|  |  |
| --- | --- |
| NGUYỄN QUANG ANH NGÀNH CÔNG NGHỆ THÔNG TIN | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------**    ĐA, KLTN ĐẠI HỌC NGÀNH CÔNG NGHỆ THÔNG TIN  **ĐỀ TÀI XÂY DỰNG WEBSITE QUẢN LÝ NHÂN SỰ**  **CHO CÔNG TY CỔ PHẦN TRUYỀN THÔNG VÀ DỊCH VỤ NODO**  **BẰNG SPRING FRAMEWORK**    **CBHD: TS. Nguyễn Thị Hoa Huệ**  **Sinh viên: Nguyễn Quang Anh**  **Mã số sinh viên: 2020602996**      Hà Nội – Năm 2024 |
| NGUYỄN QUANG ANH NGÀNH CÔNG NGHỆ THÔNG TIN | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------**    ĐA, KLTN ĐẠI HỌC NGÀNH CÔNG NGHỆ THÔNG TIN  **ĐỀ TÀI XÂY DỰNG WEBSITE QUẢN LÝ NHÂN SỰ**  **CHO CÔNG TY CỔ PHẦN TRUYỀN THÔNG VÀ DỊCH VỤ NODO**  **BẰNG SPRING FRAMEWORK**    **CBHD: TS. Nguyễn Thị Hoa Huệ**  **Sinh viên: Nguyễn Quang Anh**  **Mã số sinh viên: 2020602996**      Hà Nội – Năm 2024 |

# LỜI CẢM ƠN

Sau 4 năm học tập và rèn luyện tại Trường Đại học Công Nghiệp Hà Nội, đồ án tốt nghiệp này là một dấu mốc quan trọng đánh dấu việc em - một sinh viên đã hoàn thành nhiệm vụ của mình trên ghế giảng đường Đại học. Đầu tiên, với tình cảm sâu sắc, chân thành, cho phép em được bày tỏ lòng biết ơn sâu sắc đến gia đình, các anh chị, bạn bè đã luôn động viên, khích lệ, tạo điều kiện giúp đỡ em trong suốt quá trình học tập và nghiên cứu đề tài.

Em cũng xin gửi tới các thầy các cô khoa Công Nghệ Thông Tin Trường Đại học Công Nghiệp Hà Nội lời chào trân trọng, lời chúc sức khỏe và lời cảm ơn sâu sắc. Với sự quan tâm, dạy dỗ, chỉ bảo tận tình chu đáo của thầy cô, đến nay em đã có thể hoàn thành đồ án tốt nghiệp, đề tài: “Xây dựng website quản lý nhân sự cho công ty cổ phần truyền thông và dịch vụ Nodo bằng Spring Framework”.

Đặc biệt em xin gửi lời cảm ơn chân thành đến cô TS. Nguyễn Thị Hoa Huệ người đã tận tình giúp và hướng dẫn em hoàn thành đề tài đồ án.

Đồng thời, em xin bày tỏ lòng biết ơn đến lãnh đạo Trường Đại học Công Nghiệp Hà Nội, các Khoa, Phòng ban chức năng đã tạo điều kiện cho em được học tập tại nơi mà em yêu thích, cho em bước vào đời sống thực tế và áp dụng những kiến thức em đã học tại trường và môi trường làm việc mới của em. Qua quá trình học tập em đã tích lũy được rất nhiều kiến thức để chuẩn bị cho công việc sau này cũng như để phát triển thêm bản thân.

Trong quá trình hoàn thành đồ án tốt nghiệp không thể tránh khỏi thiếu sót, kính mong có sự góp ý từ thầy cô.

*Em xin chân thành cảm ơn!*

Nguyễn Quang Anh

MỤC LỤC

[LỜI CẢM ƠN 3](#_Toc166758004)

[DANH MỤC CÁC THUẬT NGỮ, KÝ HIỆU VÀ CÁC CHỮ VIẾT TẮT 6](#_Toc166758005)

[DANH MỤC BẢNG 7](#_Toc166758006)

[DANH MỤC CÁC HÌNH VẼ 7](#_Toc166758007)

[MỞ ĐẦU 10](#_Toc166758008)

[1. Lý do chọn đề tài 10](#_Toc166758009)

[2. Mục tiêu của đề tài 10](#_Toc166758010)

[3. Đối tượng và phạm vi 11](#_Toc166758011)

[4. Kết quả dự kiến đạt được 12](#_Toc166758012)

[5. Bố cục của báo cáo 12](#_Toc166758013)

[CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN 1](#_Toc166758014)

[1.1 Giới thiệu về công ty Nodo 1](#_Toc166758015)

[1.2 Xác định bài toán 2](#_Toc166758016)

[1.2.1 Quản lý hồ sơ nhân viên 2](#_Toc166758017)

[1.2.2 Quản lý chấm công và đăng ký lịch nghỉ phép 4](#_Toc166758018)

[1.3 Yêu cầu chức năng 5](#_Toc166758019)

[1.4 Yêu cầu phi chức năng 6](#_Toc166758020)

[1.5 Giới thiệu về công nghệ và ngôn ngữ sử dụng 6](#_Toc166758021)

[1.5.1 Angular framework 6](#_Toc166758022)

[1.5.2 SpringBoot framework 8](#_Toc166758023)

[1.5.3 Mô hình RESTful API 9](#_Toc166758024)

[1.5.4 Hệ quản trị cơ sở dữ liệu quan hệ MySQL 10](#_Toc166758025)

[CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 12](#_Toc166758026)

[2.1 Mô hình hóa chức năng 12](#_Toc166758027)

[2.1.1 Xác định các Actor (Tác nhân) 12](#_Toc166758028)

[2.1.2 Biểu đồ Use Case 13](#_Toc166758029)

[2.2 Đặc tả chi tiết các use case 18](#_Toc166758030)

[2.2.1 Đặc tả use case Đăng nhập 18](#_Toc166758031)

[2.2.2 Đặc tả use case Quản lý hồ sơ nhân viên 20](#_Toc166758032)

[2.2.3 Đặc tả use case Tra cứu hồ sơ nhân viên 21](#_Toc166758033)

[2.2.4 Đặc tả use case Quản lý phòng ban 23](#_Toc166758034)

[2.2.5 Đặc tả use case Quản lý chức vụ 25](#_Toc166758035)

[2.2.6 Đặc tả use case Quản lý chấm công 26](#_Toc166758036)

[2.2.7 Đặc tả use case Đăng ký lịch nghỉ phép 28](#_Toc166758037)

[2.2.8 Đặc tả use case Đăng ký lịch OT 30](#_Toc166758038)

[2.2.9 Đặc tả use case Quản lý vai trò 31](#_Toc166758039)

[2.2.10 Đặc tả use case Quản lý phân cấp 33](#_Toc166758040)

[2.2.11 Đặc tả use case Quản lý hợp đồng 34](#_Toc166758041)

[2.3 Mô hình hóa dữ liệu 36](#_Toc166758042)

[2.3.1 Biểu đồ lớp 36](#_Toc166758043)

[2.3.2 Biểu đồ trình tự 38](#_Toc166758044)

[2.4 Thiết kế cơ sở dữ liệu 48](#_Toc166758045)

[2.4.1 Biểu đồ thực thể liên kết mức logic 48](#_Toc166758046)

[2.4.2 Thiết kế bảng 50](#_Toc166758047)

[2.4.3 Biểu đồ thực thể liên kết mức vật lý 55](#_Toc166758048)

[2.5 Biểu đồ triển khai 56](#_Toc166758049)

[CHƯƠNG 3: KẾT QUẢ VÀ KIỂM THỬ 58](#_Toc166758050)

[3.1. Một số kết quả đạt được 58](#_Toc166758051)

[3.2 Kiểm thử 68](#_Toc166758052)

[3.2.1. Kế hoạch kiểm thử 68](#_Toc166758053)

[3.2.2. Kiểm thử chức năng 69](#_Toc166758054)

[3.2.3. Kết quả kiểm thử 74](#_Toc166758055)

[KẾT LUẬN 75](#_Toc166758056)

[TÀI LIỆU THAM KHẢO 76](#_Toc166758057)

# DANH MỤC CÁC THUẬT NGỮ, KÝ HIỆU VÀ CÁC CHỮ VIẾT TẮT

|  |  |
| --- | --- |
| **Ký hiệu** | **Giải thích** |
| API | Giao diện lập trình ứng dụng (Application programming interface) |
| HTML | Ngôn ngữ đánh dấu (Hyper Text Markup Language) |
| CSS | Ngôn ngữ tạo phong cách cho trang web (Cascading Style Sheets) |
| UC | Use Case |
| OT | Tăng ca |
| HTTP | Giao thức truyền tải (Hypertext Transfer Protocol) |
| CSDL | Cơ sở dữ liệu |
| SQL | Ngôn ngữ truy vấn có cấu trúc |
| Admin | Quản trị hệ thống |

# DANH MỤC BẢNG

[Bảng 2.1 Danh sách các tác nhân của hệ thống 12](#_Toc166714012)

[Bảng 3.1 Kiểm thử chức năng. 69](#_Toc166714013)

# DANH MỤC CÁC HÌNH VẼ

[Hình 1.1 Thư viện Spring Framework 8](#_Toc166714177)

[Hình 1.2 Mô hình RESTful API 9](#_Toc166714178)

[Hình 1.3 Hệ quản trị cơ sở dữ liệu MySQL 10](#_Toc166714179)

[Hình 2.1 Biểu đồ use case tổng quát của hệ thống 13](#_Toc166714180)

[Hình 2.2 Phân rã use case quản lý hồ sơ nhân viên 14](#_Toc166714181)

[Hình 2.3 Phân rã use case cập nhật thông tin phụ cấp 15](#_Toc166714182)

[Hình 2.4 Phân rã giữa use case cập nhật phòng ban 15](#_Toc166714183)

[Hình 2.5 Phân rã giữa case cập nhật chức vụ 16](#_Toc166714184)

[Hình 2.6 Phân rã giữa use case chấm công 16](#_Toc166714185)

[Hình 2.7 Phân rã giữa use case đăng ký đơn nghỉ phép 17](#_Toc166714186)

[Hình 2.8 Phân rã giữa use case đăng ký lịch OT 17](#_Toc166714187)

[Hình 2.9 Phân rã giữa use case phân quyền 18](#_Toc166714188)

[Hình 2.10 Biểu đồ use case đăng nhập 18](#_Toc166714189)

[Hình 2.11 Biểu đồ use case quản lý hồ sơ nhân viên 20](#_Toc166714190)

[Hình 2.12 Biểu đồ use case tra cứu hồ sơ nhân viên 22](#_Toc166714191)

[Hình 2.13 Biểu đồ use case quản lý phòng ban 23](#_Toc166714192)

[Hình 2.14 Biểu đồ use case quản lý chức vụ 25](#_Toc166714193)

[Hình 2.15 Biểu đồ use case chấm công 27](#_Toc166714194)

[Hình 2.16 Biểu đồ use case đăng ký lịch nghỉ phép 28](#_Toc166714195)

[Hình 2.17 Biểu đồ use case đăng ký lịch ot 30](#_Toc166714196)

[Hình 2.18 Biểu đồ use case quản lý vai trò 32](#_Toc166714197)

[Hình 2.19 Biểu đồ use case quản lý phân cấp 33](#_Toc166714198)

[Hình 2.20 Biểu đồ use case quản lý hợp đồng 35](#_Toc166714199)

[Hình 2.21 Biểu đồ lớp 37](#_Toc166714200)

[Hình 2.23 Biểu đồ trình tự Quản lý hồ sơ nhân viên 39](#_Toc166714201)

[Hình 2.24 Biểu đồ trình tự Tra cứu hồ sơ nhân viên 40](#_Toc166714202)

[Hình 2.25 Biểu đồ trình tự Quản lý phòng ban 41](#_Toc166714203)

[Hình 2.26 Biểu đồ trình tự Quản lý chức vụ 42](#_Toc166714204)

[Hình 2.27 Biểu đồ trình tự Quản lý chấm công 43](#_Toc166714205)

[Hình 2.28 Biểu đồ trình tự Đăng ký lịch nghỉ phép 44](#_Toc166714206)

[Hình 2.29 Biểu đồ trình tự Đăng ký lịch OT 45](#_Toc166714207)

[Hình 2.30 Biểu đồ trình tự Quản lý hợp đồng 47](#_Toc166714208)

[Hình 2.31 Biểu đồ trình tự Đăng nhập 47](#_Toc166714209)

[Hình 2.32 Mô hình dữ liệu quan hệ 48](#_Toc166714210)

[Hình 2.33 Bảng dữ liệu tài khoản 50](#_Toc166714211)

[Hình 2.34 Bảng dữ liệu nhân viên 50](#_Toc166714212)

[Hình 2.35 Bảng dữ liệu hợp đồng 51](#_Toc166714213)

[Hình 2.36 Bảng dữ liệu hợp đồng với nhân viên 51](#_Toc166714214)

[Hình 2.37 Bảng dữ liệu thông tin phụ cấp 51](#_Toc166714215)

[Hình 2.38 Bảng dữ liệu phụ cấp với nhân viên 52](#_Toc166714216)

[Hình 2.39 Bảng dữ liệu bằng cấp chứng chỉ 52](#_Toc166714217)

[Hình 2.40 Bảng dữ liệu phòng ban 52](#_Toc166714218)

[Hình 2.41 Bảng dữ liệu chức vụ 53](#_Toc166714219)

[Hình 2.42 Bảng dữ liệu chấm công 53](#_Toc166714220)

[Hình 2.43 Bảng dữ liệu đăng ký ot 53](#_Toc166714221)

[Hình 2.44 Bảng dữ liệu đăng ký nghỉ phép 54](#_Toc166714222)

[Hình 2.45 Bảng dữ liệu phân quyền 54](#_Toc166714223)

[Hình 2.46 Bảng dữ liệu menu 54](#_Toc166714224)

[Hình 2.47 Bảng dữ liệu phân quyền với vai trò 55](#_Toc166714225)

[Hình 2.48 Biểu đồ thực thể liên kết mức vật lý 55](#_Toc166714226)

[Hình 2.49 Biểu đồ triển khai 57](#_Toc166714227)

[Hình 3.1 Giao diện trang chủ 58](#_Toc166714228)

[Hình 3.2 Giao diện danh sách nhân viên 58](#_Toc166714229)

[Hình 3.3 Giao diện quản lý bằng cấp chứng chỉ 59](#_Toc166714230)

[Hình 3.4 Giao diện quản lý hợp đồng của nhân viên 59](#_Toc166714231)

[Hình 3.5 Giao diện quản lý thông tin bảo hiểm 60](#_Toc166714232)

[Hình 3.6 Giao diện quản lý thông tin phụ cấp 60](#_Toc166714233)

[Hình 3.7 Giao diện xuất hồ sơ cá nhân 61](#_Toc166714234)

[Hình 3.8 File PDF được xuất 61](#_Toc166714235)

[Hình 3.9 Giao diện quản lý thông tin phụ cấp công ty 62](#_Toc166714236)

[Hình 3.10 Giao diện quản lý hợp đồng công ty 62](#_Toc166714237)

[Hình 3.11 Giao diện quản lý phòng ban 63](#_Toc166714238)

[Hình 3.12 Giao diện quản lý chức vụ 63](#_Toc166714239)

[Hình 3.13 Giao diện quản lý chấm công 64](#_Toc166714240)

[Hình 3.14 Giao diện danh sách chấm công 64](#_Toc166714241)

[Hình 3.15 File excel xuất báo cáo thống kê chấm công 65](#_Toc166714242)

[Hình 3.16 Giao diện đăng ký lịch nghỉ phép 65](#_Toc166714243)

[Hình 3.17 Giao diện trang tạo đơn đăng ký lịch nghỉ phép 66](#_Toc166714244)

[Hình 3.18 Mẫu phản hồi gmail đơn xin duyệt nghỉ phép 66](#_Toc166714245)

[Hình 3.19 Giao diện đăng ký lịch OT 67](#_Toc166714246)

[Hình 3.20 Giao diện trang tạo đơn đăng ký lịch OT 67](#_Toc166714247)

[Hình 3.21 Giao diện tạo cấp tài khoản 68](#_Toc166714248)

[Hình 3.22 Giao diện danh sách tài khoản 68](#_Toc166714249)

# 

# MỞ ĐẦU

## 1. Lý do chọn đề tài

- Nhu cầu cải thiện quản lý nhân sự: Hiện nay, việc quản lý thông tin nhân viên của Công ty cổ phần truyền thông và dịch vụ Nodo vẫn chưa được tổ chức một cách khoa học và hiệu quả. Thông tin về nhân viên, bao gồm thông tin cá nhân, vị trí công việc, lịch sử làm việc, và các thông tin liên quan khác, đang được lưu trữ và quản lý theo các hệ thống truyền thống hoặc tài liệu giấy tờ, gây ra sự phức tạp và khó khăn trong việc truy xuất thông tin và phân bố công việc.

- Nâng cao hiệu suất làm việc: Xây dựng một website quản lý nhân sự sẽ giúp Công ty Nodo tạo ra một cơ sở dữ liệu chung, đồng bộ và dễ dàng truy cập cho tất cả các bộ phận và nhân viên. Việc này sẽ giúp cải thiện quá trình tìm kiếm thông tin, quản lý hệ thống, thực hiện đánh giá nhân viên, và tăng cường giao tiếp nội bộ.

- Tính linh hoạt và tiện lợi: Sử dụng một website quản lý nhân sự sẽ mang lại tính linh hoạt và tiện lợi cho các nhân viên và quản lý. Các tính năng như cập nhật thông tin cá nhân, xem lịch sử làm việc, đăng ký nghỉ phép, và gửi yêu cầu đề xuất sẽ được tích hợp trong hệ thống, giúp tiết kiệm thời gian và tăng cường sự linh hoạt trong quản lý.

## 2. Mục tiêu của đề tài

Đề tài**: Xây dựng website quản lý nhân sự cho Công ty cổ phần truyền thông và dịch vụ Nodo bằng Spring Framework** đáp ứng được những mục tiêu:

- Tối ưu hóa quản lý nhân sự: Tạo ra một hệ thống quản lý nhân sự mạnh mẽ và linh hoạt, giúp Công ty Nodo quản lý nhân sự một cách hiệu quả hơn. Hệ thống sẽ cung cấp các tính năng như quản lý thông tin cá nhân, quản lý chức vụ, quản lý lịch làm việc và đánh giá nhân viên.

- Tích hợp và tương thích: Đảm bảo tính tương thích và tính tích hợp của hệ thống với các hệ thống khác trong công ty, giúp dễ dàng chia sẻ thông tin và tối ưu hóa quy trình làm việc.

- Tăng cường trải nghiệm người dùng: Phát triển giao diện người dùng thân thiện và dễ sử dụng, giúp người dùng dễ dàng truy cập và sử dụng các tính năng của hệ thống một cách nhanh chóng và thuận tiện.

- Bảo mật và ổn định: Đảm bảo tính bảo mật và ổn định của hệ thống, từ việc bảo vệ thông tin cá nhân của nhân viên đến việc đảm bảo hoạt động ổn định và liên tục của hệ thống.

- Nâng cao hiệu suất làm việc: Hỗ trợ Công ty Nodo trong việc nâng cao hiệu suất làm việc của nhân viên thông qua việc cung cấp các công cụ và tính năng hỗ trợ quản lý nhân sự hiệu quả và tiện lợi.

## 3. Đối tượng và phạm vi

**Đối tượng:**

- Ban quản lý nhân sự: Bao gồm các quản lý cấp cao, quản lý phòng nhân sự và các nhân viên phụ trách công việc liên quan đến quản lý nhân sự.

- Nhân viên: Tất cả các nhân viên của Công ty Nodo, bao gồm nhân viên ở mọi cấp bậc và bộ phận, từ nhân viên văn phòng đến nhân viên kỹ thuật và nhân viên quản lý.

**Phạm vi:**

- Phân tích yêu cầu của người dùng và xây dựng các chức năng cần thiết cho hệ thống quản lý nhân sự.

- Thiết kế cơ sở dữ liệu phù hợp với nhu cầu của công ty và tính bảo mật cao. Phát triển website sử dụng Spring Framework, bao gồm cả frontend và backend.

- Kiểm thử, sửa lỗi, và triển khai hệ thống trên môi trường sản xuất.

## 4. Kết quả dự kiến đạt được

- Tạo ra một hệ thống quản lý nhân sự hoàn chỉnh, giúp Công ty Nodo tăng cường quản lý nhân sự và nâng cao hiệu suất làm việc.

- Đảm bảo tính linh hoạt, mở rộng và dễ dàng bảo trì của hệ thống.

- Xác định được yêu cầu bài toán và phạm vi của ứng dụng

- Phân tích thiết kế đầy đủ những tính năng cơ bản của hệ thống

- Hoàn thành giao diện và cài đặt được các chức năng và yêu cầu đã đặt ra.

- Báo cáo tổng hợp của đề tài.

- Thuyết minh và đĩa CD đính kèm.

- Mã nguồn được quản lý với Git và lưu trữ trên GitHub.

- Kiểm thử sản phẩm và báo cáo kết quả kiểm thử.

## 5. Bố cục của báo cáo

Nội dung của báo cáo được chia thành 3 chương:

- Chương 1: Giới thiệu tổng quan

- Chương 2: Phân tích và thiết kế hệ thống

- Chương 3: Kết quả và kiểm thử

# CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN

Trong chương 1 sẽ trình bày tổng quan về dự án xây dựng - website quản lý nhân sự cho Công ty cổ phần truyền thông và dịch vụ Nodo bằng Spring Framework. Các nội dung được đưa ra để giới thiệu cụ thể, chi tiết về ứng dụng web, người sử dụng, tính năng, lĩnh vực hoạt động của ứng dụng dự định xây dựng.

## 1.1 Giới thiệu về công ty Nodo

Công ty cổ phần truyền thông và dịch vụ Nodo ra đời với hoài bão mang công nghệ áp dụng vào cuộc sống, tiên phong trong các lĩnh vực xương sống của xã hội như Viễn thông, Quản trị doanh nghiệp, Quản trị khách hàng, Tài chính ngân hàng...Công ty Nodo quyết tâm đổi mới và đổi mới toàn diện để làm hiện đại đất nước. Công ty Nodo được thành lập bởi các chuyên gia hàng đầu trong lĩnh vực công nghệ thông tin, với đội ngũ nhân viên trẻ, chuyên môn cao được đào tạo tại các trường đại học hàng đầu tại Việt Nam.

Chúng tôi đã tiến hành một khảo sát ban đầu để hiểu rõ nhu cầu và yêu cầu của Công ty Nodo. Dưới đây là kết quả của phần khảo sát:

- Yêu cầu và nhu cầu của Công ty: Cần một hệ thống quản lý nhân sự hiệu quả và linh hoạt. Mong muốn giảm bớt công việc thủ công và tối ưu hóa quy trình làm việc. Yêu cầu tính năng quản lý thông tin nhân viên, chấm công, tính lương và quản lý tài liệu. Đòi hỏi hệ thống có khả năng mở rộng và tùy chỉnh theo nhu cầu cụ thể của Công ty.

- Nhận định và phân tích: Công ty Nodo đang gặp khó khăn trong việc quản lý nhân sự và xử lý các quy trình liên quan. Hiện tại, website hệ thống quản lý nhân sự [10] của công ty vẫn chưa đáp ứng được các yêu cầu và mong muốn của họ.

- Quản lý hồ sơ nhân viên: Tích hợp hệ thống quản lý hồ sơ nhân viên để lưu trữ và cập nhật thông tin cá nhân, kinh nghiệm làm việc, và lịch sử công việc của mỗi nhân viên.

- Quản lý chấm công: Hệ thống website của công ty chưa được tích hợp chức năng quản lý chấm công để theo dõi thời gian làm việc của nhân viên, bao gồm ca làm việc, ngày nghỉ, và công việc đã hoàn thành.

- Đăng ký lịch nghỉ phép: Hệ thống website của công ty đã cung cấp khả năng cho nhân viên đăng ký lịch nghỉ phép trực tuyến, và cho phép quản lý duyệt và quản lý lịch nghỉ phép của nhân viên.

- Thống kê và báo cáo còn hạn chế như việc cung cấp các công cụ thống kê và báo cáo để phân tích dữ liệu nhân sự, bao gồm số lượng nhân viên, thời gian làm việc, và các chỉ số hiệu suất làm việc chưa được xuất hiện.

Cải tiến các tính năng này sẽ giúp Công ty Nodo tối ưu hóa quản lý nhân sự và tăng cường hiệu suất làm việc của nhân viên.

## 1.2 Xác định bài toán

### 1.2.1 Quản lý hồ sơ nhân viên

Công ty sẽ bắt đầu lập hồ sơ nhân viên khi quá trình tuyển dụng kết thúc. Nếu ứng viên đạt được các yêu cầu mà công ty đề ra thì phòng hành chính quản trị sẽ lập hợp đồng và tuyển dụng các ứng viên đạt chỉ tiêu vào công ty. Khi cả nhân viên mới và công ty cùng đồng ý với những thỏa thuận trong hợp đồng thì phòng hành chính quản trị sẽ lập hồ sơ cho nhân viên.

Ngoài ra trong quá trình làm việc nếu nhân viên được đưa đi đào tạo cấp cao, có cống hiến với công ty (hoặc vi phạm điều luật công ty) thì sẽ có các thông báo đào tạo cấp cao, các quyết định khen thưởng, kỷ luật ... sẽ được đưa vào trong hồ sơ của nhân viên.

Khi cần tra cứu hoặc sửa chữa thông tin về 1 nhân viên nào đó, phòng hành chính quản trị sẽ xem xét và tìm kiếm túi hồ sơ về nhân viên ấy để tra cứu thông tin (hoặc sửa chữa).

Sau 1 năm sẽ đánh giá về năng suất làm việc và thái độ công tác để quyết định ký kết lại hợp đồng, tăng lương, thưởng …

Công ty quản lý các hồ sơ nhân viên thông qua các túi hồ sơ. Các túi hồ sơ bao gồm:

- Bảng sơ yếu lý lịch: lưu trữ các thông tin như họ tên, ngày sinh, địa chỉ, quê quán, số chứng minh thư, trình độ đào tạo, các thông tin về gia đình, bản thân ...

- Hợp đồng của nhân viên với công ty: lưu trữ các thông tin về mã số nhân viên, chức vụ, phòng ban, hệ số lương, thời gian hợp đồng, các thỏa thuận giữa nhân viên và công ty.

- Các bằng cấp liên quan, các thông báo đào tạo cấp cao, các quyết định khen thưởng, kỷ luật …

**Đánh giá hiện trạng hệ thống quản lý hồ sơ nhân viên hiện thời:**

- Hệ thống web hiện tại có thể cung cấp các tính năng cơ bản như lưu trữ thông tin cá nhân, hồ sơ công việc và lịch sử làm việc của nhân viên.

- Các tính năng quản lý hồ sơ nhân viên thường có thể bao gồm nhập liệu, sửa đổi thông tin, xem chi tiết hồ sơ, và tìm kiếm nhân viên.

- Tuy nhiên, hệ thống có thể gặp phải một số hạn chế như giao diện không thân thiện, khả năng tìm kiếm và lọc dữ liệu không linh hoạt, và thiếu tính năng bảo mật và quản lý quyền truy cập.

**Giải pháp đề xuất:**

- Nâng cấp giao diện người dùng: Thiết kế lại giao diện người dùng để trở nên thân thiện và dễ sử dụng hơn. Tối ưu hóa trải nghiệm người dùng bằng cách cung cấp các tính năng tương tác và giao diện đồ họa trực quan.

- Tăng cường tính linh hoạt trong tìm kiếm và lọc dữ liệu:Cải thiện khả năng tìm kiếm và sắp xếp dữ liệu để đáp ứng được nhu cầu của các bộ phận và người dùng khác nhau trong công ty.

- Cập nhật thông tin và tính năng mới: Thêm các tính năng mới như quản lý bảo hiểm và bằng cấp chứng chỉ, đánh giá hiệu suất, và quản lý hợp đồng lao động. Cập nhật thông tin hồ sơ nhân viên thường xuyên để đảm bảo tính chính xác và hoàn thiện của dữ liệu.

### 1.2.2 Quản lý chấm công và đăng ký lịch nghỉ phép

Tại công ty Nodo, việc quản lý chấm công là một phần quan trọng để đảm bảo hiệu quả và minh bạch trong việc tính lương và quản lý thời gian làm việc của nhân viên. Bài toán quản lý chấm công trong một công ty công nghệ có thể bao gồm:

- Ghi nhận thời gian làm việc: Theo dõi và ghi nhận thời gian mỗi nhân viên bắt đầu và kết thúc công việc hàng ngày, bao gồm cả giờ đến và giờ về.

- Quản lý ca làm việc: Xác định và quản lý các ca làm việc của nhân viên, bao gồm cả ca làm việc chính thức và ca làm thêm giờ.

- Xử lý nghỉ phép và ngày nghỉ: Ghi nhận các ngày nghỉ phép, ngày nghỉ bệnh và các sự kiện khác mà nhân viên không tham gia làm việc.

- Báo cáo và thống kê: Tạo các báo cáo và thống kê về thời gian làm việc, ngày nghỉ phép và lương để hỗ trợ quản lý ra quyết định và đánh giá hiệu suất.

**Đánh giá hiện trạng hệ thống quản lý chấm công và đăng ký lịch nghỉ phép hiện thời:**

- Hệ thống web hiện tại có thể không cung cấp các tính năng quản lý chấm công một cách linh hoạt và tự động. Việc nhập và xử lý dữ liệu chấm công có thể phụ thuộc nhiều vào các công đoạn thủ công, gây mất thời gian và tăng nguy cơ phạm sai sót.

- Các quy trình đăng ký và phê duyệt lịch nghỉ phép có thể thực hiện bằng các phương tiện thủ công như email hoặc tài liệu giấy. Việc quản lý lịch nghỉ phép trở nên phức tạp và dễ gây nhầm lẫn.

- Thiếu tính tự động hóa, giao diện không thân thiện, khả năng xử lý dữ liệu hạn chế, không thể tạo và quản lý các biểu mẫu chấm công theo yêu cầu cụ thể.

- Khó khăn trong việc theo dõi và xác nhận các yêu cầu nghỉ phép, dễ gây ra sự cố trùng lặp hoặc không xác định.

**Giải pháp đề xuất:**

Dựa trên đặc tả bài toán và phân tích hiện trạng, cần phải đánh giá các giải pháp công nghệ phù hợp như các phần mềm quản lý chấm công hoặc phát triển một hệ thống tùy chỉnh để đáp ứng các yêu cầu cụ thể của công ty. Cần xem xét các yếu tố như hiệu suất, chi phí và tính linh hoạt để chọn lựa giải pháp phù hợp nhất cho công ty công nghệ.

## 1.3 Yêu cầu chức năng

- Quản lý thông tin cá nhân: Cho phép nhập và cập nhật thông tin cá nhân của nhân viên như tên, địa chỉ, số điện thoại, email, v.v.

- Hỗ trợ quản lý ảnh đại diện và các tài liệu đính kèm: Hồ sơ công việc: Lưu trữ thông tin về vị trí công việc, bộ phận, chức vụ và các thông tin liên quan đến nhiệm vụ công việc của nhân viên.

- Chấm công lưu lại lịch sử làm việc: Ghi nhận và hiển thị lịch sử làm việc của nhân viên bao gồm ngày bắt đầu làm việc, thay đổi chức vụ, thưởng phạt,..

- Quản lý thông tin bảo hiểm, phụ cấp và bằng cấp chứng chỉ: Cho phép nhập và quản lý thông tin về bảo hiểm, phụ cấp, bằng cấp và chứng chỉ của nhân viên.

- Quản lý hợp đồng lao động: Lưu trữ thông tin về các hợp đồng lao động của nhân viên bao gồm ngày ký, thời hạn, mức đóng.

- Đăng ký lịch nghỉ phép: Cho phép nhân viên đăng ký lịch nghỉ phép theo các loại như nghỉ phép hàng năm, nghỉ ốm, nghỉ thai sản,... Cung cấp giao diện đơn giản và dễ sử dụng để nhân viên có thể nhập thông tin đăng ký lịch nghỉ phép một cách nhanh chóng.

- Xác nhận và phê duyệt: Hệ thống cần cung cấp chức năng xác nhận và phê duyệt đơn đăng ký nghỉ phép từ phía quản lý hoặc người có thẩm quyền. Quản lý có thể xem và duyệt đơn đăng ký nghỉ phép thông qua giao diện quản trị của hệ thống.

- Đánh giá hiệu suất: Cung cấp tính năng đánh giá hiệu suất làm việc của nhân viên dựa trên các tiêu chí đã được định sẵn.

## 1.4 Yêu cầu phi chức năng

- Bảo mật: Đảm bảo tính bảo mật của dữ liệu cá nhân và quản lý quyền truy cập thông tin dựa trên vai trò của người dùng.

- Truy cập từ xa: Hỗ trợ truy cập từ xa để nhân viên có thể cập nhật thông tin cá nhân và làm việc từ xa một cách thuận tiện.

- Hiệu suất: Đảm bảo website có thể hoạt động một cách mượt mà và nhanh chóng để tăng cường hiệu suất làm việc của người dùng.

- Tính linh hoạt: Cung cấp giao diện linh hoạt và dễ tùy chỉnh để phản ánh được nhu cầu và quy trình làm việc cụ thể của từng công ty.

Qua phân tích khảo sát hiện trạng, chúng tôi đề xuất phát triển một hệ thống quản lý nhân sự cho công ty chạy trên môi trường web đa nền tảng. Sử dụng công nghệ như: Angular Framework, Spring Framework và một vài thư viện UI/UX được giới thiệu tổng quan trong phần 1.4 dưới đây.

## 1.5 Giới thiệu về công nghệ và ngôn ngữ sử dụng

### 1.5.1 Angular framework

***1.5.1.1 Tổng quan.***

Angular [11] là một framework mã nguồn mở được phát triển bởi Google, được sử dụng để xây dựng các ứng dụng web động cao cấp và đa trang (SPA - Single Page Applications). Angular cho phép các nhà phát triển tạo ra các ứng dụng web mạnh mẽ bằng cách sử dụng TypeScript, một ngôn ngữ lập trình được biên dịch thành JavaScript.

Angular được xây dựng dựa trên các nguyên lý thiết kế mạnh mẽ và phổ biến như MVC (Model-View-Controller) và MVVM (Model-View-ViewModel). Nó cung cấp một cách tiếp cận cấu trúc và tổ chức cho việc phát triển ứng dụng web, giúp tạo ra các ứng dụng linh hoạt, dễ bảo trì và có thể mở rộng.

***1.5.1.2 Thư viện hỗ trợ xây dựng giao diện***

Ant Design [12] là một thư viện giao diện người dùng (UI library) phổ biến được phát triển bởi Ant Design Team của Trung Quốc. Thư viện này cung cấp các thành phần giao diện hiện đại và linh hoạt cho ứng dụng web, giúp các nhà phát triển tạo ra giao diện đẹp mắt và dễ sử dụng.

Ant Design Angular là một phiên bản của Ant Design được tối ưu hóa cho việc sử dụng trong ứng dụng Angular. Nó cung cấp một bộ các thành phần giao diện và công cụ hữu ích để phát triển ứng dụng web trong môi trường Angular một cách nhanh chóng và dễ dàng.

DevExtreme là một thư viện giao diện người dùng (UI library) phổ biến được phát triển bởi DevExpress, cung cấp một bộ các thành phần giao diện và công cụ hữu ích cho việc phát triển ứng dụng web trong môi trường Angular.

Thư viện này chú trọng vào việc cung cấp các thành phần giao diện chuyên nghiệp và linh hoạt để giúp các nhà phát triển xây dựng các ứng dụng web một cách nhanh chóng và dễ dàng.

### 1.5.2 Spring Framework

Spring [7] là một framework phát triển ứng dụng Java được sử dụng rộng rãi trong việc xây dựng các ứng dụng enterprise. Nó cung cấp một cách tiếp cận linh hoạt và mạnh mẽ để xây dựng các ứng dụng Java thông qua việc cung cấp các module và công cụ hỗ trợ cho việc phát triển, triển khai và quản lý ứng dụng.

Spring Framework được thiết kế để giúp đơn giản hóa việc phát triển ứng dụng Java bằng cách cung cấp các tính năng như Dependency Injection (DI) để quản lý các phụ thuộc và Inversion of Control (IoC) để kiểm soát việc tạo đối tượng và quản lý vòng đời của chúng. Nó cũng cung cấp các module hỗ trợ các lĩnh vực khác nhau như xử lý web, xử lý dữ liệu, bảo mật, tích hợp các công nghệ khác nhau như Hibernate, JDBC, RESTful, và nhiều công nghệ khác.

Spring Framework được phát triển và duy trì bởi Pivotal Software (hiện đã được VMware mua lại) và đã trở thành một trong những framework phổ biến và quan trọng trong cộng đồng phát triển Java.



Hình 1.1 Thư viện Spring Framework

### 1.5.3 Mô hình RESTful API

Mô hình RESTful API (Representational State Transfer) là một kiến trúc thiết kế cho các dịch vụ web, dựa trên các nguyên tắc và quy tắc mô tả trong tài liệu đặc tả HTTP. Dưới đây là một số điểm chính trong việc giới thiệu mô hình RESTful API:



Hình 1.2 Mô hình RESTful API

Kiến trúc dựa trên các tài nguyên (resources): Trong mô hình RESTful, mỗi đối tượng hoặc tài nguyên của hệ thống (ví dụ: người dùng, bài viết, hình ảnh) được đại diện bằng một URI (Uniform Resource Identifier). Các tài nguyên này có thể được thao tác thông qua các phương thức HTTP như GET, POST, PUT, DELETE.

Sử dụng các phương thức HTTP: RESTful API sử dụng các phương thức HTTP như GET, POST, PUT, DELETE để thực hiện các hoạt động CRUD (Create, Read, Update, Delete) trên tài nguyên.

Stateless (Không lưu trạng thái): Mô hình RESTful là stateless, điều này có nghĩa là mỗi yêu cầu từ máy khách đều đủ thông tin cần thiết cho máy chủ để hiểu và xử lý yêu cầu, không cần dựa vào bất kỳ trạng thái nào được lưu trữ trước đó.

Cung cấp các quy tắc thiết kế mô-đun (modular): RESTful API thúc đẩy việc chia nhỏ các chức năng và dữ liệu thành các module nhỏ, dễ quản lý và tái sử dụng. Điều này giúp cải thiện sự linh hoạt và khả năng mở rộng của hệ thống.

### 1.5.4 Hệ quản trị cơ sở dữ liệu quan hệ MySQL

***1.5.4.1 Tổng quan về MySQL***

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở phổ biến, được phát triển, phân phối và hỗ trợ bởi Oracle Corporation. Nó là một trong những hệ quản trị cơ sở dữ liệu phổ biến nhất trên thế giới, được sử dụng rộng rãi cho các ứng dụng web, doanh nghiệp và ứng dụng di động.



Hình 1.3 Hệ quản trị cơ sở dữ liệu MySQL

***1.5.4.2 Các thành phần cơ bản của MySQL***

MySQL bao gồm các thành phần sau:

* Là trung tâm của hệ thống MySQL, xử lý các yêu cầu từ các client, quản lý và tạo ra các quyền truy cập, và điều chỉnh các tác vụ khác nhau như backup, replication, và clustering.
* Cơ sở dữ liệu: Là nơi lưu trữ tất cả dữ liệu, được tổ chức thành các bảng, mỗi bảng chứa một tập hợp các hàng và cột.
* Bảng (Table): Là cấu trúc cơ bản của dữ liệu trong MySQL, được sắp xếp thành các hàng và cột. Mỗi hàng đại diện cho một bản ghi cụ thể, trong khi mỗi cột đại diện cho một trường dữ liệu cụ thể.
* Câu lệnh SQL (Structured Query Language): Là ngôn ngữ sử dụng để tương tác với cơ sở dữ liệu MySQL. SQL cho phép thực hiện các thao tác như truy vấn, cập nhật, xóa và chèn dữ liệu.

***1.5.4.3 Sự tương tác giữa MySQL và SpringBoot***

Spring Boot là một framework Java được sử dụng phổ biến trong phát triển ứng dụng web. Khi tích hợp với MySQL, Spring Boot cung cấp các tính năng như tạo, đọc, cập nhật và xóa dữ liệu từ cơ sở dữ liệu MySQL thông qua các phương thức RESTful API hoặc các tầng dịch vụ. Điều này giúp cho việc phát triển ứng dụng web sử dụng cơ sở dữ liệu MySQL trở nên đơn giản và hiệu quả.

# CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Chương 2 sẽ đưa ra những phương pháp và kỹ thuật cần thiết để phân tích yêu cầu, thiết kế cấu trúc và chức năng của hệ thống, từ đó tạo nền tảng vững chắc cho quá trình phát triển và vận hành.

## 2.1 Mô hình hóa chức năng

### 2.1.1 Xác định các Actor (Tác nhân)

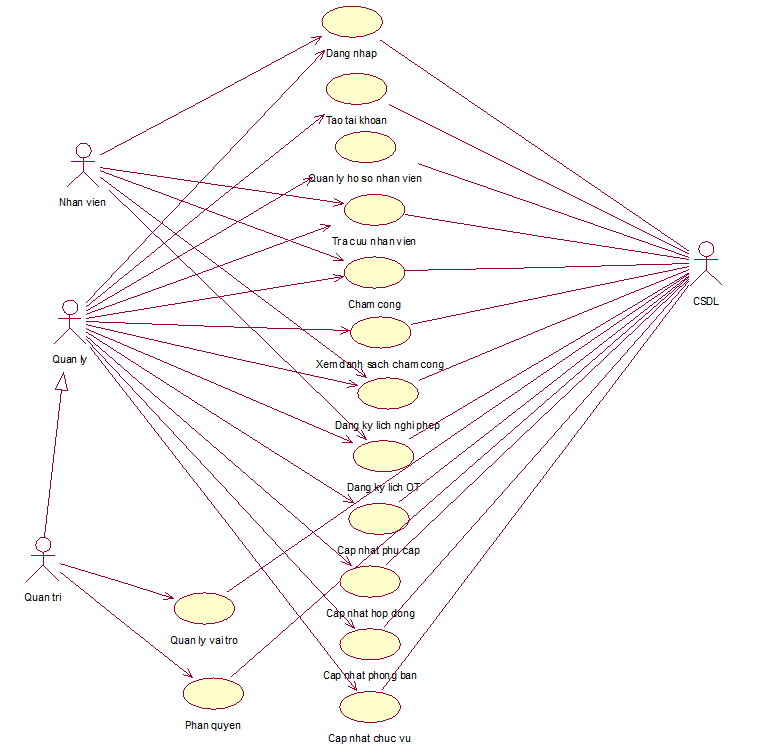
Dựa vào yêu cầu, ta có các Actor sau: Quản trị viên, Quản lý, Nhân viên.

Bảng 2.1 Danh sách các tác nhân của hệ thống

|  |  |
| --- | --- |
| Actor | Mô tả actor |
| Quản trị viên (Admin) | Người có quyền cao nhất trong hệ thống. Đăng nhập, đăng ký, quên mật khẩu, tạo tài khoản, quản lý cây phân cấp, quản lý vai trò, quản lý hồ sơ nhân sự, quản lý chấm công, quản lý danh mục, xuất báo cáo thống kê, cấu hình giao diện. |
| Quản lý | Người có quyền quản lý các chức năng trong hệ thống trừ một vài chức năng như phần quyền, quản lý vai trò, cấu hình giao diện.  Chịu sự quản lý của quản trị viên |
| Nhân viên | Nhân viên trong công ty, người chỉ được cung cấp một vài chức năng như xem hồ sơ nhân viên cá nhân, quản lý chấm công, đăng ký lịch nghỉ phép, đăng ký làm OT, xuất báo cáo thống kê. |
| CSDL | Hệ thống cơ sở dữ liệu đóng vai trò lưu trữ dữ liệu, bảo mật dữ liệu, hỗ trợ truy vấn và sao lưu khôi phục dữ liệu. |

### 2.1.2 Biểu đồ Use Case

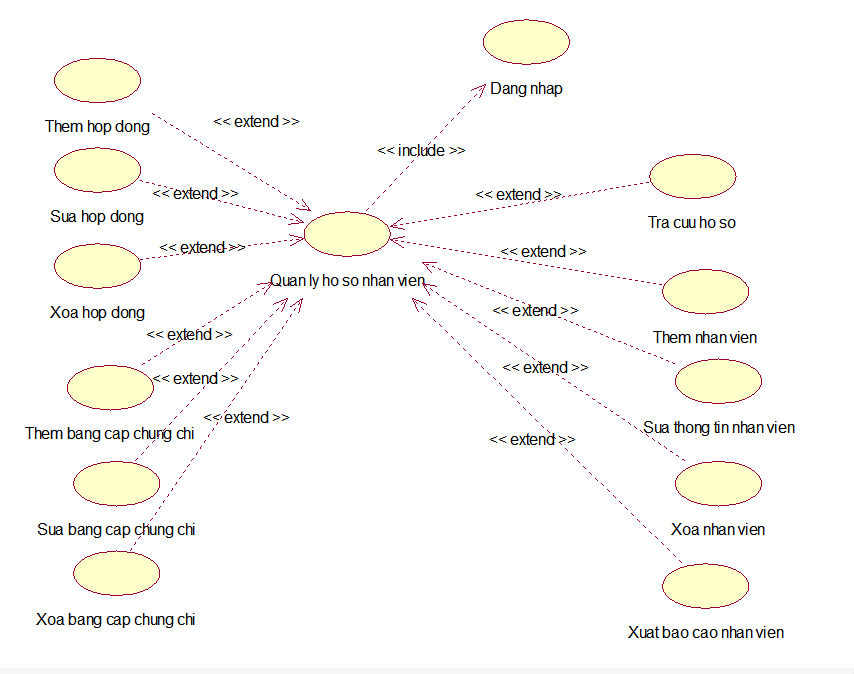
#### *2.1.2.1 Biểu đồ use case tổng quát*



Hình 2.1 Biểu đồ use case tổng quát của hệ thống

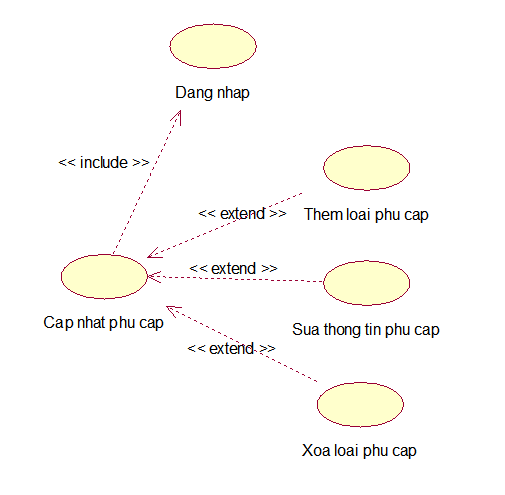
#### *2.1.2.2 Mô tả chi tiết giữa các Use case*

**Quản lý hồ sơ nhân viên**



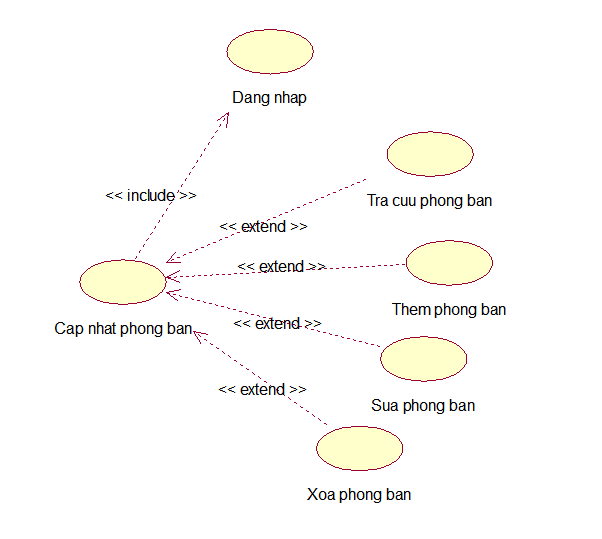
Hình 2.2 Phân rã use case quản lý hồ sơ nhân viên

**Cập nhật thông tin phụ cấp**



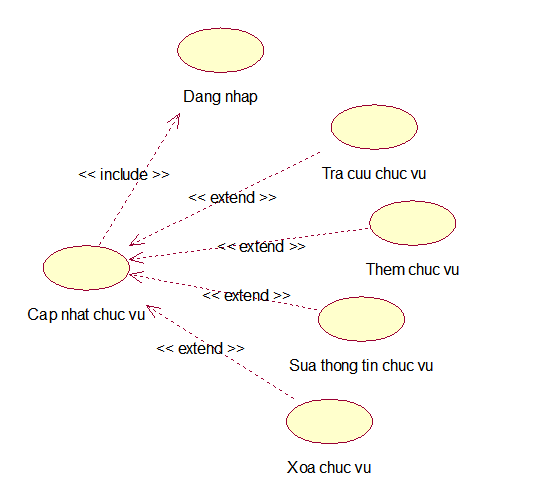
Hình 2.3 Phân rã use case cập nhật thông tin phụ cấp

**Cập nhật phòng ban**

****

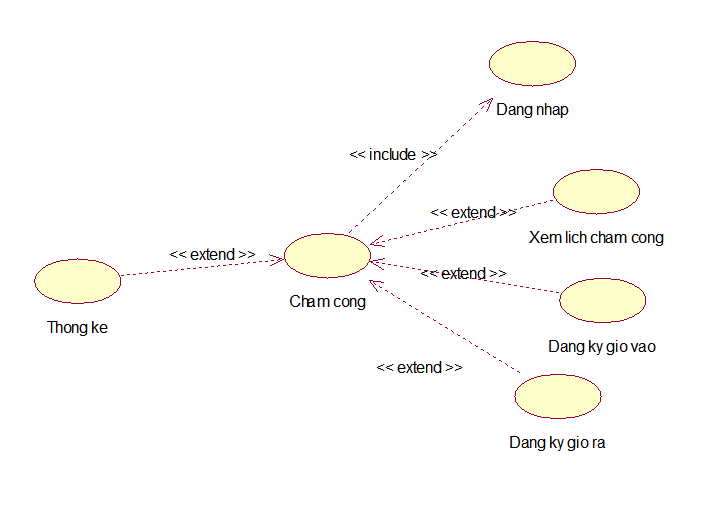
Hình 2.4 Phân rã giữa use case cập nhật phòng ban

**Cập nhật chức vụ**

****

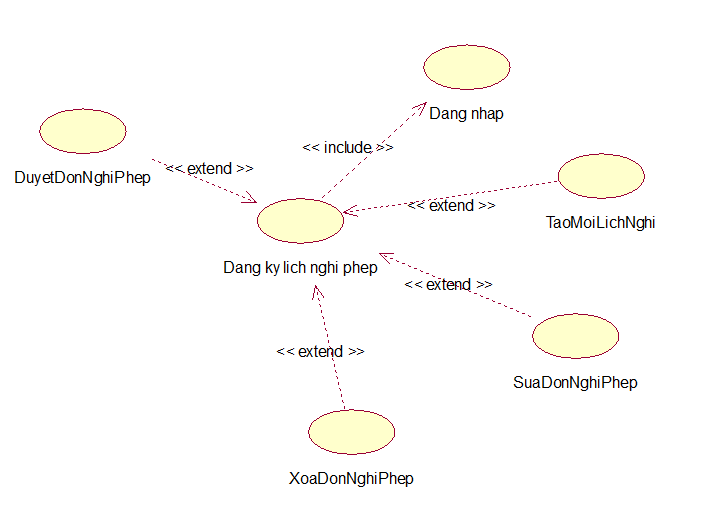
Hình 2.5 Phân rã giữa case cập nhật chức vụ

**Chấm công**

****

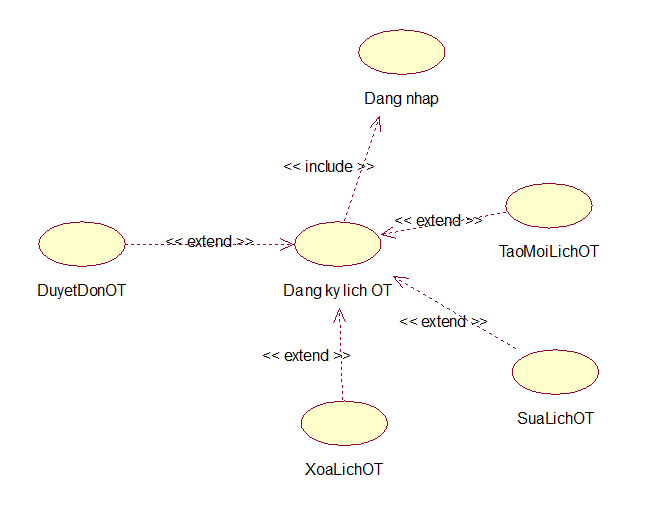
Hình 2.6 Phân rã giữa use case chấm công

**Đăng ký đơn nghỉ phép**

****

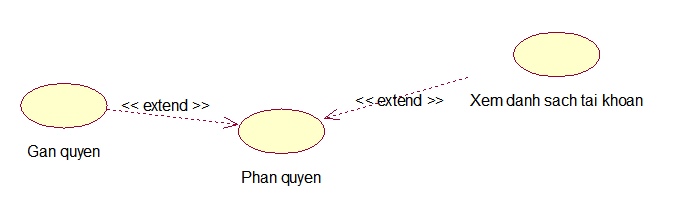
Hình 2.7 Phân rã giữa use case đăng ký đơn nghỉ phép

**Đăng ký lich OT**

****

Hình 2.8 Phân rã giữa use case đăng ký lịch OT

**Phân quyền**

**

Hình 2.9 Phân rã giữa use case phân quyền

## 2.2 Đặc tả chi tiết các use case

### 2.2.1 Đặc tả use case Đăng nhập

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.10 Biểu đồ use case đăng nhập

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này cho phép người dùng (nhân viên & quản lý) đăng nhập vào hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng muốn đăng nhập vào hệ thống. Hệ thống sẽ hiển thị màn hình yêu cầu nhập tên đăng nhập và mật khẩu.
2. Người dùng nhập email và mật khẩu sau đó click vào nút “Đăng nhập”. Hệ thống kiểm tra email, mật khẩu và quyền truy nhập trong bảng “tblTaiKhoan” và hiển thị màn hình hệ thống với các menu tương ứng với quyền. Use case kết thúc
   * **Luồng rẽ nhánh:**
3. Tại bước 2 trong luồng cơ bản nếu người dùng không nhập email hoặc không nhập mật khẩu hệ thống sẽ hiển thị thông báo “Cần phải nhập đủ thông tin!”. Người dùng cần nhập đủ thông tin để tiếp tục.
4. Tại bước 2 trong luồng cơ bản nếu người dùng nhập không đúng định dạng email hệ thống sẽ hiển thị thông báo “Sai định dạng email!”. Người dùng cần nhập đủ thông tin để tiếp tục.
5. Tại bước 2 trong luồng cơ bản nếu người dùng nhập sai tên hoặc/và mật khẩu thì hệ thống sẽ hiển thị thông báo lỗi “Thông tin đăng nhập không đúng!” và use case kết thúc.
6. Tại bước 2 trong luồng cơ bản nếu người dùng kích vào nút “Hủy bỏ” thì use case kết thúc.
7. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo “Lỗi kết nối!” và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Người dùng cần có một tài khoản trước khi có thể đăng nhập.

* **Tiền điều kiện:**

Người dùng cần có một tài khoản trước khi có thể đăng nhập.

* **Hậu điều kiện:**

Không có.

* **Điểm mở rộng:**

Không có.

### 2.2.2 Đặc tả use case Quản lý hồ sơ nhân viên

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.11 Biểu đồ use case quản lý hồ sơ nhân viên

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này cho phép người dùng có vai trò là quản lý tạo mới, cập nhật hồ sơ nhân viên.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Quản trị viên hoặc quản lý truy cập vào hệ thống quản lý hồ sơ nhân viên.
2. Hệ thống hiển thị danh sách các nhân viên hiện có.
3. Quản lý có thể thêm, sửa đổi hoặc xóa thông tin về nhân viên.
4. Quản lý nhập thông tin mới hoặc chỉnh sửa thông tin của nhân viên.
5. Hệ thống xác nhận và lưu thông tin mới hoặc cập nhật thông tin nhân viên.
6. Hệ thống cập nhật lại danh sách nhân viên và hiển thị thông báo thành công.
7. Quản lý sẽ truy cập vào thông tin chi tiết của nhân viên, sau đó sẽ thêm hoặc sửa đổi các thông tin phụ chi tiết cho nhân viên như quản lý hợp đồng, bảo hiểm xã hội, thông tin phụ cấp và bằng cấp chứng chỉ.
8. Hệ thống sẽ xác nhận và lưu thông tin chi tiết mới hoặc cập nhật thông tin chi tiết nhân viên.
9. Hệ thống sẽ cập nhật lại hồ sơ chi tiết của nhân viên và hiển thị thông báo thành công trên hệ thống. Use case kết thúc.
   * **Luồng rẽ nhánh:**
   1. Nếu quản lý chọn thêm mới nhân viên:

* Quản lý nhập thông tin mới cho nhân viên.
* Hệ thống kiểm tra tính hợp lệ của thông tin.
* Nếu thông tin hợp lệ, hệ thống thêm mới nhân viên và thông báo thành công.
* Nếu thông tin không hợp lệ, hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại.
  1. Nếu quản lý chọn chỉnh sửa thông tin nhân viên:
* Quản lý chọn nhân viên cần chỉnh sửa.
* Hệ thống hiển thị thông tin chi tiết của nhân viên.
* Quản lý chỉnh sửa thông tin cần thiết.
* Hệ thống xác nhận và cập nhật thông tin mới của nhân viên.
* Hệ thống thông báo thành công.
  1. Nếu quản lý chọn xóa nhân viên:
* Quản lý chọn nhân viên cần xóa.
* Hệ thống yêu cầu xác nhận xóa nhân viên.
* Quản lý xác nhận xóa.
* Hệ thống xóa phòng ban khỏi danh sách và thông báo thành công.
* **Các yêu cầu đặc biệt:**

Đăng nhập thành công vào hệ thống với vai trò là quản lý.

* **Tiền điều kiện:**

Người dùng cần có một tài khoản với vai trò là quản lý trước khi có thể đăng nhập.

* **Hậu điều kiện:**

Không có.

* **Điểm mở rộng:**

Không có.

### 2.2.3 Đặc tả use case Tra cứu hồ sơ nhân viên

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.12 Biểu đồ use case tra cứu hồ sơ nhân viên

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này mô tả quá trình tra cứu thông tin hồ sơ của nhân viên trong hệ thống

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Người dùng chọn chức năng "Tra cứu hồ sơ nhân viên" từ danh mục.
2. Hệ thống hiển thị giao diện hoặc biểu mẫu tra cứu thông tin nhân viên.
3. Người dùng nhập các thông tin cần thiết như tên, mã nhân viên, tên nhân viên, phòng ban.
4. Người dùng xác nhận yêu cầu tra cứu.
5. Hệ thống tìm kiếm trong cơ sở dữ liệu các hồ sơ nhân viên phù hợp với thông tin đã nhập.
6. Hệ thống hiển thị kết quả tra cứu cho người dùng, bao gồm các thông tin chi tiết về nhân viên tìm thấy.
   * **Luồng rẽ nhánh:**
   1. . Nếu người dùng không nhập thông tin cần thiết hoặc nhập thông tin không hợp lệ: Hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại thông tin.

4. Nếu không có hồ sơ nhân viên nào phù hợp với thông tin tra cứu:

- Hệ thống hiển thị thông báo rằng không tìm thấy kết quả phù hợp.

- Người dùng có thể thử lại với các thông tin khác hoặc quay lại bước 4.

* **Các yêu cầu đặc biệt:**

Đăng nhập thành công vào hệ thống với vai trò là quản lý hoặc nhân viên.

* **Tiền điều kiện:**

Người dùng cần có một tài khoản với vai trò là quản lý hoặc nhân viên của công ty trước khi có thể đăng nhập.

* **Hậu điều kiện:**

Không có.

* **Điểm mở rộng:**

Không có.

### 2.2.4 Đặc tả use case Quản lý phòng ban

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.13 Biểu đồ use case quản lý phòng ban

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này cho phép quản lý thông tin về các phòng ban trong hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Quản trị viên hoặc quản lý cấp cao truy cập vào hệ thống quản lý phòng ban.
2. Hệ thống hiển thị danh sách các phòng ban hiện có.
3. Quản lý có thể thêm, sửa đổi hoặc xóa thông tin về phòng ban.
4. Quản lý nhập thông tin mới hoặc chỉnh sửa thông tin của phòng ban.
5. Hệ thống xác nhận và lưu thông tin mới hoặc cập nhật thông tin phòng ban.
6. Hệ thống cập nhật lại danh sách phòng ban và hiển thị thông báo thành công.
   * **Luồng rẽ nhánh:**
   1. . Nếu quản lý chọn thêm mới phòng ban:

- Quản lý nhập thông tin mới cho phòng ban.

- Hệ thống kiểm tra tính hợp lệ của thông tin.

- Nếu thông tin hợp lệ, hệ thống thêm mới phòng ban và thông báo thành công.

- Nếu thông tin không hợp lệ, hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại.

* 1. . Nếu quản lý chọn chỉnh sửa thông tin phòng ban:

- Quản lý chọn phòng ban cần chỉnh sửa.

- Hệ thống hiển thị thông tin chi tiết của phòng ban.

- Quản lý chỉnh sửa thông tin cần thiết.

- Hệ thống xác nhận và cập nhật thông tin mới của phòng ban.

- Hệ thống thông báo thành công.

* 1. . Nếu quản lý chọn xóa phòng ban:

- Quản lý chọn phòng ban cần xóa.

- Hệ thống yêu cầu xác nhận xóa phòng ban.

- Quản lý xác nhận xóa.

- Hệ thống xóa phòng ban khỏi danh sách và thông báo thành công.

* **Các yêu cầu đặc biệt:**

Đăng nhập thành công vào hệ thống với vai trò là quản lý.

* **Tiền điều kiện:**

Người dùng cần có một tài khoản với vai trò là quản lý trước khi có thể đăng nhập.

* **Hậu điều kiện:**

Thông tin về phòng ban được cập nhật hoặc thêm mới thành công.

* **Điểm mở rộng:**

Không có.

### 2.2.5 Đặc tả use case Quản lý chức vụ

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.14 Biểu đồ use case quản lý chức vụ

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này cho phép quản lý thông tin về các chức vụ từng phòng ban trong hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Quản trị viên hoặc quản lý cấp cao truy cập vào hệ thống quản lý chức vụ.
2. Hệ thống hiển thị danh sách các chức vụ hiện có.
3. Người dùng có thể thêm, sửa đổi hoặc xóa thông tin về chức vụ.
4. Người dùng nhập thông tin mới hoặc chỉnh sửa thông tin của chức vụ.
5. Hệ thống xác nhận và lưu thông tin mới hoặc cập nhật thông tin chức vụ.
6. Hệ thống cập nhật lại danh sách chức vụ và hiển thị thông báo thành công.
   * **Luồng rẽ nhánh:**
   1. . Tại bước 3 nếu quản lý chọn thêm mới chức vụ:

- Quản lý nhập thông tin mới cho chức vụ.

- Hệ thống kiểm tra tính hợp lệ của thông tin.

- Nếu thông tin hợp lệ, hệ thống thêm mới chức vụ và thông báo thành - công.

- Nếu thông tin không hợp lệ, hệ thống hiển thị thông báo lỗi và yêu cầu quản lý nhập lại.

3.2. Tại bước nếu quản lý chọn chỉnh sửa thông tin chức vụ:

- Quản lý chọn chức vụ cần chỉnh sửa.

- Hệ thống hiển thị thông tin chi tiết của chức vụ.

- Quản lý chỉnh sửa thông tin cần thiết.

- Hệ thống xác nhận và cập nhật thông tin mới của chức vụ.

- Hệ thống thông báo thành công.

3.3. Tại bước nếu quản lý chọn xóa chức vụ:

- Quản lý chọn chức vụ cần xóa.

- Hệ thống yêu cầu xác nhận xóa chức vụ.

- Quản lý xác nhận xóa.

- Hệ thống xóa chức vụ khỏi danh sách và thông báo thành công.

* **Các yêu cầu đặc biệt:**

Đăng nhập thành công vào hệ thống với vai trò là quản lý.

* **Tiền điều kiện:**

Người dùng cần có một tài khoản với vai trò là quản lý trước khi có thể đăng nhập.

* **Hậu điều kiện:**

Thông tin về chức vụ được cập nhật hoặc thêm mới thành công.

* **Điểm mở rộng:**

Không có.

### 2.2.6 Đặc tả use case Quản lý chấm công

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.15 Biểu đồ use case chấm công

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này mô tả quy trình chấm công hàng ngày của nhân viên trong hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Nhân viên truy cập vào chức năng chấm công trong hệ thống.
2. Hệ thống hiển thị giao diện chấm công cho nhân viên.
3. Nhân viên chọn loại chấm công (ví dụ: vào làm, tan làm...).
4. Nhân viên xác nhận thời gian chấm công.
5. Hệ thống ghi nhận thời gian chấm công vào cơ sở dữ liệu.
6. Hệ thống hiển thị thông báo xác nhận thành công cho nhân viên.
   * **Luồng rẽ nhánh:**
   1. Tại bước 4, nếu hệ thống gặp lỗi khi ghi nhận thời gian chấm công:

- Hệ thống hiển thị thông báo lỗi cho nhân viên.

- Nhân viên thử lại hoặc liên hệ với bộ phận hỗ trợ kỹ thuật để giải quyết vấn đề.

* **Các yêu cầu đặc biệt:**

Đăng nhập thành công vào hệ thống với vai trò là quản lý hoặc nhân viên.

* **Tiền điều kiện:**

Nhân viên đã đăng nhập vào hệ thống.

Đến thời điểm chấm công.

* **Hậu điều kiện:**

Thời gian làm việc của nhân viên được ghi nhận trong hệ thống.

* **Điểm mở rộng:**

Không có.

### 2.2.7 Đặc tả use case Đăng ký lịch nghỉ phép

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.16 Biểu đồ use case đăng ký lịch nghỉ phép

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này mô tả quy trình đăng ký lịch nghỉ phép của nhân viên trong hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Nhân viên truy cập vào chức năng đăng ký lịch nghỉ phép trong hệ thống.
2. Hệ thống hiển thị giao diện đăng ký lịch nghỉ phép cho nhân viên.
3. Nhân viên chọn loại nghỉ phép (nghỉ phép có lương, không lương, nghỉ phép bệnh, nghỉ phép cưới...).
4. Nhân viên chọn ngày bắt đầu và kết thúc của khoảng thời gian nghỉ phép.
5. Nhân viên cung cấp lý do/nội dung cho việc đăng ký nghỉ phép.
6. Nhân viên xác nhận việc đăng ký lịch nghỉ phép.
7. Hệ thống gửi yêu cầu đăng ký lịch nghỉ phép cho quản lý hoặc người phê duyệt tương ứng.
8. Hệ thống ghi nhận thông tin đăng ký lịch nghỉ phép vào cơ sở dữ liệu.
9. Hệ thống hiển thị thông báo xác nhận đăng ký lịch nghỉ phép thành công cho nhân viên.
   * **Luồng rẽ nhánh:**
   1. Nếu nhân viên muốn thay đổi hoặc hủy bỏ lịch nghỉ phép đã đăng ký:

- Nhân viên truy cập vào lịch sử đăng ký lịch nghỉ phép trong hệ thống.

- Nhân viên chọn bản ghi lịch nghỉ phép cần chỉnh sửa hoặc hủy bỏ.

- Nhân viên thực hiện chỉnh sửa hoặc hủy bỏ lịch nghỉ phép.

- Nhân viên xác nhận thay đổi hoặc hủy bỏ lịch nghỉ phép.

- Hệ thống cập nhật lại thông tin và gửi thông báo xác nhận cho nhân viên.

* 1. Nếu quản lý/phê duyệt từ chối yêu cầu đăng ký lịch nghỉ phép:

- Hệ thống gửi thông báo từ chối đến nhân viên.

- Nhân viên có thể xem lại và sửa đổi yêu cầu đăng ký lịch nghỉ phép hoặc liên hệ với quản lý để biết thêm chi tiết.

1. Tại bước 8, nếu hệ thống gặp lỗi khi ghi nhận thông tin đăng ký lịch nghỉ phép:

- Hệ thống hiển thị thông báo lỗi cho nhân viên.

- Nhân viên thử lại hoặc liên hệ với bộ phận hỗ trợ kỹ thuật để giải quyết vấn đề.

* **Các yêu cầu đặc biệt:**

Nhân viên đã đăng nhập vào hệ thống.

* **Tiền điều kiện:**

Người dùng cần có một tài khoản với vai trò là nhân viên của công ty trước khi có thể đăng nhập.

* **Hậu điều kiện:**

Yêu cầu đăng ký lịch nghỉ phép được gửi thành công và chờ phê duyệt từ quản lý hoặc người theo dõi.

* **Điểm mở rộng:**

Không có.

### 2.2.8 Đặc tả use case Đăng ký lịch OT

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.17 Biểu đồ use case đăng ký lịch ot

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này mô tả quy trình đăng ký làm thêm giờ (OT) của nhân viên trong hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Nhân viên truy cập vào chức năng đăng ký làm thêm giờ trong hệ thống.
2. Hệ thống hiển thị giao diện đăng ký làm thêm giờ cho nhân viên.
3. Nhân viên chọn ngày và thời gian muốn làm thêm giờ.
4. Nhân viên cung cấp lý do/nội dung cho việc đăng ký làm thêm giờ.
5. Nhân viên xác nhận việc đăng ký làm thêm giờ.
6. Hệ thống gửi yêu cầu đăng ký làm thêm giờ cho quản lý hoặc người phê duyệt tương ứng.
7. Hệ thống ghi nhận thông tin đăng ký làm thêm giờ vào cơ sở dữ liệu.
8. Hệ thống hiển thị thông báo xác nhận đăng ký làm thêm giờ thành công cho nhân viên.
   * **Luồng rẽ nhánh:**
   1. Nếu nhân viên muốn thay đổi hoặc hủy bỏ lịch OT đã đăng ký:

- Nhân viên truy cập vào lịch sử đăng ký làm thêm giờ trong hệ thống.

- Nhân viên chọn bản ghi OT cần chỉnh sửa hoặc hủy bỏ.

- Nhân viên thực hiện chỉnh sửa hoặc hủy bỏ lịch OT.

- Nhân viên xác nhận thay đổi hoặc hủy bỏ lịch OT.

- Hệ thống cập nhật lại thông tin và gửi thông báo xác nhận cho nhân viên.

* 1. Nếu quản lý/phê duyệt từ chối yêu cầu đăng ký OT:

- Hệ thống gửi thông báo từ chối đến nhân viên.

- Nhân viên có thể xem lại và sửa đổi yêu cầu đăng ký OT hoặc liên hệ - với quản lý để biết thêm chi tiết.

7 Tại bước 7, nếu hệ thống gặp lỗi khi ghi nhận thông tin đăng ký làm thêm giờ:

- Hệ thống hiển thị thông báo lỗi cho nhân viên.

- Nhân viên thử lại hoặc liên hệ với bộ phận hỗ trợ kỹ thuật để giải quyết vấn đề.

* **Các yêu cầu đặc biệt:**

Đăng nhập thành công vào hệ thống với vai trò là quản lý hoặc nhân viên.

* **Tiền điều kiện:**

Nhân viên đã đăng nhập vào hệ thống.

* **Hậu điều kiện:**

Yêu cầu đăng ký OT được gửi thành công và chờ phê duyệt từ quản lý.

* **Điểm mở rộng:**

Không có.

### 2.2.9 Đặc tả use case Quản lý vai trò

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.18 Biểu đồ use case quản lý vai trò

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này cho phép quản trị viên thêm vai trò, thay đổi trang thái vai trò, sửa vai trò và gán quyền cho vai trò.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. UC bắt đầu khi quản trị viên kích vào quản lý vai trò trong màn hình quản trị viên.
2. Thêm vai trò: Quản trị viên nhập thông tin vào form thêm vai trò bao gồm tên vai trò, mô tả. Sau đó ấn vào nút lưu hệ thống sẽ lưu vai trò vào bảng Roles.
3. Sửa vai trò: Quản trị viên ấn vào nút sửa vai trò trong danh sách vai trò. Hệ thống sẽ lấy ra thông tin vai trò đó và hiển thị lên màn hình chỉnh sửa vai trò. Sau đó quản trị viên có thể sửa thông tin vai trò và ấn vào nút lưu thì bản ghi sẽ được cập nhật lại trong bảng Role.
4. Thay đổi trạng thái vai trò: Quản trị viên ấn vào nút thay đổi trạng thái, hệ thống sẽ lưu lại thông tin trạng thái vai trò vào bảng Role.
5. Gán quyền: Quản trị viên ấn vào nút gán quyền, hệ thống sẽ hiển thị lên popup gồm tất cả các quyền và quản trị viên sẽ chọn quyền gán vào vai trò và ấn lưu. Thông tin sẽ được lưu lại vào trong cơ sở dữ liệu.
6. UC kết thúc.
   * **Luồng rẽ nhánh:**
7. Tại thời điểm nào mà client không kết nối được server hoặc server không kết nối được cơ sở sở dữ liệu thì hệ thống sẽ xuất hiện thống báo lỗi và UC sẽ kết thúc.

* **Các yêu cầu đặc biệt:**

Đăng nhập thành công vào hệ thống với vai trò là quản lý hoặc nhân viên.

* **Tiền điều kiện:**

Đăng nhập vào quản trị viên và có quyền quản lý vai trò.

* **Hậu điều kiện:**

Không có.

* **Điểm mở rộng:**

Không có.

### 2.2.10 Đặc tả use case Quản lý phân cấp

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.19 Biểu đồ use case quản lý phân cấp

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case cho phép cấu trúc lại thanh menu danh mục trong quản trị viên

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. UC bắt đầu khi quản trị viên kích vào nút quản lý cây phân cấp.
2. Tạo cây danh mục: quản trị viên kích vào nút thêm mới. Hệ thống sẽ hiển thị thông tin như tên danh mục, đường dẫn, icon, thứ tự hiển thị và quyền. Sau khi ấn nút lưu hệ thống sẽ lưu thông tin vào bảng Menus
3. Sửa cây danh mục: quản trị viên kích vào nút sửa, hệ thống sẽ hiển thị thông tin cây danh mục đó và cho phép quản trị viên chỉnh sửa thông tin cây danh mục. Kích vào nút lưu để lưu lại thông tin cây danh mục vào bảng Menus.
4. Thay đổi trạng thái cây danh mục: quản trị viên có thể ẩn hiện cây danh mục bằng nút thay đổi trạng thái cây danh mục.
5. UC kết thúc.
   * **Luồng rẽ nhánh:**
6. Tại thời điểm nào mà client không kết nối được server hoặc server không kết nối được cơ sở sở dữ liệu thì hệ thống sẽ xuất hiện thống báo lỗi và UC sẽ kết thúc.

* **Các yêu cầu đặc biệt:**

Không có

* **Tiền điều kiện:**

Đăng nhập vào quản trị viên và có quyền quản lý cây phân cấp.

* **Hậu điều kiện:**

Không có.

* **Điểm mở rộng:**

Không có.

### 2.2.11 Đặc tả use case Quản lý hợp đồng

a) Biểu đồ use case

|  |
| --- |
|  |

Hình 2.20 Biểu đồ use case quản lý hợp đồng

b) Mô tả use case

**Mô tả vắn tắt use case:** Use case này cho phép quản lý thông tin về các phòng ban trong hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Quản trị viên hoặc quản lý cấp cao truy cập vào hệ thống quản lý hợp đồng.
2. Hệ thống hiển thị danh sách các hợp đồng hiện có.
3. Quản lý có thể thêm, sửa đổi hoặc xóa thông tin về hợp đồng.
4. Quản lý nhập thông tin mới hoặc chỉnh sửa thông tin của hợp đồng.
5. Hệ thống xác nhận và lưu thông tin mới hoặc cập nhật thông tin hợp đồng.
6. Hệ thống cập nhật lại danh sách hợp đồng và hiển thị thông báo thành công.
   * **Luồng rẽ nhánh:**
   1. Nếu quản lý chọn thêm mới hợp đồng:

- Quản lý nhập thông tin mới cho hợp đồng.

- Hệ thống kiểm tra tính hợp lệ của thông tin.

- Nếu thông tin hợp lệ, hệ thống thêm mới hợp đồng và thông báo thành công.

- Nếu thông tin không hợp lệ, hệ thống hiển thị thông báo lỗi và yêu cầu quản lý nhập lại.

* 1. Nếu quản lý chọn chỉnh sửa thông tin hợp đồng:

- Quản lý chọn hợp đồng cần chỉnh sửa.

- Hệ thống hiển thị thông tin chi tiết của hợp đồng.

- Quản lý chỉnh sửa thông tin cần thiết.

- Hệ thống xác nhận và cập nhật thông tin mới của hợp đồng.

- Hệ thống thông báo thành công.

* 1. Nếu quản lý chọn xóa hợp đồng:

- Quản lý chọn phòng ban cần xóa.

- Hệ thống yêu cầu xác nhận xóa hợp đồng.

- Quản lý xác nhận xóa.

- Hệ thống xóa hợp đồng khỏi danh sách và thông báo thành công.

* **Các yêu cầu đặc biệt:**

Đăng nhập thành công vào hệ thống với vai trò là quản lý.

* **Tiền điều kiện:**

Người dùng cần có một tài khoản với vai trò là quản lý trước khi có thể đăng nhập.

* **Hậu điều kiện:**

Thông tin về hợp đồng được cập nhật hoặc thêm mới thành công.

* **Điểm mở rộng:**

Không có.

## 2.3 Mô hình hóa dữ liệu

### 2.3.1 Biểu đồ lớp

|  |
| --- |
|  |

Hình 2.21 Biểu đồ lớp

### 2.3.2 Biểu đồ trình tự

#### *2.3.2.1 Biểu đồ trình tự use case Quản lý hồ sơ nhân viên*

|  |
| --- |
|  |

Hình 2.23 Biểu đồ trình tự Quản lý hồ sơ nhân viên

#### *2.3.2.2 Biểu đồ trình tự use case Tra cứu hồ sơ nhân viên*

|  |
| --- |
|  |

Hình 2.24 Biểu đồ trình tự Tra cứu hồ sơ nhân viên

#### *2.3.2.3 Biểu đồ trình tự use case Quản lý phòng ban*

|  |
| --- |
|  |

Hình 2.25 Biểu đồ trình tự Quản lý phòng ban

#### *2.3.2.4 Biểu đồ trình tự use case Quản lý chức vụ*

|  |
| --- |
|  |

Hình 2.26 Biểu đồ trình tự Quản lý chức vụ

#### *2.3.2.5 Biểu đồ trình tự use case Quản lý chấm công*

|  |
| --- |
|  |

Hình 2.27 Biểu đồ trình tự Quản lý chấm công

#### *2.3.2.6 Biểu đồ trình tự use case Đăng ký lịch nghỉ phép*

|  |
| --- |
|  |

Hình 2.28 Biểu đồ trình tự Đăng ký lịch nghỉ phép

#### *2.3.2.7 Biểu đồ trình tự use case Đăng ký lịch OT*

#### 

|  |
| --- |
|  |

Hình 2.29 Biểu đồ trình tự Đăng ký lịch OT

#### *2.3.2.8 Biểu đồ trình tự use case Quản lý hợp đồng*

|  |
| --- |
|  |

Hình 2.30 Biểu đồ trình tự Quản lý hợp đồng

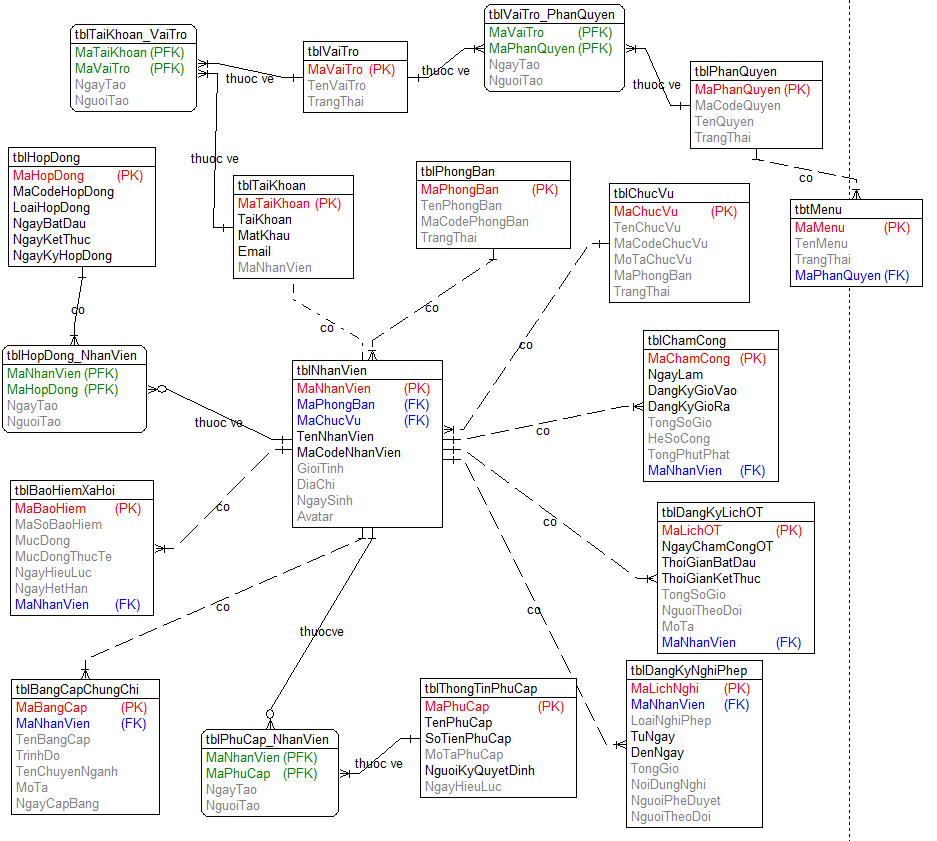
#### *2.3.2.9 Biểu đồ trình tự use case Đăng nhập*

|  |
| --- |
|  |

Hình 2.31 Biểu đồ trình tự Đăng nhập

## 2.4 Thiết kế cơ sở dữ liệu

### 2.4.1 Biểu đồ thực thể liên kết mức logic



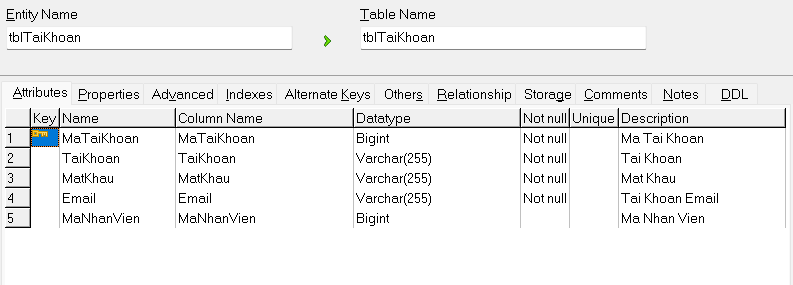
Hình 2.32 Mô hình dữ liệu quan hệ

Dưới đây sẽ trình bày về các bảng trong CSDL quan hệ của hệ thống được xây dựng:

* **tblTaiKhoan**: Lưu thông tin về tài khoản để đăng nhập có quyền truy cập vào trang web. Chi tiết về các tài khoản bao gồm mã tài khoản, tên tài khoản, mật khẩu, email
* **tblVaiTro:** Lưu thông tin về vai trò để đăng nhập có quyền truy cập vào trang web. Chi tiết về bảng vai trò bao gồm mã vai trò, tên vai trò, trạng thái.
* **tblNhanVien**: Lưu thông tin chi tiết về một nhân viên trong công ty Nodo. Thông tin nhân viên được quan tâm là mã nhân viên, tên nhân viên, tên phòng ban, tên chức vụ, giới tính, địa chỉ, ngày sinh, avatar.
* **tblPhongBan:** Lưu thông tin về phòng ban trong công ty. Chi tiết về các phòng ban bao gồm mã phòng ban, tên phòng ban, mô tả phòng ban, trạng thái phòng.
* **tblChucVu:** Lưu trữ thông tin về các chức vụ tồn tại trong công ty. Chi tiết về các chức vụ bao gồm mã chức vụ, tên chức vụ, mô tả chức vụ, thuộc về phòng ban, trạng thái.
* **tblBaoHiemXaHoi:** Lưu thông tin về hồ sơ nhân viên phần bảo hiểm xã hội của nhân viên. Chi tiết về thông tin bảo hiểm xã hội bao gồm mã bảo hiểm xã hội, mã số bảo hiểm xã hội, mức đóng, mức đóng thực tế, ngày hiệu lực, ngày hết hạn.
* **tblThongTinPhuCap:** Lưu thông tin về hồ sơ nhân viên phần thông tin phụ cấp của nhân viên. Chi tiết về thông tin phụ cấp bao gồm mã phụ cấp, tên phụ cấp, số tiền phụ cấp, mô tả phụ cấp, ngày hiệu lực, người ký quyết định.
* **tblBangCapChungChi:** Lưu thông tin về hồ sơ nhân viên phần bằng cấp chứng chỉ của nhân viên. Chi tiết về thông tin bằng cấp chứng chỉ bao gồm mã bằng cấp, tên bằng cấp, loại bằng cấp, trình độ, tên chuyên ngành, mô tả, ngày cấp bằng.
* **tblHopDong:** Lưu thông tin về hồ sơ nhân viên phần hợp đồng của nhân viên. Chi tiết về thông tin hợp đồng bao gồm mã hợp đồng, loại hợp đồng, ngày bắt đầu, ngày kết thúc, ngày ký hợp đồng.
* **tblChamCong:** Lưu thông tin về lịch chấm công của nhân viên trong công ty để ghi nhận thông tin về thời gian làm việc của nhân viên trong một khoảng thời gian nhất định. Bảng này chứa các cột để ghi lại thông tin về ngày làm việc, giờ vào làm, giờ ra làm, số giờ làm việc, hệ số công và tổng phút phạt.
* **tblDangKyLichNghiPhep:** Để ghi nhận thông tin về các yêu cầu nghỉ phép của nhân viên trong công ty. Bảng này thường chứa các thông tin về loại nghỉ phép, thời gian nghỉ, lý do nghỉ, trạng thái của yêu cầu, người phê duyệt và người theo dõi.
* **tblDangKyLichOT:** Được sử dụng để ghi nhận thông tin về các yêu cầu làm thêm giờ của nhân viên trong công ty. Bao gồm ngày chấm công làm thêm, thời gian bắt đầu, thời gian kết thúc, người theo dõi, lý do làm thêm và người theo dõi.

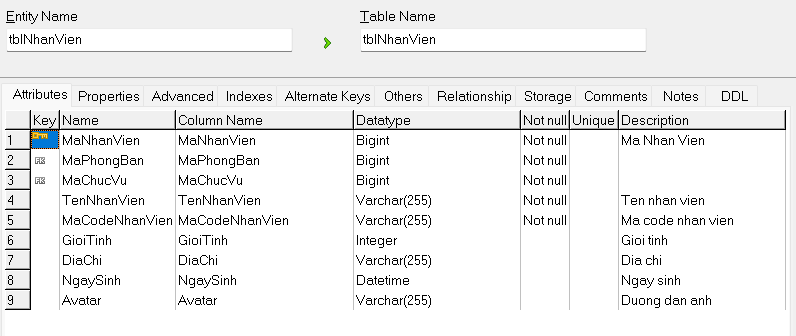
### 2.4.2 Thiết kế bảng

* **Bảng Tài Khoản**



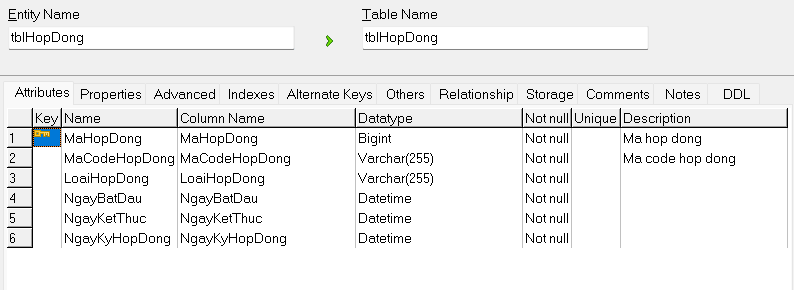
Hình 2.33 Bảng dữ liệu tài khoản

* **Bảng Nhân Viên**



Hình 2.34 Bảng dữ liệu nhân viên

* **Bảng Hợp Đồng**



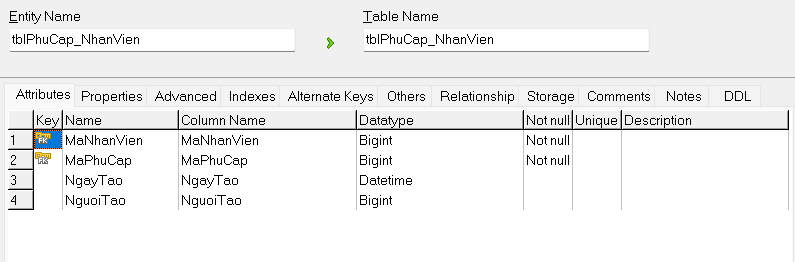
Hình 2.35 Bảng dữ liệu hợp đồng

* **Bảng Hợp Đồng Với Nhân Viên**

****

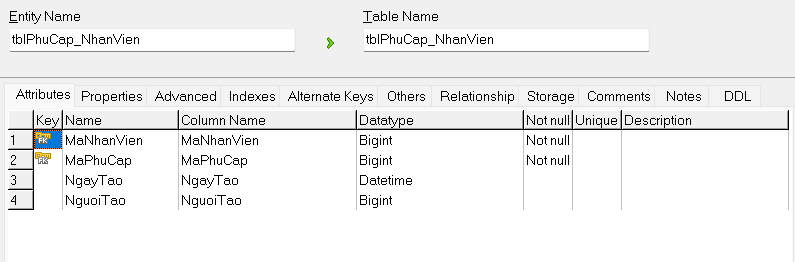
Hình 2.36 Bảng dữ liệu hợp đồng với nhân viên

* **Bảng Phụ Cấp**

****

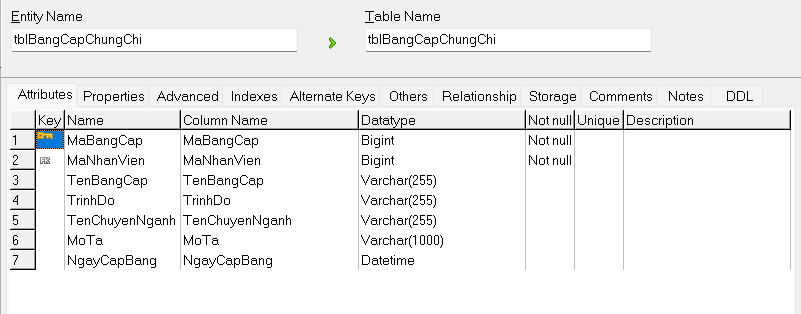
Hình 2.37 Bảng dữ liệu thông tin phụ cấp

* **Bảng Phụ Cấp Với Nhân Viên**

****

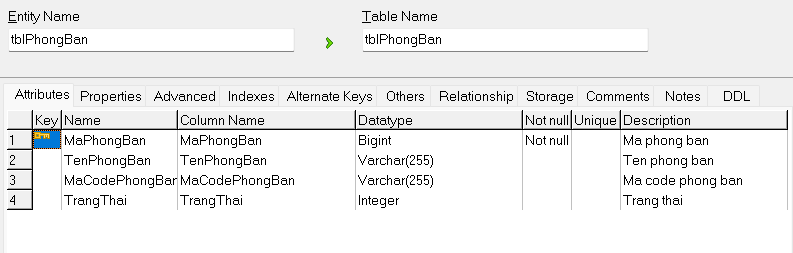
Hình 2.38 Bảng dữ liệu phụ cấp với nhân viên

* **Bảng Bằng Cấp Chứng Chỉ**

****

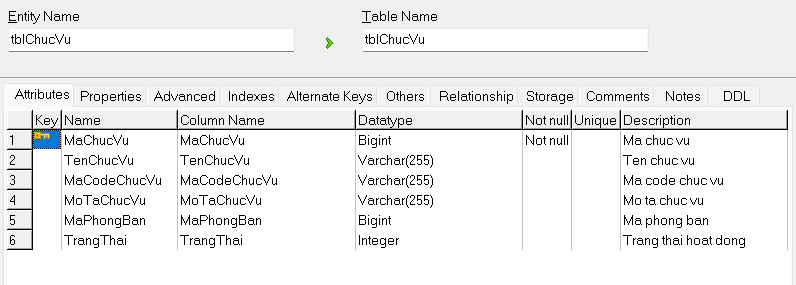
Hình 2.39 Bảng dữ liệu bằng cấp chứng chỉ

* **Bảng Phòng Ban**

****

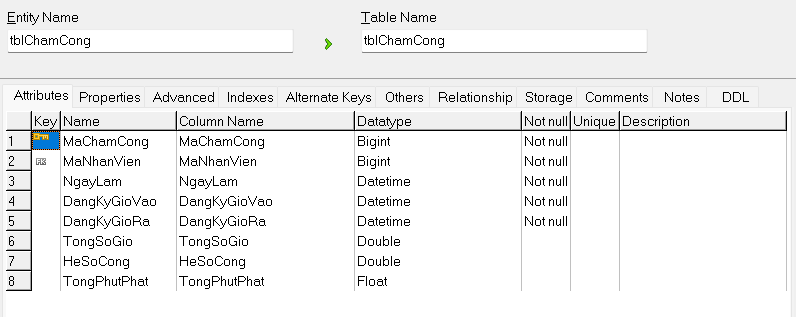
Hình 2.40 Bảng dữ liệu phòng ban

* **Bảng Chức Vụ**

****

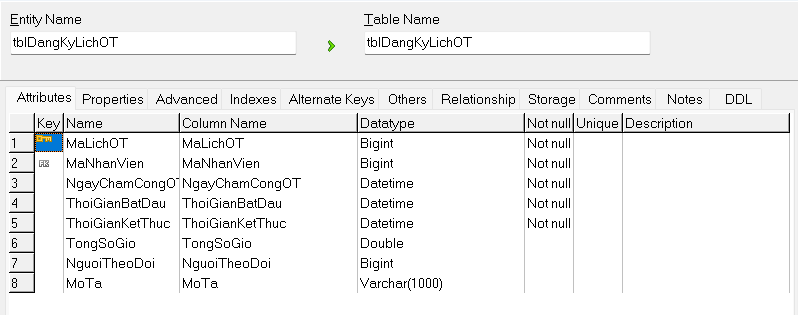
Hình 2.41 Bảng dữ liệu chức vụ

* **Bảng Chấm Công**

****

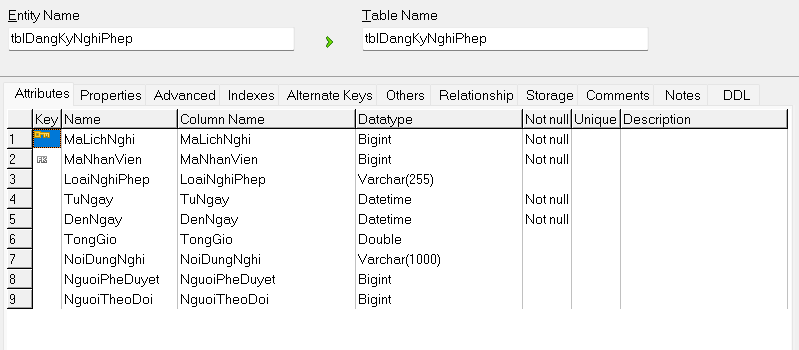
Hình 2.42 Bảng dữ liệu chấm công

* **Bảng Đăng Ký OT**

****

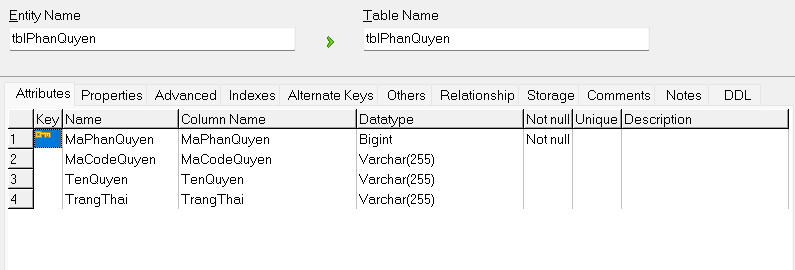
Hình 2.43 Bảng dữ liệu đăng ký ot

* **Bảng Đăng Ký Nghỉ Phép**

****

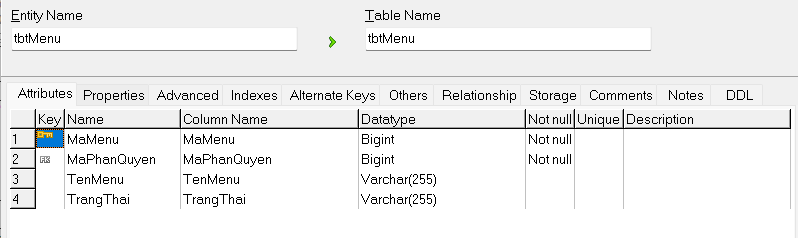
Hình 2.44 Bảng dữ liệu đăng ký nghỉ phép

* **Bảng Phân Quyền**

****

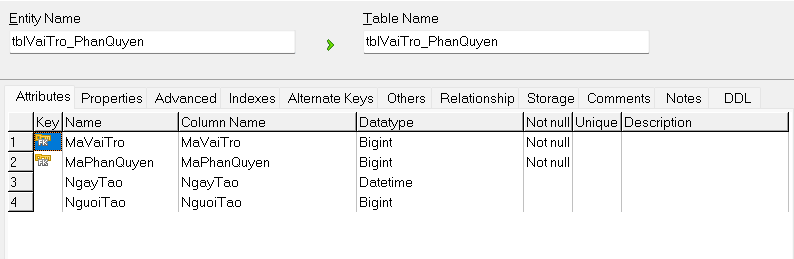
Hình 2.45 Bảng dữ liệu phân quyền

* **Bảng Menu**

****

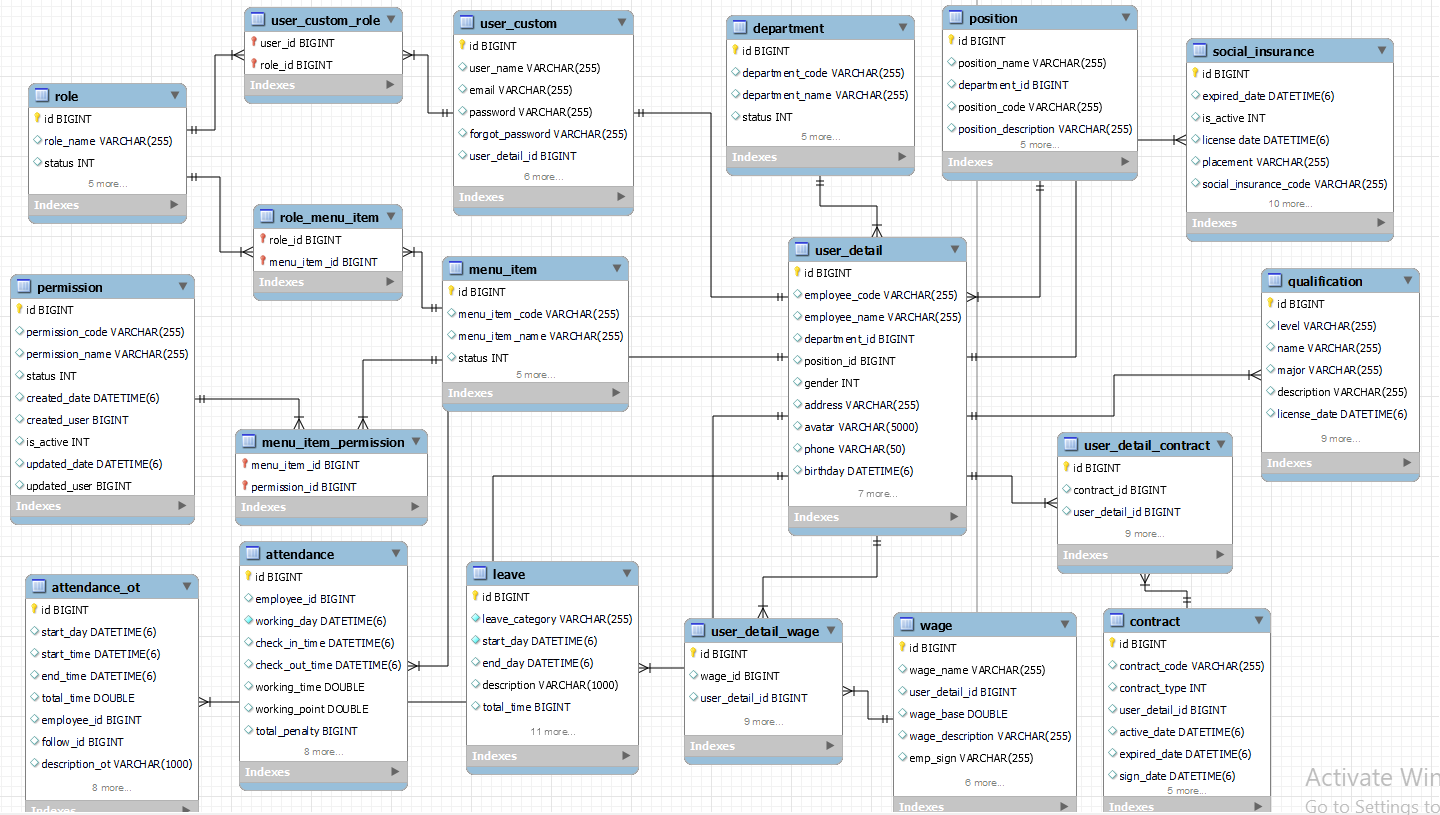
Hình 2.46 Bảng dữ liệu menu

* **Bảng Phân Quyền Với Vai Trò**

****

Hình 2.47 Bảng dữ liệu phân quyền với vai trò

### 2.4.3 Biểu đồ thực thể liên kết mức vật lý



Hình 2.48 Biểu đồ thực thể liên kết mức vật lý

## 2.5 Biểu đồ triển khai

Hệ thống được triển khai theo mô hình RESTful API, chúng ta sẽ mô tả các thành phần chính của hệ thống, bao gồm cấu hình máy chủ, ngôn ngữ lập trình, cơ sở dữ liệu, và các yêu cầu khác.Mô tả hệ thống:

1. Máy chủ

* CPU: Intel Xeon E5-2680 v4, 2.40GHz, 14 cores
* RAM: 32GB
* Hệ điều hành: Ubuntu Server 20.04 LTS
* Ổ cứng: 1TB SSD
* Mạng: Kết nối băng thông cao, tối thiểu 1Gbps
* Khả năng mở rộng: Hỗ trợ load balancing, có khả năng mở rộng theo chiều ngang (horizontal scaling)

1. Ngôn ngữ lập trình và Framework

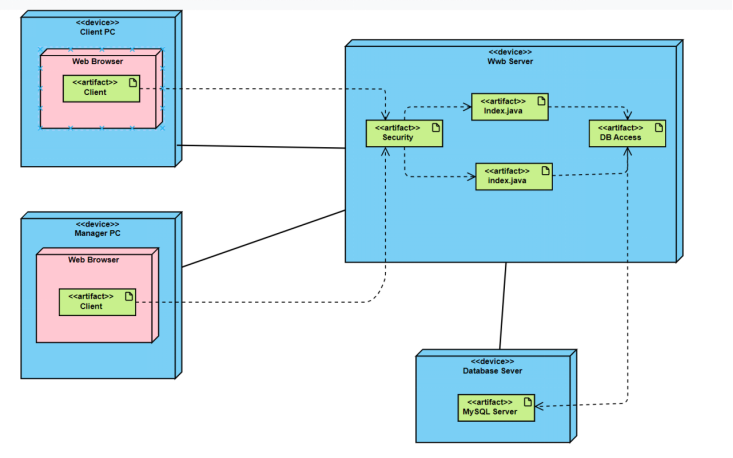
* Ngôn ngữ lập trình: Java
* Framework: Spring Boot cho API backend

1. Cơ sở dữ liệu

* Loại cơ sở dữ liệu: Relational Database
* Hệ quản trị cơ sở dữ liệu: MySQL
* Phiên bản: MySQL 8.0

1. Các yêu cầu khác

* Authentication: JSON Web Token (JWT)
* API Documentation: OpenAPI/Swagger
* Công cụ quản lý phiên bản: Git
* Continuous Integration/Continuous Deployment (CI/CD): GitHub Actions hoặc Jenkins
* Containerization: Docker
* Orchestration: Kubernetes



Hình 2.49 Biểu đồ triển khai

**Các yêu cầu phi chức năng:**

- Bảo mật: Sử dụng HTTPS để mã hóa dữ liệu truyền tải, JWT cho xác thực và phân quyền.

- Hiệu suất: Hệ thống phải đáp ứng các yêu cầu về tốc độ phản hồi và khả năng xử lý lượng truy cập lớn.

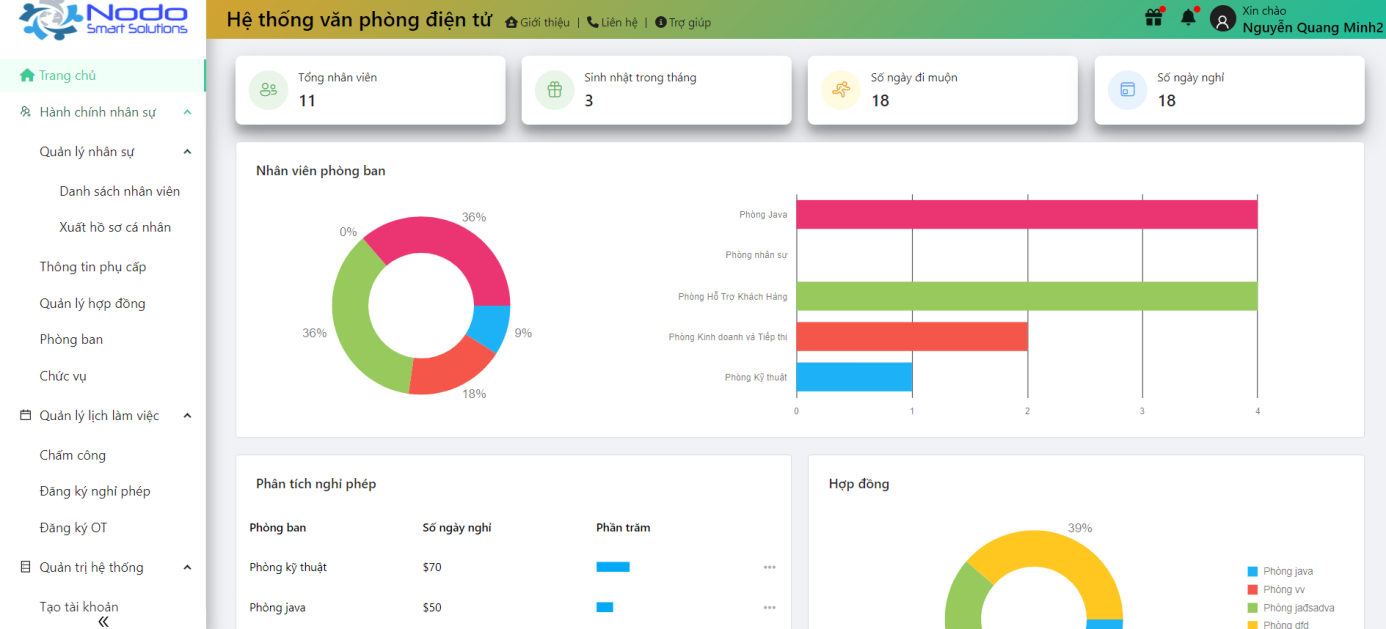
- Khả năng mở rộng: Hỗ trợ mở rộng dễ dàng cả theo chiều ngang (thêm máy chủ API) và chiều dọc (tăng tài nguyên cho máy chủ hiện có).

- Sao lưu và phục hồi: Thiết lập cơ chế sao lưu và phục hồi dữ liệu cho cơ sở dữ liệu MySQL.

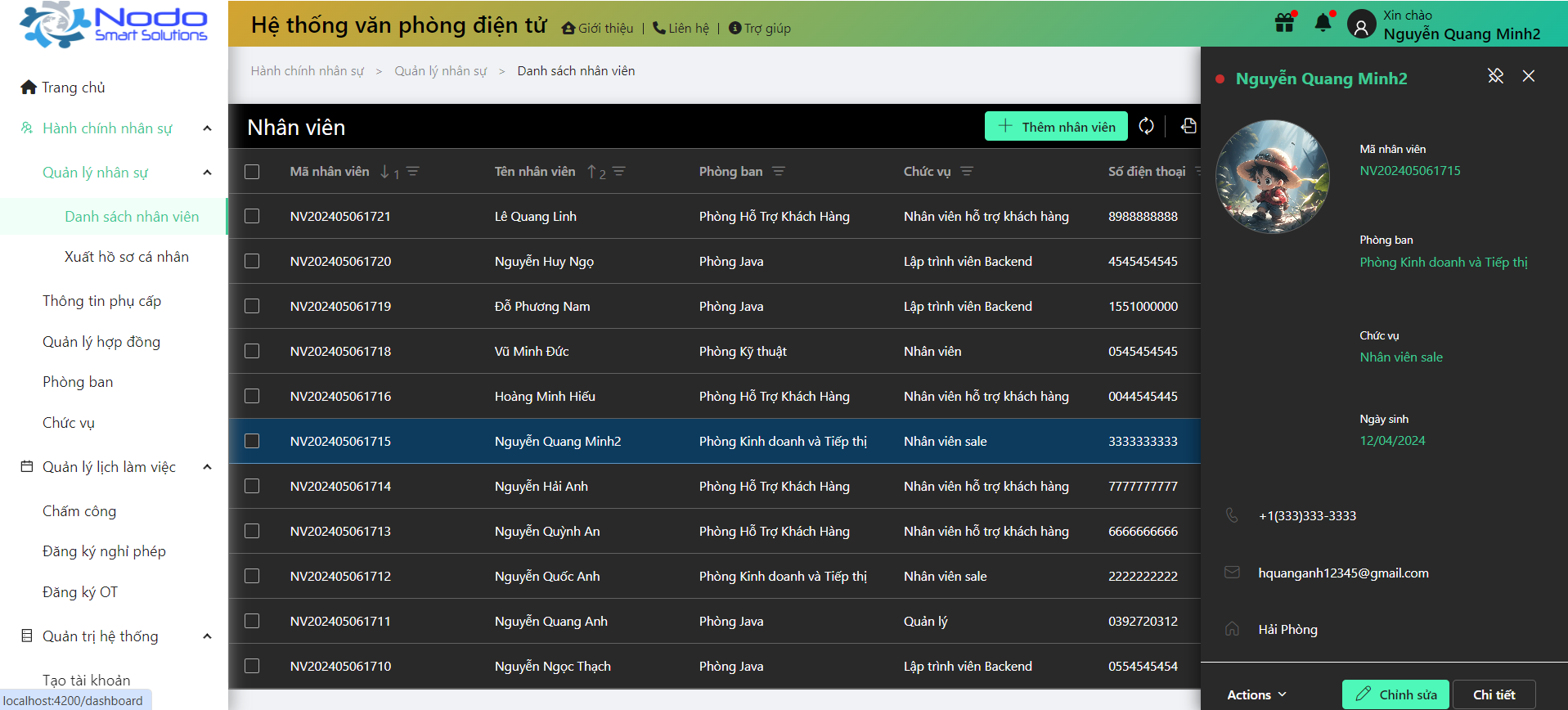
# CHƯƠNG 3: KẾT QUẢ VÀ KIỂM THỬ

Từ việc phân tích và thiết kế hệ thống trong chương 2, chương 3 sẽ đề cập đến kết quả thực hiện của trang web quản lý nhân sự. Chương 3 sẽ trình bày các chức năng cần có của trang web, được hiển thị trực tiếp trên màn hình. Đồng thời, cũng sẽ thực hiện kiểm thử phần mềm để đánh giá xem hệ thống hoạt động đúng theo yêu cầu đã được đặt ra hay không.

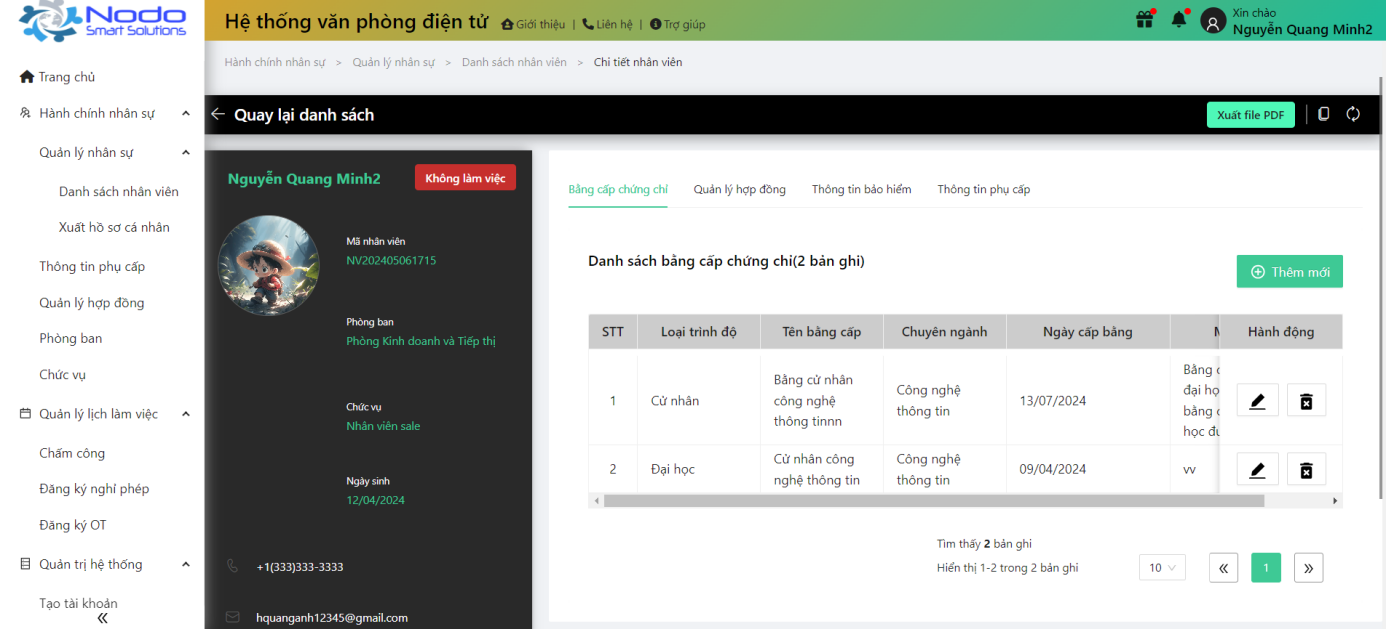
## 3.1. Một số kết quả đạt được



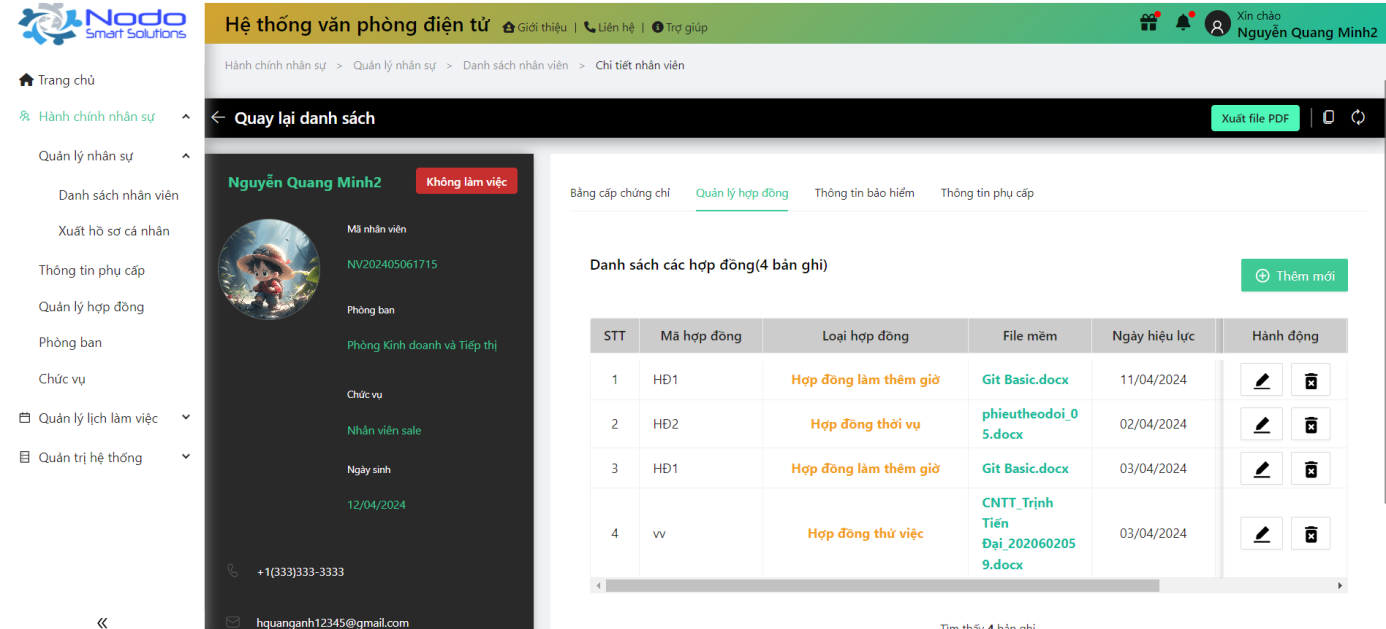
Hình 3.1 Giao diện trang chủ



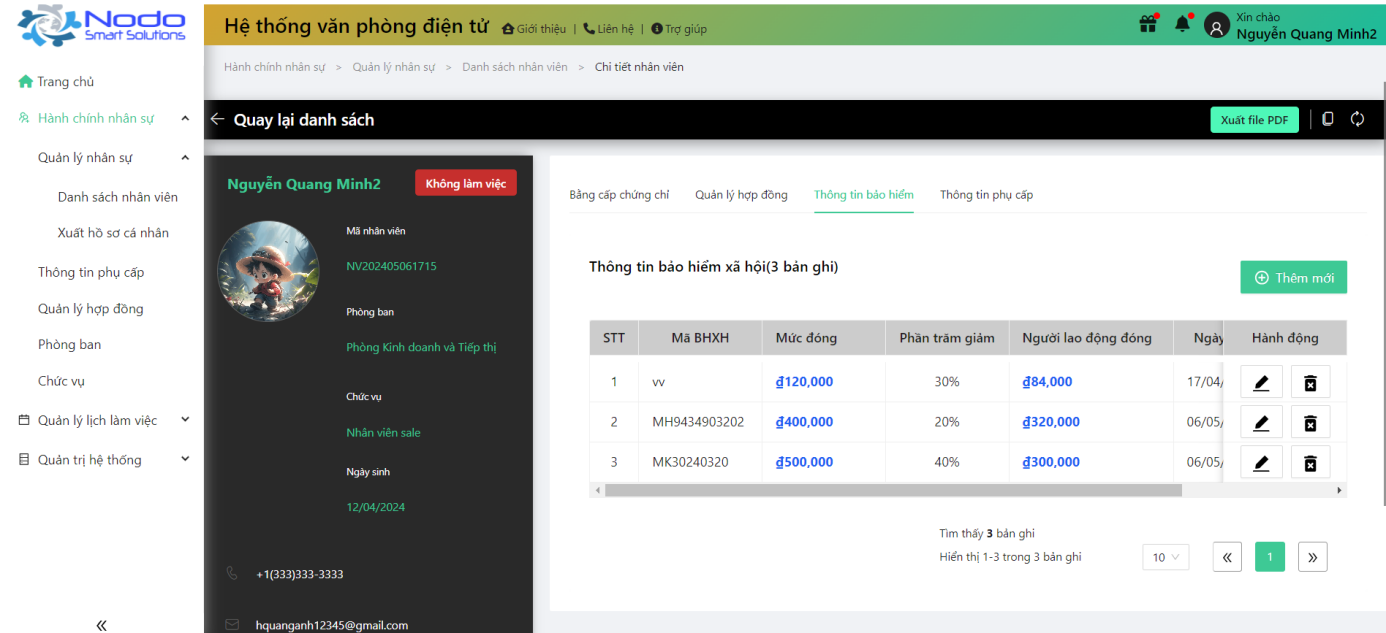
Hình 3.2 Giao diện danh sách nhân viên



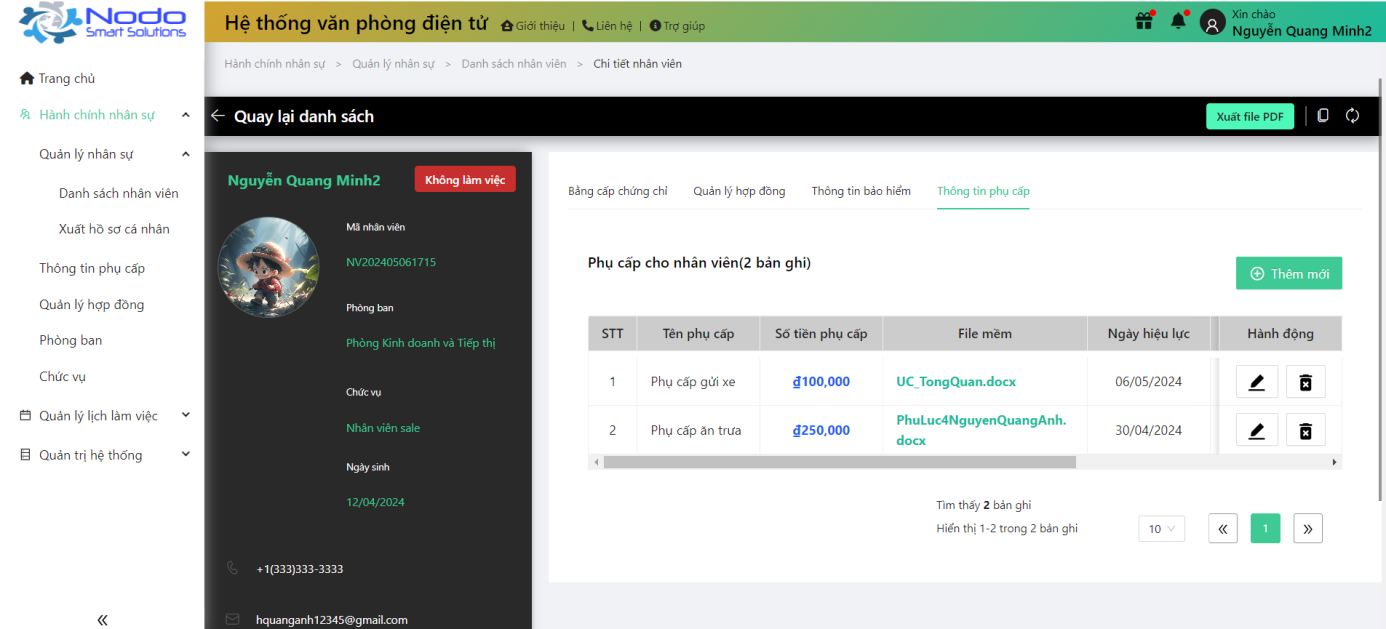
Hình 3.3 Giao diện quản lý bằng cấp chứng chỉ



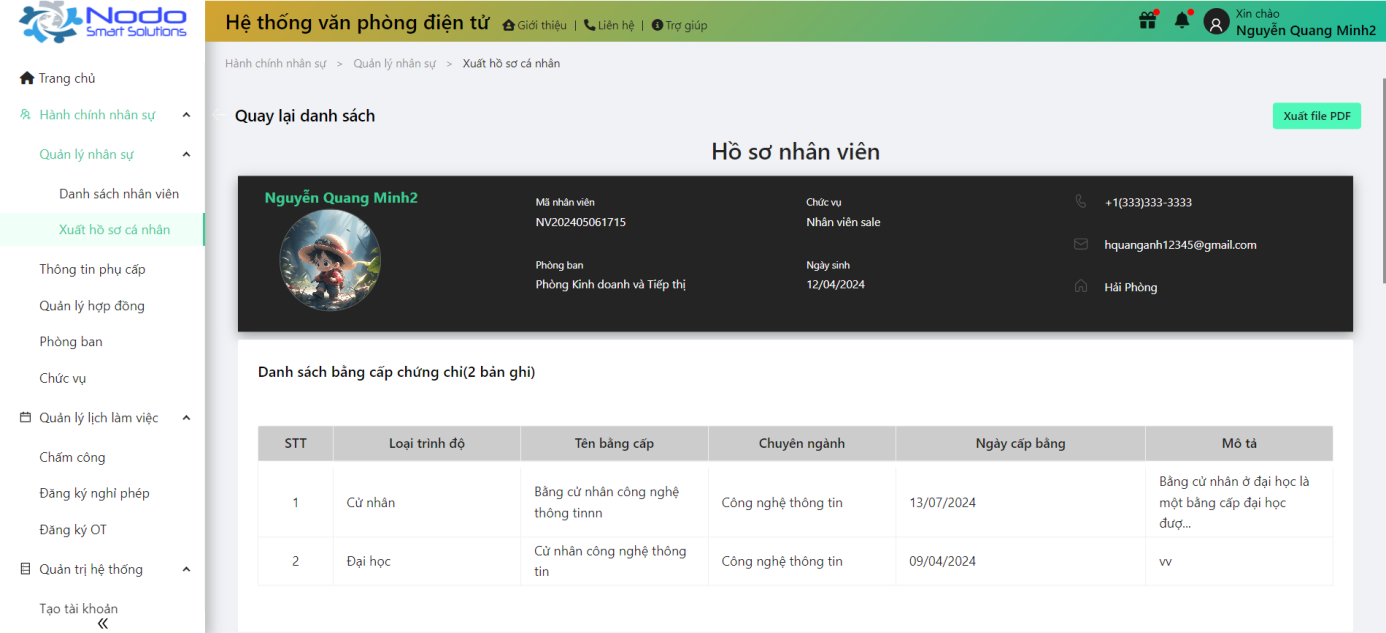
Hình 3.4 Giao diện quản lý hợp đồng của nhân viên



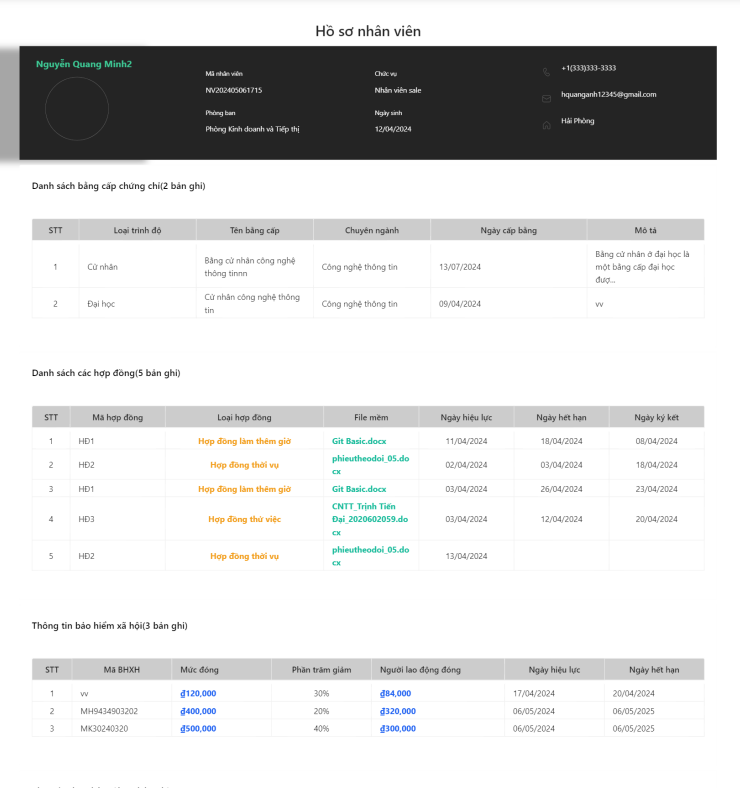
Hình 3.5 Giao diện quản lý thông tin bảo hiểm



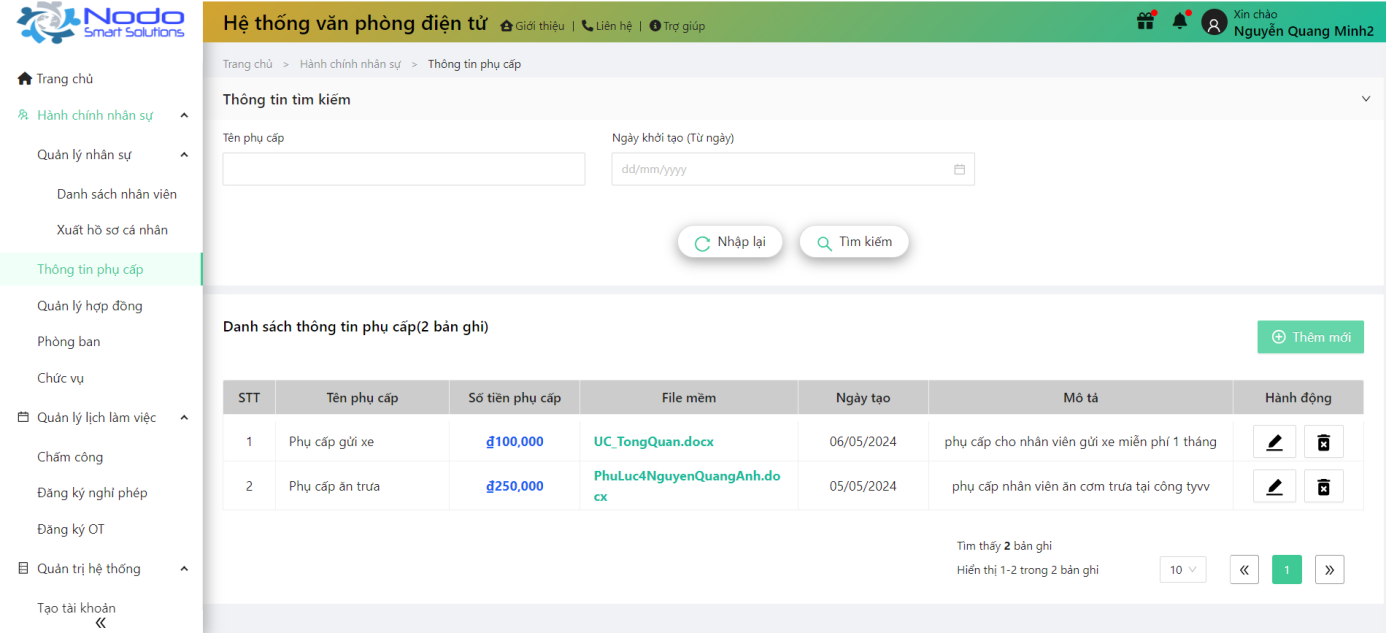
Hình 3.6 Giao diện quản lý thông tin phụ cấp



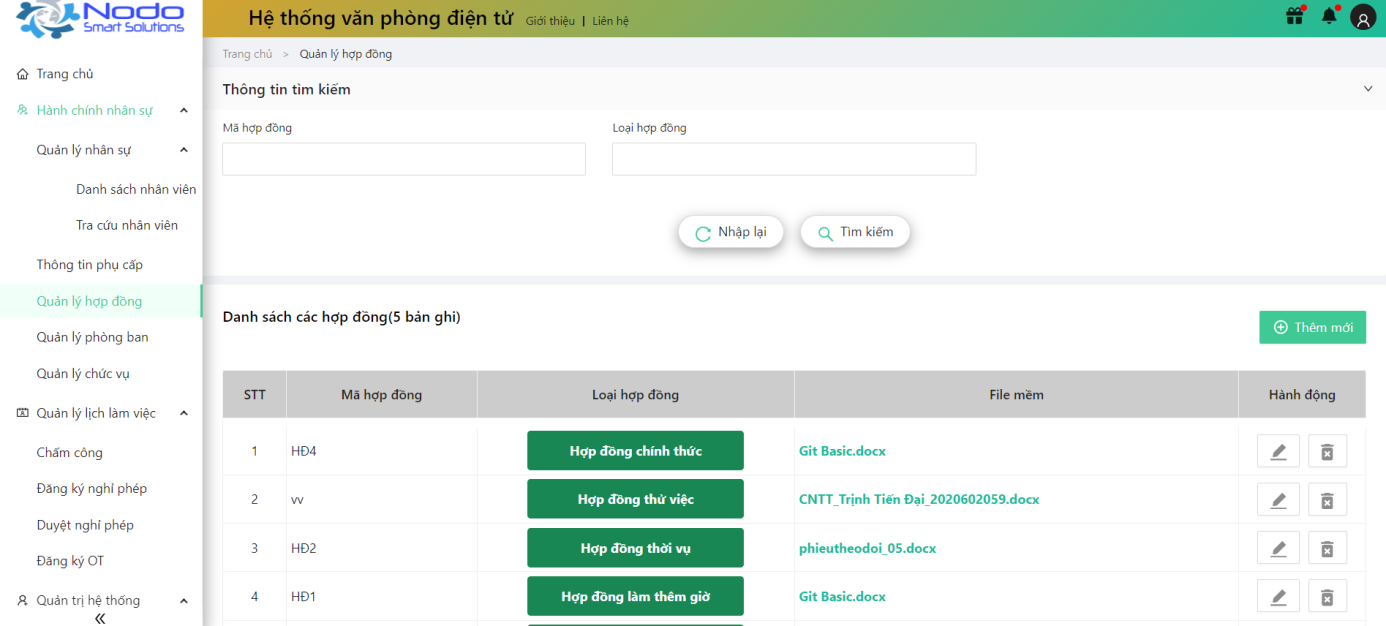
Hình 3.7 Giao diện xuất hồ sơ cá nhân



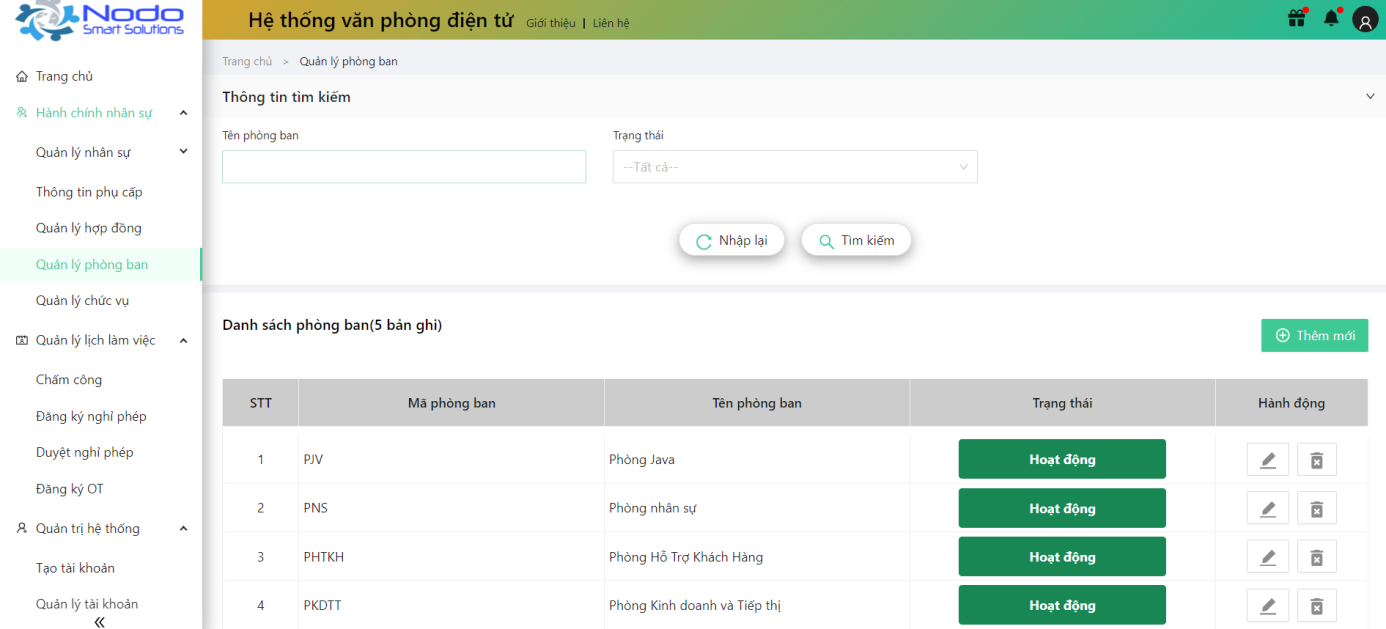
Hình 3.8 File PDF được xuất



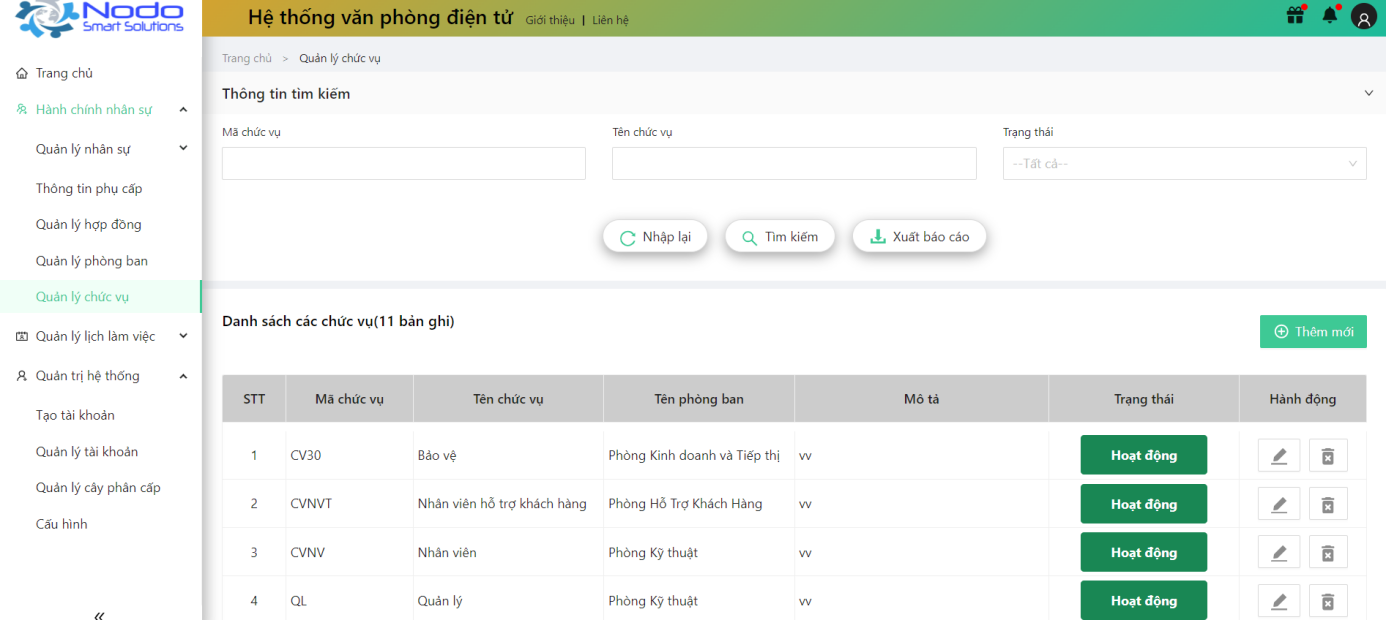
Hình 3.9 Giao diện quản lý thông tin phụ cấp công ty



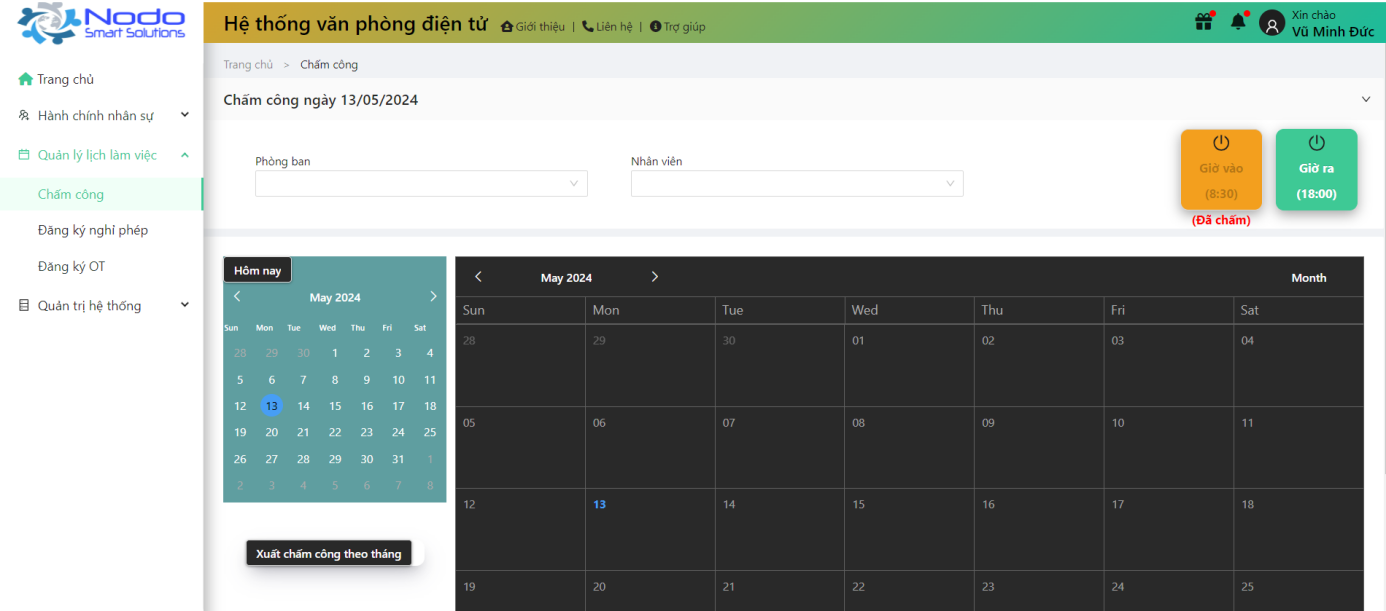
Hình 3.10 Giao diện quản lý hợp đồng công ty



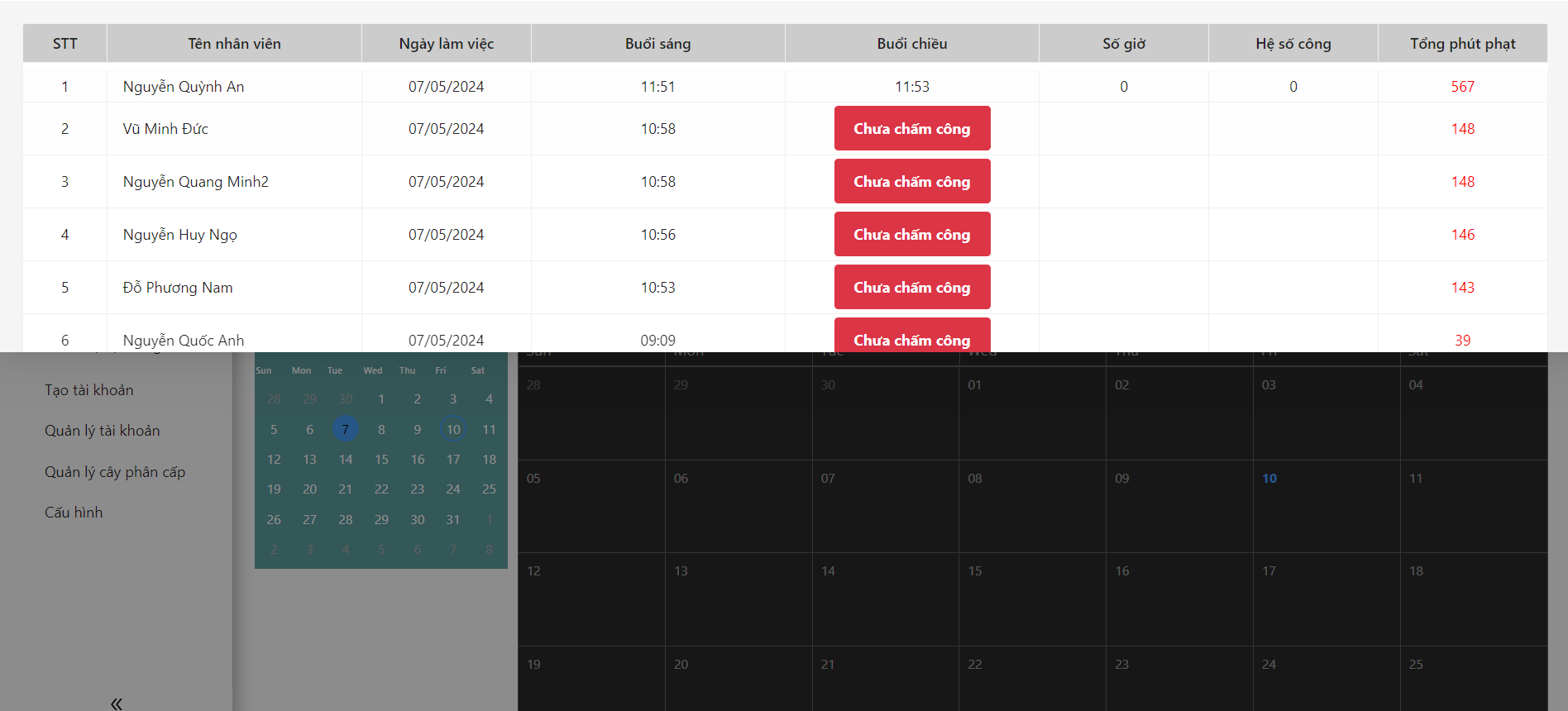
Hình 3.11 Giao diện quản lý phòng ban



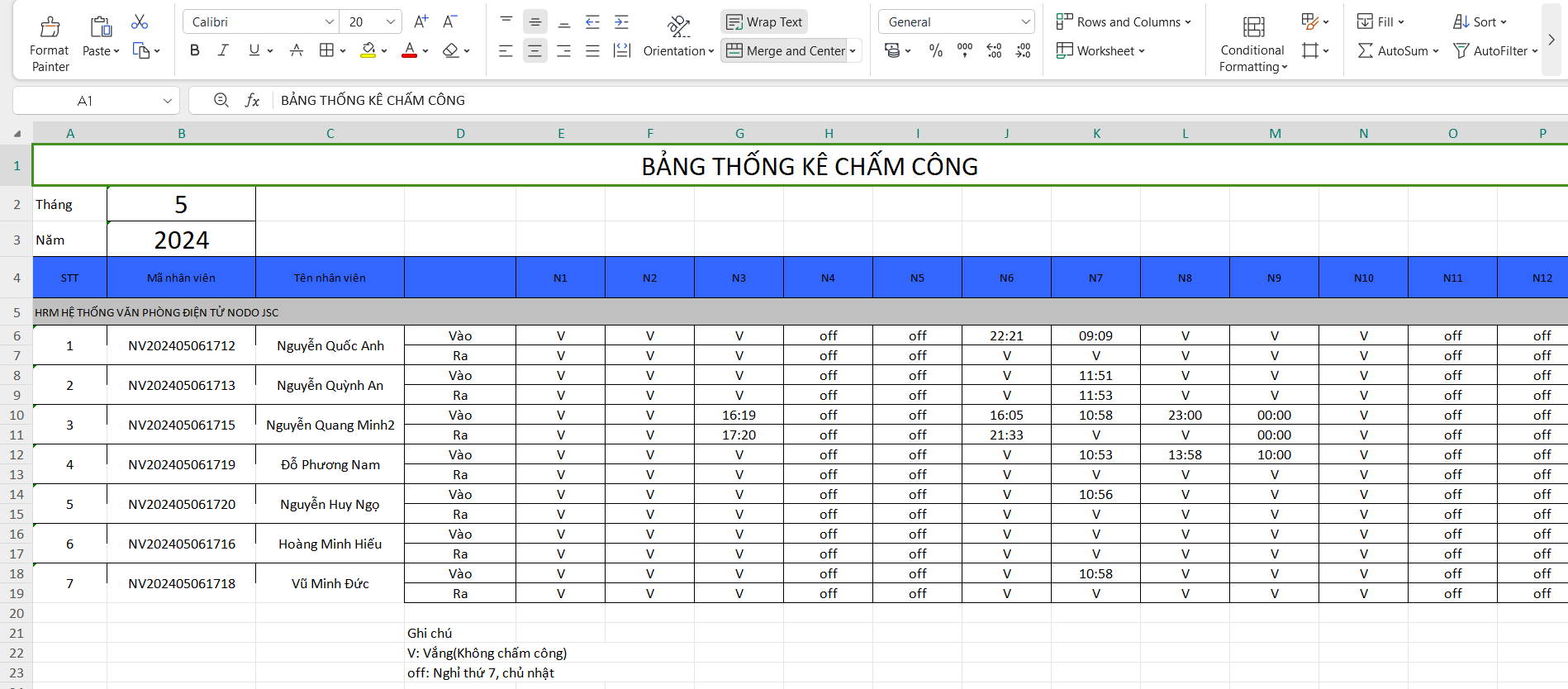
Hình 3.12 Giao diện quản lý chức vụ



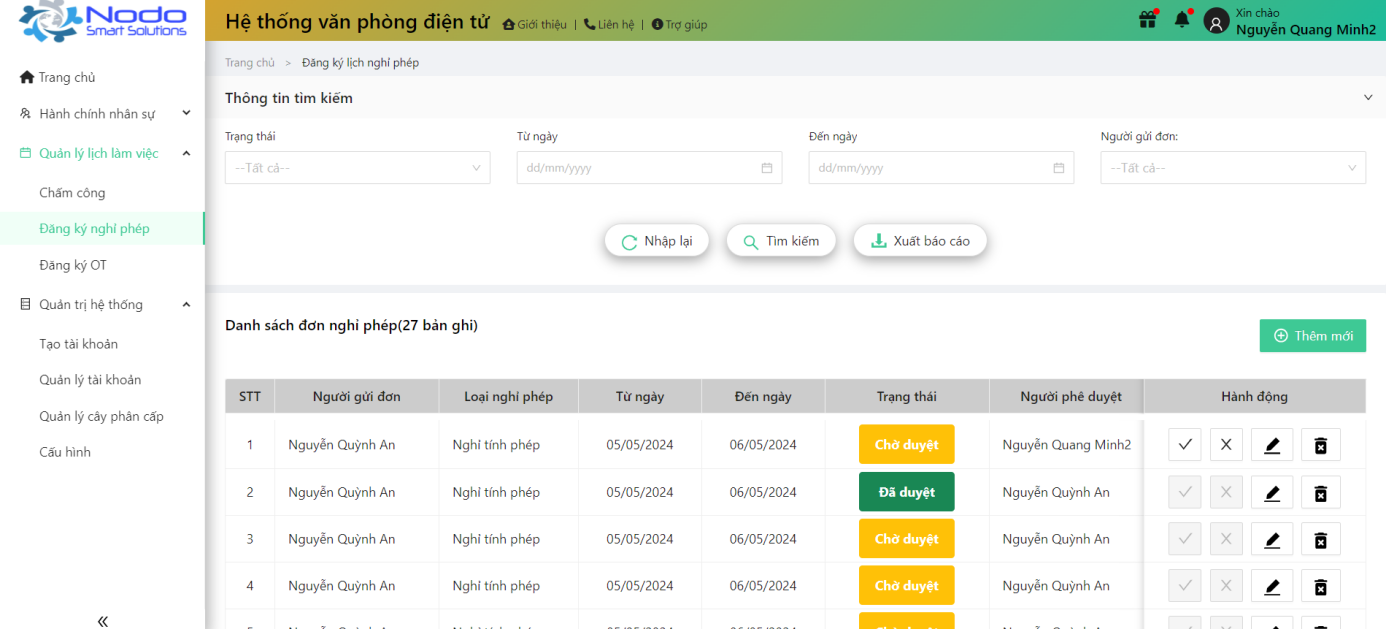
Hình 3.13 Giao diện quản lý chấm công



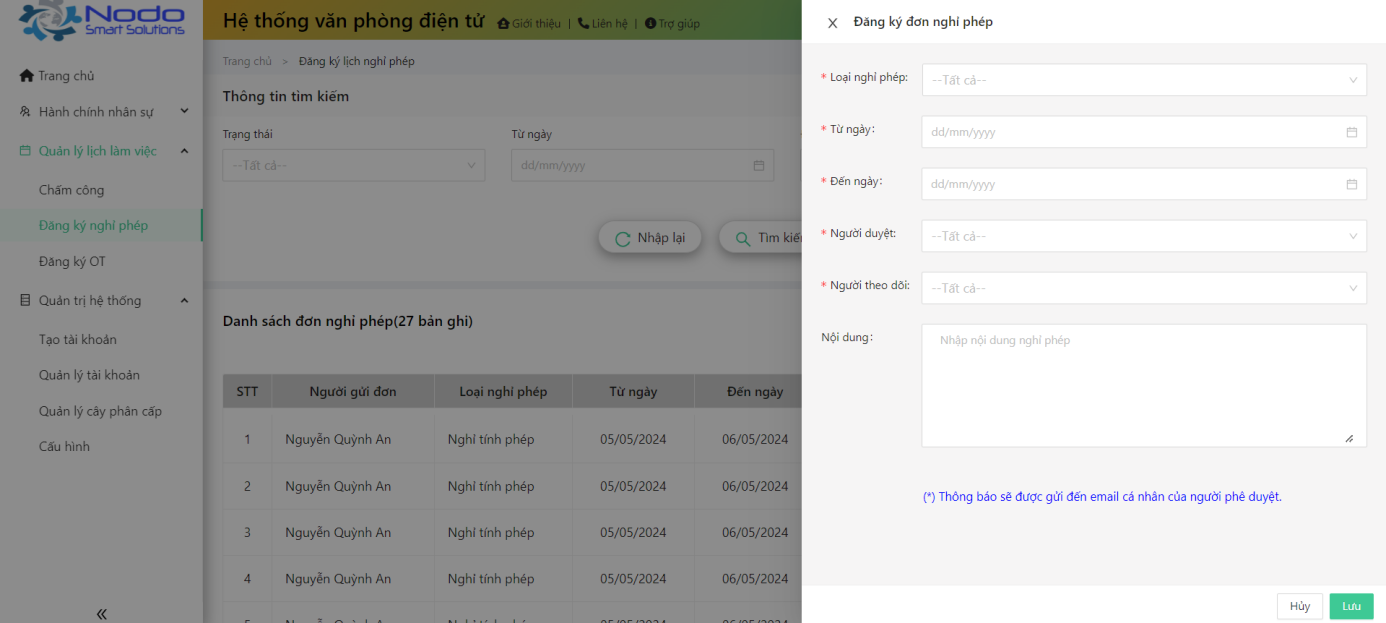
Hình 3.14 Giao diện danh sách chấm công



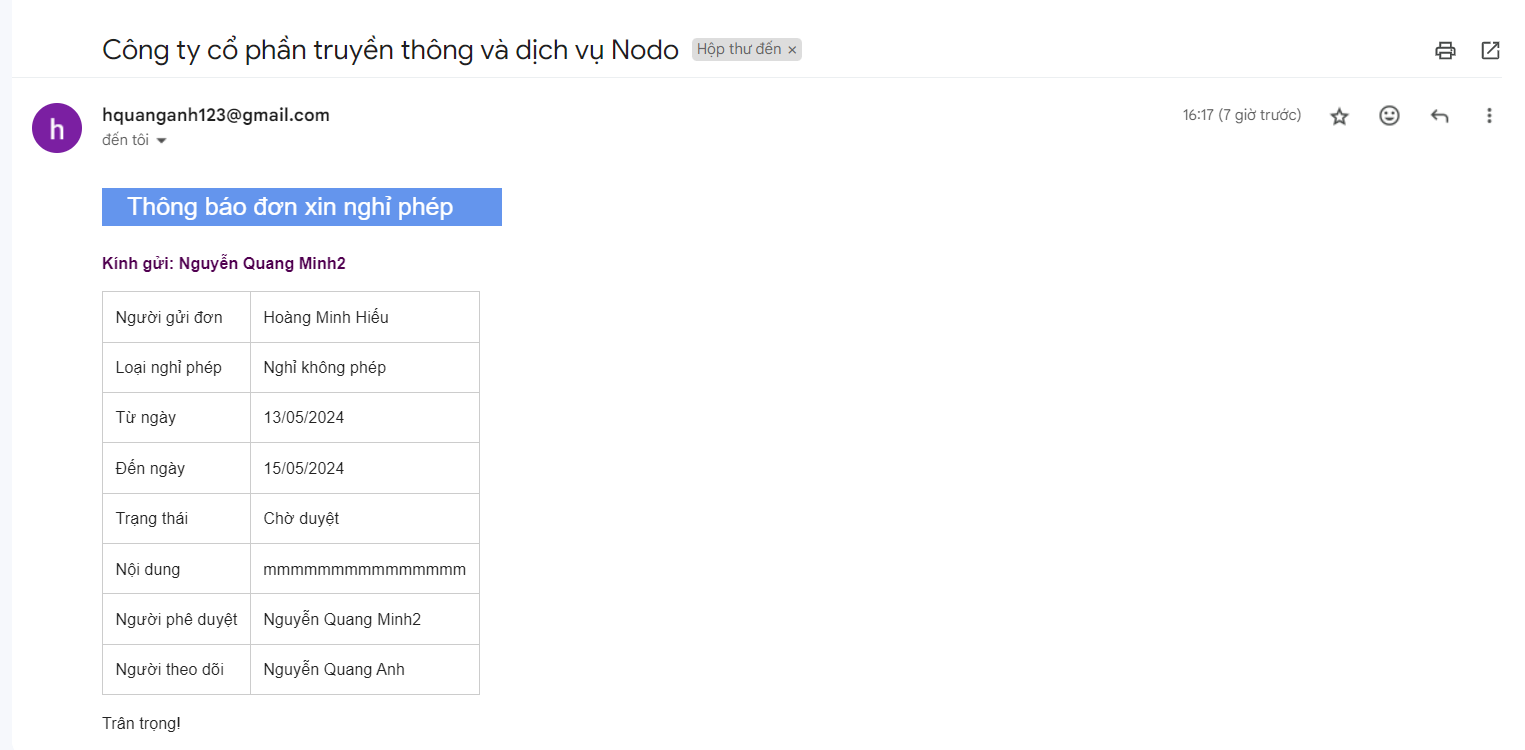
Hình 3.15 File excel xuất báo cáo thống kê chấm công



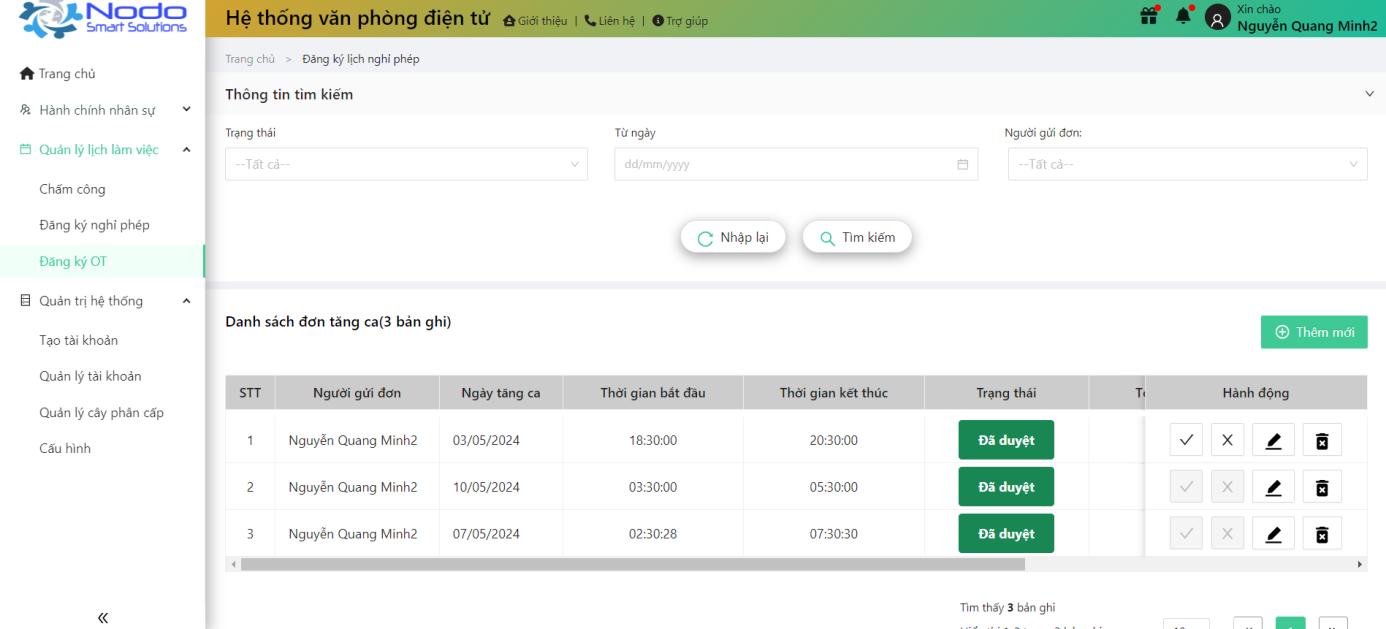
Hình 3.16 Giao diện đăng ký lịch nghỉ phép



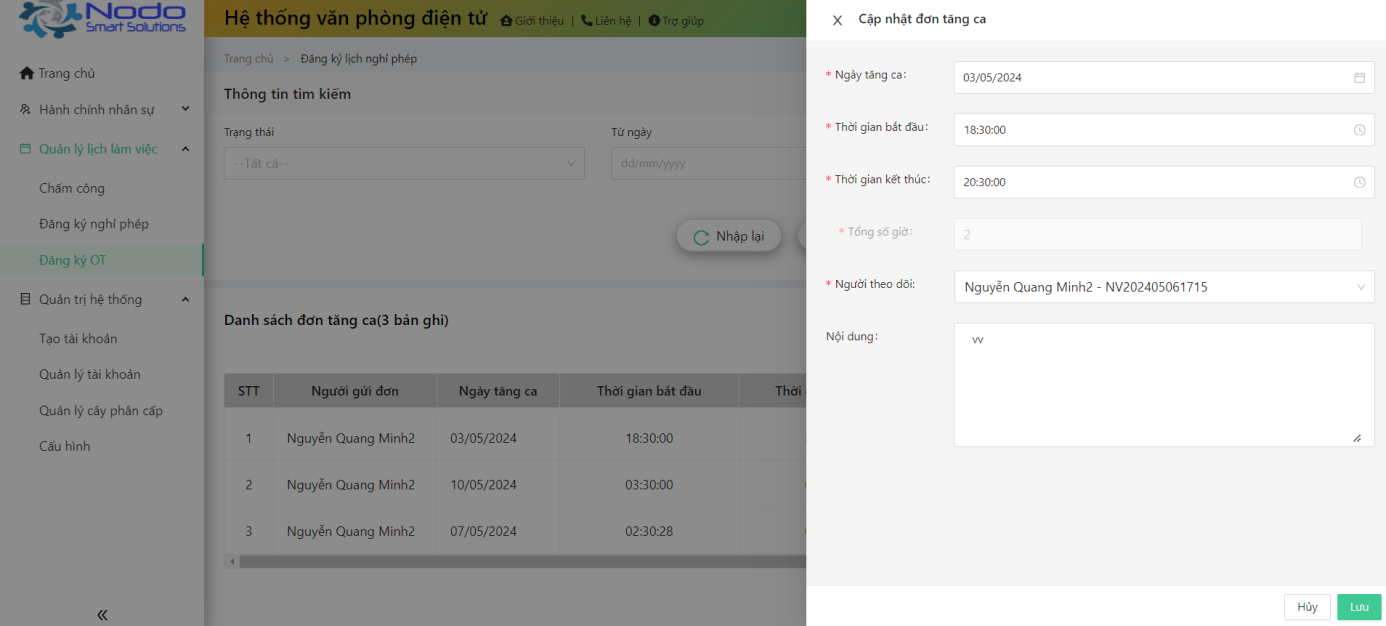
Hình 3.17 Giao diện trang tạo đơn đăng ký lịch nghỉ phép



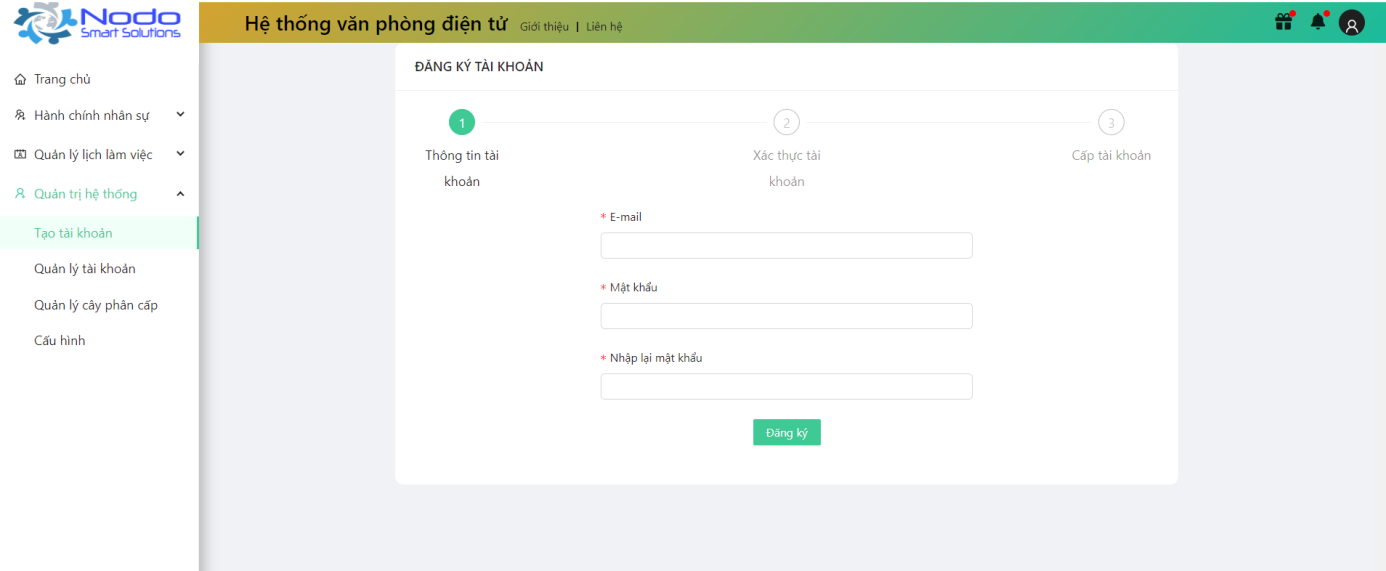
Hình 3.18 Mẫu phản hồi gmail đơn xin duyệt nghỉ phép



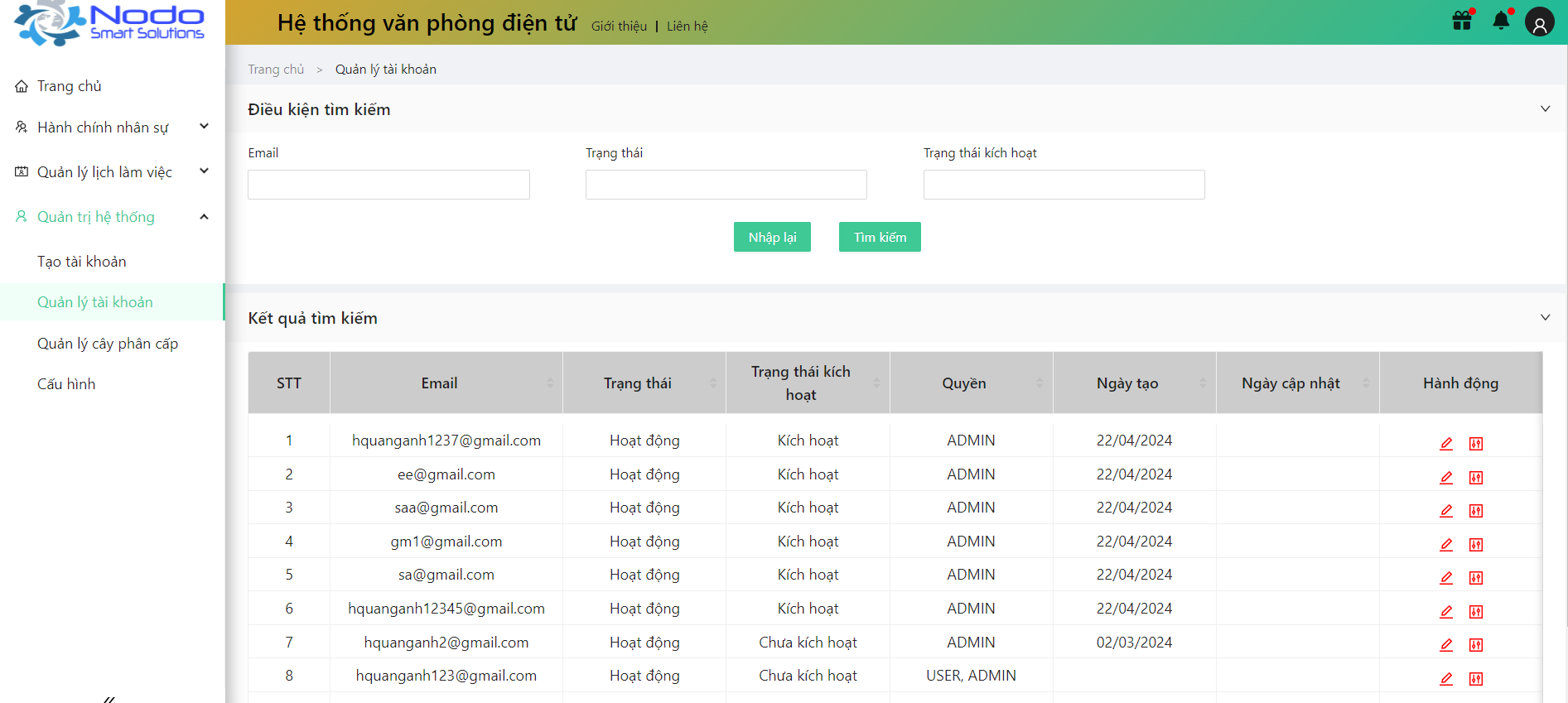
Hình 3.19 Giao diện đăng ký lịch OT



Hình 3.20 Giao diện trang tạo đơn đăng ký lịch OT



Hình 3.21 Giao diện tạo cấp tài khoản



Hình 3.22 Giao diện danh sách tài khoản

## 3.2 Kiểm thử

### 3.2.1. Kế hoạch kiểm thử

Những chức năng được kiểm thử:

- Đăng nhập: Người dùng đăng nhập vào hệ thống

- Tạo cấp phát tài khoản: Người dùng đăng ký tài khoản

- Quên mật khẩu: Người dùng quên mật khẩu

- Quản lý hồ sơ nhân viên: Quản lý có thể tra cứu, thêm, sửa, xóa hồ sơ nhân viên

- Chấm công: Nhân viên chấm công và theo dõi lịch chấm công.

- Quản lý hợp đồng: Quản lý có thể thêm sửa xóa hợp đồng của công ty.

- Quản lý phụ cấp: Quản lý có thể thêm sửa xóa thông tin phụ cấp của công ty.

- Đăng ký lịch nghỉ phép: Nhân viên có thể xin nghỉ phép bằng việc tạo đơn đăng ký lịch nghỉ phép.

- Đăng ký lịch OT: Nhân viên có thể đăng ký làm tăng ca bằng việc tạo đơn đăng ký lịch ot.

- Gán vai trò: Quản lý có thể gán chức năng cho từng vai trò trong hệ thống.

### 3.2.2. Kiểm thử chức năng

Bảng 3.1 Kiểm thử chức năng.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Trường hợp** | **Đầu vào** | **Đầu ra** | **Kết quả** |
| 1 | Kiểm thử chức năng đăng nhập | Nhập email và mật khẩu hợp lệ. | Đăng nhập thành công điều hướng về trang chủ. | Đạt |
| Nhập email và mật khẩu không hợp lệ. | Đăng nhập không thành công và có thông báo ra màn hình. | Đạt |
| 2 | Kiểm thử chức năng tạo cấp phát tài khoản | Nhập thông tin đăng ký email hợp lệ, xác nhận lại mật khẩu đã khớp. | Đăng ký thành công rồi điều hướng đến mục xác thực tài khoản. | Đạt |
| Nhập thông tin xác thực không đúng. | Xác thực thành công thì điều hướng đến mục cấp tài khoản. | Đạt |
| Nhập thông tin đăng ký đã tồn tại. | Đăng ký không thành công và hiển thị thông báo. | Đạt |
| Đồng ý cấp phát tài khoản. | Hiển thị thông báo thành công và điều hướng đến màn hình danh sách tài khoản. | Đạt |
| Không đồng ý cấp phát tài khoản. | Quay lại màn hình thông tin tài khoản. | Đạt |
| 3 | Kiểm thử chức năng quên mật khẩu | Nhập thông tin gmail vào màn hình quên mật khẩu. | Hiển thị thông báo thành công và gửi mail để nhập lại mật khẩu | Đạt |
| 4 | Kiểm thử chức năng quản lý tra cứu, thêm, sửa, xóa hồ sơ nhân viên | Nhập tên nhân viên cần tìm có tồn tại. | Hiển thị tất cả nhân viên có tên trùng với từ khóa vừa nhập | Đạt |
| Nhập từ khóa nhân viên cần tìm không tồn tại. | Không hiển thị nhân viên nào | Đạt |
| Nhập thông tin hợp lệ. | Hiển thị màn hình thông báo thành công. | Đạt |
| Nhập thông tin không hợp lệ. | Hiển thị màn hình thêm, sửa thất bại. | Đạt |
| Đồng ý xóa hồ sơ nhân viên. | Hiển thị thông báo xóa thành công. | Đạt |
| 5 | Kiểm thử chức năng chấm công | Nhấn nút chấm công giờ vào | Hiển thị thông báo đã chấm công và chặn ấn nút | Đạt |
| Nhấn nút chấm công giờ ra | Hiển thị thông báo đã chấm công thành công | Đạt |
| Chọn nhân viên, phòng ban cần tìm để tra cứu danh sách chấm công | Hiển thị danh sách tương ứng thành công | Đạt |
| Chọn tháng để xuất báo cáo chấm công theo tháng | Xuất về file excel báo cáo bảng chấm công tháng thành công | Đạt |
| 6 | Kiểm thử chức năng quản thêm, sửa hợp đồng | Nhập thông tin hợp lệ. | Hiển thị màn hình thông báo thành công. | Đạt |
| Nhập thông tin không hợp lệ. | Hiển thị màn hình thêm, sửa thất bại. | Đạt |
| 7 | Kiểm thử chức năng đăng ký lịch nghỉ phép | Nhập thông tin từ ngày nghỉ phép lớn hơn đến ngày nghỉ phép | Hiển thị thông báo lỗi ra màn hình | Đạt |
| Nhập thông tin từ ngày nghỉ phép nhỏ hơn đến ngày nghỉ phép | Không hiển thị thông báo lỗi trên màn hình | Đạt |
| Nhập thông tin thêm, sửa đơn xin nghỉ phép hợp lệ. | Hiển thị thông báo thêm, sửa thành công, gửi mail cho người phê duyệt và quay lại màn hình danh sách đơn nghỉ phép. | Đạt |
| 8 | Kiểm thử chức năng thêm, sửa phụ cấp | Nhập thông tin thêm, sửa phụ cấp hợp lệ. | Hiển thị thông báo thêm, sửa thành công và quay lại màn hình danh sách phụ cấp. | Đạt |
| Chọn đồng ý xóa phụ cấp. | Hiển thị thông báo xóa thành công và quay lại màn hình danh sách phụ cấp. | Đạt |
| 9 | Kiểm thử chức năng xóa phụ cấp | Đồng ý xóa phụ cấp. | Hiển thị thông báo thành công. | Đạt |
| Không đồng ý xóa phụ cấp. | Quay lại màn hình danh sách phụ cấp. | Đạt |
| 10 | Kiểm thử chức năng đăng ký đơn lịch ot | Nhập thông tin thêm, sửa đơn lịch ot hợp lệ. | Hiển thị thông báo thêm, sửa thành công và quay lại màn hình danh sách đăng ký lịch ot. | Đạt |
| Chọn đồng ý xóa đơn lịch ot. | Hiển thị thông báo xóa thành công và quay lại màn hình danh sách đơn lịch ot. | Đạt |
| 11 | Kiểm thử chức năng gán vai trò | Chọn vai trò và ấn lưu. | Hiển thị thông báo thành công và quay lại màn hình danh sách tài khoản | Đạt |

### 3.2.3. Kết quả kiểm thử

- Tỉ lệ test case đạt: 100%

- Tỉ lệ test case thất bại: 0%

- Hệ thống chạy ổn định trên các trình duyệt web khác nhau như Firefox, Google Chrome, Microsoft Edge.

# KẾT LUẬN

Đề tài “**Xây dựng website quản lý nhân sự cho Công ty cổ phần truyền thông và dịch vụ Nodo bằng Spring Framework**” xuất phát từ những nhu cầu quản lý nhân sự của công ty và mang lại tiềm năng phát triển trong tương lai. Qua quá trình thực hiện đề tài, em đã tìm hiểu, tích lũy và vận dụng được những kiến thức công nghệ được học từ trường, từ nơi làm việc và đã hoàn thành được những phần sau đây.

**Kết quả đạt được**

• Cho phép theo dõi, tra cứu thông tin hồ sơ nhân viên của công ty.

• Cung cấp một hệ thống chấm công tiện ích.

• Xây dựng một hệ thống quản lý nhân sự toàn diện và hiệu quả cho Công ty Nodo, ban quản lý nhân sự có thể quản lý thông tin hồ sơ nhân viên, quản lý chấm công, quản lý số ngày phép của nhân viên và báo cáo thống kê xếp loại chất lượng nhân viên.

**Hướng nghiên cứu trong tương lai**

• Tiếp tục hoàn thiện các tính năng mở rộng phần mềm nhằm cải tiến và nâng cấp chương trình.

• Hoàn thiện chương trình, sửa các lỗi và cải thiện tính bảo mật.

• Tích hợp Single Sign On như facebook, gmail…

• Tối ưu cách viết code và cơ sở dữ liệu.

• Xây dựng thêm quản lý tính lương tự động.

• Chấm công nhận diện bằng khuôn mặt

# TÀI LIỆU THAM KHẢO

**Tiếng việt**

[1] Hoàng Quang Huy, Phùng Đức Hòa, Trịnh Bá Quý, Nhập môn công nghệ phần mềm, NXB Đại học Công nghiệp Hà Nội.

[2] Nguyễn Thị Thanh Huyền, Ngô Thị Bích Thúy, Phạm Thị Kim Phượng, Giáo trình phân tích thiết kế hệ thống, NXB Giáo dục VN.

[3] Nguyễn Bá Nghiễn, Lập trình java, NXB thông tin và truyền thông.

[4] Giáo trình thiết kế web, Trường đại học Công Nghiệp Hà Nội.

[5] Trang tài liệu MySQL(truy cập lần cuối 07/05/2023) [https://www.mysql.com/](https://www.tutorialspoint.com/uml/index.htm)

[6] Trang tài liệu Java(truy cập lần cuối 13/05/2023)<https://www.w3schools.com/java/default.asp>

[7] Trang tài liệu Spring Framework(truy cập lần cuối 13/05/2023) <https://spring.io/projects/spring-boot>

[8] Trang tài liệu JavaScript(truy cập lần cuối 16/05/2023) [https://javascript.info](https://javascript.info/)

[9] Trang tin tức về lập trình(truy cập lần cuối 16/05/2023) [https://viblo.asia](https://viblo.asia/)

[10] Trang quản lý nhân sự của công ty cổ phần truyền thông và dịch vụ Nodo <http://noffice.nodo.vn/> (truy cập lần cuối 10/05/2023)

[11] Trang tài liệu Angular(truy cập lần cuối 14/05/2023) https://angular.io/docs

[12] Trang tài liệu thư viện Ant Design(truy cập lần cuối 14/05/2023) https://angular.io/docs