|  |  |
| --- | --- |
| Đại học Quốc gia Hà Nội **Trường Đại học Việt Nhật** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập - Tự do - Hạnh phúc** |

**BÁO CÁO DỰ ÁN QUẢN LÝ KHÁCH SẠN**

Môn học: Lập trình hướng đối tượng

Nhóm sinh viên thực hiện:        Phạm Hồng Phúc

Phạm Việt Hưng

Quản Minh Tiến

Đặng Huy Hoàng

Giảng viên: TS. Bùi Huy Kiên

Lớp: BCSE-2023

Khóa: 2023-2027

Ngành: FATE

Hà Nội, 5/2024

**MỤC LỤC**

1.Giới thiệu về dự án và ứng dụng quản lý khách sạn

2. Các Thiết Kế của Dự Án

• Thiết Kế Giao Diện Người Dùng

• Các Kỹ Thuật Lập Trình Hướng Đối Tượng

3. Các kỹ thuật lập trình sử dụng trong dự án

4. Các Công Cụ Sử Dụng để Thiết Kế và Phát Triển Dự Án

5. Các Yêu Cầu Kiểm Thử và Đánh Giá Dự Án

• Yêu Cầu Kiểm Thử

• Yêu Cầu Đánh Giá

6. Đánh giá ưu nhược điểm của sản phẩm

7. Kế Hoạch Phát Triển Tiếp Theo của Dự Án

• Mở Rộng Chức Năng Ứng Dụng

• Cải Thiện Giao Diện Người Dùng

• Cập Nhật Giọng Đọc

• Bổ Sung Nội Dung Học Tập

• Cải Tiến Hệ Thống Lưu Trữ Thông Tin

**1. Giới thiệu về dự án và ứng dụng quản lý khách sạn**

Trong bối cảnh ngành du lịch và dịch vụ khách sạn ngày càng phát triển, việc nâng cao chất lượng quản lý và dịch vụ là một yêu cầu thiết yếu. Dự án Quản lý Khách sạn ra đời với mục tiêu cung cấp một giải pháp toàn diện và hiệu quả cho các doanh nghiệp kinh doanh trong lĩnh vực này.

*Mục tiêu chính của dự án:*

- Tối ưu hóa quy trình quản lý từ đặt phòng, check-in/check-out

- Cải thiện trải nghiệm khách hàng thông qua việc cá nhân hóa và đáp ứng nhanh chóng các yêu cầu.

- Tăng cường hiệu suất làm việc và giảm thiểu sai sót trong các hoạt động hàng ngày.

- Tối đa hóa doanh thu thông qua việc quản lý và phân tích dữ liệu kinh doanh.

- Đảm bảo tuân thủ các quy định pháp luật và bảo mật thông tin khách hàng.

*Lợi ích mang lại:*

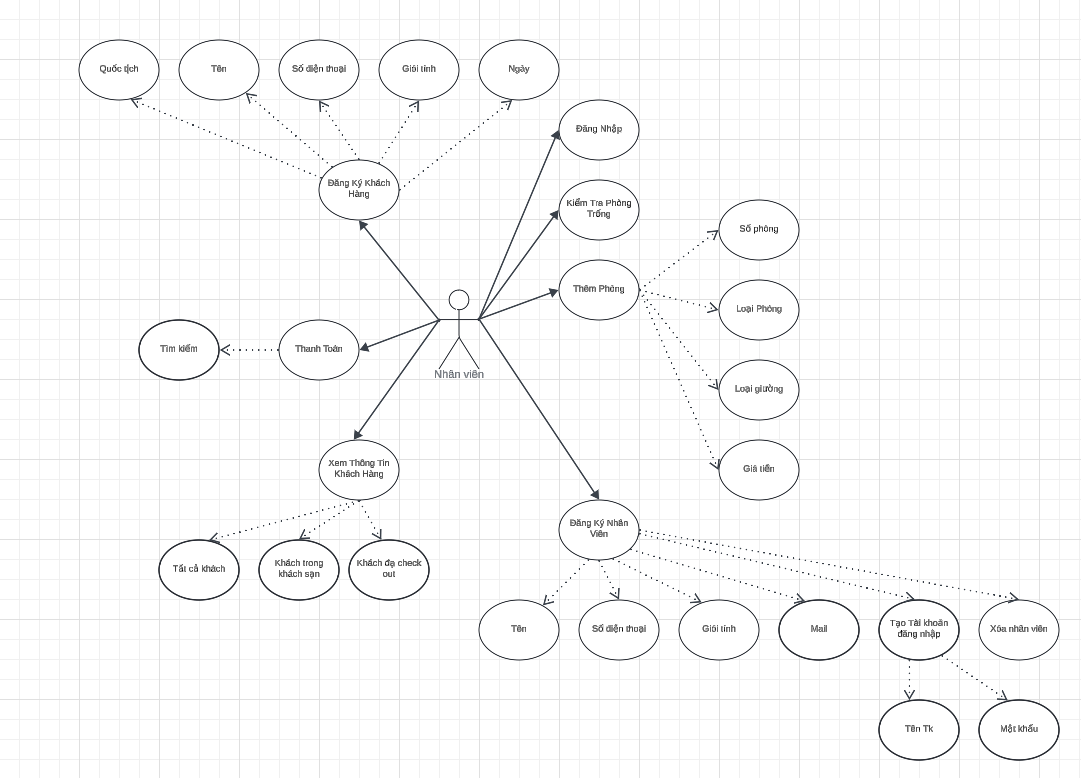
- Cho doanh nghiệp: Nâng cao hiệu quả hoạt động, tối ưu hóa chi phí, gia tăng doanh thu và duy trì lợi thế cạnh tranh.

- Cho khách hàng: Mang lại trải nghiệm lưu trú tốt hơn, check-in nhanh chóng, đến việc được phục vụ tận tình và chu đáo.

Dự án Quản lý Khách sạn không chỉ là một giải pháp công nghệ, mà còn là công cụ hỗ trợ mạnh mẽ cho sự phát triển bền vững của các doanh nghiệp trong ngành dịch vụ khách sạn.

**2. Các thiết kế của dự án**

*a. Sơ đồ Use case*

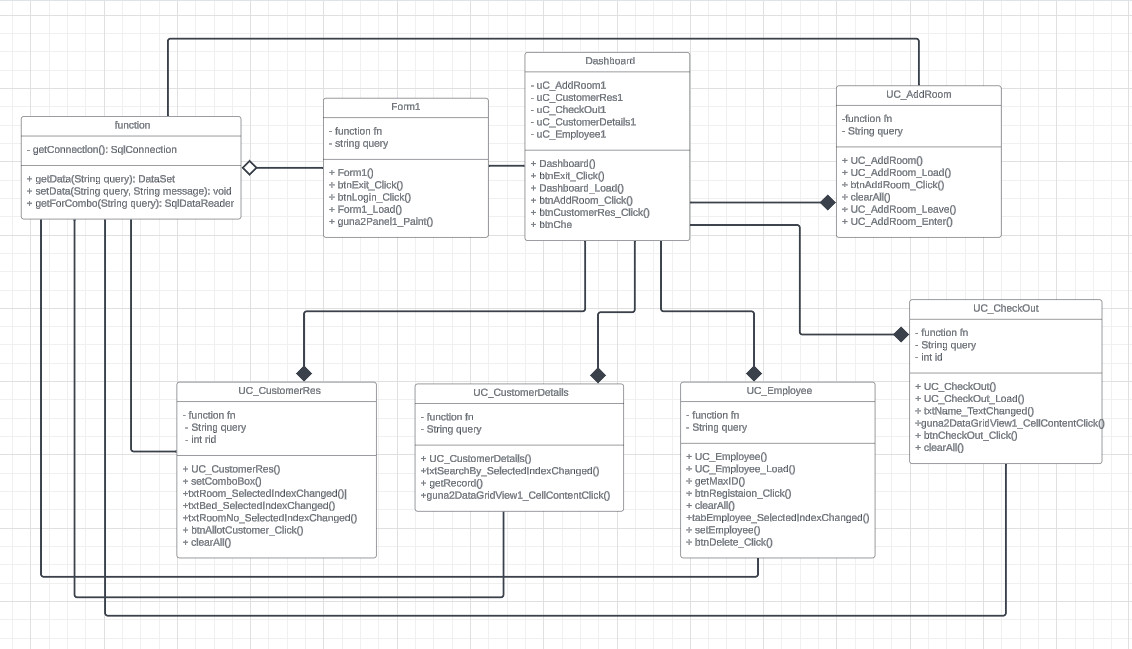
Trong dự án phát triển ứng dụng Quản lý khách sạn, sơ đồ Use Case có vai trò quan trọng trong việc mô tả các tác vụ và chức năng mà người dùng có thể thực hiện trong ứng dụng. Dưới đây là bản thiết kế sơ đồ Use Case của dự án:

<https://lucid.app/lucidchart/82cb1350-e3ee-4891-9b67-70963d414b9f/edit?viewport_loc=-1864%2C-340%2C3232%2C1656%2C0_0&invitationId=inv_502c62ea-ac19-4fee-bfc9-11339858ec58>

*b. Sơ đồ Class*

Trong dự án phát triển ứng dụng Quản lý khách sạn sơ đồ class có vai trò quan trọng trong việc mô tả cấu trúc và quan hệ

giữa các đối tượng (objects) trong hệ thống : Mô tả cấu trúc dữ liệu, quan hệ giữa các đối tượng, hiểu rõ cấu trúc hệ thống, hỗ trợ thiết kế và phát triển. Dưới đây là bản thiết kế sơ đồ Class của dự án:



**3. Các kỹ thuật lập trình hướng đối tượng sử dụng trong dự án**

Dựa trên yêu cầu của bạn, dưới đây là cách mà các khái niệm và các tính chất của lập trình hướng đối tượng được áp dụng trong các lớp đã được cung cấp:

*1. Tính trừu tượng (Abstraction):*

   - Các lớp như `function`, `UC\_AddRoom`, `UC\_CustomerRes`, `UC\_CheckOut`, `UC\_CustomerDetails`, và `UC\_Employee` đều sử dụng tính trừu tượng để che giấu chi tiết cài đặt nội bộ của các phương thức và thuộc tính của chúng.

*2. Tính đóng gói (Encapsulation):*

   - Các lớp đã được thiết kế với tính đóng gói bằng cách giữ các phương thức và thuộc tính riêng biệt, chỉ tiết lộ các phương thức công khai và thuộc tính cần thiết cho việc sử dụng bên ngoài.

   - các phương thức trong lớp `function` được sử dụng `getData`, `setData`, và `getForCombo`

*3.Tính đa hình (Polymorphism):*

   - Trong các lớp giao diện người dùng như `UC\_AddRoom`, `UC\_CustomerRes`, `UC\_CheckOut`, `UC\_CustomerDetails`, và `UC\_Employee`, chúng ta thường sử dụng sự đa hình để gọi các phương thức giống nhau với các tên khác nhau.

   -`Load`, `Click` được gọi từ các lớp giao diện người dùng khác nhau nhưng chúng thực hiện hành động khác nhau tùy thuộc vào ngữ cảnh.

*4. Tính kế thừa (Inheritance):*

   - Trong lớp `UC\_Employee`, khi người dùng nhấn vào nút "Đăng Ký" (`btnRegistaion\_Click`), thông tin của nhân viên mới được nhập vào các trường dữ liệu như tên, số điện thoại, email, giới tính, tên đăng nhập và mật khẩu.

   - Thông tin này được gửi đến phương thức `setData` của lớp `function`, nơi mà nó được thêm vào cơ sở dữ liệu như một bản ghi mới.

 - Trong lớp `Form1`, khi người dùng nhập tên đăng nhập và mật khẩu và nhấn nút "Đăng Nhập" (`btnLogin\_Click`), một truy vấn SQL được tạo để kiểm tra xem thông tin đăng nhập có khớp với bất kỳ bản ghi nào trong cơ sở dữ liệu không.

   - Nếu có, người dùng sẽ được chuyển hướng đến bảng điều khiển (`Dashboard`), ngược lại sẽ hiển thị một thông báo lỗi.

*5. Nạp chồng, ghi đè phương thức (Method Overloading, Overriding):*

   - Các phương thức trong các lớp có thể được nạp chồng hoặc ghi đè theo nhu cầu cụ thể của từng lớp.

   - Nạp chồng phương thức `setData` trong lớp `function` để hỗ trợ nhiều loại truy vấn hoặc có thể ghi đè phương thức `btnExit\_Click` trong lớp `Form1` để thực hiện hành động thoát khác nhau.

*6. Getter (), setter():*

   - Một số thuộc tính của các lớp có thể được truy cập thông qua các phương thức getter và setter để đảm bảo tính bảo mật và kiểm soát dữ liệu.

   - Trong lớp `UC\_AddRoom`, các phương thức như `txtRoomNo\_SelectedIndexChanged` có thể được sử dụng để lấy và thiết lập các giá trị của thuộc tính `rid`.

*7. Lưu trữ dữ liệu trên File hoặc sử dụng Hệ quản trị cơ sở dữ liệu:*

   - Trong các lớp như `function`, dữ liệu được lưu trữ và truy vấn từ một cơ sở dữ liệu SQL Server được xác định trong chuỗi kết nối.

   - Các phương thức như `getData`, `setData`, `getForCombo` được sử dụng để tương tác với cơ sở dữ liệu, trong khi các lớp giao diện người dùng sử dụng chúng để thực hiện các hoạt động liên quan đến cơ sở dữ liệu.

**4. Các công cụ sử dụng để thiết kế và phát triển dự án**

Ngôn ngữ lập trình: C#

Thư viện: guna2

Công cụ: thiết kế giao diện web bằng Visual studio

**5. Đánh giá tính năng ứng dụng**

*Ưu điểm:*

1. Tính linh hoạt: Dự án có sử dụng một số lượng lớn các lớp và giao diện người dùng, cho phép nó mở rộng và thay đổi dễ dàng khi cần thiết.

2. Tính tái sử dụng cao: Các lớp và phương thức được thiết kế để có thể tái sử dụng, giúp giảm thiểu việc viết lại mã và tăng hiệu quả phát triển.

3. Tính chia nhỏ và cấu trúc hóa tốt: Dự án được chia thành các lớp riêng biệt với trách nhiệm cụ thể, giúp dễ dàng quản lý và bảo trì mã nguồn.

4. Tính bảo mật: Việc sử dụng cơ sở dữ liệu và quản lý đăng nhập có thể cung cấp tính bảo mật cao cho ứng dụng.

*Nhược điểm:*

1. Thiếu minh bạch về kiến trúc: Dự án có thể cần thêm tài liệu hoặc giải thích để hiểu rõ về cấu trúc và mối quan hệ giữa các thành phần.

2. Thiếu kiểm tra và xử lý lỗi: Mặc dù các phương thức có tính đóng gói và trừu tượng, nhưng việc kiểm tra và xử lý lỗi có thể không đầy đủ.

3. Chưa có kiểm thử đầy đủ: Dự án có thể cần thêm kiểm thử để đảm bảo tính ổn định và độ tin cậy của ứng dụng.

**6. Kế hoạch phát triển tiếp theo của dự án**

*1. Mở rộng chức năng ứng dụng:*

   - Thêm chức năng quản lý nhân viên: Bao gồm việc thêm, sửa, xóa thông tin nhân viên, phân quyền truy cập, quản lý lịch làm việc, và thống kê hiệu suất làm việc.

   - Tích hợp tính năng đặt phòng trực tuyến: Cho phép khách hàng đặt phòng trực tuyến thông qua ứng dụng và xem tình trạng phòng trống.

   - Thêm tính năng quản lý dịch vụ: Bao gồm quản lý các dịch vụ khách sạn như nhà hàng, spa, hồ bơi, và cung cấp dịch vụ đi kèm.

*2. Cải thiện giao diện người dùng:*

   - Thiết kế giao diện người dùng đồ họa (GUI) thân thiện hơn: Tối ưu hóa trải nghiệm người dùng bằng cách cải thiện giao diện, thêm hiệu ứng và phản hồi người dùng tốt hơn.

   - Tối ưu hóa trải nghiệm người dùng trên thiết bị di động: Đảm bảo ứng dụng hiển thị và hoạt động một cách tốt nhất trên các thiết bị di động khác nhau bằng cách sử dụng Responsive Design hoặc thiết kế ứng dụng di động riêng.

*3. Kiểm thử và đảm bảo chất lượng:*

   - Thực hiện kiểm thử đầy đủ: Kiểm tra ứng dụng để đảm bảo tính đúng đắn, tính ổn định và hiệu suất.

   - Sửa lỗi và tối ưu hóa: Xử lý các lỗi phát sinh và tối ưu hóa hiệu suất của ứng dụng để đảm bảo hoạt động mượt mà và ổn định.

*4. Thu thập và phản hồi người dùng:*

   - Thu thập phản hồi từ người dùng: Hỏi ý kiến từ người dùng và sử dụng phản hồi này để cải thiện và mở rộng tính năng ứng dụng.

   - Liên tục cập nhật và phát triển: Dựa vào phản hồi từ người dùng và xu hướng thị trường để liên tục cập nhật và phát triển ứng dụng.