**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH  
BỘ MÔN CÔNG NGHỆ THÔNG TIN**

A blue and yellow logo

Description automatically generated with low confidence

**BÁO CÁO BÀI TẬP LỚN**

**PHÂN TÍCH THIẾT KẾ HƯỚNG ĐỐI TƯỢNG**

**ĐỀ TÀI: XÂY DỰNG WEB APPLICATION**

**QUẢN LÝ NHÂN SỰ**

Nhóm sinh viên thực hiện:

|  |  |
| --- | --- |
| **Họ và tên** | **Mã số sinh viên** |
| 1.Nguyễn Phạm Thành Hưng | 6151071056 |
| 2.Phạm Nguyễn Thùy Dung | 6151071003 |
| 3.Trần Thiên Phú | 6151071083 |
| 4.Nguyễn Thị Quỳnh Quyên | 6151071091 |

**Giảng viên hướng dẫn:** ThS. Phạm Thị Miên

Thành phố Hồ Chí Minh, tháng 11, năm 2023

# LỜI NÓI ĐẦU

Trải qua khoảng thời gian học tập môn Phân tích thiết kế hướng đối tượng trong học kỳ I năm 2023-2024 vừa qua kèm những kiến thức đã học từ các môn trước như Phân Tích Thiết Kế Hệ Thống , Thiết Kế Cơ Sở Dữ Liệu, Lập Trình Web,… Nhóm em đã vận dụng những kiến thức đã được học để xây dựng nên trang web application Quản Lý Nhân Sự.

Để hoàn thành đề tài này trước hết nhóm em xin gửi đến quý thầy cô Bộ môn Công Nghệ Thông Tin – Trường Đại học Giao Thông Vận Tải Phân hiệu Thành Phố Hồ Chí Minh nói chung và cô Phạm Thị Miên nói riêng, lời cảm ơn chân thành vì đã tạo điều kiện cho chúng em có môi trường học tập vận dụng tốt. Đặc biệt chúng em xin gửi đến cô Phạm Thị Miên lời cảm sâu sắc vì đã tận tình hướng dẫn, truyền đạt kiến thức để chúng em có thể hoàn thành đề tài môn Phân tích thiết kế hướng đối tượng.

Vì thời gian làm đề tài này có hạn cũng như hiểu biết cá nhân em còn hạn chế, chúng em cũng đã nỗ lực hết sức để hoàn thành đề tài này một cách tốt nhất, nhưng chắc chắn vẫn sẽ có những thiếu sót không thể tránh khỏi. Nhóm em kính mong nhận được sự thông cảm và những ý kiến đóng góp chân thành từ quý thầy cô để em có thể rút kinh nghiệm, hoàn thiện hơn.

Sau cùng, nhóm em xin kính chúc cô Phạm Thị Miên nhiều sức khỏe hạnh phúc và thành công trong công việc và trong cuộc sống.

Trân trọng cảm ơn!

**MỤC LỤC**

[LỜI NÓI ĐẦU i](#_Toc138955896)

**CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI...............................................................................2**

1.1. Lý do chọn đề tài:.....................................................................................................2

[1.2.Mục tiêu nghiên cứu: 2](#_Toc138955899)

[1.3.Phương pháp nghiên cứu: 2](#_Toc138955900)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 3](#_Toc138955901)

[2.1. ASP.Net MVC 3](#_Toc138955902)

[2.2.Mô Hình MVC : 4](#_Toc138955903)

[2.3.Entity Framework 5](#_Toc138955904)

[2.4.SQL Server 7](#_Toc138955905)

[2.5.Thư viện Bootstrap 8](#_Toc138955906)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 9](#_Toc138955907)

[3.1.Mô tả hệ thống: 9](#_Toc138955908)

[3.2.Sơ đồ Phân cấp chức năng(BFD): 1](#_Toc138955909)2

[3.3.Sơ đồ Class: 13](#_Toc138955910)

[3.4. Sơ đồ Usecase: 1](#_Toc138955920)3

[3.5. Mô hình ER: 18](#_Toc138955921)

[3.6. Sơ đồ Luồng dữ liệu (DFD): 19](#_Toc138955922)

[3.7. Sơ đồ Luồng dữ liệu chi tiết: 2](#_Toc138955923)0

[3.8. Sơ đồ Tuần tự: 2](#_Toc138955923)4

[3.7. Sơ đồ Hoạt động: 2](#_Toc138955923)9

[CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH 3](#_Toc138955934)4

[4.1. Giao diện Đăng nhập: 3](#_Toc138955935)4

[4.2. Giao diện Trang chủ: 3](#_Toc138955935)4

[4.3. Giao diện Chấm công: 3](#_Toc138955935)5

[4.4. Giao diện Danh sách chấm công: 3](#_Toc138955935)5

[4.5. Giao diện Thêm nhân viên: 3](#_Toc138955935)6

[4.6. Giao diện Sửa nhân viên: 3](#_Toc138955935)6

[4.7. Giao diện Xóa nhân viên: 3](#_Toc138955935)7

[4.8. Giao diện Danh sách hợp đồng: 3](#_Toc138955935)7

[4.9. Giao diện Lương nhân viên: 3](#_Toc138955935)8

[4.10. Giao diện Danh sách phòng ban: 3](#_Toc138955935)8

[4.11. Giao diện Thêm phòng ban: 3](#_Toc138955935)9

[4.12. Giao diện Sửa phòng ban: 3](#_Toc138955935)9

[4.13. Giao diện Xóa phòng ban: 40](#_Toc138955935)

[4.14. Giao diện Danh sách chức vụ: 40](#_Toc138955935)

[4.15. Giao diện Thêm chức vụ: 4](#_Toc138955935)1

[4.16. Giao diện Sửa chức vụ: 4](#_Toc138955935)1

[4.17. Giao diện Xóa chức vụ: 4](#_Toc138955935)2

[CHƯƠNG 5: KẾT LUẬN VÀ TỔNG KẾT 4](#_Toc138955946)3

[5.1.Kết quả đạt được: 4](#_Toc138955947)3

[5.2.Hạn chế: 4](#_Toc138955948)3

[5.3.Hướng phát triển: 4](#_Toc138955949)3

[5.4.Tài liệu tham khảo: 4](#_Toc138955950)3

CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI

1.1 Lí do chọn đề tài

Nhân lực là tài sản quan trọng nhất của một doanh nghiệp. Cách quản lý nhân sự của doanh nghiệp quyết định tính hiệu quả và thành công của doanh nghiệp. Quản lý nhân sự bao gồm quản lý nhân viên, chấm công, quản lý tiền lương và liên kết các chính sách nhân sự với mục tiêu kinh doanh. Chủ doanh nghiệp cần truyền đạt mục tiêu kinh doanh cho tất cả nhân viên trong doanh nghiệp để đạt được mục tiêu chiến lược. Quản lý nhân sự là một hoạt động nhằm đạt được mục tiêu chiến lược của doanh nghiệp. Đề tài hướng đến việc quản lý thông tin nhân viên và tính lương cho các nhân viên trong công ty. Đối tượng nghiên cứu bao gồm các trang về hình thức tính lương, hệ số mức đóng bảo hiểm và thông tin chi tiết của các đối tượng cần quản lý.

Phạm vi và đối tượng nghiên cứu bao gồm các trang về hình thức tính lương, hệ số mức đóng bảo hiểm, cũng như tham khảo thêm về các thông tin chi tiết các đối tượng cần quản lý để mang đến người dùng môi trường phù hợp và thực tiễn.

1.2Mục tiêu nghiên cứu

Xây dựng được website quản lý nhân sự đảm bảo các chức năng cơ bản của việc quản lý, giúp công việc quản lý được thuận tiện và chuyên nghiệp hơn.

Giao diện trang web tối giản , dễ sử dụng, thân thiện với người dùng

1.3Phương pháp nghiên cứu

Trong đề tài này, nhóm em sử dụng kết hợp một số phương pháp nghiên cứu: Phương pháp định lượng, phương pháp thực nghiệm khảo sát, phương pháp phân tích và tổng hợp.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 ASP.Net MVC

ASP.Net MVC là một framework phát triển ứng dụng web được thiết kế bởi Microsoft, được xây dựng trên nền tảng .Net Framework. Nó cho phép lập trình viên phát triển ứng dụng web theo mô hình kiến trúc MVC (Model – View – Controller), là một trong những kiến trúc phát triển ứng dụng web phổ biến nhất hiện nay.

ASP.Net MVC sử dụng định tuyến (routing) để xác định cách xử lý các yêu cầu từ truy cập đến trang web. Khi một yêu cầu được chuyển tới một địa chỉ URL cụ thể, ASP.Net MVC sử dụng các định tuyến để phân loại yêu cầu và chuyển nó tới Controller tương ứng để xử lý.

Các định tuyến giúp cho ASP.Net MVC xác định URL của yêu cầu được giải quyết bởi Controller nào trong ứng dụng. Khi một yêu cầu được gửi đến trang web, nó sẽ được so sánh với một trong những định tuyến có sẵn trong ứng dụng theo thứ tự đã đăng ký nếu khớp nó sẽ đưa đến Controller tương ứng để xử lý yêu cầu.

Ví dụ, với định tuyến mặc định cho một ứng dụng ASP.Net MVC, khi một yêu cầu đến từ địa chỉ URL “/Home/Index”, nó sẽ được xử lý bởi Controller “HomeController” với phương thức “Index”. Các định tuyến có thể được xác định bằng cách thêm các tệp định tuyến vào thư mục App\_Start của ứng dụng.

Định tuyến giúp cho các ứng dụng ASP.Net MVC quản lý yêu cầu người dùng một cách dễ dàng và hiệu quả. Nó cung cấp cho các lập trình viên khả năng tùy chỉnh các URL, tạo URL thân thiện với các công cụ tìm kiếm và giúp dễ dàng thay đổi hoặc xóa các yêu cầu trong trường hợp cần thiết. Để sử dụng định tuyến, lập trình viên cần phải đăng ký các tệp định tuyến cho ứng dụng ASP.Net MVC.

ASP.Net MVC cung cấp các lớp hỗ trợ cho xử lý dữ liệu, bao gồm các ORM (Object-Relational Mapping) như Entity Framework cho phép tạo các tin chủ các đối tượng trong ứng dụng. Ngoài ra, ASP.Net MVC hỗ trợ một số tính năng mạng khác như nhận dạng và xác thực người dùng, phân quyền, và lưu trữ các thông tin session.

Một điểm khác biệt quan trọng của ASP.Net MVC so với những Framework khác là quản lý các tập tin mã nguồn. Với ASP.Net MVC, các tài nguyên động (các tập tin JavaScript và CSS, hình ảnh, v.v.) được quản lý và tổ chức theo cấu trúc thư mục riêng và được liên kết giữa các tập tin Mã nguồn, giúp cho các lập trình viên quản lý tập trung và kiểm soát các tài nguyên động.

Trong .Net MVC cho phép sử dụng các hàm Helper để tạo ra các thành phần trên giao diện như các form input, button, hay các menu ,… Các hàm Helper này giúp cho quá trình tạo giao diện trở nên dễ dàng và nhanh chóng hơn , tránh việc phải viết mã HTML một cách đầy đủ. Các hàm Helper trong .Net MVC có thể được sử dụng trong các file Razor view được gọi bằng cách sử dụng ký tự ‘@’ và các từ khóa cú pháp tương ứng.

Ví dụ: @Html.TextBox(“Name”)

@Html.ActionLink(“Trang chủ”, “Index”)

Trong đó : Hàm TextBox sử dụng để tạo ra một TextBox , hàm ActionLink tạo ra một Hyperlink với chữ “Trang chủ”, sử dụng đường dẫn đến Action “Index” của Controller hiện tại.

Các Helper là một trong những tính năng quan trọng và mạnh mẽ của .Net MVC, giúp cho việc phát triển ứng dụng trở nên dễ dàng và nhanh chóng hơn.

Tóm lại, ASP.Net MVC là một công nghệ phát triển ứng dụng web đáng được chú ý với mô hình kiến trúc MVC, cung cấp các tính năng mạnh mẽ và nhiều tiện ích hỗ trợ giúp lập trình viên viết mã nguồn ứng dụng web nhanh chóng và dễ dàng.

2.2 Mô Hình MVC

Mô hình MVC (Model – View – Controller) là một mô hình thiết kết phần mềm được sử dụng phổ biến trong nhiều ứng dụng web và trong .Net MVC. Mô hình này giúp cho ứng dụng được phân tầng một cách rõ ràng, giúp cho quá trình phát triển và bảo trì ứng dụng được dễ dàng hơn.

MVC gồm ba thành phần chính là Model, View và Controller:

Model: đại diện cho dữ liệu và thao tác xử lý dữ liệu trong ứng dụng.

Model thường sử dụng các lớp (classes) để mô tả cấu trúc của dữ liệu cũng như để thực hiện các chức năng thao tác dữ liệu bao gồm lấy dữ liệu từ cơ sở dữ liệu, thêm, sửa, xóa dữ liệu.

View: đại diện cho giao diện người dùng của ứng dụng, chịu trách nhiệm hiển thị dữ liệu cho người dùng và tương tác với họ. View thường được viết bằng HTML, CSS và JavaScript.

Controller: liên kết giữa Model và View, điều khiển các yêu cầu của người dùng và xử lý các thao tác trên dữ liệu dựa trên yêu cầu của họ. Controller thường thực hiện định tuyến (routing) yêu cầu và gọi các hàm của Model để truy vấn hoặc thao tác với dữ liệu sau đó truyền dữ liệu đó cho View để hiển thị cho người dùng.

Trong .Net MVC, tất cả các yêu cầu đến ứng dụng đều được định tuyến đến một Action trong Controller để xử lý. Controller sau đó sẽ gọi các phương thức của Model để truy vấn hoặc thao tác với dữ liệu , sau đó trả về dữ liệu để cho View hiển thị. Ngoài ra, .Net MVC cho phép sử dụng các hàm Helper để tạo ra các thành phần trên giao diện như các form input, button, hay các menu giúp cho quá trình tạo giao diện trở nên dễ dàng hơn.

2.3 Entity Framework

Entity Framework là một công cụ hỗ trợ rất mạnh mẽ trong việc phát triển các ứng dụng .NET MVC liên quan đến cơ sở dữ liệu. Nó cho phép lập trình viên tạo ra các ứng dụng ở một mức độ trừu tượng cao hơn, giảm thiểu sự phụ thuộc vào cơ sở dữ liệu và tạo ra các đơn vị độc lập trong thiết kế ứng dụng.

Entity Framework sử dụng mô hình ORM (Object-Relational Mapping) để tạo đối tượng và đối tượng hóa cơ sở dữ liệu. Nó cho phép các đối tượng trong ứng dụng trực tiếp tương tác với cơ sở dữ liệu, mà không phải sử dụng ngôn ngữ truy vấn cơ sở dữ liệu truyền thống.

Trong ứng dụng .NET MVC, Entity Framework được tích hợp vào kiến trúc của mô hình 3 lớp. Lớp Mô hình đại diện cho các đối tượng chứa dữ liệu trong ứng dụng, bao gồm các lớp đối tượng được sử dụng để tương tác với cơ sở dữ liệu. Lớp Điều khiển thực hiện các xử lý logic bên trong ứng dụng, trong khi Lớp Chế độ xem làm việc với dữ liệu và thực hiện hiển thị cho người dùng. Entity Framework cho phép các lớp mô hình và đối tượng được sử dụng trong các lớp điều khiển và chế độ xem để tương tác với cơ sở dữ liệu.

Có hai cách để tạo kết nối với cơ sở dữ liệu sử dụng Entity Framework: First Approach Code-First và Second Approach Database-First. Với First Approach, các lớp đối tượng được xác định trước và sau đó cơ sở dữ liệu được tạo ra từ các lớp đối tượng đó. Với Second Approach, các bảng cơ sở dữ liệu được xác định trước và các lớp đối tượng được tạo ra từ các bảng đó.

Entity Framework cho phép lập trình viên tạo ra truy vấn LINQ (Language Integrated Query) để truy xuất và thay đổi dữ liệu trong cơ sở dữ liệu. Nó cũng cho phép lập trình viên sử dụng các kỹ thuật Eager Loading, Lazy Loading và Explicit Loading để tối ưu hóa hiệu suất truy vấn cơ sở dữ liệu.

Entity Framework cũng hỗ trợ nhiều tính năng khác nhau, bao gồm kết nối với nhiều cơ sở dữ liệu, xử lý lỗi, sử dụng Trực quan hóa Entity Framework tích hợp trong Visual Studio và hỗ trợ các tiện ích được cung cấp bởi bộ công cụ của Entity Framework.

Tóm lại, Entity Framework cho phép lập trình viên tạo những ứng dụng .NET MVC ở mức độ trừu tượng cao hơn, giảm thiểu sự phụ thuộc vào cơ sở dữ liệu và tạo ra các đơn vị độc lập trong thiết kế ứng dụng. Nó cung cấp hàng loạt tính năng hỗ trợ cho việc xử lý dữ liệu và truy xuất cơ sở dữ liệu trong ứng dụng .NET MVC, giúp tăng tính đáp ứng và hiệu suất của ứng dụng.

2.4 Hệ quản trị cơ sở dữ liệu SQL Server

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ do Microsoft phát triển. Nó cung cấp các công cụ và khả năng để quản lý và lưu trữ dữ liệu, cũng như cho phép thực hiện các thao tác truy vấn dữ liệu dễ dàng một cách hiệu quả. Dưới đây là một số khái niệm cơ bản trong SQL Server:

Database: là một tập hợp các đối tượng như bảng, view, procedure, trigger , constraint và cấu trúc lưu trữ khác.

Table: là một đối tượng chính trong database, chứa các dòng và cột để lưu trữ dữ liệu. Mỗi table sẽ có một tập hợp các thuộc tính như tên table, các cột, các ràng buộc (constraint)

Column (cột): mỗi cột trong bảng xác định kiểu dữ liệu mà nó chứa. Các kiểu dữ liệu bao gồm string , int , date, v.v… Ngoài ra, mỗi cột còn có các thuộc tính khác như khóa chính (primary key), khóa ngoại (foreign key), giá trị mặc định (default value) và ràng buộc (constraint).

Primary key (khóa chính): là một cột hoặc một tập hợp các cột trong bảng, được dùng để xác định duy nhất một bản ghi trong bảng. Khóa chính được sử dụng để liên kết các bảng với nhau thông qua khóa ngoại.

Foreign key (khóa ngoại): là một cột trong bảng, tham chiếu đến khóa chính của một bảng khác. Khóa ngoại được sử dụng để xác định quan hệ giữa hai bảng và giúp kết nối chúng với nhau.

SQL Server còn cung cấp nhiều tính năng và công cụ khác để thao tác với cơ sở dữ liệu bao gồm các lệnh SQL như SELECT, INSERT, UPDATE, DELETE và các Stored Procedure, View, Trigger , Constraint ,… Nó cũng hỗ trợ các tính năng quản lý cơ sở dữ liệu như Backup, Restore, Replication và log shipping.

2.5 Thư viện Bootstrap

Bootstrap là một thư viện CSS và JavaScript được phát triển bởi Twitter, cung cấp cho các nhà phát triển một bộ công cụ để thiết kế và xây dựng các giao diện web đáp ứng và đẹp mắt một cách nhanh chóng và dễ dàng. Bootstrap không chỉ là một thư viện CSS, mà nó còn cung cấp những yếu tố JavaScript để tạo ra các hiệu ứng tương tác cho trang web.

Bootstrap là một trong những thư viện Front-end phổ biến nhất trên thế giới được sử dụng bởi hàng triệu lập trình viên để phát triển các ứng dụng web, trang web và các dự án khác. Bootstrap rất đa năng và cá nhân hóa cho phép các nhà phát triển tùy chỉnh các thành phần của thư viện sao cho phù hợp với nhu cầu của mình.

Bootstrap bao gồm nhiều thành phần khác nhau có thể sử dụng để tạo ra những đoạn mã HTML đẹp và đáp ứng với nhu cầu cơ bản đề ra. Một số thành phần của thư viện Bootstrap bao gồm:

Grid: cung cấp hệ thống lưới để phân chia trang web thành các cột và hàng, giúp bạn chia sẻ các nội dung trên trang dễ dàng và hiệu quả từ phiên bản trang web cho PC đến các ứng dụng trên điện thoại di động.

Form: cung cấp các khung cho người dùng để nhập dữ liệu vào các trường thông tin và nút gửi.

Button: cung cấp một loạt nút để thêm tính năng tương tác cho người dùng.

Navbar: cung cấp một thanh điều hướng cho phép người dùng điều hương qua các trang web của mình một cách dễ dàng nhanh chóng.

Modal: cung cấp một cửa sổ nổi lên trên web để thông báo cho người dùng về các thông tin quan trọng và yêu cầu sự xác nhận của họ.

Thư viện này còn rất nhiều các thành phần khác để giúp nhà phát triển tạo ra các trang web đẹp và tương tác nhanh chóng, dễ dàng với người dùng.

# **CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

**3.1 Mô tả hệ thống**

Web Application quản lý nhân sự cho phép người dùng có thể quản lí nhân viên của mình.

* **Yêu cầu chức năng**

- Đăng nhập:

+ Người dùng phải đăng nhập để có thể sử dụng các chức năng của web.

+ Tài khoản của nhân viên sẽ do admin cung cấp

- Thông tin Nhân Viên

+ Nhân viên có thể xem thông tin của nhân viên như ID, họ tên, ngày sinh, giới tính, email,… Nhân viên có thể chỉnh sửa thông tin của mình.

+ Admin quyền xem, thêm, sửa, xoá thông tin của Nhân Viên

- Thông tin Phòng Ban:

+ Admin quyền xem, thêm, chỉnh sửa, xóa thông tin Phòng Ban (Đảm bảo không có nhân viên nào thuộc phòng ban muốn xóa)

- Thông tin Chức Vụ:

+ Admin có thể xem, thêm ,chỉnh sửa, xóa thông tin Chức Vụ (Đảm bảo không có nhân viên nào thuộc chức vụ muốn xóa)

- Thông tin Hợp Đồng:

+ Nhân viên có thể xem các thông tin Hợp Đồng của mình như tên hợp đồng, ngày bắt đầu hợp đồng, ngày kết thúc hợp đồng, họ tên của nhân viên

+ Admin có quyền thêm, chỉnh sửa, xoá Hợp Đồng

- Thông tin Chấm Công:

+ Nhân viên có thể xem thông tin về Chấm Công như Ngày làm, có vi phạm hay không

+ Admin có thể xem thông tin Chấm Công, sửa, xóa thông tin Chấm Công

- Thông tin Tính lương:

+ Lương có công thức tính riêng

- Thông tin Lương:

+ Admin thao tác để tạo ra bảng Lương Cho các Nhân Viên

* **Yêu cầu phi chức năng**

- Giao diện website bán hàng phải đơn giản, thân thiện với người dùng, tạo cho

khách hàng cảm giác thích thú mà muốn khám phá ngay lần đầu tiên truy cập vào

website.

- Giao diện phải đảm bảo không bị vỡ khi thay đổi kích thước của trang web.

- Màu sắc không được quá bắt mắt, chữ viết phải dễ nhìn, dễ đọc

- Bố cục trang web phải gần gũi, phù hợp

* **Yêu cầu khác**

- Trang web phải đơn giản, dễ sử dụng, hoạt động ổn định

- Tốc độ tải dữ liệu và tốc độ mở trang web không được quá chậm, tránh việc

gây khó chịu cho người dùng.

- Chạy ổn định trên các trình duyệt phổ biến: Firefox, Cốc Cốc, Chrome…

* **Yêu cầu người dùng**

- Đối với tất cả người dùng:

+ Cần có thiết bị có kết nối internet để truy cập website

- Đối với nhân viên:

+ Cần có tài khoản được admin cấp để truy cập các tính năng dành cho nhân viên

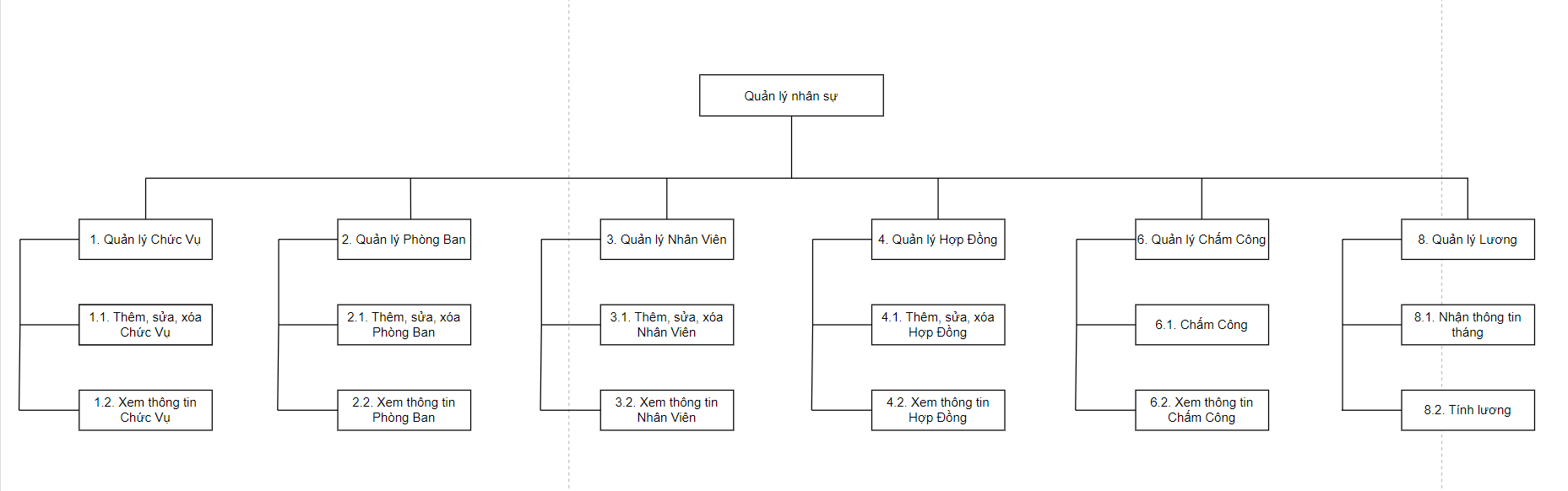
- Đối với quản trị viên:

+ Cần có tài khoản được cấp quyền admin để truy cập hệ thống.

3.2 Sơ đồ phân cấp chức năng (BFD)

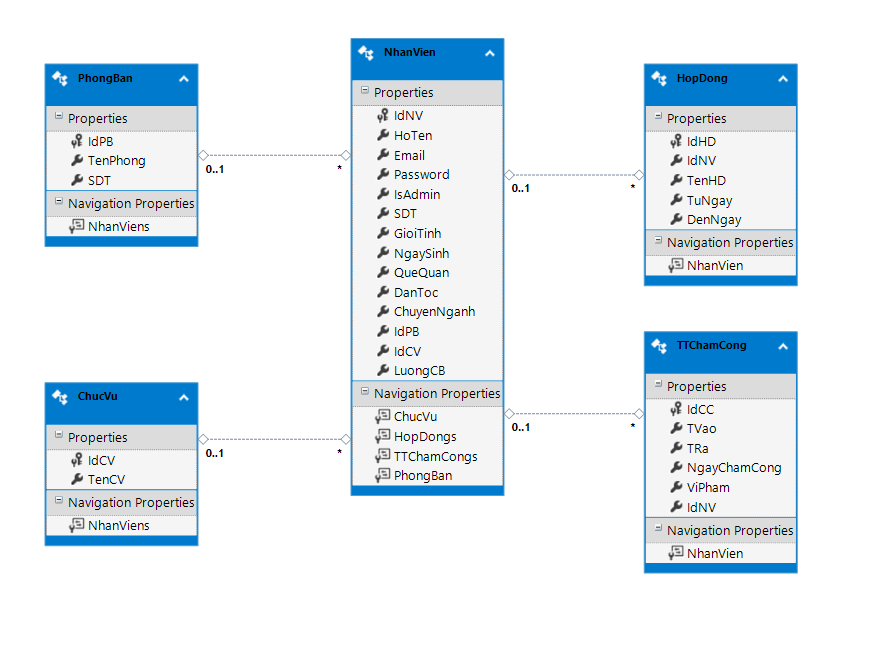
Sau khi khảo sát và đánh giá một số website quản lý nhân sự, nhóm đã phân tích

Quy trình hoạt động của những website đó và đưa ra được sơ đồ phân rã chức năng sau:



*Hình 3.1 Sơ đồ phân cấp chức năng*

3.3 Sơ đồ Class



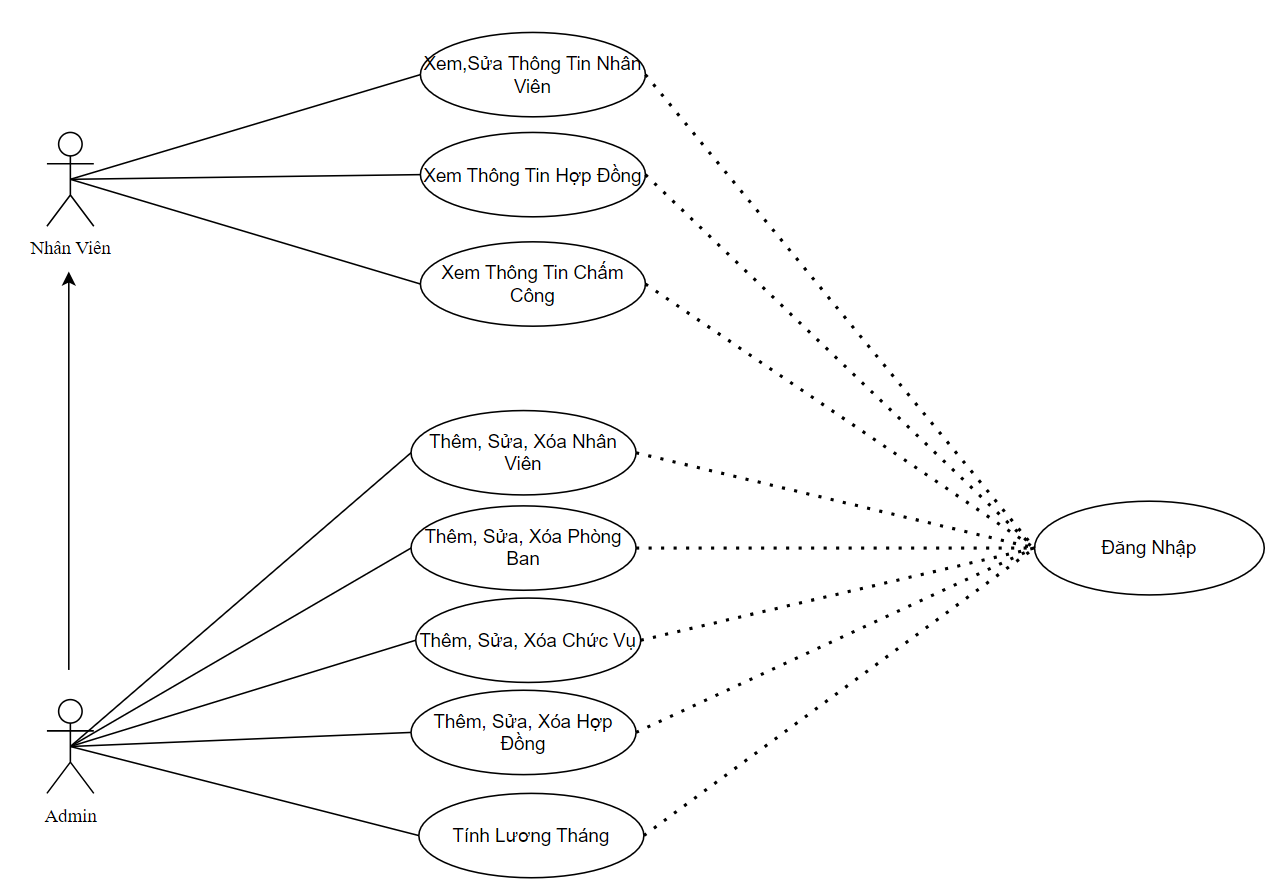
*Hình 3.2 Sơ đồ Class hệ thống*

3.4 Sơ đồ UseCase

3.4.1 Danh sách các tác nhân

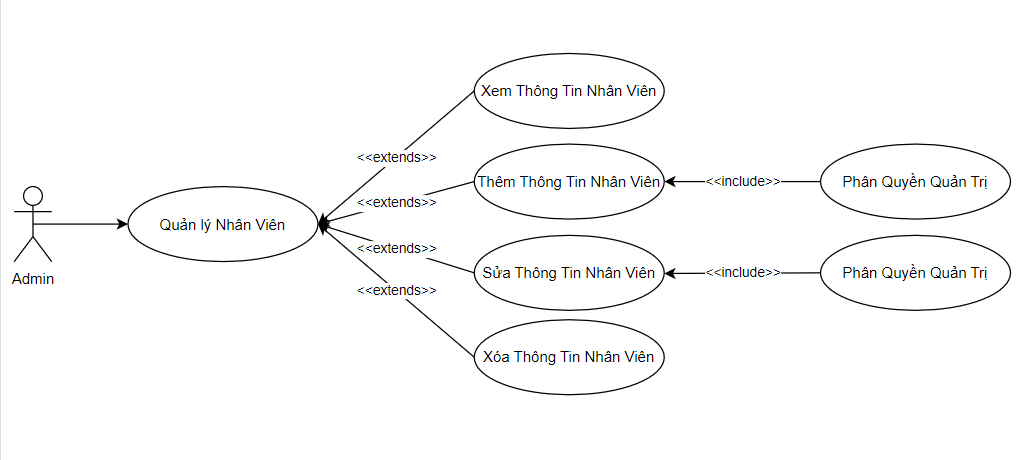
|  |  |  |
| --- | --- | --- |
| STT | Tác nhân | Diễn giải |
| 1 | Nhân viên | Là những người đã được admin cấp tài khoản và được truy cập sự dụng các tính năng cho nhân viên |
| 2 | Admin | Là người cấp tài khoản cho nhân viên, cấp quyền được sử dụng nhiều tính năng hơn nhân viên |

3.4.2 Sơ đồ UseCase tổng quát:



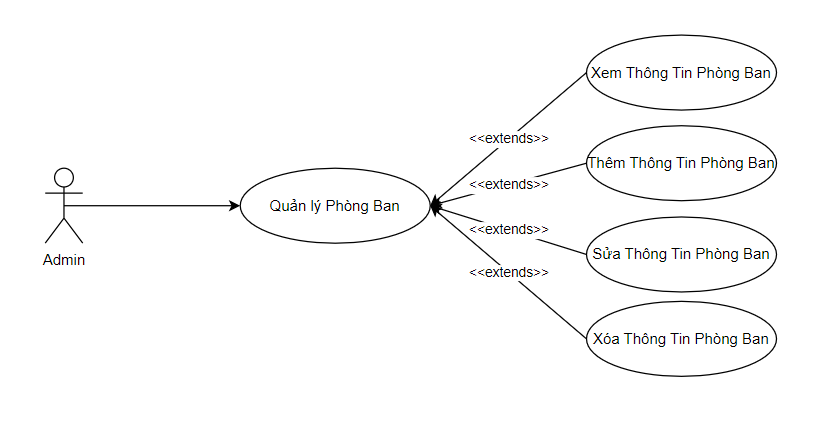
*Hình 3.3 Sơ đồ UseCase Tổng Quát Hệ Thống*

3.4.3 Sơ đồ UseCase Quản Lý Thông Tin Nhân Viên:



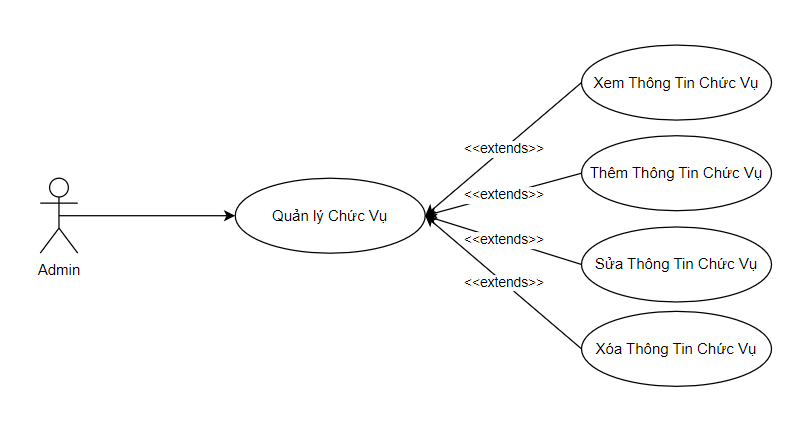
*Hình 3.5 UseCase Quản Lý Thông Tin Nhân Viên*

3.4.4 Sơ đồ UseCase Quản Lý Thông Tin Phòng Ban:



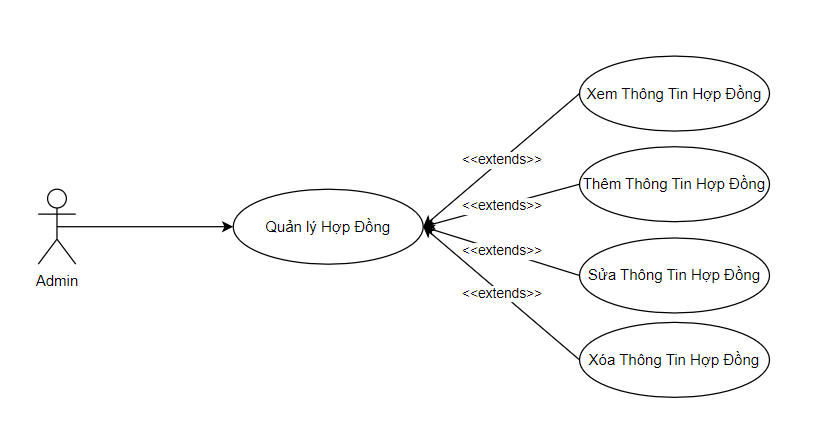
*Hình 3.6 UseCase Quản lý Phòng Ban Admin*

3.4.5 Sơ đồ UseCase Quản Lý Thông Tin Chức Vụ:



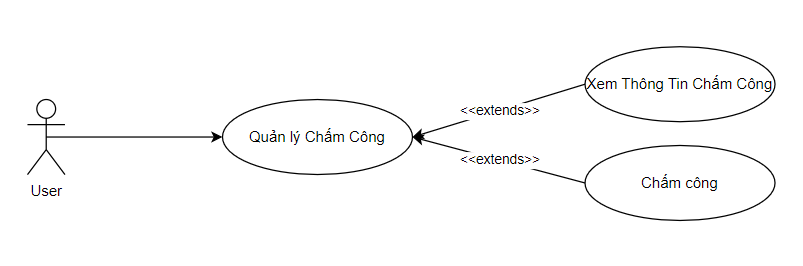
*Hình 3.7 Usecase quản lý thông tin chức vụ*

3.4.6 Sơ đồ UseCase Quản Lý Thông Tin Hợp Đồng:



*Hình 3.8 Sơ đồ UseCase Quản lý Hợp Đồng Admin*

3.4.7 Sơ đồ UseCase Quản Lý Thông Tin Chấm Công:



*Hình 3.9 Sơ đồ UseCase Quản lý Chấm Công Actor Nhân Viên*

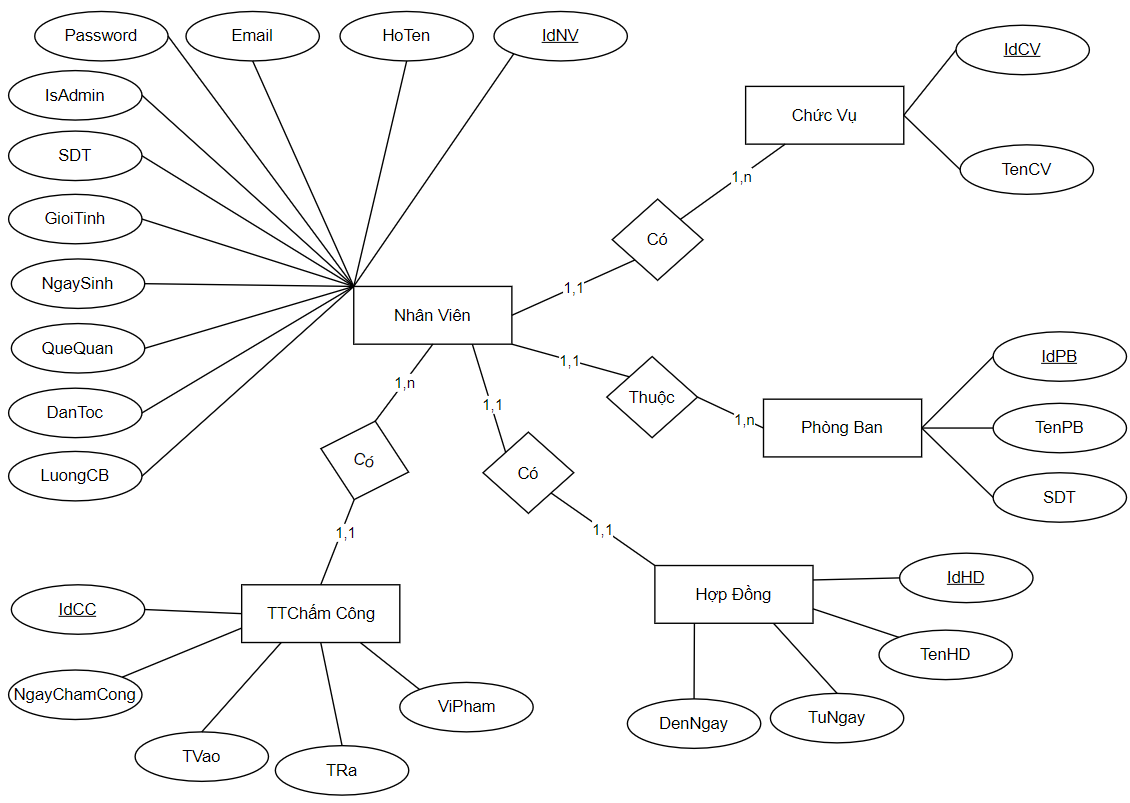
3.4.8 Sơ đồ UseCase Quản Lý Thông Tin Lương:

A picture containing diagram, line, circle, text

Description automatically generated

*Hình 3.12 Sơ đồ UseCase Quản lý Lương Admin*

3.5 Mô hình ER



*Hình 3.13 Mô hình ER*

3.6 Sơ đồ luồng dữ liệu (DFD)

Sơ đồ luồng dữ liệu (Data Flow Diagram - DFD) là một cách biểu diễn luồng dữ

liệu thông qua một quy trình hoặc hệ thống. DFD cũng cung cấp thông tin về đầu ra và

đầu vào của mỗi thực thể và của chính quá trình. Một sơ đồ luồng dữ liệu không có

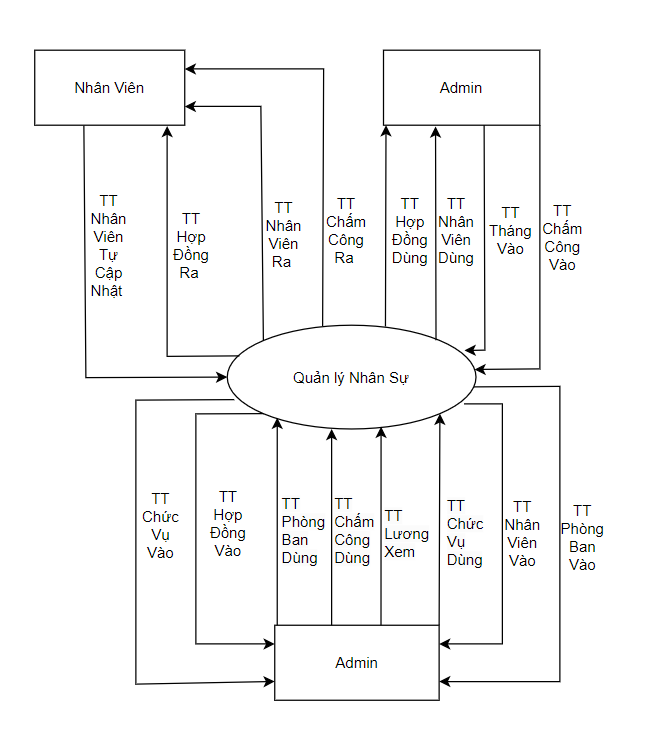
luồng điều khiển, không có quy tắc quyết định và không có vòng lặp.

Đặc trưng của sơ đồ DFD là tiếp cận chủ yếu theo hướng từ trên xuống theo cách

xem các hoạt động xử lý như hộp đen và quan tâm đến việc lưu trữ và xử lý thông tin

giữa các hộp đen này. Mặc dù DFD biểu diễn cả xử lý lẫn dữ liệu hệ thống, nhưng chú

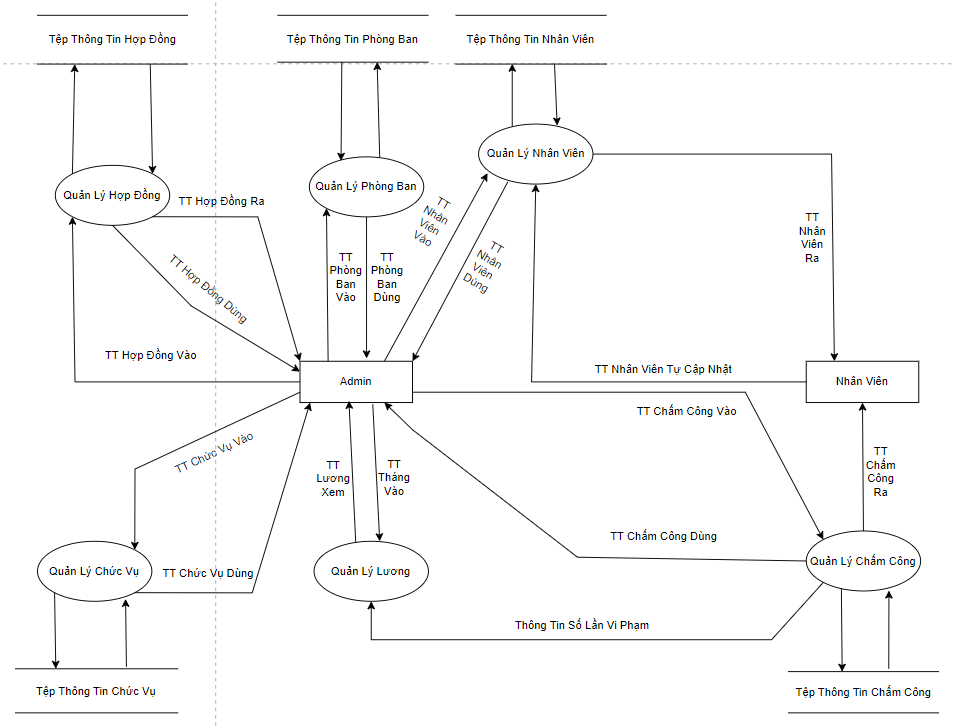
ý đến xử lý hơn là dữ liệu.



*Hình 3.14 Sơ đồ luồng dữ liệu(DFD)*

3.7 Sơ đồ luồng dữ liệu chi tiết

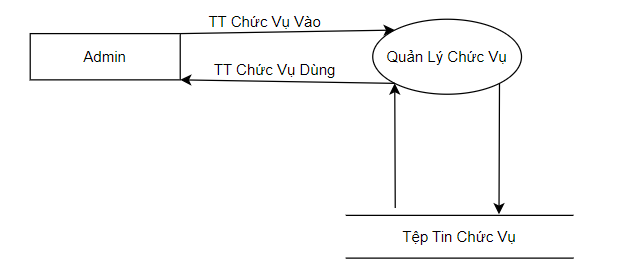
3.7.1 Mức đỉnh



*Hình 3.15 Sơ đồ luồng dữ liệu Mức Đỉnh*

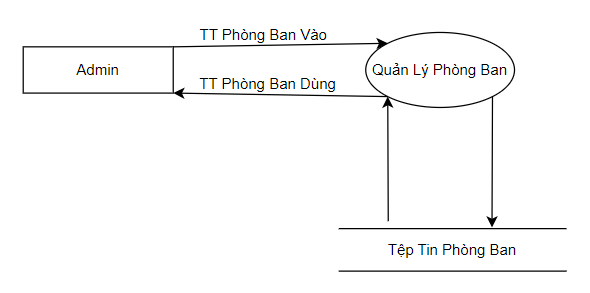
3.7.2 Mức dưới đỉnh (Mức 1)

Quản lý Chức Vụ



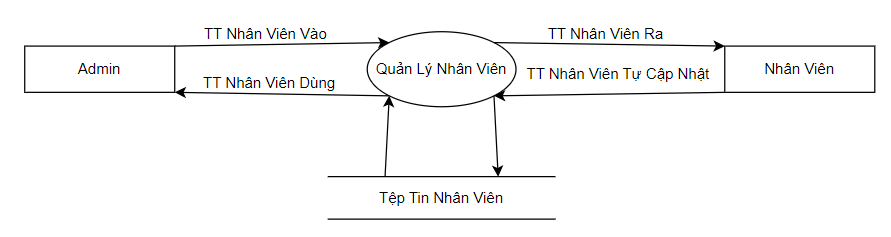
*Hình 3.16 Sơ đồ luồng mức dưới đỉnh Quản lý Chức Vụ*

Quản lý Phòng Ban



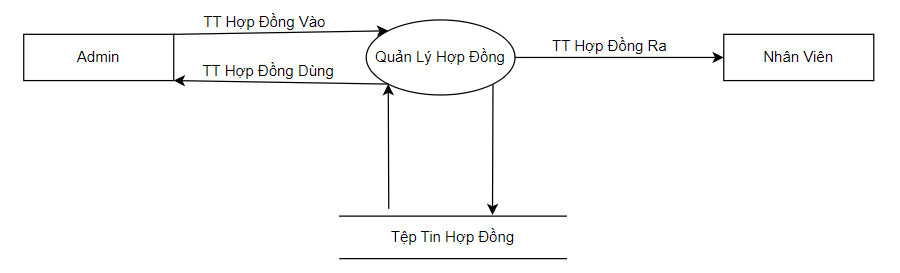
*Hình 3.17 Sơ đồ luồng dưới mức đỉnh Quản lý Phòng Ban*

Quản lý Nhân Viên



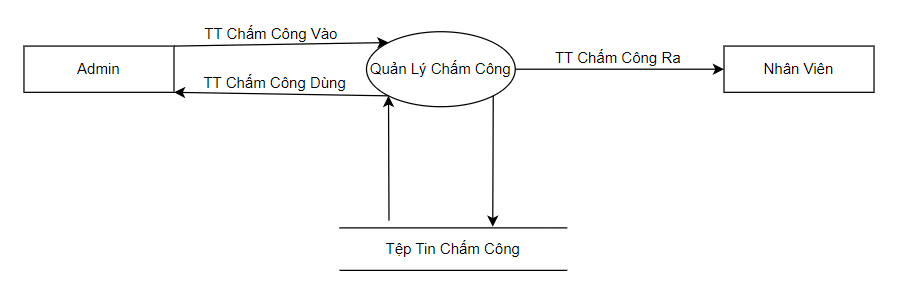
*Hình 3.18 Sơ đồ luồng dưới mức đỉnh Quản lý Nhân Viên*

Quản lý Hợp Đồng



*Hình 3.19 Sơ đồ luồng dưới mức đỉnh Quản lý Hợp Đồng*

Quản lý Chấm Công



*Hình 3.20 Sơ đồ luồng dưới mức đỉnh Quản lý Chấm Công*

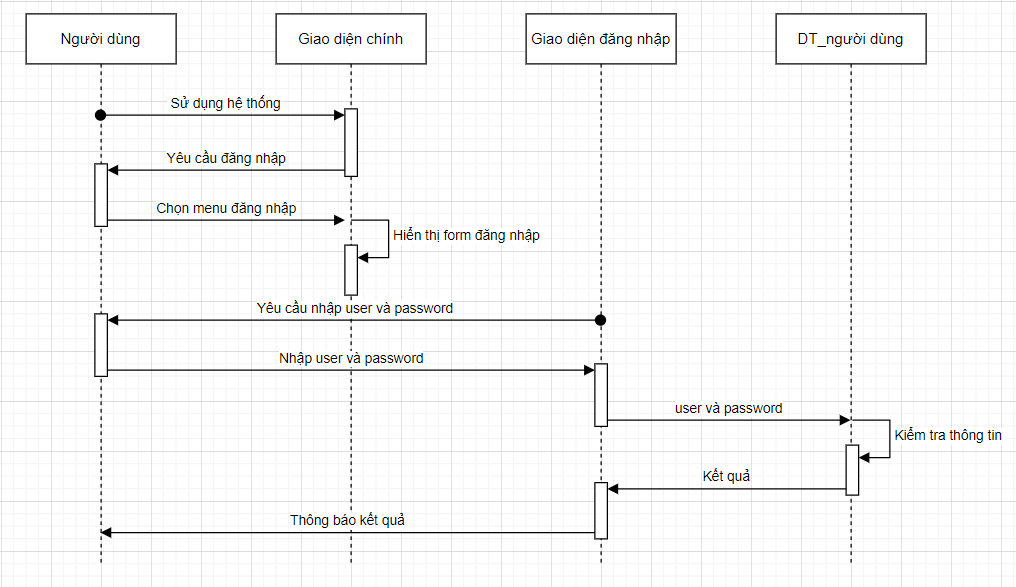
Quản Lý Lương



*Hình 3.21: Sơ đồ hệ thống chi tiết dưới mức đỉnh Quản lý Lương*

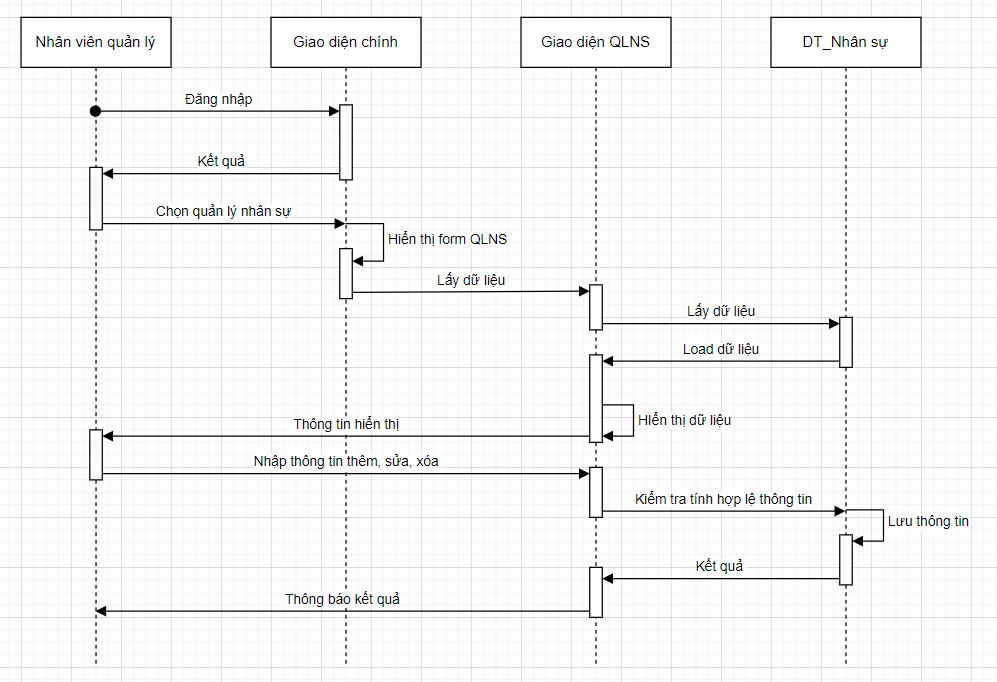
**3.8 Sơ đồ tuần tự**

**3.8.1 Sơ đồ tuần tự quá trình đăng nhập**



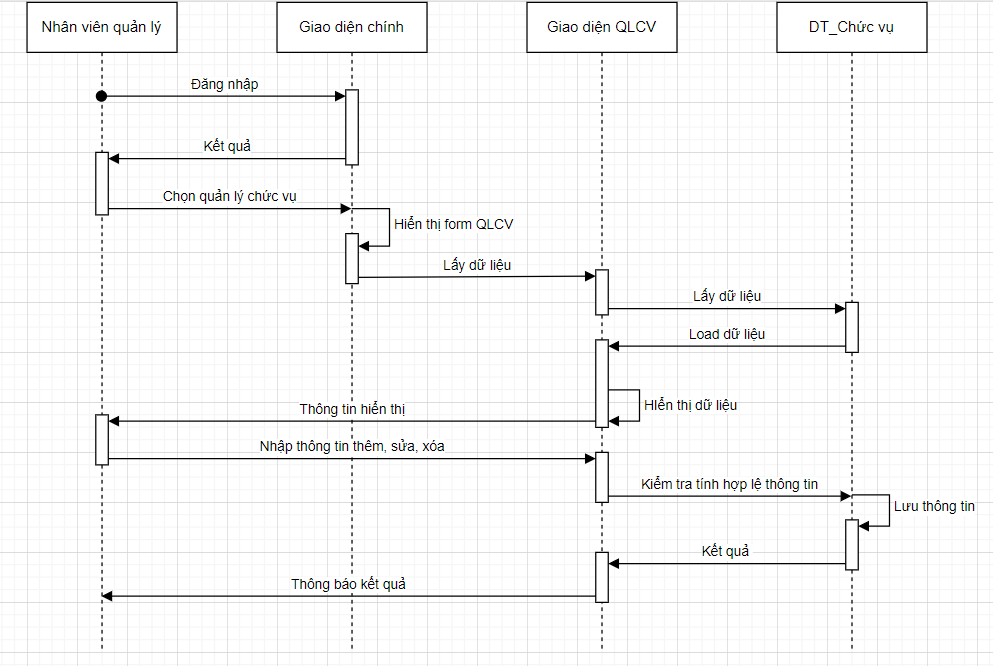
*Hình 3.22 Sơ đồ tuần tự quá trình đăng nhập*

**3.8.2 Sơ đồ tuần tự quá trình quản lý thông tin nhân viên**



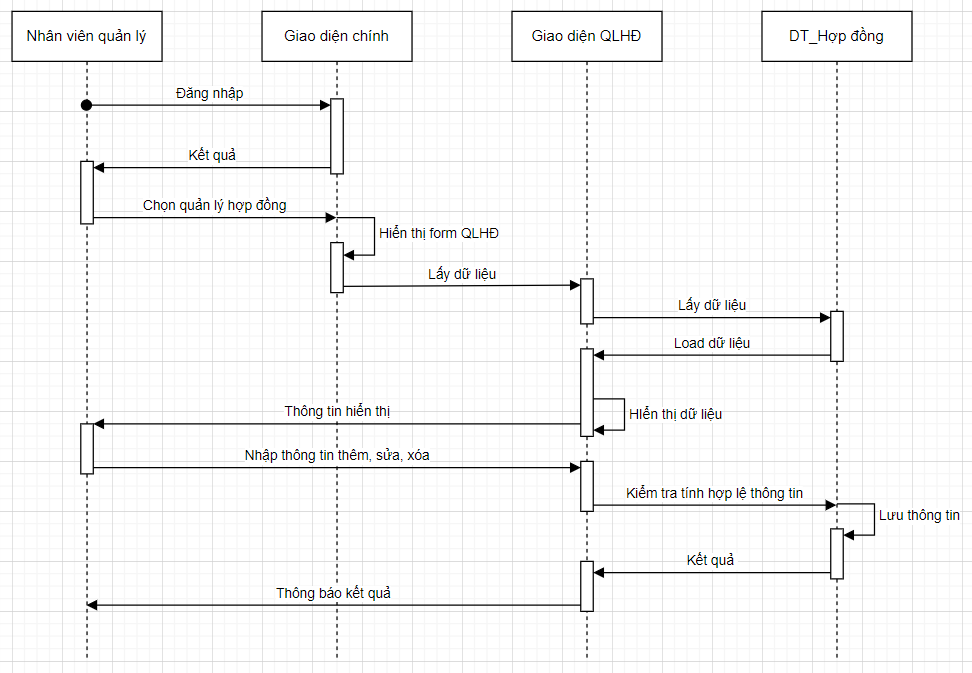
*Hình 3.23 Sơ đồ tuần tự quá trình quản lý thông tin nhân viên*

**3.8.3 Sơ đồ tuần tự quá trình quản lý chức vụ**



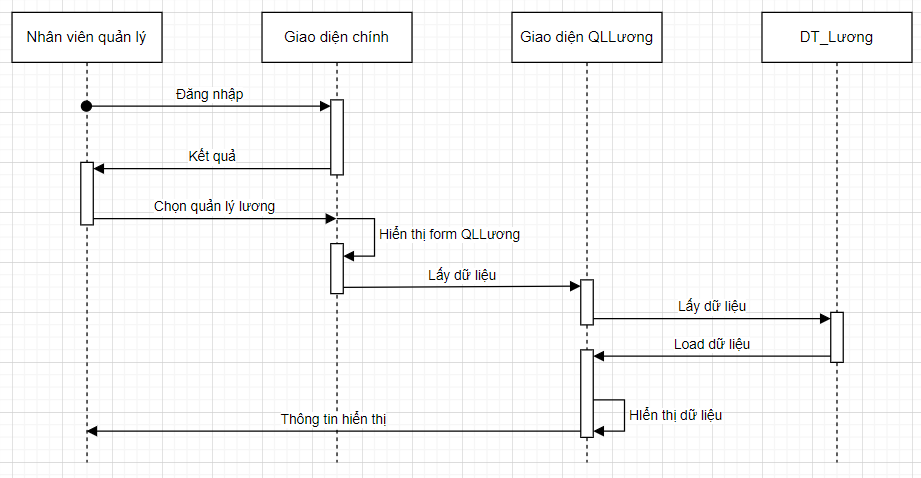
*Hình 3.24 Sơ đồ tuần tự quá trình quản lý chức vụ*

**3.8.4 Sơ đồ tuần tự quá trình quản lý hợp đồng**



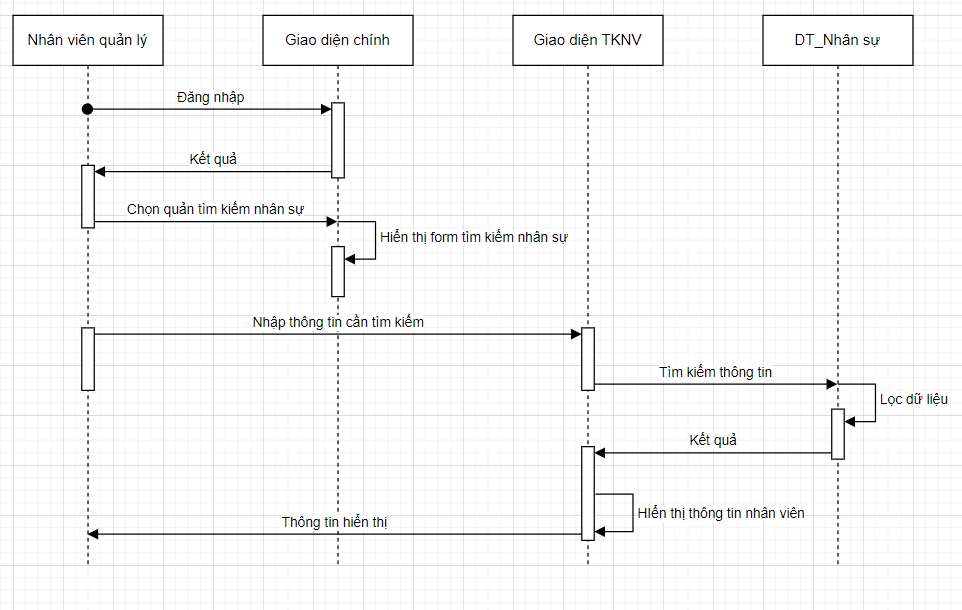
*Hình 3.25 Sơ đồ tuần tự quá trình quản lý hợp đồng*

**3.8.5 Sơ đồ tuần tự quá trình quản lý lương**



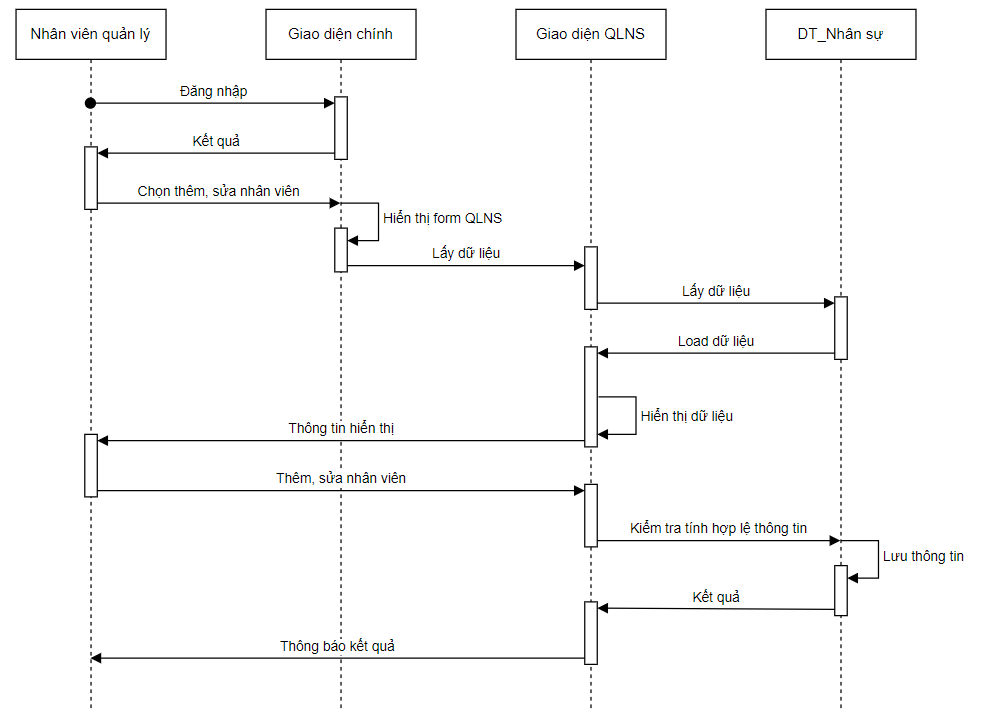
*Hình 3.26 Sơ đồ tuần tự quá trình quản lý lương*

**3.8.6 Sơ đồ tuần tự quá trình tìm kiếm nhân viên**



*Hình 3.26 Sơ đồ tuần tự quá trình tìm kiếm nhân viên*

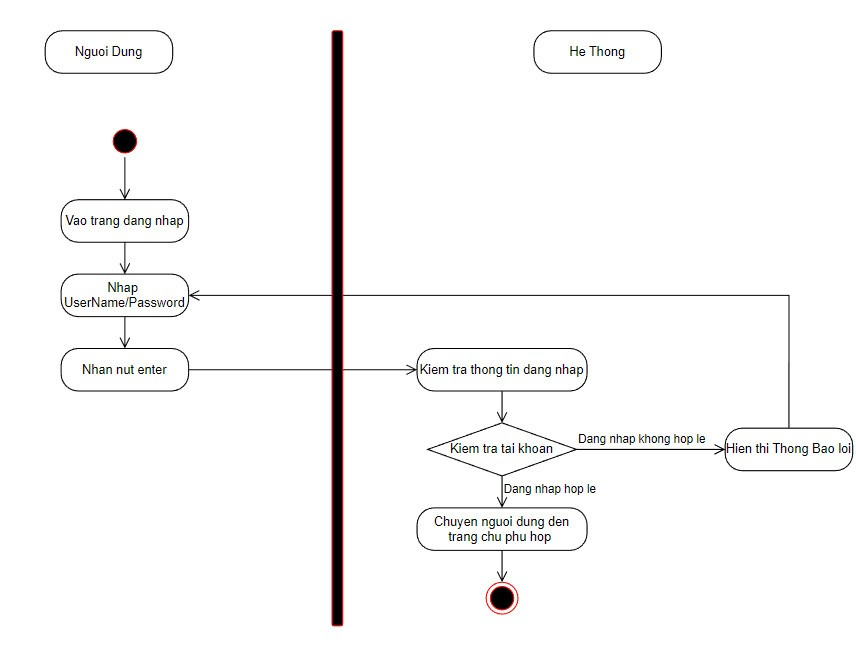
**3.8.7 Sơ đồ tuần tự quá trình quản lý phân quyền**



*Hình 3.27 Sơ đồ tuần tự quá trình quản lý phân quyền*

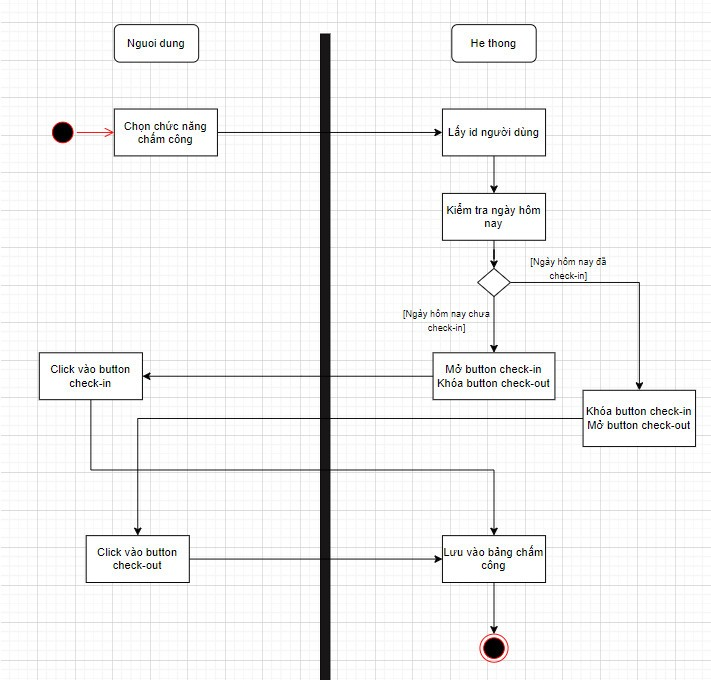
**3.9 Sơ đồ hoạt động**

**3.9.1 Sơ đồ hoạt động đăng nhập**



*Hình 3.28 Sơ đồ hoạt động đăng nhập*

**3.9.2 Sơ đồ hoạt động chấm công**



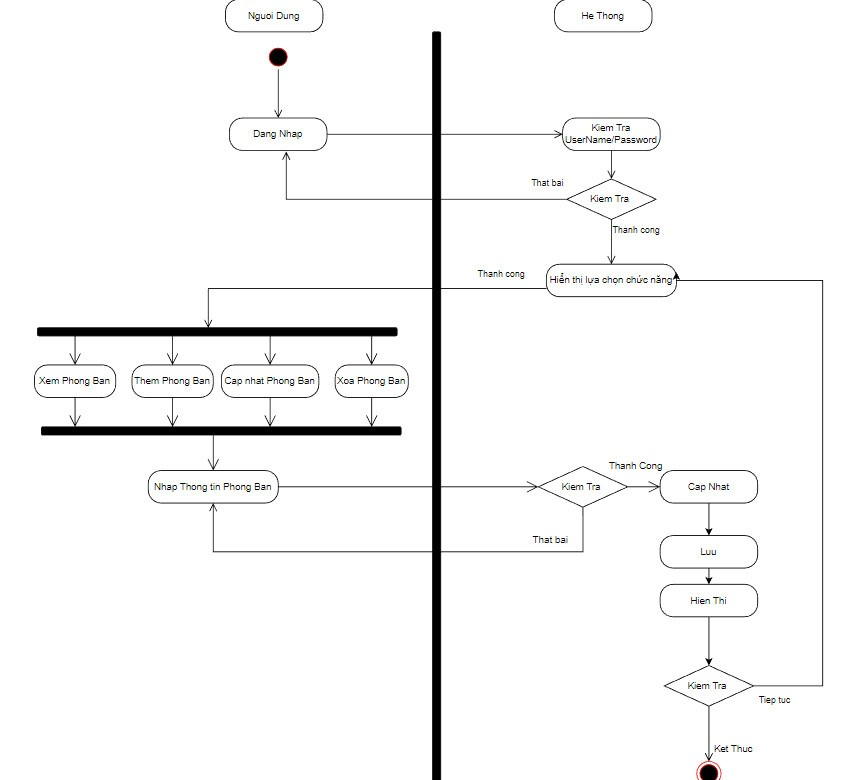
*Hình 3.29 Sơ đồ hoạt động chấm công*

**3.9.3 Sơ đồ hoạt động quản lý thông tin nhân viên**



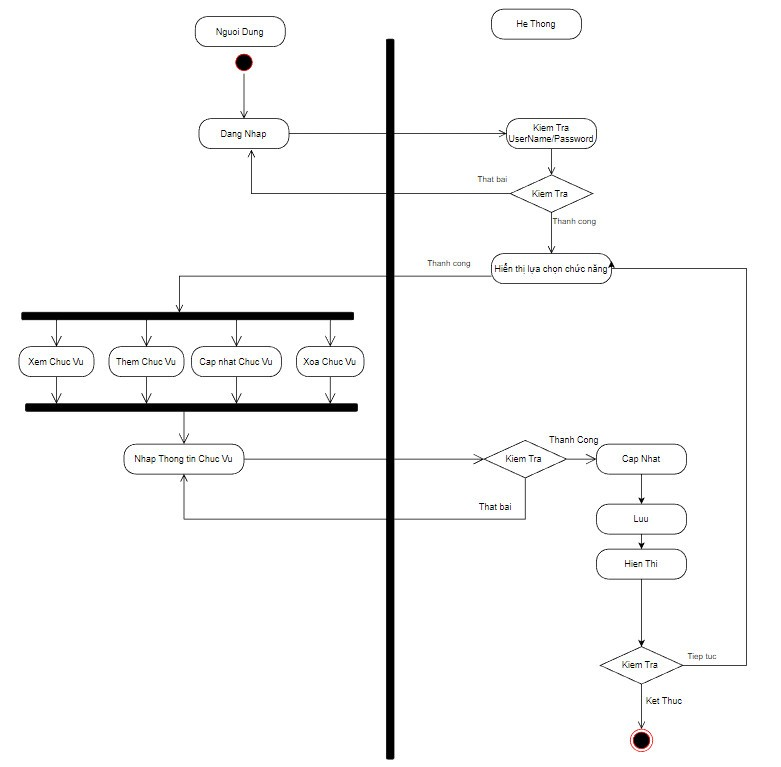
*Hình 3.30 Sơ đồ hoạt động quản lý thông tin nhân viên*

**3.9.4 Sơ đồ hoạt động quản lý phòng ban**



*Hình 3.31 Sơ đồ hoạt động quản lý phòng ban*

**3.9.5 Sơ đồ hoạt động quản lý chức vụ**



*Hình 3.32 Sơ đồ hoạt động quản lý chức vụ*

**3.9.6 Sơ đồ hoạt động quản lý hợp đồng**



*Hình 3.33 Sơ đồ hoạt động quản lý hợp đồng*

CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH

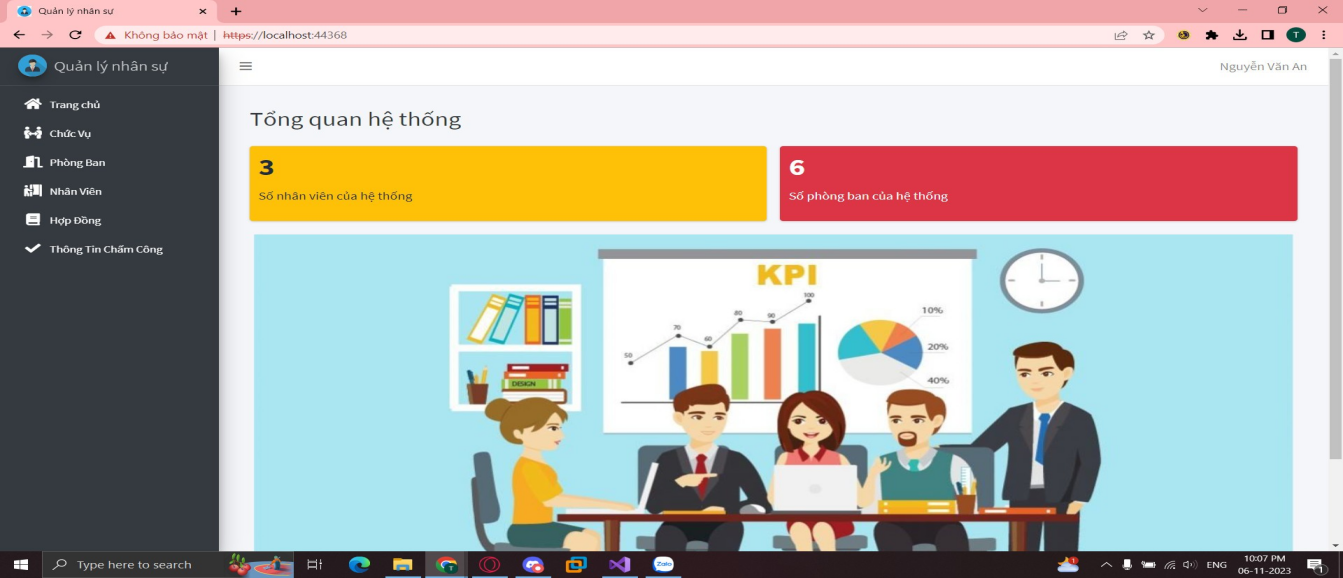
4.1. Giao diện Đăng Nhập

A screenshot of a computer

Description automatically generated

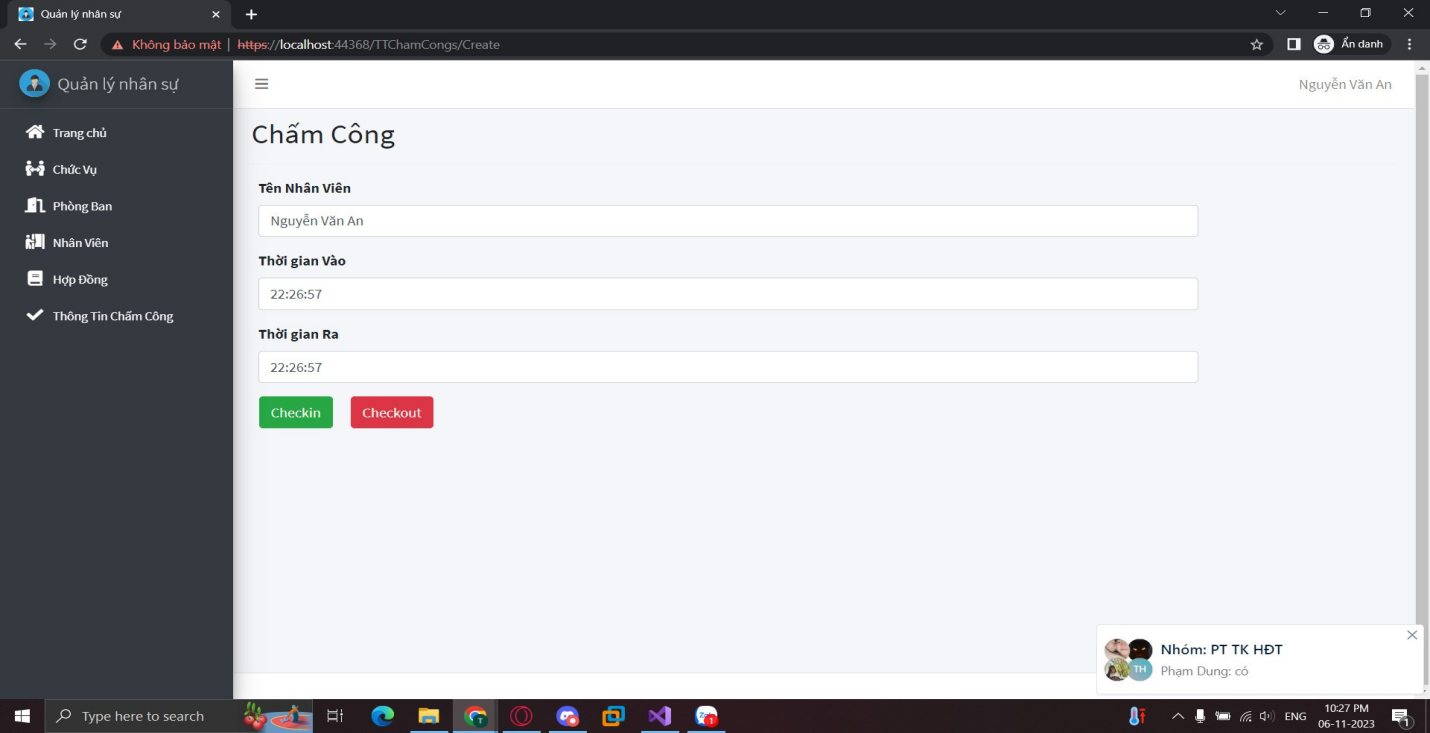
*Hình 4.1 Giao diện đăng nhập*

4.2 Giao diện trang chủ



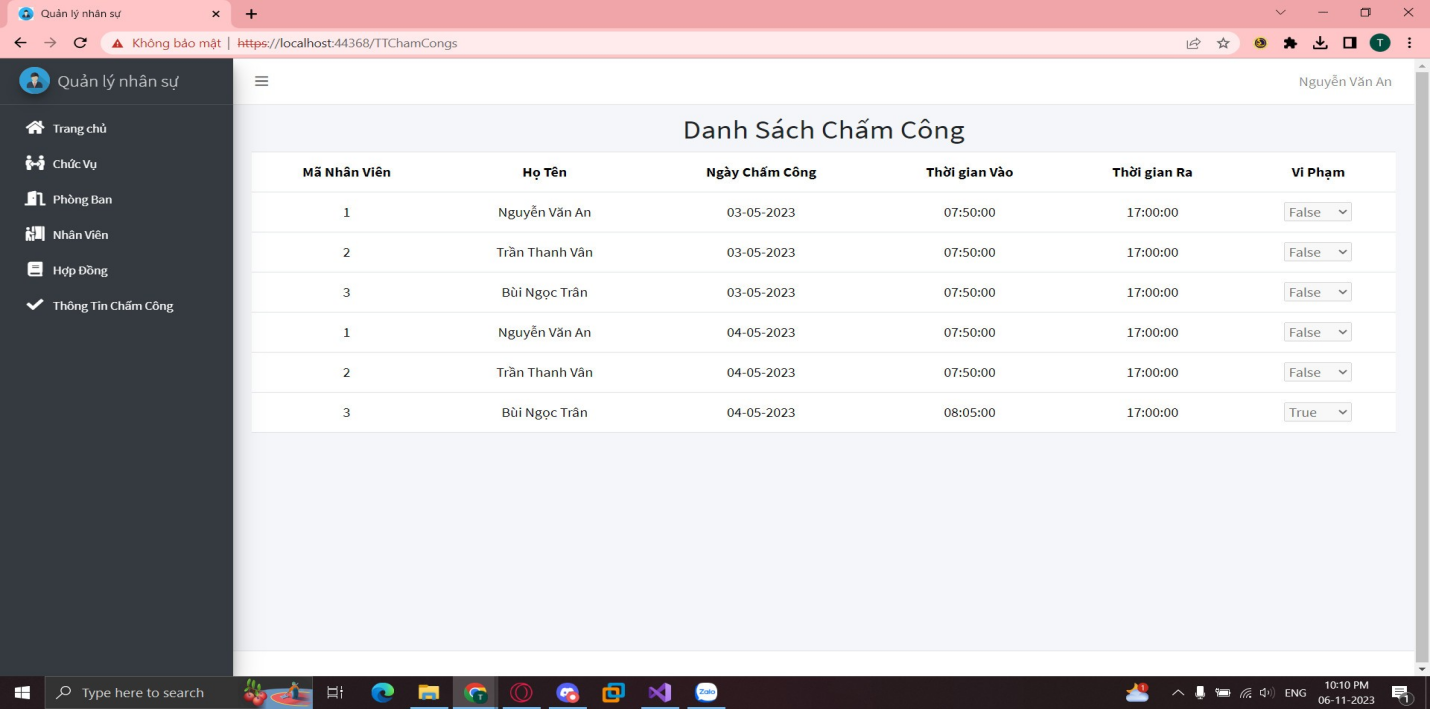
*Hình 4.2 Giao diện trang chủ*

4.3 Giao diện chấm công



*Hình 4.3 Giao diện chấm công*

4.4 Giao diện danh sách chấm công



*Hình 4.4 Giao diện danh sách chấm công*

4.5 Giao diện thêm nhân viên



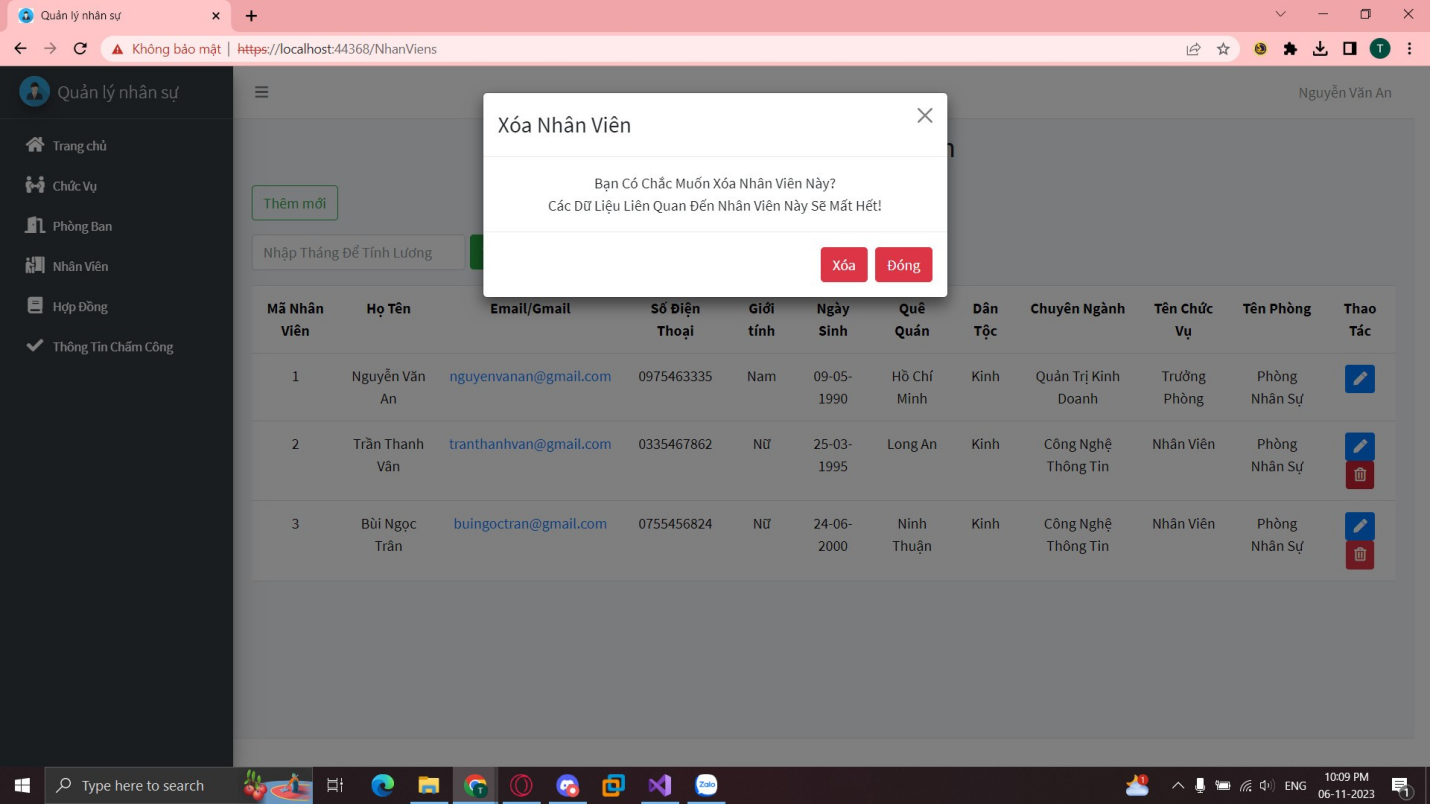
*Hình 4.5 Giao diện thêm nhân viên*

4.6 Giao diện sửa thông tin nhân viên



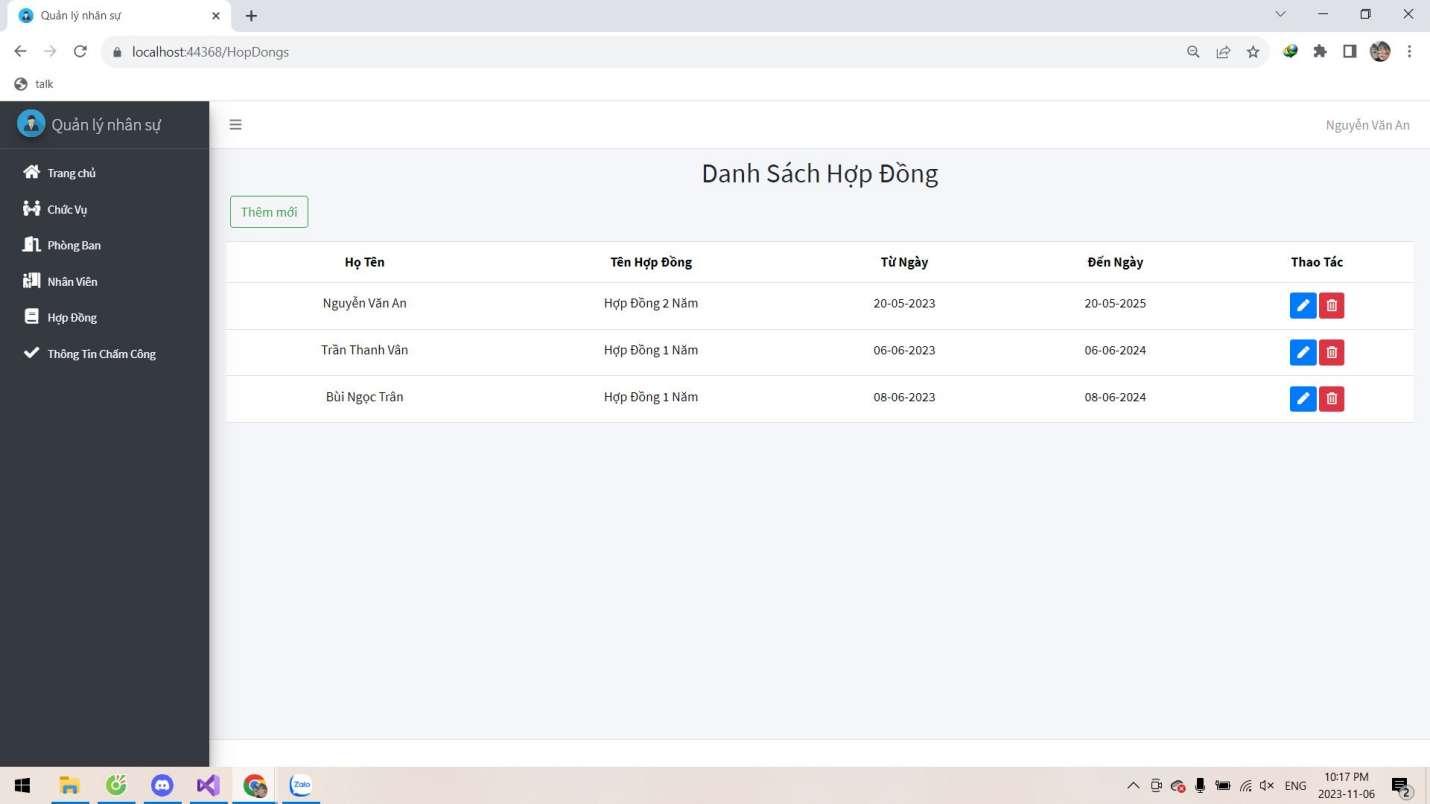
*Hình 4.6 Giao diện sửa thông tin nhân viên*

4.7 Giao diện xóa nhân viên



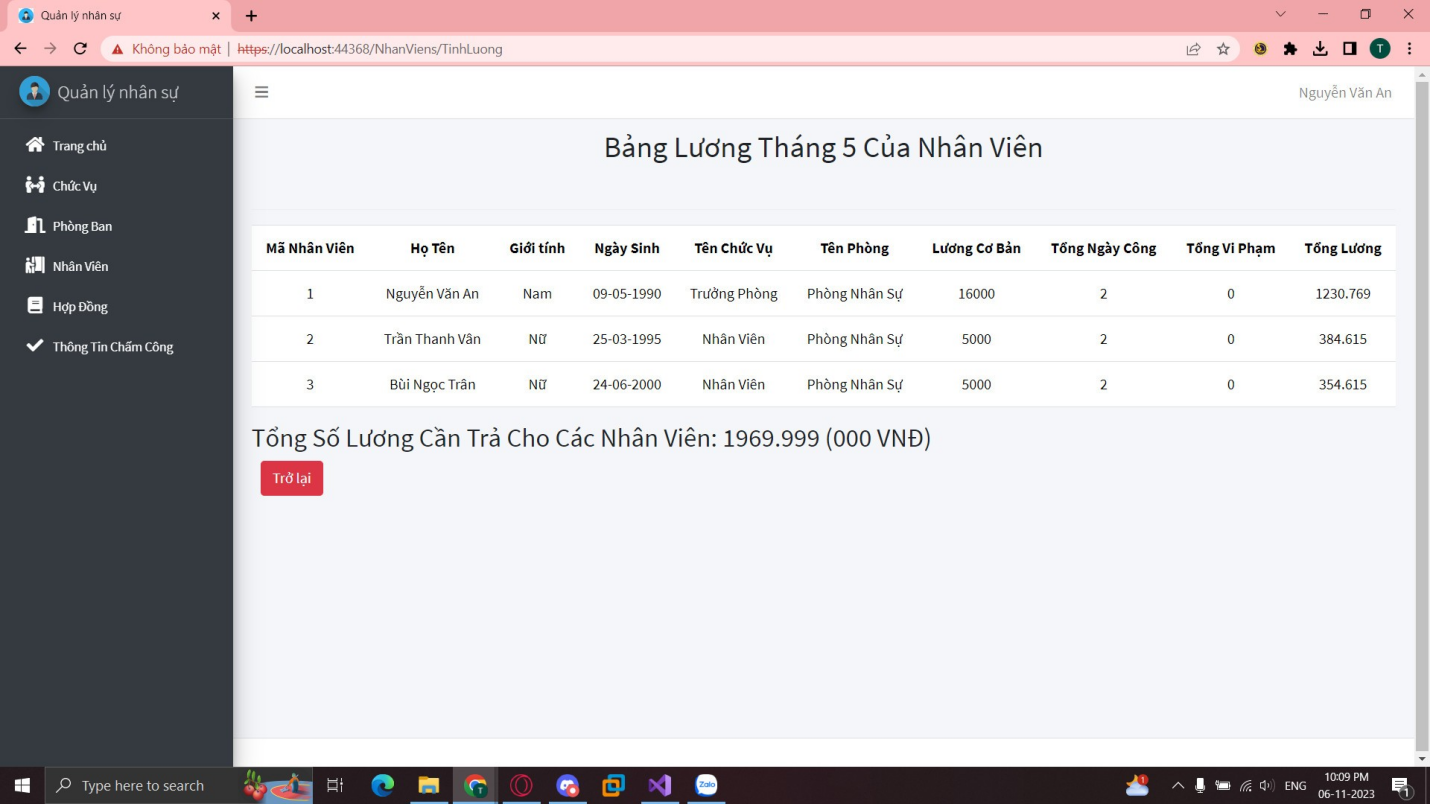
*Hình 4.7 Giao diện xóa nhân viên*

4.8 Giao diện danh sách hợp đồng



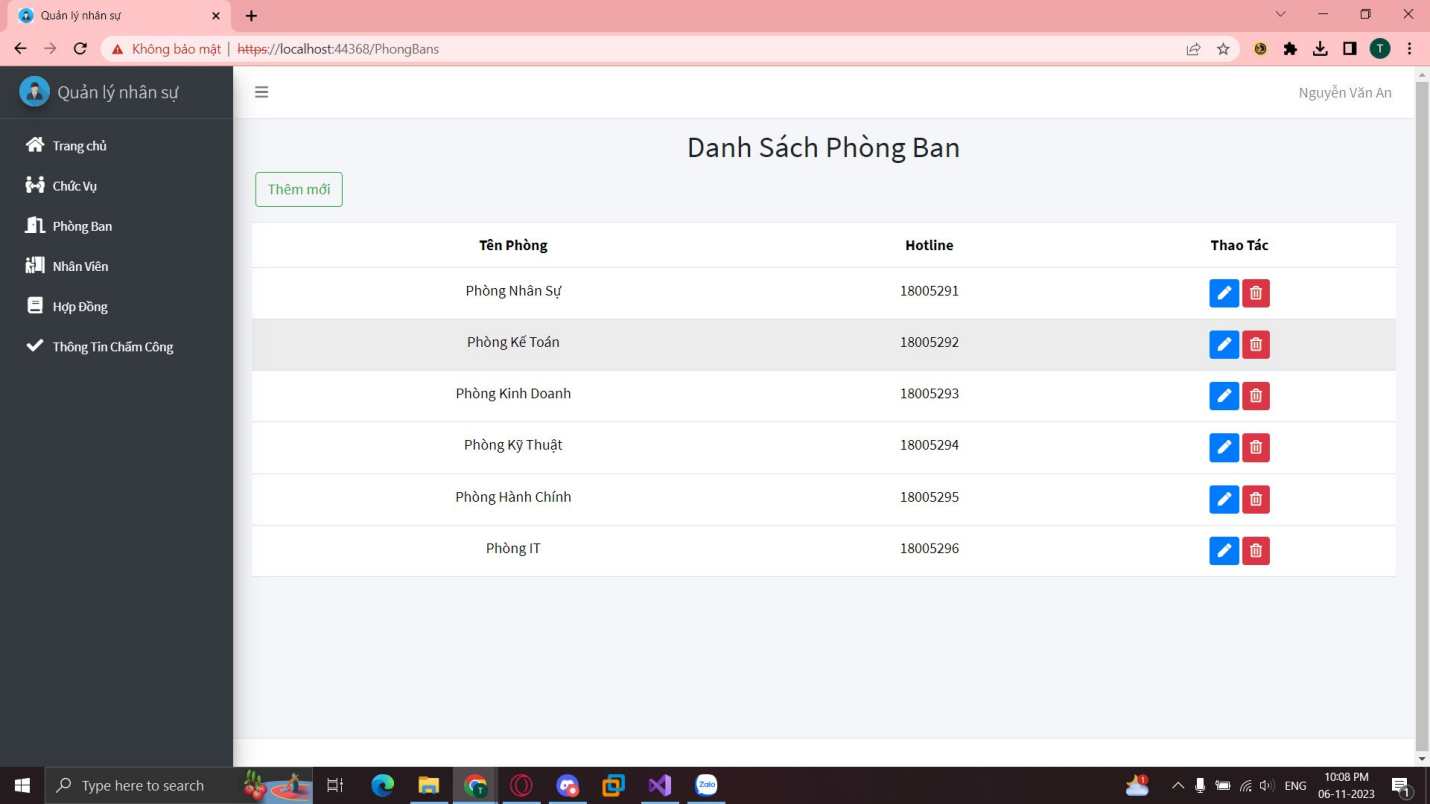
*Hình 4.8 Giao diện danh sách hợp đồng*

4.9 Giao diện lương nhân viên



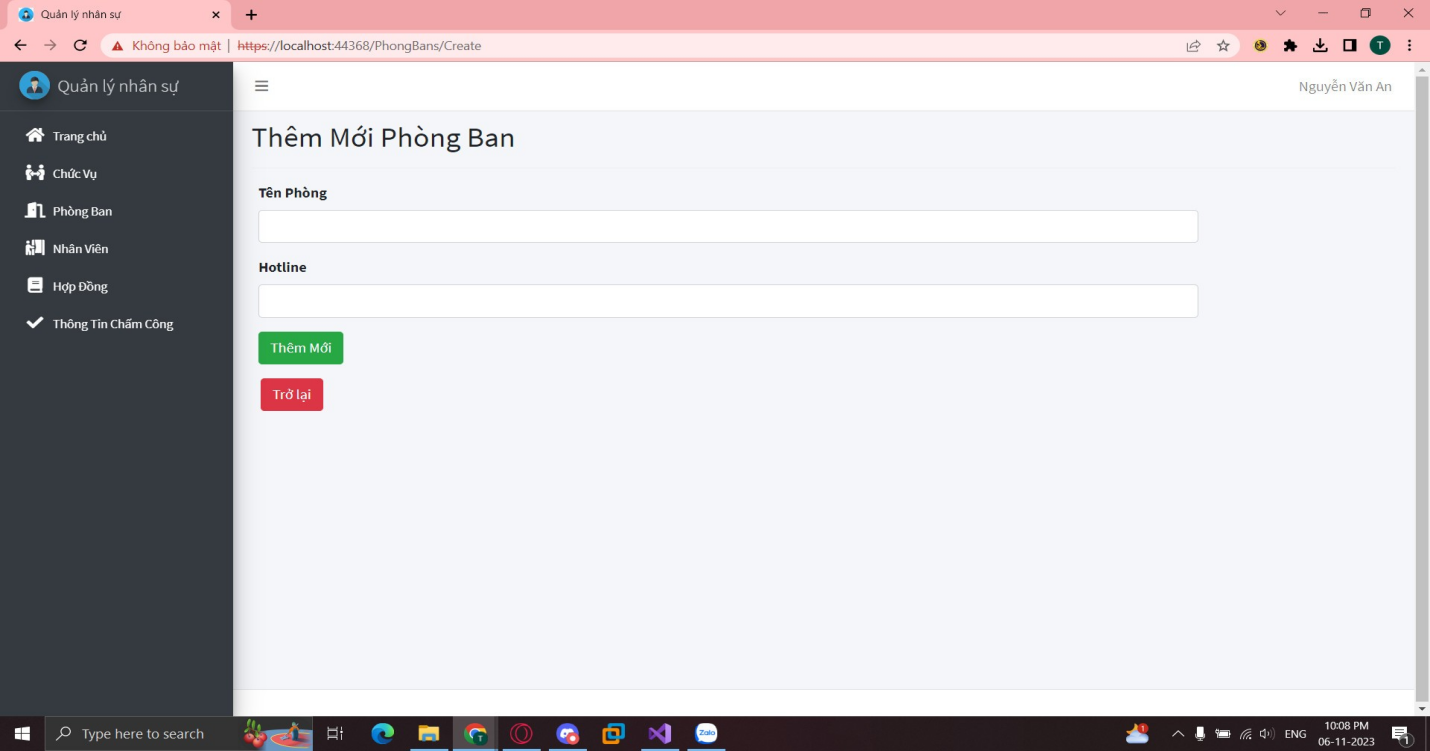
*Hình 4.9 Giao diện danh sách lương nhân viên*

4.10 Giao diện danh sách phòng ban



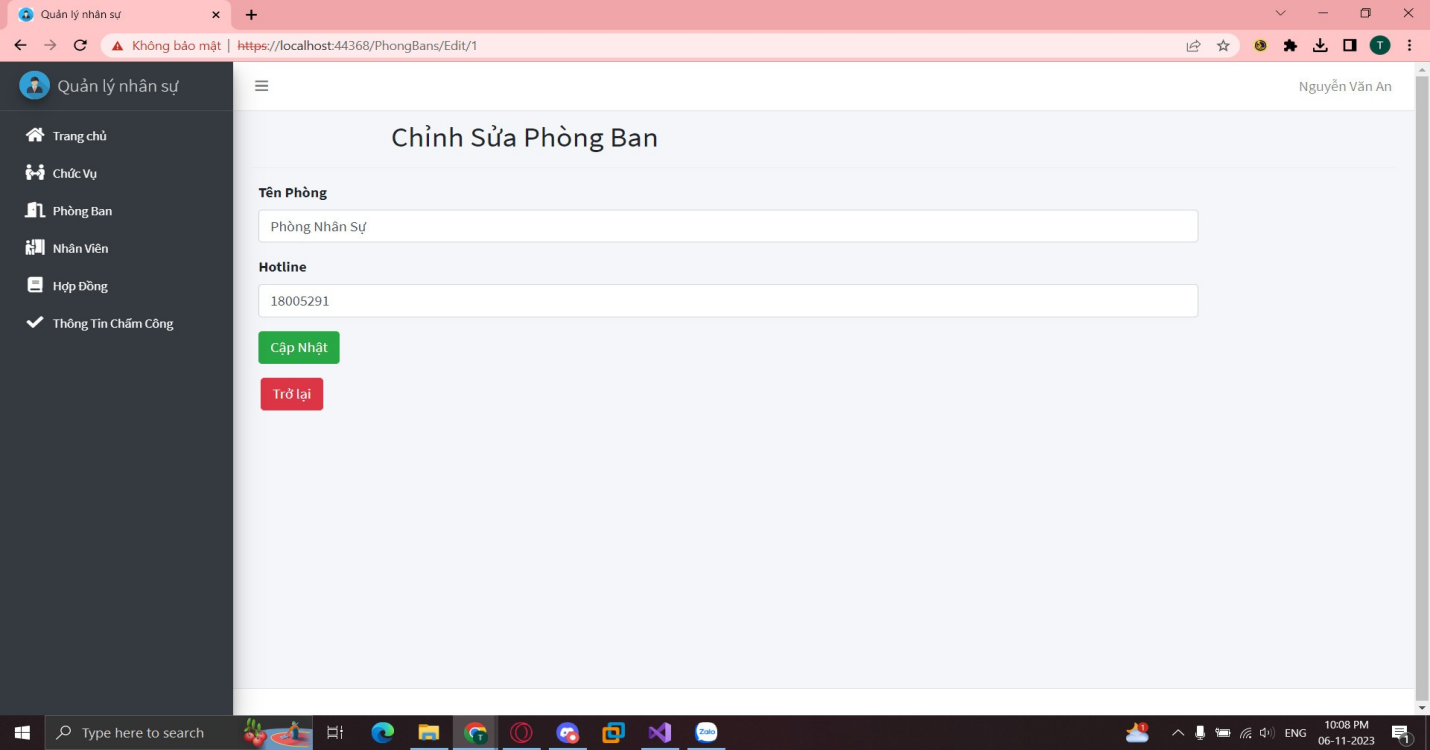
*Hình 4.10 Giao diện danh sách phòng ban*

4.11 Giao diện thêm phòng ban



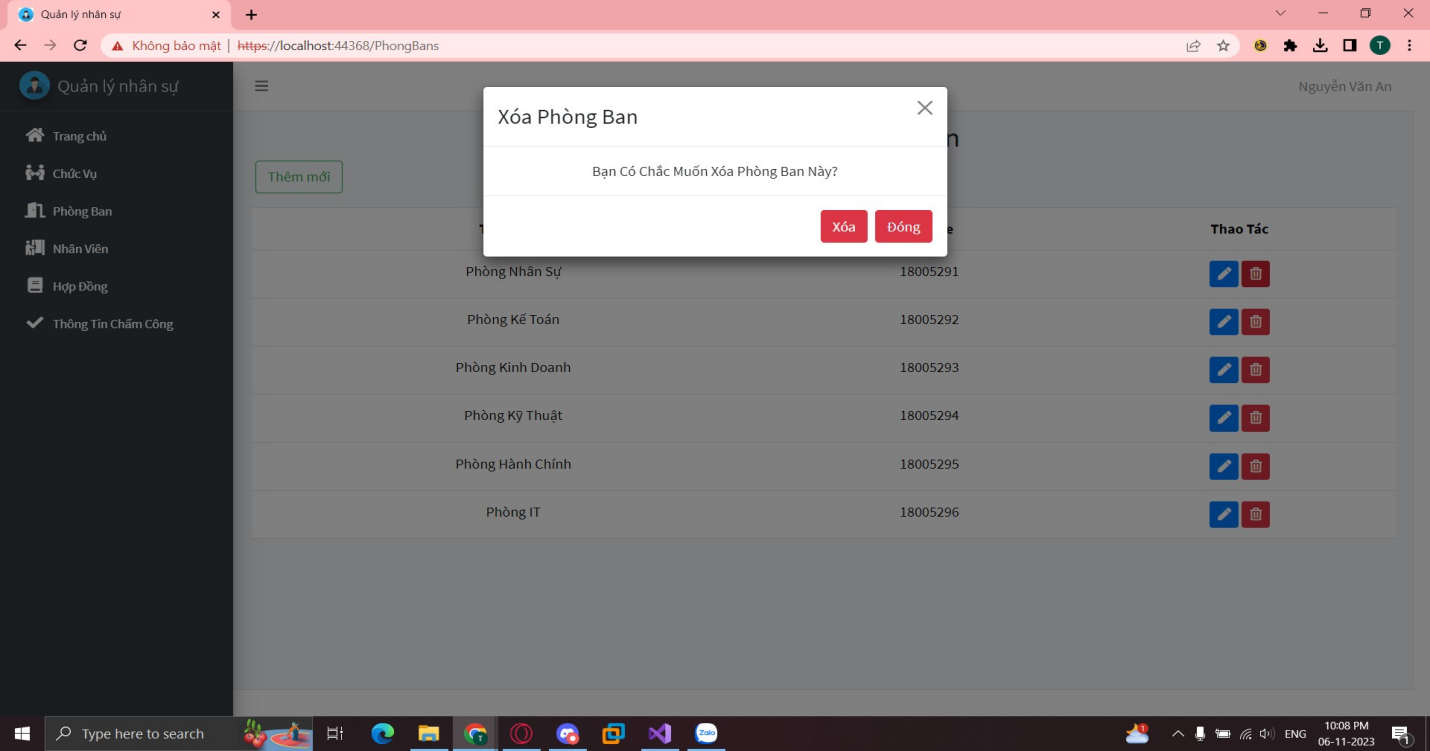
*Hình 4.11 Giao diện thêm mới phòng ban*

4.12 Giao diện sửa phòng ban



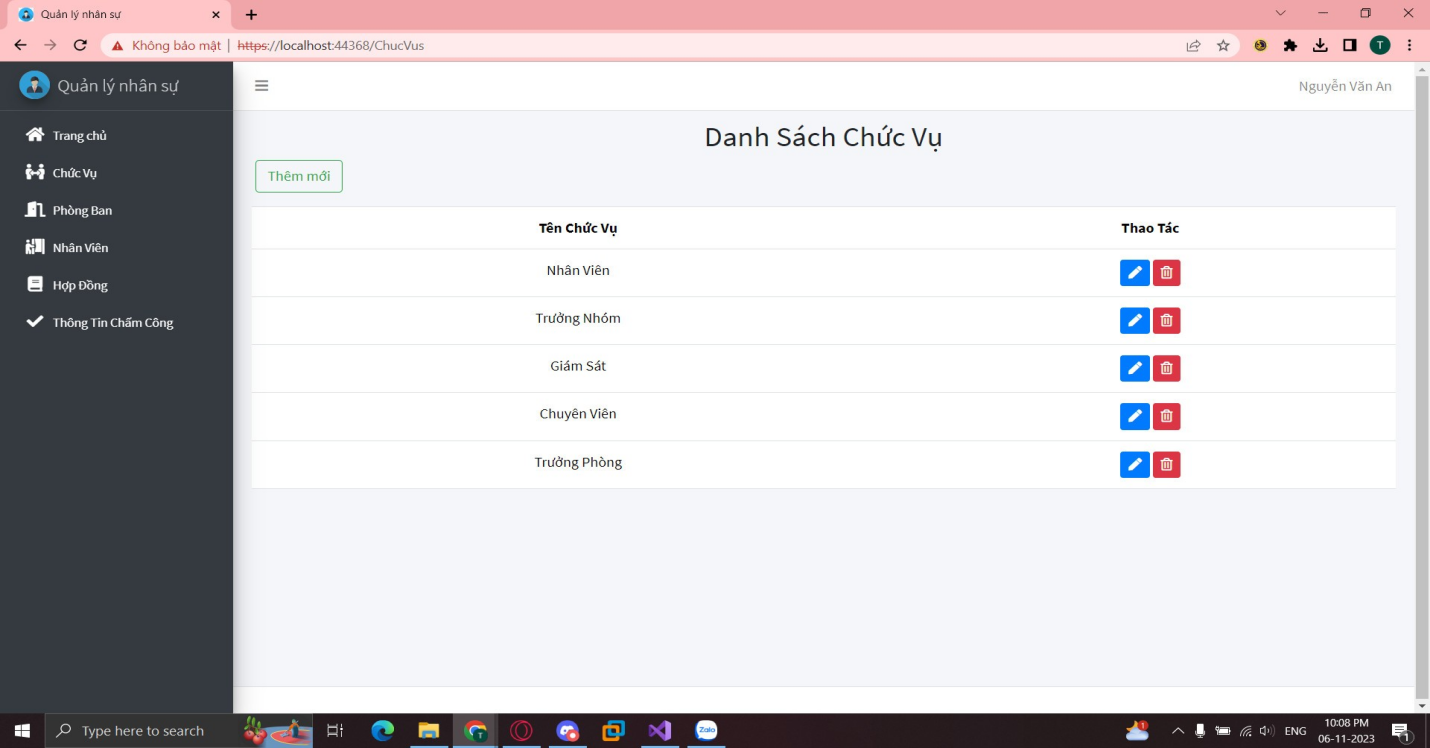
*Hình 4.12 Giao diện chỉnh sửa phòng ban*

4.13 Giao diện xóa phòng ban



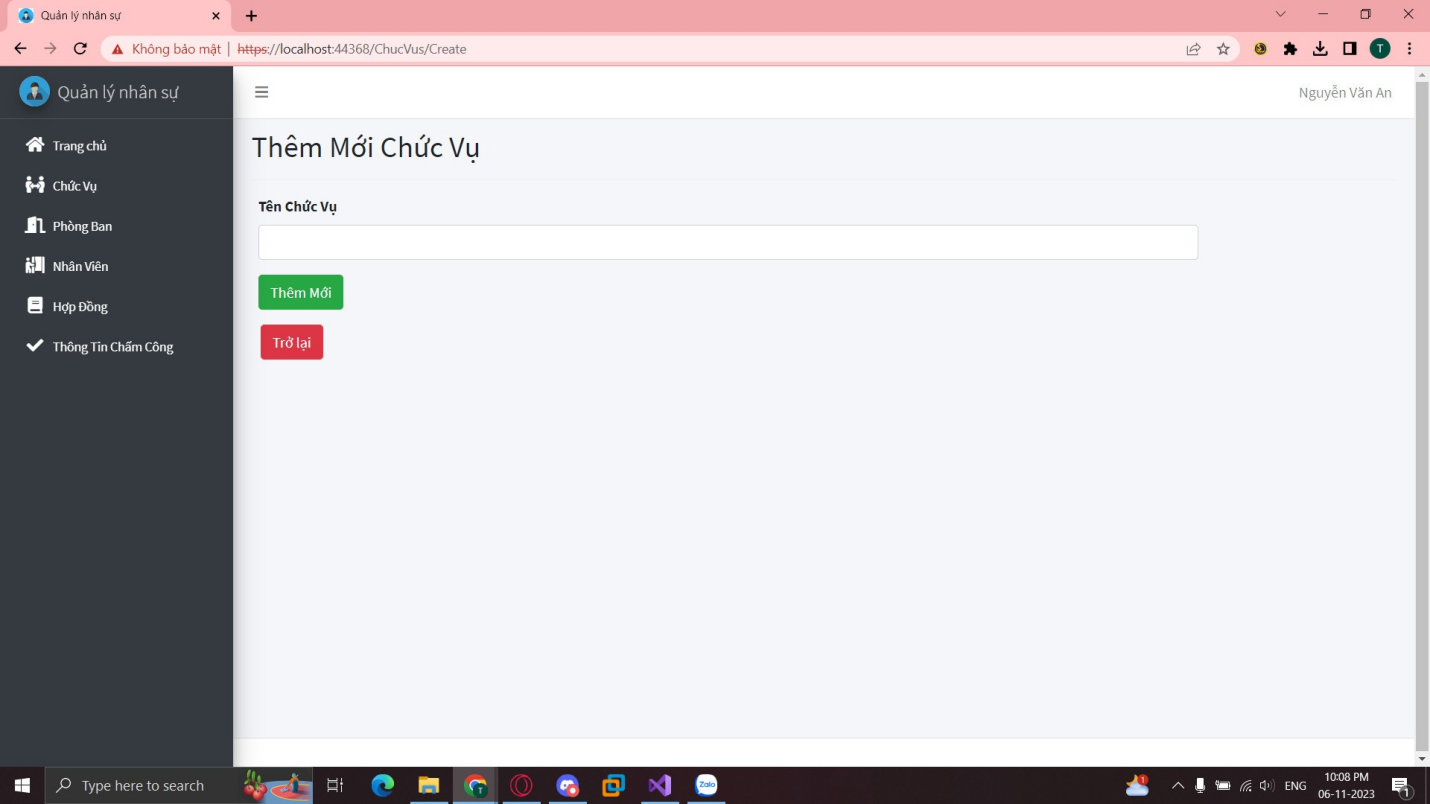
*Hình 4.13 Giao diện xóa phòng ban*

4.14 Giao diện danh sách chức vụ



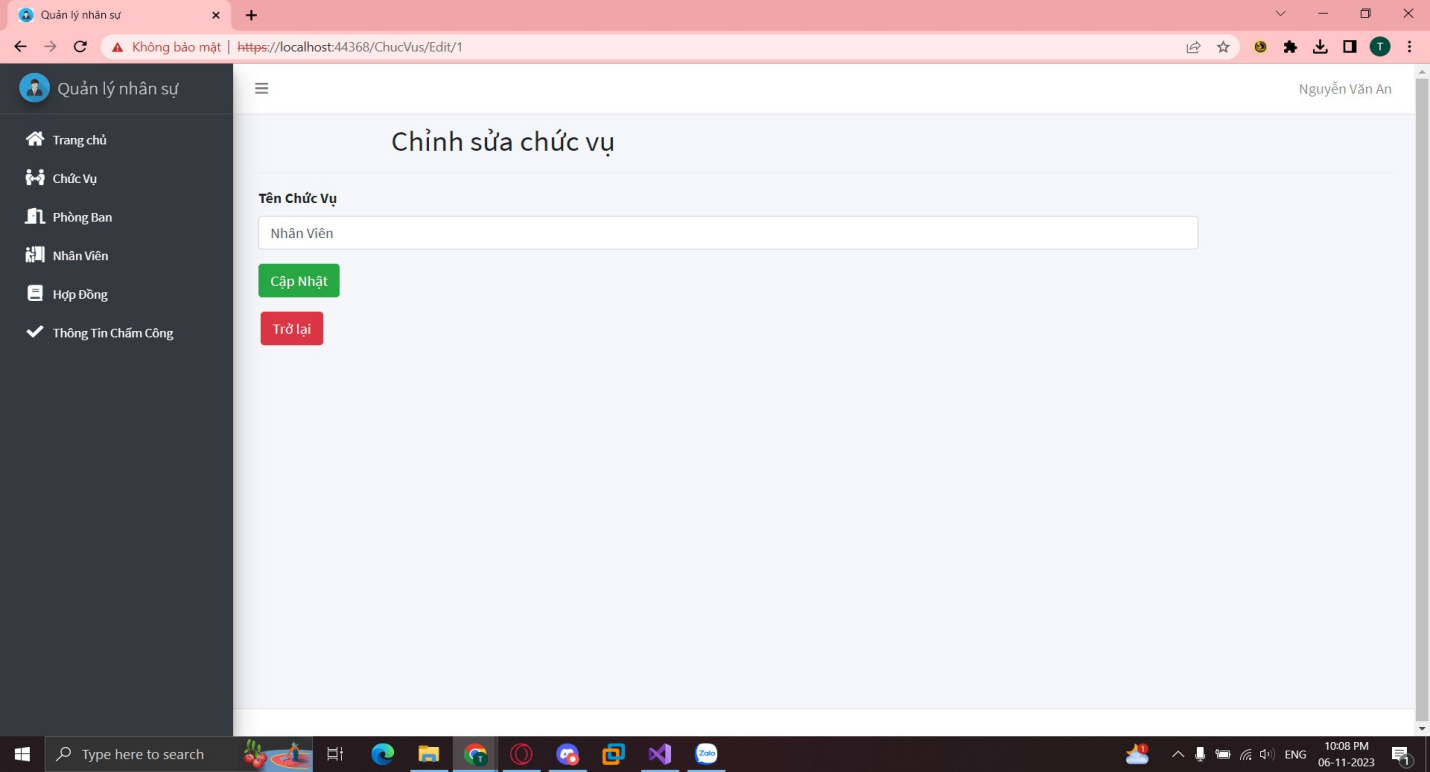
*Hình 4.14 Giao diện danh sách chức vụ*

4.15 Giao diện thêm chức vụ



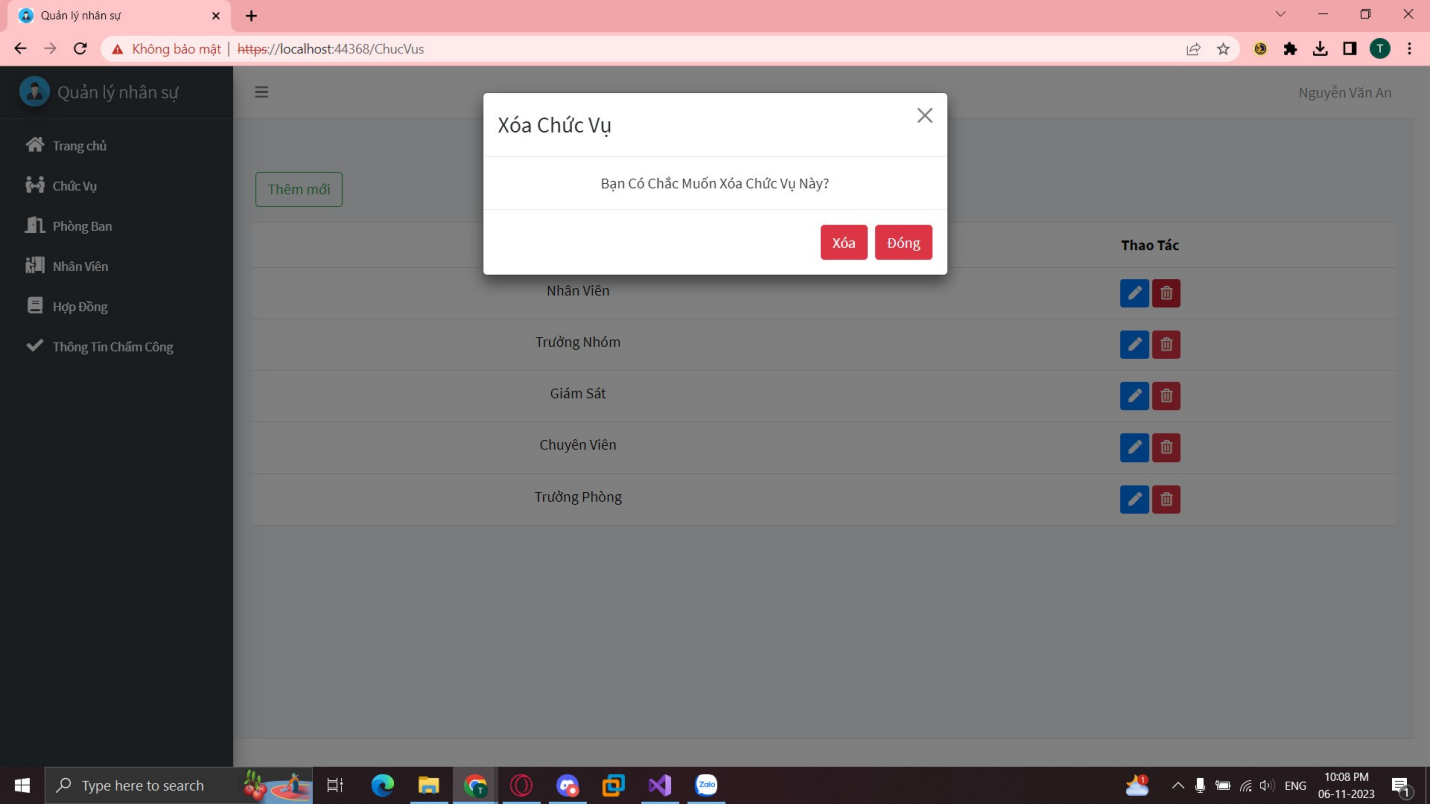
*Hình 4.15 Giao diện thêm chức vụ*

4.16 Giao diện sửa chức vụ



*Hình 4.16 Giao diện sửa chức vụ*

4.17 Giao diện xóa chức vụ



*Hình 4.17 Giao diện xóa chức vụ*

CHƯƠNG 5: KẾT LUẬN VÀ TỔNG KẾT

5.1.Kết quả đạt được:

Sau quá trình tự học hỏi , tìm kiếm và thực hiện xây dựng Web quản lý nhân sự có các chức năng cơ bản đối với các thông tin như:

Thêm sửa xóa thông tin nhân viên

Thêm sửa xóa thông tin chức vụ

Thêm sửa xóa thông tin Phòng ban

Thêm sửa xóa thông tin Hợp đồng

Sửa xóa thông tin Chấm công

5.2.Hạn chế:

-Các chức năng nâng cao, phù hợp với thực tiễn chưa xây dựng được  
 -Database còn nhỏ và chưa đầy đủ để có thể ứng dụng ra bên ngoài thực tiễn

5.3.Hướng phát triển:

-Cần tìm hiểu thêm về các nhu cầu thực tiễn

-Tiếp tục xây dựng những chức năng còn thiếu

-Phát triển thêm về giao diện

5.4.Tài liệu tham khảo:

Các tài liệu tham khảo được sử dụng trong quá trình nghiên cứu và viết đồ án:

1. Thư Viện Bootstrap: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
2. Entity Framework trong ASP.NET MVC: [https://learn.microsoft.com/en-us/aspnet/mvc/overview/getting-started/getting-started-with-ef-using-mvc/implementing-basic-crud-functionality-with-the-entity-framework-in-asp-net-mvc-application#overpost](https://learn.microsoft.com/en-us/aspnet/mvc/overview/getting-started/getting-started-with-ef-using-mvc/implementing-basic-crud-functionality-with-the-entity-framework-in-asp-net-mvc-application" \l "overpost)
3. Giới thiệu ASP.Net MVC: <https://learn.microsoft.com/en-us/aspnet/mvc/overview/getting-started/introduction/getting-started>
4. Giới thiệu SQL : <https://www.w3schools.com/sql/sql_intro.asp>
5. Tham khảo giao diện trên W3Schools: <https://www.w3schools.com/>