

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**🕯✡🕮🕮✡🕯**

**ĐỒ ÁN**

***Đề tài:* WEB QUẢN LÝ ĐẶT PHÒNG**

***Môn học: Hệ quản trị cở sở dữ liệu***

**GVHD: Ths. HOÀNG LONG**

**Sinh viên thực hiện MSSV**

1. **Nguyễn Duy Phước 16146172**
2. **Phan Thị Ngọc Tuyền 17110248**

**Tp. Hồ Chí Minh, tháng 12 năm 2019**

**MỤC LỤC**

**[LỜI NÓI ĐẦU 4](#_Toc22759)**

**[CHƯƠNG I: GIỚI THIỆU 5](#_Toc13269)**

**[1. Hiện trạng 5](#_Toc5121)**

**[CHƯƠNG II: YÊU CẦU HỆ THỐNG 7](#_Toc6795)**

**[1. Use-Case 7](#_Toc25582)**

**[2. Chức năng 8](#_Toc14890)**

**[2. 1. Chủ trang web 8](#_Toc1015)**

**[2. 2. Nhân viên 8](#_Toc22015)**

**[2. 3. Khách hàng 8](#_Toc21197)**

**[3. Cơ sở dữ liệu 9](#_Toc16421)**

**[CHƯƠNG III: PHÂN TÍCH HỆ THỐNG 10](#_Toc22405)**

**[1. Thiết kế kiến trúc 10](#_Toc4896)**

**[1.1 MVC là gì? 10](#_Toc15183)**

**[1.2.1. Model: 11](#_Toc20782)**

**[1.2.2. Control: 12](#_Toc17917)**

**[1.2.3. VIEW: 12](#_Toc24385)**

**[2. Thiết kế chi tiết 13](#_Toc7054)**

**[2.1. Admin 13](#_Toc16585)**

**[2.1.1. PhongController 14](#_Toc11107)**

**[2.1.2. LoaiPhongController 15](#_Toc25258)**

**[2.1.3. ChuongTrinhController 16](#_Toc1476)**

**[2.1.4. DatPhongController 17](#_Toc30490)**

**[2.1.5. DatPhongCartController 18](#_Toc13020)**

**[2.1.6. DichVusController 19](#_Toc12603)**

**[2.1.7. LoaiDichVusController 20](#_Toc15965)**

**[2.1.8. GetListDichVuController và GetListDatPhongController 20](#_Toc26280)**

**[2.2. Customer 21](#_Toc1379)**

**[3. Thiết kế giao diện 22](#_Toc3666)**

**[3.1.1. Trang home 22](#_Toc81)**

**[3.1.2. Room 23](#_Toc3937)**

**[3.2.1. HomeAdmin 24](#_Toc20670)**

**[3.2.2. Phòng 25](#_Toc10223)**

**[3.2.3. Loại phòng 27](#_Toc23472)**

**[3.2.4. Dịch vụ 28](#_Toc13022)**

**[3.2.5. Loại dịch vụ 30](#_Toc15024)**

**[3.2.6. Chương trình 32](#_Toc3350)**

**[3.2.7. Account 34](#_Toc25215)**

**[3.3.1. Đặt phòng 36](#_Toc7657)**

**[3.3.2. Đặt, Nhận, Thanh toán 36](#_Toc5233)**

**[3.3.3. Hủy đặt phòng 38](#_Toc25461)**

**[4. Thiết kế cơ sở dữ liệu 39](#_Toc6082)**

**[4.1. Mô hình ERD 39](#_Toc11068)**

**[4.2. Database diagram 39](#_Toc25179)**

**[4.3. Phân tích chi tiết thực thể 40](#_Toc8263)**

**[4.4. Giải thích các mối quan hệ 44](#_Toc21722)**

**[4.4.1. Phong và LoaiPhong 44](#_Toc25484)**

**[4.4.2. Phong và ChuongTrinh 44](#_Toc3046)**

**[4.4.3. DichVu và LoaiDichVu 44](#_Toc18552)**

**[4.4.4. Phong và DatPhong 44](#_Toc6417)**

**[4.4.5. DatPhong và DichVu 44](#_Toc24262)**

**[4.4.6. DatPhong và User 44](#_Toc31754)**

**[4.4.7. Phong va TrangThai 44](#_Toc30580)**

**[4.5. Các hàm trong bảng 45](#_Toc19097)**

**[4.6. Các Function - Proce đặc biệt 48](#_Toc6323)**

**[CHƯƠNG III: HIỆN THỰC HÓA HỆ THỐNG 49](#_Toc20663)**

**[1. Database 49](#_Toc5857)**

**[CHƯƠNG IV: KẾT LUẬN 60](#_Toc14900)**

**[1. Ưu điểm 60](#_Toc28026)**

**[2. Nhược điểm 60](#_Toc16344)**

**[3. Hướng phát triển 60](#_Toc30816)**

**LỜI NÓI ĐẦU**

Ngày nay, ngành Công nghệ thông tin phát triển với tốc độ chóng mặt, đặc biệt là internet ngày càng phổ biến. Nó thúc đẩy quá trình phẳng hóa thế giới nhanh hơn. Nhu cầu sử dụng mạng ngày càng cao, giúp mọi người có thể giao tiếp với nhau mà không cần quan tâm đến khoảng cách địa lý. Từ đó các ngành kinh doanh cũng phát triển mạnh mẽ vì người dùng có thể đặt trực tuyến qua mạng. Điển hình là ngành du lịch, khách hàng có thể lên trang web của khách sạn để xem thông tin phòng trống, giá phòng, các loại dịch vụ, sự kiện và đặt phòng cho chuyến đi của mình mà không cần lo lắng đến chỗ nghỉ ngơi khi đi xa. Dựa theo đó, việc quản lý đặt phòng cho khách hàng thuận tiên hơn, giúp cho doanh thu của khách sạn đi lên nhanh chóng và bắt kịp với nền công nghiệp 4.0.

Vì vây, chúng em chọn ý tưởng xây dựng hệ thống quản lý khách sạn và phát triển lên ứng dụng web đưa ra môi trường kết nối khách sạn và các khách hàng thường xuyên có nhiều chuyến đi.

Chúng em xin chân thành cảm ơn thầy Hoàng Long đã giúp đỡ chúng em hoàn thành đồ án này. Tuy đã có nhiều cố gắng nhưng trong quá trình làm đề tài không thể tránh được sai sót. Chúng em rất mong nhận được ý kiến đóng góp từ thầy để chúng em rút kinh nghiệm thực hiện tốt hơn ở các đồ án sau.

Chúng em xin chân thành cảm ơn!

CHƯƠNG I: GIỚI THIỆU

1. Hiện trạng

Ngày nay, do phát triển của Internet khiến thói quen tìm kiếm của người dùng thay đổi khá nhiều, hầu hết mọi người đều lên internet tìm kiếm thông tin trước khi đến tận nơi để trải nghiệm dịch vụ, đây là điều tất nhiên khi mà mọi thông tin đều có sẵn trên internet và bạn hoàn toàn không phải trả phí khi tìm kiếm, ngược lại bạn có thể tìm được những thứ hay ho hơn hoặc những lựa chọn tốt hơn. Chính vì vậy việc thiết kế một trang web để giới thiệu dịch vụ của mình đến khách hàng trên internet là vô cùng quan trọng trong thời điểm hiện tại, thời điểm mà hầu như mọi người đều sở hữu cho mình những thiết bị công nghệ, kèm theo các gói internet di động giúp họ có thể tìm kiếm thông tin mọi lúc mọi nơi. Không chỉ có thể cung cấp thông tin, hỗ trợ quảng bá thương hiệu, thiết kế web riêng riêng còn giúp bạn thể hiện sự chuyên nghiệp trong hoạt động và cách quản lý của bạn.

2. Lý do xây dựng đề tài

Khách sạn đang quản lí một cách thủ công, cần có một công cụ hỗ trợ việc cho việc kinh doanh khách sạn, vì vậy…

Viết ứng dụng web quản lý cho khách sạn xây dựng môi trường giao tiếp giữa khách sạn và khách hàng của họ mà không cần e ngại về khoảng cách.

3. Đối tượng sử dụng

**Chủ khách sạn** có quyền cao nhất, quyền quản lý hệ thống bằng cách thêm xóa sửa phòng, loại phòng, dịch vụ, loại dịch vụ và nhân viên.

**Nhân viên** có quyền đăng nhập vào hệ thống sử dụng các quyền của nhân viên như: đặt phòng cho khách, cho khách nhận / trả phòng, thêm và xóa dịch vụ cho khách, xuất hóa đơn cho khách.

**Khách hàng** có quyền xem thông tin các loại phòng, loại dịch vụ trên trang web. Sau đó, khi muốn đăt phòng, họ sẽ liên hệ đến khách sạn và tiến hành đặt phòng.

CHƯƠNG II: YÊU CẦU HỆ THỐNG

1. Use-Case



1. Chức năng
2. Chủ trang web

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên chức năng** | **Mô tả** |
| 1 | Đặt phòng:   1. Tìm phòng trống 2. Chọn phòng/dịch vụ 3. Xác nhận đặt phòng | Khi khách hàng liên hệ đặt phòng |
| 2 | Thêm ,xóa, sửa (phòng, dịchvụ, chương trình) | Khi chủ homestay có nhu cầu thay đổi kết cấu |

1. Nhân viên

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên chức năng** | **Mô tả** |
| 1 | Đặt/Nhận phòng | Khi khách hàng liên hệ đặt phòng |
| 2 | Thanh Toán | Khi khách hàng hoàn tất việc trả phòng |

1. Khách hàng

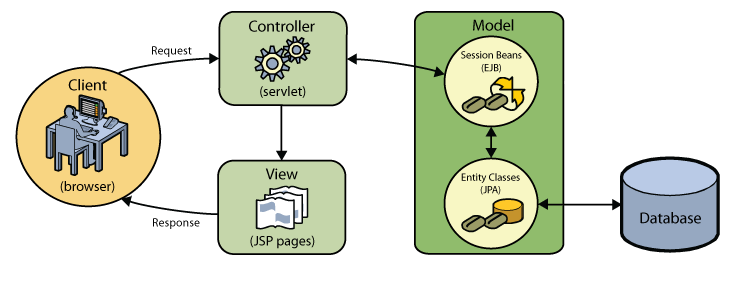
|  |  |  |
| --- | --- | --- |
| **STT** | **TÊN CHỨC NĂNG** | **MÔ TẢ** |
| 1 | Xem thông tin khách sạn |  |

1. Cơ sở dữ liệu

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **TÊN CHỨC NĂNG** | | **YÊU CẦU VỀ CƠ SỞ DỮ LIỆU** |
| 1 | Đặt phòng:  - Tìm phòng trống  - Chọn phòng/dịch vụ  - Xác nhận đặt phòng | | Xem thông tin phòng trống và trữ thông tin: Tên khách hàng, phòng, dịch vụ (nếu có) |
| 2 | Thêm, xóa, sửa | Phòng | Lưu trữ thông tin phòng: Tên, giá, hình ảnh,… |
| Loại Phòng | Lưu trữ thông tin loại phòng: Tên |
| Dịch vụ | Lưu trữ thông tin dịch vụ: Tên ,giá,mô tả… |
| Loại dịch vụ | Lưu trữ thông tin loại dịch vụ: Tên |
| Chương trình | Lưu trữ thông tin chương trình áp dụng cho khách sạn vụ: Tên, sự thay đổi giá |
| 3 | Xem thông tin khách sạn | | Xem thông tin phòng |

CHƯƠNG III: PHÂN TÍCH HỆ THỐNG

1. Thiết kế kiến trúc
   1. MVC là gì?



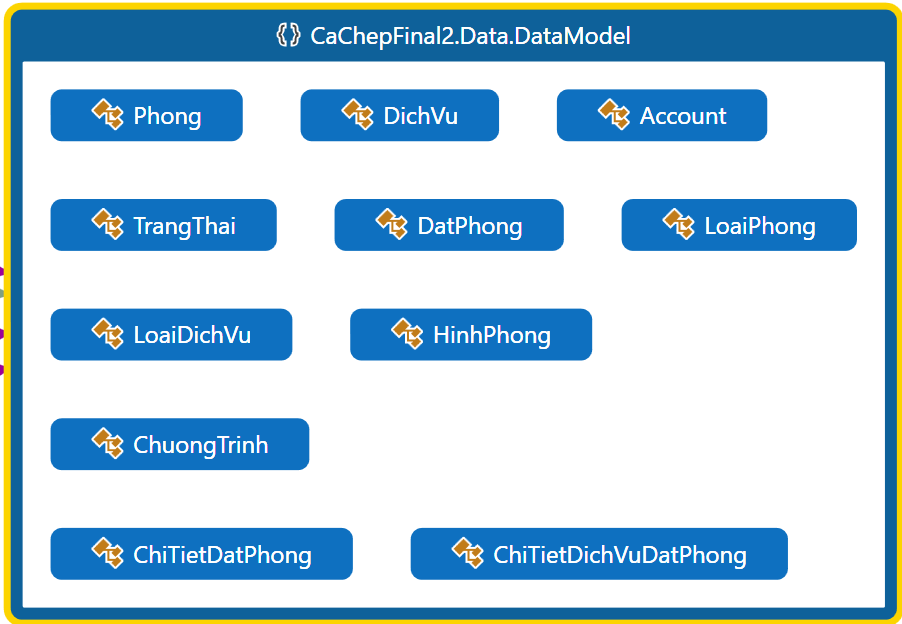
Control: Giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng… Chẳng hạn thành phần này sẽ nhận request từ url và form để thao tác trực tiếp với Model.

Model: Đây là thành phần chứa tất cả các nghiệp vụ logic, phương thức xử lý, truy xuất database, đối tượng mô tả dữ liệu như các Class, hàm xử lý…

View: Đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images… Hiểu một cách đơn giản, nó là tập hợp các form hoặc các file HTML.

* 1. Xây dựng WEB ĐẶT PHÒNG HOMESTAY theo mô hình MVC ASP.net core

1. Model:



1. Control:

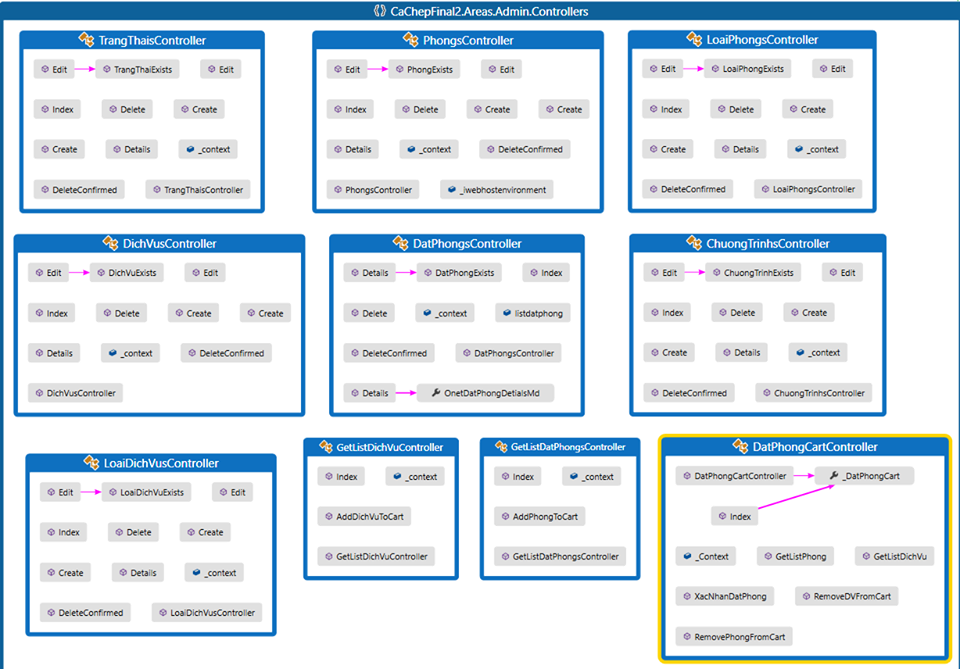
Web đặt phòng có hai nhóm control chính: Admin, và Customer.



1. VIEW:

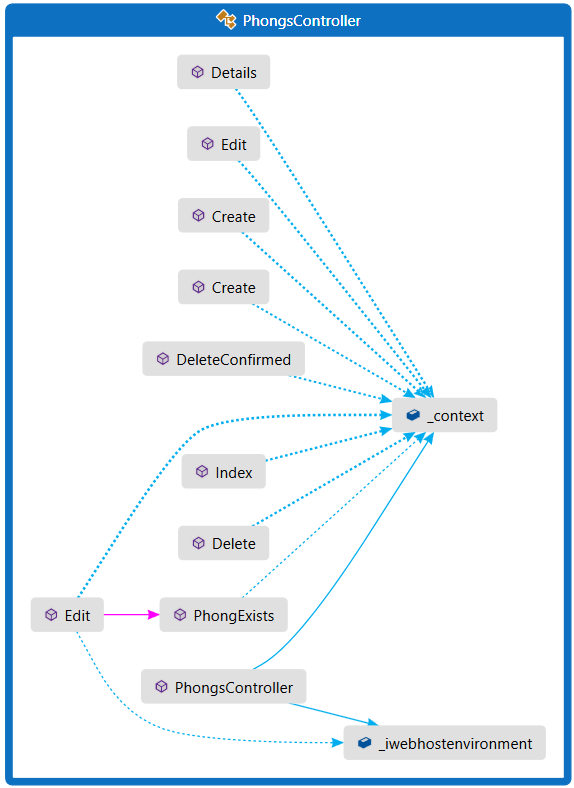
phân tích chi tiết ở phần thiết kế giao diện

1. Thiết kế chi tiết
2. Admin



1. PhongController

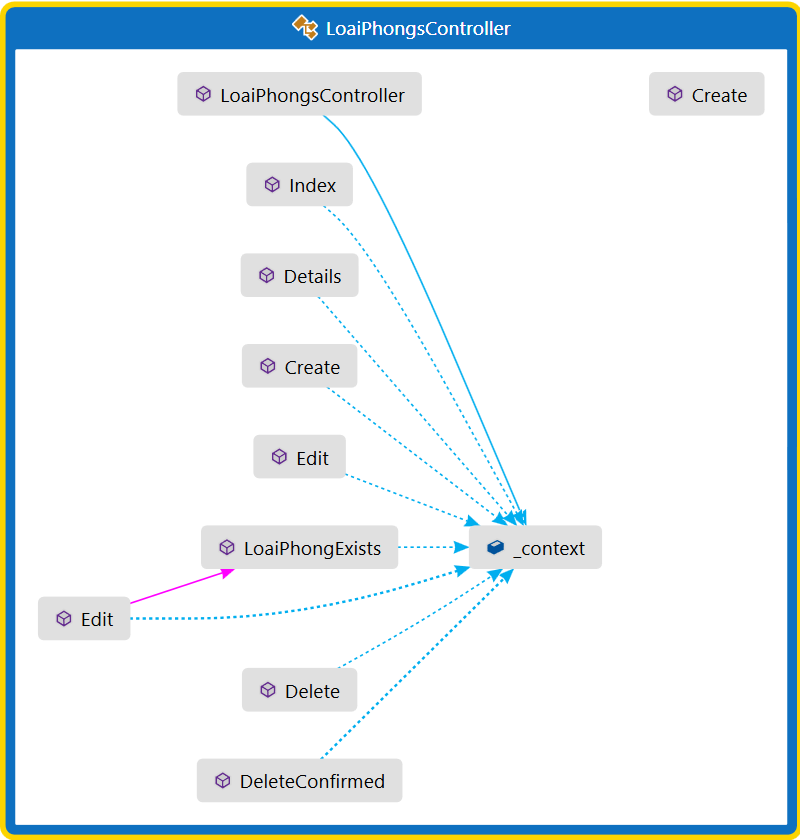
phục vụ cho chức năng xem, xóa, sửa, số lượng, thông tin phòng

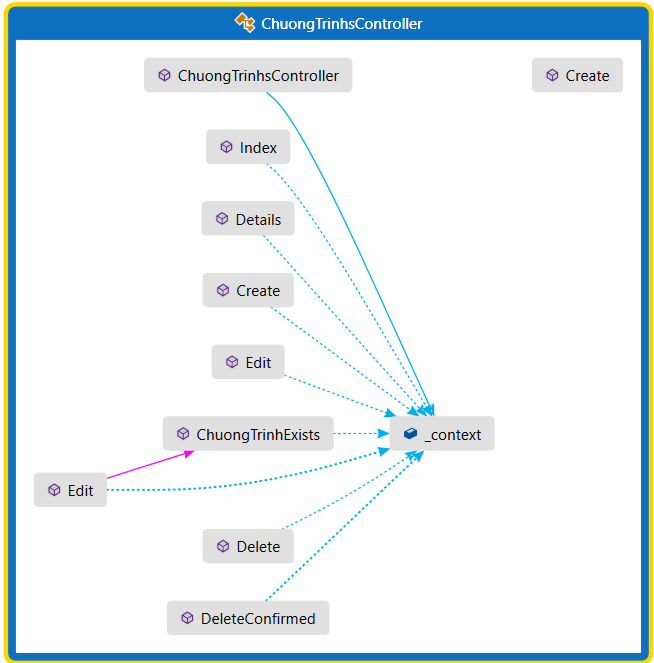


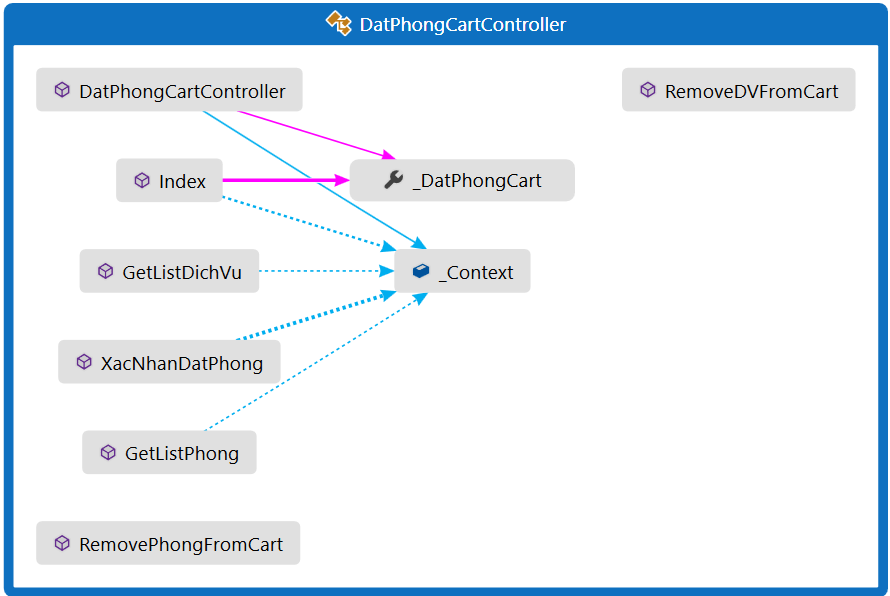
1. LoaiPhongController

Phục vụ cho chức năng xem, xóa, sửa lại

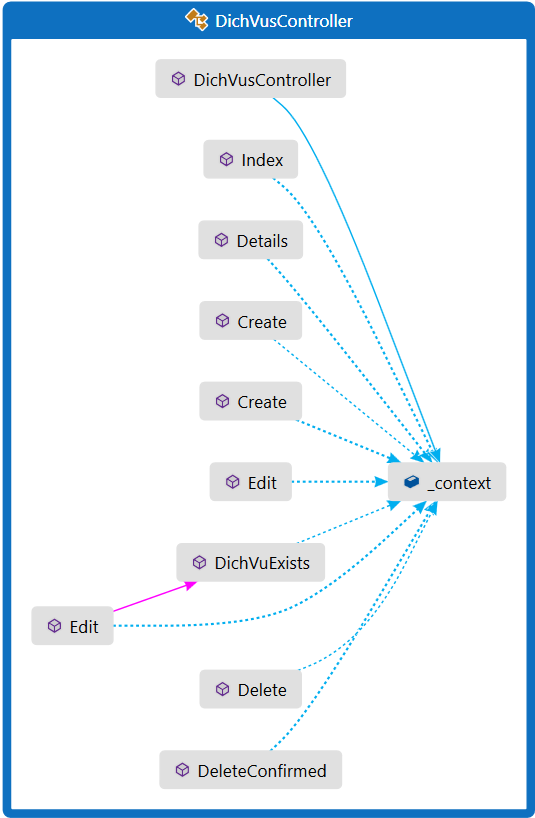
Phòng

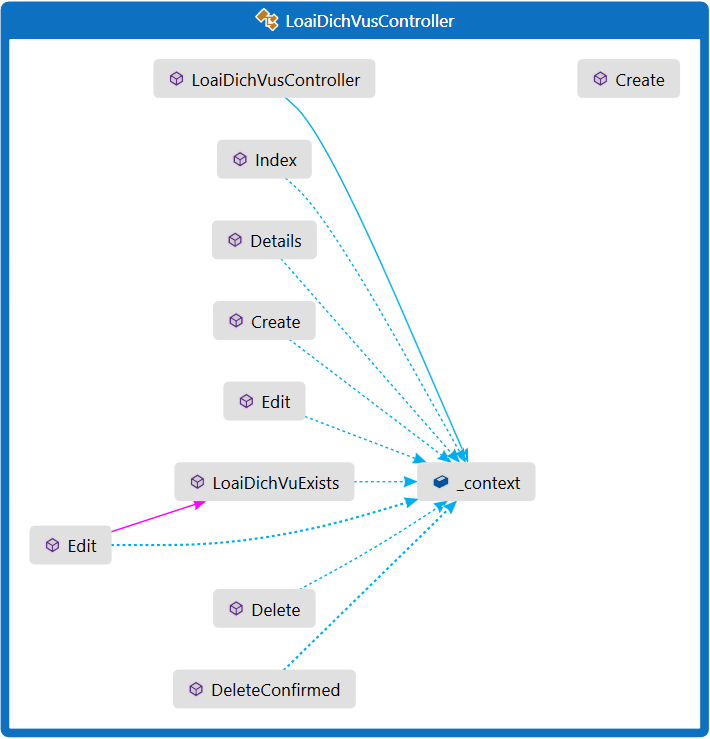


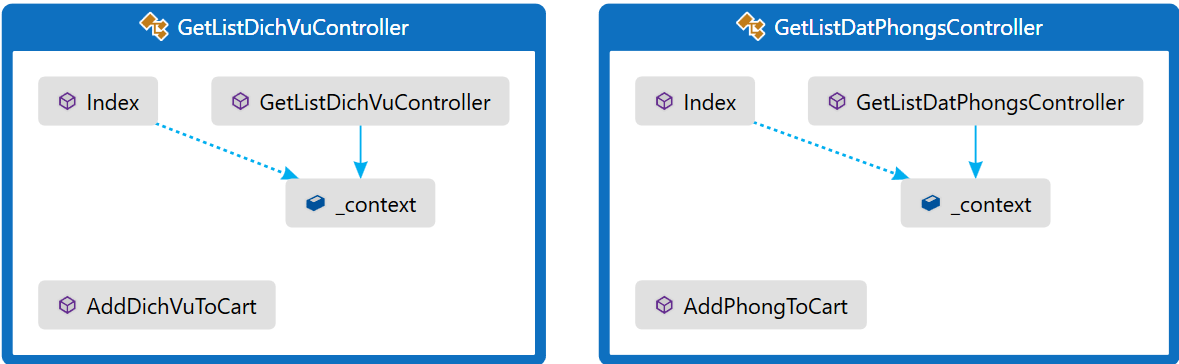
1. ChuongTrinhController
2. DatPhongController
3. DatPhongCartController



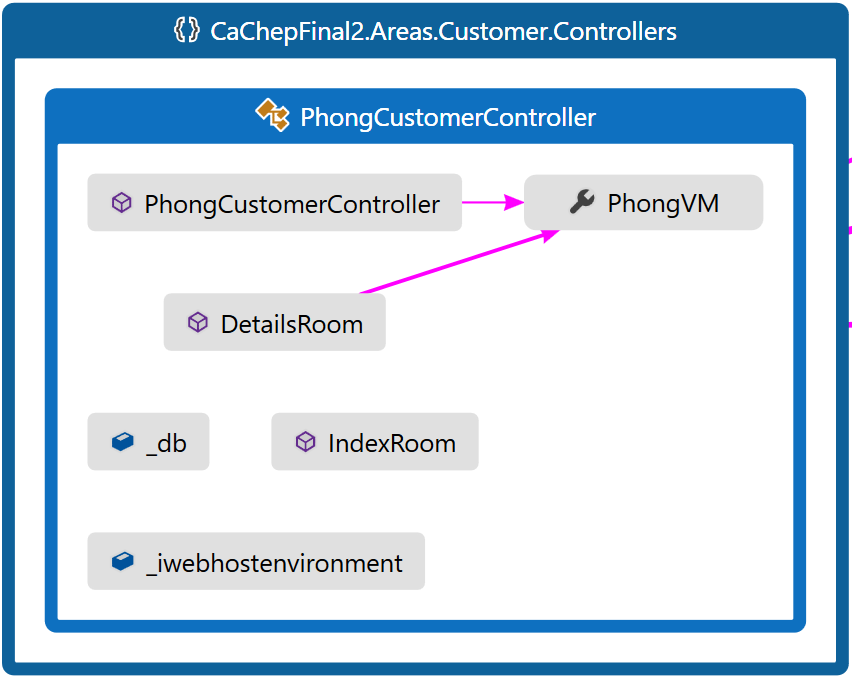
1. DichVusController



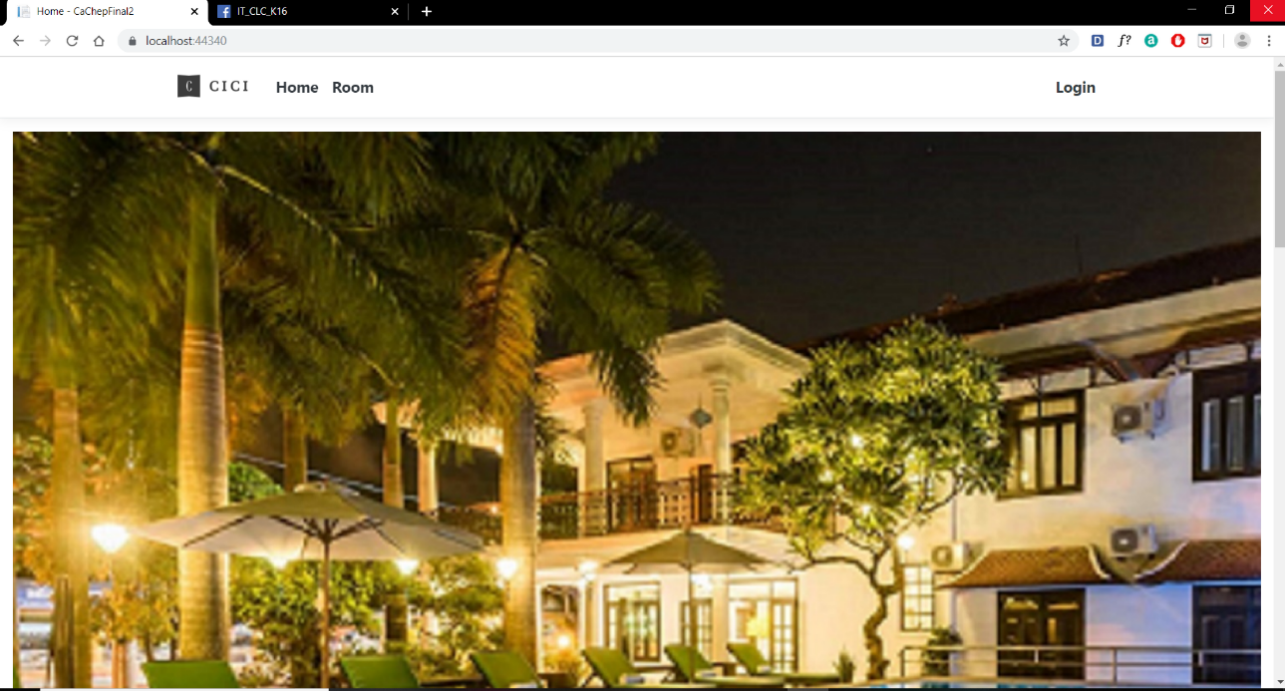
1. LoaiDichVusController
2. GetListDichVuController và GetListDatPhongController

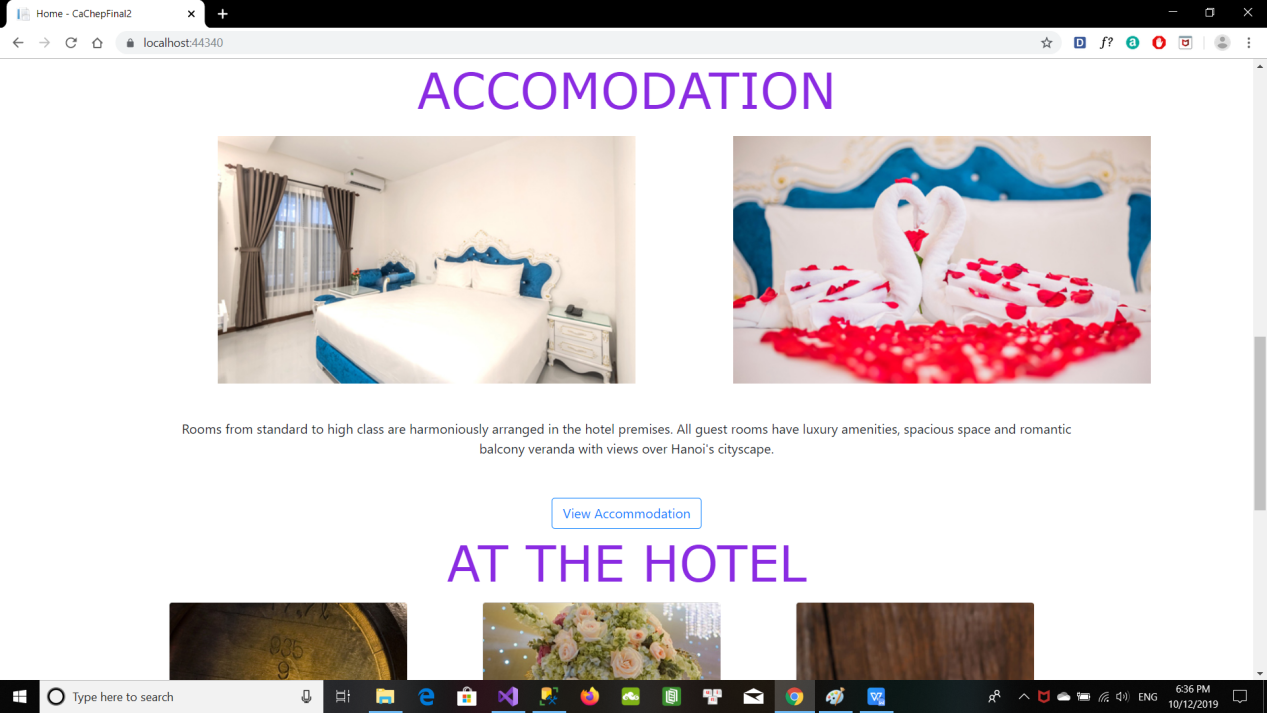


1. Customer



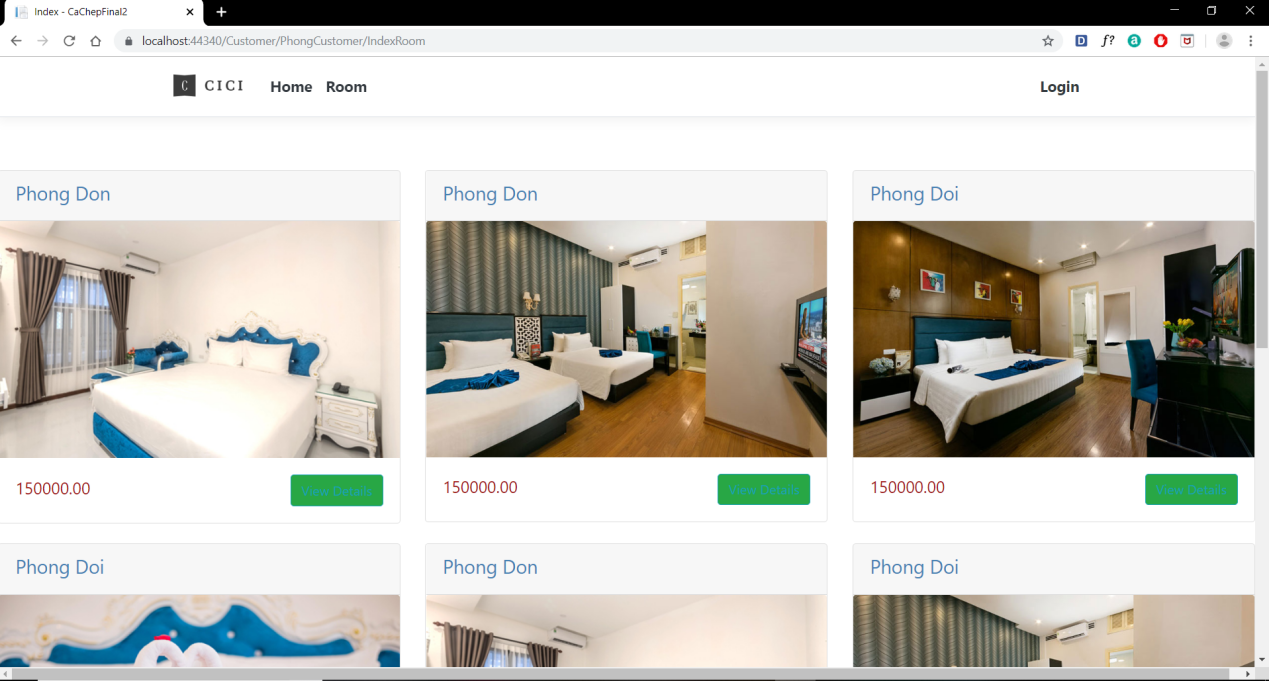
1. Thiết kế giao diện
2. Giao diện khách hàng
3. Trang home



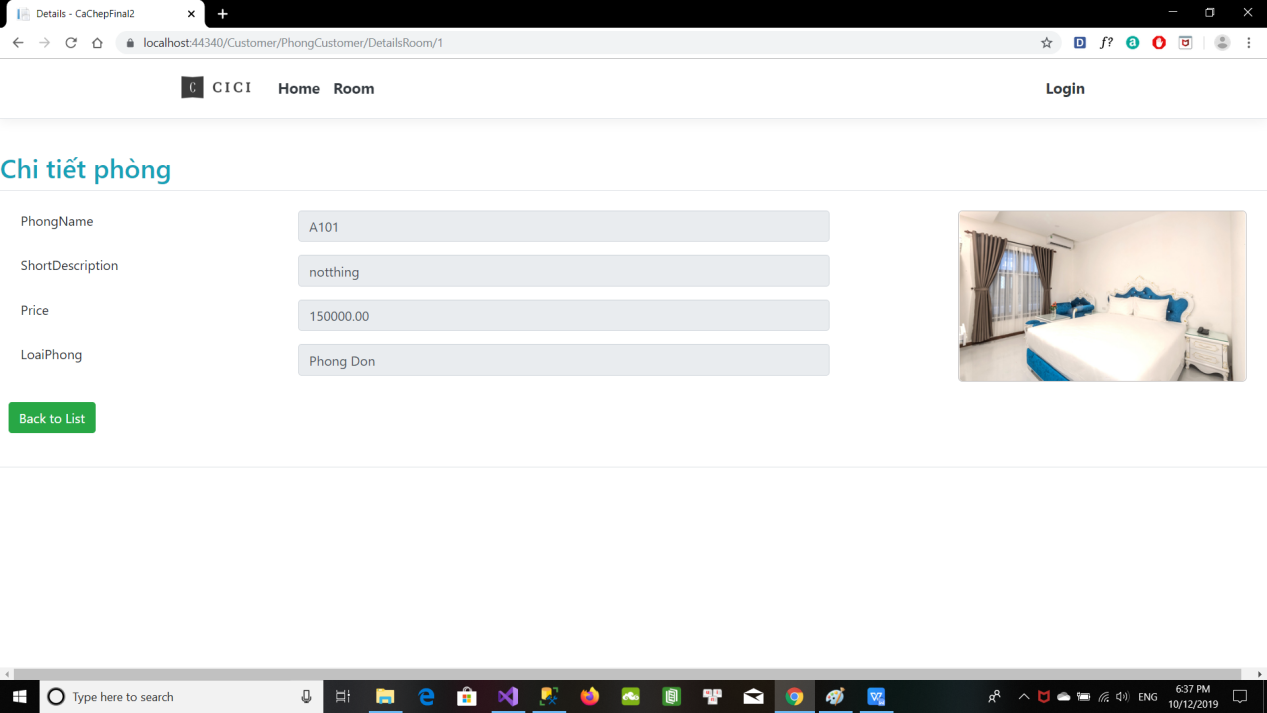


1. Room

Tổng quan



Chi tiết

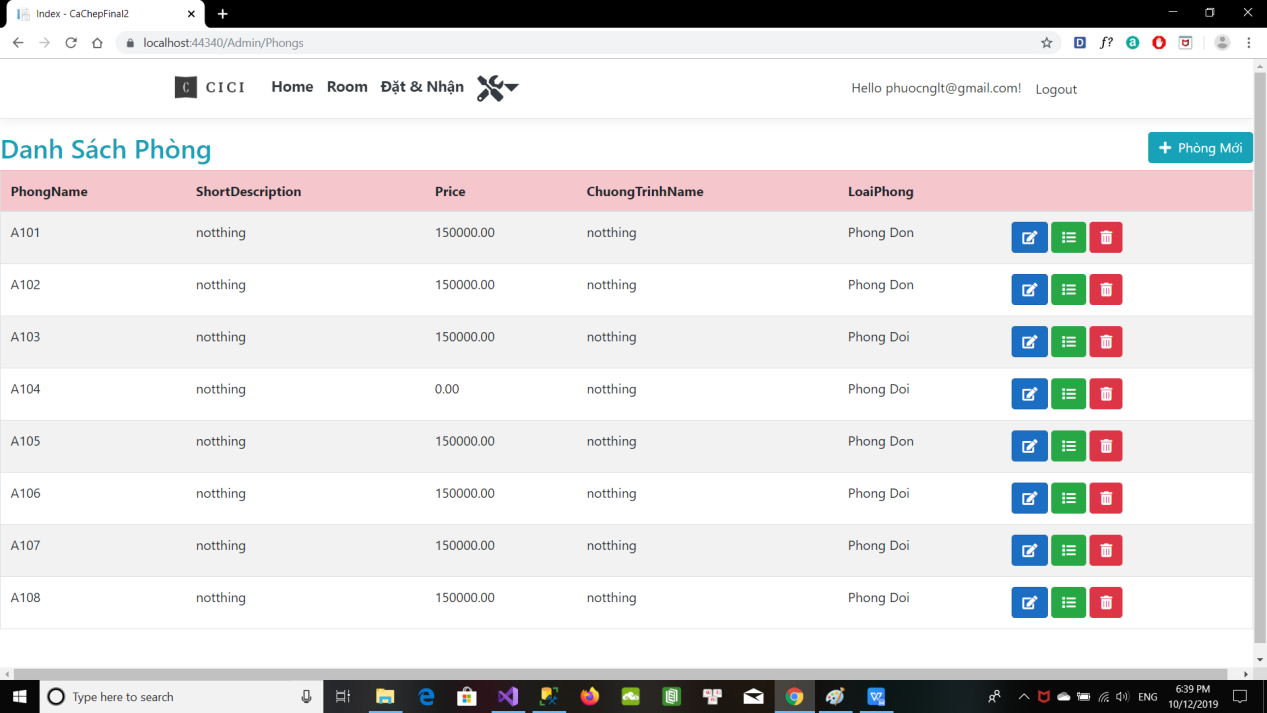


1. Admin – Chủ khách sạn

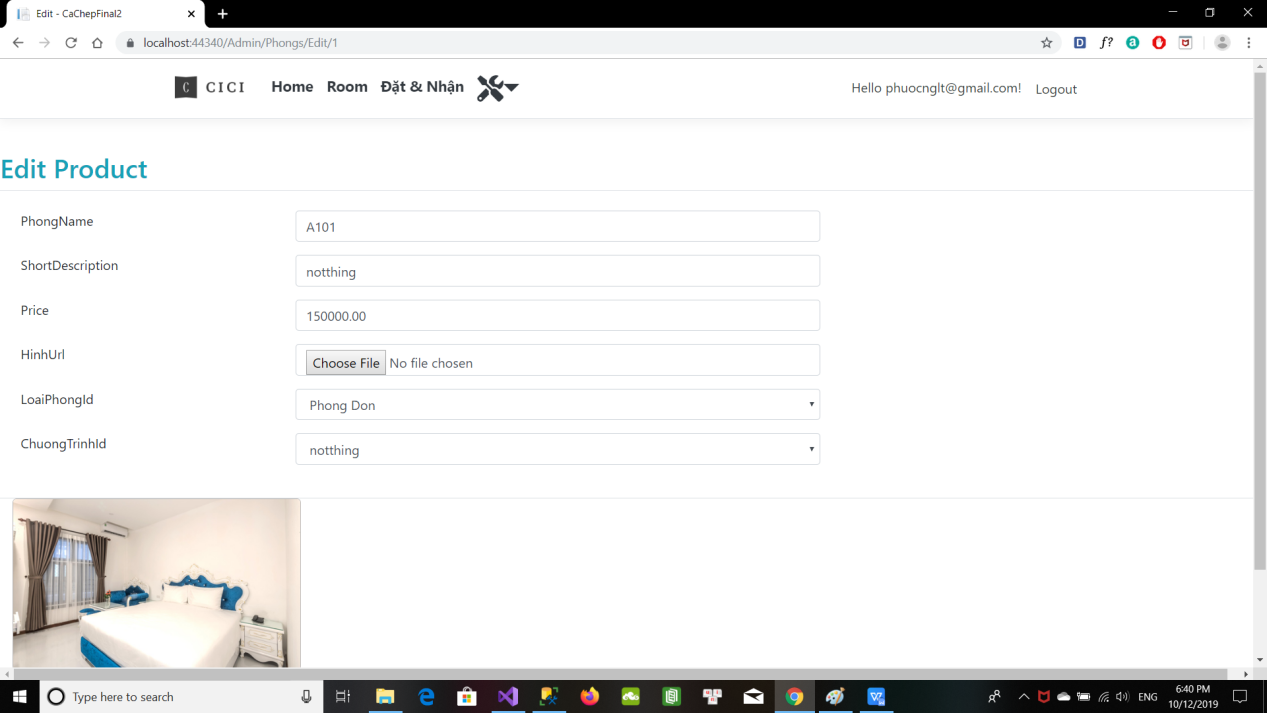
Chủ khách sạn được dùng các chức năng sau: Home, Room, Đặt & nhận, Cấu hình hệ thống

1. HomeAdmin
2. Phòng

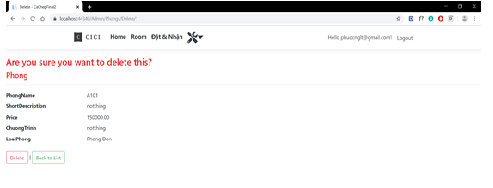
Index



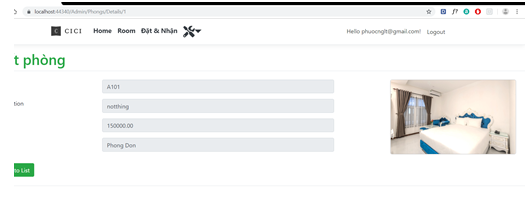
Edit



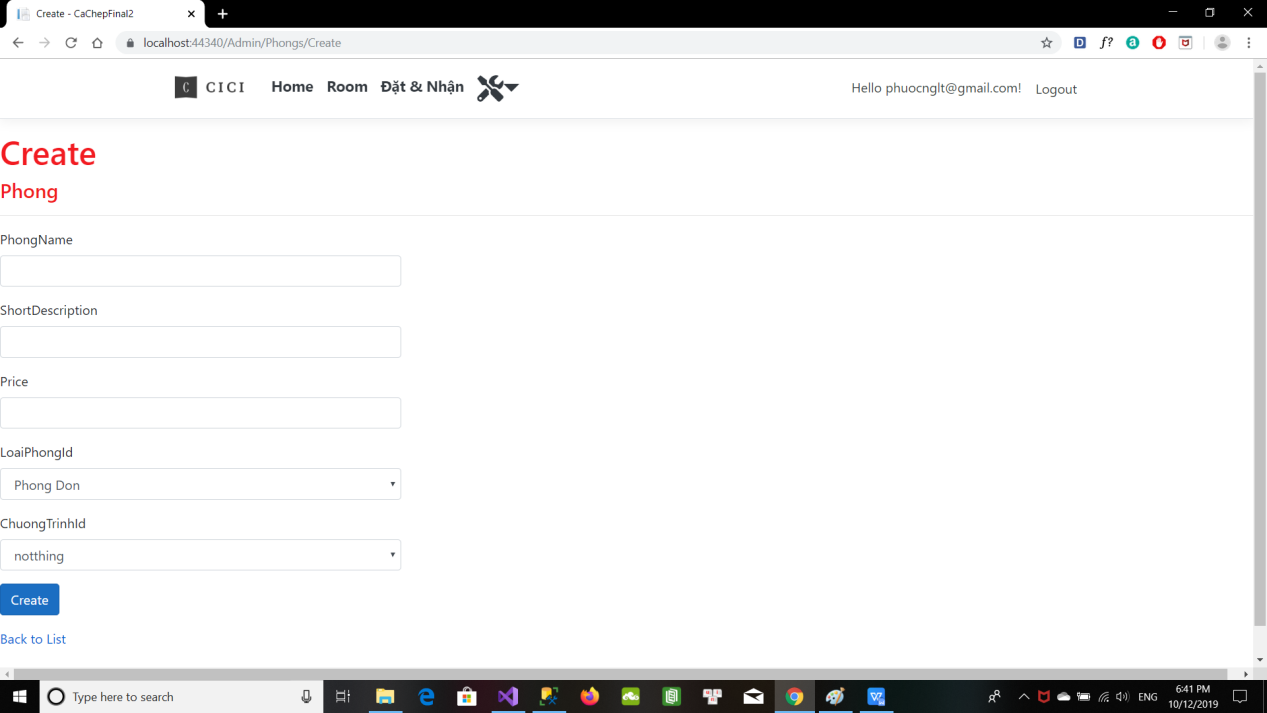
Delete



Detail



Create

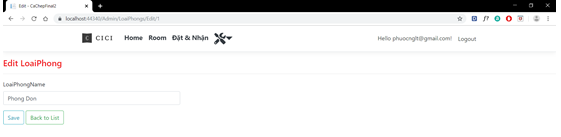


1. Loại phòng

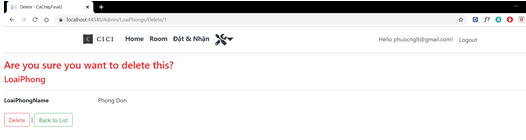
Index



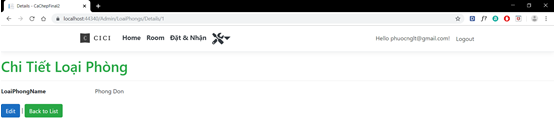
Edit



Delete

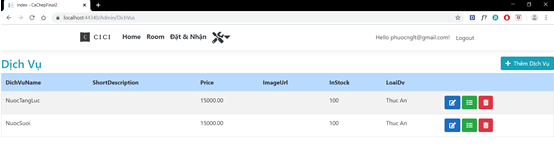


Detail

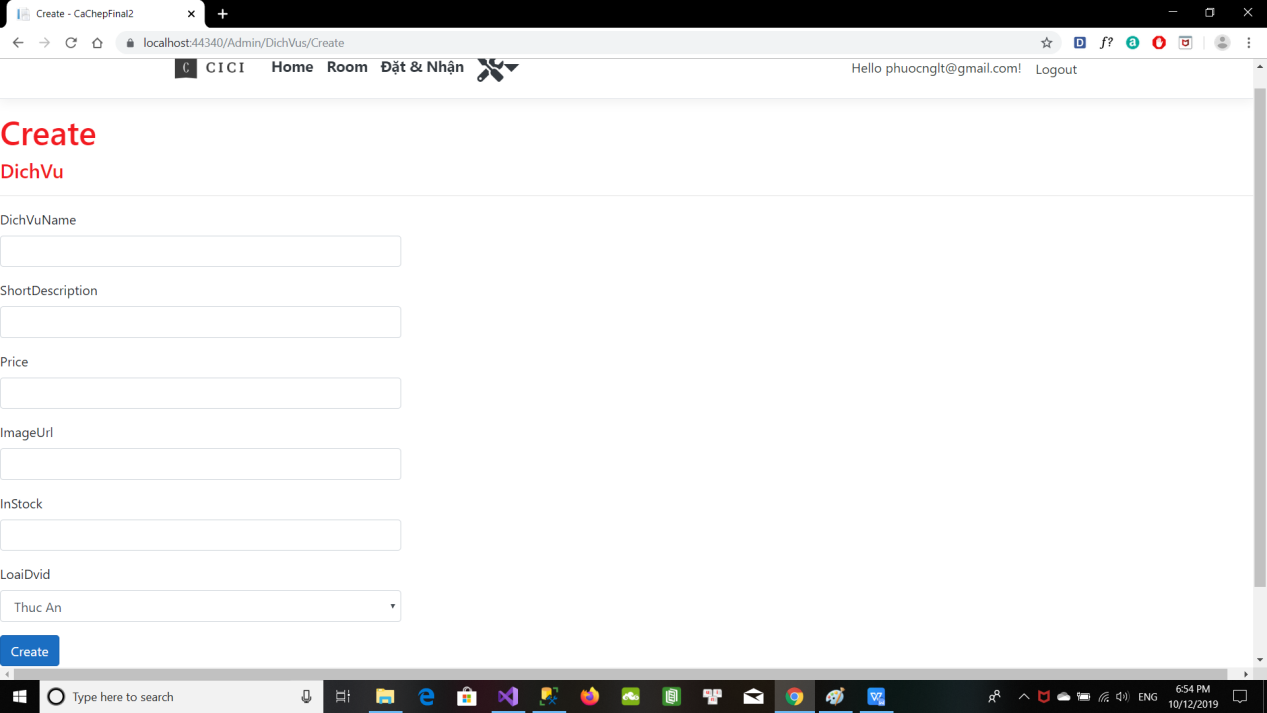


1. Dịch vụ

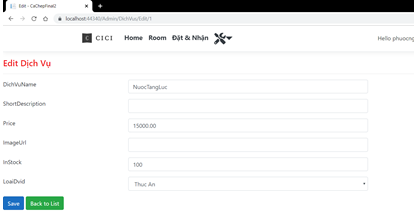
Index



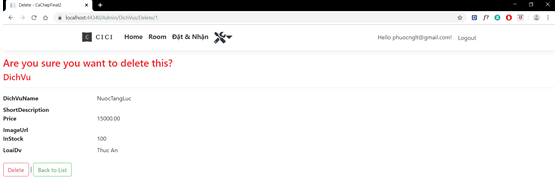
Create



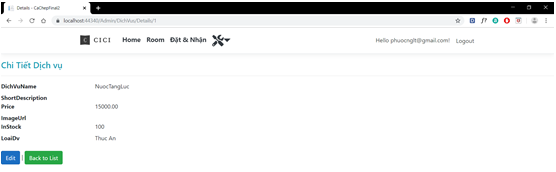
Edit



Delete

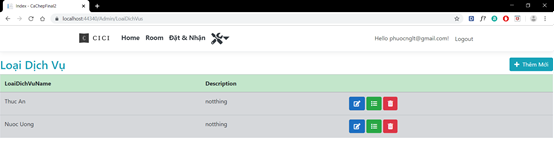


Detail

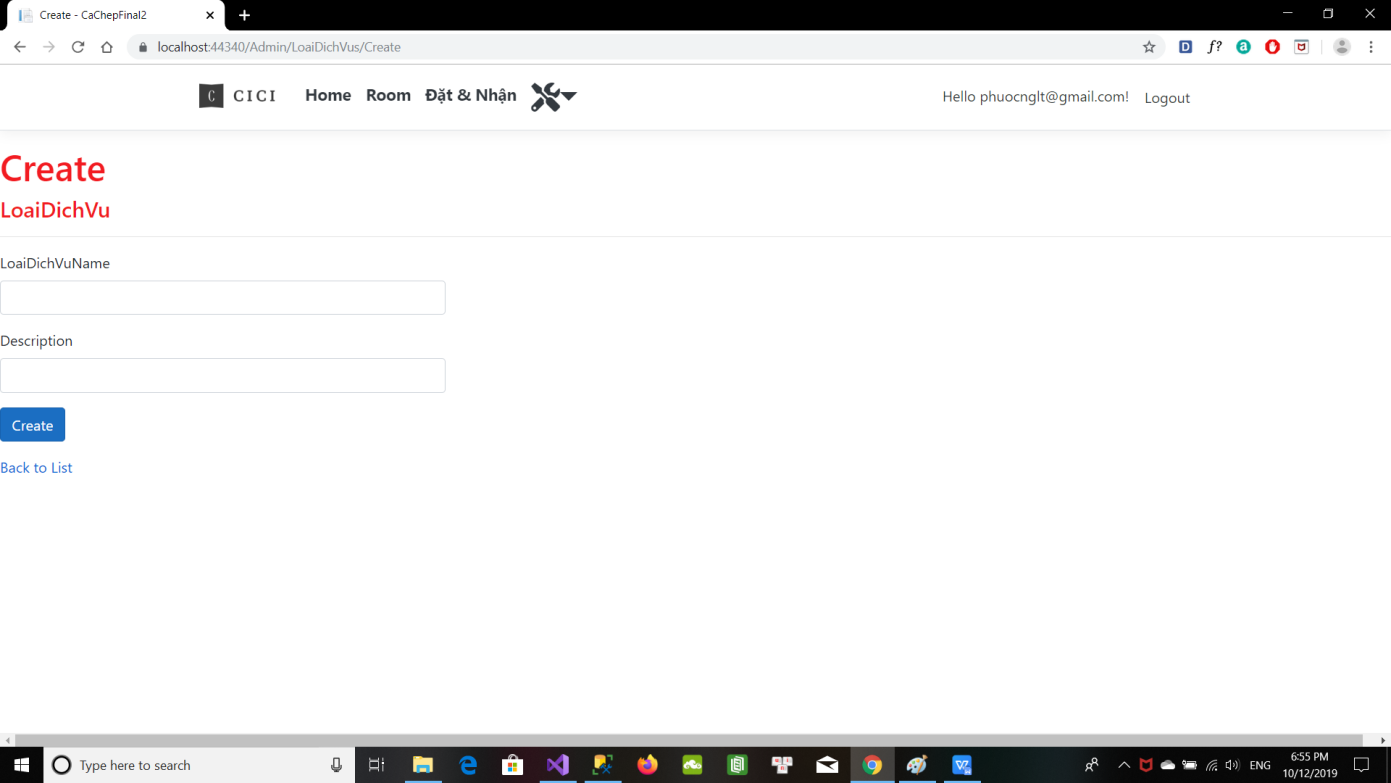


1. Loại dịch vụ

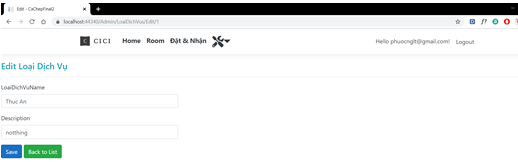
Index



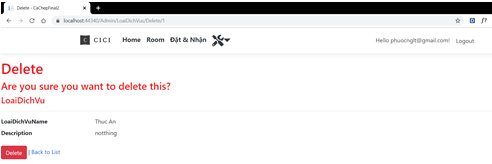
Create



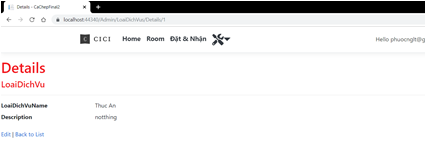
Edit



Delete

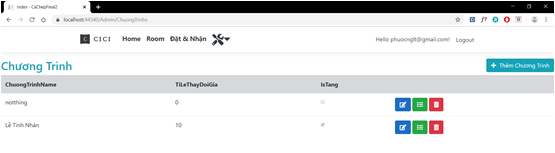


Detail

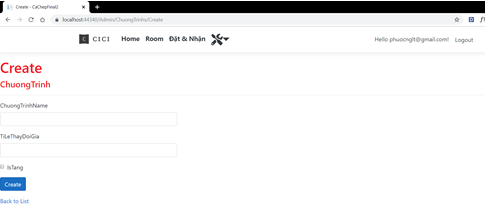


1. Chương trình

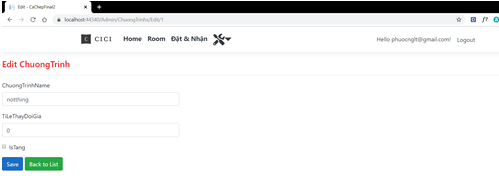
Index



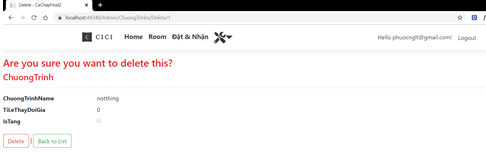
Create



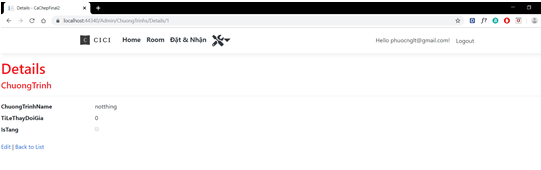
Edit



Delete



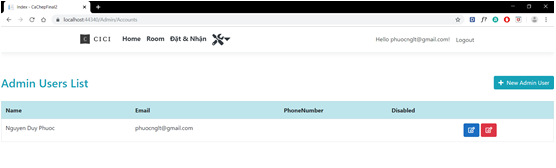
Detail



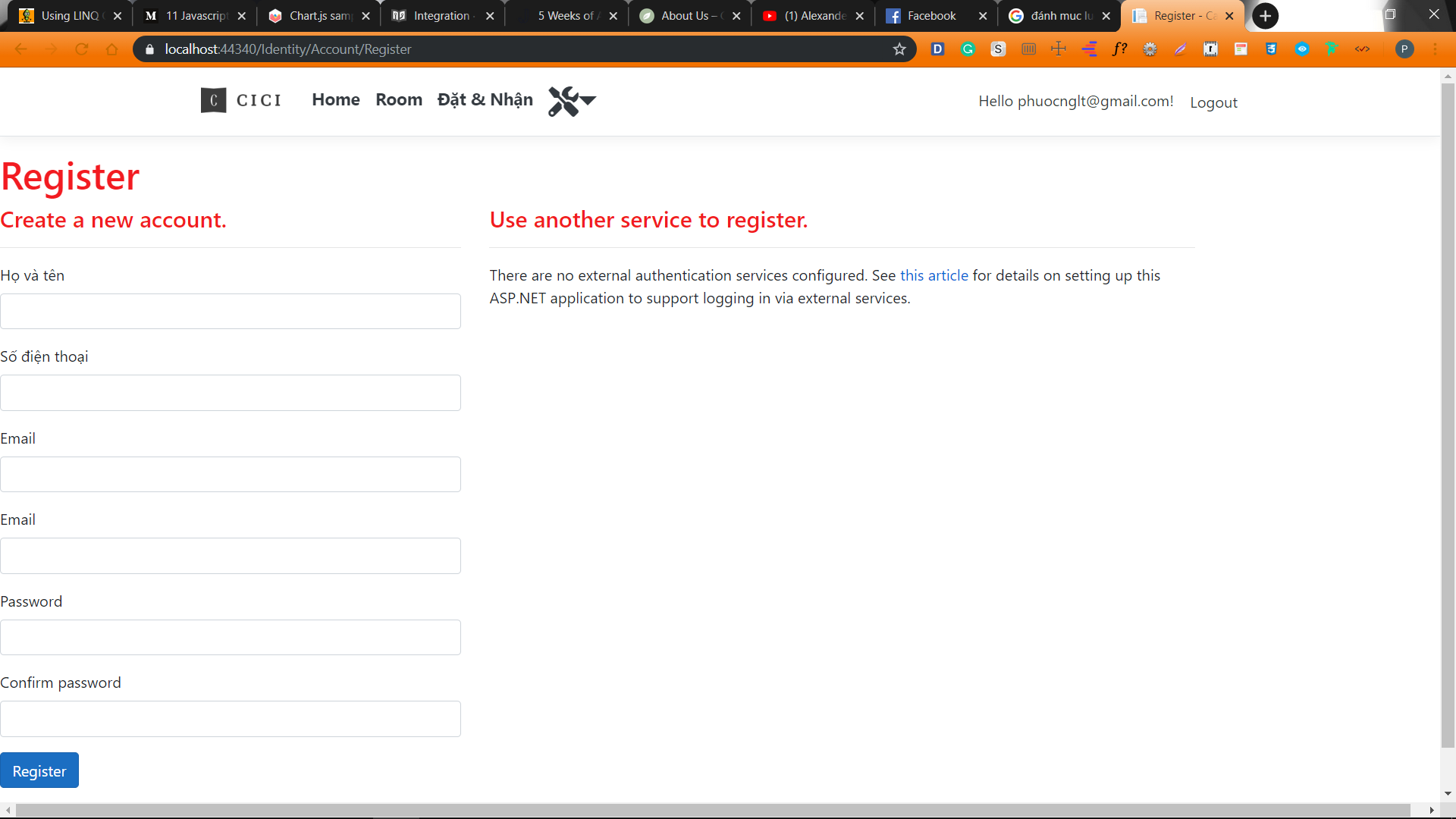
1. Account

Nơi chủ khách sạn đăng kí account cho nhân viên

Index



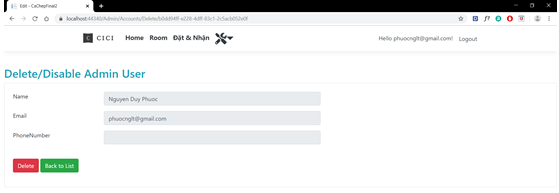
Create



Edit



Delete

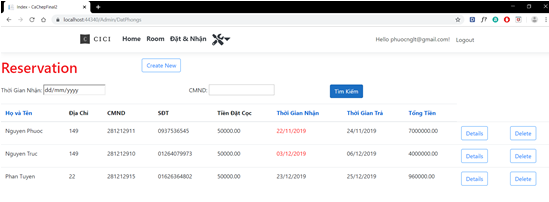


1. Nhân viên

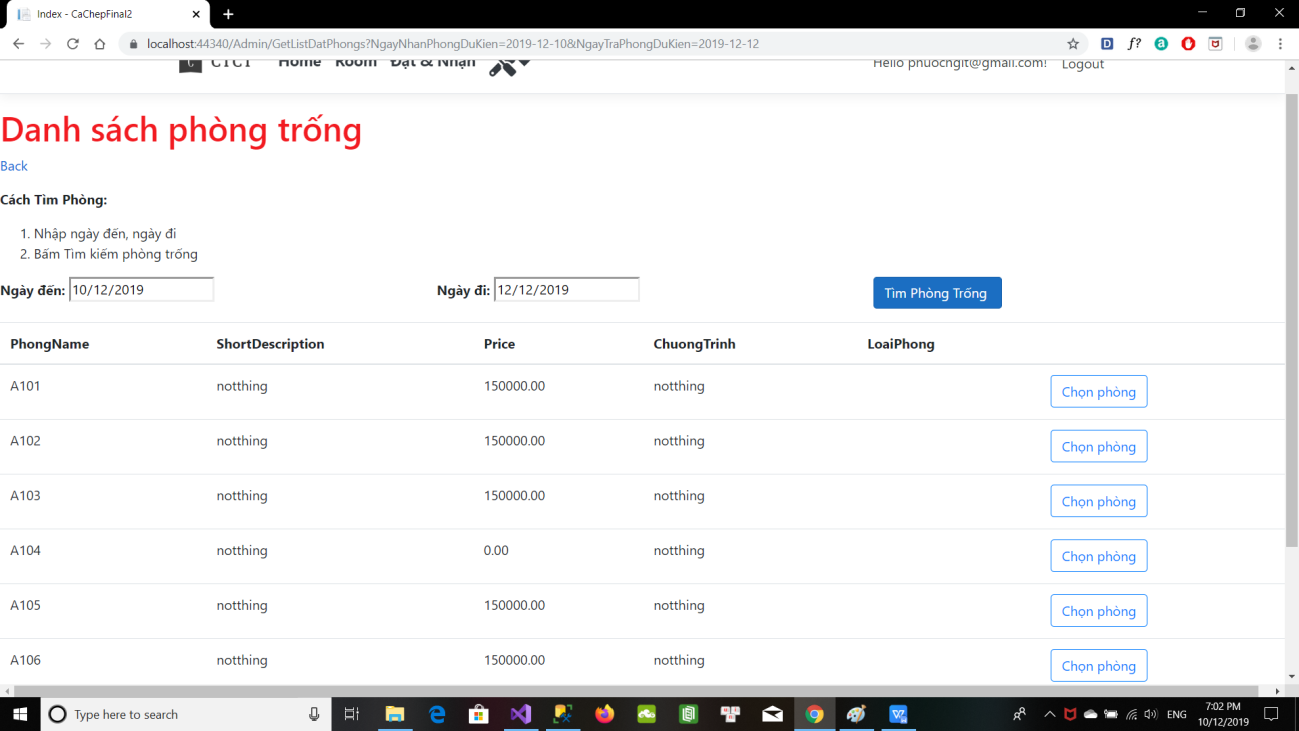
Nhân viên được dùng các chức năng sau: Home, Room, Đặt & nhận

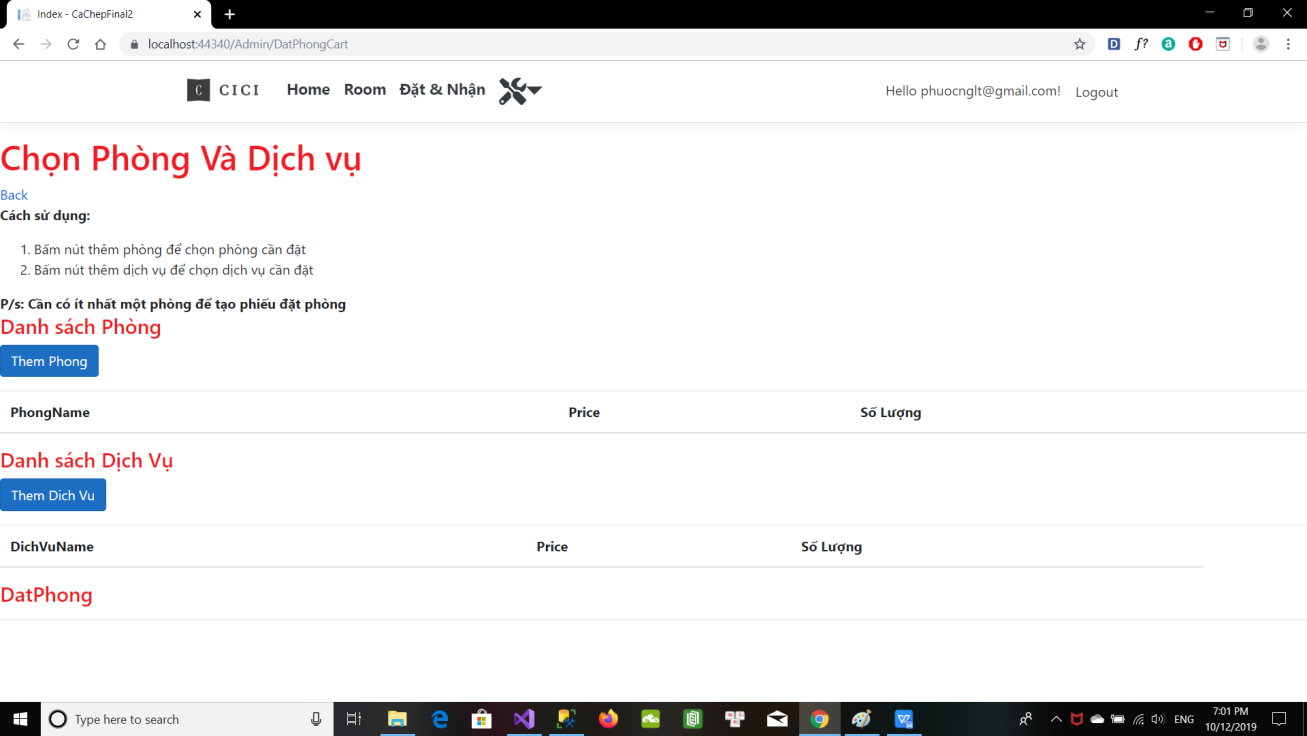
1. Đặt phòng

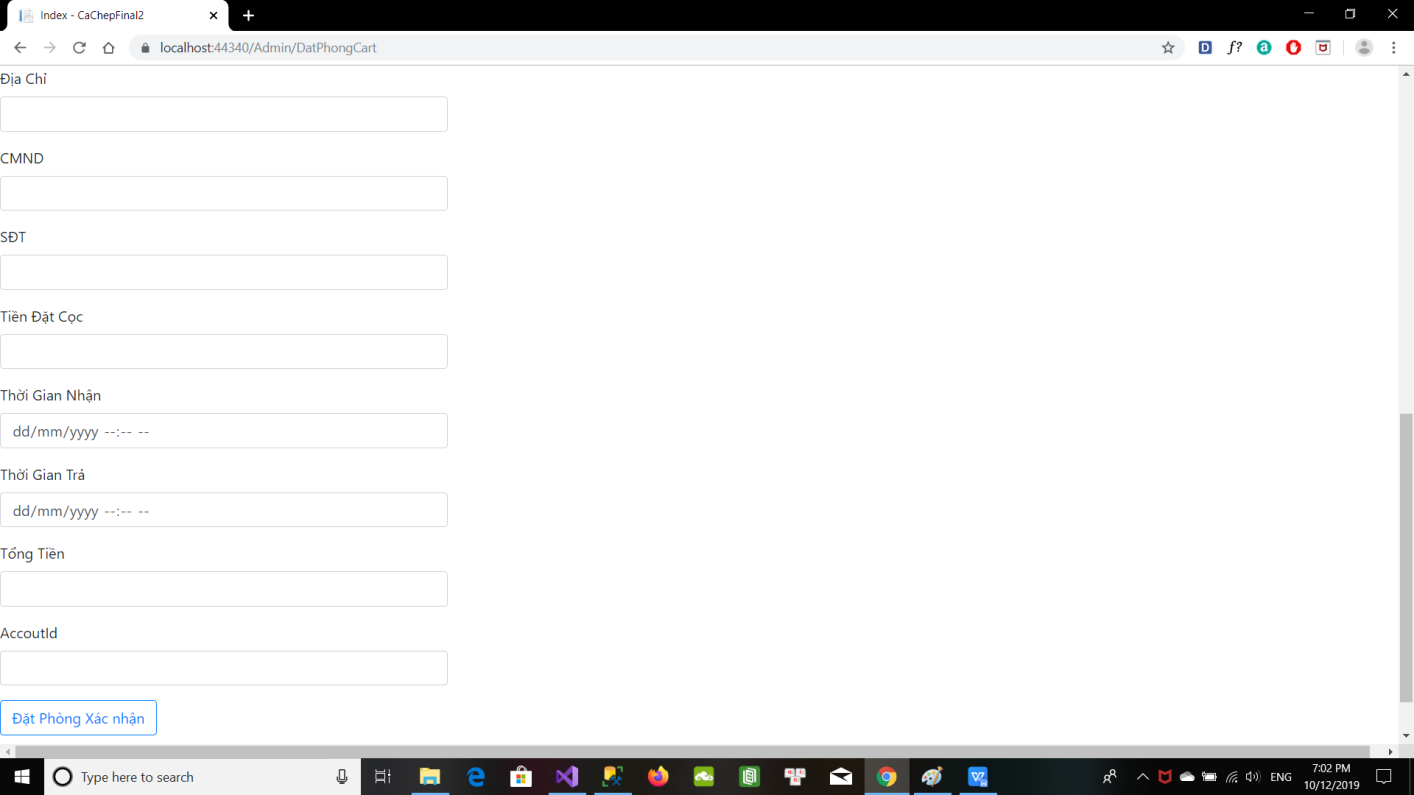
Index

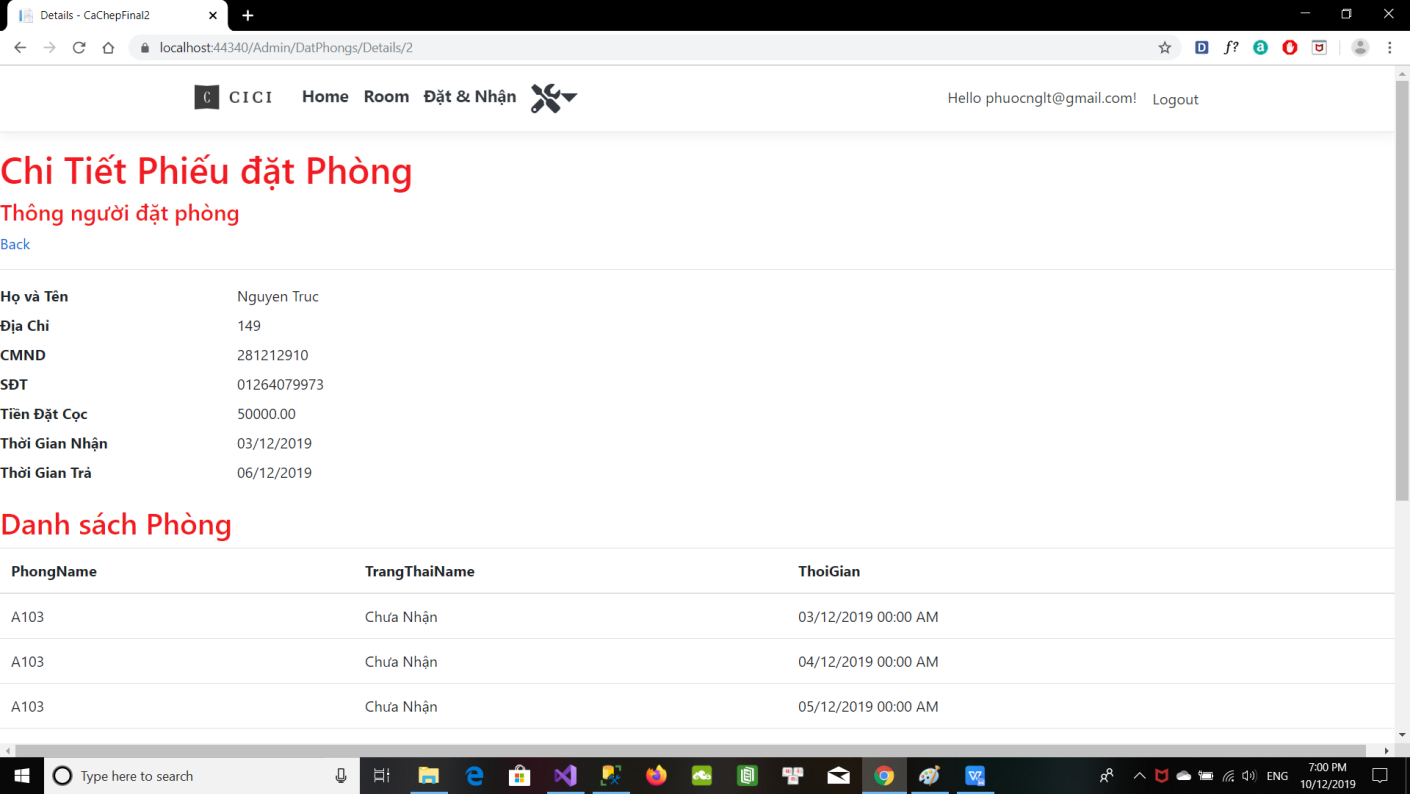


1. **Đặt, Nhận, Thanh toán**

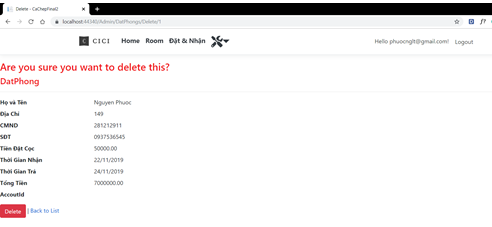




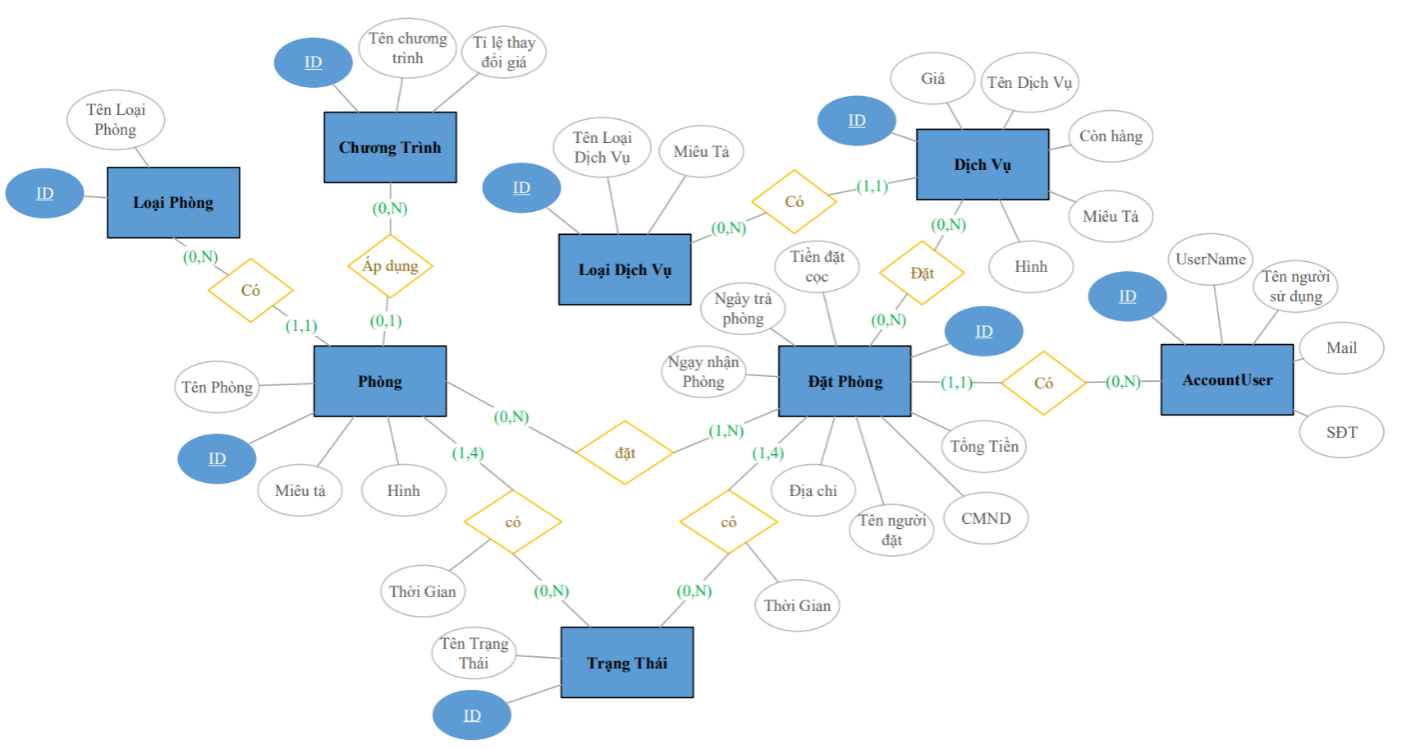




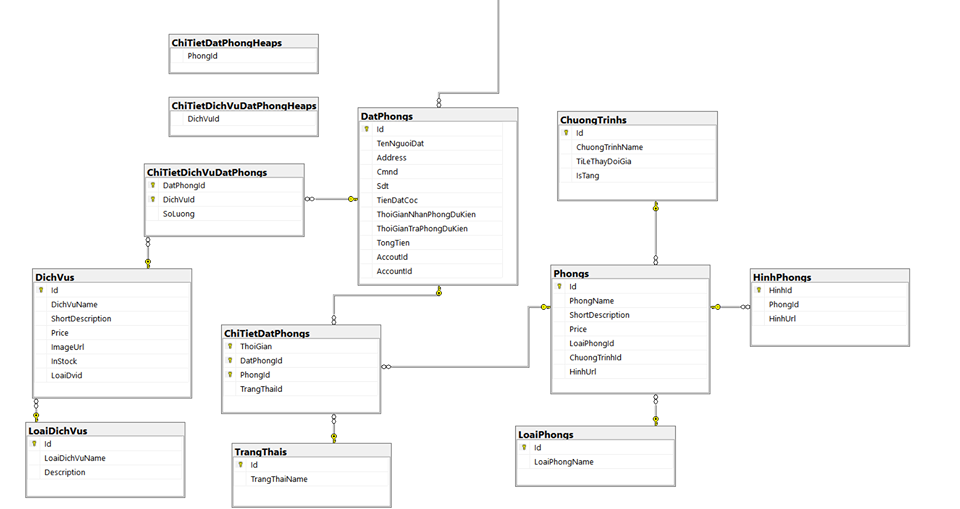
1. **Hủy đặt phòng**



1. Thiết kế cơ sở dữ liệu
2. Mô hình ERD



1. Database diagram



1. Phân tích chi tiết thực thể

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Thực thể** | **TT** | **Thuộc tính** | **Kiểu dữ liệu** | **Mục đích** |
| LoaiPhongs | 1 | Id | Int not null | Lưu mã của từng loại phòng |
| 2 | Name | Nvarchar(max) not null | Lưu tên của từng loại phòng |
| Phongs | 1 | Id | Int not null | Lưu mã của từng phòng |
| 2 | Name | Nvarchar(max) not null | Lưu tên phòng |
| 3 | ShortDescription | Nvarchar(max) null | Mô tả ngắn về phòng |
| 4 | Price | Money not null | Định giá tiền thuê của phòng |
| 5 | LoaiPhongId | FK Int null | Cho biết phòng thuộc loại phòng nào |
| 6 | ChuongTrinhId | Fk int null | Chương trình khuyến mãi đang được áp dụng |
| 7 | HinhUrl | Nvarchar(max) | Hình ảnh của phòng |
| HinhPhongs | 1 | HinhId | Int not null | Mã của mỗi hình |
| 2 | PhongId | FK int not null | Mã của phòng tương ứng |
| 3 | HinhUrl | Nvarchar(max) | Hình ảnh của phòng |
| ChuongTrinhs | 1 | Id | Int not null | Mã của từng chương trình |
| 2 | TenChuongTrinh | Nvarchar(max) | Lưu trữ tên của chương trình |
| 3 | TiLeThayDoiGia | Int not null | Tỉ lệ thay đổi giá của mỗi chương trình |
| 4 | IsTang | Bit | Trả về 1 nếu giá tăng, 0 nêu không tăng |
| LoaiDichVus | 1 | Id | Int not null | Lưu mã của từng loại dịch vụ |
| 2 | Name | Nvarchar(max) not null | Lưu tên loại dịch vụ |
| 3 | Description | Nvarchar(max) null | Lưu mô tả của loại dịch vụ (nếu có) |
| DichVus | 1 | Id | Int not null | Lưu mã của từng dịch vụ |
| 2 | Name | Nvarchar(max) not null | Lưu tên dịch vụ |
| 3 | ShortDescription | Nvarchar(max) null | Mô tả ngắn về dịch vụ (nếu có) |
| 4 | Price | Money not null | Định giá tiền của từng dịch vụ |
| 5 | ImageUrl | Nvarchar(max) | Mô tả hình ảnh của dịch vụ |
| 6 | Instock | Int null | Lưu trữ số lượng sản phẩm còn trong kho |
| 7 | LoaiDvid | Int not null | Cho biết dịch vụ thuộc loại dịch vụ nào |
| DatPhongs | 1 | Id | Int not null | Lưu mã của phiếu đặt phòng |
| 2 | TenNguoiDat | Nvarchar(max) | Lưu tên người đặt phòng |
| 3 | Address | Nvarchar(max) | Lưu địa chỉ của người đặt phòng |
| 4 | Cmnd | Nvarchar(max) | Lưu chứng minh nhân dân của người đặt phòng |
| 5 | Sdt | Nvarchar(max) | Lưu số điện thoại của người đặt phòng |
| 6 | TienDatCoc | Money not null | Lưu tiền đặt cọc trước khi nhận phòng |
| 7 | ThoiGianNhanPhongDuKien | datetime2(7) not null | Lưu thời gian nhận phòng dự kiến |
| 8 | ThoiGianTraPhongDuKien | datetime2(7) not null | Lưu thời gian trả phòng dự kiến |
| 9 | TongTien | Money | ???? |
| 10 | AccountId | Int | ???? |
| ChiTietDatPhongs | 1 | DatPhongId | FK Int not null | Lưu mã của phiếu đặt phòng |
| 2 | PhongId | FK Int not null | Lưu mã của phòng được đặt |
| 3 | ThoiGian | datetime2(7) not null | Lưu thời gian thiết lập chi tiết đặt phòng |
| 4 | TrangThaiId | FK Int not null | Lưu trạng thái |
| TrangThais | 1 | Id | Int not null | Lưu mã trạng thái |
| 2 | Name | Nvarchar(max) not null | Lưu tên trạng thái |

1. Giải thích các mối quan hệ
2. Phong và LoaiPhong

* 1 phòng thuộc một loại phòng
* 1 loại phòng có 0 hoặc nhiều phòng

1. Phong và ChuongTrinh

* 1 phòng được áp dụng bởi 0 hoặc 1 chương trình
* 1 chương trình áp dụng cho nhiều phòng

1. DichVu và LoaiDichVu

* 1 dịch vụ thuộc một loại dịch vụ
* 1 loại dịch vụ có không hoăch nhiều dịch vụ

1. Phong và DatPhong

* 1 phòng hay nhiều phòng được đặt bởi một đơn đặt phòng
* 1 đặt phòng thì có một hoặc nhiều phòng

1. DatPhong và DichVu

* 1 đặt phòng có thể có không hoặc nhiều dịch vụ
* Dịch vụ có thể được đặt bởi không nhiều đơn đặt phòng

1. DatPhong và User

* 1 đơn đặt phòng được đặt bởi một nhân viên
* 1 nhân viên được quyền đặt phòng cho không hoặc nhiều khách hàng

1. Phong va TrangThai

* 1 phòng có 1 trong 4 trạng thái
* 0 hoặc nhiều trạng thái được chứa trong nhiều phòng

1. Các hàm trong bảng

Các Procedure căn bản của các bảng

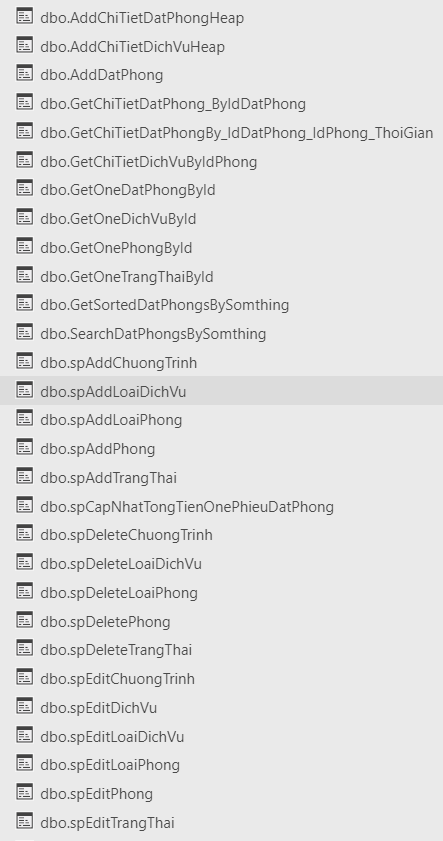
|  |  |  |
| --- | --- | --- |
| **Tên bảng** | **Tên hàm** | **Chức năng** |
| **Phongs** | spGetPhong | Lấy tất cả thông tin của bảng phòng |
| spEditPhong | Dùng để chỉnh sửa thông tin phòng |
| spDeletePhong | Xóa dữ liệu phòng theo id |
| spGetOneByIdPhong | Lấy thông tin theo id phòng |
| spAddPhong | Thêm dữ liệu vào phòng |
| **LoaiPhongs** | spAddLoaiPhong | Thêm dữ liệu vào loại phòng |
| spGetLoaiPhong | Lấy dữ liệu loại phòng |
| spGetOneByIdLoaiPhong | Lấy dữ liệu loại phòng theo id |
| spEditLoaiPhong | Chỉnh sửa loại phòng theo id |
| spDeleteLoaiPhong | Xóa dữ liệu trong bảng loại phòng theo id |
| **LoaiDichVus** | spAddLoaiDichVu | Thêm loại dịch vụ |
| spEditLoaiDichVu | Chỉnh sửa loại dịch vụ theo id |
| spDeleteLoaiDichVu | Xóa theo id |
| spGetLoaiDichVu | Lấy thông tin của bảng |
| spGetOneByIdLoaiDichVu | Lấy một thông tin của bảng theo id |
| **ChuongTrinh** | spAddChuongTrinh | Thêm dữ liệu vào chương trình áp dụng |
| spEditChuongTrinh | Chỉnh sửa thông tin chương trình |
| spDeleteChuongTrinh | Xóa dữ liệu của một chương trình |
| spGetChuongTrinh | Lấy thông tin |
| spGetOneByIdChuongTrinh | Lấy thông tin theo id |
| **DichVus** | spAddDichVu | Thêm dữ liệu vào bảng dịch vụ |
| spEditDichVu | Chỉnh sửa thông tin dịch vụ theo id |
| spDeleteDichVu | Xóa dịch vụ theo id |
| spGetDichVu | Lấy tất cả thông tin của bảng dịch vụ |
| spGetOneByIdDichVu | Lấy một thông tin theo dịch vụ |
| **DatPhongs** | SearchDatPhongsBySomting | Tìm kiếm danh sách đặt phòng theo: CMND, SDT, thời gian |
| GetSortedDatPhongsBySomthing | Sắp xếp danh sách đặt phòng giảm dần |
| GetOneDatPhongById | Lấy danh sách đặt phòng theo id |
| spCapNhatTongTienOnePhieuDatPhong | Cập nhật lại tổng tiền trong đặt phòng khi khách hàng có thêm dịch vụ… |
| **ChiTietDatPhong** | GetChiTietDatPhong | Lấy thông tin chi tiết đặt phòng theo id |
| TimPhongDangDcDat | Tìm kiếm phòng đã được đặt |
| TimPhongLienTuc | Tìm phòng trống |
| **ChiTietDichVuDatPhong** | GetChiTietDichVuByIdPhong | Lấy thông tin chi tiết dịch vụ đặt phòng theo id |

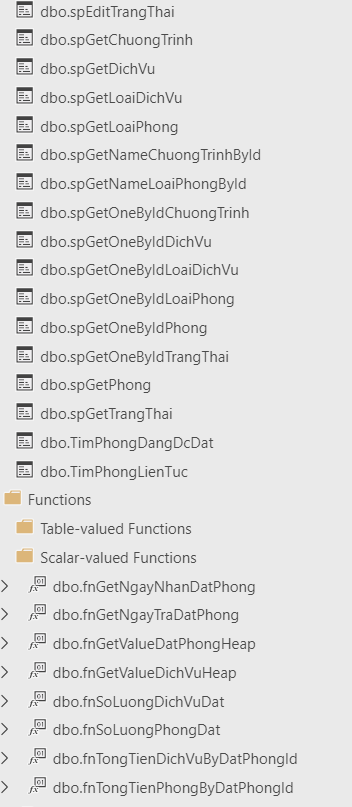
1. Các Function - Proce đặc biệt

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên hàm** | **Chức năng** |
| 1 | fnTongTienDichVuByDatPhongId | Tính tổng tiền của các dịch vụ trong một phòng |
| 2 | fnTongTienPhongByDatPhongId | Tính tiền (gồm phòng ,dịch vụ) |
| 3 | fnGetValueDatPhongHeap(@vitriinput) | Lấy giá trị tại một ví trí trong bảng PhongDatPhongHeap |
| 4 | fnGetValueDichVuHeap(@vitriinput) | Lấy giá trị tại một ví trí trong bảng DichVuDatPhongHeap |
| 5 | **spAddPhieuDatPhong** | Sp này áp dụng transaction,  add phiếu đặt phòng vào bảng đặt phòng.  Cách xây dựng sp này như sau: lưu idphong, iddich vu vào bảng heap, sao đó thực thiện transaction tạo phiếu đặt phòng |
| 6 | spSoLuongPhongDat | Sp này đếm số lượng phòng đặt trong bảng heap |

CHƯƠNG III: HIỆN THỰC HÓA HỆ THỐNG

1. Database (Xem Chi Tiết file SPFinal.sql). Ở đây em chỉ liệt kê một vài cái ạ





|  |  |  |
| --- | --- | --- |
| **Tên bảng** | **Tên hàm** | **Code** |
| **Phongs** | spGetPhong | create proc spGetPhong  as  begin  Select b.Id, b.PhongName, b.ShortDescription, b.Price, a.LoaiPhongName, b.ChuongTrinhId, c.ChuongTrinhName, b.LoaiPhongId, b.HinhUrl  from LoaiPhongs a join Phongs b on a.Id=b.LoaiPhongId  join ChuongTrinhs c on  c.Id=b.ChuongTrinhId  end; |
| spEditPhong | create proc spEditPhong  @id int ,  @Ten nvarchar(50) ,  @Mota nvarchar(max),  @Price money,  @LoaiphongId int,  @ChuongtrinhId int,  @HinhUrl nvarchar(max) = null  as  begin  if(@id !=0 )  update Phongs  set  PhongName=@Ten,  ShortDescription=@Mota,  Price=@Price,  LoaiPhongId=@LoaiphongId,  ChuongTrinhId=@ChuongtrinhId,  HinhUrl=@HinhUrl  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| spDeletePhong | create proc spDeletePhong  @id int =0  as  begin  if(@id !=0)  delete Phongs  from Phongs  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| spGetOneByIdPhong | create proc spGetOneByIdPhong  @id int  as  begin  Select b.Id, b.PhongName, ShortDescription, Price, a.LoaiPhongName, c.ChuongTrinhName, b.ChuongTrinhId, b.LoaiPhongId, b.HinhUrl  from LoaiPhongs a join Phongs b  on a.Id=b.LoaiPhongId  join ChuongTrinhs c  on c.Id=b.ChuongTrinhId  where b.Id=@id  end; |
| spAddPhong | create proc spAddPhong  @Ten nvarchar(50),  @Mota nvarchar(max),  @Price money,  @LoaiphongId int,  @ChuongtrinhId int,  @HinhUrl nvarchar(max) = null  as  begin  insert Phongs  values (@Ten,@Mota,@Price,@LoaiphongId,@ChuongtrinhId,@HinhUrl)  end; |
| **LoaiPhongs** | spAddLoaiPhong | create proc spAddLoaiPhong  @Ten nvarchar(50)  as  begin  insert LoaiPhongs  values (@Ten)  end |
| spGetLoaiPhong | create proc spGetLoaiPhong  as  begin  Select \*  from LoaiPhongs  end; |
| spGetOneByIdLoaiPhong | create proc spGetOneByIdLoaiPhong  @id int  as  begin  Select \*  from LoaiPhongs  where Id=@id  end; |
| spEditLoaiPhong | create proc spEditLoaiPhong  @id int null,  @Ten nvarchar(50)  as  begin  if(@id is not null)  update LoaiPhongs  set LoaiPhongName=@Ten where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| spDeleteLoaiPhong | create proc spDeleteLoaiPhong  @id int null  as  begin  if(@id is not null)  delete LoaiPhongs  from LoaiPhongs  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| **LoaiDichVus** | spAddLoaiDichVu | create proc spAddLoaiDichVu  @Ten nvarchar(50),  @mota nvarchar(max)='notthing'  as  begin  insert LoaiDichVus  values  (@Ten, @mota)  end |
| spEditLoaiDichVu | create proc spEditLoaiDichVu  @id int null,  @Ten nvarchar(50),  @mota nvarchar(max)  as  begin  if(@id is not null)  update LoaiDichVus  set LoaiDichVuName=@Ten,  Description=@mota  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| spDeleteLoaiDichVu | create proc spDeleteLoaiDichVu  @id int null  as  begin  if(@id is not null)  delete LoaiDichVus  from LoaiDichVus  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| spGetLoaiDichVu | create proc spGetLoaiDichVu  as  begin  Select \*  from LoaiDichVus  end; |
| spGetOneByIdLoaiDichVu | create proc spGetOneByIdLoaiDichVu  @id int  as  begin  Select \*  from LoaiDichVus  where Id=@id  end; |
| **ChuongTrinh** | spAddChuongTrinh | create proc spAddChuongTrinh  @Name nvarchar(max),  @Tile int,  @Tang bit  as  begin  insert ChuongTrinhs  values  (@Name, @Tile, @Tang)  end; |
| spEditChuongTrinh | create proc spEditChuongTrinh  @id int null,  @Ten nvarchar(max),  @Tile int,  @Tang bit  as  begin  if(@id is not null)  update ChuongTrinhs  set  ChuongTrinhName=@Ten,  TiLeThayDoiGia=@Tile,  IsTang=@Tang  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| spDeleteChuongTrinh | create proc spDeleteChuongTrinh  @id int null  as  begin  if(@id is not null)  delete ChuongTrinhs  from ChuongTrinhs  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| spGetChuongTrinh | create proc spGetChuongTrinh  as  begin  Select \*  from ChuongTrinhs  end; |
| spGetOneByIdChuongTrinh | create proc spGetOneByIdChuongTrinh  @id int null  as  begin  if @id is not null  Select \*  from ChuongTrinhs  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| **DichVus** | spAddDichVu | create proc spAddDichVu  @Name nvarchar(50),  @Mota nvarchar(500),  @Price money,  @hinh nvarchar(max),  @InStock int,  @LoaiDVuId int  as  begin  insert DichVus  values(  @Name,@Mota,@Price,@hinh,@InStock,@LoaiDVuId)  end; |
| spEditDichVu | create proc spEditDichVu  @id int null,  @Name nvarchar(50),  @Mota nvarchar(500),  @Price money,  @hinh nvarchar(max),  @InStock int,  @LoaiDVuId int  as  begin  if(@id is not null)  update DichVus  set  DichVuName=@Name,  ShortDescription= @mota,  Price=@Price,  ImageUrl =@hinh,  InStock= @InStock,  LoaiDvid=@LoaiDVuId  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| spDeleteDichVu | create proc spDeleteDichVu  @id int null  as  begin  if(@id is not null)  delete DichVus  from DichVus  where Id=@id  else  THROW 50001, 'Nhap sai dieu kien!!!!.', 1;  end; |
| spGetDichVu | create proc spGetDichVu  as  begin  Select a.Id, a.DichVuName, a.ShortDescription, a.Price, a.ImageUrl, a.InStock, a.LoaiDvid  , b.LoaiDichVuName  from DichVus a join LoaiDichVus b  on a.LoaiDvid=b.Id  end; |
| spGetOneByIdDichVu | create proc spGetOneByIdDichVu  @id int  as  begin  Select a.Id, a.DichVuName, a.ShortDescription, a.Price, a.ImageUrl, a.InStock  , b.LoaiDichVuName  from DichVus a join LoaiDichVus b  on a.LoaiDvid=b.Id  where a.Id=@id  end; |
| **DatPhongs** | SearchDatPhongsBySomting | CREATE PROC dbo.SearchDatPhongsBySomthing  @Cmnd as NVARCHAR(255) = NULL,  @Sdt as NVARCHAR(255) = NULL,  @ThoiGianNhanPhongDuKien as date = NULL  AS  declare @date date = CONVERT(VARCHAR(10),@ThoiGianNhanPhongDuKien, 111);  DECLARE @sql AS NVARCHAR(1000);  SET @sql =N'SELECT \* FROM dbo.DatPhongs'  + N' WHERE 1 = 1 '  + CASE WHEN @Cmnd IS NOT NULL THEN  N' AND Cmnd = @Cd' ELSE N'' END  + CASE WHEN @Sdt IS NOT NULL THEN  N' AND Sdt = @St' ELSE N'' END  + CASE WHEN @ThoiGianNhanPhongDuKien IS NOT NULL THEN N' AND ThoiGianNhanPhongDuKien <= @Dt AND @Dt <= ThoiGianTraPhongDuKien ' ELSE N'' END;  EXEC sp\_executesql  @stmt = @sql,  @params = N'@Cd AS NVARCHAR(255), @St AS NVARCHAR(255), @Dt AS DATE',  @Cd = @Cmnd,  @St = @Sdt,  @Dt = @date;  go |
| GetSortedDatPhongsBySomthing | CREATE PROC GetSortedDatPhongsBySomthing  @colname AS sysname,  @TypeSort AS CHAR(1) = 'I'  AS  IF @colname NOT IN(N'TongTien', N'ThoiGianNhanPhongDuKien',N'ThoiGianTraPhongDuKien',N'TenNguoiDat')  THROW 50001,'Column name not supported. Possibly a SQL injection attempt.', 1;  DECLARE @sql AS NVARCHAR(1000);  SET @sql = N'SELECT \* FROM dbo.DatPhongs ORDER BY '  + QUOTENAME(@colname) +  CASE @TypeSort WHEN 'D' THEN N' DESC' ELSE '' END + ';';  EXEC sys.sp\_executesql @stmt = @sql;  GO |
| GetOneDatPhongById | CREATE PROCEDURE dbo.GetOneDatPhongById  @Id int = 0  AS  SELECT \*  FROM DatPhongs  WHERE Id=@Id |
| spCapNhatTongTienOnePhieuDatPhong | create proc spCapNhatTongTienOnePhieuDatPhong  @Id int =0  as  BEGIN  IF(@Id is not null)  UPDATE DatPhongs  SET TongTien = dbo.fnTongTienPhongByDatPhongId(@Id) + dbo.fnTongTienDichVuByDatPhongId(1)  WHERE Id=@Id;  ELSE  THROW 50001, 'không tồn tại id', 1;  END; |
| **ChiTietDatPhong** | GetChiTietDatPhong | CREATE PROCEDURE dbo.GetChiTietDatPhong\_ByIdDatPhong  @Id int = 0;  AS  SELECT \*  FROM ChiTietDatPhongs  WHERE DatPhongId =@Id |
| TimPhongDangDcDat | CREATE PROCEDURE dbo.TimPhongDangDcDat  @ThoiGianNhan DATETIME = NULL,  @ThoiGianTra DATETIME = NULL  AS  -- body of the stored procedure  SELECT distinct PhongId FROM  ChiTietDatPhongs  WHERE @ThoiGianNhan <=ThoiGian and ThoiGian <=@ThoiGianTra |
| TimPhongLienTuc | CREATE PROCEDURE dbo.TimPhongLienTuc  @ThoiGianNhan DATETIME = NULL,  @ThoiGianTra DATETIME = NULL  AS  -- body of the stored procedure  SELECT \* from Phongs WHERE Id not IN  (SELECT distinct PhongId FROM  ChiTietDatPhongs  WHERE @ThoiGianNhan <=ThoiGian and ThoiGian <=@ThoiGianTra) |
| **ChiTietDichVuDatPhong** | GetChiTietDichVuByIdPhong | CREATE PROCEDURE dbo.GetChiTietDichVuByIdPhong  @Id int = 0  AS  SELECT a.DichVuId, a.DatPhongId, a.SoLuong, b.Price, b.DichVuName, TongTien= a.SoLuong\*b.Price  from ChiTietDichVuDatPhongs a, DichVus b  WHERE a.DatPhongId= @Id  and a.DichVuId=b.Id |

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên hàm** | **Code** |
| 1 | fnTongTienDichVuByDatPhongId | IF OBJECT\_ID('fnTongTienDichVuByDatPhongId') IS NOT NULL   DROP FUNCTION fnTongTienDichVuByDatPhongId;   go  create FUNCTION fnTongTienDichVuByDatPhongId( @IdDatPhong int =0)   RETURNS money   BEGIN      RETURN (select sum(TongTien)      from          (SELECT TongTien = (a.SoLuong\*b.Price)          from ChiTietDichVuDatPhongs a, DichVus b          WHERE a.DatPhongId=1              and a.DichVuId=b.Id)c  )  END;  go |
| 2 | fnTongTienPhongByDatPhongId | IF OBJECT\_ID('fnTongTienPhongByDatPhongId') IS NOT NULL   DROP FUNCTION fnTongTienPhongByDatPhongId;   go  create FUNCTION fnTongTienPhongByDatPhongId( @IdDatPhong int =0)   RETURNS money   BEGIN      RETURN ( select sum(b.Price)      from ChiTietDatPhongs a, Phongs  b      where DatPhongId =@IdDatPhong          and a.PhongId=b.Id )  END;  GO |
| 3 | fnGetValueDatPhongHeap(@vitriinput) | IF OBJECT\_ID('fnGetValueDatPhongHeap') IS NOT NULL   DROP FUNCTION fnGetValueDatPhongHeap;   GO  create FUNCTION fnGetValueDatPhongHeap(@Vitriinput int =0)   RETURNS int   BEGIN      RETURN (SELECT PhongId              FROM ChiTietDatPhongHeaps              ORDER BY PhongId              OFFSET @Vitriinput ROWS              FETCH NEXT 1 ROWS ONLY)  END;  go |
| 4 | fnGetValueDichVuHeap(@vitriinput) | IF OBJECT\_ID('fnGetValueDichVuHeap') IS NOT NULL   DROP FUNCTION fnGetValueDichVuHeap;   GO  create FUNCTION fnGetValueDichVuHeap(@Vitriinput int =0)   RETURNS int   BEGIN      RETURN (SELECT DichVuId              FROM ChiTietDichVuDatPhongHeaps              ORDER BY DichVuId              OFFSET @Vitriinput ROWS              FETCH NEXT 1 ROWS ONLY)  END;  go |
| 5 | **spAddPhieuDatPhong** | declare @Tenkh nvarchar(max);  declare @DiaChi nvarchar(max);  declare @CMND nvarchar(max);  declare @SDT nvarchar(max);  declare @TienCoc nvarchar(max);  declare @TongTien money;  declare @account int;  declare @AccountId int;  declare @iddatphong int = 0;  declare @ThoiGian datetime2(7) ;  declare @TrangThai int ;  declare @NgayNhan datetime2(7) ='2019-12-15';  declare @NgayTra datetime2(7)='2019-12-17';  --declare @NgayTra datetime2(7);  begin try      begin tran          insert DatPhongs          values ('T','as','13','13231',10000,@NgayNhan,@NgayTra,100000,null,null);          set @iddatphong=@@IDENTITY          DECLARE @soluongPhong int;          EXEC    @soluongPhong =  dbo.SoLuongPhongDat          DECLARE @soluongDichVu int;          EXEC    @soluongDichVu =  dbo.SoLuongDichVuDat            DECLARE @value int =0;  -- giá trị tại vị trí của mảng          declare @vitrii int = 0;  -- vị trí bắt đầu của mảng          -- add phong          -- dem so luong phong can add          ---vd có 2 phòng:          --- cần add phòng 1,2 cùng 1 ngày          --      add tiếp 1,2 ngày tiếp theo          while (@NgayNhan <=@NgayTra)          begin              WHILE (@vitrii <@soluongPhong )              BEGIN                  EXEC @value =  fnGetValueDatPhongHeap @Vitriinput=@vitrii                  insert ChiTietDatPhongs                  values (@NgayNhan,@iddatphong,@value,1);                  --Tăng vị trí lên 1                  set @vitrii=@vitrii+1;              end              --reset vitri về 0              set @vitrii=0;              set @NgayNhan=(DATEADD(day, 1, @NgayNhan))          end          -- sau khi add Phong thì RemoveHeap            -- add dich vu          WHILE (@vitrii <@soluongDichVu )          BEGIN              BEGIN                  EXEC @value =  fnGetValueDichVuHeap @Vitriinput=@vitrii                  insert ChiTietDichVuDatPhongs                  values (@iddatphong,@value,1);                  --Tăng vị trí lên 1                  set @vitrii=@vitrii+1;              end              --reset vitri về 0              set @vitrii=0;          END          -- sau khi add DichVu thì RemoveHeap      commit tran      DELETE FROM ChiTietDatPhongHeaps      DELETE FROM ChiTietDichVuDatPhongHeaps      print 'transaction committed';  end try  begin catch   print 'error when inserting, rolling back transaction';   rollback tran;  end catch |
| 6 | spSoLuongPhongDat | IF EXISTS (  SELECT \*      FROM INFORMATION\_SCHEMA.ROUTINES  WHERE SPECIFIC\_SCHEMA = N'dbo'      AND SPECIFIC\_NAME = N'spSoLuongPhongDat'      AND ROUTINE\_TYPE = N'PROCEDURE'  )  DROP PROCEDURE dbo.spSoLuongPhongDat  GO  -- Create the stored procedure in the specified schema  CREATE PROCEDURE dbo.spSoLuongPhongDat  AS      select count(PhongId)[SoLuongP] from ChiTietDatPhongHeaps  GO |

CHƯƠNG IV: KẾT LUẬN

1. Ưu điểm

Khách hàng có thể tìm kiếm phòng một cách thuận lợi mà không tốn bất kì chi phi nào

Việc sửa đổi, tìm kiếm, cập nhật dữ liệu được thực hiện một cách nhanh chóng đáp ứng cho yêu cầu đặt, nhận và thanh toán phòng cho khách hàng

1. Nhược điểm

Những hạn ở phần cài đặt chương trình thử nghiệm, do thời gian không nhiều, trình độ chuyên môn của em chưa cao nên chưa giải quyết được trọn vẹn những vấn đề nảy sinh trong quá trình thực hiện đặt phòng và phần xử lý dữ liệu vẫn chưa đạt kết quả mong muốn.

1. Hướng phát triển

Hướng phát triển của web site tiến đến việc khách hàng tự mình đặt phòng theo ý muốn của mình và thương mại điện tử với đầy đủ các chức năng thanh toán tiền qua mạng