4. Mô hình cơ sở dữ liệu.



Sơ đồ 1.1 : Sơ Đồ RM của DataBase

Khóa Chính (Primary Key - PK):

TaiKhoan : MaTK

NhanVien : MaNV

DatPhong : MaDatPhong

KhachHang : MaKH

Phong : MaPhong

LoaiPhong : MaLoaiPhong

ChiTietDatPhong : MaChiTiet

HoaDon : MaHD

Khóa Ngoại (Khóa Ngoại - FK):

-TaiKhoan : MaNV \rightarrow NhanVien(MaNV)

-DatPhong :

+MaKH \rightarrow KhachHang(MaKH)

+MaNV \rightarrow NhanVien(MaNV)

+MaPhong \rightarrow Phong(MaPhong)

ChiTietDatPhong :

+MaDatPhong \rightarrow DatPhong(MaDatPhong)

+ MaPhong \rightarrow Phong(MaPhong)

HoaDon : MaDatPhong \rightarrow DatPhong(MaDatPhong)

Phong : MaLoaiPhong \rightarrow LoaiPhong(MaLoaiPhong)

2.5 Các tiện ích.

Trong C#, **class method** = phương thức **static**.

Gọi là **class method** vì:

- Thuộc về **class**, không thuộc về **đối tượng (object)**.
- Không cần tạo instance để gọi.
- Gọi trực tiếp qua class:

```
// Thêm method để vẽ border cho control rounded
public static void DrawRoundedBorder(Control control, PaintEventArgs e, int radius, Color borderColor, int borderWidth = 1)
{
    if (control == null || e == null) return;

    using (GraphicsPath path = GetRoundedRectanglePath(
        new Rectangle(0, 0, control.Width - 1, control.Height - 1), radius))
    using (Pen pen = new Pen(borderColor, borderWidth))
    {
        e.Graphics.SmoothingMode = SmoothingMode.AntiAlias;
        e.Graphics.DrawPath(pen, path);
    }
}
```



```
// Set bo tròn cho Button
public static void SetRoundedButton(Button btn, int radius = 8)
{
    if (btn == null) return;

    btn.FlatStyle = FlatStyle.Flat;
    btn.FlatAppearance.BorderSize = 0;
    btn.Region?.Dispose();

    GraphicsPath path = GetRoundedRectanglePath(new Rectangle(0, 0, btn.Width, btn.Height), radius);
    btn.Region = new Region(path);
    path.Dispose();
    btn.Resize += (s, e) =>
    {
        btn.Region?.Dispose();
        GraphicsPath newPath = GetRoundedRectanglePath(new Rectangle(0, 0, btn.Width, btn.Height), radius);
        btn.Region = new Region(newPath);
        newPath.Dispose();
    };
}
```

```
// Set bo tròn cho TextBox
public static void SetRoundedTextBox(TextBox txt, int radius = 8, int height = 35)
{
    if (txt == null) return;

    txt.BorderStyle = BorderStyle.None;
    txt.Height = height;

    txt.Region?.Dispose();

    GraphicsPath path = GetRoundedRectanglePath(new Rectangle(0, 0, txt.Width, txt.Height), radius);
    txt.Region = new Region(path);
    path.Dispose();
    txt.Resize += (s, e) =>
    {
        txt.Region?.Dispose();
        GraphicsPath newPath = GetRoundedRectanglePath(new Rectangle(0, 0, txt.Width, txt.Height), radius);
        txt.Region = new Region(newPath);
        newPath.Dispose();
    };
}
```

```
// Tạo GraphicsPath cho hình chữ nhật bo tròn
private static GraphicsPath GetRoundedRectanglePath(Rectangle rect, int radius)
{
    GraphicsPath path = new GraphicsPath();
    int diameter = radius * 2;
    if (diameter > rect.Width) diameter = rect.Width;
    if (diameter > rect.Height) diameter = rect.Height;

    Rectangle arc = new Rectangle(rect.Location, new Size(diameter, diameter));

    // Top left arc
    path.AddArc(arc, 180, 90);

    // Top right arc
    arc.X = rect.Right - diameter;
    path.AddArc(arc, 270, 90);

    // Bottom right arc
    arc.Y = rect.Bottom - diameter;
    path.AddArc(arc, 0, 90);

    // Bottom left arc
    arc.X = rect.Left;
    path.AddArc(arc, 90, 90);

    path.CloseFigure();
    return path;
}
```

```
// hover cho Button
public static void SetButtonHoverEffect(Button btn, Color normalColor, float lightFactor = 0.2f)
{
    if (btn == null) return;

    Color hoverColor = ControlPaint.Light(normalColor, lightFactor);

    btn.MouseEnter += (s, e) =>
    {
        btn.BackColor = hoverColor;
        btn.Cursor = Cursors.Hand;
    };

    btn.MouseLeave += (s, e) =>
    {
        btn.BackColor = normalColor;
        btn.Cursor = Cursors.Default;
    };
}
```

Chương 3. Phát triển hệ thống.

3.1. Lớp GUI

GUI (Graphic User Interface) là lớp giao diện người dùng.

Trong hệ thống WinForms, lớp GUI gồm các **Form**, mỗi Form đảm nhiệm một chức năng riêng. Nó gồm các thành phần giao diện (win form, web form,...) và thực hiện các công việc như nhập liệu, hiển thị dữ liệu, kiểm tra tính đúng đắn dữ liệu trước khi gọi lớp BusinessLogic Layer (BLL).

- **Frm Login** : Hiển thị thông tin yêu cầu nhập tên đăng nhập , mật khẩu .Người dùng cần phải có tài khoản riêng được cấp phép truy cập sử dụng với mục đích bảo mật .



WELCOME !

Tên đăng nhập:

admin

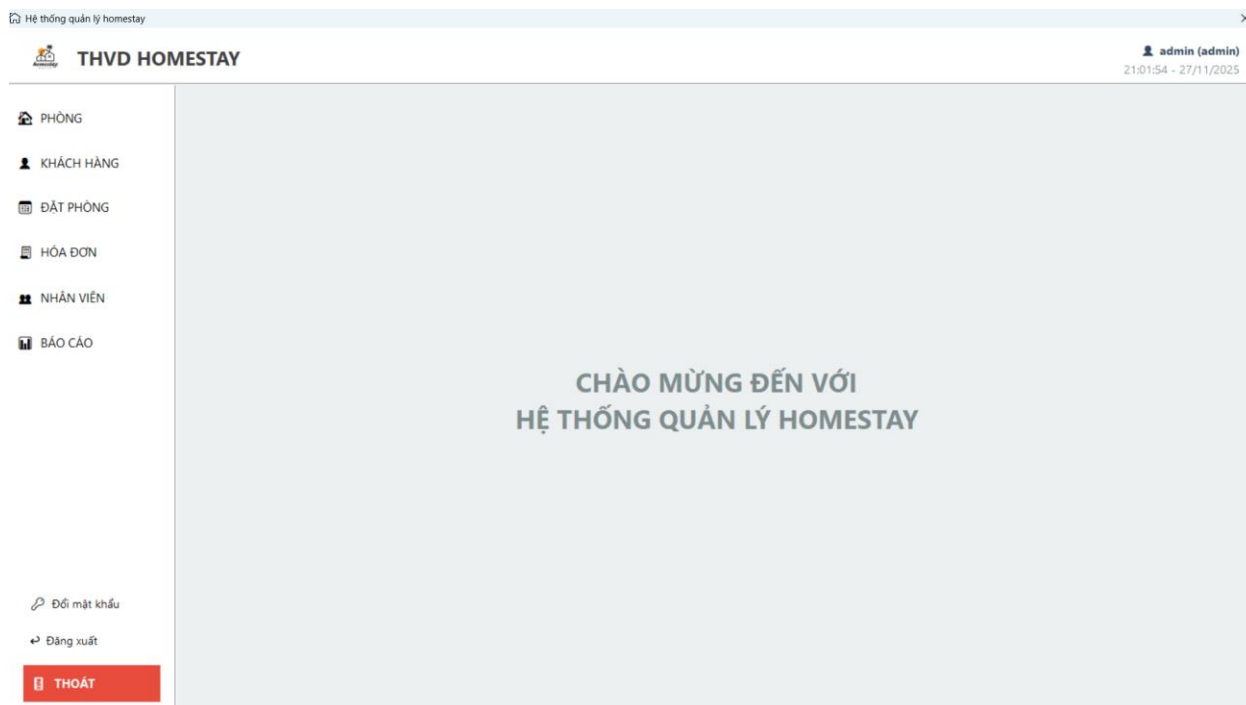
Mật khẩu:

•••

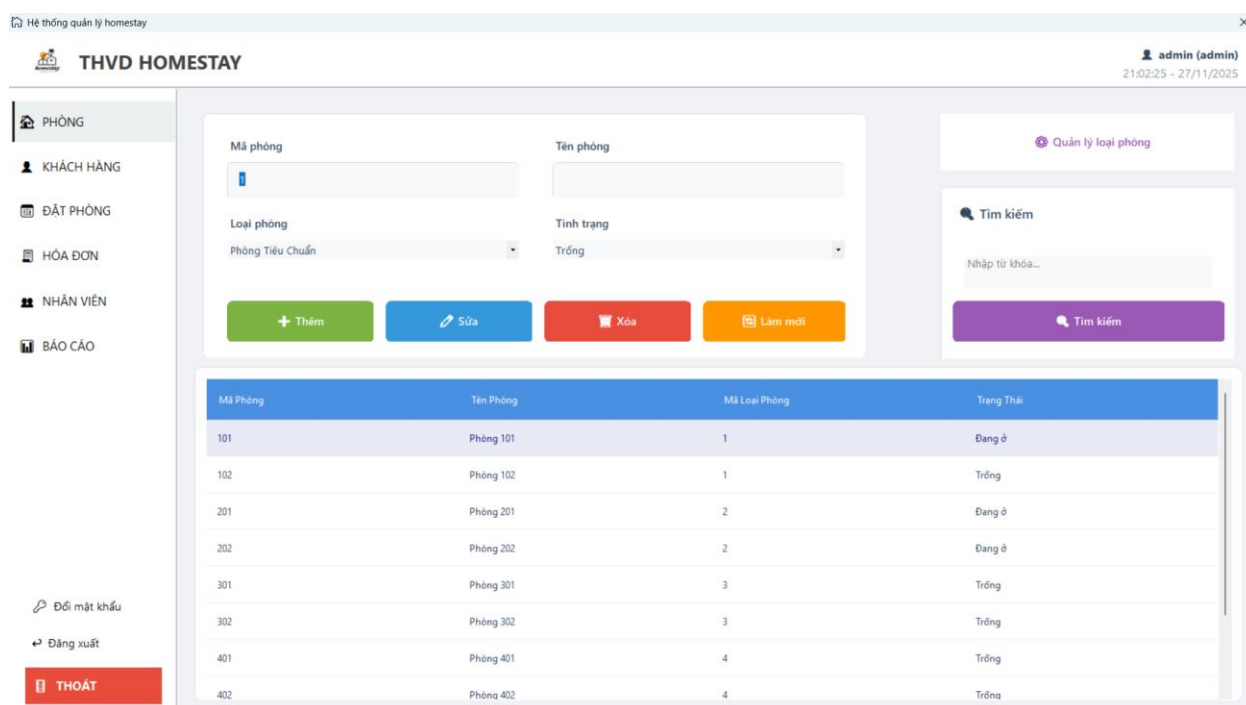
Đăng nhập

Thoát

- **Frm Main** : Nơi lựa chọn chức năng của quản lý Phòng , Khách Hàng , Hóa Đơn , Báo Cáo Doanh Thu , Đổi Mật Khẩu .



- **Frm Phòng** : Cung Cấp Mã Phòng , Tên Phòng , Loại Phòng , Tình Trạng Phòng và các chức năng Thêm , Sửa , Xóa , Làm Mới , Tìm kiếm Phòng .



- **Frm Loại Phòng** : Cung cấp Mã Loại Phòng , Tên Loại Phòng , Mô Tả , Giá Cơ Bản và các chức năng Thêm , Sửa , Xóa , Làm Mới .

THVD HOMESTAY

admin (admin)

17:08:43 - 28/11/2025

PHÒNG

KHÁCH HÀNG

ĐẶT PHÒNG

HÓA ĐƠN

NHÂN VIÊN

BÁO CÁO

Thông tin loại phòng

Mã loại phòng

Tên loại phòng

Giá cơ bản (VNĐ)

Mô tả

Thêm

Sửa

Xóa

Làm mới

Mã Loại	Tên Loại Phòng	Giá Phòng (VNĐ)	Mô Tả
1	Phòng Tiêu Chuẩn	500,000	Phòng đơn tiêu chuẩn
2	Phòng Cao Cấp	800,000	Phòng đôi cao cấp
3	Phòng Hạng Sang	1,200,000	Phòng hạng sang
4	Phòng Tổng Thống	2,000,000	Phòng sang trọng, có phòng khách riêng
5	Phòng Hoàng Gia	3,000,000	Phòng cao cấp nhất, view đẹp

Đổi mật khẩu

Đăng xuất

THOÁT

- **Frm Khách Hàng** : Cung cấp Mã Khách Hàng , Tên Khách Hàng , Số Điện Thoại , CMND/CCCD , Giới Tính và các chức năng Thêm , Sửa , Xóa , Làm Mới và Tìm

THVD HOMESTAY

admin (admin)

17:11:18 - 28/11/2025

PHÒNG

KHÁCH HÀNG

ĐẶT PHÒNG

HÓA ĐƠN

NHÂN VIÊN

BÁO CÁO

Thông tin khách hàng

Mã khách hàng:

Tên khách hàng:

Giới tính:

Số điện thoại:

CMND/CCCD:

Tìm mã:

Thêm

Sửa

Xóa

Làm mới


Tìm kiếm

Mã Khách Hàng	Tên Khách Hàng	Giới Tính	Số Điện Thoại	Chứng Minh Nhân Dân
1	Nguyễn Minh Anh	Nam	0901234567	001234567890
2	Trần Thị Bích	Nữ	09123456781	002345678902
3	Lê Văn Cường	Nam	0923456789	003456789012
4	Phạm Thị Dung	Nữ	0934567890	004567890123
5	Hoàng Văn Em	Nam	0945678901	005678901234
6	Vũ Thị Phương	Nữ	0956789012	006789012345
7	Đặng Văn Giang	Nam	0967890123	007890123456
8	Bùi Thị Hoa	Nữ	0978901234	008901234567

Đổi mật khẩu

Đăng xuất

THOÁT

- 

THVD HOMESTAY

admin (admin)

17:11:49 - 28/11/2025

PHÒNG

KHÁCH HÀNG

ĐẶT PHÒNG

HÓA ĐƠN

NHÂN VIÊN

BÁO CÁO

Thông tin đặt phòng

Mã đặt phòng

Khách hàng

Nguyễn Minh Anh

Nhân viên

Nguyễn Văn A

Phòng

Phòng 103

Ngày đặt

11/28/2025

Ngày nhận phòng

11/28/2025

Ngày trả phòng

11/29/2025

Số ngày

1

Đơn giá

500,000

Tổng tiền

500,000 VND

Đặt phòng

Trả phòng

Làm mới

Hóa đơn

Tìm kiếm

Mã đặt phòng	Khách hàng	Nhân viên	Phòng	Ngày đặt	Ngày đến	Ngày đi	Tổng tiền
1	Nguyễn Minh Anh	Nguyễn Văn A	Phòng 101	10/20/2025	10/25/2025	10/27/2025	1000000.00
2	Trần Thị Bích	Lê Thị B	Phòng 201	10/21/2025	10/26/2025	10/28/2025	1600000.00
3	Lê Văn Cường	Nguyễn Văn A	Phòng 301	10/22/2025	10/27/2025	10/30/2025	3600000.00
4	Phạm Thị Dung	Lê Thị B	Phòng 401	10/23/2025	10/28/2025	11/1/2025	8000000.00
5	Hoàng Văn Em	Nguyễn Văn A	Phòng 501	10/24/2025	10/29/2025	11/2/2025	12000000.00
6	Vũ Thị Phương	Trần Văn C	Phòng 102	11/1/2025	11/5/2025	11/7/2025	1000000.00
7	Đặng Văn Giang	Trần Văn C	Phòng 202	11/2/2025	11/6/2025	11/9/2025	2400000.00
8	Bùi Thị Hoa	Phạm Thị D	Phòng 302	11/3/2025	11/8/2025	11/11/2025	3600000.00
9	Mai Văn Hùng	Phạm Thị D	Phòng 402	11/4/2025	11/10/2025	11/13/2025	6000000.00

- 53

THVD HOMESTAY

admin (admin)

17:12:08 - 28/11/2025

PHÒNG

KHÁCH HÀNG

ĐẶT PHÒNG

HÓA ĐƠN

NHÂN VIÊN

BẢO CÁO

Thông tin hóa đơn

Mã hóa đơn:

Mã đặt phòng:

Ngày lập: 11/28/2025

Tổng tiền:

Thao tác

Sửa

Xóa

Làm mới

Tìm kiếm

Tìm kiếm

Mã hóa đơn	Mã đặt phòng	Ngày lập	Tổng tiền
1	1	10/27/2025	1000000.00
2	2	10/28/2025	1600000.00
3	3	10/30/2025	3600000.00
4	4	11/1/2025	8000000.00
5	5	11/2/2025	12000000.00
6	6	11/7/2025	1000000.00
7	7	11/9/2025	2400000.00
8	8	11/11/2025	3600000.00
9	9	11/13/2025	6000000.00
10	10	11/15/2025	9000000.00

Đổi mật khẩu

Đăng xuất

THOÁT

Frm Nhân Viên : Cung cấp Tên Nhân Viên , Địa Chỉ , Giới Tính và các chức năng Thêm , Sửa , Xóa , Làm Mới .

THVD HOMESTAY

admin (admin)

17:12:41 - 28/11/2025

PHÒNG

KHÁCH HÀNG

ĐẶT PHÒNG

HÓA ĐƠN

NHÂN VIÊN

BẢO CÁO

Thông tin nhân viên

Tên nhân viên

Giới tính

Nam

Ngày sinh

11/28/2025

Địa chỉ

Tìm kiếm (ID)

Tìm kiếm

Thêm

Sửa

Xóa

Làm mới

Mã nhân viên	Tên nhân viên	Giới tính	Ngày sinh	Địa chỉ
1	Nguyễn Văn A	Nam	5/10/1995	Hà Nội
2	Lê Thị B	Nữ	8/20/1998	Hồ Chí Minh
3	Trần Văn C	Nam	3/15/1997	Đà Nẵng
4	Phạm Thị D	Nữ	11/25/1999	Hải Phòng
5	Hoàng Văn E	Nam	7/8/1996	Cần Thơ
6	Vũ Thị F	Nữ	2/14/2000	Hà Nội
7	Đặng Văn G	Nam	9/30/1998	Hồ Chí Minh
8	Tùng	Nam	11/1/2025	HaNoi

Đổi mật khẩu

Đăng xuất

THOÁT

- **Frm Báo Cáo Doanh Thu** : Cung Cấp Bộ lọc thời gian để hiện thu nhập theo tháng năm và cung cấp Mã Hóa Đơn , Mã Đặt Phòng , Ngày Lập , Tổng Tiền và chức năng thống kê và Xuất Excel , hiện tổng doanh thu .

THVD HOMESTAY

admin (admin)
 17:13:00 - 28/11/2025

PHÒNG
 KHÁCH HÀNG
 ĐẶT PHÒNG
 HÓA ĐƠN
 NHÂN VIÊN
BÁO CÁO

Bộ lọc thời gian
 Tháng: 11 Năm: 2025

Thống kê
 Xuất Excel

Mã hóa đơn	Mã đặt phòng	Ngày lập	Tổng tiền
4	4	11/1/2025	8,000,000
5	5	11/2/2025	12,000,000
6	6	11/7/2025	1,000,000
7	7	11/9/2025	2,400,000
8	8	11/11/2025	3,600,000
9	9	11/13/2025	6,000,000
10	10	11/15/2025	9,000,000

Đổi mật khẩu
 Đăng xuất

THOÁT
 Tổng doanh thu: 42,000,000 VNĐ

- **Frm Đổi Mật Khẩu** : Điền Thông Tin Mật Khẩu Cũ , Mật Khẩu mới , Xác Nhận Mật Khẩu để đổi mật khẩu và các chức năng Lưu , Hủy .



ĐỔI MẬT KHẨU

Mật khẩu cũ:

Mật khẩu mới:

Xác nhận mật khẩu:

LƯU

HỦY

3.2 Lớp BUS

Các class nghiệp vụ có trong hệ thống, như xử lý, báo cáo, thống kê. Chứa các nghiệp vụ, kiểm tra dữ liệu, xử lý logic, gọi BLL or gọi là lớp BUS

Business Logic Layer (BLL)

Layer này phân ra 2 thành nhiệm vụ:

- Đây là nơi đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, xử lý chỉnh nguồn dữ liệu từ Presentation layer trước khi truyền xuống Data Layer và lưu xuống hệ quản trị CSDL.

Đây còn là nơi kiểm tra các ràng buộc, tính toàn vẹn của dữ liệu, thực hiện tính toán và xử lý yêu cầu nghiệp vụ, trước khi trả kết quả về Presentation Layer.

BaoCaoDoanhThuBLL là lớp xử lý nghiệp vụ liên quan đến báo cáo doanh thu. Lớp thực hiện kiểm tra dữ liệu đầu vào, gọi tầng DAL để lấy dữ liệu, tính toán và định dạng doanh thu.

Chức năng chính

- **LayBaoCaoDoanhThu(month, year):** Lấy danh sách báo cáo doanh thu theo tháng/năm (kèm kiểm tra đầu vào).
- **TinhTongDoanhThu(month, year):** Tính tổng doanh thu của tháng.
- **FormatTien(tien):** Định dạng tiền theo dạng “1,000 VNĐ”.
- **LayBaoCaoTheoKhoangThoiGian(tuNgay, denNgay):** Lấy báo cáo theo khoảng ngày (có kiểm tra hợp lệ ngày).
- **LayDoanhThuTheoThang(year):** Lấy doanh thu của từng tháng trong năm.
- **KiemTraCoDuLieu(month, year):** Kiểm tra tháng đó có dữ liệu báo cáo hay không.
- **TinhPhanTramTangTruong(month, year):** Tính % tăng trưởng so với tháng trước.

DatPhongBLL là lớp xử lý nghiệp vụ cho quy trình đặt phòng trong hệ thống quản lý homestay. Lớp thực hiện kiểm tra dữ liệu, tính toán số ngày – tổng tiền, và điều phối các thao tác giữa giao diện và DAL.

Chức năng chính

- **LayDanhSachDatPhong()**
Lấy toàn bộ danh sách lượt đặt phòng.
- **LayDanhSachKhachHang() / LayDanhSachNhanVien() / LayDanhSachPhongTrong()**
Lấy danh sách khách hàng, nhân viên và các phòng đang trống.
- **LayDonGiaPhong(maPhong)**
Lấy đơn giá của phòng.

- **LayMaDatPhongTiepTheo()**
Sinh mã đặt phòng mới.
- **TinhTongTien(donGia, ngayDen, ngayDi)**
Tính tổng tiền dựa trên số ngày ở (tối thiểu 1 ngày).
- **TinhSoNgay(ngayDen, ngayDi)**
Tính số ngày khách lưu trú.
- **ValidateDatPhong(...)**
Kiểm tra dữ liệu đầu vào: khách hàng, nhân viên, phòng, ngày đến – ngày đi.
- **ThemDatPhong(dto, out error)**
Thêm một lượt đặt phòng mới gồm:
 - Kiểm tra dữ liệu
 - Kiểm tra mã đặt phòng
 - Kiểm tra trạng thái phòng
 - Lưu đặt phòng + chi tiết đặt phòng
 - Cập nhật trạng thái phòng sang **Đang ở**
 - Tạo hóa đơn
- **XoaDatPhong(maDatPhong, out error)**
Xóa thông tin một lượt đặt phòng.
- **TimKiemDatPhong(keyword)**
Tìm kiếm bằng mã hoặc theo tên/keyword.

HoaDonBLL là lớp xử lý nghiệp vụ cho hóa đơn, thực hiện các thao tác lấy dữ liệu, cập nhật, xóa và kiểm tra dữ liệu đầu vào trước khi gửi xuống DAL.

Chức năng chính

- **LayDanhSachHoaDon()**
Lấy toàn bộ danh sách hóa đơn.
- **LayHoaDonTheoMa(maHD)**
Lấy chi tiết hóa đơn theo mã.

- **CapNhatHoaDon(maHD, maDatPhong, ngayLap, tongTien)**
Cập nhật thông tin hóa đơn (ngày lập, tổng tiền, mã đặt phòng...).
 - **XoaHoaDon(maHD)**
Xóa hóa đơn theo mã.
 - **TimKiemHoaDon(keyword)**
Tìm kiếm hóa đơn theo mã, ngày lập hoặc từ khóa.
 - **KiemTraMaDatPhong(text, out maDatPhong)**
Kiểm tra chuỗi nhập có phải số hợp lệ để dùng làm mã đặt phòng.
 - **KiemTraTongTien(text, out tongTien)**
Chuyển chuỗi sang decimal theo chuẩn VN hoặc Invariant, dùng để validate số tiền.
-

Lớp KháchHangBLL là lớp xử lý nghiệp vụ cho khách hàng, thực hiện các thao tác lấy, thêm, cập nhật, xóa, và tìm kiếm dữ liệu khách hàng. Nó tương tác với lớp DAL để thao tác dữ liệu và kiểm tra tính hợp lệ của đầu vào.

Các phương thức chính:

1. **LayDanhSachKhachHang()**: Lấy toàn bộ danh sách khách hàng.
2. **LayKhachHangTheoMa(int maKH)**: Lấy thông tin khách hàng theo mã.
3. **ThemKhachHang(int maKH, string tenKH, string gioiTinh, string sdt, string cmnd)**: Thêm khách hàng mới (kiểm tra mã khách hàng và tên hợp lệ).
4. **CapNhatKhachHang(int maKH, string tenKH, string gioiTinh, string sdt, string cmnd)**: Cập nhật thông tin khách hàng.
5. **XoaKhachHang(int maKH)**: Xóa khách hàng nếu không đang được tham chiếu.
6. **TimKiemKhachHangTheoMa(int maKH)**: Tìm kiếm khách hàng theo mã.
7. **LayMaKHTiepTheo()**: Lấy mã khách hàng tiếp theo chưa sử dụng.

Lớp này giúp kiểm tra dữ liệu hợp lệ trước khi gửi xuống DAL để thao tác với cơ sở dữ liệu.

LoaiPhongBLL là lớp xử lý nghiệp vụ cho loại phòng, thực hiện các thao tác thêm, cập nhật, xóa và lấy thông tin loại phòng, trước khi gửi xuống DAL.

Chức năng chính

- **LayDanhSachLoaiPhong()**
Lấy toàn bộ danh sách loại phòng.
 - **ThemLoaiPhong(tenLoai, giaPhong, moTa)**
Thêm loại phòng mới.
 - **CapNhatLoaiPhong(maLoai, tenLoai, giaPhong, moTa)**
Cập nhật thông tin loại phòng.
 - **XoaLoaiPhong(maLoai)**
Xóa loại phòng theo mã.
 - **TimGiaTheoTen(tenLoai)**
Tìm và trả về giá theo tên loại phòng.
-

LoginBLL là lớp xử lý nghiệp vụ đăng nhập, thực hiện kiểm tra tài khoản người dùng bằng cách gọi xuống lớp DAL.

Chức năng chính

- **GetTaiKhoan(username, password)**
Lấy thông tin tài khoản dựa trên tên đăng nhập và mật khẩu để xác thực đăng nhập.
-

NhanVienBLL là lớp xử lý nghiệp vụ nhân viên, thực hiện các thao tác thêm, cập nhật, xóa và lấy danh sách nhân viên, trước khi gửi yêu cầu xuống DAL.

Chức năng chính

- **LayDanhSachNhanVien()**
Lấy toàn bộ danh sách nhân viên.
 - **ThemNhanVien(nv)**
Thêm nhân viên mới.
 - **CapNhatNhanVien(nv)**
Cập nhật thông tin nhân viên.
 - **XoaNhanVien(maNv)**
Xóa nhân viên theo mã.
-

PhongBLL là lớp xử lý nghiệp vụ cho phòng, thực hiện các thao tác thêm, sửa, xóa, lấy danh sách và tìm kiếm phòng trước khi gửi dữ liệu xuống DAL.

Chức năng chính

- **LayDanhSachPhong()**
Lấy toàn bộ danh sách phòng.
- **LayPhongTheoMa(maPhong)**
Lấy thông tin chi tiết của phòng theo mã.
- **ThemPhong(maPhong, tenPhong, maLoaiPhong, trangThai)**
Thêm phòng mới.
- **CapNhatPhong(maPhong, tenPhong, maLoaiPhong, trangThai)**
Cập nhật thông tin phòng.
- **XoaPhong(maPhong)**
Xóa phòng theo mã.
- **TimKiemPhong(keyword)**
Tìm kiếm phòng theo tên, mã hoặc trạng thái.
- **KiemTraMaPhongTonTai(maPhong)**
Kiểm tra mã phòng đã tồn tại hay chưa.
- **LayMaPhongTiepTheo()**
Tìm mã phòng tiếp theo chưa được sử dụng.

TaiKhoanBLL là lớp xử lý nghiệp vụ tài khoản, chủ yếu phục vụ thay đổi mật khẩu và kiểm tra tính hợp lệ trước khi gửi xuống DAL.

Chức năng chính

- **DoiMatKhau(maNv, matKhauCu, matKhauMoi)**
Kiểm tra mật khẩu cũ, xác thực tài khoản theo mã nhân viên và cập nhật mật khẩu mới nếu hợp lệ.

3.3. Lớp DAL.

Các lớp thực thể có trong hệ thống.. Chứa các code làm việc với SQL Server , truy vấn Database .

Data Access Layer (DAL)

Lớp này có chức năng giao tiếp với hệ quản trị CSDL như thực hiện các công việc liên quan đến lưu trữ và truy vấn dữ liệu (tìm kiếm, thêm, xóa, sửa,...).

BaoCaoDoanhThuDA : Lớp này dùng để lấy dữ liệu báo cáo doanh thu từ bảng **HoaDon**, bao gồm các phương thức:

```
// Lấy báo cáo doanh thu theo tháng và năm
2 references
public List<BaoCaoDoanhThuDTO> LayBaoCaoDoanhThu(int month, int year)
{
    try
    {
        DateTime startDate = new DateTime(year, month, 1);
        DateTime endDate = startDate.AddMonths(1).AddDays(-1);

        string query = @"SELECT MaHD, MaDatPhong, NgayLap, TongTien
                        FROM HoaDon
                        WHERE CAST(NgayLap AS DATE) >= @startDate
                        AND CAST(NgayLap AS DATE) <= @endDate
                        ORDER BY NgayLap";

        DataTable data = DataProvider.Instance.ExecuteQuery(query,
            new object[] { startDate, endDate });

        List<BaoCaoDoanhThuDTO> danhSach = new List<BaoCaoDoanhThuDTO>();

        foreach (DataRow row in data.Rows)
        {
            BaoCaoDoanhThuDTO dto = new BaoCaoDoanhThuDTO(
                Convert.ToInt32(row["MaHD"]),
                row["MaDatPhong"] != DBNull.Value ? (int?)Convert.ToInt32(row["MaDatPhong"]) : null,
                row["NgayLap"] != DBNull.Value ? (DateTime?)Convert.ToDateTime(row["NgayLap"]) : null,
                row["TongTien"] != DBNull.Value ? (decimal?)Convert.ToDecimal(row["TongTien"]) : null
            );
            danhSach.Add(dto);
        }

        return danhSach;
    }
    catch (Exception ex)
    {
        throw new Exception("Lỗi khi lấy báo cáo doanh thu: " + ex.Message);
    }
}
```

// Tính tổng doanh thu theo tháng và năm

3 references

```
public decimal TinhTongDoanhThu(int month, int year)
{
    try
    {
        DateTime startDate = new DateTime(year, month, 1);
        DateTime endDate = startDate.AddMonths(1).AddDays(-1);

        string query = @"SELECT ISNULL(SUM(TongTien), 0)
                        FROM HoaDon
                        WHERE CAST(NgayLap AS DATE) >= @startDate
                        AND CAST(NgayLap AS DATE) <= @endDate ";

        object result = DataProvider.Instance.ExecuteScalar(query,
            new object[] { startDate, endDate });

        return result != null ? Convert.ToDecimal(result) : 0m;
    }
    catch (Exception ex)
    {
        throw new Exception("Lỗi khi tính tổng doanh thu: " + ex.Message);
    }
}
```


// Lấy báo cáo theo khoảng thời gian

1 reference

```
public List<BaoCaoDoanhThuDTO> LayBaoCaoTheoKhoangThoiGian(DateTime tuNgay, DateTime denNgay)
{
    try
    {
        string query = @"SELECT MaHD, MaDatPhong, NgayLap, TongTien
                        FROM HoaDon
                        WHERE CAST(NgayLap AS DATE) >= @tuNgay
                        AND CAST(NgayLap AS DATE) <= @denNgay
                        ORDER BY NgayLap";

        DataTable data = DataProvider.Instance.ExecuteQuery(query,
            new object[] { tuNgay, denNgay });

        List<BaoCaoDoanhThuDTO> danhSach = new List<BaoCaoDoanhThuDTO>();

        foreach (DataRow row in data.Rows)
        {
            BaoCaoDoanhThuDTO dto = new BaoCaoDoanhThuDTO(
                Convert.ToInt32(row["MaHD"]),
                row["MaDatPhong"] != DBNull.Value ? (int?)Convert.ToInt32(row["MaDatPhong"]) : null,
                row["NgayLap"] != DBNull.Value ? (DateTime?)Convert.ToDateTime(row["NgayLap"]) : null,
                row["TongTien"] != DBNull.Value ? (decimal?)Convert.ToDecimal(row["TongTien"]) : null
            );
            danhSach.Add(dto);
        }

        return danhSach;
    }
    catch (Exception ex)
    {
        throw new Exception("Lỗi khi lấy báo cáo theo khoảng thời gian: " + ex.Message);
    }
}
```

// Lấy doanh thu theo năm (theo tháng)

1 reference

```
public DataTable LayDoanhThuTheoThang(int year)
{
    try
    {
        string query = @"SELECT MONTH(NgayLap) AS Thang,
                        ISNULL(SUM(TongTien), 0) AS DoanhThu
                        FROM HoaDon
                        WHERE YEAR(NgayLap) = @year
                        GROUP BY MONTH(NgayLap)
                        ORDER BY MONTH(NgayLap)";

        return DataProvider.Instance.ExecuteQuery(query, new object[] { year });
    }
    catch (Exception ex)
    {
        throw new Exception("Lỗi khi lấy doanh thu theo tháng: " + ex.Message);
    }
}
```

Lớp **DataProvider** trong đoạn code bạn gửi là lớp trung gian (**Data Access Layer – DAL**), chức năng chính là **giao tiếp với cơ sở dữ liệu SQL Server**. Mình tóm tắt chi tiết các chức năng chính của lớp này :

a) ExecuteQuery

- Thực thi câu lệnh **SELECT**.
- Trả về **DataTable**.
- Hỗ trợ **tham số hóa** để tránh SQL Injection.

b) ExecuteScalar

- Thực thi câu lệnh trả về **một giá trị duy nhất**, ví dụ COUNT, SUM.

c) ExecuteNonQuery

- Thực thi các câu lệnh **INSERT, UPDATE, DELETE**.
- Trả về số dòng bị ảnh hưởng.

Lớp **DatPhongDAL** đóng vai trò **trung gian giữa ứng dụng và cơ sở dữ liệu**, đảm nhiệm các chức năng:

- Truy xuất dữ liệu (SELECT)
- Thêm dữ liệu (INSERT)
- Cập nhật dữ liệu (UPDATE)
- Xóa dữ liệu (DELETE)
- Kiểm tra trạng thái hoặc tồn tại của dữ liệu
- Hỗ trợ tìm kiếm và quản lý mã tự động

```
// Tìm kiếm theo mã đặt phòng
1 reference
public List<DatPhongDTO> TimKiemTheoMa(int maDatPhong)
{
    string query = @"SELECT dp.MaDatPhong, kh.TenKH, nv.TenNV, p.TenPhong,
dp.NgayDat, dp.NgayDen, dp.NgayDi,
ISNULL(hd.TongTien, 0) as TongTien
FROM DatPhong dp
JOIN KhachHang kh ON dp.MaKH = kh.MaKH
JOIN NhanVien nv ON dp.MaNV = nv.MaNV
JOIN Phong p ON dp.MaPhong = p.MaPhong
LEFT JOIN HoaDon hd ON dp.MaDatPhong = hd.MaDatPhong
WHERE dp.MaDatPhong = @maDatPhong
ORDER BY dp.NgayDat DESC";

    DataTable data = DataProvider.Instance.ExecuteNonQuery(query, new object[] { maDatPhong });
    List<DatPhongDTO> danhSach = new List<DatPhongDTO>();

    foreach (DataRow row in data.Rows)
    {
        DatPhongDTO dto = new DatPhongDTO(
            Convert.ToInt32(row["MaDatPhong"]),
            row["TenKH"].ToString(),
            row["TenNV"].ToString(),
            row["TenPhong"].ToString(),
            Convert.ToDateTime(row["NgayDat"]),
            Convert.ToDateTime(row["NgayDen"]),
            Convert.ToDateTime(row["NgayDi"]),
            Convert.ToDecimal(row["TongTien"])
        );
        danhSach.Add(dto);
    }
    return danhSach;
}
```

```

// Xóa đặt phòng (trả phòng)
1 reference
public bool XoaDatPhong(int maDatPhong, out string error)
{
    error = string.Empty;
    try
    {
        // Cập nhật hóa đơn trước
        string query1 = "UPDATE HoaDon SET MaDatPhong = NULL WHERE MaDatPhong = @maDatPhong ";
        DataProvider.Instance.ExecuteNonQuery(query1, new object[] { maDatPhong });

        // Lấy danh sách phòng từ chi tiết để cập nhật trạng thái
        string query2 = "SELECT MaPhong FROM ChiTietDatPhong WHERE MaDatPhong = @maDatPhong ";
        DataTable dtPhong = DataProvider.Instance.ExecuteQuery(query2, new object[] { maDatPhong });

        foreach (DataRow row in dtPhong.Rows)
        {
            int maPhong = Convert.ToInt32(row["MaPhong"]);
            CapNhatTrangThaiPhong(maPhong, "Trống", out _);
        }

        // Xóa chi tiết đặt phòng
        string query3 = "DELETE FROM ChiTietDatPhong WHERE MaDatPhong = @maDatPhong ";
        DataProvider.Instance.ExecuteNonQuery(query3, new object[] { maDatPhong });

        // Xóa đặt phòng
        string query4 = "DELETE FROM DatPhong WHERE MaDatPhong = @maDatPhong ";
        int result = DataProvider.Instance.ExecuteNonQuery(query4, new object[] { maDatPhong });

        return result > 0;
    }
    catch (Exception ex)
    {
        error = ex.Message;
        return false;
    }
}

```

Lớp **HoaDonDAL** đóng vai trò trung gian giữa ứng dụng và cơ sở dữ liệu cho bảng HoaDon, ẩn chi tiết SQL khỏi lớp giao diện và cung cấp các phương thức tiện lợi để truy xuất, cập nhật, xóa, tìm kiếm, lấy hóa đơn theo mã, và quản lý liên kết với đặt phòng trong ứng dụng.

```

1 reference
public List<HoaDonDTO> TimKiemHoaDon(string keyword)
{
    string query = "SELECT * FROM HoaDon WHERE CAST(MaHD AS NVARCHAR) LIKE '%" + keyword + "%' OR CAST(MaDatPhong AS NVARCHAR) LIKE '%" + keyword + "%'";
    DataTable data = DataProvider.Instance.ExecuteNonQuery(query);
    List<HoaDonDTO> list = new List<HoaDonDTO>();
    foreach (DataRow row in data.Rows)
    {
        HoaDonDTO hd = new HoaDonDTO(
            Convert.ToInt32(row["MaHD"]),
            row["MaDatPhong"] != DBNull.Value ? Convert.ToInt32(row["MaDatPhong"]) : 0,
            row["NgayLap"] != DBNull.Value ? Convert.ToDateTime(row["NgayLap"]) : DateTime.Now,
            row["TongTien"] != DBNull.Value ? Convert.ToDecimal(row["TongTien"]) : 0
        );
        list.Add(hd);
    }
    return list;
}

```

Lớp **KhachHangDAL** là lớp trung gian giữa ứng dụng và cơ sở dữ liệu cho bảng KhachHang, ẩn SQL và cung cấp các phương thức để truy xuất, thêm, cập nhật, xóa, tìm kiếm, kiểm tra tồn tại, lấy danh sách mã, kiểm tra ràng buộc khách hàng, cũng như hỗ trợ lấy khách hàng theo mã và lọc danh sách khách hàng đã sử dụng.

```

1 reference
public bool KiemTraMaKHTonTai(int maKH)
{
    string query = "SELECT COUNT(*) FROM KhachHang WHERE MaKH=" + maKH;
    int count = (int)DataProvider.Instance.ExecuteScalar(query);
    return count > 0;
}

1 reference
public List<int> LayDanhSachMaKHDaSuDung()
{
    string query = "SELECT MaKH FROM KhachHang ORDER BY MaKH";
    DataTable data = DataProvider.Instance.ExecuteQuery(query);
    List<int> list = new List<int>();
    foreach (DataRow row in data.Rows)
    {
        list.Add(Convert.ToInt32(row["MaKH"]));
    }
    return list;
}

1 reference
public bool KiemTraKhachHangDangDuocThamChieu(int maKH)
{
    string query = "SELECT COUNT(*) FROM DatPhong WHERE MaKH=" + maKH;
    int count = (int)DataProvider.Instance.ExecuteScalar(query);
    return count > 0;
}

1 reference
public bool CapNhatKhachHang(int maKH, string tenKH, string gioiTinh, string sdt, string cmnd)
{
    string query = "UPDATE KhachHang SET TenKH=@tenKH, GioiTinh=@gioiTinh, SDT=@sdt, CMND=@cmnd WHERE MaKH=@maKH";
    int result = DataProvider.Instance.ExecuteNonQuery(query,
        new object[] { tenKH, gioiTinh, sdt, cmnd, maKH });
    return result > 0;
}

1 reference
public bool XoaKhachHang(int maKH)
{
    string query = "DELETE FROM KhachHang WHERE MaKH=@maKH";
    int result = DataProvider.Instance.ExecuteNonQuery(query, new object[] { maKH });
    return result > 0;
}

1 reference
public List<KhachHangDTO> TimKiemKhachHangTheoMa(int maKH)
{
    string query = "SELECT MaKH, TenKH, GioiTinh, SDT, CMND FROM KhachHang WHERE MaKH=@maKH";
    DataTable data = DataProvider.Instance.ExecuteQuery(query, new object[] { maKH });
    List<KhachHangDTO> list = new List<KhachHangDTO>();
    foreach (DataRow row in data.Rows)
    {
        KhachHangDTO kh = new KhachHangDTO(
            Convert.ToInt32(row["MaKH"]),
            row["TenKH"].ToString(),
            row["GioiTinh"].ToString(),
            row["SDT"].ToString(),
            row["CMND"].ToString()
        );
        list.Add(kh);
    }
    return list;
}

```

Lớp **LoaiPhongDAL** cung cấp các phương thức truy xuất, thêm, cập nhật, xóa và tìm kiếm thông tin loại phòng trong cơ sở dữ liệu, ẩn chi tiết SQL khỏi lớp giao diện.

```

1 reference
public void ThemLoaiPhong(string tenLoai, decimal giaPhong, string moTa)
{
    string query = "INSERT INTO LoaiPhong (TenLoai, GiaPhong, MoTa) VALUES (@tenLoai, @giaPhong, @moTa)";
    DataProvider.Instance.ExecuteNonQuery(query, new object[] { tenLoai, giaPhong, moTa });
}

1 reference
public void CapNhatLoaiPhong(int maLoai, string tenLoai, decimal giaPhong, string moTa)
{
    string query = "UPDATE LoaiPhong SET TenLoai = @tenLoai, GiaPhong = @giaPhong, MoTa = @moTa WHERE MaLoaiPhong = @maLoai";
    DataProvider.Instance.ExecuteNonQuery(query, new object[] { tenLoai, giaPhong, moTa, maLoai });
}

1 reference
public void XoaLoaiPhong(int maLoai)
{
    string query = "DELETE FROM LoaiPhong WHERE MaLoaiPhong = @maLoai";
    DataProvider.Instance.ExecuteNonQuery(query, new object[] { maLoai });
}

// Ví dụ truy vấn kiểu anh nói:
1 reference
public decimal? TimGiaTheoTen(string tenLoai)
{
    string query = "SELECT GiaPhong FROM LoaiPhong WHERE TenLoai = @tenLoai";
    object result = DataProvider.Instance.ExecuteScalar(query, new object[] { tenLoai });
    return result != null ? Convert.ToDecimal(result) : (decimal?)null;
}

1 reference
public List<LoaiPhongDTO> LayDanhSachLoaiPhong()
{
    string query = "SELECT MaLoaiPhong, TenLoai, GiaPhong, MoTa FROM LoaiPhong";
    DataTable data = DataProvider.Instance.ExecuteQuery(query);

    List<LoaiPhongDTO> list = new List<LoaiPhongDTO>();

    foreach (DataRow row in data.Rows)
    {
        list.Add(new LoaiPhongDTO(
            Convert.ToInt32(row["MaLoaiPhong"]),
            row["TenLoai"].ToString(),
            Convert.ToDecimal(row["GiaPhong"]),
            row["MoTa"].ToString()
        ));
    }

    return list;
}

```

Lớp **LoginDAL** cung cấp phương thức truy xuất thông tin tài khoản từ cơ sở dữ liệu theo tên đăng nhập và mật khẩu, ẩn chi tiết SQL khỏi lớp giao diện.

```

private DataProvider dp = DataProvider.Instance;

1 reference
public TaiKhoanDTO GetTaiKhoan(string username, string password)
{
    string query = "SELECT MaNV, TenDangNhap, Quyen FROM TaiKhoan " +
        "WHERE TenDangNhap = @username AND MatKhai = @password";

    DataTable dt = dp.ExecuteQuery(query, new object[] { username, password });

    if (dt.Rows.Count > 0)
    {
        DataRow row = dt.Rows[0];

        return new TaiKhoanDTO
        {
            MaNV = row["MaNV"] == DBNull.Value ? 0 : Convert.ToInt32(row["MaNV"]),
            TenDangNhap = row["TenDangNhap"]?.ToString() ?? "",
            Quyen = row["Quyen"]?.ToString() ?? "User"
        };
    }
    return null;
}

```

Lớp **NhanVienDAL** cung cấp các phương thức truy xuất danh sách nhân viên, lấy thông tin nhân viên theo mã, thêm nhân viên mới, cập nhật thông tin, xóa nhân viên, tìm kiếm nhân viên theo tên hoặc mã, kiểm tra nhân viên tồn tại, ẩn chi tiết SQL và quản lý kết nối cơ sở dữ liệu, giúp thao tác dữ liệu nhân viên toàn diện và tiện lợi trong ứng dụng.

```

public List<NhanVienDTO> LayDanhSachNhanVien()
{
    string query = "SELECT * FROM NHANVIEN";
    DataTable data = DataProvider.Instance.ExecuteQuery(query);
    List<NhanVienDTO> list = new List<NhanVienDTO>();

    foreach (DataRow row in data.Rows)
    {
        NhanVienDTO nv = new NhanVienDTO(
            Convert.ToInt32(row["MaNV"]),
            row["TenNV"].ToString(),
            row["GioiTinh"].ToString(),
            Convert.ToDateTime(row["NgaySinh"]),
            row["DiaChi"].ToString()
        );
        list.Add(nv);
    }
    return list;
}

1 reference
public bool ThemNhanVien(NhanVienDTO nv)
{
    string query = "INSERT INTO NHANVIEN (TenNV, GioiTinh, NgaySinh, DiaChi) VALUES (@ten, @gioitinh, @ngaysinh, @diachi)";
    int result = DataProvider.Instance.ExecuteNonQuery(query,
        new object[] { nv.TenNV, nv.GioiTinh, nv.NgaySinh, nv.DiaChi });
    return result > 0;
}

1 reference
public bool CapNhatNhanVien(NhanVienDTO nv)
{
    string query = "UPDATE NHANVIEN SET TenNV=@ten, GioiTinh=@gioitinh, NgaySinh=@ngaysinh, DiaChi=@diachi WHERE MaNV=@manv";
    int result = DataProvider.Instance.ExecuteNonQuery(query,
        new object[] { nv.TenNV, nv.GioiTinh, nv.NgaySinh, nv.DiaChi, nv.MaNV });
    return result > 0;
}

1 reference
public bool XoaNhanVien(int maNV)
{
    string query = "DELETE FROM NHANVIEN WHERE MaNV=@manv";
    int result = DataProvider.Instance.ExecuteNonQuery(query, new object[] { maNV });
    return result > 0;
}

```

Lớp **PhongDAL** cung cấp các phương thức truy xuất danh sách phòng, lấy phòng theo mã, thêm, cập nhật, xóa phòng, tìm kiếm phòng theo từ khóa, kiểm tra mã phòng tồn tại và lấy danh sách mã phòng đã sử dụng, ẩn chi tiết SQL, giúp quản lý dữ liệu phòng toàn diện và tiện lợi trong ứng dụng.

```
1 reference
public bool XoaPhong(int maPhong)
{
    string query = "DELETE FROM Phong WHERE MaPhong=@maPhong";
    int result = DataProvider.Instance.ExecuteNonQuery(query, new object[] { maPhong });
    return result > 0;
}

1 reference
public List<PhongDTO> TimKiemPhong(string keyword)
{
    string query = "SELECT p.MaPhong, p.TenPhong, p.MaLoaiPhong, ISNULL(lp.TenLoai, '') AS Loai, p.TrangThai " +
        "FROM Phong p LEFT JOIN LoaiPhong lp ON p.MaLoaiPhong = lp.MaLoaiPhong " +
        "WHERE CAST(p.MaPhong AS NVARCHAR) LIKE '%" + keyword + "%'";
    DataTable data = DataProvider.Instance.ExecuteQuery(query);
    List<PhongDTO> list = new List<PhongDTO>();
    foreach (DataRow row in data.Rows)
    {
        PhongDTO p = new PhongDTO(
            Convert.ToInt32(row["MaPhong"]),
            row["TenPhong"].ToString(),
            Convert.ToInt32(row["MaLoaiPhong"]),
            row["TrangThai"].ToString()
        );
        list.Add(p);
    }
    return list;
}

1 reference
public bool KiemTraMaPhongTonTai(int maPhong)
{
    string query = "SELECT COUNT(*) FROM Phong WHERE MaPhong=" + maPhong;
    int count = (int)DataProvider.Instance.ExecuteScalar(query);
    return count > 0;
}

1 reference
public List<int> LayDanhSachMaPhongDaSuDung()
{
    string query = "SELECT MaPhong FROM Phong ORDER BY MaPhong";
    DataTable data = DataProvider.Instance.ExecuteQuery(query);
    List<int> list = new List<int>();
    foreach (DataRow row in data.Rows)
    {
        list.Add(Convert.ToInt32(row["MaPhong"]));
    }
    return list;
}
```

Lớp **TaiKhoanDAL** cung cấp các phương thức lấy tài khoản theo mã nhân viên và cập nhật mật khẩu, ẩn chi tiết SQL, giúp quản lý thông tin đăng nhập tiện lợi và bảo mật trong ứng dụng.


```

public class TaiKhoanDAL
{
    1 reference
    public TaiKhoanDTO LayTaiKhoanTheoMaNV(int maNV)
    {
        string query = "SELECT MaNV, MatKhai FROM TaiKhoan WHERE MaNV=@maNV";
        DataTable data = DataProvider.Instance.ExecuteNonQuery(query, new object[] { maNV });
        if (data.Rows.Count > 0)
        {
            DataRow row = data.Rows[0];
            return new TaiKhoanDTO(
                Convert.ToInt32(row["MaNV"]),
                row["MatKhai"].ToString()
            );
        }
        return null;
    }

    1 reference
    public bool CapNhatMatKhai(int maNV, string matKhaiMoi)
    {
        string query = "UPDATE TaiKhoan SET MatKhai=@matKhai WHERE MaNV=@maNV";
        int result = DataProvider.Instance.ExecuteNonQuery(query,
            new object[] { matKhaiMoi, maNV });
        return result > 0;
    }
}

```

3.4 Các Procedured

Lưu script tạo bảng, **procedure**, **function**, **trigger**... Bao gồm **folder SQL** và **packages** chứa các gói mở rộng từ Microsoft và hệ thống.

- **Folder SQL:**
 - Chứa toàn bộ các file script dùng để tạo cơ sở dữ liệu, bảng, stored procedure, function và trigger.

```

-----
CREATE PROCEDURE findPhong
    @keyword NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    SELECT
        p.MaPhong,
        p.TenPhong, |
        p.MaLoaiPhong,
        ISNULL(lp.TenLoai, '') AS Loai,
        p.TrangThai
    FROM Phong p
    LEFT JOIN LoaiPhong lp ON p.MaLoaiPhong = lp.MaLoaiPhong
    WHERE CAST(p.MaPhong AS NVARCHAR(50)) LIKE '%' + @keyword + '%'
    OR p.TenPhong LIKE '%' + @keyword + '%'
    OR lp.TenLoai LIKE '%' + @keyword + '%';
END
GO

```





















- Giúp quản lý và triển khai các thay đổi về cơ sở dữ liệu một cách có hệ thống.

```
CREATE PROCEDURE getDonGia
    @MaPhong INT
AS
BEGIN
    SET NOCOUNT ON;

    SELECT lp.GiaPhong
    FROM Phong p
    JOIN LoaiPhong lp ON p.MaLoaiPhong = lp.MaLoaiPhong
    WHERE p.MaPhong = @MaPhong;
END
GO
```

- **Packages:**

- Bao gồm các gói mở rộng từ Microsoft và hệ thống, hỗ trợ việc kết nối, thao tác và xử lý dữ liệu trên SQL Server.

Name	Date modified	Type	Size
 Azure.Core.1.47.1	11/15/2025 3:29 PM	File folder	
 Azure.Identity.1.14.2	11/15/2025 3:29 PM	File folder	
 EntityFramework.6.5.1	11/15/2025 12:15 AM	File folder	
 Microsoft.Bcl.AsyncInterfaces.8.0.0	11/15/2025 3:29 PM	File folder	
 Microsoft.Bcl.Cryptography.8.0.0	11/15/2025 3:29 PM	File folder	
 Microsoft.Data.SqlClient.6.1.3	11/15/2025 3:30 PM	File folder	
 Microsoft.Data.SqlClient.SNI.6.0.2	11/15/2025 3:29 PM	File folder	
 Microsoft.Extensions.Caching.Abstractions...	11/15/2025 3:29 PM	File folder	
 Microsoft.Extensions.Caching.Memory.8.0.1	11/15/2025 3:29 PM	File folder	
 Microsoft.Extensions.DependencyInjection...	11/15/2025 3:29 PM	File folder	
 Microsoft.Extensions.Logging.Abstractions...	11/15/2025 3:29 PM	File folder	
 Microsoft.Extensions.Options.8.0.2	11/15/2025 3:29 PM	File folder	
 Microsoft.Extensions.Primitives.8.0.0	11/15/2025 3:29 PM	File folder	
 Microsoft.Identity.Client.4.73.1	11/15/2025 3:29 PM	File folder	
 Microsoft.Identity.Client.Extensions.Msal....	11/15/2025 3:29 PM	File folder	
 Microsoft.IdentityModel.Abstractions.7.7.1	11/15/2025 3:29 PM	File folder	
 Microsoft.IdentityModel.JsonWebTokens....	11/15/2025 3:29 PM	File folder	
 Microsoft.IdentityModel.Logging.7.7.1	11/15/2025 3:29 PM	File folder	
 Microsoft.IdentityModel.Protocols.7.7.1	11/15/2025 3:29 PM	File folder	

- Đảm bảo các tính năng nâng cao và tích hợp với ứng dụng được thực hiện ổn định.

System.Diagnostics.DiagnosticSource.8.0.1	11/15/2025 3:29 PM	File folder
System.IdentityModel.Tokens.Jwt.7.7.1	11/15/2025 3:29 PM	File folder
System.IO.FileSystem.AccessControl.5.0.0	11/15/2025 3:29 PM	File folder
System.Memory.4.5.5	11/15/2025 3:29 PM	File folder
System.Memory.Data.8.0.1	11/15/2025 3:29 PM	File folder
System.Numerics.Vectors.4.5.0	11/15/2025 3:29 PM	File folder
System.Runtime.CompilerServices.Unsafe....	11/15/2025 3:29 PM	File folder
System.Security.AccessControl.5.0.0	11/15/2025 3:29 PM	File folder
System.Security.Cryptography.Pkcs.8.0.1	11/15/2025 3:29 PM	File folder
System.Security.Cryptography.ProtectedD...	11/15/2025 3:29 PM	File folder
System.Security.Principal.Windows.5.0.0	11/15/2025 3:29 PM	File folder
System.Text.Encodings.Web.8.0.0	11/15/2025 3:29 PM	File folder
System.Text.Json.8.0.5	11/15/2025 3:29 PM	File folder
System.Threading.Tasks.Extensions.4.5.4	11/15/2025 3:29 PM	File folder
System.ValueTuple.4.5.0	11/15/2025 3:29 PM	File folder

3.5 Lớp DTO

Lớp Thực thể (Entity) mô tả dữ liệu của từng bảng gọi là lớp DTO .

BaoCaoDoanhThuDTO là lớp dùng để **truyền dữ liệu báo cáo doanh thu**, chứa các thông tin:

- **MaHD** – Mã hóa đơn
- **MaDatPhong** – Mã đặt phòng liên quan
- **NgayLap** – Ngày lập hóa đơn
- **TongTien** – Tổng số tiền thu được

Lớp có **constructor mặc định** và **constructor đầy đủ tham số** để khởi tạo nhanh đối tượng.

ChiTietDatPhongDTO là lớp dùng để **truyền dữ liệu chi tiết của một lượt đặt phòng**, bao gồm:

- **MaDatPhong** – Mã đặt phòng
- **MaPhong** – Phòng được đặt
- **DonGia** – Đơn giá của phòng tại thời điểm đặt

Lớp có **constructor mặc định** và **constructor đầy đủ tham số** để thuận tiện tạo đối tượng khi xử lý dữ liệu.

DatPhongDTO là lớp dùng để **truyền dữ liệu của một lượt đặt phòng** trong hệ thống quản lý homestay. Lớp bao gồm các thông tin hiển thị cho người dùng và các khóa liên quan để xử lý nghiệp vụ:

Thuộc tính hiển thị

- **MaDatPhong** – Mã lượt đặt phòng
- **TenKH** – Tên khách hàng
- **TenNV** – Tên nhân viên xử lý
- **TenPhong** – Tên phòng được đặt
- **NgayDat** – Ngày tạo phiếu đặt phòng
- **NgayDen** – Ngày khách đến
- **NgayDi** – Ngày khách trả phòng
- **TongTien** – Tổng số tiền phải thanh toán

Thuộc tính ẩn (dùng xử lý nội bộ, không hiển thị DataGridView)

- **MaKH** – Mã khách hàng
- **MaNV** – Mã nhân viên
- **MaPhong** – Mã phòng

Constructor

- **Constructor mặc định**
- **Constructor với mã (MaKH, MaNV, MaPhong)** dùng khi thêm / cập nhật dữ liệu

- **Constructor với tên (TenKH, TenNV, TenPhong)** dùng khi load dữ liệu hiển thị lên giao diện
-

HoaDonDTO là lớp dùng để truyền dữ liệu của một hóa đơn trong hệ thống quản lý homestay. Lớp chứa thông tin hiển thị cho người dùng và phục vụ cho việc xử lý nghiệp vụ thanh toán.

Thuộc tính

- **MaHD** – Mã hóa đơn
- **MaDatPhong** – Mã lượt đặt phòng liên quan
- **NgayLap** – Ngày lập hóa đơn
- **TongTien** – Tổng số tiền phải thanh toán

Constructor

- Constructor mặc định
 - Constructor đầy đủ tham số (**maHD, maDatPhong, ngayLap, tongTien**) dùng khi khởi tạo hóa đơn với đầy đủ thông tin
-

KhachHangDTO là lớp dùng để truyền dữ liệu của khách hàng trong hệ thống quản lý homestay. Lớp chứa các thông tin cơ bản phục vụ cho việc lưu trữ, tìm kiếm và hiển thị khách hàng.

Thuộc tính

- **MaKH** – Mã khách hàng
- **TenKH** – Tên khách hàng
- **GioiTinh** – Giới tính
- **SĐT** – Số điện thoại
- **CMND** – Chứng minh nhân dân / CCCD

Constructor

- Constructor mặc định

- Constructor đầy đủ tham số (**maKH**, **tenKH**, **gioiTinh**, **sdt**, **cmnd**) dùng khi tạo mới hoặc nạp dữ liệu khách hàng
-

LoaiPhongDTO là lớp dùng để truyền dữ liệu về loại phòng trong hệ thống quản lý homestay. Lớp chứa thông tin phục vụ hiển thị và xử lý nghiệp vụ liên quan đến phân loại phòng.

Thuộc tính

- **MaLoaiPhong** – Mã loại phòng
- **TenLoai** – Tên loại phòng (ví dụ: Standard, VIP, Suite...)
- **GiaPhong** – Đơn giá của loại phòng
- **MoTa** – Mô tả chi tiết về loại phòng

Constructor

- Constructor mặc định
 - Constructor đầy đủ tham số (**maLoaiPhong**, **tenLoai**, **giaPhong**, **moTa**) dùng để khởi tạo loại phòng với đầy đủ thông tin
-

NhanVienDTO là lớp dùng để truyền dữ liệu của nhân viên trong hệ thống quản lý homestay. Lớp chứa các thông tin cơ bản phục vụ cho việc quản lý, tìm kiếm và hiển thị danh sách nhân viên.

Thuộc tính

- **MaNV** – Mã nhân viên
- **TenNV** – Tên nhân viên
- **GioiTinh** – Giới tính
- **NgaySinh** – Ngày sinh
- **DiaChi** – Địa chỉ liên hệ

Constructor

- Constructor mặc định
 - Constructor đầy đủ tham số (**maNV**, **tenNV**, **gioiTinh**, **ngaySinh**, **diaChi**) dùng để khởi tạo hoặc nạp dữ liệu nhân viên
-

PhongDTO là lớp dùng để truyền dữ liệu của phòng trong hệ thống quản lý homestay .
Lớp cung cấp thông tin hiển thị và hỗ trợ xử lý nghiệp vụ quản lý phòng.

Thuộc tính

- **MaPhong** – Mã phòng
- **TenPhong** – Tên phòng
- **MaLoaiPhong** – Mã loại phòng
- **TrangThai** – Trạng thái phòng (Trống, Đang thuê, Bảo trì...)

Constructor

- Constructor mặc định
 - Constructor đầy đủ tham số (**maPhong**, **tenPhong**, **maLoaiPhong**, **trangThai**) dùng khi tạo mới hoặc nạp dữ liệu phòng
-

TaiKhoanDTO là lớp dùng để truyền dữ liệu tài khoản đăng nhập trong hệ thống quản lý homestay. Lớp chứa thông tin xác thực và phân quyền của nhân viên.

Thuộc tính

- **TenDangNhap** – Tên đăng nhập
- **MatKhau** – Mật khẩu
- **MaNV** – Mã nhân viên liên kết với tài khoản
- **Quyen** – Quyền truy cập (Admin, Nhân viên...)

Constructor

- Constructor mặc định
- Constructor rút gọn (**maNV**, **matKhau**) dùng cho các thao tác kiểm tra hoặc cập nhật mật khẩu
- Constructor đầy đủ (**tenDangNhap**, **matKhau**, **maNV**, **quyen**) dùng khi khởi tạo tài khoản với đầy đủ thông tin

3.6 Kết chương.

Trong chương này, chúng tôi đã trình bày thiết kế cơ sở dữ liệu và triển khai các lớp truy cập dữ liệu (DAL) cho hệ thống quản lý homestay, bao gồm các bảng chính như Khách hàng, Nhân viên, Phòng, Loại phòng, Hóa đơn và Tài khoản. Các lớp DAL cung cấp các phương thức thêm, sửa, xóa, truy xuất và tìm kiếm dữ liệu, giúp tách biệt giao diện người dùng khỏi chi tiết SQL, đảm bảo tính bảo trì và mở rộng của hệ thống. Chương này kết luận rằng DAL đóng vai trò trung gian quan trọng, đảm bảo dữ liệu được quản lý hiệu quả, an toàn và thuận tiện cho việc triển khai các chức năng nghiệp vụ tiếp theo.

Kết luận

Sau quá trình nghiên cứu, tìm hiểu và xây dựng chương trình **Quản lý Homestay**, đề tài đã đạt được các kết quả nhất định như:

- Hoàn thiện giao diện thân thiện, dễ sử dụng cho người dùng.
- Thực hiện được các chức năng cơ bản như quản lý phòng, khách hàng, đặt phòng, hóa đơn và thanh toán.
- Cơ sở dữ liệu được thiết kế hợp lý, đảm bảo tính toàn vẹn và dễ dàng mở rộng.
- Kết nối giữa ứng dụng và cơ sở dữ liệu SQL Server hoạt động ổn định, giúp việc xử lý dữ liệu nhanh chóng và chính xác.
- Kết nối được với Excel để xuất dữ liệu ra file Excel lấy dữ liệu từ SQL .

Kết quả đạt được chứng minh rằng việc áp dụng công nghệ .NET và SQL Server trong xây dựng hệ thống quản lý homestay là hoàn toàn khả thi và hiệu quả.

Hạn chế và hướng phát triển của đề tài.

Hạn chế:

- Chưa có chức năng quản lý **dịch vụ** mà khách hàng sử dụng trong quá trình lưu trú (ăn uống, giặt ủi, thuê xe, v.v.).
- Giao diện tuy trực quan nhưng còn đơn giản, chưa thân thiện trên nhiều độ phân giải màn hình.
- Hệ thống chưa triển khai trên môi trường mạng LAN hoặc Internet, nên chỉ hoạt động cục bộ.

Hướng phát triển:

- Bổ sung module **quản lý dịch vụ** và **thanh toán dịch vụ đi kèm**.
- Nâng cấp giao diện hiện đại hơn bằng **WPF** hoặc **WebApp (.NET Core MVC)**.
- Phát triển phiên bản **trực tuyến (Online Booking)** cho phép khách hàng đặt phòng qua website.
- Áp dụng **báo cáo thống kê trực quan (charts, dashboard)** giúp quản lý theo dõi tình hình kinh doanh dễ dàng hơn.

Danh mục sách tham khảo

1. Trang web <https://learn.microsoft.com> – Tài liệu hướng dẫn về .NET Framework và ADO.NET [tham khảo ngày 10 tháng 11 năm 2025]
2. Nguyễn Văn Vĩnh (2020), Lập trình C# cơ bản và nâng cao, Nhà xuất bản Thông tin và Truyền thông .
3. Cao Thị Luyện – Slide bài giảng và video hướng dẫn môn Lập Trình Trực Quan – UTC .
4. Trang web <https://www.sqltutorial.org> – Hướng dẫn sử dụng SQL Server [tham khảo ngày 9 tháng 11 năm 2025].