**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC KINH TẾ-KỸ THUẬT CÔNG NGHIỆP**

**Ngành đào tạo: CÔNG NGHỆ THÔNG TIN**

**Họ và tên sinh viên: Lưu Quang Tuấn – Lớp: Ti12A2HN**

**Họ và tên sinh viên: Nguyễn Đăng Nhân – Lớp: Ti12A1HN**

**Người hướng dẫn khóa luận tốt nghiệp**

**ThS. LÊ THỊ THU HIỀN**

**NGHIÊN CỨU NỀN TẢNG NODEJS XÂY DỰNG ỨNG DỤNG QUẢN LÝ NHÂN SỰ**

**Hà Nội - 2022**

TÊN ĐỀ TÀI KHÓA LUẬN:

**KHÓA LUẬN TỐT NGHIỆP**

# LỜI CAM ĐOAN

Em xin cam đoan số liệu và kết quả nghiên cứu trong khóa luận “Nghiên cứu nền tảng nodejs xây dựng ứng dụng quản lý nhân sự” là một công trình nghiên cứu độc lập dưới sự hướng dẫn của Ths. Lê Thị Thu Hiền. Đề tài, nội dung báo cáo khóa luận là sản phẩm mà em đã nỗ lực nghiên cứu trong quá trình học tập tại trường. Các tài liệu được sử dụng trong báo cáo đã trích dẫn của các tác giả liệt kê đầy đủ, không sử dụng bất cứ tài liệu nào mà không có trích dẫn. Các số liệu, kết quả trình bày trong báo cáo là hoàn toàn trung thực. Mọi sao chép không hợp lệ, vi phạm quy chế đào tạo em xin hoàn toàn chịu trách nhiệm.

Em xin chịu hoàn toàn trách nhiệm về lời cam đoan trên.

Hà Nội, ngày… tháng… năm 2020

Sinh viên

Lưu Quang Tuấn

Nguyễn Đăng Nhân

# LỜI CẢM ƠN

Em xin chân thành cảm ơn cô Ths. Lê Thị Thu Hiền. Trong suốt quá trình tiến hành khóa luận đã hướng dẫn và chỉ bảo em tận tình. Cô đã giúp em chỉ ra nhưng thiếu sót của bài luận. Em xin chân thành cảm ơn cô rất nhiều.

Đồng thời, em cũng xin chân thành cảm ơn các thầy cô khóa Công nghệ thông tin của trường Đại học Kinh tế - Kỹ thuật Công nghiệp đã dạy dỗ và truyền đạt cho em những kiến thức chuyên môn trong bốn năm học vừa qua để em có những kiến thức cơ sở và nền tảng vững chắc nhằm hoàn thành bài luận.

Em cũng xin chân thành cảm ơn sự quan tâm, động viên của gia đình, chúng ta bè đã giúp đỡ em rất nhiều và là động lực để em phấn đầu vươn lên trong thời gian vừa qua.

Mặc dù đã cố gắng hoàn thành khóa luận với tất cả sự nỗ lực của bản thân, nhưng do kiến thức và năng lực có hạn nên chắc chắn không thể tránh khỏi những thiếu sót. Kính mong quý thầy cô có thể tận tình chỉ bảo để giúp đề tài này được hoàn thiện hơn.

Hà Nội, ngày … tháng … năm 2020

Sinh viên

Lưu Quang Tuấn

Nguyễn Đăng Nhân

# NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

.................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................

Hà Nội, ngày … tháng … năm 2020

GIÁO VIÊN HƯỚNG DẪN

# NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

..................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................

Hà Nội, ngày … tháng … năm 2020

HỘI ĐỒNG PHẢN BIỆN

**MỤC LỤC**

[LỜI CAM ĐOAN ii](#_Toc99718458)

[LỜI CẢM ƠN iii](#_Toc99718459)

[NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN iv](#_Toc99718460)

[NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN v](#_Toc99718461)

[DANH MỤC HÌNH ẢNH viii](#_Toc99718462)

[DANH MỤC BẢNG BIỂU x](#_Toc99718463)

[DANH MỤC TỪ VIẾT TẮT x](#_Toc99718464)

[LỜI NÓI ĐẦU 1](#_Toc99718465)

[CHƯƠNG 1: TÌM HIỂU VỀ QUẢN LÝ NHÂN SỰ, CÁCH THỨC HOẠT ĐỘNG 3](#_Toc99718466)

[**1.1. TÌM HIỂU SƠ LƯỢC VỀ HỆ THỐNG THÔNG TIN QUẢN LÝ** 3](#_Toc99718467)

[**1.2. KHÁI NIỆM HỆ THỐNG – HỆ THỐNG THÔN TIN – HỆ THỐNG THÔNG TIN QUẢN LÝ** 3](#_Toc99718468)

[**1.3. MỘT SỐ KHÁI NIỆM LIÊN QUAN ĐẾN QUY TRÌNH QUẢN LÝ NHÂN SỰ** 3](#_Toc99718469)

[CHƯƠNG 2: GIỚI THIỆU CÁC NGÔN NGỮ LẬP TRÌNH 3](#_Toc99718470)

[**I. JAVASCRIPT, NODEJS, REACTJS** 3](#_Toc99718471)

[***1.1. TÌM HIỂU JAVASCRIPT*** 3](#_Toc99718472)

[***1.2. TÌM HIỂU VỀ NODEJS FRAMEWORK*** 7](#_Toc99718473)

[***1.3. TÌM HIỂU REACTJS*** 16](#_Toc99718474)

[**II.TÌM HIỂU VỀ C#,Python** 19](#_Toc99718475)

[***2.1. NGÔN NGỮ LẬP TRÌNH C#*** 19](#_Toc99718476)

[***2.2. TÌM HIỂU VỀ ENTITY FRAMEWORK C#, WEB API*** 21](#_Toc99718477)

[***2.3. TÌM HIỂU VỀ PYTHON*** 27](#_Toc99718478)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG WEBSITE ỨNG DỤNG QUẢN LÝ NHÂN SỰ 30](#_Toc99718479)

[3.1. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 30](#_Toc99718480)

[3.1.1. Sơ đồ phân rã chức năng 30](#_Toc99718481)

[3.1.2. Đặc tả chức năng 30](#_Toc99718482)

[3.1.3. Biểu đồ luồng dữ liệu mức ngữ cảnh 37](#_Toc99718483)

[3.1.4. Biểu đồ luồng dữ liệu mức đỉnh 38](#_Toc99718484)

[3.1.5. Biểu đồ luồng dữ liệu cho chức năng quản lý mục tiêu 39](#_Toc99718485)

[3.1.6. Biểu đồ luồng dữ liệu cho chức nằn quản lý nhân viên 39](#_Toc99718486)

[3.2. THIẾT KẾ DỮ LIỆU 40](#_Toc99718487)

[3.2.1. Mô tả các thuộc tính của thực thể 40](#_Toc99718488)

[3.2.2. Sơ đồ thực thể liên kết 44](#_Toc99718489)

[3.2.3. Các bảng dữ liệu 44](#_Toc99718490)

[CHƯƠNG 4: THIẾT KẾ GIAO DIỆN WEBSITE QUẢN LÝ NHÂN SỰ 49](#_Toc99718491)

[4.1. GIAO DIỆN CỦA QUẢN TRỊ VIÊN 49](#_Toc99718492)

[4.1.1. Giao diện đăng nhập 49](#_Toc99718493)

[4.1.2. Trang chủ 49](#_Toc99718494)

[4.1.3. Quản lý mục tiêu 50](#_Toc99718495)

[4.1.4. Quản lý nhân viên 51](#_Toc99718496)

[4.1.5. Giao việc 53](#_Toc99718497)

[4.1.6. Tiến độ thực hiện của nhân viên 53](#_Toc99718498)

[4.1.7. Đánh giá 54](#_Toc99718499)

[4.1.8. Tính lương 55](#_Toc99718500)

[4.2. GIAO DIỆN CỦA NHÂN VIÊN 55](#_Toc99718501)

[4.2.1. Trang chủ 55](#_Toc99718502)

[4.2.2. Thực hiện mục tiêu 55](#_Toc99718503)

[4.2.3. Xem và cập nhật thông tin cá nhân 56](#_Toc99718504)

[KẾT LUẬN 57](#_Toc99718505)

[TÀI LIỆU THAM KHẢO 58](#_Toc99718506)

# DANH MỤC HÌNH ẢNH

[Hình 1.1: KPI 3](#_Toc41549739)

[Hình 1.2: Tiêu chí SMART 5](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549740)

[Hình 1.3: Phương diện của KPI 10](#_Toc41549741)

[Hình 2.1: Cài đặt Composer 22](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549742)

[Hình 2. 2: Chọn đường dẫn cài đặt Composer 22](#_Toc41549743)

[Hình 2. 3: Màn hình khi gõ lệnh Composer 23](#_Toc41549744)

[Hình 2. 4: Cửa sổ Environment Varialbles 23](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549745)

[Hình 2. 5: Edit Path 24](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549746)

[Hình 2. 6: Thêm một Path mới 25](#_Toc41549747)

[Hình 2.7: Giao diện khi cài đặt thành công 26](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549748)

[Hình 2.8: Giao diện của Laravel 26](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549749)

[Hình 3. 1: Sơ đồ phân rã chức năng 29](#_Toc41549750)

[Hình 3. 2: Biểu đồ luồng dữ liệu mức ngữ cảnh 35](#_Toc41549751)

[Hình 3. 3: Biểu đồ luồng dữ liệu mức đỉnh 36](#_Toc41549752)

[Hình 3. 4: Biểu đồ luồng dữ liệu cho chức năng quản lý mục tiêu 37](#_Toc41549753)

[Hình 3. 5: Biểu đồ luồng dữ liệu cho chức năng quản lý nhân viên 37](#_Toc41549754)

[Hình 3. 6: Thực thể Staff 38](#_Toc41549755)

[Hình 3.7: Thực thể Depart 38](#_Toc41549756)

[Hình 3.8: Thực thể Target 39](#_Toc41549757)

[Hình 3.9: Thực thể Rate 39](#_Toc41549758)

[Hình 3.10: Thực thể Todo\_List 40](#_Toc41549759)

[Hình 3.11: Thực thể Targetgroup 40](#_Toc41549760)

[Hình 3.12: Thực thể Position 41](#_Toc41549761)

[Hình 3.13: Thực thể WorkPro 41](#_Toc41549762)

[Hình 3. 14: Sơ đồ thực thể liên kết 42](#_Toc41549763)

[Hình 3. 15: Mô hình liên kết 46](#_Toc41549764)

[Hình 4. 1: Giao diện đăng nhập hệ thống 47](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549765)

[Hình 4. 2: Trang chủ giao diện Admin 48](#_Toc41549766)

[Hình 4. 3: Giao diện quản lý mục tiêu 48](#_Toc41549767)

[Hình 4. 4: Giao diện thêm mới KPI 49](#_Toc41549768)

[Hình 4. 5: Giao diện quản lý nhân viên 49](#_Toc41549769)

[Hình 4. 6: Giao diện sửa thông tin nhân viên 50](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549770)

[Hình 4. 7: Giao diện thêm mới nhân viên 50](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549771)

[Hình 4. 8: Giao diện giao việc 51](#_Toc41549772)

[Hình 4. 9: Danh sách công việc và nhân viên 51](#_Toc41549773)

[Hình 4. 10: Giao diện theo dõi tiến độ thực hiện của nhân viên 52](file:///C:\Users\Luu%20Hue\Desktop\Khóa%20luận%20tn.docx#_Toc41549774)

[Hình 4. 11: Giao diện đánh giá 52](#_Toc41549775)

[Hình 4. 12: Danh sách đánh giá 52](#_Toc41549776)

[Hình 4. 13: Tính lương 53](#_Toc41549777)

[Hình 4. 14: Giao diện trang chủ của nhân viên 53](#_Toc41549778)

[Hình 4. 15: Giao diện thực hiện mục tiêu 54](#_Toc41549779)

[Hình 4. 16: Giao diện thông tin nhân viên 54](#_Toc41549780)

# DANH MỤC BẢNG BIỂU

[Bảng 3. 1: Bảng Nhân Viên 45](#_Toc99718214)

[Bảng 3. 2: Bảng Đơn xin nghỉ phép 46](#_Toc99718215)

[Bảng 3. 3: Bảng Lương Tổng 47](#_Toc99718216)

[Bảng 3. 4: Bảng Lương Làm Thêm 47](#_Toc99718217)

[Bảng 3. 5: Bảng dư án 47](#_Toc99718218)

[Bảng 3. 6: Bảng Chấm công 48](#_Toc99718219)

[Bảng 3. 7: Bảng Lương Khấu Trừ 48](#_Toc99718220)

[Bảng 3. 8: Bảng Sự Kiện 48](#_Toc99718221)

# DANH MỤC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **Viết tắt** | **Tiếng Anh** | **Tiếng Việt** |
| DOM | Document Object Model | Mô hình Các Đối tượng Tài liệu |
| RAM | Random Access Memory | Bộ nhớ lưu trữ tạm |
| HTML | Hypertext Markup Language | Ngôn ngữ Đánh dấu Siêu văn bản |
| CSS | Cascading Style Sheets | Ngôn ngữ tạo phong cách cho trang web |
| I/O | **Input/output** | đầu vào/đầu ra |
| DB | **Database** | Cơ sở dữ liệu |
| API | Application Programming Interface | Một giao diện lập trình ứng dụng |
| ORM | Object Relational Mapping | Ánh xạ quan hệ đối tượng |
| HTTP | HyperText Transfer Protocol | Giao thức truyền tải siêu văn bản |
| CLI | Command Line Interface | Giao diện dòng lệnh |
| URL | Uniform Resource Locator | Hệ thống định vị tài nguyên thống nhất |
| MVC | Model-View-Controller | mô hình thiết kế sử dụng trong kỹ thuật phần mềm |
| SPA | Single Page Application |  |
| IoT | Internet of Things | Internet vạn vật |

# LỜI NÓI ĐẦU

1. LÝ DO CHỌN ĐỀ TÀI

Quản lý nhân sự là một trong những vấn đề then chốt trong mọi công ty. Quản lý nhân sự không tốt sẽ dẫn đến nhiều vấn đề bất cập trong công ty. Chúng ta luôn thấy những tình trạng chấm nhầm công hoặc lưu trữ sai thông tin nhân viên, như vậy sẽ dẫn đến những những xáo trộn khi phát lương.... Vì vậy yêu cầu đặt ra ở đây là làm thế nào để có thể lưu trữ hồ sơ và lương của nhân viên một cách hiệu quả nhất, giúp cho nhân viên trong công ty có thể yên tâm và thoải mái làm việc đạt kết quả cao.Ngoài những lý do trên còn có một lý do khiến em quyết định chọn đề tài tốt nghiệp là “phần mềm quản lý nhân sự” là em mong muốn phần mềm quản lý nhân sự sẽ là người hỗ trợ đắc lực cho việc quản lý nhân sự của công ty. Mọi người không cần thiết phải biết nhiều về tin học, Mọi người vẫn có thể sử dụng phần mềm quản lý nhân sự một cách hiệu quả. Giao diện thuận tiện dễ sử dụng của chương trình sẽ luôn khiến Mọi người cảm thấy thật thoải mái khi sử dụng phần mềm này.Xuất phát từ những lý do trên, em đã chọn đề tài **“Nghiên cứu nền tảng NodeJS xây dựng ứng dụng quản lý nhân sự”.**

1. MỤC TIÊU NGHIÊN CỨU

* Tìm hiểu và nghiên cứu về NodeJS, Javascript, ReactJs, C#, Python.
* Tìm hiểu và nghiên cứu cách thức hoạt động của một ứng dụng quản lý nhân sự
* Xây dựng website quản lý nhân sự.

1. ĐỐI TƯỢNG VÀ NHIỆM VỤ CỦA KHÓA LUẬN

* Đối tượng:

Đối tượng nghiên cứu của khóa luận là NodeJS, Javascript, ReactJs, C#, Python, quản lý nhân sự.

* Nhiệm vụ:

Khóa luận tập trung vào xây dựng website quản lý nhân sự.

1. PHƯƠNG PHÁP VÀ NỘI DUNG NGHIÊN CỨU

* Phương pháp nghiên cứu

1. Phương pháp nghiên cứu tài liệu
2. Phương pháp thực hành

* Nội dung nghiên cứu

1. Nghiên cứu về ngôn ngữ lập trình NodeJS, Javascript, ReactJs, C#, Python
2. Tìm hiêu cách thức hoạt động của một website quản lý nhân sự
3. Xây dựng website thực nghiệm.
4. CẤU TRÚC CỦA KHÓA LUẬN

Khóa luận gồm 4 chương:

* Chương 1:Tìm hiểu về quản lý thông tin, nhân sự, cách thức hoạt động.
* Chương 2: Giới thiệu các ngôn ngữ lập trình: NodeJS, Javascript, ReactJs, C#, Python.
* Chương 3: Phân tích thiết kế hệ thống website quản lý nhân sự. Chương này trình bày về phân tích thiết kế hệ thống, các bảng dữ liệu sử dụng.
* Chương 4: Thiết kế giao diện website.

# CHƯƠNG 1: TÌM HIỂU VỀ QUẢN LÝ NHÂN SỰ, CÁCH THỨC HOẠT ĐỘNG

## **1.1. TÌM HIỂU SƠ LƯỢC VỀ HỆ THỐNG THÔNG TIN QUẢN LÝ**

* + 1. ***Tìm hiểu sợ lược về thông tin***

Thông tin là sự phản ánh sự vật, sự việc hiện tượng của thế giới khách quan và các hoạt động của con người trong đời sống xã hội. Điều cơ bản là con người thông qua việc cảm nhận làm tăng hiểu biết cho mình và tiến hành những hoạt động có ích cho cộng động.

Môi trường **vận động thông tin là môi trường truyền tin, nó bao gồm các kênh liên lạc tự nhiên hoặc nhân tạo. Kênh liên lạc thường kết nối các thiết bị của máy móc với nhau hay nối với gon người. Con người có hình thức liên lạc tự nhiên và cao cấp là tiếng nói từ đó nghĩ ra chữ viết. Ngày nay nhiều công cự phổ biển thông tin đã xuất hiện: Máy in, điện tín, phát thanh truyền hình, hình ảnh…**

* + 1. ***Khái niệm thông tin***

**Thông tin là tin tức mà con người trao đổi với nhau, hay nói rộng hơn thông tin bao gồm những tri thức về đối tượng.**

**Theo quan điểm của nhà quản lí thì thông tin là những thông báo, số liệu dung làm cơ sở cho việc ra các quyết định. Các tài liệu này chia làm 4 loại:**

* **Tài liệu, số liệu có ích cho việc ra quyết định : thông tin về năng suất, giá thành,doanh thu..**
* **Tài liệu số liệu dùg ra quyết định sau này: thông tin lưu trữ**
* **Tài liệu, số liệu không liên quan đến việc ra quyết định: tư liệu, số liệu thừa…**
* **Tài liệu, số liệu giả do các đối thủ tung ra gây nhiễu**

**Có nhiều cách phân loại thông tin. Chúng ta cần quan tâm đến cách phân loại dự vào đặc tính liên tục hay rời rạc của tín hiệu vật lý.**

* + 1. ***Các loại thông tin quản lý***

**Thông tin chiến lược là: Thông tin sử dụng cho chính sách dài hạn của tổ chức, chủ yếu phục vụ cho nhà quản lí cao cấp khi dự đoán tương lai. Loại thông tin này đòi hỏi tính khái quát, tổng hợp cao. Dự liệu để xử lý ra loại thông tin ngày thường là từ bên ngoài tổ chức. Đây là loại thông tin được cung cấp trong những trường hợp đặc biệt.**

**Thông tin chiến thuật: Là thông tin sử dụng cho chính sách ngắn hạn, chủ yếu phục vụ cho các nhà quản lí phòng ban trong tổ chức. Loại thông tin này trong khi cần mang tính tổng hợp vẫn đòi hỏi phải có mức độ chi tiết nhất định dạng thống kê**

**Thông tin điều hành: ( Thông tin tác nghiệp) sử dụng cho công tác điều hành tổ chức hàng ngày và chủ yếu phục vụ cho người giám sát hoạt động tác nghiệp của tổ chức. Loại thông tin này cần chi tiết, được rút ra từ quá trình xử lý các dữ liệu trong tổ chức. Đây là loại thông tin cần được cung cấp thường xuyên**

* + 1. ***Vai trò của thông tin trong tổ chức***

**Thông** tin có vai trò rất quan trọng đối với người lãnh đạo, quản lí trong mọi tổ chức hoạt động để hoàn thành mục tiêu ngắn hạn, trung hạn, và dài hạn. Công việc lãnh đạo và quản lí đòi hỏi thông tin phải chính xác và kịp thời đề ra các quyết định đúng đắn, đêm lại hiệu quả trong các hoạt động. Chất lượng và tính kịp thời của thông tin là yếu tố quan trọng chỉ sau yếu tố con người trong công việc lãnh đạo và quản lí. Vấn đề chủ yếu hiện nay đang đặt ra cho những người làm công tác lãnh đạo và quản lý là thường xuyên xử lý khối lượng thông tin rất lớn để đề ra được quyết định đúng đắn. Với khối lương tin càng lớn và càng đa dạng thì việc xử lý và quản lý thông tin càng khó khăn.

## **1.2. KHÁI NIỆM HỆ THỐNG – HỆ THỐNG THÔN TIN – HỆ THỐNG THÔNG TIN QUẢN LÝ**

* + 1. ***Khái niệm hệ thống***

Hệ thống: là tổng thể gồm các bộ phận hoặc các phần từ khác nhau có mối quan hệ tác động qua lại với nhau và được sắp xếp theo một trình tự nhằm tạo thành một chỉnh thể thống nhất, có những tính chất mới, có khả năng thực hiện một số chức năng hoặc mục tiêu nhất định

* + 1. ***Các yếu tố cơ bản của hệ thống***

Mô trường: Là tất cả các yếu tố, điều kiện nằm bên ngoài hệ thống có ảnh hưởng đến kết quả hoạt động của hệ thống

* Yếu tố đầu vào, ra:
  + Đầu vào: Tập hợp các tác động có thể của môi trường lên hệ thống
  + Đầu ra: Tập hợp các tác động ngược lại của hệ thống lên môi trường
* Cấu trúc hệ thống: Là sự sắp xếp các bộ phận hoặc phần tử bên trong của một hệ thống và mối quan hệ qua lại giữa chúng.
* Mục tiêu của hệ thống: Là trạng thái mong muốn của hệ thống của hệ thống sau một thời gian nhất định.
  + 1. ***Hệ thống thông tin***

Hệ thống thông tin là một tập hợp nhiều thành phần có chức năng thu nhập, lưu trữ, xử lý và phân phối thông tin để hỗ trợ việc làm quyết định điều hành trong một tổ chức.

Hệ thống thông tin tối thiểu bao gồm con người, quy trình, và dữ liệu. Con người tuân theo quy trình để xử lý dữ liệu tạo ra thông tin

* + 1. ***Hệ thông quản lý thông tin***

Hệ thống thông tin quản lí là hệ thống có nhiệm vụ cung cấp các thông tin cần thiết phục vụ cho việc quản lí điều hành một tổ chức. Thành phần chiếm vị trí quan trọng trong hệ thống thông tin quản lí là một cơ sở dữ liệu hợp nhất chứa các thông tin phản ánh cấu trúc nội tạng của hệ thống và các thông tin về các hoạt động diễn ra trong hệ thống

Với hạt nhân là cơ sở dữ liệu hợp nhất, hệ thống thông tin quản lí có thể hỗ trợ cho nhiều lĩnh vực chức năng khác nhau và có thể cung cấp cho các nhà quản lí công cụ và khả năng dễ dàng truy cập thông tin, hệ thống thông tin quản lí có các chức năng chính:

* + Thu nhập, phân tích và lưu trữ các thông tin một cách hệ thống, những thông tin có ích được cấu trúc hóa để có thể lưu trữ và khai thác trên các phương tiện tin học.
  + Thay đổi, sửa chữa, tiến hành tính toán trên các nhóm chỉ tiêu, tạo ra thông tin mới.
  + Phân phối và cung cấp thông tin.

Hệ thống thông tin thường được cấu thành bởi:

* + Các phần cứng
  + Phần mềm
  + Các hệ mạng
  + Dữ liệu
  + Con người trong hệ thống thông tin

## **1.3. MỘT SỐ KHÁI NIỆM LIÊN QUAN ĐẾN QUY TRÌNH QUẢN LÝ NHÂN SỰ**

**Nhân sự**: Là nguồm lao động của công ty, thành hay bại của công ty phụ thuộc vào yếu tố này, là một quá trình hoạt động tạo ra nền kinh tế cho bản thân và sự phát triển cho xã hội

**Lương**: là khoản tiền nhận được của nhân viên, sau khi hoàn thành công việc được dựa trên công lao động của mỗi nhân viên

**Hữu trí**: Là nhân viên khi đến độ tuổi nghỉ hưu và không thể làm việc được nữa, sẽ được hưởng chế độ này theo thâm niên công tác tại nơi làm việc

**Chế độ bảo hiểm**: Là chế dộ mà mỗi nhân viên đều được nhận, tùy theo mức lương mà nhân viên đóng phí bảo hiểm theo mức lương đo, trong đó bao gồm bảo hiểm y tế, bảo hiểm xã hội và khi không có khả năng làm việc nhan viên vẫn được hưởn chế độ này quy theo mức đóng phí của nhân viên đó

**Chế độ thai sản**: Là chế độ mà nhân viên nào nghỉ sinh cũng được nhận phụ cấp quyệt định tại nơi làm việc của mình.

**Khen thưởng**: Là hình thức khuyến khích nhan viên lao động tốt, được áp dụng cho hầu hết tất cả các công ty, mỗi nhân viên hoàn thành công việc tốt hoặc có thành tích tốt đều được nhận.

**Kỷ luật**: Là hình thức phạt mỗi khi nhân viên trong công ty vi phạm quy định của công ty hoặc làm việc không hiểu quả

# CHƯƠNG 2: GIỚI THIỆU CÁC NGÔN NGỮ LẬP TRÌNH

## **I. JAVASCRIPT, NODEJS, REACTJS**

### ***1.1. TÌM HIỂU JAVASCRIPT***

* + 1. Ngôn ngữ Javascript là gì?

Javascript chính là một ngôn ngữ lập trình với khả năng đem tới sự sinh động khi thiết kế website. Đây là dạng ngôn ngữ theo kịch bản, dựa trên chính đối trượng phát triển có sẵn, hoặc là tự định nghĩa ra. Chính vì tính tiện lợi, hiệu quả ứng dụng cao mà ngôn ngữ lập trình này được tin dùng ngày càng nhiều, ứng dụng rộng rãi trong các website hiệu quả.

Việc sử dụng ngôn ngữ Javascript có thể ứng dụng cho mọi trình duyệt khác nhau, hiện được sử dụng phổ biến như Chrome, hay Firefox,… Hơn nữa, đây còn là ngôn ngữ lập trình hoạt động hiệu quả, được hỗ trợ đầy đủ trên các trình duyệt của thiết bị di động. Bởi thế mà việc sử dụng đa dạng, có thể đáp ứng tốt cho nhiều nhu cầu, những đòi hỏi khác nhau của người dùng.

Nhìn chung, ngôn ngữ lập trình Javascript khi sử dụng được đánh giá dễ học, được phát triển bởi Netscape và sử dụng trong hơn 92% các website ngày nay. Việc sử dụng Javascript có thể gắn trong một element trong website, hoặc có thể trong một sự kiện của trang web tiện lợi và hiệu quả. Chính việc có thể hoạt động nhanh, nhẹ và mượt mà,… nên ngôn ngữ lập trình này càng được tin tưởng và ứng dụng nhiều hơn cho nhu cầu của người dùng. Chúng ta có thể thêm trực tiếp Javascript trực tiếp với HTML, cũng có thể lưu lại trên một files hoàn toàn riêng biệt và sử dụng khi cần thiết.

* + 1. Ưu nhược điểm của Javascript
* **Ưu điểm**: Khi tìm hiểu những thông tin cơ bản liên quan tới ngôn ngữ Javascript thì hiểu về ưu điểm, thế mạnh của ngôn ngữ này là không thể bỏ qua. Theo đó, với Javascript khi sử dụng mang tới cho chúng ta nhiều ưu điểm nổi bật, dưới đây là một số ưu điểm của Javascript mà Mona Media đã tổng hợp từ hơn 8 năm kinh nghiệm sử dụng chúng trong thiết kế web:
  + Sử dụng Javascript được thực hiện từ phía khách hàng: Điều này thức là các mã được tiến hành thực hiện thông qua bộ vi xử lý của chính người dùng, thay vì sử dụng trên máy chủ web. Chính đặc điểm này giúp việc tiết kiệm năng thông, đồng thời giảm những căng thẳng, hoạt động quá nhiều trên máy chủ web dễ dàng hơn.
  + Là ngôn ngữ lập trình dễ dàng sử dụng:

Đơn giản, dễ dàng tìm hiểu và sử dụng là đặc điểm nổi bật của ngôn ngữ lập trình này. Với những cú pháp khá tương đồng với tiếng Anh thì việc sử dụng Javascript trở nên dễ dàng, dễ tiếp cận hơn rất nhiều. Thông qua mô hình DOM được sử dụng, cung cấp tới nhiều tính năng hữu ích, được viết sẵn đem lại khả năng đáp ứng tốt cho những nhu cầu, những đòi hỏi khác nhau từ phía người dùng. Với nhiều tính năng hữu ích mà ngôn ngữ lập trình Javascript mang lại thì việc có thể phát triển kịch bản nhằm giải quyết yêu cầu, mục đích nào đó trở nên dễ dàng hơn. Những mục đích tùy chính của con người được đáp ứng hiệu quả với sự hỗ trợ của Javascript.

Với nhiều tính năng hữu ích mà ngôn ngữ lập trình Javascript mang lại thì việc có thể phát triển kịch bản nhằm giải quyết yêu cầu, mục đích nào đó trở nên dễ dàng hơn. Những mục đích tùy chính của con người được đáp ứng hiệu quả với sự hỗ trợ của Javascript.

* + Đánh giá cao ở độ nhanh chóng với người dùng cuối: Việc sử dụng ngôn ngữ Javascript có khả năng đảm bảo khi mã được thực hiện trên thiết bị máy tính của người dùng thì quá trình xử lý, lẫn kết quả đều hoàn thành gần như ngay lập tức. Thời gian cụ thể phụ thuộc vào từng nhiệm vụ cụ thể mất thời gian nhất định, song thường nó rất nhanh có thể giúp vấn đề được giải quyết.
  + Mang tới các tính năng bổ sung cho các website:

Thông qua bên thứ ba như Greasemonkey giúp đảm bảo cho các nhà phát triển ngôn ngữ Javascript dễ dàng viết các đoạn mã, từ đó thực hiện trên website hiệu quả như mong muốn. Việc mở rộng thêm các tính năng của trang web trở nên đơn giản và dễ dàng hơn rất nhiều.

Đối với những trang web khi sử dụng yêu cầu cần sở hữu tính năng nhất định thì chúng ta hoàn toàn có khả năng tự viết, đồng thời sử dụng thêm trình bổ sung Greasemonkey nhằm giúp việc thực hiện thuận lợi, trên website chúng ta đang dùng.

* + Sử dụng đa dạng trên nhiều trình duyệt: Một ưu điểm nổi bật giúp Javascript ngày càng được tin dùng, ứng dụng phong phú là ở khả năng sử dụng thông qua nhiều trình duyệt khác nhau. Từ các trình duyệt trên máy tính như Chrome, tới Firefox,… hay những trình duyệt trên thiết bị di động,… đều có thể sử dụng được đầy đủ và hiệu quả. Chính vì sự thích hợp với nhiều trình duyệt, nhiều nền tảng khác nhau giúp nó được tin dùng nhiều hơn để đáp ứng tốt cho đòi hỏi của con người.
  + Dễ dàng khi sử dụng với ngôn ngữ Javascript:

Sử dụng Javascript đảm bảo dễ học, dễ dàng phát hiện lỗi khi xuất hiện, đồng thời nó hoạt động nhanh chóng và nhẹ nhàng hơn nhiều ngôn ngữ lập trình khác. Bởi thế, quá trình sử dụng có thể thích hợp với nhiều đối tượng người dùng, đem lại hiệu quả cao và tránh những ảnh hưởng không mong muốn có thể xảy ra.

* + Sở hữu giao diện thân thiện, nhiều tính năng:

Nếu so sánh với nhiều ngôn ngữ lập trình khác thì việc sử dụng Javascript mang tới ưu điểm ở giao diện thân thiện, có thể phù hợp với mọi đối tượng người dùng. Đem tới tính tương tác cao giúp quá trình ứng dụng của mỗi người có được kết quả tốt đẹp. Cùng với nhiều tính năng mà ngôn ngữ này mang tới chắc chắn sẽ giúp website chúng ta tạo ra có được hiệu quả sử dụng lý tưởng.

Có thể kể tới như việc tiến hành thêm slider, hay slideshow, thực hiện hiệu ứng cuộn trang khi chuột di tới, tính năng kéo thả,… đều được cung cấp đầy đủ. Từ đó giúp việc sử dụng ngôn ngữ này trong lập trình đem lại lợi ích lớn, thiết thực.

* **Nhược điểm:** Tồn tại song song với những ưu điểm là hạn chế mà ngôn ngữ lập trình Javascript đang phải đối mặt. Việc tìm hiểu và xác định được những hạn chế giúp chúng ta chủ động trong đánh giá, đưa ra quyết định sử dụng sao cho thích hợp nhất.
  + Là ngôn ngữ lập trình dễ bị khai thác: Việc dễ dàng bị khai thác, với nhiều đối tượng người dùng khác nhau vô tình khiến việc bảo mật cho ngôn ngữ lập trình này không được đánh giá cao. Bởi thế, quá trình sử dụng luôn tồn tại những nguy cơ nhất định cần chúng ta tìm hiểu và chủ động chú ý cẩn trọng.
  + Một hạn chế của ngôn ngữ Javascript chính là việc chúng ta có thể sử dụng để thực thi mã độc trên máy tính của người dùng. Tình trạng này vô tình gây ra những thiệt hại, những ảnh hưởng không hề nhỏ tới công việc mà chúng ta đang thực hiện.
  + Trong một số trường hợp, trong một vài trình duyệt cụ thể việc sử dụng ngôn ngữ lập trình Javascript không thể thực hiện được. Nó gây ra ảnh hưởng tới việc ứng dụng, đáp ứng cho nhu cầu của từng người.
  + Sử dụng ngôn ngữ lập trình này có một hạn chế khác khi có Javascript code snippets khá lớn.
  + Khi có khả năng triển khai khác nhau tùy từng thiết bị, nó vừa là ưu điểm song cũng tồn tại những hạn chế nhất định. Chính điều này khiến nguy cơ không đồng nhất có thể xảy ra, gây ảnh hưởng tới quá trình thực hiện mục tiêu cuối cùng.
    1. Hoạt động của ngôn ngữ Javascript trên website

Sử dụng ngôn ngữ Javascript hiện nay trở nên phổ biến, thông dụng đối với nhiều người dùng. Đối với ngôn ngữ lập trình này chúng được tiến hành nhúng trực tiếp vào trang web, hoặc được tiến hành tham chiếu thông qua một file.js riêng biệt. Đây là ngôn ngữ phía client, điều này tức là script được tải đầy đủ về máy của khách hàng khi truy cập. Đồng thời, nó được xử lý ngay tại đó, thay vì bên server là thực hiện xử lý ngay trên server trước khi đưa kết quả tới khách hàng truy cập.

Việc sử dụng ngôn ngữ Javascript hỗ trợ đầy đủ việc khách hàng muốn tắt hay mở trên các trình duyệt web được ứng dụng cơ bản hiện nay. Nhờ vậy, việc có thể xác định được website hoạt động như thế nào, tình hình thực tế ra sao khi không có ngôn ngữ Javascript đang hoạt động.

Việc sử dụng ngôn ngữ Javascript hỗ trợ đầy đủ việc khách hàng muốn tắt hay mở trên các trình duyệt web được ứng dụng cơ bản hiện nay. Nhờ vậy, việc có thể xác định được website hoạt động như thế nào, tình hình thực tế ra sao khi không có ngôn ngữ Javascript đang hoạt động.

* Sử dụng Javascript cùng HTML, hay CSS để trở thành ngôn ngữ không thể thay thế đối với một website.
* Có khá nhiều những framework JavaScript khách nhau ở phía front-end, trong đó tiêu biểu như: ReactJS, Vue.js, Angular.js, Angular2, Meteor.js,…
* Ngôn ngữ lập trình Javascript được sử dụng ở phía các server với framework như Node.js.
* Có một số database sử dụng ngôn ngữ Javascript dưới dạng kịch bản, đồng thời được coi là ngôn ngữ query như CouchDB, MongoDB,…
* Dùng ngôn ngữ Javascript ứng dụng trong việc xây dựng ứng dụng của Desktop với framework là Electron, tiêu biểu là những ứng dụng tiêu biểu, nổi tiếng như WordPress.com, Atom, Visual Studio Code,…
* Sử dụng ngôn ngữ JavaScript trong việc xây dựng nên các ứng dụng trên điện thoại, đảm bảo đa nền tảng cho Android, hay iOS,… hiệu quả nhằm đáp ứng tốt cho nhu cầu, đòi hỏi khác nhau của người dùng.

### ***1.2. TÌM HIỂU VỀ NODEJS FRAMEWORK***

**1.2.1**. NodeJS là gì?

Nodejs là một nền tảng (Platform) phát triển độc lập được xây dựng trên V8 JavaScript Engine – trình thông dịch thực thi mã JavaScript giúp chúng ta có thể xây dựng được các ứng dụng web như các trang video clip, các forum và đặc biệt là trang mạng xã hội phạm vi hẹp một cách nhanh chóng và dễ dàng mở rộng.

NodeJS có thể chạy trên nhiều nền tảng hệ điều hành khác nhau từ Window cho tới Linux, OS X nên đó cũng là một lợi thế. NodeJS cung cấp các thư viện phong phú ở dạng Javascript Module khác nhau giúp đơn giản hóa việc lập trình và giảm thời gian ở mức thấp nhất.

Node.js được xây dựng và phát triển từ năm 2009, bảo trợ bởi công ty Joyent, trụ sở tại California, Hoa Kỳ.

Ý tưởng chính của Node js là sử dụng non-blocking, hướng sự vào ra dữ liệu thông qua các tác vụ thời gian thực một cách nhanh chóng. Bởi vì, Node js có khả năng mở rộng nhanh chóng, khả năng xử lý một số lượng lớn các kết nối đồng thời bằng thông lượng cao.

Nếu như các ứng dụng web truyền thống, các request tạo ra một luồng xử lý yêu cầu mới và chiếm RAM của hệ thống thì việc tài nguyên của hệ thống sẽ được sử dụng không hiệu quả. Chính vì lẽ đó giải pháp mà Node js đưa ra là sử dụng luồng đơn (Single-Threaded), kết hợp với non-blocking I/O để thực thi các request, cho phép hỗ trợ hàng chục ngàn kết nối đồng thời.

**1.2.2**. Cấu trúc NodeJS

- Mô-đun: Module giống như các thư viện JavaScript sử dụng trong ứng dụng Node.js application bao gồm một bộ các chức năng. Để đưa một module vào ứng dụng Node.js thì phải sử dụng hàm require() function with the với dấu ngoặc đơn chứa tên module.

* Console: Bảng điều khiển cung cấp phương thức gỡ lỗi tương tự như bảng điều khiển cơ bản của JavaScript trên các trình duyệt internet. Nó sẽ in các thông báo ra stdout và stderr.
* Cluster: Node.js được xây dựng dựa trên ý tưởng lập trình đơn luồng. Cluster là một module cho phép đa luồng bằng cách tạo ra các quy trình con có chung cổng máy chủ và chạy đồng thời.
* Global: Biến toàn cục trong Node.js tồn tại trong tất cả các module. Những biến này bao gồm các hàm, module, string, v…v….

Lỗi trong Node.js được sử lý qua các exception. Chẳng hạn, chúng ta phải xử lý một lỗi xảy ra khi chia cho 0. Lỗi này sẽ làm crash ứng dụng Node.js, vì vậy chúng ta phải xử lý lỗi này để ứng dụng có thể hoạt động bình thường.

**1.2.3.** NodeJS framework sử dụng phổ biến

1.2.3.1. Express

Khi nói đến các framework Node.js phổ biến nhất, Express là lựa chọn đầu tiên của các nhà phát triển vì nó được 73% các nhà phát triển yêu thích và sử dụng.

Express là một trong những Node.Js Framework động cung cấp sự linh hoạt hoàn toàn và sự tự do cho các nhà phát triển trong việc tùy chỉnh giao diện của ứng dụng.

Vì nó nhẹ, do đó, nó cực kỳ nhanh và giúp định hướng các máy chủ và bộ định tuyến. Ngoài ra, khung này lý tưởng cho các ứng dụng web một trang, trang web hoặc các API HTTP công khai.

* Lợi ích của Express:
  + Cung cấp hỗ trợ cho các plugin và tiện ích mở rộng khác nhau để tăng cường chức năng của nó
  + Tích hợp dữ liệu liền mạch
  + Một đường cong học tập dễ dàng cho các nhà phát triển đã quen thuộc với Node.Js
  + Cơ chế định tuyến dựa trên URL sử dụng HTTP
  + Mạnh mẽ, có thể mở rộng và hoàn toàn có thể tùy chỉnh
  + Đi kèm với MVC giúp tạo các ứng dụng tập trung

Hỗ trợ hơn 14 công cụ mẫu và mang lại hiệu suất tuyệt vờiExpress rất phù hợp cho tất cả các loại phát triển web và ứng dụng di động ngay từ cấp nhỏ đến cấp doanh nghiệp. Vì khung công tác này của Node.Js có sẵn để tạo API, do đó, chúng ta có thể phát triển ứng dụng web nhanh hơn với Express Ngoài ra, khung công tác này đi kèm với tính năng định tuyến mạnh mẽ, các tính năng bảo mật và điều khoản xử lý lỗi, vì vậy các nhà phát triển có thể dễ dàng sử dụng nó để chế tạo cấp doanh nghiệp hoặc ứng dụng dựa trên trình duyệt.

1.2.3.2 socketIO

Socket.io là một full-stack framework có thể được sử dụng bởi nhiều ngôn ngữ lập trình. Nó cung cấp một API đơn giản và dễ hiểu, dễ dàng hoạt động hài hòa với hầu hết mọi dịch vụ. Với tốc độ phát triển vượt trội và độ tin cậy của nó, các nhà phát triển thực sự ấn tượng với framework này.

* Lợi ích của việc sử dụng socketIO:
  + Cung cấp phân tích thời gian thực
  + Phát trực tuyến nhị phân và quản lý ổ cắm dễ dàng
  + Khả năng điều chỉnh định tuyến URL cho các ổ cắm web
  + Tự động phát hiện và sửa lỗi Socket.io về cơ bản được sử dụng để phát triển các ứng dụng thời gian thực như API bảng điều khiển, Phiếu điểm, Chatbots, ứng dụng hội nghị truyền hình và hơn thế nữa, trong đó máy chủ được yêu cầu để đẩy dữ liệu mà không cần yêu cầu từ phía máy khách.
* NPM: The Node Package Manager

Trình quản lý gói Node.js (NPM) là trình quản lý gói mặc định và phổ biến nhất trong hệ sinh thái Node.js và chủ yếu được sử dụng để cài đặt và quản lý các module bên ngoài trong dự án Node.js. Nó cũng thường được sử dụng để cài đặt một loạt các công cụ CLI và chạy các tập lệnh dự án. NPM theo dõi các module được cài đặt trong một dự án bằng file package.json , nằm trong folder của dự án và chứa:

* + Tất cả các module cần thiết cho một dự án và các version đã cài đặt của chúng
  + Tất cả metadata cho một dự án, chẳng hạn như tác giả, giấy phép, v.v.
  + Các tập lệnh có thể được chạy để tự động hóa các việc trong dự án
    1. Công dụng của NodeJS
* Backend for Social Media Networking: Nhiều tên tuổi nổi tiếng như LinkedIn và Medium khác được xây dựng bởi Node.js. Có một sự kết hợp thiết yếu của các nút để tạo phần phụ trợ cho một trang mạng truyền thông xã hội. Node.js cung cấp khả năng định tuyến cực nhanh với động cơ V8 cùng với xác thực an toàn. Khả năng mở rộng là một trong những yếu tố quan trọng nhất làm phong phú thêm mạng truyền thông xã hội.
* Single-page Application (SPA) Development: Cũng giống như các trang web đơn lẻ, Node.js có thể xây dựng một ứng dụng một trang, nơi giao diện tương tự như một ứng dụng dành cho máy tính để bàn. Do các tùy chọn linh hoạt của nó, Node.js rất phù hợp để xây dựng nền tảng mạng xã hội, trang web động và giải pháp gửi thư. Hơn nữa, chất lượng luồng dữ liệu không đồng bộ trên phần phụ trợ trong Node.js khiến nó trở thành lựa chọn tốt nhất để phát triển SPA.
* Chatbots: Node.js cung cấp các chức năng nâng cao độc quyền cho chatbots và ứng dụng trò chuyện thời gian thực. Các tính năng chính như ứng dụng nhiều người dùng, dữ liệu chuyên sâu, lưu lượng truy cập lớn, trên tất cả các thiết bị đều rất phổ biến trong chatbots. Node.js đã đạt được điểm tốt, vì nó chạy trên các thiết bị khác nhau và bao gồm tất cả các mô hình trong ứng dụng trò chuyện và chatbot. Node.js giúp việc thực thi các thông báo đẩy và vòng lặp sự kiện phía máy chủ trở nên cực kỳ dễ dàng được sử dụng phổ biến trong IMs và ứng dụng real-time.
* Data Streaming: Những gã khổng lồ như Netflix đã chào đón Node.js để phục vụ 190 quốc gia và hơn 100 triệu giờ phát trực tuyến trên 120 triệu người dùng và còn tiếp tục tăng. Node.js có một lợi thế rõ ràng về điều này là các tệp xử lý ở tốc độ siêu nhanh, đồng thời mã hóa và tải lên nhẹ nhàng. Điều này đã được nhiều cửa hàng điện tử thời trang trực tuyến và các ứng dụng có video truyền tải dữ liệu khổng lồ áp dụng.
* IoT Application Development: IoT đã trở nên phổ biến mạnh mẽ trong thập kỷ gần đây và kể từ đó Node.js là một trong những giải pháp được ưa thích nhất cho các tổ chức muốn xây dựng các hệ thống giải pháp IoT công cộng và riêng tư. Lợi thế cốt lõi của Node.js là khả năng xử lý các yêu cầu đồng thời liền mạch với hàng nghìn sự kiện được phát hành bởi hàng tỷ thiết bị trên mạng của nó. Các mạng IoT. Hơn nữa, Node.js hoạt động trên các kênh và luồng có thể ghi và đọc được, đó là lý do tại sao nó là nền tảng phù hợp nhất để phát triển ứng dụng IoT.
  + 1. Ưu điểm của NodeJS
* Khả năng mở rộng: Các ứng dụng Node.js có thể dễ dàng mở rộng, theo chiều dọc và chiều ngang. Trong khi quy mô theo chiều dọc cho phép chúng ta thêm nhiều tài nguyên hơn vào các nút hiện tại, quy mô theo chiều ngang cho phép chúng ta thêm các nút mới nhanh hơn. Hơn nữa, các ứng dụng Node.js không yêu cầu một khối lớn trong toàn bộ quá trình phát triển, vì nó hoạt động cùng với một tập hợp các microservices và mô-đun. Nó đơn giản, dễ dàng và hoàn toàn phù hợp cho các công ty khởi nghiệp đang muốn phát triển.
* Hiệu suất cao: Node.js chạy trên công cụ Google V8 JS tuân thủ mã Javascript trực tiếp vào mã máy. Điều này làm cho nó hiệu quả về tài nguyên cũng như tăng tốc độ của khung. Trên thực tế, bản thân Google đầu tư rất nhiều vào công cụ tìm kiếm của họ để cải thiện hiệu suất một cách nhất quán
* Hỗ trợ cộng đồng lớn: Được thành lập vào năm 2015, Node.js Foundation là một nhóm các nhà sáng lập ấn tượng đến từ các doanh nghiệp hàng đầu như IBM, Microsoft, Fidelity và SAP. Một số lượng lớn các nhà phát triển phần mềm và nhiều người hâm mộ Node.js tích cực đang không ngừng đóng góp vào cộng đồng Node.js không ngừng phát triển và phát triển để hỗ trợ các nhà phát triển đồng nghiệp. Ngoài ra, Node.js được hỗ trợ trên Github, giúp việc phát triển và hỗ trợ framework trở nên cực kỳ dễ dàng và nhanh chóng.
* Tăng tốc độ phát triển: Node.js cho phép chúng ta các nhà phát triển đẩy nhanh quá trình phát triển ứng dụng web bằng các công cụ nhẹ và đa dạng của nó. Sổ đăng ký npm cung cấp nhiều giải pháp, mô-đun và thư viện để tạo các ứng dụng web bằng cách thu được nhiều giá trị hơn và đầu tư với ít nỗ lực hơn. Node.js, cùng với các thư viện npm, giúp chúng ta giảm lỗi và giảm kích thước ứng dụng web của mình bằng các mẫu có thể sử dụng lại của nó. Sự tối ưu hóa này ảnh hưởng trực tiếp đến thời gian đưa ra thị trường.
* Tự do phát triển ứng dụng: Các nhà phát triển làm việc với Node.js có quyền tự do phát triển các ứng dụng đa nền tảng. Với các khuôn khổ như Electron và NW.js, chúng ta có thể tạo các ứng dụng web đa nền tảng với Node.js vì nó tương thích với máy tính để bàn, thiết bị di động và phát triển web cho Mac, Windows và Linux. Node.js hiện cũng được lựa chọn rộng rãi cho các giải pháp IoT. Vì vậy, nếu chúng ta đang tìm cách nâng cao hiệu suất và hiệu suất mã hóa của nhóm, thì Node.js là lựa chọn tốt nhất dành cho chúng ta.
* Ngôn ngữ lập trình đơn: Node.js cung cấp một loạt các ngôn ngữ tương tự trên Javascript front-end và back-end. Nó thực sự dễ tiếp cận và linh hoạt hơn âm thanh. Nó làm cho toàn bộ chu trình phát triển ứng dụng web trở nên mạnh mẽ và liền mạch cho các nhà phát triển và doanh nghiệp. Nó giúp tiết kiệm thời gian, năng lượng và tài nguyên đầu tư vào giờ làm việc hiệu quả hơn và có lợi nhuận. Node.js không chỉ giúp cho việc phát triển ứng dụng web trở nên thuận tiện mà còn khiến nó trở thành công nghệ được ưa chuộng nhất trên thị trường hiện nay.
* Đóng góp cho Fullstack & MEANstack: Node.js đã chứng minh được tiềm năng của nó và là một phần của hệ thống công nghệ mạnh mẽ với Javascript là ngôn ngữ lập trình hàng đầu không thể tranh cãi. Node.js, là một công cụ độc lập, mang lại nhiều lợi ích cho ngăn xếp công nghệ; chẳng hạn, nó cung cấp thư viện mô-đun JSON. Với hơn 368,985,988 lượt tải xuống, tức là trung bình 1,010,921 mỗi ngày, Node.js đã trở thành một người chơi quan trọng trong nhiều hệ sinh thái công nghệ.
* Hệ sinh thái phong phú: Khi nói đến sự đơn giản và phong phú của Node.js, npm là từ duy nhất xuất hiện trong đầu chúng ta. Chúng ta có biết không? npm có 836.000 thư viện và 10.000 trở lên được xuất bản mỗi tuần. Đúng! hơn 475.000 gói và đăng ký có sẵn và 11.000 công cụ mới hàng tuần. Nó đang giảm rất nhiều thời gian và chi phí liên quan đến việc phát triển và đưa ra thị trường.

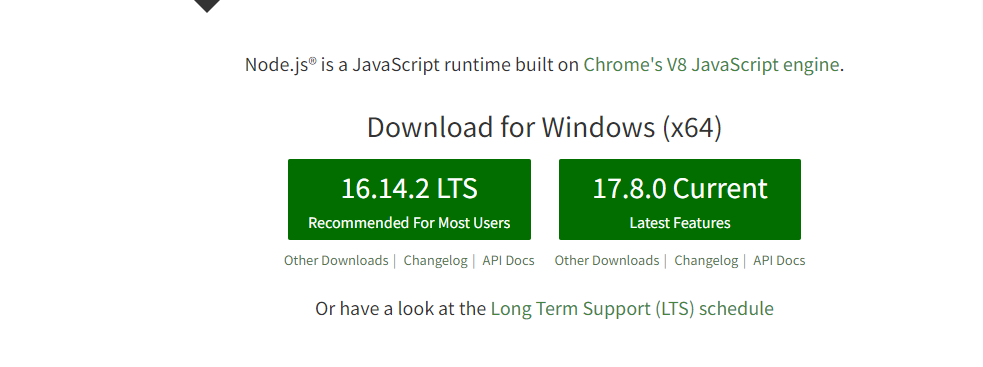
Node.js được 98% công ty trong danh sách Fortune 500 sử dụng thường xuyên, theo Node.js Foundation? Lý do? Nhẹ, nhanh và có thể mở rộng. Node.js cũng hoạt động hiệu quả và hiệu quả trong việc xử lý nhiều yêu cầu cùng một lúc. Việc xử lý lưu lượng truy cập ứng dụng sẽ tiết kiệm thời gian và chi phí hơn các công nghệ khác.

* + 1. Cài đặt nodeJS trên Window

Để cài đặt NodeJS trên Window thực hiện các bước như sau:

* **Bước 1**: Download NodeJS từ trang chủ Nodejs.org

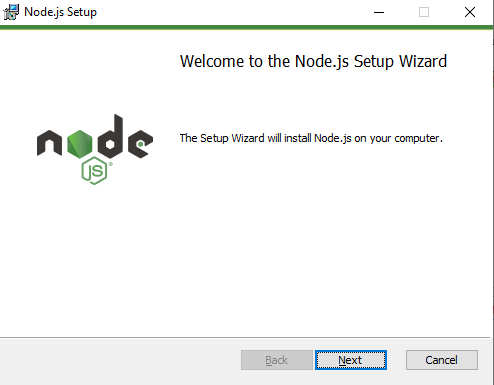
Truy cập vào địa chỉ [**nodejs.org**](https://nodejs.org/en/) sẽ xuât hiện một trang web với giao diện như sau:



Website này đã phát hiện đang sử dụng hệ điều hành Window nên đã cố ý show gói download dành cho Window luôn. Trường hợp muốn download các gói khác thì vào đường dẫn [nodejs.org/en/download](https://nodejs.org/en/download/) để tìm gói cần tải.

* **Bước 2**: Cài đặt phần mềm NodeJS:

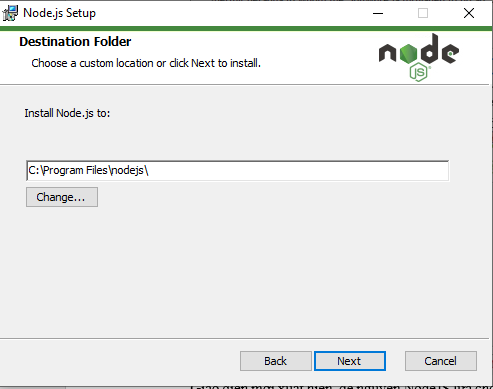
Chạy file vừa download và thực hiện cài đặt, lúc này hệ thống xuất hiện giao diện sau:



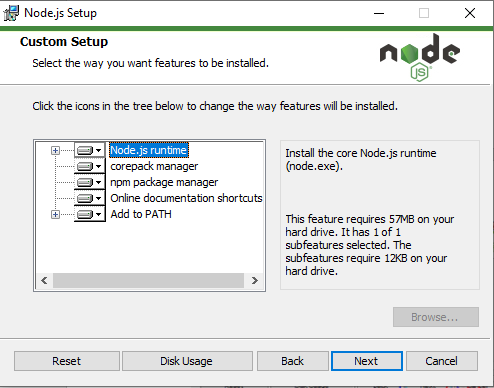
Click **Next**, một giao diện mới xuất hiện.



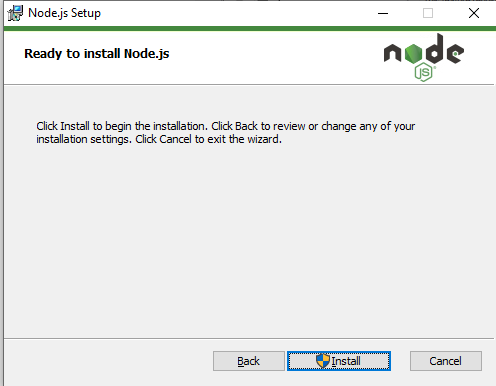
Chúng ta check vào **ô I accept the terms in the License Agreement** và click **Next**. Giao diện mới xuất hiện, chọn đường dẫn lưu và click **Next**.



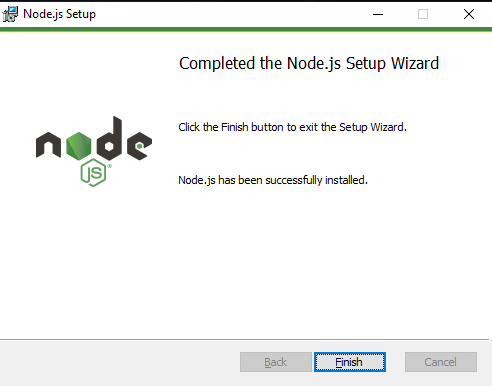
Giao diện mới xuất hiện, để nguyên NodeJS lựa chọn các gói mặc định, click vào nút **Next**.



Tiếp tục click vào nút **Install** để cài đặt.



Công đoạn cài đặt sẽ mất khoảng một phút đổ lại nên vui lòng chờ nó chạy xong và click vào nút **Finish**.



### ***1.3. TÌM HIỂU REACTJS***

**1.3.1**. ReactJS là gì?

ReactJS hay react là một thư viện, được viết bằng JavaScript, dùng để xây dựng giao diện người dùng. Tính đến thời điểm hiện tại, đã có khoảng 1300 developer và hơn 94000 trang web đang sử dụng. ReactJS được hiểu nôm na là một thư viện, có chứa nhiều JavaScript mã nguồn và “cha đẻ” đó chính là Facebook. Mục đích chính đó chính là mỗi website sử dụng reactJS thì phải chạy thật mượt hoặc thật nhanh, có khả năng mở rộng cao và đơn giản thực hiện.

Về cơ bản, các tính năng của reactJS thường xuất phát từ việc tập trung các phần mềm riêng lẻ, cho phép các developer có chức năng phá vỡ giao diện của người dùng từ một cách phức tạp và biến nó trở thành các phần mềm đơn giản. Hiểu đơn giản thì các render dữ liệu không chỉ được thực hiện ở vị trí server mà còn ở vị trí client khi sử dụng reactJS.

Hiện nay, reactJS thường được dùng để thiết kế bố cục cho trang web, nhưng hạn chế đó chính là cấu trúc khó. Thay vào đó sử dụng JSX và nhúng vào các đoạn HTML và javascript, mọi người sẽ thấy cú pháp dễ hiểu hơn và JSX cũng có thể tối ưu các code khi biên soạn, vừa giúp ích cho người lập trình tiện cho việc biên dịch.

**1.3.2.** Các thành phần chính trong ReactJS

- Redux. Chính là một predictable state management tool cho ứng dụng javascrip. Đây là thành phần vô cùng quan trọng đối với reactJS giúp các ứng dụng hoạt động một cách nhất quán. Redux bao gồm actions, store, reducers. Một reactJS không bao gồm module chuyên dụng để xử lý dữ liệu. Do đó, reactJS được thiết lập độc lập bằng việc chia nhỏ view thành các component để chúng dễ dàng liên kết chặt chẽ với nhau.

- JSX. JSX cho phép lập trình viên trích dẫn HTML và sử dụng các cú pháp của thẻ HTML để render các subcomponent. Sự có mặt của JSX giúp tối ưu hóa code nên web sẽ chạy nhanh hơn so với code javascript tương đương.

- Single – way data flow. Là luồng dữ liệu một chiều từ component cha xuống component con. Cơ chế này sẽ phát huy được vai trò quan trọng của mình khi cấu trúc và chức năng của view trở nên phức tạp.

- Virtual DOM. Lợi thế khi sử dụng virtula – DOM là làm cho tốc độ ứng dụng được tăng đáng kể. Do virutal – DOM vừa đóng vai trò model vừa đóng vai trò là view nên mọi sự thay đổi trên model kéo theo cả sự thay đổi của view và ngược lại. Điều này có nghĩa là view vẫn luôn thực hiện được cơ chế data – binding mặc dù lập trình viên không tác động trực tiếp vào các phần tử DOM ở view.

**1.3.3**. Những điều cần biết về ReactJS

- ReactJS là một thư viện giúp cho các lập trình dễ dàng quản lý, phát triển web. Ngoài ra, nó còn hỗ trợ xây dựng UI khi có tính tương tác cao. Xây dựng các quanh các component, khi hoạt động react hoạt động trên cả client và render trên cả sever để cả 2 phần này có thể kết nối được với nhau.

- ReactJS sử dụng ít state components. Vì state sẽ làm test trở nên khó khăn hơn. Nó liên quan đến việc render hiển thị đã khởi tạo cho state hay chưa và có sự thay đổi nào hay không?. Bên cạnh đó, state chỉ tồn tại trong một components để trao đổi các dữ liệu bên ngoài nên việc sử dụng state là điều không cần thiết.

- Luôn giữ components nhỏ gọn. Một phần mềm nhỏ gọn thì càng dễ hiểu hơn. Còn với reactJS thì việc giữ component được nhỏ nhất là để tái sử dụng và phát huy được công năng cao nhất.

- Kết hợp reactJS với redux.js. Redux là một nguồn dữ liệu luồng được nhiều người sử dụng để giúp quá trình đóng gói và tận dụng các thư viện trở nên triệt để hơn. Ngoài ra, còn có thể sử dụng kết hợp với jsx, ES6, babel,…

- Trình Dev Tool của redux và react. Để có thể nhanh chóng tìm được các lỗi trong ứng dụng và inspect ngược lại các component của react thì không thể không tận dụng hai nhân tố trên. Không những thế, trình dev tools của redux và react còn giúp các nhà phát triển web quan sát được trạng thái action và các lý do thay đổi state, back lại state trước.

**1.3.4.** Ưu, nhược điểm của ReactJS

- ƯU điểm:

+ Vì ReactJS sử dụng DOM ảo để cache cấu trúc dữ liệu trong bộ nhớ và chỉ những thay đổi cuối cùng mới được cập nhật vào trong DOM trình duyệt. Điều này làm cho ứng dụng trở nên nhanh hơn.

+ Cthể tạo các component theo từng chức năng mà mình muốn bằng cách sử dụng tính năng react component. Các component này có thể tái sử dụng theo nhu cầu của mỗi người, đồng thời việc tạo các component theo từng chức năng cũng giúp cho việc bảo trì sau này trở nên dễ dàng hơn.

+ ReactJS là một opensource, vì vậy cũng rất dễ cho những người mới bắt đầu tìm hiểu nó.

+ Trong những năm gần đây, ReactJS đang trở nên phổ biến hơn và được duy trì bởi Facebook và Instagram. Ngoài ra nó cũng được sử dụng bởi các công ty nổi tiếng như Apple, Netflix, …

+ Facebook vẫn đang duy trì, phát triển, và cho ra những thay đổi mới. Vì thế chúng ta cứ yên tâm sử dụng ReactJS cho những dự án của bản thân hoặc những dự án dành do doanh nghiệp.

+ ReactJS có thể được sử dụng để xây dựng giao diện người dùng cho cả các ứng dụng dành cho máy tính và các ứng dụng di động.

+ Dễ dàng cho việc test và debug, vì hầu hết các code đều được thực hiện bằng JavaScript chứ không phải bằng HTML.

* Nhược điểm:

+ Vì hầu hết code được viết dưới dạng JSX, tức là HTML và CSS là một phần của JavaScript, nó không giống như những framework khác vẫn tách biệt giữa HTML và CSS nên những người mới làm quen với ReactJS sẽ hơi lúng túng và dễ nhầm lẫn giữa JSX và HTML. Tuy nhiên sẽ nhanh chóng quen với cách kết hợp này của React mà thôi.

+ Một nhược điểm nữa của ReactJS đó là dung lượng các file của nó hơi lớn.

## **II.TÌM HIỂU VỀ C#,Python**

### ***2.1. NGÔN NGỮ LẬP TRÌNH C#***

**2.1.1** C# là gì?

C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000. C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.

Trong các ứng dụng Windows truyền thống, mã nguồn chương trình được biên dịch trực tiếp thành mã thực thi của hệ điều hành. Trong các ứng dụng sử dụng .NET Framework, mã nguồn chương trình (C#, VB.NET) được biên dịch thành mã ngôn ngữ trung gian MSIL (Microsoft intermediate language).

2.1.2 Đặc trưng của C#

* C# là ngôn ngữ đơn giản:

C# loại bỏ một vài sự phức tạp và rối rắm của những ngôn ngữ như Java và c++, bao gồm việc loại bỏ những macro, những template, đa kế thừa, và lớp cơ sở ảo (virtual base class).

Ngôn ngữ C# đơn giản vì nó dựa trên nền tảng C và C++. Nếu chúng ta thân thiện với C và C++ hoặc thậm chí là Java, chúng ta sẽ thấy C# khá giống về diện mạo, cú pháp, biểu thức, toán tử và những chức năng khác được lấy trực tiếp từ ngôn ngữ C và C++, nhưng nó đã được cải tiến để làm cho ngôn ngữ đơn giản hơn.

- C# là ngôn ngữ hiện đại:

Điều gì làm cho một ngôn ngữ hiện đại? Những đặc tính như là xử lý ngoại lệ, thu gom bộ nhớ tự động, những kiểu dữ liệu mở rộng, và bảo mật mã nguồn là những đặc tính được mong đợi trong một ngôn ngữ hiện đại. C# chứa tất cả những đặc tính trên. Nếu là người mới học lập trình có thể chúng ta sẽ cảm thấy những đặc tính trên phức tạp và khó hiểu. Tuy nhiên, cũng đừng lo lắng chúng ta sẽ dần dần được tìm hiểu những đặc tính qua các nội dung khoá học này.

- C# là một ngôn ngữ lập trình thuần hướng đối tượng:

Lập trình hướng đối tượng (OOP: Object-oriented programming) là một phương pháp lập trình có 4 tính chất. Đó là tính trừu tượng (abstraction), tính đóng gói (encapsulation), tính đa hình (polymorphism) và tính kế thừa (inheritance). C# hỗ trợ cho chúng ta tất cả những đặc tính trên.

- C# là một ngôn ngữ ít từ khóa:

C# là ngôn ngữ sử dụng giới hạn những từ khóa. Phần lớn các từ khóa được sử dụng để mô tả thông tin. Chúng ta có thể nghĩ rằng một ngôn ngữ có nhiều từ khóa thì sẽ mạnh hơn. Điều này không phải sự thật, ít nhất là trong trường hợp ngôn ngữ C#, chúng ta có thể tìm thấy rằng ngôn ngữ này có thể được sử dụng để làm bất cứ nhiệm vụ nào.

2.1.3. Nền tảng .Net

- .NET Framework được Microsoft đưa ra chính thức từ năm 2002. .NET Framework chỉ hoạt động trên Windows. Những nền tảng ứng dụng như WPF, Winforms, ASP.NET(1-4) hoạt động dựa trên .NET Framework.

- Mono là phiên bản cộng đồng nhằm mang .NET đến những nền tảng ngoài Windows. Mono được phát triển chủ yếu nhằm xây dựng những ứng dụng với giao diện người dùng và được sử dụng rất rộng rãi: Unity Game, Xamarin…

- Cho đến năm 2013, Microsoft định hướng đi đa nền tảng và phát triển .NET core. .NET core hiện được sử dụng trong các ứng dụng Universal Windows platform và ASP.NET Core. Từ đây, C# có thể được sử dụng để phát triển các loại ứng dụng đa nền tảng trên các hệ điều hành khác nhau (Windows, Linux, MacOS,…)

**2.1.4.** Ứng dụng của C#

* Phát triển web backend (ASP.NET MVC, ASP.NET core, Web API,Graph API,gPRC, Blazor sevver,Uno platform, Mono)
* Phát triển web front end (Blazor WebAssembly,Uno platform)
* Phát triển desktop app (Winform, WPF, UWP,WinUI, Mono, Uno, MAUI,Blazor desktop...)
* Phát triển game 2D, 3D đa nền tảng (Game engine:Unity, Monogame, Godot, Stride, CryEngine, NeoAxis..)
* Phát triển thực tế ảo (VR), thực tế tăng cường(AR), thực tế hỗn hợp (MR) (HoloLens,Unity,CryEngine, Oculus quest..)
* Phát triển ứng dụng đồ họa 2D,3D đa nền tảng (2D: SkiaSharp, ImageSharp...; 3D: OpenTK, SharpDX, SharpVulkan, Vulkan.NET, Veldrid, Silk.NET..)
* Phát triển mobile app, IOS native, Android native (Xamarin,MAUI,Uno platform)
* Phát triển đám mây (Azure,AWS,Google Cloud...)
* Học máy và trí tuệ nhân tạo (ML.Net, TensorFlow, csiSharp..)
* Data science, bigdata (csiSharp, Apache Spark)
* Hệ thống nhúng (Raspberry pi, PLC)
* Microservices and containers

***2.2. TÌM HIỂU VỀ ENTITY FRAMEWORK C#, WEB API***

**2.2.1**. Entity Framework là gì?

Entity Framework được phát hành lần đầu tiên vào năm 2008, nó là phương tiện tương tác chính giữa các ứng dụng .NET và cơ sở dữ liệu quan hệ.

Entity Framework là một ánh xạ quan hệ đối tượng (ORM), là một loại công cụ giúp đơn giản hóa ánh xạ giữa các đối tượng trong phần mềm với các bảng và cột của cơ sở dữ liệu quan hệ.

* Entity Framework (EF) là một framework ORM mã nguồn mở độc lập với .NET Framework.
* Entity Framework đảm nhiệm việc tạo các kết nối cơ sở dữ liệu và thực thi các lệnh, cũng như lấy kết quả truy vấn và tự động ánh xạ các kết quả đó thành các đối tượng trong ứng dụng
* Entity Framework giúp theo dõi các thay đổi của đối tượng và cập nhật các thay đổi đó trở lại cơ sở dữ liệu

**2.2.2.** Tại sao nên sử dụng Entity Framework

Entity Framework được thiết kế nhằm mục đích tăng năng suất của nhà phát triển bằng cách giảm các công việc dư thừa như là duy trì dữ liệu được sử dụng trong các ứng dụng.

* Entity Framework có thể tạo các câu lệnh truy cấn cần thiết để đọc hoặc ghi dữ liệu trong cơ sở dữ liệu và thực thi chúng cho người sử dụng.
* Nếu cần truy vấn dữ liệu, chúng ta có thể sử dụng LINQ to Entities
* Entity Framework sẽ thực hiện các truy vấn có liên quan trong cơ sở dữ liệu và sau đó ánh xạ kết quả trả về thành các đối tượng để chúng ta làm việc trong ứng dụng của mình.

Có các ORM khác trên thị trường như NHibernate và LLBLGen Pro. Hầu hết các ORM thường ánh xạ các loại miền trực tiếp vào lược đồ cơ sở dữ liệu.

Entity Framework có lớp ánh xạ chi tiết hơn để chúng ta có thể tùy chỉnh ánh xạ. Ví dụ, bằng cách ánh xạ một thực thể vào nhiều bảng trong cơ sở dữ liệu hoặc thậm chí nhiều thực thể vào một bảng.

* ADO.NET là công nghệ được sử dụng để làm việc trực tiếp cơ sở dữ liệu ở mức thấp.
* Microsoft khuyên chúng ta nên sử dụng Entity Framework thay vì sử dụng ADO.NET hoặc LINQ to SQL cho tất cả những ứng dụng mới.

**2.2.3.** Tính năng của Entity Framework

Sau đây là các tính năng cơ bản của Entity Framework. Danh sách này được tạo dựa trên các tính năng đáng chú ý nhất và cũng từ các câu hỏi thường gặp về Entity Framework.

* Entity Framework là một công cụ của Microsoft.
* Entity Framework đang được phát triển như một sản phẩm mã nguồn mở.
* Entity Framework không còn bị ràng buộc hoặc phụ thuộc vào chu kỳ phát hành của .NET Framework.
* Làm việc với bất kỳ cơ sở dữ liệu quan hệ nào có Entity Framework provider hợp lệ.
* Tạo truy vấn SQL từ LINQ to Entities.
* Entity Framework sẽ tạo các truy vấn có tham số.
* Theo dõi các thay đổi của các đối tượng trong bộ nhớ.
* Cho phép tạo các câu lệnh thêm, cập nhật và xóa.
* Entity Framework hỗ trợ stored procedure.

**2.2.4**. Kiến trúc Entity Framework

Kiến trúc của Entity Framework được minh họa trong hình sau:

* Application: Application (ứng dụng) là tầng chứa giao diện tương tác với người dùng, có thể là ứng dụng web hoặc ứng dụng Windows Forms. Tầng này thường sử dụng Repository pattern để làm việc với cơ sở dữ liệu thông qua Entity Framework.
* Object Services: Object Services là tầng chứa quá trình tương tác giữa ứng dụng và database, hay nói cách khác nó là nơi chủ yếu để truy cập dữ liệu từ database và trả ngược kết quả về giao diện. Object Services cung cấp các tiện ích để truy vết các thay đổi và cập nhật thay đổi trở lại ở database.
* EntityClient Data Provider: Đây là tầng cung cấp các kết nối, diễn dịch các truy vấn thực thể thành truy vấn nguồn dữ liệu, trả về data reader để EF dùng chuyển dữ liệu thực thể thành các đối tượng. Phần này kết nối ADO.NET Data Providers để gửi hoặc lấy dữ liệu từ database.
* ADO.NET Data Providers: Đây là tầng thấp nhấp để dịch các truy vấn LINQ to Entity thành các câu lệnh SQL và thực thi các câu lệnh trong hệ thống DBMS. Tầng này kết với database sử dụng ADO.NET.
* Entity Data Model (EDM): Entity Data Model gồm 3 phần chính: mô hình khái niệm (CSDL – Conceptual schema definition language), mô hình ánh xạ (MSL – mapping specification language) và mô hình lưu trữ (SSDL – store schema definition language). Entity Data Model khác với EntityClient Data Provider ở chỗ nó sử dụng LINQ là ngôn ngữ truy vấn tương tác với database.
* Mô hình khái niệm (CSDL): Mô hình khái niệm chứa các lớp mô hình và mối quan hệ giữa các lớp này. Nó độc lập với mô hình quan hệ các bảng trong database
* Mô hình lưu trữ (SSDL): Mô hình lưu trữ là 1 mô hình thiết kế database bao gồm các table, view, stored procedure, mối quan hệ giữa chúng các bảng và khóa. Mô hình này thể hiện gần giống mô hình quan hệ các bảng trong database
* Mô hình ánh xạ (MSL): Mô hình lưu trữ là 1 mô hình thiết kế database bao gồm các table, view, stored procedure, mối quan hệ giữa chúng các bảng và khóa. Mô hình này thể hiện gần giống mô hình quan hệ các bảng trong database

**2.2.5.** Web API

**2.2.5.1.** Web API là gì?

Web API là một phương thức dùng để cho phép các ứng dụng khác nhau có thể giao tiếp, trao đổi dữ liệu qua lại. Dữ liệu được Web API trả lại thường ở dạng JSON hoặc XML thông qua giao thức HTTP hoặc HTTPS.

**2.2.5.2.** Những điểm nổi bật của web api

- Tự động hóa sản phẩm

- Khả năng tích hợp linh động

- Cập nhật thông tin thời gian thực

- Có tiêu chuẩn chung dễ sử dụng

**2.2.5.3.** Cách thức hoạt động của Web API

- Đầu tiên là xây dựng URL API để bên thứ ba có thể gửi request dữ liệu đến máy chủ cung cấp nội dung, dịch vụ thông qua giao thức HTTP hoặc HTTPS.

- Tại web server cung cấp nội dung, các ứng dụng nguồn sẽ thực hiện kiểm tra xác thực nếu có và tìm đến tài nguyên thích hợp để tạo nội dung trả về kết quả.

- Server trả về kết quả theo định dạng JSON hoặc XML thông qua giao thức HTTP/HTTPS.

- Tại nơi yêu cầu ban đầu là ứng dụng web hoặc ứng dụng di động , dữ liệu JSON/XML sẽ được parse để lấy data. Sau khi có được data thì thực hiện tiếp các hoạt động như lưu dữ liệu xuống Cơ sở dữ liệu, hiển thị dữ liệu…

**2.2.5.4.** Ưu và nhược điểm của Web API

**\*Ưu điểm:**

**-** Web API được sử dụng hầu hết trên các ứng dụng desktop, ứng dụng mobile và ứng dụng website.

- Linh hoạt với các định dạng dữ liệu khi trả về client: Json, XML hay định dạng khác.

- Nhanh chóng xây dựng HTTP service: URI, request/response headers, caching, versioning, content formats và có thể host trong ứng dụng hoặc trên IIS.

- Mã nguồn mở, hỗ trợ chức năng RESTful đầy đủ, sử dụng bởi bất kì client nào hỗ trợ XML, Json.

- Hỗ trợ đầy đủ các thành phần MVC như: routing, controller, action result, filter, model binder, IoC container, dependency injection, unit test.

- Giao tiếp hai chiều được xác nhận trong các giao dịch, đảm bảo độ tin cậy cao.

**\*Nhược điểm:**

- Web API chưa hoàn toàn phải là RESTful service, mới chỉ hỗ trợ mặc định GET, POST

- Để sử dụng hiệu quả cần có kiến thức chuyên sâu, có kinh nghiệm backend tốt

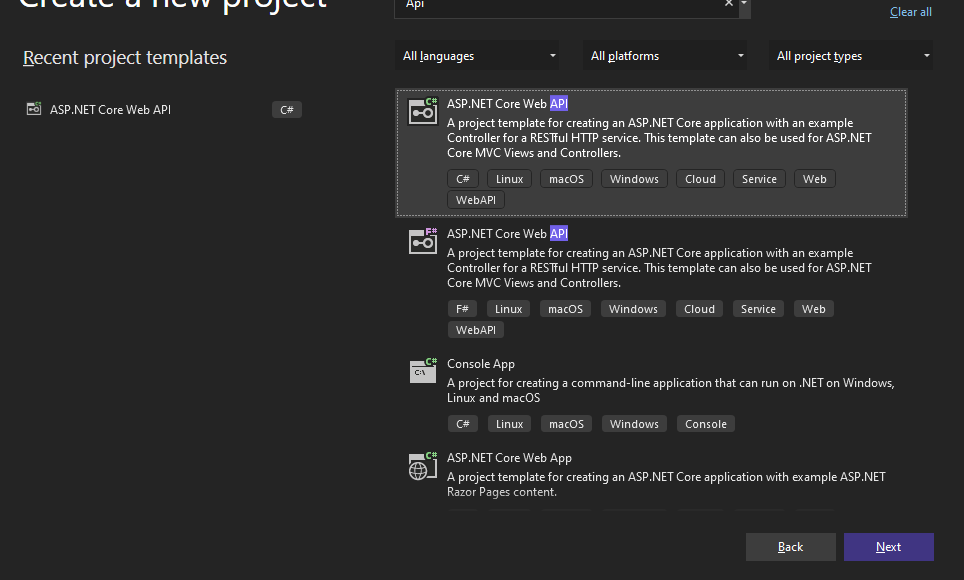
- Tốn thời gian và chi phí cho việc phát triển, nâng cấp và vận hành

- Có thể gặp vấn đề về bảo mật khi hệ thống bị tấn công nếu không giới hạn điều kiện kỹ.

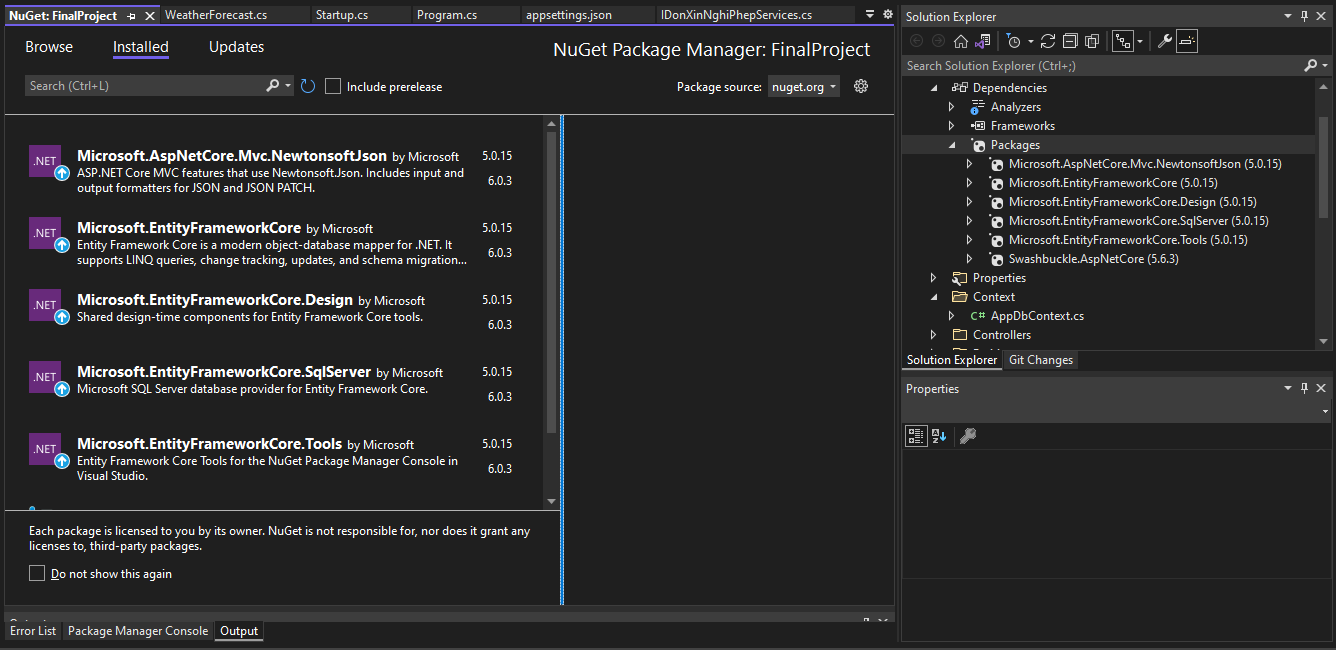
**2.2.6.** Cài đặt Web API Entity Framework

Để cài đặt chương trình Web API Entity Framework như sau:

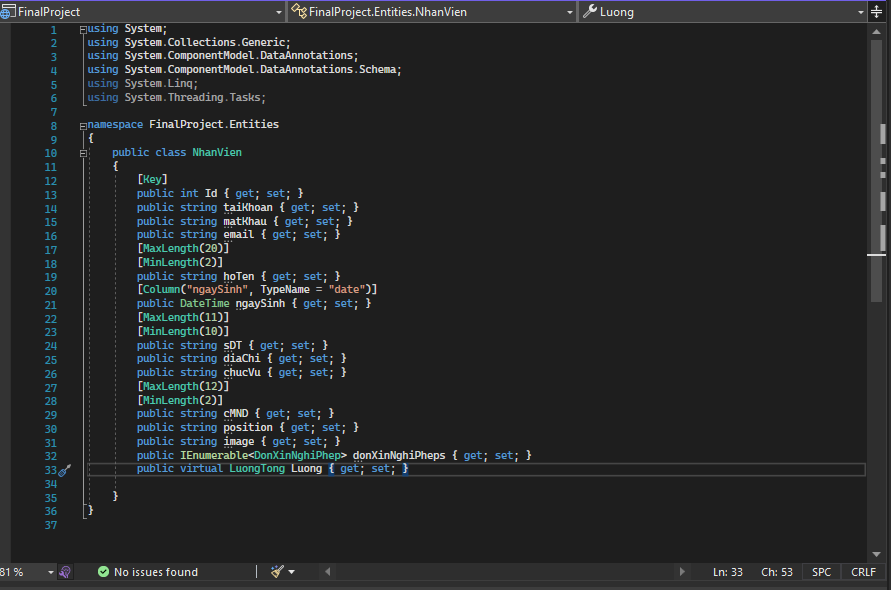
Bước 1: Tạo project ASP.NET Core Web API:

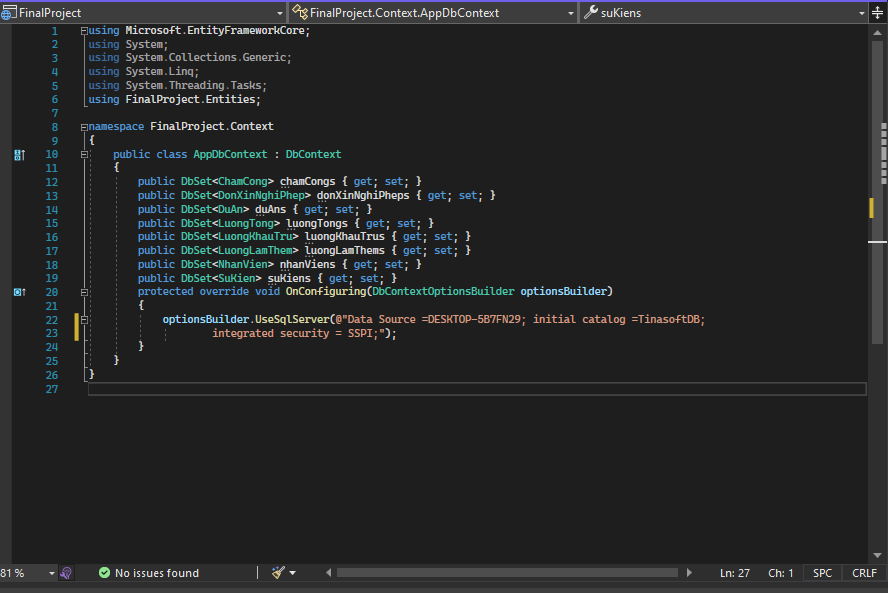


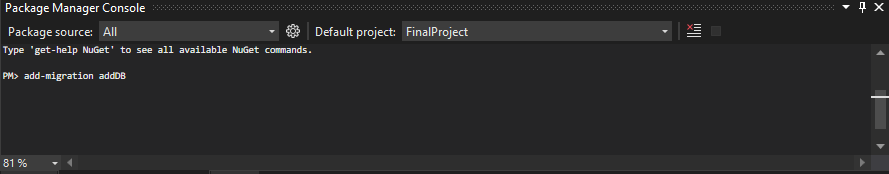
Bước 2: Cài đặt các gói package:

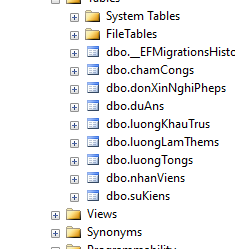


Bước 3: Tạo các Entities ( Các bảng trong SQL) và kết nối với cơ sở dữ liệu:









***2.3. TÌM HIỂU VỀ PYTHON***

**2.3.1.** Khái niệm Python

Là ngôn ngữ lập trình bậc cao, phục vụ cho các mục đích lập trình đa năng. Ưu điểm nổi bật nhất đó chính là dễ đọc, dễ nhớ, dễ học. Python là ngôn ngữ có cấu trúc tương đối rõ ràng, thuận tiện cho người mới học lập trình. Cấu trúc của python còn cho phép người dùng sử dụng để viết mã lệnh với số lần gõ phím tối thiểu.

Nói cách khác, khái niệm lập trình python là gì còn được hiểu là cửa ngõ để mọi người bước vào thế giới lập trình máy tính hay là một phương tiện để nhận được một khoản lương “đáng mơ ước” từ một công việc sáng tạo.

Python mang bản chất là ngôn ngữ lập trình bậc cao, được tạo ra bởi Guido van Rossum. Ngôn ngữ python được thiết kế hướng tới đối tượng với cấu trúc hàng và cách xử lý dữ liệu đơn giản, dễ đọc. Nó sẽ giúp người dùng tạo ra những chương trình hay với số lượng dòng code ít nhất.

Python hiện có 2 version phổ biến đó chính là python 2 và python 3. Một số hệ điều hành Linux cũ không được hỗ trợ python 3 mà cần phải cài đặt. Do đó người dùng nên lựa chọn một phiên bản của python để viết lập trình.

=> Python là ngôn ngữ lập trình với mã nguồn mở đa mục đích, hỗ trợ nhiều phong cách lập trình cốt lõi khác nhau, thiết kế của python đem tới sự thuận lợi trong việc đọc hiểu code, đơn giản và rõ ràng.

**2.3.2**. Các tính năng nổi bật của phần mềm python.

Ngôn ngữ lập trình python có các tính năng nổi bật sau:

* ***Miễn phí, mã nguồn mở:*** Chúng ta có thể thỏa mái sử dụng và phân phối python thậm chí là có thể sử dụng chúng để phục vụ cho mục đích thương mại. Bởi chúng là mã nguồn mở, chúng ta không chỉ sử dụng các phần mềm, chương trình được viết trong python mà còn có thể thay đổi mã nguồn. Python có một cộng đồng lớn, thường xuyên cập nhật, không ngừng cải tiến.
* ***Ngôn ngữ lập trình đơn giản, dễ đọc:*** Python có cấu trúc ngữ ngáp đơn giản, rõ ràng. Nó dễ đọc và viết đơn giản hơn nhiều khi so sánh với ngôn ngữ lập trình khác như C ++, Java, C#. Python làm cho việc lập trình trở nên thú vị, giúp các lập trình viên tập trung vào những giải pháp chứ không phải là cú pháp.
* ***Khả năng di chuyển:*** Các chương trình trên python có thể di chuyển từ nền tảng này đến nền tảng khác mà không gặp phải bất kỳ thay đổi nào khi chạy. Nó chạy liền mạch trên các nền tảng như Mac, Windows, Linux.
* ***Khả năng mở rộng và có thể nhúng:*** Nếu một ứng dụng đòi hỏi sự phức tạp lớn, chúng ta có thể dễ dàng kết hợp với các phần code bằng C, C ++ và những ngôn ngữ khác vào code python. Bởi vậy, sẽ giúp ứng dụng của chúng ta có những tính năng tốt hơn, khả năng scripting mà các ngôn ngữ lập trình khác khó có thể làm được.
* ***Ngôn ngữ thông dịch cấp cao:*** Với python chúng ta không cần phải lo lắng về việc quản lý bộ nhớ, dọn dẹp những dữ liệu vô nghĩa,…Khi chạy, ngôn ngữ python sẽ tự động chuyển đổi code sang ngôn ngữ để máy tính có thể hiểu và chúng ta cũng không cần lo lắng về bất kỳ hoạt động ở cấp thấp nào.
* ***Hướng đối tượng:*** Mọi thứ của python đều hướng đối tượng. Lập trình đối tượng sẽ giúp giải quyết các vấn đề một cách trực quan nhất. Với lập trình đối tượng, chúng ta có thể phân chia nhiều phức tạp thành các tập nhỏ hơn bằng cách tạo ra các đối tượng.
* ***Thư viện tiêu chuẩn lớn*:** Ngôn ngữ python có thư viện lớn giúp cho việc lập trình trở nên dễ dàng hơn vì không phải tự viết tất cả các code. Những thư viện này được kiểm tra kỹ lưỡng, nên chắc chắn nó không làm hỏng code hay ứng dụng nào của chúng ta.

**2.3.3.** Ưu điểm của Python

- ***Cú pháp đơn giản, dễ hiểu:*** Hiện nay, có nhiều ngôn ngữ lập trình khác nhau nhưng python vẫn được lựa chọn hơn cả nhờ cú pháp đơn giản, dễ sử dụng. Nó gần như tiếng anh và có thể đọc hiểu, viết nhanh ngay cả đối với những ai chưa có kinh nghiệm lập trình.

- ***Cấu trúc rõ ràng, ngắn gọn:*** Nếu như chúng ta đã từng làm việc với các ngôn ngữ C, C++, Java,…ít nhiều cũng sẽ phải đau đầu khi gặp phải các lỗi sai cú pháp, sai hàm. Nhưng với python thì hoàn toàn khác, công việc này sẽ được thực hiện tự động không còn có các dấu dấu chấm phẩy cuối dòng lệnh hay không cần phải xác định kiểu biến,…

- ***Giảm lượng code cần viết:*** Python cho phép người dùng viết các chương trình có chức năng tốt hơn với ít dòng code hơn.

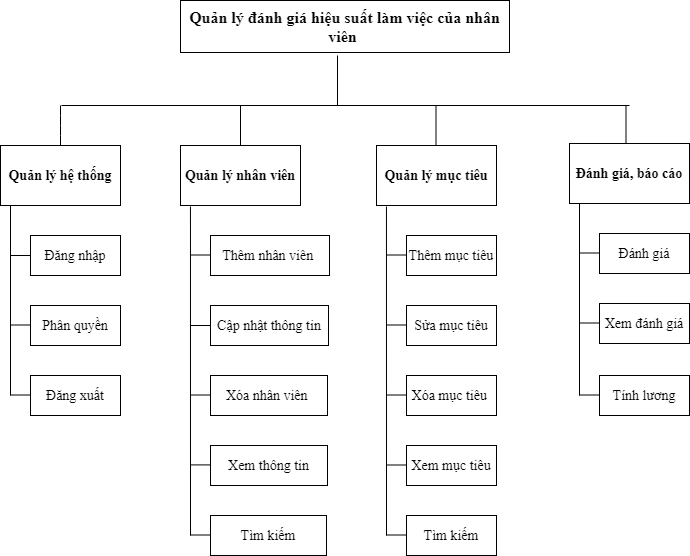
- ***Cộng đồng hỗ trợ tốt hơn:*** Python có một cộng đồng hỗ trợ rất lớn, có cả các diễn đàn hoạt động trực tuyến giúp chúng ta giải quyết các vấn đề nào đó trong quá trình sử dụng python.

**2.3.4.** Cài đặt Python

# CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG WEBSITE ỨNG DỤNG QUẢN LÝ NHÂN SỰ

## 3.1. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

3.1.1. Sơ đồ phân rã chức năng



Hình 3. : Sơ đồ phân rã chức năng

3.1.2. Đặc tả chức năng

3.1.2.1 Đối với quản trị viên

* Đăng nhập vào hệ thống quản trị Admin

|  |  |
| --- | --- |
| Miêu tả | Chức năng đăng nhâp vào hệ thống quản trị Admin dành cho những ai có tài khoản Admin. Chức năng cho phép quản trị viên đăng nhập vào giao diện quản trị website |
| Đầu vào | Người dùng tiến hành điền thông tin vào form đăng nhập gồm tên và mật khẩu |
| Xử lý | * Hệ thống thực hiện việc kiểm tra dữ liệu đầu vào đã đúng định dạng hay chưa, đã nhập đủ các trường yêu cầu hay chưa. * Tiếp đó hệ thống sẽ tiến hành kiếm tra xem có thông tin của admin nào tương ứng với tên và mật khẩu người dùng nhập hay không. Nếu có thì thực hiện đăng nhập, nếu sai sẽ thông báo kết quả cho người dùng |
| Xuất | Thông báo kết quả cho người dùng và thực hiện đăng nhập vào hệ thống |

* Đăng xuất khỏi hệ thống

|  |  |
| --- | --- |
| Miêu tả | Chức năng đăng xuất khỏi hệ thống quản trị website đối với những ai đã có tài khoản admin và đăng nhập hệ thống. Chức năng cho phép người dùng có thể đăng xuất tài khoản ra khỏi hệ thống giao diện admin của website. |
| Đầu vào | Người dùng thực hiện click vào nút đăng xuất |
| Xử lý | Hệ thống thực hiện việc thoát tài khoản ra khỏi giao diện quản trị Admin. |
| Xuất | Thông báo kết quả cho người dùng và quay lại giao diện đăng nhập tài khoản. |

* Quản lý mục tiêu

|  |  |
| --- | --- |
| Miêu tả | Chức năng quản lý mục tiêu cho phép người quản trị Admin có thể thêm, sửa, xóa các mục tiêu trong cơ sở dữ liệu. |
| Đầu vào | Chọn vào nút được thiết kế các chức năng thêm, sửa, xóa tương ứng. |
| Xử lý | * Thêm   Ở bước này, admin sẽ nhập các thông tin cần thiết, hệ thống sẽ kiểm tra tính đúng đắn về kiểu dữ liệu nhập vào, các trường bắt buộc phải nhập và kiểm tra các trường dữ liệu đã bị trùng và thông báo cho admin biết khi có lỗi.  Hệ thống thực hiện thêm dữ liệu vào cơ sở dữ liệu.   * Sửa   Chức năng này cho phép admin sửa dữ liệu của các bản ghi đã được lưu trữ ở trong cơ sở dữ liệu.  Hệ thống kiểm tra tính đúng của dữ liệu và thực hiện lưu dữ liệu vào cơ sở dữ liệu.   * Xóa   Admin sẽ click vào nút xóa tương ứng với bản dữ liệu muốn xóa, hệ thống sẽ hỏi có chắc chắn xóa hay không.  Nếu người dùng ấn Yes thì thực hiện xóa dữ liệu trong cơ sở dữ liệu, nếu No thì không thực hiện. |
| Xuất | Thông báo kết quả thành công hoặc lỗi cho người dùng. |

* Quản lý nhân viên

|  |  |
| --- | --- |
| Miêu tả | Chức năng này cho phép người quản trị Admin có thể thêm, sửa, xóa thông tin nhân viên. |
| Đầu vào | Chọn vào nút được thiết kế sẵn để thực hiện các chức năng thêm, sửa, xóa tương ứng. |
| Xử lý | * Thêm nhân viên   Admin tiến hành nhập dữ liệu thông tin về nhân viên muốn thêm, hệ thống sẽ kiểm tra tính đúng của dữ liệu nhập vào.  Hệ thống thực hiện thêm tài khoản vào cơ sở dữ liệu.   * Sửa   Chức năng này cho phép admin sửa dữ liệu của các bản ghi đã được lưu trong cơ sở dữ liệu.  Hệ thống sẽ tiến hành kiểm tra dữ liệu sửa và tiến hành lưu dữ liệu vào trong cơ sở dữ liệu.   * Xóa   Admin sẽ kích vào nút xóa tương ứng với bản dữ liệu muốn xóa, hệ thống sẽ hỏi có muốn xóa hay không.  Nếu người dùng kích Yes thì thực hiện xóa dữ liệu trong cơ sở dữ liệu, nếu No thì không thực hiện gì cả. |
| Xuất | Thông báo kết quả thành công hoặc lỗi cho người dùng. |

* Tìm kiếm

|  |  |
| --- | --- |
| Miêu tả | Chức năng này cho phép Admin có thể tìm kiếm mục tiêu, tìm kiếm nhân viên theo tên mục tiêu, tên nhân viên được nhập vào từ ô tìm kiếm. |
| Đầu vào | A Admin tiến hành nhập dữ liệu và kích chọn vào ô tìm kiếm được thiết kế sẵn trên giao diện. |
| Xử lý | Admin xem được thông tin mà mình yêu cầu. |
| Xuất | Hiển thị kết quả tìm kiếm. |

* Đánh giá

|  |  |
| --- | --- |
| Miêu tả | Chức năng này cho phép Admin có thể đánh giá hiệu quả làm việc của nhân viên thông qua công việc đã giao trước đó. |
| Đầu vào | Admin tiến hành xem chi tiết mục tiêu được giao cho nhân viên và tiến hành đánh giá thông qua nút được thiết kế sẵn. |
| Xử lý | Admin đánh giá dựa trên kết quả làm việc của nhân viên. |
| Xuất | Hiển thị kết quả. |

* Tính lương

|  |  |
| --- | --- |
| Miêu tả | Chức năng này cho phép Admin có thể tính lương của nhân viên thông qua hiệu suất đã được đánh giá trước đó. |
| Đầu vào | Tổng hiệu suất đã được đánh giá, lương cơ bản của nhân viên, lương thưởng và trừ. |
| Xử lý | Phần mềm tính toán theo công thức có sẵn. |
| Xuất | Hiển thị lương nhân viên |

* Nhắc nhở

|  |  |
| --- | --- |
| Miêu tả | Chức năng này cho phép Admin nhắc nhở nhân viên hoàn thành đúng tiến độ công việc được giao nếu gần đến ngày hoàn thành mà nhân viên vẫn chưa hoàn thiện xong công việc được giao. |
| Đầu vào | Dữ liệu ngày hoàn thành công việc theo mục tiêu, tiến độ hoàn thành công việc của nhân viên. |
| Xử lý | Hệ thống tiến hành kiểm tra dữ liệu và tiến hành gửi mail nhắc nhở theo mail nhân viên đã đăng kí trước đó. |
| Xuất | Nhân viên nhận được mail thông báo trong hộp thư từ quản trị viên gửi tới. |

3.1.2.2 Đối với nhân viên

* Đăng nhập

|  |  |
| --- | --- |
| Miêu tả | Chức năng đăng nhập dùng để đăng nhập vào website với những nhân viên đã được cấp tài khoản. |
| Đầu vào | Người dùng tiến hành nhập tên và mật khẩu vào form đăng nhập tài khoản. |
| Xử lý | * Hệ thống thực hiện việc kiểm tra dữ liệu đầu vào đã đúng định dạng chưa, các trường bắt nhập đã nhập chưa… * Tiếp đó hệ thống sẽ kiểm tra xem có thông tin người dùng nào tương ứng với tên và mật khẩu người dùng nhập hay không. Nếu có, thực hiện đăng nhập cho người dùng, nếu sai thông báo kết quả cho người dùng. |
| Xuất | Thông báo kết quả cho người dùng và thực hiện đăng nhập vào hệ thống. |

* Xem thông tin mục tiêu

|  |  |
| --- | --- |
| Miêu tả | Chức năng xem thông tin mục tiêu cho phép nhân viên xemm thông tin tên mục tiêu, tên nhóm mục tiêu, ngày bắt đầu và ngày kết thúc mục tiêu. |
| Đầu vào | Người dùng kích vào mục tiêu muốn xem thông tin được giao |
| Xử lý | Tìm kiếm mục tiêu |
| Xuất | Hiển thị ra thông tin mục tiêu cho người dùng |

* Xem thông tin cá nhân

|  |  |
| --- | --- |
| Miêu tả | Chức năng xem thông tin cá nhân cho phép nhân viên xem các thông tin cá nhân của mình bao gồm tên, chức vụ, phòng ban, email… |
| Đầu vào | Người dùng đăng nhập vào hệ thống và click vào nút tài khoản của tôi |
| Xử lý | Hệ thống sẽ tìm kiếm những thông tin tương ứng với tài khoản của người dùng đã đã đăng nhập vào hệ thống. |
| Xuất | Hiện thị thông tin tài khoản của người dùng. |

* Tìm kiếm

|  |  |
| --- | --- |
| Miêu tả | Chức năng này cho phép nhân viên có thể tìm kiếm mục tiêu được nhập vào từ ô tim kiếm. |
| Đầu vào | Nhân viên tiến hành nhập dữ liệu và kích chọn vào ô tìm kiếm được thiết kế sẵn trên giao diện. |
| Xử lý | Hệ thống xử lý thông tin và thực hiện tìm kiếm. |
| Xuất | Hiển thị kết quả tìm kiếm. |

* Xem đánh giá

|  |  |
| --- | --- |
| Miêu tả | Chức năng này cho phép nhân viên có thể xem các mục tiêu mình đã hoàn thành và đã được đánh giá trong suốt quá trình làm việc. |
| Đầu vào | Nhân viên tiến hành kích chọn vào mục đánh giá |
| Xử lý | Hệ thống sẽ tìm kiếm thông tin. |
| Xuất | Hiển thị kết quả tìm kiếm. |

* Xem thông tin danh sách công việc

|  |  |
| --- | --- |
| Miêu tả | Chức năng xem thông tin các mục tiêu đã thực hiện cho phép nhân viên xem danh sách các mục tiêu mình đã hoàn thành. |
| Đầu vào | Nhân viên đăng nhập vào tài khoản và tiến hành chọn xem danh sách công việc của mình. |
| Xử lý | Hệ thống sẽ đưa vào thông tin đăng nhập của nhân viên đồng thời tìm kiếm danh sách các công việc của nhân viên. |
| Xuất | Hiện thị thông tin danh sách công việc của nhân viên. |

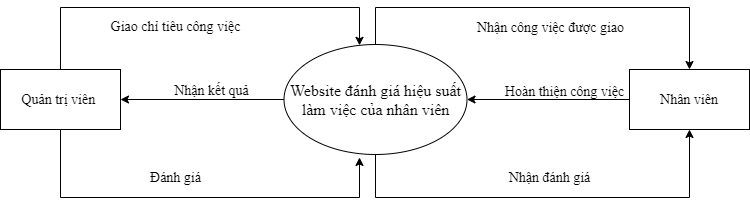
* Cập nhật thông tin nhân viên

|  |  |
| --- | --- |
| Miêu tả | Chức năng cập nhật thông tin nhân viên đối với những nhân viên đã được cấp tài khoản nhân viên. Chức năng cho phép người dùng có thể sửa thông tin tài khoản của mình gồm tên, mật khẩu và email. |
| Đầu vào | Người dùng đăng nhập vào hệ thống và xem và click sửa thông tin chi tiết nếu thấy sai. |
| Xử lý | Hệ thống sẽ kiểm tra việc nhập đúng các thồn tin đầu vào và các trường bắt buộc phải nhập sau đó tiến hành lưu thông tin vào trong cơ sở dữ liệu. |
| Xuất | Thông báo kết quả cho người dùng. |

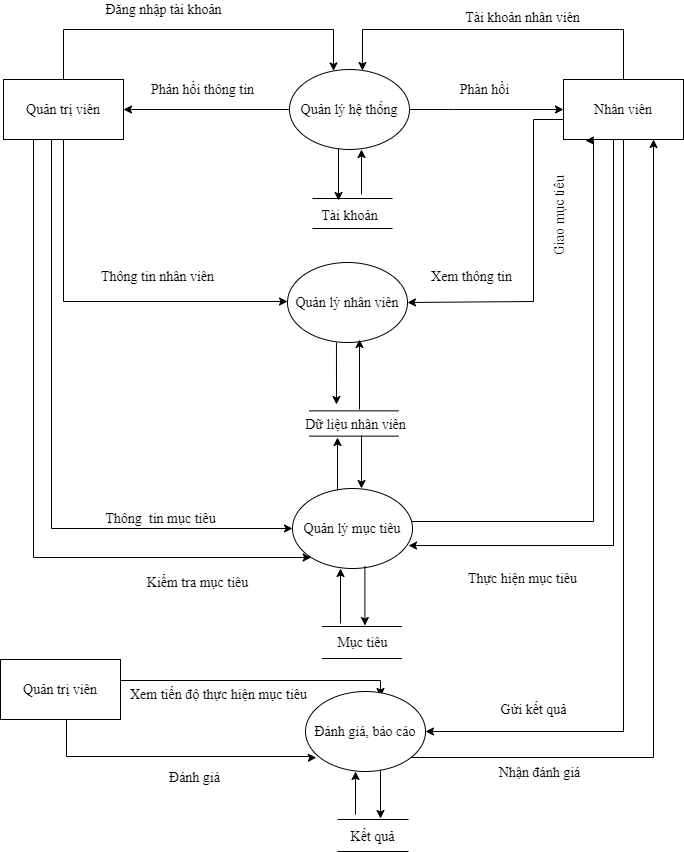
* Đăng xuất

|  |  |
| --- | --- |
| Miêu tả | Chức năng đăng xuất nhân viên dùng cho những nhân viên có tài khoản và đã đăng nhập vào hệ thống. Chức năng cho phép người dùng có thể đăng xuất tài khoản ra khỏi hệ thống. |
| Đầu vào | Người dùng kích vào nút đăng xuất. |
| Xử lý | Hệ thống thực hiện việc đăng xuất tài khoản. |
| Xuất | Thông báo kết quả cho người dùng và hiển thị giao diện đăng nhập. |

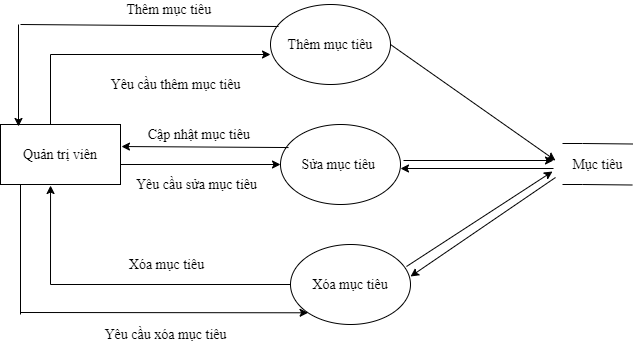
3.1.3. Biểu đồ luồng dữ liệu mức ngữ cảnh



Hình 3. : Biểu đồ luồng dữ liệu mức ngữ cảnh

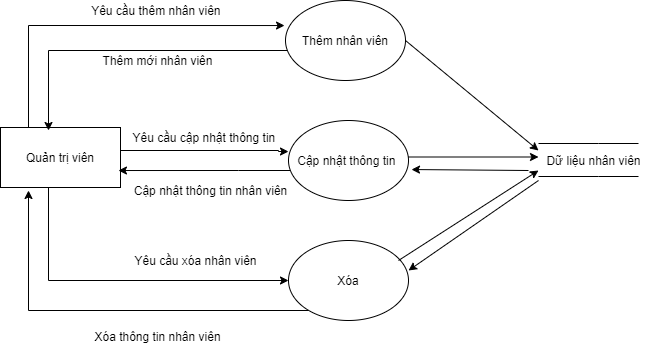
3.1.4. Biểu đồ luồng dữ liệu mức đỉnh

Hình 3. : Biểu đồ luồng dữ liệu mức đỉnh

3.1.5. Biểu đồ luồng dữ liệu cho chức năng quản lý mục tiêu

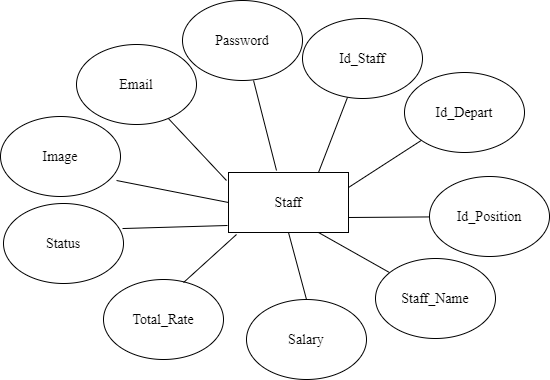
Hình 3. : Biểu đồ luồng dữ liệu cho chức năng quản lý mục tiêu

3.1.6. Biểu đồ luồng dữ liệu cho chức nằn quản lý nhân viên

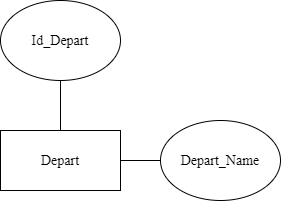


Hình 3. : Biểu đồ luồng dữ liệu cho chức năng quản lý nhân viên

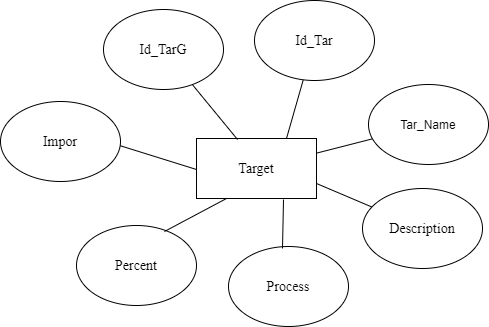
## 3.2. THIẾT KẾ DỮ LIỆU

3.2.1. Mô tả các thuộc tính của thực thể

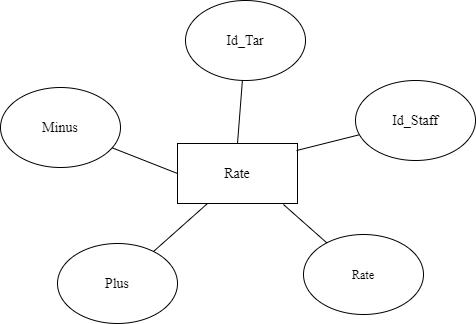
Hình 3. : Thực thể Staff



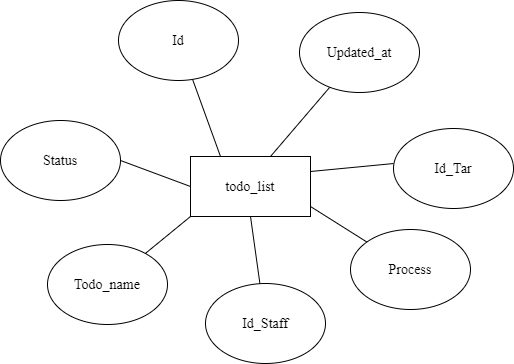
Hình 3.7: Thực thể Depart



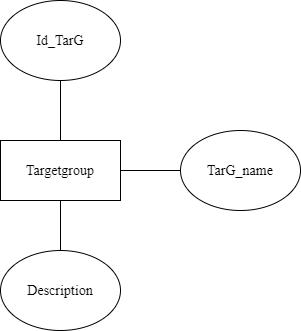
Hình 3.8: Thực thể Target



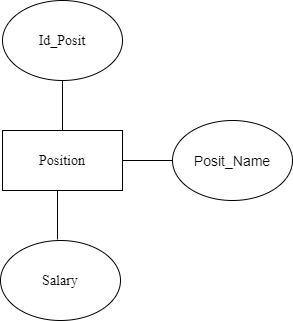
Hình 3.9: Thực thể Rate



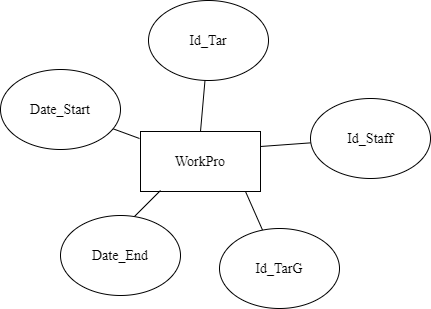
Hình 3.10: Thực thể Todo\_List



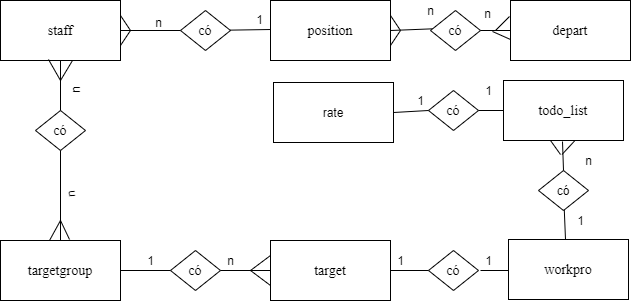
Hình 3.11: Thực thể Targetgroup



Hình 3.12: Thực thể Position



Hình 3.13: Thực thể WorkPro

3.2.2. Sơ đồ thực thể liên kết

Hình 3. : Sơ đồ thực thể liên kết

3.2.3. Các bảng dữ liệu

3.2.3.1. Bảng Nhân Viên

* Lưu trữ thông tin nhân viên

Bảng 3. : Bảng Nhân Viên

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu**  **DataType** | **Null** | **Ràng buộc**  **Contraint** | **Mô tả**  **Content** |
| Id | Int(11) | No | Khóa chính | Mã nhân viên |
| taiKhoan | Nvarchar |  |  | Tài Khoản |
| matKhau | Nvarchar |  |  | Mật Khẩu |
| email | Nvarchar |  |  | Email |
| hoTen | Nvarchar |  |  | Họ và tên |
| ngaySinh | Date |  |  | Ngày Sinh |
| sDT | Nvarchar |  |  | Số Điện Thoại |
| diaChi | Nvarchar |  |  | Địa Chỉ |
| chucVu | Nvarchar |  |  | Chức Vụ |
| cMND | Nvarchar |  |  | CMND |
| position | Nvarchar |  |  | Vị Trí |
| image | Nvarchar |  |  | Ảnh |

3.2.3.2. Bảng Đơn xin nghỉ phép

* Lưu trữ thông tin đơn xin nghỉ phép

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu**  **DataType** | **Null** | **Ràng buộc**  **Contraint** | **Mô tả**  **Content** |
| Id | Int | No | Khóa chính | Mã Đơn |
| nhanVienId | Int | No | Khóa ngoại | Mã Nhân Viên |
| soNgayNghi | Int |  |  | Số Ngày Nghỉ |
| lyDo | Nvarchar |  |  | Lý Do |
| lyDoTuChoi | Nvarchar |  |  | Lý do từ chối |
| dateTime | Date |  |  | Ngày làm đơn |

Bảng 3. : Bảng Đơn xin nghỉ phép

3.2.3.3. Bảng Lương Tổng

* Lưu trữ thông tin Lương tổng của nhân viên

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu**  **DataType** | **Null** | **Ràng buộc**  **Contraint** | **Mô tả**  **Content** |
| Id | Int | No | Khóa chính | Mã Lương Tổng |
| dateTime | Date |  |  | Ngày |
| luongLamThem | Float |  |  | Lương Làm Thêm |
| luongCung | Float |  |  | Lương Cứng |
| luongKhauTru | Float |  |  | Lương Khấu Trừ |
| luongTong | Float |  |  | Lương Tổng |

Bảng 3. : Bảng Lương Tổng

3.2.3.4. Bảng Lương Làm Thêm

* Lưu trữ thông tin lương làm thêm

Bảng 3. : Bảng Lương Làm Thêm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu**  **DataType** | **Null** | **Ràng buộc**  **Contraint** | **Mô tả**  **Content** |
| Id | Int | No | Khóa chính | Mã Lương Làm Thêm |
| luongTongId | Int | No | Khóa ngoại | Mã Lương Tổng |
| duAnId | Int | No | Khóa ngoại | Mã Dự Án |
| soGioLam | Int |  |  | Số Giờ Làm |
| tienThuong | Float |  |  | Tiền Thưởng |
| dateTime | Date |  |  | Ngày |

3.2.3.5. Bảng Dư án

* Lưu trữ thông tin Dự án

Bảng 3. : Bảng dư án

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu**  **DataType** | **Null** | **Ràng buộc**  **Contraint** | **Mô tả**  **Content** |
| Id | Int | No | Khóa chính | Mã dự án |
| tenDuAn | Nvarchar |  |  | Tên dự án |
| luongThuong | Float |  |  | Lương Thưởng |

3.2.3.6. Bảng Chấm công

* Lưu trữ thông tin chấm công

Bảng 3. : Bảng Chấm công

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu**  **DataType** | **Null** | **Ràng buộc**  **Contraint** | **Mô tả**  **Content** |
| Id | Int | No | Khóa Chính | Mã Chấm Công |
| luongTongId | Int | No | Khóa ngọai | Mã Lương Tổng |
| trangThai | Nvarchar |  |  | Trạng Thái |
| dateTime | Date |  |  | Thời gian tạo |

3.2.3.7. Bảng Lương Khấu Trừ

* Lưu trữ thông tin Lương Khấu Trừ

Bảng 3. : Bảng Lương Khấu Trừ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu**  **DataType** | **Null** | **Ràng buộc**  **Contraint** | **Mô tả**  **Content** |
| Id | Int | No | Khóa chính | Mã Lương Khấu Trừ |
| luongTongId | Int | No | Khóa ngoại | Mã Lương Tổng |
| soNgayMuon | Int |  |  | Số Ngày Muộn |
| soGioMuon | Int |  |  | Số Giờ Muộn |
| tienKhauTru | Float |  |  | Tiền Khấu Trừ |
| dateTime | Date |  |  | Ngày tạo |

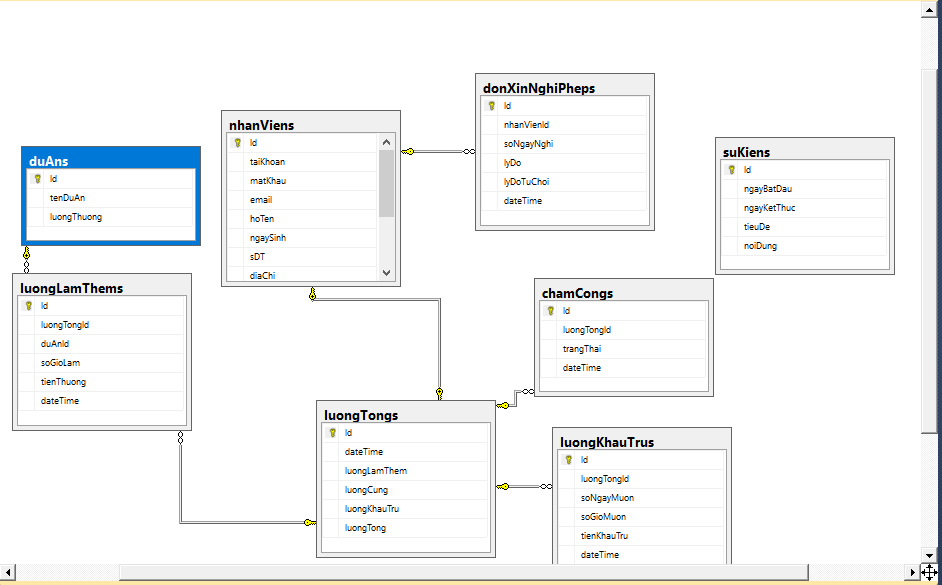
3.2.3.8. Bảng Sự Kiện

* Lưu trữ thông tin Sự Kiện

Bảng 3. : Bảng Sự Kiện

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu**  **DataType** | **Null** | **Ràng buộc**  **Contraint** | **Mô tả**  **Content** |
| Id | Int | No | Khóa chính | Mã Sự Kiện |
| ngayBatDau | Date |  |  | Ngày Bắt Đầu |
| ngayKetThuc | Date |  |  | Ngày Kết Thúc |
| tieuDe | Nvarchar |  |  | Tiêu Đề |
| noiDung | Nvarchar |  |  | Nội Dung |

3.2.3.9. Mô hình liên kết

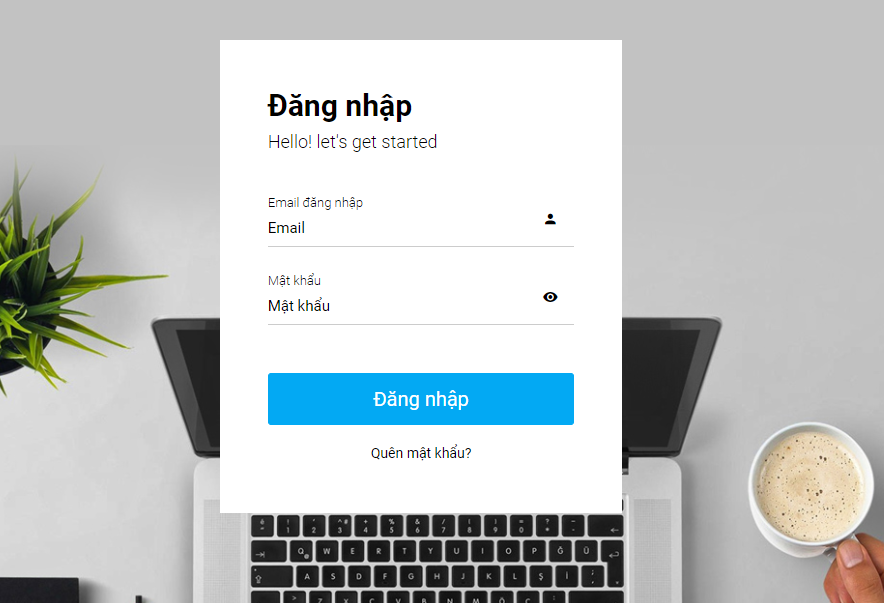
 Hình 3. : Mô hình liên kết

# CHƯƠNG 4: THIẾT KẾ GIAO DIỆN WEBSITE QUẢN LÝ NHÂN SỰ

## 4.1. GIAO DIỆN CỦA QUẢN TRỊ VIÊN

4.1.1. Giao diện đăng nhập

Tại giao diện đăng nhập, người dùng tiến hành nhập vào email và mật khẩu để đăng nhập vào website. Khi nhập đúng email và mật khẩu của tài khoản quản trị viên, giao diện website sẽ được hiển thị khác so với tài khoản nhân viên.

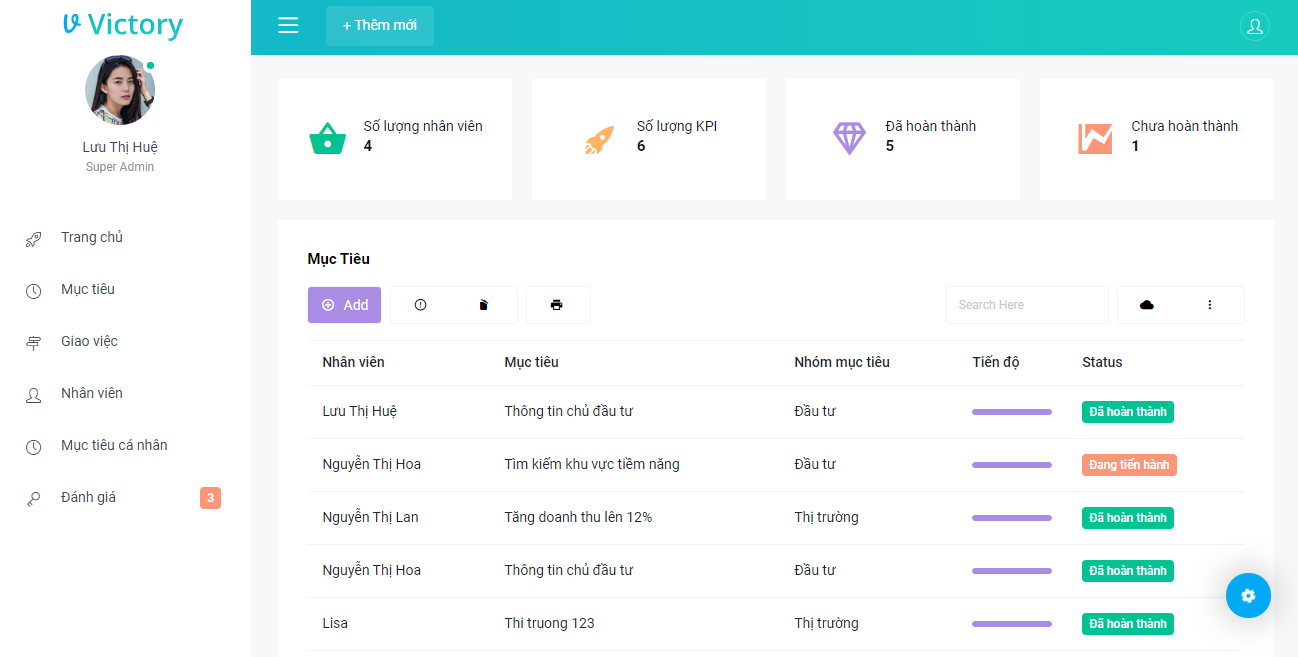
Nếu người dùng nhập sai email đăng nhập hoặc mật khẩu, hệ thống sẽ gửi thông báo lỗi đăng nhập và để người dùng đăng nhập lại.

Hình 4. : Giao diện đăng nhập hệ thống

4.1.2. Trang chủ

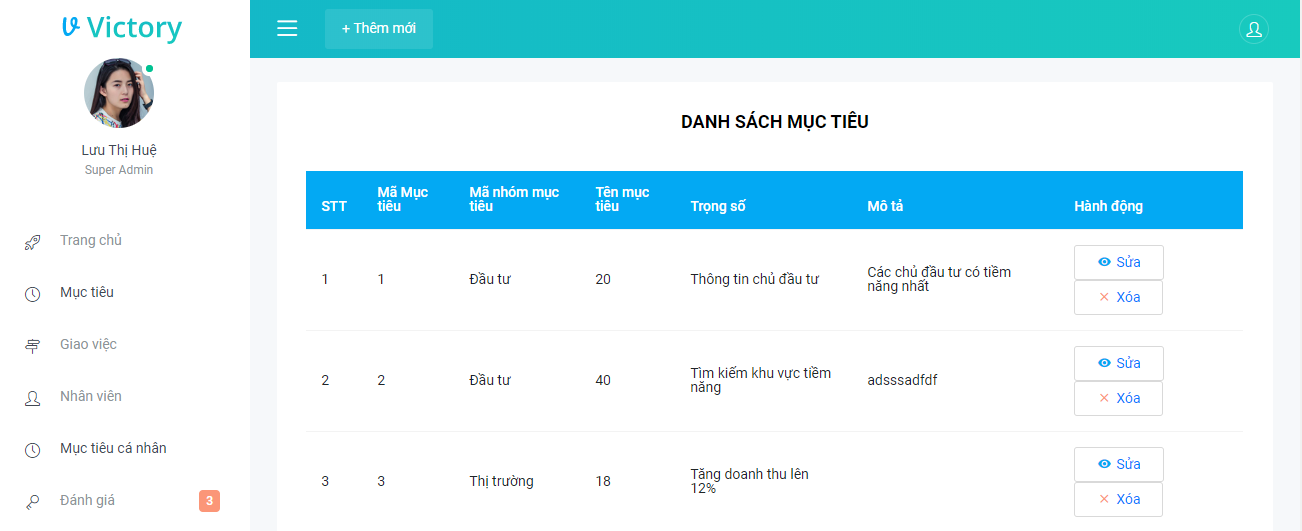
Tại giao diện trang chủ của quản trị viên, quản trị viên sẽ theo dõi được tổng số lượng nhân viên trong công ty, tổng số lượng KPI đề ra, số lượng KPI đã được hoàn thành và số lượng KPI chưa được hoàn thành.

Quản trị viên sẽ theo dõi được tiến độ thực hiện của nhân viên bằng cách quan sát vào tiến độ qua bảng hiển thị.

Ngoài ra, quản trị viên có thể dễ dàng quan sát được công việc nào đã được hoàn thành, công việc nào chưa được hoàn thành.

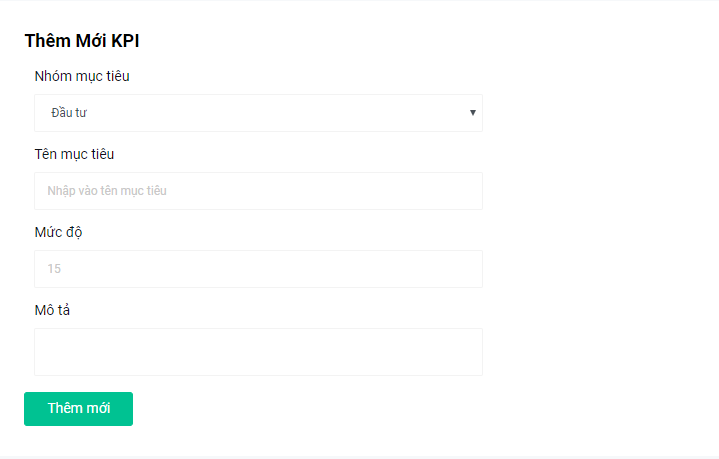
Hình 4. : Trang chủ giao diện Admin

4.1.3. Quản lý mục tiêu

Khi click vào Mục tiêu, website sẽ hiển thi giao diện quản lý quản lý mục tiêu bao gồm các chức năng sửa và xóa mục tiêu.

Hình 4. : Giao diện quản lý mục tiêu

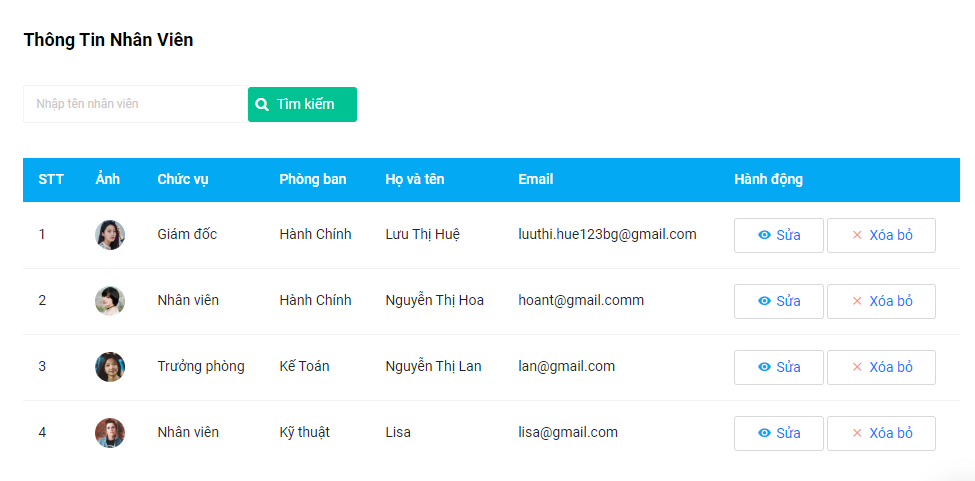
Khi tiến hành click vào Thêm mới và chọn Thêm mới mục tiêu. Website sẽ hiển thị form thêm mới mục tiêu như hình 4.5 để quản trị viên có thể thêm mới mục tiêu cho nhân viên.



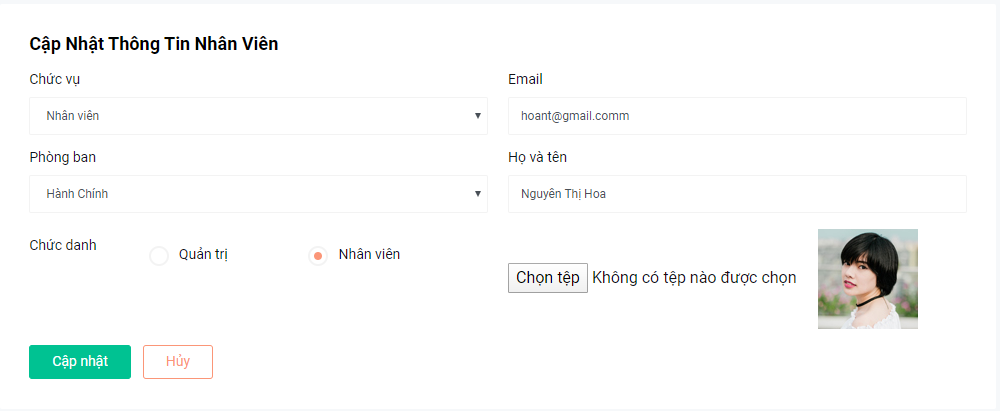
Hình 4. : Giao diện thêm mới KPI

Trong quá trình thêm mới mục tiêu, quản trị viên sẽ chọn nhóm mục tiêu được tạo từ trước, nhằm tạo hiệu quả tối đa cho công việc. Ngoài ra, nếu người dùng không nhập tên mục tiêu, hệ thống sẽ báo lỗi không được để trống tên mục tiêu để người dùng nhập lại.

4.1.4. Quản lý nhân viên

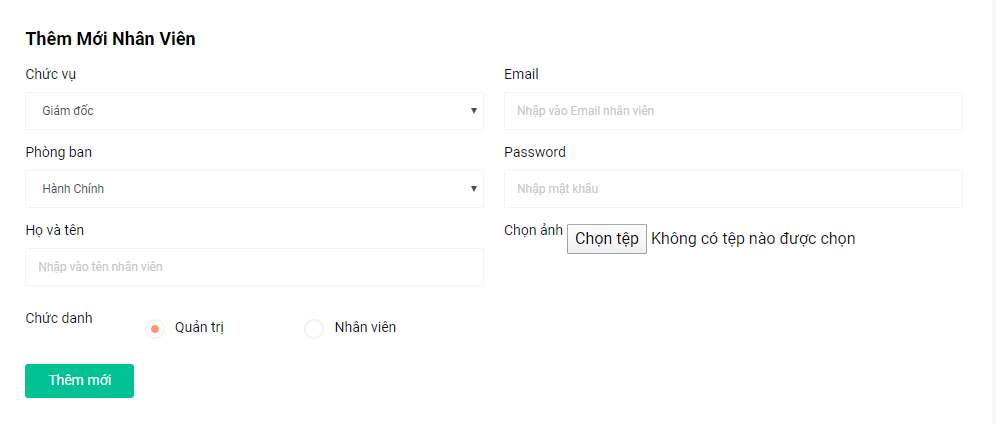
Khi quản trị viên kích chọn vào Nhân viên, website sẽ hiển thị thông tin của toàn bộ nhân viên trong công ty. Đồng thời quản trị có thể sửa thông tin hoặc xóa thông tin nhân viên trong công ty.

Hình 4. : Giao diện quản lý nhân viên

Quản trị có thể tìm kiếm nhân viên trong công ty bằng cách nhập vào tên của nhân viên trong ô tìm kiếm sau đó ấn tìm kiếm. Hệ thống sẽ hiển thị danh sách thông tin nhân viên cần tìm ngay sau đó.

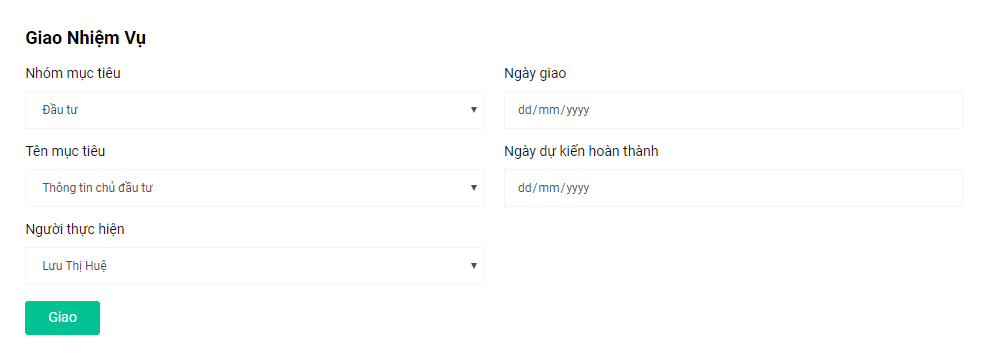
Hình 4. : Giao diện sửa thông tin nhân viên

Khi click chọn Thêm mới và chọn Thêm nhân viên, hệ thống sẽ hiển thị form thêm mới nhân viên để quản trị viên thêm mới nhân viên.

Khi tiến hành thêm mới nhân viên, nếu trường tên nhân viên, mật khẩu và email của nhân viên bị bỏ trống, hệ thống sẽ thông báo lỗi để người dùng nhập lại và lưu vào cơ sở dữ liệu. Sau khi thêm mới thành công, nhân viên sẽ được hiển thị trong danh sách nhân viên. Mỗi tài khoản nhân viên sẽ dùng một email để đăng nhập vào tài khoản của mình, nếu email nhân viên bị trùng, hệ thống sẽ thông báo lỗi email đã tồn tại.

Hình 4. : Giao diện thêm mới nhân viên

4.1.5. Giao việc

Khi click vào Giao việc, hệ thống sẽ hiển thị giao diện giao việc gồm nhóm mục tiêu, tên mục tiêu để quản trị viên có thể dễ dàng giao việc cho nhân viên của mình.

Hình 4. : Giao diện giao việc

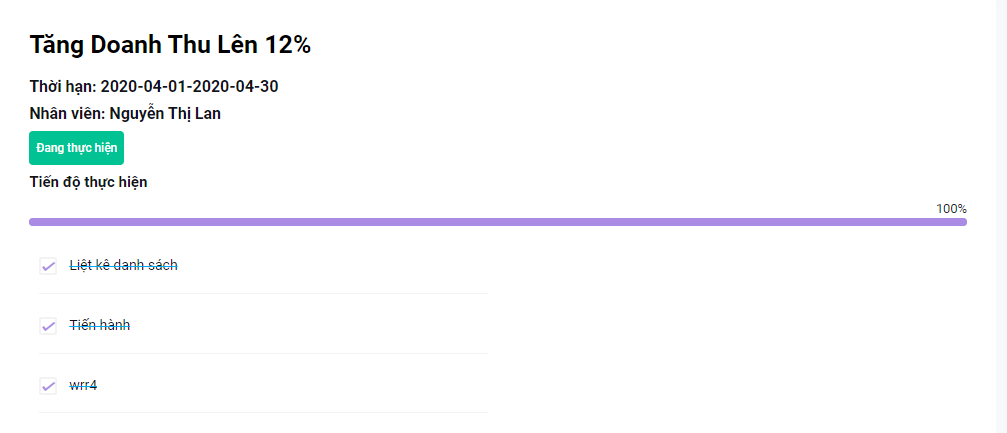
Sau khi giao việc thành công, danh sách công việc được giao sẽ được hiển thị theo một danh sách, cho phép quản trị viên có thể dễ dàng quan sát được nhân viên nào được giao nhiệm vụ nào và tiến độ thực hiện ra sao.

Hình 4. : Danh sách công việc và nhân viên

4.1.6. Tiến độ thực hiện của nhân viên

Khi quản trị viên muốn theo dõi nhân viên thực hiện mục tiêu đến đâu, đã hoàn thành hay chưa sẽ kích chọn vào xem trong danh sách hiện thị công việc được giao.

Tại giao diện theo dõi, quản trị viên có thể xem checklist công việc cần thực hiện của nhân viên, các công việc nhân viên đã thực hiện và phần trăm công việc đã hoàn thành.

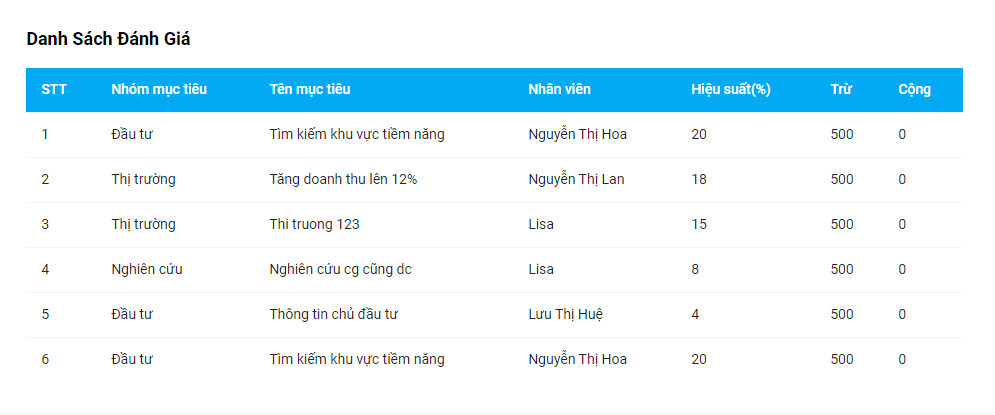
4.1.7. Đánh giá

Hình 4. : Giao diện theo dõi tiến độ thực hiện của nhân viên

Sau khi nhân viên tiến hành hoàn thành công việc, quản trị viên tiến hành đánh giá kết quả thực hiện của nhân viên.



Hình 4. : Giao diện đánh giá

Quản trị viên kích chọn vào danh sách giá để xem thông tin đánh giá nhân viên trong tháng.

Hình 4. : Danh sách đánh giá

4.1.8. Tính lương

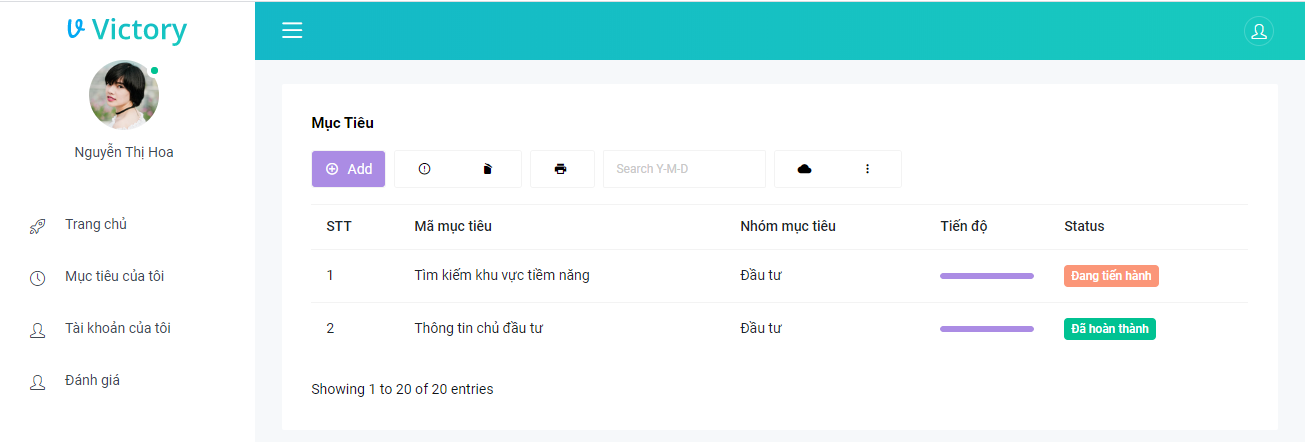
Sau khi đánh giá hiệu suất nhân viên, tổng hiệu suất nhân viên làm việc trong tháng sẽ được tổng hợp và tính lương. Trong quá trình hoàn thành công việc, nếu nhân viên hoàn thành đúng hạn và đạt yêu cầu sẽ được thưởng và ngược lại. Nhân viên không hoàn thành mục tiêu đề ra, hoàn thành muộn so với yêu cầu sẽ bị trừ lương.



Hình 4. : Tính lương

## 4.2. GIAO DIỆN CỦA NHÂN VIÊN

4.2.1. Trang chủ

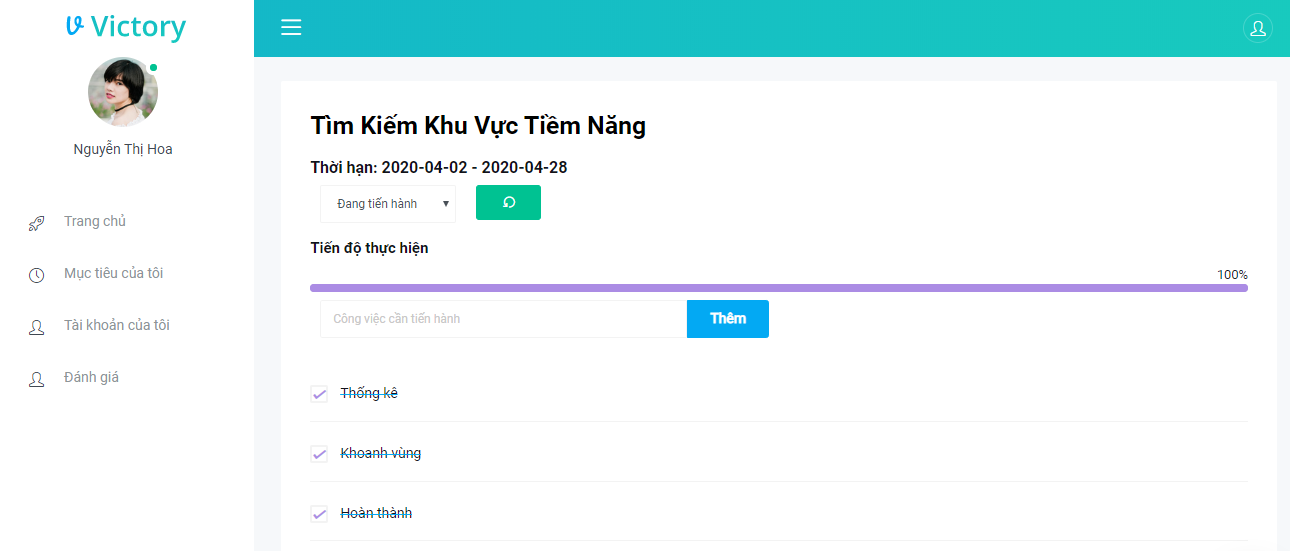
Tại giao diện trang chủ của nhân viên, nhân viên sẽ quan sát được số lượng công việc mình được giao và tiến độ thực hiện của mình. Ngoài ra, nhân viên có thể quan sát hạn mục tiêu để có thể hoàn thành mục tiêu đúng hạn.

Hình 4. : Giao diện trang chủ của nhân viên

4.2.2. Thực hiện mục tiêu

Khi tiến hành kích chọn vào Mục tiêu của tôi, nhân viên sẽ theo dõi được mục tiêu của mình và đồng thời tiến hành thực hiện mục tiêu.

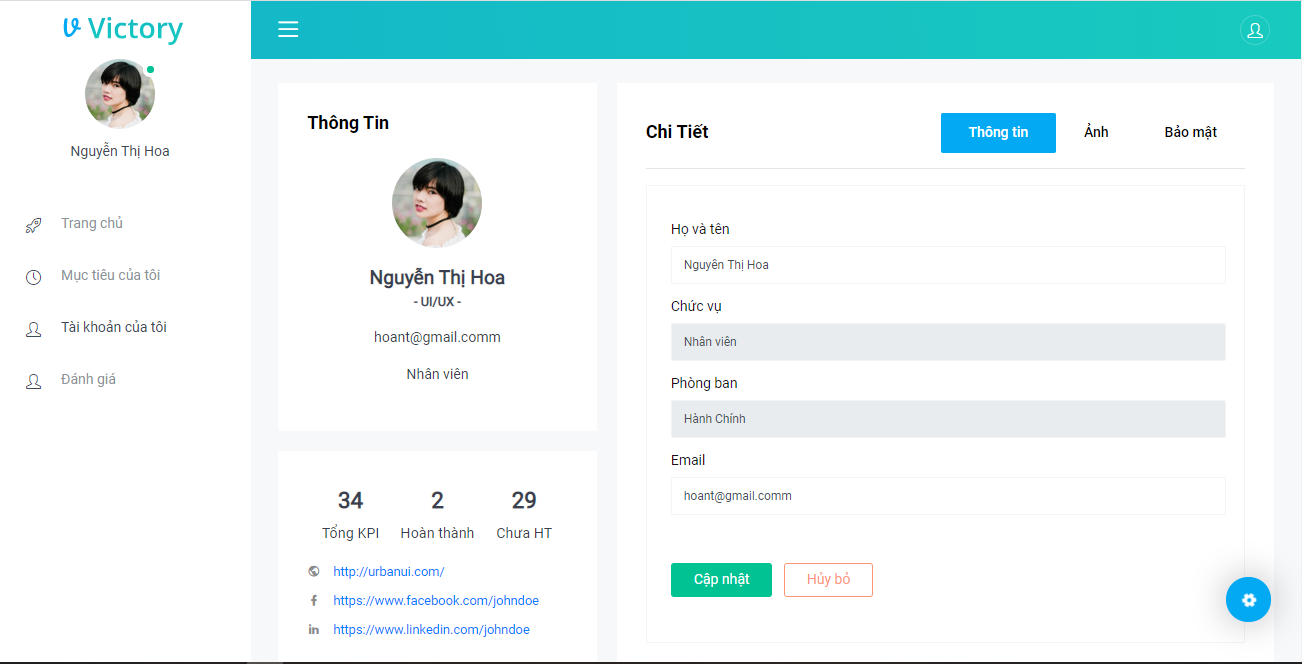
Trong quá trình thực hiện công việc, nhân viên sẽ chia nhỏ công việc được giao thành nhiều phần nhỏ theo checklist. Khi thực hiện xong nhân viên tiến hành tích chọn với công việc tương ứng đã hoàn thành. Dựa vào đây, nhân viên có thể theo dõi phần trăm hoàn thành công việc của mình.

Khi thực hiện xong, nhân viên cập nhật trạng thái công việc thành đã hoàn thành và lưu vào cơ sở dữ liệu.

Hình 4. : Giao diện thực hiện mục tiêu

4.2.3. Xem và cập nhật thông tin cá nhân

Kích chọn vào Tài khoản của tôi cho phép nhân viên xem và cập nhật thông tin cá nhân của mình bao gồm họ tên, emai, ảnh đại diện và mật khẩu.

Khi đổi thông tin thành công, hệ thống sẽ hiển thị thông báo để nhân viên nắm được.

Hình 4. : Giao diện thông tin nhân viên

# KẾT LUẬN

1. Kết quả đạt được

Sau quá trình nghiên cứu và xây dựng website quản lý nhân sự em đã thu được những kết quả sau:

Với nhà quản lý:

* + Nhà quản lý dễ dàng quản lý được nhân viên, số lượng mục tiêu đề ra cho doanh nghiệp.
  + Nắm bắt được tiến trình làm việc của nhân viên, đánh giá được hiệu suất làm việc của nhân viên
  + Đánh giá hiệu suất nhân viên công bằng, từ đó nhà quản lý có thể theo dõi được nhân viên nào đóng góp cho công ty nhiều, hiệu quả. Từ đó đưa ra mức lương thưởng xứng đáng so với công sức lao động, cống hiến của nhân viên với công ty.
  + Tiết kiệm được nhiều thời gian cho công tác quản lý hồ sơ nhân viên, và đánh giá.

Với nhân viên:

* + Thông qua website, nhân viên có thể dễ dàng theo dõi số công việc mình đã thực hiện…
  + Bác bỏ tình trạng nghi ngờ trong khâu đánh giá, tính lương.

1. Hướng phát triển

Trước nhu cầu cấp thiết sử dụng và hiệu quả mang lại, website quản lý nhân sự còn có thể mở rộng với nhiều tính năng hữu ích hơn nữa để giúp ích cho doanh nghiệp cũng như công tác quản lý.

Em sẽ tiếp tục nghiên cứu để phát triển, sửa chữa những sai sót và tiếp tục nghiên cứu để ứng dụng có thể được đưa vào sử dụng thực tế hiện nay.

# TÀI LIỆU THAM KHẢO

[1].<https://tailieu.vn/doc/bao-cao-thuc-tap-phan-men-quan-ly-nhan-su-762134.html>

[2].<https://monamedia.co/javascript-la-gi-gioi-thieu-ve-ngon-ngu-js-cho-nguoi-moi-hoc/>

[3].<https://niithanoi.edu.vn/nodejs-la-gi-tong-hop-day-du-ve-nodejs-ban-can-biet.html>

[4]. [https://freetuts.net/cai-dat-nodejs-tren-local-585.html#](https://freetuts.net/cai-dat-nodejs-tren-local-585.html)

[5].<https://kiemtienblog.com/react-js-la-gi/#:~:text=ReactJS%20l%C3%A0%20m%E1%BB%99t%20th%C6%B0%20vi%E1%BB%87n,k%E1%BA%BFt%20n%E1%BB%91i%20%C4%91%C6%B0%E1%BB%A3c%20v%E1%BB%9Bi%20nhau>.

[6]. <https://ruaxetudong.org/reactjs-la-gi/>

[7].<https://codegym.vn/blog/2020/06/29/c-la-gi-tim-hieu-ve-ngon-ngu-lap-trinh-c/>

[8]. <https://comdy.vn/entity-framework/gioi-thieu-entity-framework/>

[9].<https://topdev.vn/blog/api-la-gi/?utm_source=google&utm_medium=cpc&utm_campaign=topdev&utm_term=api%20l%C3%A0%20g%C3%AC&utm_content=blog&gclid=Cj0KCQjw3IqSBhCoARIsAMBkTb2mf_XVJWaV6JwHbXR1N6rSXPoEBhZ7TDXe4tMsrLalBIxWT3yoizwaAqeuEALw_wcB>

[10]. <https://ruaxetudong.org/python-la-gi/>