**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 THỰC TẬP CHUYÊN MÔN**

**ĐỀ TÀI: XÂY DỰNG WINFORM APP**

**ĐIỀU KHIỂN THIẾT BỊ ĐIỀU HOÀ NHIỆT ĐỘ**

**Thông tin sinh viên thực hiện:**

Họ tên : Phạm Nguyễn Thùy Dung Mã sinh viên : 6151071003

Lớp : CQ.61.CNTT Hệ : Chính quy

Ngành đào tạo : Công nghệ thông tin Khoá : 61

Email : 6151071003@st.utc2.edu.vn Số điện thoại : 0774757976

**Thông tin giảng viên hướng dẫn:**

Họ tên : Trần Thi Dung Học vị : Thạc sĩ

Email : ttdung@utc2.edu.vn Số điện thoại : 0388389579

Đơn vị công tác: Đại học Giao thông Vận tải Phân hiệu Thành phố Hồ Chí Minh

Họ tên : Trần Quốc Khánh Học vị : Kỹ sư

Email : tqkhanh@utc2.edu.vn Số điện thoại : 0384742790

Đơn vị công tác: Đại học Giao thông Vận tải Phân hiệu Thành phố Hồ Chí Minh

Thành phố Hồ Chí Minh, ngày 1, tháng 7, năm 2023

NHẬN XÉT CỦA GIẢNG VIÊN

**LỜI NÓI ĐẦU**

Trải qua khoảng thời gian học tập và rèn luyện của các môn như Phân Tích Thiết Kế Hệ Thống , Thiết Kế Cơ Sở Dữ Liệu, Lập Trình Web,… 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 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 Trần Thị Dung cùng thầy Trần Quốc Khánh 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 em xin gửi đến cô Trần Thị Dung cùng thầy Trần Quốc Khánh lời cảm sâu sắc vì đã tận tình hướng dẫn, truyền đạt kiến thức để chúng em có thể hoàn thành đề tài này.

Vì thời gian làm đề tài này có hạn cũng như hiểu biết cá nhân em còn hạn chế, 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. 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, em xin kính chúc cô Trần Thị Dung và thầy Trần Quốc Khánh cùng Quý thầy cô trong Bộ môn Công nghệ thông tin hạnh phúc và thành công trong công việc và trong cuộc sống.

Trân trọng cảm ơn!

**MỤC LỤC**

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.

1.2.Mục 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

1.3.Phương pháp nghiên cứu:

Phương pháp quan sát thực tế các mô hình quản lý có sẵn

Khảo sát tình hình thực tiễn, bám sát vào những nhu cầu cơ bản của doanh nghiệp để xây dựng website

Phương pháp nghiên cứu và phân tích tổng hợp

Phương pháp thu thập số liệu về thông tin cần quản lý

Phương pháp thử nghiệm cải tiến

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.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:

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

*•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 Bảo Hiểm:*

Nhân viên có thể xem thông tin Bảo Hiểm của mình như ID, Tên nhân viên, Ngày còn hiệu lực, Ngày hết hiệu lực, Tiền bảo hiểm

Admin có quyền thêm sửa xóa thông tin Bảo Hiểm.

*•Thông tin Phụ Cấp:*

Admin có quyền thêm sửa xóa thông tin Phụ Cấp

*•Thông tin Lương:*

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

3.2.Sơ đồ Diagram:

A screenshot of a computer

Description automatically generated with medium confidence

3.3.Sơ đồ UseCase:

3.3.1.Sơ đồ UseCase tổng quát:

A picture containing diagram, sketch, drawing, line

Description automatically generated

*Hình 1: Sơ đồ UseCase Tổng Quát Hệ Thống*

3.3.2.Sơ đồ UseCase Quản Lý Thông Tin Nhân Viên:

A picture containing diagram, circle, line, screenshot

Description automatically generated

*Hình 2: Sơ đồ UseCase Quản lý nhân viên Actor Nhân Viên*

A picture containing text, diagram, circle, line

Description automatically generated

*Hình 3: Sơ đồ UseCase Quản lý nhân viên Actor Admin*

3.3.3.Sơ đồ UseCase Quản Lý Thông Tin Phòng Ban:

A picture containing text, diagram, circle, line

Description automatically generated

*Hình 4: Sơ đồ UseCase Quản lý Phòng Ban Actor Admin*

3.3.4.Sơ đồ UseCase Quản Lý Thông Tin Chức Vụ:

A picture containing text, diagram, circle, line

Description automatically generated

*Hình 5: Sơ đồ UseCase Quản lý Chức Vụ Actor Admin*

3.3.5.Sơ đồ UseCase Quản Lý Thông Tin Hợp Đồng:

A picture containing text, diagram, circle, line

Description automatically generated

*Hình 6: Sơ đồ UseCase Quản lý Hợp Đồng Actor Admin*

3.3.6.Sơ đồ UseCase Quản Lý Thông Tin Bảo Hiểm:

A picture containing text, diagram, circle, line

Description automatically generated

*Hình 7: Sơ đồ UseCase Quản lý Bảo Hiểm Actor Admin*

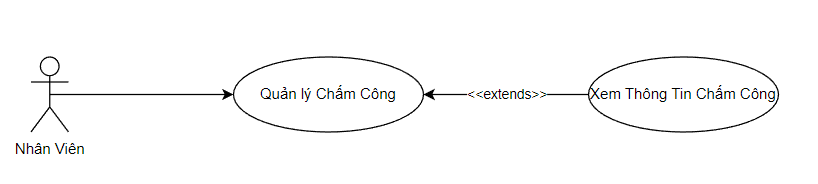
3.3.7.Sơ đồ UseCase Quản Lý Thông Tin Phụ Cấp:

A picture containing text, diagram, circle, line

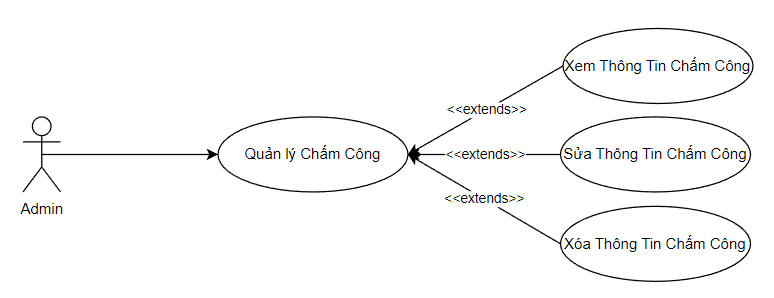
Description automatically generated

*Hình 8: Sơ đồ UseCase Quản lý Phụ Cấp Actor Admin*

3.3.8.Sơ đồ UseCase Quản Lý Thông Tin Chấm Công:



*Hình 9: Sơ đồ UseCase Quản lý Chấm Công Actor Nhân Viên*



*Hình 10: Sơ đồ UseCase Quản lý Chấm Công Actor Admin*

3.3.9.Sơ đồ UseCase Quản Lý Thông Tin Lương:

A picture containing diagram, line, circle, text

Description automatically generated

*Hình 11: Sơ đồ UseCase Quản lý Lương Actor Admin*

3.4. Mô hình ER :

A picture containing drawing, diagram, sketch, pattern

Description automatically generated

3.5. Biểu đồ phân rã chức năng (BFD):

CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH

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

Thêm sửa xóa thông tin Bảo hiểm

Thêm sửa xóa thông tin Phụ Cấp

Có thể xem thông tin Lương của nhân viên với cách tính lương cụ thể:

* Lương = Lương Cơ Bản/26 \*Số Ngày Làm + Tiền Phụ Cấp – Tiền Bảo Hiểm – Số Lần Vi Phạm \*30 (30.000đ)

Thông báo cho nhân viên biết đi làm trễ giờ quy định hoặc về sớm hơn giờ quy định

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:

Thư Viện Bootstrap: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>

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>

Giới thiệu ASP.Net MVC: <https://learn.microsoft.com/en-us/aspnet/mvc/overview/getting-started/introduction/getting-started>

Giới thiệu SQL : <https://www.w3schools.com/sql/sql_intro.asp>

Tham khảo giao diện trên W3Schools: <https://www.w3schools.com/>

5.5.Kế hoạch thực hiện và tiến độ nghiên cứu:

Thời gian và nội dung công việc theo tuần.

|  |  |  |
| --- | --- | --- |
| **Thời gian** | **Nội dung công việc** | **Ghi chú** |
| Tuần 1 (15/05 – 21/05) | Chọn đề tài, xây dựng đề cương. |  |
| Tuần 2 (22/05 – 28/05) | Tham khảo tài liệu và sách |  |
| Tuần 3 (29/05 – 04/06) | Phân tích thiết kế hệ thống và xây dựng |  |
| Tuần 4 (05/06 – 11/06) | Xây dựng cơ sở dữ liệu |  |
| Tuần 5 (12/06 – 18/06) | Xây dựng Controller |  |
| Tuần 6 (19/06 - 25/06) | Thiết kế giao diện người dùng |  |
| Tuần 7(25/06 – 29/06) | Sửa lỗi và hoàn thành các nghiệp vụ |  |