**TRƯỜNG ĐẠI HỌC QUỐC TẾ HỒNG BÀNG**

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

**🙢🕮🙠**



**BÁO CÁO ĐỒ ÁN**

**MÔN HỌC  
CÔNG NGHỆ .NET**

Đề tài: Xây dựng phần mềm quản lý khách sạn

**Giảng viên hướng dẫn: Nguyễn Minh Đế**

**Sinh viên thực hiện: Nguyễn Minh Vũ, Nguyễn Võ Văn**

**Mã số sinh viên: 2211110063, 2211110024**

**TP. Hồ Chí Minh, 2024**

1. LỜI CẢM ƠN

Để hoàn thành đề tài này, em đã nhận được sự hướng dẫn, giúp đỡ và góp ý nhiệt tình của quý thầy cô trường Đại Học Quốc Tế Hồng Bàng và thầy Nguyễn Minh Đế.

Em xin gửi lời biết ơn sâu sắc đến thầy Nguyễn Minh Đế đã dành nhiều thời gian và tâm huyết hướng dẫn nghiên cứu và giúp em hoàn thành môn học.

Em cũng xin chân thành cảm ơn đến quý thầy cô trường Đại học Quốc Tế Hồng Bàng, đặc biệt là những thầy cô đã tận tình dạy bảo cho em suốt thời gian học tập tại trường.

Em đã có nhiều cố gắng hoàn thiện dự án bằng tất cả năng lực của mình, tuy nhiên không thể tránh khỏi nhiều thiếu sót, rất mong nhận được những đóng góp quý báu của quý thầy cô và các bạn.

TP.HCM, ... Tháng ... Năm 2024

Người thực hiện

Nguyễn Minh Vũ, Nguyễn Võ Văn

1. TRANG CAM KẾT

Tôi xin cam kết báo cáo thường kỳ này được hoàn thành dựa trên các kết quả thực hiện bài thực hành của tôi và các mã nguồn và kết quả này chưa được dùng cho bất cứ báo cáo của sinh viên nào khác.

*TP.HCM, ngày ... tháng ... năm 2024*

Người thực hiện

Nguyễn Minh Vũ, Nguyễn Võ Văn

1. NHẬN XÉT CỦA GIẢNG VIÊN

TP.HCM, Ngày ... Tháng ... Năm 2024

Chữ ký giảng viên

1. MỤC LỤC

[I. LỜI CẢM ƠN 1](#_Toc181969019)

[II. TRANG CAM KẾT 2](#_Toc181969020)

[III. NHẬN XÉT CỦA GIẢNG VIÊN 3](#_Toc181969021)

[IV. MỤC LỤC 4](#_Toc181969022)

[I. GIỚI THIỆU ĐỒ ÁN 6](#_Toc181969023)

[1.1. Định nghĩa vấn đề 6](#_Toc181969024)

[1.2. Phạm vi đồ án 6](#_Toc181969025)

[1.3. Tính cấp thiết của đề tài 6](#_Toc181969026)

[1.3.1. Mục tiêu 7](#_Toc181969027)

[1.3.2. Tại sao chọn đề tài này ? 8](#_Toc181969028)

[II. PHÂN TÍCH ĐỀ TÀI 8](#_Toc181969029)

[2.1. Phân tích yêu cầu hệ thống 8](#_Toc181969031)

[2.1.1. Yêu cầu người dùng 8](#_Toc181969032)

[2.2. Các công việc cần giải quyết 9](#_Toc181969033)

[III. THIẾT KẾ 10](#_Toc181969034)

[3.1. Database Design 10](#_Toc181969036)

[3.1.1. Mô tả các thực thể 10](#_Toc181969037)

[3.1.2. Mô hình dữ liệu 12](#_Toc181969038)

[3.2. Sơ đồ cấu trúc hệ thống 13](#_Toc181969039)

[IV. CÀI ĐẶT VÀ KIỂM THỬ 14](#_Toc181969040)

[4.1. Giao diện đăng nhập 14](#_Toc181969042)

[4.2. Giao diện trang chủ 16](#_Toc181969043)

[4.3. Giao diện thêm phòng (Add Room) 19](#_Toc181969044)

[4.4. Giao diện thêm thông tin khách hàng (Customer) 27](#_Toc181969045)

[4.5. Giao diện đặt phòng (Bookings) 29](#_Toc181969046)

[4.6. Giao diện đặt dịch vụ (Service) 31](#_Toc181969047)

[4.7. Giao diện thanh toán và trả phòng (Check out) 33](#_Toc181969048)

[4.8. Giao diện của quản lý (Manage) 34](#_Toc181969049)

[4.9. Giao diện thu nhập (Total salary) 37](#_Toc181969050)

[V. ĐỊNH HƯỚNG PHÁT TRIỂN 39](#_Toc181969051)

[VI. KẾT LUẬN 40](#_Toc181969052)

[VII. TÀI LIỆU THAM KHẢO 41](#_Toc181969053)

[VIII. PHỤ LỤC 41](#_Toc181969054)

1. GIỚI THIỆU ĐỒ ÁN

## Định nghĩa vấn đề

Trong thời đại công nghệ số hiện nay, việc quản lý các hoạt động của khách sạn bằng phương pháp thủ công hoặc thông qua các phần mềm đơn giản đã không còn đáp ứng được nhu cầu ngày càng phức tạp. Các khách sạn thường gặp phải nhiều khó khăn trong việc quản lý phòng, thông tin khách hàng, đặt phòng, cập nhật tình trạng phòng, và quá trình thanh toán, dẫn đến tình trạng đặt phòng trùng lặp, thông tin không chính xác, và khó khăn trong việc phục vụ khách hàng.

Để giải quyết các vấn đề trên, một hệ thống phần mềm quản lý khách sạn được phát triển nhằm giúp quản lý các hoạt động của khách sạn một cách dễ dàng và hiệu quả hơn. Phần mềm này sẽ hỗ trợ nhân viên khách sạn trong việc quản lý đặt phòng, thông tin khách hàng, thanh toán, và các dịch vụ đi kèm khác. Hệ thống không chỉ giúp tối ưu hoá công việc quản lý mà còn cải thiện trải nghiệm cho cả khách hàng và nhân viên, từ đó tăng cường sự hài lòng và hiệu suất làm việc.

Phần mềm sẽ đảm bảo thông tin phòng được cập nhật liên tục, thông tin khách hàng được quản lý chính xác, và quy trình thanh toán được thực hiện nhanh chóng, minh bạch. Ngoài ra, các dịch vụ khác như gọi phục vụ, yêu cầu dọn phòng cũng sẽ được tích hợp, giúp khách sạn vận hành một cách trơn tru và hiện đại hơn.

## Phạm vi đồ án

Đồ án này nhằm phát triển hệ thống phần mềm quản lý khách sạn, giúp tối ưu hoá các quy trình quản lý phòng, khách hàng, và thanh toán. Phần mềm sẽ hỗ trợ các tính năng chính sau:

* **Quản lý đặt phòng:** Cho phép nhân viên khách sạn thực hiện các thao tác liên quan đến đặt phòng như thêm, sửa, hủy đặt phòng và kiểm tra tình trạng phòng theo thời gian thực.
* **Quản lý thông tin khách hàng:** Hỗ trợ lưu trữ và quản lý thông tin chi tiết của khách hàng, bao gồm họ tên, số điện thoại, địa chỉ, số căn cước công dân, và các yêu cầu đặc biệt.
* **Quản lý phòng:** Theo dõi tình trạng phòng (phòng trống, đã đặt, đang sử dụng), loại phòng (Standard, Deluxe, Suite), và giá phòng. Hỗ trợ cập nhật trạng thái phòng (sạch, đang dọn dẹp, cần sửa chữa).
* **Quản lý thanh toán và hóa đơn:** Hỗ trợ tính toán chi phí phòng và các dịch vụ bổ sung. Hệ thống sẽ tạo hóa đơn tự động cho khách khi check-out, và ghi nhận các giao dịch thanh toán.
* **Quản lý nhân viên:** Hỗ trợ quản lý thông tin nhân viên như họ tên, căn cước, lương thưởng và lịch làm việc, giúp dễ dàng theo dõi hiệu suất làm việc. Ngoài ra để sử dụng phần mềm, cần các nhận viên có trách nhiệm riêng để sử dụng phần mềm, trách người lạ hoặc không phận sự sử dụng được phần mềm. Các nhân viên chỉ định sẽ được cấp các tài khoản nhất định để thực hiện các chức năng cơ bản phục vụ khách hàng.
* **Quản lý dịch vụ:** Hỗ trợ theo dõi và quản lý các dịch vụ phục vụ phòng, ăn uống và các dịch vụ bổ sung khác mà khách hàng yêu cầu trong quá trình lưu trú. Cũng như phản hồi của khách hàng về khách sạn.
* **Quản lý thu nhập và chi tiêu:** Hỗ trợ quản lý nhập các mặt hàng đồ dùng, nhu yếu phẩm cũng như là lương của từng nhân viên trực thuộc. Quản lý thu nhập doanh nghiệp cũng là một phần quan trọng để quản lý chi tiêu và nhu cầu khách hàng.

**Ngoài phạm vi:**

* Hệ thống không bao gồm tích hợp với các trang web đặt phòng trực tuyến như Booking.com hay Agoda.
* Hệ thống không tích hợp các phương thức thanh toán trực tuyến qua thẻ ngân hàng hay cổng thanh toán.
* Hệ thống không hỗ trợ giao diện dành cho khách hàng sử dụng trực tiếp, chỉ dành cho nhân viên khách sạn.

## Tính cấp thiết của đề tài

Cùng với sự phát triển của ngành du lịch và dịch vụ lưu trú, các khách sạn cần phải áp dụng công nghệ hiện đại để quản lý hoạt động hiệu quả hơn. Việc sử dụng phần mềm quản lý khách sạn không chỉ giúp giảm thiểu sai sót trong quá trình vận hành mà còn mang lại sự tiện lợi và minh bạch trong việc phục vụ khách hàng. Trong bối cảnh cạnh tranh ngày càng khốc liệt, một phần mềm quản lý chuyên nghiệp là yếu tố cần thiết để giúp khách sạn phát triển bền vững.

### Mục tiêu

Xây dựng phần mềm quản lý khách sạn với các chức năng cơ bản như quản lý đặt phòng, khách hàng, dịch vụ, và thanh toán. Phần mềm cũng cung cấp chức năng dành cho quản trị viên để quản lý thông tin khách sạn và điều chỉnh hệ thống. Giao diện phần mềm cần phải dễ sử dụng, thân thiện với người dùng, và phù hợp với nhiều loại thiết bị

### Tại sao chọn đề tài này ?

Đề tài "Hệ thống Quản lý Khách sạn" được chọn vì mang tính thực tiễn cao, giúp quản lý hiệu quả các hoạt động quan trọng như đặt phòng, thanh toán và quản lý thông tin khách hàng, phù hợp với nhu cầu thực tế của ngành du lịch và khách sạn. Hệ thống này sẽ giúp tối ưu hóa quy trình làm việc, giảm thiểu các công việc thủ công, từ đó tăng cường hiệu quả quản lý và tiết kiệm thời gian cho nhân viên. Việc thực hiện đề tài còn là cơ hội để áp dụng và phát triển các kiến thức về lập trình, cơ sở dữ liệu và thiết kế hệ thống, mang đến trải nghiệm học tập bổ ích. Ngoài ra, hệ thống còn có tiềm năng mở rộng với nhiều tính năng khác như quản lý sự kiện, dịch vụ khách hàng và đặt phòng trực tuyến. Điều này không chỉ giúp khách sạn hoạt động chuyên nghiệp hơn mà còn cải thiện đáng kể trải nghiệm khách hàng, từ đó nâng cao chất lượng dịch vụ.

1. PHÂN TÍCH ĐỀ TÀI

## Phân tích yêu cầu hệ thống

Hệ thống quản lý khách sạn cần cung cấp các chức năng chính sau:

* **Quản lý đăng nhập và phân quyền người dùng:**
  + Hệ thống phải hỗ trợ quản trị viên và nhân viên khách sạn đăng nhập bằng tài khoản đã được tạo trước.
  + Có phân quyền để quản lý các tính năng phù hợp với vai trò của người dùng (quản trị viên, nhân viên).
    - Với quyền là Admin sẽ có tài khoản mặc định là:

Username: admin

Password: admin

* **Quản lý đặt phòng:**
  + Hỗ trợ đặt phòng cho khách hàng bao gồm các thông tin như số phòng, loại phòng, ngày đến và ngày đi, giá phòng,…v.v.
  + Kiểm tra tình trạng phòng trống theo thời gian thực, cập nhật trạng thái phòng sau mỗi lần đặt.
* **Quản lý thông tin khách hàng:**
  + Hệ thống cần lưu trữ và quản lý thông tin cá nhân của khách hàng (họ tên, số CCCD, địa chỉ, số điện thoại, email, ngày sinh, giới tính).
  + Hỗ trợ tìm kiếm thông tin khách hàng nhanh chóng.
* **Quản lý thanh toán:**
  + Hệ thống phải hỗ trợ tính toán chi phí dịch vụ và giá phòng cho khách hàng.
  + Quản lý các phương thức thanh toán như tiền mặt.
* **Quản lý thông tin phòng:**
  + Thông tin chi tiết về phòng như loại phòng (STD, SUP, DLX, SUT), số giường, view phòng, giá phòng cần được quản lý chặt chẽ.
  + Cập nhật tình trạng vệ sinh và bảo dưỡng phòng (sạch, đang dọn dẹp, cần sửa chữa).
* **Quản lý nhân viên:**
  + Lưu trữ thông tin nhân viên (họ tên, số CCCD, ngày sinh, giới tính, email, lương).
  + Theo dõi và quản lý chấm công hàng ngày của nhân viên.
* **Quản lý báo cáo và phục vụ:**
  + Cho phép nhân viên ghi nhận các báo cáo sự cố từ khách hàng.
  + Theo dõi và quản lý các yêu cầu dịch vụ phòng của khách (gọi đồ ăn, dọn phòng...).
* **Quản lý thu nhập và chi tiêu:**
  + Được biểu diễn dưới các dạng biểu đồ khác nhau.
  + Cho phép quản lý và người chủ biết được chi tiêu, thu nhập qua từng tháng để có thể điều tiết và vận hành phù hợp với các mảng đối tượng khách hàng.
  + Phát triển các xu hướng, mặt hàng, các mốc thời gian quan trọng trong việc quản lý khách sạn.

### Yêu cầu người dùng

 **Quản trị viên:** Quản lý toàn bộ hệ thống.

 **Nhân viên lễ tân:** Xử lý đặt phòng, check-in, check-out và quản lý các yêu cầu dịch vụ của khách hàng.

 **Nhân viên:** Phục vụ khách hàng, cập nhật tình trạng vệ sinh phòng và báo cáo các vấn đề cần sửa chữa.

## Các công việc cần giải quyết

1. **Phân tích yêu cầu**

* Tiến hành thu thập và phân tích yêu cầu từ khách hàng, nhân viên và quản trị viên của khách sạn để đảm bảo hệ thống đáp ứng đầy đủ các yêu cầu thực tế.

**2. Thiết kế hệ thống**

* Thiết kế giao diện người dùng thân thiện, dễ sử dụng, phù hợp với nghiệp vụ của khách sạn.
* Thiết kế cơ sở dữ liệu bao gồm các bảng liên quan đến phòng, khách hàng, nhân viên, hóa đơn, và dịch vụ.

**3. Xây dựng và phát triển hệ thống**

* Lập trình các chức năng quản lý phòng, khách hàng, nhân viên, dịch vụ, và thanh toán.
* Phát triển giao diện đăng nhập và phân quyền người dùng.

**4. Kiểm thử hệ thống**

* Kiểm thử các chức năng của hệ thống để đảm bảo tính ổn định và chính xác.
* Thử nghiệm hệ thống với các trường hợp thực tế để đảm bảo hệ thống hoạt động hiệu quả trong môi trường thực tế.

**5. Triển khai và đào tạo**

* Triển khai hệ thống tại khách sạn và tổ chức đào tạo nhân viên khách sạn để sử dụng hệ thống hiệu quả.

**6. Bảo trì và nâng cấp**

* Định kỳ bảo trì hệ thống để đảm bảo hoạt động ổn định.
* Nâng cấp và bổ sung tính năng mới dựa trên yêu cầu của khách sạn trong tương lai.

1. THIẾT KẾ

## Database Design

### Mô tả các thực thể

* **Room**: Chứa thông tin về phòng (maphong, roomnumber, roomtype, numbed, view\_room, image\_room, price).
* **Update\_room**: Lưu trữ trạng thái phòng (maphong, status\_room, house\_keeping).
* **Customer**: Lưu trữ thông tin khách hàng (cccd\_cus, first\_name, last\_name, sdt, email, gioitinh, ngaysinh, address\_cus).
* **Bookings**: Lưu trữ thông tin đặt phòng (stt, cccd\_cus, status\_room, house\_keeping, roomtype, numbed, view\_room, date\_ci, date\_co, group\_customer, maphong, roomnumber, cccd\_em, price).
* **Employee**: Lưu trữ thông tin nhân viên (cccd\_em, first\_name, last\_name, sdt, email, gioitinh, ngaysinh, luong).
* **Report**: Chứa các báo cáo về khách và phòng (cccd\_cus, maphong, ghichu).
* **Serve**: Lưu trữ thông tin dịch vụ phòng (stt, cccd\_cus, maphong, other\_booking, anuong, call\_serve, cost)
* **Chamcong**: Lưu trữ thông tin chấm công (maphieu, cccd\_em, ngay, ca1, ca2, ca3, ca4, note).
* **Account**: Chứa tài khoản nhân viên (id, username, password, cccd\_em).
* **Checkout**: Lưu trữ thông tin trả phòng (id, cccd\_cus, first\_name, last\_name, maphong, sophong, date\_ci, date\_co).
* **Chitieu**: Lưu trữ các khoản chi tiêu của khách sạn (sttchi, ngay, tendogiadung, gianhapdogiadung, tennguyenlieu, gianhapnguyenlieu, tennhuyeupham, gianhuyeupham).
* **Salary**: Chứa thông tin về lương và chi phí doanh nghiệp (stt, month, year, tongthu, tongchi, loinhuandoanhnghiep).

### Mô hình dữ liệu

A screenshot of a computer flow chart

Description automatically generated

A screenshot of a checkout

Description automatically generated

## Sơ đồ cấu trúc hệ thống

A grid with text on it

Description automatically generated

1. CÀI ĐẶT VÀ KIỂM THỬ

## Giao diện đăng nhập

A screenshot of a hotel

Description automatically generated

Hình 4.1. Giao diện đăng nhập

* Khi người dùng đăng nhập vào hệ thống thì hệ thống sẽ kiểm tra username và password từ database.
* Nếu sai thì sẽ đưa ra thông báo ("username or password was wrong!!! "), còn nếu đăng nhập đúng sẽ vào giao diện trang chủ để có thể sử dụng phần mềm, quyền sử dụng tùy vào từng tài khoản.
* Admin sẽ là tài khoản có quyền cao nhất. Với:

+ User name : admin

+ Password: admin

* Nếu người dung ấn thoát thì chương trình sẽ tự động thoát ngay lập tức.
* Nguyên lý hoạt động:
* B1: Thiết kế giao diện như ảnh
* B2: Cài đặt Event Click cho Button Submit và Destroy App (góc phải màn hình)
* Cài đặt Button Submit

+ Thêm ràng buộc để buộc người dùng phải nhập đầy đủ những thông tin cần thiết trong TextBox.

+ Sử dụng Package SQL để thực hiện truy vấn vào Database

+ Kiểm tra Username và Password được nhập vào TextBox. Nếu tồn tại thì thực hiện chức năng đăng nhập và xuất hiện màn hình Dashboard.

public string Username { get { return textusername.Text; } }

public string Password { get { return textpassword.Text; } }

public LoginFormNew()

{

InitializeComponent();

this.textusername.KeyDown += new KeyEventHandler(this.TextBox\_KeyDown);

this.textpassword.KeyDown += new KeyEventHandler(this.TextBox\_KeyDown);

}

private void guna2Button1\_Click(object sender, EventArgs e)

{

DAL dal = new DAL();

string connectionString = DatabaseConnection.Connection();

if (ValidateInput())

{

using (SqlConnection conn = new SqlConnection(connectionString))

{

Response response = dal.Getaccount(conn);

if (response.list3.Any(account => account.username == textusername.Text && account.password == textpassword.Text) ||

(textusername.Text == "admin" && textpassword.Text == "admin"))

{

this.DialogResult = DialogResult.OK;

errormess.Visible = false;

Dashboard ds = new Dashboard();

this.Hide();

ds.Show();

}

else

{

errormess.Visible = true;

textpassword.Clear();

}

}

}

}

private bool ValidateInput()

{

if (textusername.Text == "" || textpassword.Text == "")

{

MessageBox.Show("Please fill in all the fields.");

return false;

}

return true;

}

private void TextBox\_KeyDown(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.Enter)

{

guna2Button1\_Click(sender, e);

e.SuppressKeyPress = true;

}

}

* Cài đặt Button Destroy
* Cài đặt Event Click, nếu người dùng nhấn vào biểu tượng thì phần mềm lập tức đóng.

private void guna2ImageButton1\_Click(object sender, EventArgs e)

{

Application.Exit();

}

## Giao diện trang chủ

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Hình 4.2. Giao diện trang chủ

* Trang chủ sẽ hiển thị toàn bộ thông tin.
* Trang chủ có thanh công cụ ở trên có thể chọn qua từng phần mà người dùng muốn, khi chọn thì giao diện sẽ đổi theo tương ứng với mục người dùng chọn.
* Phần giao diện Single room, Double room, TripeRoom (phòng một giường, phòng hai giường, phòng ba giường):
* Nếu phòng còn trống thì sẽ có giao diện là Free room và có màu xanh lá, số phòng, trạng thái phòng, tình trạng căn phòng, thể loại phòng.
* Còn phòng đã có khách hàng thì sẽ hiển thị màu khác và có tên khách hàng trên giao diện và ngày sử dụng, số phòng trạng thái phòng, tình trạng căn phòng, thể loại phòng.
* Phần giao diện Check Out (phòng thanh toán và rời đi):
* Có số tổng ngày sẽ thanh toán và rời đi. Tính từ ngày vào.
* Có số phòng, tên khách hàng, số căn cước công dân, ngày và giờ vào, ngày và giờ đi.
* Có phần Today Details (chi tiết hôm nay) sẽ cho chúng ta biết:
* Đang có bao nhiều phòng trống.
* Đang có bao nhiêu khách hàng,
* Đang có bao nhiêu nhân viên.
* Đã có bao nhiêu phòng check out (thanh toán).
* Phần Status (trạng thái phòng) sẽ cho chúng ta biết:
* Đang có bao nhiêu phòng Reserved (phòng đã được đặt trước).
* Đang có bao nhiêu phòng Oppcupied (phòng đang có khách lưu trú).
* Đang có bao nhiêu phòng Available (phòng đang còn trống).
* Đã có bao nhiêu phòng Checked out (phòng khách đã thoanh toán và rời đi).
* Phần Type (thể loại phòng) sẽ cho chúng ta biết:
* Đang có bao nhiêu phòng Single (phòng có một giường).
* Đang có bao nhiều phòng Double (phòng có hai giường).
* Đang có bao nhiêu phòng Triple (phòng có ba giường).
* Phần House Keeping (tình trạng căn phòng) sẽ cho chúng ta biết:
* Đang có bao nhiêu phòng Clean (phòng đã dọn dẹp).
* Đang có bao nhiêu phòng Not Clean (phòng chưa dọn dẹp).
* Đang có bao nhiêu phòng In Progress (phòng đang được dọn dẹp).
* Đang co bao nhiêu phòng Repair (phòng đang được sửa chữa).
* Nguyên lý hoạt động:
* B1:
* Tạo giao diện khung và các thành phần không thay đổi.
* B2:
* Các thành phần thay đổi sẽ được cài đặt riêng bởi các hàm trong chương trình.
* Các bảng dữ liệu chính
* Các bảng hiện thị đầy đủ các thông tin theo từng loại phòng và được chia nhỏ theo từng Panel Box. Giúp người dùng dễ dàng quản lý.

A close-up of a card

Description automatically generated

* Tùy theo từng loại phòng, Check Out
* Ta sẽ có những câu ràng buộc SQL khác nhau chuyên dụng để xử lý chương trình. Sự thay đổi dễ thấy như keyword: numbed (1, 2, 3).
* Cột thể hiện Status, Type, House\_keeping:
* Các ràng buộc được thiết lập đảm bảo chương trình sẽ trả về dữ liệu theo yêu cầu của người dùng.
* Khi người dùng Checked vào các CheckBox, ta sẽ thêm những ràng buộc mới vào ràng buộc ban đầu.Ví dụ như Check Box Occupied:

private void cboccupied\_CheckedChanged(object sender, EventArgs e)

{

if (cboccupied.Checked && cbclean.Checked)

{

uc\_dashboard.LoadSingleRoom("Occupied", "Clean");

uc\_dashboard.LoadDoubleRoom("Occupied", "Clean");

uc\_dashboard.LoadTripleRoom("Occupied", "Clean");

}

else if (cboccupied.Checked && cbnotclean.Checked)

{

uc\_dashboard.LoadSingleRoom("Occupied", "Not Clean");

uc\_dashboard.LoadDoubleRoom("Occupied", "Not Clean");

uc\_dashboard.LoadTripleRoom("Occupied", "Not Clean");

}

else if (cboccupied.Checked && cbinprogress.Checked)

{

uc\_dashboard.LoadSingleRoom("Occupied", "In Progress");

uc\_dashboard.LoadDoubleRoom("Occupied", "In Progress");

uc\_dashboard.LoadTripleRoom("Occupied", "In Progress");

}

else if (cboccupied.Checked && cbrepair.Checked)

{

uc\_dashboard.LoadSingleRoom("Occupied", "Repair");

uc\_dashboard.LoadDoubleRoom("Occupied", "Repair");

uc\_dashboard.LoadTripleRoom("Occupied", "Repair");

}

else if(cboccupied.Checked)

{

uc\_dashboard.LoadSingleRoom("Occupied");

uc\_dashboard.LoadDoubleRoom("Occupied");

uc\_dashboard.LoadTripleRoom("Occupied");

}

else

{

uc\_dashboard.LoadSingleRoom();

uc\_dashboard.LoadDoubleRoom();

uc\_dashboard.LoadTripleRoom();

}

}

* Sẽ có các trường hợp người dùng check vào chỉ 1 CheckBox Occupied hay người dùng check vào 2 trường khác nhau như Occupied và Clean. Khi đó câu lệnh SQL sẽ thêm ràng buộc ứng với hành động của người dùng.

string query0 = @" SELECT COUNT(\*) FROM Room INNER JOIN Update\_room ON Room.maphong = Update\_room.maphong WHERE numbed = 1";

string query1 = @" SELECT roomnumber FROM Room INNER JOIN Update\_room ON Room.maphong = Update\_room.maphong WHERE numbed = 1";

string query2 = @" SELECT status\_room

FROM Update\_room

INNER JOIN Room ON Update\_room.maphong = Room.maphong

WHERE Room.numbed = 1";

string query3 = @" SELECT first\_name + ' ' + last\_name as fullname

FROM Customer

INNER JOIN Bookings ON Customer.cccd\_cus = Bookings.cccd\_cus

INNER JOIN Update\_room ON Bookings.maphong = Update\_room.maphong

WHERE numbed = 1";

string query4 = @" SELECT DATEDIFF(DAY, date\_ci, date\_co) AS demngay

FROM Bookings

INNER JOIN Update\_room ON Bookings.maphong = Update\_room.maphong

WHERE numbed = 1";

string query5 = @" SELECT house\_keeping

FROM Update\_room

INNER JOIN Room ON Update\_room.maphong = Room.maphong

WHERE numbed = 1";

if (!string.IsNullOrEmpty(statusCon1) && !string.IsNullOrEmpty(statusCon2))

{

query0 += $" AND Update\_room.status\_room = '{statusCon1}' AND Update\_room.house\_keeping = '{statusCon2}'";

query1 += $" AND Update\_room.status\_room = '{statusCon1}' AND Update\_room.house\_keeping = '{statusCon2}'";

query2 += $" AND Update\_room.status\_room = '{statusCon1}' AND Update\_room.house\_keeping = '{statusCon2}'";

query3 += $" AND Update\_room.status\_room = '{statusCon1}' AND Update\_room.house\_keeping = '{statusCon2}'";

query4 += $" AND Update\_room.status\_room = '{statusCon1}' AND Update\_room.house\_keeping = '{statusCon2}'";

query5 += $" AND Update\_room.status\_room = '{statusCon1}' AND Update\_room.house\_keeping = '{statusCon2}'";

}

else if (!string.IsNullOrEmpty(statusCon1) && string.IsNullOrEmpty(statusCon2))

{

if (statusCon1 == "Reserved" || statusCon1 == "Occupied" || statusCon1 == "Available" || statusCon1 == "Check Out")

{

query0 += $" AND Update\_room.status\_room = '{statusCon1}'";

query1 += $" AND Update\_room.status\_room = '{statusCon1}'";

query2 += $" AND Update\_room.status\_room = '{statusCon1}'";

query3 += $" AND Update\_room.status\_room = '{statusCon1}'";

query4 += $" AND Update\_room.status\_room = '{statusCon1}'";

query5 += $" AND Update\_room.status\_room = '{statusCon1}'";

}

else if (statusCon1 == "Clean" || statusCon1 == "Not Clean" || statusCon1 == "In Progress" || statusCon1 == "Repair")

{

query0 += $" AND Update\_room.house\_keeping = '{statusCon1}'";

query1 += $" AND Update\_room.house\_keeping = '{statusCon1}'";

query2 += $" AND Update\_room.house\_keeping = '{statusCon1}'";

query3 += $" AND Update\_room.house\_keeping = '{statusCon1}'";

query4 += $" AND Update\_room.house\_keeping = '{statusCon1}'";

query5 += $" AND Update\_room.house\_keeping = '{statusCon1}'";

}

}

else

{

query0 = @" SELECT COUNT(\*) FROM Room INNER JOIN Update\_room ON Room.maphong = Update\_room.maphong WHERE numbed = 1";

}

## Giao diện thêm phòng (Add Room)

A screenshot of a computer

Description automatically generated

Hình 4.3. Giao diện thêm phòng

* Có chức năng tạo ra một phòng mới, khi điền vào đầy đủ thông tin và chọn hình phòng sau đó nhấn nút Submit để thêm được phòng mới. Nếu điền thiếu thông tin thì sẽ hiển thị một thông báo cho người dùng biết là cần điền đầy đủ thông tin thì mới có thể thêm phòng mới.
* Chức năng Update Room (cập nhật phòng) chỉ cần nhập đủ thông tin phòng cần sửa xong rồi nhấn nút Submit thì phòng đó sẽ được cập nhật. Nếu không nhập đủ thông tin, sẽ thông báo cho người dùng biết cần nhập đủ thông tin thì mới cập nhật phòng được.
* Chức năng Delete Room (xóa phòng) chỉ cần nhập Room ID (ID phòng) và nhấn nút Submit là chúng ta có thể xóa được phòng cần xóa.
* Có 2 bảng thông tin phòng và bảng cập nhật thông tin phòng cho người dùng biết được thông tin và trạng thái, tình trạng của các phòng như thế nào.
* Cách thức hoạt động:
* B1:
* Tạo PROC sp\_addroom với phương thức truyền vào là maphong, roomnumber, roomtype, numbed, view\_room, image\_room, price và errormessage. Nếu tồn tại phòng có maphong so với TABLE Room. Thì trả về @ErrorMessage : ‘Them phong khong thanh cong’. Ngược lại nếu không tồn tại thực hiện thêm vào bảng Room 1 dòng dữ liệu mới với các tham số được truyền vào. PROC thực hiện thành công thì @ErrorMessage: ‘Successfully’.

GO

CREATE OR ALTER PROC sp\_addroom @maphong INT, @roomnumber INT, @roomtype NVARCHAR(200), @numbed INT, @view\_room NVARCHAR(200), @image\_room NVARCHAR(200), @price INT, @ErrorMessage NVARCHAR(200) OUTPUT

AS

BEGIN

IF NOT EXISTS ( SELECT 1 FROM Room

WHERE maphong = @maphong)

BEGIN

INSERT INTO Room(maphong, roomnumber, roomtype, numbed, view\_room, image\_room, price) VALUES (@maphong, @roomnumber, @roomtype, @numbed, @view\_room, @image\_room, @price)

SET @ErrorMessage = 'Successfully'

END

ELSE

BEGIN

SET @ErrorMessage = 'Them phong khong thanh cong'

END

END

* B2:
* Tạo Model Room.cs như sau:

public class Room

{

public int maphong { get; set; }

public int roomnumber { get; set; }

public string roomtype { get; set; }

public string numbed { get; set; }

public string view\_room { get; set; }

public string image\_room { get; set; }

public int price { get; set; }

}

public class RoomUpdate

{

public int maphong { get; set; }

public int roomnumber { get; set; }

public string status\_room { get; set; }

public string house\_keeping { get; set; }

}

public class DetailRoom : Room

{

public string status\_room { get; set; }

public string house\_keeping { get; set; }

}

* B3:
* Tạo class Response.cs như sau:

public class Response

{

public List<Room> list { get; set; }

}

* B4:
* Tạo class DAL.cs như sau:

public Response Addroom(Room room, SqlConnection conn)

{

Response response = new Response();

try

{

string imageroom = Path.GetFileName(room.image\_room);

SqlCommand cmd = new SqlCommand("sp\_addroom", conn);

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@maphong", room.maphong);

cmd.Parameters.AddWithValue("@roomnumber", room.roomnumber);

cmd.Parameters.AddWithValue("@roomtype", room.roomtype);

cmd.Parameters.AddWithValue("@numbed", room.numbed);

cmd.Parameters.AddWithValue("@view\_room", room.view\_room);

cmd.Parameters.AddWithValue("@image\_room", imageroom);

cmd.Parameters.AddWithValue("@price", room.price);

cmd.Parameters.Add("@ErrorMessage", SqlDbType.Char, 200);

cmd.Parameters["@ErrorMessage"].Direction = ParameterDirection.Output;

conn.Open();

int i = cmd.ExecuteNonQuery();

conn.Close();

string mess = (string)cmd.Parameters["@ErrorMessage"].Value;

response.statusmessage = mess;

}

catch (Exception ex)

{

response.statusmessage = ex.Message;

}

return response;

}

* Với package SQL, ta sử dụng phướng thức SqlConnection và SqlCommand để thực hiện truy vấn tới cơ sở dữ liệu qua đường dẫn sau:

public class DatabaseConnection

{

public static string Connection()

{

//return "Server=BIUUUBIUUU\\MSSQLSERVER02;Initial Catalog=Hotel\_Management;User ID=sa;Password=1201;TrustServerCertificate=True;";

return "Server=ZAN\\ZAN;Initial Catalog=Hotel\_Management1;User ID=sa;Password=123;TrustServerCertificate=True;";

}

}

* Ta sẽ lấy từng tham số gán cho từng biến trong PROC

string imageroom = Path.GetFileName(room.image\_room);

SqlCommand cmd = new SqlCommand("sp\_addroom", conn);

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@maphong", room.maphong);

cmd.Parameters.AddWithValue("@roomnumber", room.roomnumber);

cmd.Parameters.AddWithValue("@roomtype", room.roomtype);

cmd.Parameters.AddWithValue("@numbed", room.numbed);

cmd.Parameters.AddWithValue("@view\_room", room.view\_room);

cmd.Parameters.AddWithValue("@image\_room", imageroom);

cmd.Parameters.AddWithValue("@price", room.price);

* B5:
* Tạo giao diện có các fields đủ và tướng ứng. Cài đặt các name Design để thực hiện cho chương trình.
* B6
* Cài đặt Event Click cho button Submit như sau:

private void guna2Button1\_Click(object sender, EventArgs e)

{

if (ValidateInput())

{

Room room = new Room

{

maphong = int.Parse(txtmaphong.Text),

roomnumber = int.Parse(txtsophong.Text),

roomtype = txtloaiphong.SelectedItem.ToString(),

numbed = txtloaigiuong.SelectedItem.ToString(),

view\_room = txtviewroom.SelectedItem.ToString(),

image\_room = Image1.ImageLocation,

price = int.Parse(txtgia.Text)

};

DAL dal = new DAL();

string connectionString = DatabaseConnection.Connection();

using (SqlConnection conn = new SqlConnection(connectionString))

{

Response response = dal.Addroom(room, conn);

MessageBox.Show(response.statusmessage);

if (response.statusmessage.Contains("Successfully"))

{

RefreshControl();

}

}

}

}

* Thêm các ràng buộc vào các trường fields như sau:

private bool ValidateInput()

{

if (txtmaphong.Text == "" || txtsophong.Text == "" || txtloaiphong.SelectedItem == null || txtloaigiuong.SelectedItem == null || txtviewroom.SelectedItem == null || txtgia.Text == "")

{

MessageBox.Show("Please fill in all the fields.");

return false;

}

return true;

}

* Thực hiện khởi tạo room và truyền vào các biến tương ứng là các tham số nhận được từ TextBox với các name Design đã cài đặt trước đó .

Khởi tạo DAL, sử dụng package SQL để thực hiện thêm một tài khoản.

Nếu chương trình trả về thông báo Successfully từ PROC. Thực hiện cập nhật lại DataGridView cũng như làm mới lại TextBox trong Room đã nhập trước đó. Sau đó Rename lại các trường field của DataGridView. Nếu không có dữ liệu hoặc lỗi trong quá trình truy vấn chương trình sẽ hiện MessageBox để thông báo.

private void RefreshControl()

{

clearAll();

LoadRoomData();

LoadRoomUpData();

}

public void clearAll()

{

txtmaphong.Clear();

txtsophong.Clear();

txtloaiphong.SelectedIndex = -1;

txtloaigiuong.SelectedIndex = -1;

txtviewroom.SelectedIndex = -1;

txtgia.Clear();

txtmaphongupdateroom.Clear();

txtstatusroom.SelectedIndex = -1;

txthousekeeping.SelectedIndex = -1;

txtdeletemaphong.Clear();

Image1.Image = null;

}

* Tương tự ta thực hiện với Room Update với các thuộc tính: House\_keeping, Status\_room.
* Khi thực hiện chức năng xóa
* Ta sẽ nhập Room ID vào textBox.
* Cài đặt chức năng Event Click cho button xóa.
* Chương trình sẽ thực hiện kiểm tra, nếu có phòng với ID tương tự thì bắt đầu thực hiện xóa phòng và thông báo ‘Successfully’ và Refesh lại dữ liệu trong các DataGridView.
* Nếu không tồn tại Room ID tương tự chương trình sẽ thực hiện thông báo ‘Xoa phong khong thanh cong’.
* PROC sp\_deleteroom:

GO

CREATE OR ALTER PROC sp\_deleteroom @maphong INT, @ErrorMessage NVARCHAR(200) OUTPUT

AS

BEGIN

IF EXISTS ( SELECT 1 FROM Room

WHERE maphong = @maphong)

BEGIN

DELETE FROM Room WHERE maphong = @maphong

SET @ErrorMessage = 'Successfully'

END

ELSE

BEGIN

SET @ErrorMessage = 'Xoa Phong khong thanh cong'

END

END

* Button Event Click:

private void guna2Button2\_Click(object sender, EventArgs e)

{

if (txtdeletemaphong.Text != "")

{

Room room = new Room

{

maphong = int.Parse(txtdeletemaphong.Text),

};

DAL dal = new DAL();

string connectionString = DatabaseConnection.Connection();

using (SqlConnection conn = new SqlConnection(connectionString))

{

Response response = dal.Deleteroom(room, conn);

MessageBox.Show(response.statusmessage);

if (response.statusmessage.Contains("Successfully"))

{

RefreshControl();

}

}

}

else

{

MessageBox.Show("Please fill in all the fields.");

}

}

* Class DAL.cs:

public Response Deleteroom(Room room, SqlConnection conn)

{

Response response = new Response();

try

{

SqlCommand cmd = new SqlCommand("sp\_deleteroom", conn);

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@maphong", room.maphong);

cmd.Parameters.Add("@ErrorMessage", SqlDbType.Char, 200);

cmd.Parameters["@ErrorMessage"].Direction = ParameterDirection.Output;

conn.Open();

int i = cmd.ExecuteNonQuery();

conn.Close();

string mess = (string)cmd.Parameters["@ErrorMessage"].Value;

response.statusmessage = mess;

}

catch (Exception ex)

{

response.statusmessage = ex.Message;

}

return response;

}

* Sử dụng package SQL để truy vấn cơ sở dữ liệu. Nhằm lấy dữ liệu dạng bảng cho DataGridView.
* Ta sử dụng các chức năng như SqlConnection, SqlCommand, SqlDataReader thực hiện truy vấn và khởi tạo thuộc tính nhằm hứng dữ liệu thu được từ câu lệnh SQL. Sau đó thêm vào mảng List.

Public Response Getroom(SqlConnection conn)

{

Response response = new Response();

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

try

{

string query= @”SELECT \* FROM Room”;

using (SqlCommand cmd = new SqlCommand(query, conn))

{

conn.Open();

using (SqlDataReader reader = cmd.ExecuteReader())

{

while (reader.Read())

{

Room room = new Room

{

maphong = int.Parse(reader[“maphong”].ToString()),

roomnumber = int.Parse(reader[“roomnumber”].ToString()),

roomtype = reader[“roomtype”].ToString(),

numbed = reader[“numbed”].ToString(),

view\_room = reader[“view\_room”].ToString(),

image\_room = reader[“image\_room”].ToString(),

price = int.Parse(reader[“price”].ToString()),

};

list.Add(room);

}

}

}

response.list = list;

}

catch(Exception ex)

{

response.statusmessage = ex.Message;

}

finally

{

if(conn.State == ConnectionState.Open)

conn.Close();

}

return response;

}

* Chức năng Xuất dữ liệu thông qua DataGridView:

private void LoadRoomData()

{

DAL dal = new DAL();

string connectionString = DatabaseConnection.Connection();

using (SqlConnection conn = new SqlConnection(connectionString))

{

Response response = dal.Getroom(conn);

if ((response.list != null && response.list.Count > 0))

{

DataGridView1.DataSource = null; // Clear previous data

DataGridView1.DataSource = response.list;

DataGridView1.ColumnHeadersHeight = 25;

DataGridView1.Columns[“maphong”].HeaderText = “ID Phòng”;

DataGridView1.Columns[“roomnumber”].HeaderText = “Mã Phòng”;

DataGridView1.Columns[“roomtype”].HeaderText = “Thể loại phòng”;

DataGridView1.Columns[“numbed”].HeaderText = “Số giường”;

DataGridView1.Columns[“view\_room”].HeaderText = “Dạng phòng”;

DataGridView1.Columns[“image\_room”].HeaderText = “Image room”;

DataGridView1.Columns[“price”].HeaderText = “Giá cả";

DataGridView1.AutoSizeColumnsMode = DataGridViewAutoSizeColumnsMode.None;

DataGridView1.AutoSizeRowsMode = DataGridViewAutoSizeRowsMode.AllCells;

DataGridView1.ScrollBars = System.Windows.Forms.ScrollBars.Both;

DataGridView1.Refresh(); // Refresh the grid view

}

else

{

MessageBox.Show("No data available or " + response.statusmessage);

}

}

}

## Giao diện thêm thông tin khách hàng (Customer)

Screens screenshot of a computer

Description automatically generated

Hình 4.4.1 Giao diện diện thêm thông tin khách hàng

* Chức năng thêm thông tin khách hàng cần nhập đủ thông tin và nhấn nút Submit để thêm thông tin khách hàng mới. Nếu không nhập đủ thông tin, sẽ thông báo cho người dùng biết cần nhập đủ thông tin thì mới thêm khách hàng mới vào được.
* Chức năng cập nhật thông tin khách hàng thì cần phải nhập đúng CCCD của khách hàng thì mới có thể sửa thông tin của khách hàng đó được. Nếu nhập sai CCCD hoặc thiếu thông tin khác thì không cập nhật dược thông tin của khách hàng.
* Có bảng hiển thị thông tin khách hàng cho người dùng biết được thông tin của khách hàng đang có, và cũng có thể cho biết được thông tin khách hàng đã được thêm mới hoặc cập nhật thành công hay chưa.
* Chức năng Export to CSV cho bảng hiển thị thông tin khách hàng để tạo thuận lợi cho quá trình xử lý và quản lý dữ liệu lương một cách linh hoạt và hiệu quả hơn. Cụ thể chức năng:

+ Sử dụng các chức năng truy vấn SQL để lấy dữ liệu từ cơ sở dữ liệu. Sau đó tạo đường dẫn chứa file CSV.  
+ Ứng với từng trường được lấy từ dữ liệu sẽ được lần lượt ghi vào file CSV và cách nhau bởi dấu phẩy nhờ StreamWriter.

+ Mỗi lần người dùng click button export sẽ tạo ra các file mới khác nhau. Tránh trường hợp bị ghi đè, từ đó có sự nhầm lẫn trong quá trình đọc file.

int count = 0;

private void guna2Button3\_Click(object sender, EventArgs e)

{

DAL dal = new DAL();

string connectionString = DatabaseConnection.Connection();

using (SqlConnection conn = new SqlConnection(connectionString))

{

Response response = dal.Getcustomer(conn);

List<Customer> customers = response.list1;

try

{

string filePath = $@"E:\App.NET\WindowsForms\_Project\Project\_WinForm\WindowsForm\_Project\File CSV\Customer\Customer\_{count++}.csv";

using (StreamWriter sw = new StreamWriter(filePath, true))

{

sw.WriteLine("CCCD\_Customer, First\_Name, Last\_Name, SDT, Email, GioiTinh, NgaySinh, Address");

foreach (var customer in customers)

{

sw.WriteLine($"{customer.cccd\_cus}, {customer.first\_name}, {customer.last\_name}, {customer.sdt}, {customer.email}, {customer.gioitinh}, {customer.ngaysinh}, {customer.address\_cus}");

}

}

MessageBox.Show("File exported successfully!");

}

catch (Exception ex)

{

Console.WriteLine($"Error: {ex.Message}");

}

finally

{

Console.WriteLine("Completed file writing.");

}

}

}

* Các cách thức hoạt động được kế thừa từ Giao diện thêm phòng.

## Giao diện đặt phòng (Bookings)

A screenshot of a computer

Description automatically generated

Hình 4.5.1 Giao diện đặt phòng

A screenshot of a computer

Description automatically generated

Hình 4.5.2 Giao diện đặt phòng

* Chức năng đặt phòng người dùng cần chọn phần Status Room thì hệ thống sẽ tự động đề xuất thông tin theo từng mục mà người dùng chọn như:
* House Keeping
* Room Type
* Num Bed
* View Room
* ID Room
* Room NumberS
* Các mục trên người dùng có thể dựa theo bảng thông tin phòng còn trống để dùng. Sau đó chỉ cần nhập:
  + CCCD Customer (CCCD khách hàng).
  + Group Customer (số người đi cùng).
  + CCCD Employee (CCCD nhân viên).
  + Date Check In (ngày vào).
  + Date Check Out (ngày ra).
* Sau khi nhập xong những thông tin trên thì phần Price sẽ tự động tính tổng số tiền cần phải thanh toán dựa trên ngày vào và ngày ra của khách hàng. Nhấn nút Submit sẽ đặt phòng thành công. Nếu nhập không đủ thông tin thì hệ thống sẽ thông báo để cần nhập đủ thông tin mới có thể đặt phòng được.
* Khi thực hiện đặt phòng thành công thì thông tin đó sẽ được lưu trong bảng thông tin đặt phòng.
* Bảng thông tin khách hàng chưa đặt phòng cung cấp thông tin khách hàng cho người dùng để nhập vào cho chính xác CCCD Customer ( CCCD khách hàng ).
* Bảng thông tin đặt phòng cho người dùng biết thông tin, tình trạng, trạng thái của phòng để người dùng biết.
* Chức năng Export to CSV cho bảng thông tin đặt phòng để tạo thuận lợi cho quá trình xử lý và quản lý dữ liệu lương một cách linh hoạt và hiệu quả hơn.
* Các chức năng trong Giao diện được kế thừa từ Giao diện Thêm phòng.

## Giao diện đặt dịch vụ (Service)

A screenshot of a computer

Description automatically generated

Hình 4.6.1 Giao diện dịch vụ

A screenshot of a computer

Description automatically generated

Hình 4.6.2 Giao diện dịch vụ

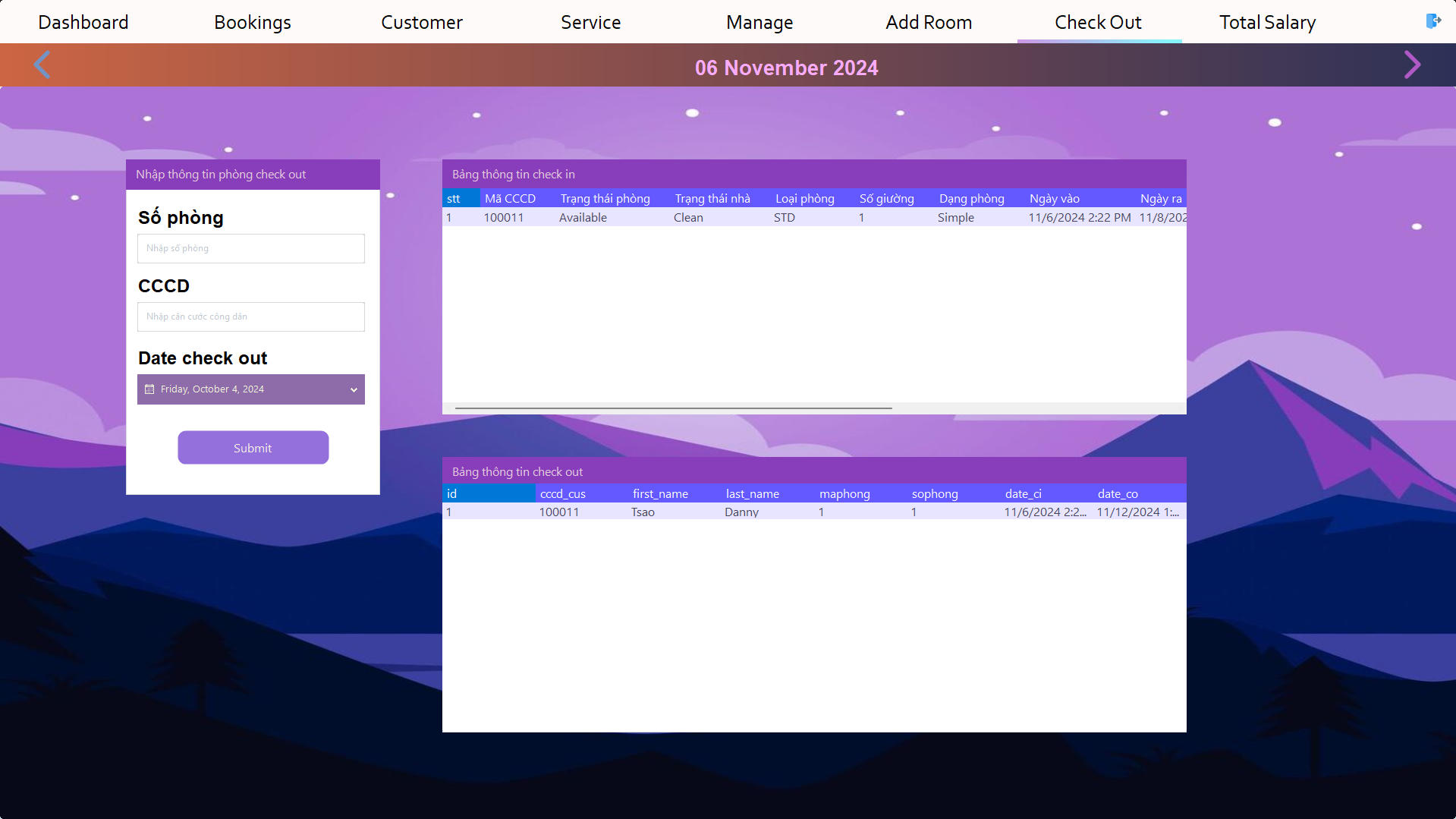
A screenshot of a computer

Description automatically generated

Hình 4.6.3 Giao diện dịch vụ

* Chức năng gọi dịch vụ khi người dùng nhập CCCD của khách hàng vào thì hệ thống sẽ tự động đề xuất RoomID của khách hàng đó. Sau đó nhập thông tin dịch vụ mà khách hàng yêu cầu, rồi tick vào nút Call Serve (gọi phục vụ), nhập giá tiền của dịch vụ mà khách hàng yêu cầu vào, rồi nhấn nút Submit là đặt thành công dịch vụ. Nếu người dùng nhập không đủ thông tin thì hệ thống sẽ thông báo cần nhập đủ thông tin thì mới gọi được dịch vụ được.
* Khi thực hiện xong thì những dịch vụ của khách hàng sẽ được lưu trong bảng dịch vụ.
* Chức năng ghi chú khi nhấn vào nút ghi chú nằm bên phải màn hình. Sẽ xuất hiện giao diện của phần ghi chú. Người dùng cần phải nhập đủ thông tin của khách hàng muốn phản hồi và nhấn nút Submit là ghi chú thành công, nếu người dùng nhập không đủ thông tin thì hệ thống sẽ thông báo cần nhập đủ thông tin thì mới ghi chú mới thành công. Chức năng này giúp lưu lại những phản hồi của khách hàng về khách sạn.
* Khi nhập ghi chú thành công thì ghi chú sẽ được hiển thị trong bảng thông tin ghi chú.
* Có 3 bảng cung cấp thông tin cho người dùng. Người dùng có thể dựa vào bảng thông tin phòng đã đặt để nhập thông tin để chính xác hơn.
* Các chức năng trong Giao diện được kế thừa từ Giao diện Thêm phòng.

## Giao diện thanh toán và trả phòng (Check out)



Hình 4.7.1 Giao diện thanh toán và trả phòng

* Chức năng này giúp người dùng thanh toán và trả phòng cho khách hàng nhanh chóng. Người dùng chỉ cần nhập số phòng và CCCD, ngày check out của khách hàng. Khi nhập xong thì nhấn Submit thì thông tin sẽ được hiển thị ở bảng thông tin check out. Nếu không nhập đủ thì hệ thống sẽ thông báo yêu cầu nhập đủ thông tin thì mới thực hiện được.
* Có 2 bảng giúp người dùng có thể xem được phòng nào đang check in và check out.
* Các chức năng trong Giao diện được kế thừa từ Giao diện Thêm phòng.

## Giao diện của quản lý (Manage)

Screens screenshot of a computer

Description automatically generated

Hình 4.8.1 Giao diện quản lý

A screenshot of a computer

Description automatically generated

Hình 4.8.2 Giao diện quản lý

A screenshot of a computer

Description automatically generated

Hình 4.8.3 Giao diện quản lý

* Để vào được giao diện quản lý thì người dùng cần nhấp vào Manage trên thanh công cụ rồi sẽ xuất hiện một thông báo ("Bạn có muốn tiếp tục đăng nhập dưới quyền quản lý không"). Sau đó khi nhấn vào OK thì sẽ xuất hiện màn hình đăng nhập thì người dùng cần phải đăng nhập tài khoản của quản lý vào để thực hiện các chức năng có trong giao diện quản lý.
* Chức năng nhập thông tin nhân viên giúp cho người dùng nhập thông tin một nhân viên mới nhanh chóng. Người dùng cần nhập đầy đủ thông tin để và nhấn nút Submit thì sẽ thêm nhân viên thành công. Nếu người dùng nhập không đủ thông tin thì hệ thống sẽ thông báo yêu cầu nhập đủ thông tin thì chức năng mới thực hiện được.
* Chức năng cập nhật thông tin nhân viên giúp người dùng cập nhật thông tin nhân viên nhanh chóng. Người dùng cần nhập đúng CCCD và đủ thông tin của nhân viên cần sửa, nhấn nút Submit thì thông tin sẽ được cập nhật. Nếu nhập sai CCCD hoặc thiếu thông tin thì hệ thống sẽ thông báo yêu cầu nhập đủ thông tin thì chức năng mới thực hiện được.
* Khi thực hiện xong một trong hai chức năng trên thì bảng thông tin nhân viên sẽ thay đổi theo tùy chức năng.
* Chức năng nhập ca làm nhân viên giúp người dùng nhập ca làm cho nhân viên nhanh chóng. Người dùng sẽ phải nhập CCCD đúng và chọn ngày làm, các ca làm (từ ca làm từ 1 đến 4) của nhân viên và có thể ghi chú lại nhân viên hôm nay làm như thế nào. Khi nhập xong nhấn vào nút Submit thì thông tin sẽ được thêm thành công. Nếu nhập sai CCCD hoặc thiếu thông tin thì hệ thống sẽ thông báo yêu cầu nhập đủ thông tin thì chức năng mới thực hiện được.
* Khi nhập ca làm xong thì sẽ tính tổng tiền lương của từng ca làm của nhân viên dựa vào thông tin của nhân viên sẽ được hiển thị trong bảng tổng lương và bảng kiểm tra ca làm nhân viên.
* Bên phải màn hình có nút Account khi nhấn vào thì sẽ có giao diện của thêm tài khoản cho nhân viên.
* Chức năng thêm tài khoản cho nhân viên cho phép người dùng tạo tài khoản mới để nhân viên có thể thực hiện các chức năng cần thiết. Để hoàn tất, cần nhập đúng và đầy đủ thông tin và nhấn vào nút Submit. Nếu thông tin bị thiếu hoặc không chính xác, hệ thống sẽ yêu cầu bổ sung trước khi tiếp tục. Các chức năng của nhân viên như:
* Add Room (Thêm phòng mới).
* Customer (Thêm thông tin khách hàng mới).
* Bookings (Đặt phòng).
* Service (Gọi dịch vụ).
* Check Out (Thanh toán và trả phòng).
* Chức năng xóa tài khoản giúp người dùng xóa tài khoản của nhân viên nhanh chóng. Người dùng chỉ cần nhập ID Account cần xóa có trong bảng thông tin và nhấn nút Submit thì thông tin sẽ xóa thành công. Nếu không nhập đúng ID Account hoặc thiếu thì hệ thống sẽ thông báo yêu cầu nhập đủ thông tin thì chức năng mới thực hiện được.
* Các chức năng trong Giao diện được kế thừa từ Giao diện Thêm phòng.

## Giao diện thu nhập (Total salary)

A screenshot of a computer

Description automatically generated

Hình 4.9.1 Giao diện thu nhập

A screenshot of a computer

Description automatically generated

Hình 4.9.2 Giao diện thu nhập

* Chức năng nhập hàng hóa (Total Expenses) cho phép người dùng ghi nhận lại tên mặt hàng và giá tiền đã chi tiêu trong tháng. Khi sử dụng chức năng này người dùng cần nhập đúng và đủ thông tin và nhấn nút Submit thì hệ thống sẽ báo nhập thành công. Nếu nhập thiếu thông tin thì hệ thống sẽ thông báo yêu cầu nhập đủ thông tin thì chức năng mới thực hiện được.
* Nút Details sẽ hiển thị bảng lưu trữ dữ liệu đã hàng hóa đã mua trong tháng. Trong bảng sẽ có các thông tin như:
* Ngày mua.
* Tên của đồ gia dụng, nguyên liệu, nhu yếu phẩm.
* Giá tiền của đồ gia dụng, nguyên liệu, nhu yếu phẩm.
* Nút Details total là hiển thị bảng lưu trữ dữ liệu. Trong bảng sẽ có các thông tin như:
* Ngày và năm
* Tổng thu
* Tổng chi
* Lợi nhuận
* Salary export xuất dữ liệu trong bảng detail total ra thành file csv để tạo thuận lợi cho quá trình xử lý và quản lý dữ liệu lương một cách linh hoạt và hiệu quả hơn.
* Chart doughnut (biểu đồ rỗng hình tròn) thể hiện phần trăm của từng thành phần như:
  + Tổng Thu nhập: sẽ được tính toán theo: S1 = tổng giá phòng được thuê + tổng phí dịch vụ + (tổng số lượng nguyên liệu \* 70.000Đ).
  + Tổng Chi tiêu: sẽ được tính toán theo công thức: S2 = tổng giá nhập đồ gia dụng + tổng giá nhu yếu phẩm + tổng giá nhập nguyên liệu + tổng lương của từng nhân viên làm theo ca.
  + Tổng lương của nhân viên: Sẽ được tính theo tổng lương của từng nhân viên làm theo ca.
  + Tổng lợi nhuận: Sẽ được tính theo công thức S (tổng lợi nhuận) = S1(tổng thu nhập) - S2 (tổng chi tiêu).
* Biểu đồ Spline sẽ biểu hiện cho người dùng thấy được tổng thu nhập (Đường màu xanh) và tổng chi tiêu (Đường màu đỏ) theo từng tháng.
* Biểu đồ Employee salary thể hiện cột tổng lương theo ca của từng nhân viên, mỗi nhân viên sẽ được chú thích bằng một màu khác nhau.
* Tương ứng theo từng biểu đồ ta sẽ có các câu truy vấn SQL khác nhau nhằm tính tổng các mục. Sau đó tùy thuộc vào loại biểu đồ mà ta sẽ filter Tổng tính được theo từng tháng. Nhằm giúp cải thiện việc chi tiêu cũng như theo dõi được các nhu cầu, hoạt động, thu nhập của mô hình kinh doanh.

1. ĐỊNH HƯỚNG PHÁT TRIỂN

Để nâng cao chất lượng và hiệu quả của hệ thống quản lý khách sạn, chúng tôi đưa ra các định hướng phát triển sau đây:

1. **Tích Hợp với Các Nền Tảng Đặt Phòng Trực Tuyến**  
   Trong tương lai, phần mềm có thể được tích hợp với các nền tảng đặt phòng trực tuyến như Booking.com, Agoda, hoặc Expedia. Điều này sẽ giúp khách sạn dễ dàng cập nhật tình trạng phòng trống và giá phòng trên các nền tảng này, từ đó tăng cơ hội tiếp cận với nhiều khách hàng hơn.
2. **Bổ Sung Hỗ Trợ Thanh Toán Trực Tuyến**  
   Tích hợp các cổng thanh toán trực tuyến như thẻ tín dụng, PayPal, hoặc các ví điện tử sẽ giúp khách hàng thanh toán dễ dàng hơn. Điều này cũng đảm bảo quy trình thanh toán trở nên tiện lợi, nhanh chóng và bảo mật hơn cho cả khách hàng và nhân viên.
3. **Phát Triển Ứng Dụng Di Động**  
   Việc xây dựng một ứng dụng di động cho hệ thống quản lý khách sạn sẽ giúp nhân viên dễ dàng quản lý công việc trên điện thoại. Khách hàng cũng có thể sử dụng ứng dụng này để thực hiện các thao tác đặt phòng, yêu cầu dịch vụ hoặc thanh toán qua điện thoại.
4. **Hệ Thống Báo Cáo và Thống Kê Nâng Cao**  
   Cải thiện hệ thống báo cáo bằng cách thêm các biểu đồ thống kê chi tiết hơn về tình trạng phòng, doanh thu, hiệu suất nhân viên và các dịch vụ được sử dụng. Các báo cáo này có thể được hiển thị theo thời gian thực để ban quản lý dễ dàng theo dõi và ra quyết định nhanh chóng.
5. **Tối Ưu Giao Diện Người Dùng**  
   Nâng cấp giao diện người dùng để thân thiện hơn và dễ thao tác trên cả máy tính và thiết bị di động. Việc áp dụng thiết kế UI/UX tốt hơn sẽ giúp tăng cường trải nghiệm của người dùng, giảm thiểu sai sót khi sử dụng phần mềm và cải thiện tốc độ xử lý công việc. Cải thiện các chức năng và nâng cấp giao diện tránh xảy ra tình trạng lỗi.
6. **Áp Dụng Công Nghệ AI và Machine Learning**  
   Sử dụng AI để phân tích dữ liệu khách hàng, đưa ra các gợi ý phòng phù hợp hoặc dự đoán nhu cầu của khách hàng dựa trên hành vi trước đó. AI cũng có thể được dùng để tối ưu hóa quản lý nhân sự, dự đoán tình trạng phòng trống, và điều chỉnh giá phòng linh hoạt dựa trên nhu cầu.
7. **Tăng Cường An Ninh và Bảo Mật Hệ Thống**  
   Áp dụng các biện pháp bảo mật mạnh mẽ hơn, như mã hóa dữ liệu và xác thực đa yếu tố, để bảo vệ thông tin của khách hàng và nhân viên. Đồng thời, thiết lập các quy trình sao lưu và khôi phục dữ liệu để đảm bảo hệ thống hoạt động ổn định và an toàn.
8. **Hỗ Trợ Đa Ngôn Ngữ**  
   Hệ thống có thể mở rộng để hỗ trợ nhiều ngôn ngữ khác nhau, nhằm phục vụ khách hàng quốc tế và tạo thuận lợi cho nhân viên trong quá trình làm việc với nhiều khách hàng đến từ các quốc gia khác nhau.
9. KẾT LUẬN

Em xin gửi lời biết ơn sâu sắc đến thầy Nguyễn Minh Đế đã dành nhiều thời gian và tâm huyết hướng dẫn nghiên cứu và giúp em hoàn thành môn học.

Em đã có nhiều cố gắng hoàn thiện dự án bằng tất cả năng lực của mình, tuy nhiên không thể tránh khỏi nhiều thiếu sót, rất mong nhận được những đóng góp quý báu của quý thầy cô và các bạn.

1. TÀI LIỆU THAM KHẢO
2. Microsoft Documentation về C# và .NET Framework
3. Microsoft Documentation về SQL Server
4. Programming Microsoft Windows Forms
5. Website W3Schools: SQL Server
6. Website TutorialsPoint: C# Windows Forms
7. PHỤ LỤC
8. **Danh sách các Thực thể trong Cơ sở Dữ liệu:**
9. **Room**: Lưu thông tin phòng (mã phòng, số phòng, loại phòng, số giường, view, giá).
10. **Update\_room**: Lưu trạng thái phòng (mã phòng, tình trạng phòng, dọn dẹp).
11. **Customer**: Lưu thông tin khách hàng (CCCD, họ tên, số điện thoại, email, giới tính, ngày sinh, địa chỉ).
12. **Bookings**: Thông tin đặt phòng (CCCD khách, trạng thái phòng, housekeeping, loại phòng, số giường, view, ngày nhận, ngày trả, nhóm khách hàng, mã phòng, số phòng, giá).
13. **Employee**: Thông tin nhân viên (CCCD, họ tên, số điện thoại, email, giới tính, ngày sinh, lương).
14. **Report**: Báo cáo về khách và phòng (CCCD khách, mã phòng, ghi chú).
15. **Serve**: Thông tin dịch vụ (CCCD khách, mã phòng, đặt dịch vụ, ăn uống, yêu cầu phục vụ).
16. **Chamcong**: Thông tin chấm công (mã phiếu, CCCD nhân viên, ngày, các ca làm việc, ghi chú).
17. **Account**: Tài khoản nhân viên (ID, username, password, CCCD nhân viên).
18. **Checkout**: Thông tin trả phòng (ID, CCCD khách, họ tên khách, mã phòng, số phòng, ngày nhận, ngày trả).
19. **Chitieu**: Các khoản chi tiêu (STT, ngày, tên đồ gia dụng, giá nhập đồ gia dụng, nguyên liệu, giá nhập nguyên liệu, nhu yếu phẩm, giá nhập nhu yếu phẩm).
20. **Salary**: Thông tin lương và chi phí doanh nghiệp (STT, tháng, năm, tổng thu, tổng chi, lợi nhuận).

**B. Danh sách các giao diện chính:**

1. **Đăng nhập**: Kiểm tra thông tin tài khoản và quyền truy cập.
2. **Trang chủ**: Hiển thị tổng quan tình trạng phòng, khách hàng, và nhân viên.
3. **Thêm phòng**: Chức năng thêm, sửa, xóa thông tin phòng.
4. **Thêm khách hàng**: Nhập và cập nhật thông tin khách hàng.
5. **Đặt phòng**: Chọn trạng thái phòng và nhập thông tin đặt phòng.
6. **Dịch vụ**: Đặt dịch vụ phòng cho khách hàng.
7. **Thanh toán và trả phòng**: Xử lý thanh toán và trả phòng.
8. **Quản lý**: Nhập thông tin, cập nhật, và quản lý ca làm của nhân viên.
9. **Thu nhập**: Nhập các khoản chi, hiển thị biểu đồ chi phí, lương, lợi nhuận doanh nghiệp.

**C. Phương pháp tính toán:**

1. **Tổng thu nhập**: Tổng giá phòng + phí dịch vụ + (số lượng nguyên liệu \* 70,000).
2. **Tổng chi tiêu**: Giá nhập đồ gia dụng + nhu yếu phẩm + nguyên liệu + lương nhân viên.
3. **Tổng lợi nhuận**: Tổng thu nhập - Tổng chi tiêu.

**D. Các biểu đồ thống kê:**

1. **Biểu đồ Donut**: Tỉ lệ phần trăm các khoản thu nhập, chi tiêu, lương, và lợi nhuận.
2. **Biểu đồ Spline**: Tổng thu nhập (màu xanh) và chi tiêu (màu đỏ) theo tháng.
3. **Biểu đồ Column**: Lương của từng nhân viên theo ca.

**E. Phân quyền người dùng:**

1. **Quản trị viên**: Quản lý toàn bộ hệ thống và người dùng.
2. **Nhân viên lễ tân**: Đặt phòng, check-in, check-out, và quản lý dịch vụ.
3. **Nhân viên**: Cập nhật tình trạng vệ sinh và bảo trì phòng.

**F. Các trường hợp kiểm thử:**

1. **Kiểm thử đăng nhập**: Nhập sai username/password, và các tài khoản khác nhau.
2. **Kiểm thử thêm phòng**: Nhập đủ và thiếu thông tin phòng.
3. **Kiểm thử đặt phòng**: Đặt phòng trùng, nhập sai CCCD khách hàng.
4. **Kiểm thử dịch vụ**: Gọi dịch vụ với thông tin không đầy đủ.