

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA CÔNG NGHỆ PHẦN MỀM

Lê Vinh Quang – 19522093

Trần Trung Tín - 19522351

ĐỒ ÁN MÔN HỌC

NHÓM N07

NGÔN NGỮ LẬP TRÌNH JAVA - SE330.M21.PMCL
PHẦN MỀM QUẢN LÝ NHÂN VIÊN VÀ TÍNH LƯƠNG

GIẢNG VIÊN GIẢNG DẠY

Nguyễn Hồng Thủy

TP. HỒ CHÍ MINH, 2022

LỜI CẢM ƠN

Trước tiên nhóm em xin được bày tỏ sự trân trọng và lòng biết ơn đối với cô giáo Nguyễn Hồng Thủy giảng viên Bộ môn Ngôn ngữ lập trình Java - Khoa Công nghệ phần mềm - Trường Đại học Công Nghệ Thông Tin. Trong suốt thời gian học và làm đồ án môn học, cô đã dành rất nhiều thời gian quý báu để tận tình chỉ bảo, hướng dẫn và đã góp ý, sửa chữa và hoàn thiện để đồ án môn học của nhóm em được hoàn thành tốt hơn.

Nhóm em xin chân thành cảm ơn các thầy cô trong trường Đại học Công Nghệ Thông Tin đã trang bị cho em những kiến thức cơ bản cần thiết để nhóm em có thể hoàn thành đồ án môn học.

Nhóm em xin gửi lời cảm đến bạn bè những người đã tạo điều kiện thuận lợi cho nhóm em, tận tình giúp đỡ chỉ bảo nhóm em những gì còn thiếu sót trong quá trình làm đồ án môn học.

Vì thời gian có hạn, kiến thức của bản thân còn hạn chế. Cho nên trong đồ án không tránh khỏi những thiếu sót, nhóm em rất mong nhận được sự đóng góp ý kiến của cô và các bạn để đồ án của nhóm em được hoàn thiện hơn.

Nhóm em xin chân thành cảm ơn

MỤC LỤC

Chương 1. GIỚI THIỆU CHUNG.....	8
1.1. Lý do chọn đề tài	8
1.2. Tổng quan về chức năng	10
Chương 2. CƠ SỞ LÝ THUYẾT	12
2.1. HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU SQL SERVER 2019	12
2.2. Tìm hiểu về Spring Boot	12
2.3. Tìm hiểu về Spring JPA	15
2.4. Tìm hiểu Spring MVC	17
2.5. Tìm hiểu Thymeleaf	19
Chương 3. XÂY DỰNG HỆ THỐNG	20
3.1. Xây dựng kiến trúc hệ thống	20
3.1.2. Sơ đồ use case.....	21
3.1.4. Cơ sở dữ liệu	23
3.2. Thiết kế giao diện và mô tả xử lý	29
Chương 4. KẾT LUẬN	35
4.1. Kết quả đạt được	35
4.2. Ưu điểm	36
4.3. Nhược điểm và giải pháp	36

DANH MỤC HÌNH

Hình 1.1. Hình ảnh bảng lương

Hình 2.1. Query store SQL SERVER 2019

Hình 2.2. Spring Boot Framework

Hình 2.3. Triển khai RoomRepository trong java

Hình 2.3. Triển khai RoomRepository trong java

Hình 2.4. Ví dụ Spring Data JPA tự sinh câu truy vấn dựa vào tên hàm

Hình 2.5. Ví dụ custom một câu query

Hình 2.6. Mô hình MVC

Hình 2.7. Lấy và hiển thị dữ liệu lên file HTML bằng Thymeleaf

Hình 3.1. Sơ đồ use case

Hình 3.2. Sơ đồ liên kết thực thể

Hình 3.3. Mô hình quan hệ

Hình 3.4. Giao diện menu chính

Hình 3.5. Màn hình giao diện hiển thị thông tin staff

Hình 3.6. Màn hình giao diện trang thêm mới nhân viên

Hình 3.7. Màn hình giao diện trang update thông tin nhân viên

Hình 3.8. Màn hình giao diện hiển thị thông tin room

Hình 3.9. Màn hình giao diện hiển thị insert new room

Hình 3.10. Màn hình giao diện hiển thị thông tin Vị trí công việc và thêm vị trí công việc mới

Hình 3.11. Màn hình hiển thị thông tin bảng công và thêm mới một bảng công

Hình 3.12. Màn hình hiển thị thông tin các bảng công được tìm bằng id của nhân viên

Hình 3.13. Màn hình hiển thị tất cả lương cơ bản và thêm mới lương cơ bản cho nhân viên trong công ty

Hình 3.14. Màn hình hiển thị giao diện tính lương cho nhân viên

Hình 3.15. Chi tiết lương của một nhân viên

Hình 3.16. Trang hiển thị khi không tìm thấy dữ liệu phù hợp để tính lương

DANH MỤC BẢNG

Bảng 3.1. Bảng danh sách Actor

Bảng 3.2. Bảng danh sách use case

Bảng 3.3. Bảng Degree

Bảng 3.4. Bảng Position

Bảng 3.5. Bảng Room

Bảng 3.6. Bảng Staff

Bảng 3.7. Bảng Timesheets

Bảng 3.8. Bảng Salary

Bảng 3.9. Bảng SalaryPayment

DANH MỤC TỪ VIẾT TẮT

JPA : Java persistence API

MVC : Model – View - Controller

CSDL : Cơ sở dữ liệu

RAD : Rapid Application Development

DAO : Data access object

POJO : Plain old java object

TÓM TẮT ĐỒ ÁN

Nhóm chúng em xây dựng một ứng dụng web bằng Spring Boot và các thư viện hỗ trợ theo mô hình Spring MVC giúp quản lý các nhân viên trong công ty gồm các chức năng cơ bản :

- Có thể thêm xóa sửa cập nhật thông tin của một nhân viên và các thông tin liên quan đến nhân viên đó như là thông tin lương, bảng chấm công, bảng đại học, bảng thanh toán lương, ...
- Quản lý được tất cả nhân viên và có thể tìm các nhân viên theo mỗi phòng trong công ty
- Cung cấp chức năng tính lương cho nhân viên, có thể tính cho một phòng, cho một nhân viên, hoặc là tất cả nhân viên trong công ty

Cụ thể hơn như sau :

Giả sử một người phụ trách quản lý nhân viên và tính tiền lương cho một công ty. Những công ty có hàng trăm, hàng ngàn người cùng làm việc thì việc cập nhật thông tin cũng như tính lương của các nhân viên đó sẽ rất khó khăn. Để giải quyết vấn đề đó sao cho việc quản lý và tính tiền lương sẽ được tiết kiệm thời gian nhất một cách có thể, nhóm em đã thực hiện phát triển “Phần mềm quản lý nhân viên và tính tiền lương” để giúp giải quyết những vấn đề nêu trên. Giúp cho người dùng quản lý dễ dàng, có thể dễ dàng cập nhật, sửa đổi thông tin của nhân viên, dễ dàng tính tiền lương cho nhân viên và tiết kiệm được thời gian hơn so với cách quản lý thủ công.

Chương 1. GIỚI THIỆU CHUNG

1.1. Lý do chọn đề tài

Hiện nay, công việc quản lý nhân sự và tiền lương của doanh nghiệp vẫn đang dùng phương pháp thủ công. Thông tin của nhân viên phải sử dụng nhiều loại hồ sơ giấy tờ để lưu trữ, do đó mỗi khi có sự thay đổi, tra cứu hay cập nhật thông tin nhân viên, người quản lý sẽ tốn rất nhiều thời gian và công sức, bên cạnh đó việc lưu trữ trên giấy còn gây ra một trở ngại nữa là sự an toàn về thông tin, dữ liệu. Chẳng hạn như thông tin lưu trữ trên giấy có thể bị thất lạc hay dữ liệu qua nhiều năm có thể bị phai mờ, điều đó ảnh hưởng rất lớn đến công việc quản lý trong doanh nghiệp. Công việc quản lý tiền lương cho nhân viên thì doanh nghiệp đang sử dụng bảng tính Excel để hỗ trợ quản lý tuy nhiên cũng không đem lại nhiều lợi ích cho người sử dụng. Bởi Excel chỉ là công cụ tính toán chính xác và nhanh chóng nhưng nó không có khả năng quản lý một hệ thống CSDL, không thể giúp nhà quản lý nắm toàn bộ thông tin đầy đủ về toàn bộ hệ thống.

Do đó nhóm em chọn đề tài này vì muốn tạo ra được một phần mềm để quản lý nhân viên và tính tiền lương cho họ một cách nhanh chóng và tiết kiệm được thời gian.

II. PHẦN EXCEL: (5 điểm)

Thí sinh đặt tên tập tin: sobaodanh.xls; lưu vào thư mục gốc ổ đĩa D:
(Ví dụ: thí sinh có số báo danh là 050105 thì đặt tên tập tin là: D:\ 050105.xls)

Bảng lương tháng 4 năm 2010

Stt	Họ tên	Chức vụ	Hệ số lương	Hệ số phụ cấp	Lương cơ bản:		Tiền lương	Tiền thuế thu nhập
					Tổng hệ số	650000		
1	Nguyễn Văn Tuấn	HT	4.5	0.3				
2	Lê Hữu Thọ	GV	3.66					
3	Nguyễn Loan Duyên	GV	3					
4	Lê Phước Thành	GV	2.87					
5	Lê Minh Hùng	PHT	3.33	0.2				
6	Trần Văn Tới	GV	4.8					

Yêu cầu:

- 1/ Nhập dữ liệu như bảng tính trên.
- 2/ Tính cột Tổng hệ số= Hệ số lương + Hệ số phụ cấp.
- 3/ Tính cột Tiền lương= Tổng hệ số x Lương cơ bản
- 4/ Tính cột Tiền thuế thu nhập; biết rằng:
 - Tiền lương >= 3000000 thì Tiền thuế thu nhập = 5% x Tiền lương
 - Tiền lương < 3000000 thì Tiền thuế thu nhập = 0
- 5/ Yêu cầu:
 - Sắp xếp bảng tính theo thứ tự giảm dần của cột Tiền lương
 - Lọc ra danh sách những người ở cột chức vụ là "GV".

Hình 1.1. Hình ảnh bảng lương

1.2. Tổng quan về chức năng

Chức năng quản lý nhân sự

1.2.1 Quản lý nhân viên

- a. Cập nhật hồ sơ thông tin nhân viên :
 - + Thêm mới, chỉnh sửa, xóa hồ sơ thông tin của một nhân viên
 - + Tìm kiếm một nhân viên
- b. Tìm kiếm hồ sơ thông tin nhân viên :
 - + Tìm kiếm hồ sơ của một nhân viên
 - + Tìm kiếm hồ sơ của các nhân viên chung một phòng ban

1.2.2 Quản lý bằng cấp

- a. Cập nhật thông tin bằng cấp của nhân viên:
 - + Thêm mới, chỉnh sửa, xóa thông tin bằng cấp của các nhân viên trong công ty

1.2.3 Quản lý phòng ban

- a. Cập nhật thông tin các phòng ban trong công ty:
 - + Thêm mới, chỉnh sửa, xóa thông tin phòng ban

1.2.4 Quản lý chức vụ

- a. Cập nhật thông tin các chức vụ hiện có trong công ty:

Chức năng Tính lương

1.2.1 Quản lý bản công

- a. Cập nhật thông tin bản công của các nhân viên trong công ty:
 - + Thêm mới, chỉnh sửa, xóa thông tin bản công

1.2.2 Quản lý lương cơ bản

- a. Cập nhật thông tin lương cơ bản của các nhân viên trong công ty:

- + Thêm mới, chỉnh sửa, xóa thông tin lương

1.2.3 Quản lý chi tiết lương

- Tính lương của 1 tháng trong năm dựa theo bảng công của nhân viên :
 - + Tính lương cho tất cả nhân viên trong công ty
 - + Tính lương cho một nhân viên
 - + Tính lương cho các nhân viên trong một phòng ban
- Lưu và xem lại thông tin chi tiết lương tháng của nhân viên

Bảng 1.2. Các nhóm thực thể và nhóm chức năng

Các thực thể	Các nhóm chức năng
Thông tin nhân viên	Quản lý nhân sự
Thông tin bằng cấp	
Thông tin phòng ban	
Thông tin chức vụ	
Thông tin bảng công	Tính lương
Thông tin lương cơ bản	
Thông tin bảng tính lương	

Chương 2. CƠ SỞ LÝ THUYẾT

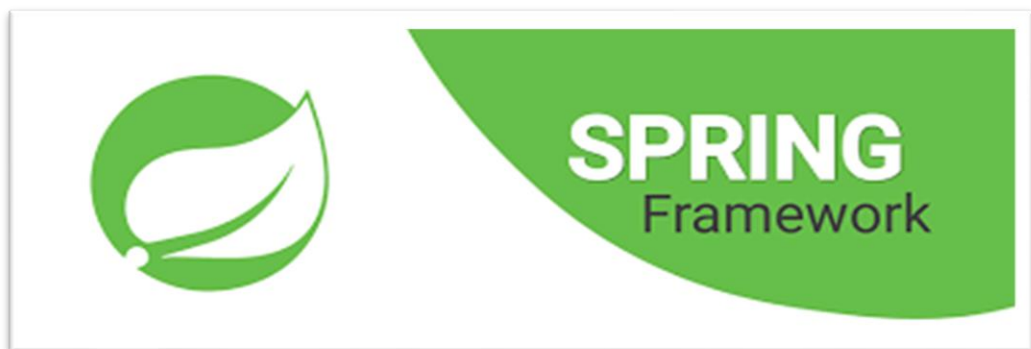
2.1. HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU SQL SERVER 2019

2.2. Tìm hiểu về Spring Boot

2.2.1. Giới thiệu

Spring Boot là một module của Spring Framework, cung cấp các tính năng RAD (Rapid Application Development) – phát triển ứng dụng nhanh, và có nhiều khả năng giúp cho lập trình viên giải quyết được nhiều vấn đề. So với framework Spring thông thường, Spring Boot tỏ ra những lợi thế vượt trội. Khi sử dụng Spring Boot, rất nhiều thứ được cải tiến hỗ trợ lập trình viên như:

- + Auto config: tự động cấu hình tất cả thay lập trình viên, chỉ cần viết code và tiến hành chạy hệ thống là được.
- + Tiện lợi hơn nhờ vào các Annotation
- + Server Tomcat có thể được nhúng ngay trong file JAR build ra và có thể chạy ở bất kì đâu mà java chạy được



Hình 2.2. Spring Boot Framework

2.2.2. Đặc điểm và tính năng nổi bật

- **SpringApplication:** Đây là class được dùng để khởi chạy ứng dụng từ hàm `main()`. Vì thế, khi khởi chạy ứng dụng, các lập trình viên chỉ cần gọi method `run()` là được.
- **Externalized Configuration:** Spring Boot cho phép người dùng có thể sử dụng cấu hình (config) từ bên ngoài. Chính vì vậy mà mỗi ứng dụng được tạo ra có thể chạy được trên nhiều môi trường khác nhau.
- **Profiles:** Dùng để phân chia các loại cấu hình cho các môi trường khác nhau.
- **Loggin:** Sử dụng phục vụ cho toàn bộ chức năng log trong phạm vi nội bộ. Những logging này sẽ được quản lý một cách mặc định.

Ngoài ra còn rất nhiều tính năng hữu ích khác của Spring Boot

2.3. TÌM HIỂU VỀ SPRING DATA JPA

2.3.1 JPA Là gì

JPA (Java Persistence API) là 1 giao diện lập trình ứng dụng Java, nó mô tả cách quản lý các mối quan hệ dữ liệu trong ứng dụng sử dụng Java Platform.

JPA cung cấp một mô hình POJO persistence cho phép ánh xạ các table/các mối quan hệ giữa các table trong database sang các class/mối quan hệ giữa các object.

Ví dụ: table Staff với các column (Id, name, age...) sẽ tương ứng với class Staff.java với các field Id, name, age... từ đó mỗi khi truy vấn table hay các column ta sẽ truy vấn trực tiếp trên các class,...

2.3.2 Hibernate là gì

Hibernate là 1 ORM (Object Relational Mapping) framework cho phép người lập trình thao tác với database một cách hoàn toàn tự nhiên thông qua các đối tượng. Lập trình viên hoàn toàn không cần quan tâm đến loại database sử dụng, SQL...

Hay nói cách khác, Hibernate chính là cài đặt của JPA (JPA là 1 tập các interface, còn Hibernate implements các interface ấy 1 cách chi tiết). Ngoài ra còn một số framework khác cũng implements JPA như là : Open JPA, Eclipselink,...

2.3.3 Môi quan hệ giữa Spring Data JPA, JPA và Hibernate

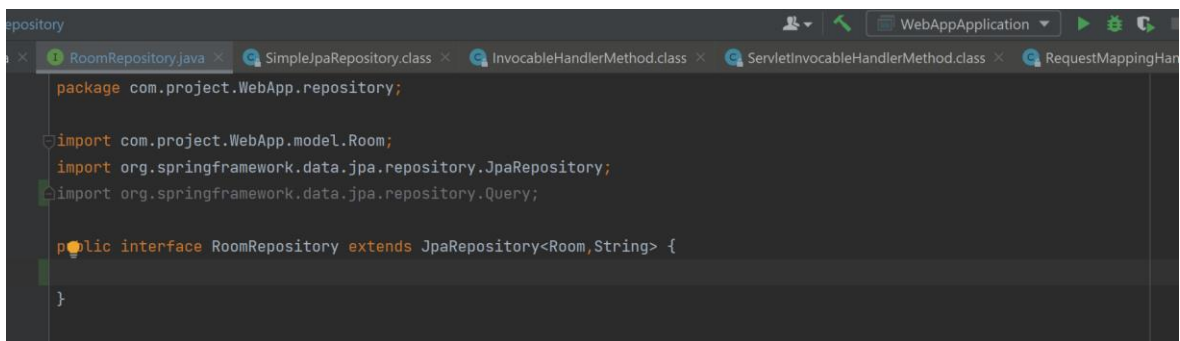
JPA là một đặc tả định nghĩa một API cho các object-relational mapping và quản lý các persistent object. Hibernate là JPA implements được sử dụng phổ biến trong các ứng dụng Java.

Và Spring Data JPA là được tầng được xây dựng phía trên tầng JPA. Có nghĩa là nó sẽ sử dụng tất cả các chức năng của JPA như entity mapping, entity life-lifecycle, JPA query,.. để xây dựng một số tính năng như no-code repository, khởi tạo query thông qua tên hàm, ...

2.3.4 Các Tính Năng Hỗ Trợ Nổi Bật Của Spring Data JPA

No-code Repository

Đối với mỗi repository chúng ta đều cần phải triển khai các hàm CRUD, việc này khiến mã nguồn bị lặp đi lặp lại nhiều lần. Spring Data JPA cung cấp một tập hợp các repository interface, chỉ cần extends chúng thì mặc định nó đã cung cấp sẵn các hàm CRUD phổ biến và các nhiều tính năng rất hữu ích.



Hình 2.3. Triển khai RoomRepository trong java sử dụng Spring data JPA

Ví dụ : Khi triển khai một RoomRepository như trên thì Spring Data JPA đã cung cấp sẵn một số các hàm cần thiết như :

- + Thêm, xóa, một Room
- + Tìm kiếm Room theo khóa chính
- + Tìm kiếm tất cả Room hiện có
- + Kiểm tra Room có tồn tại hay không dựa vào khóa chính
- + Tìm kiếm theo bất kì trường nào của Room bằng cách custom tên hàm ,...

VD : Ví dụ như hàm `findByStaffID`, Spring Data JPA sẽ sinh ra câu truy vấn tìm kiếm Room entity theo StaffID một cách tự động, đây giá trị StaffID từ tham số đầu vào câu truy vấn.

```
Iterable<Room> findByStaffID(String staffID);
```

Hình 2.4. Ví dụ Spring Data JPA tự sinh câu truy vấn dựa vào tên hàm

2.3.5 Spring Data annotation @Query

Khi mà các hàm có sẵn do Spring Data JPA cung cấp không thể đáp ứng được yêu cầu thì chúng ta sẽ sử dụng annotation `@Query` để khai báo một câu query sql cho method trong repository

Để sử dụng annotation `@Query`, ta cần khai báo trước một method trong interface extends từ `JpaRepository`, `CrudRepository` , và truyền câu sql vào bên trong

VD : Tạo câu query để tìm tất cả Timesheets theo staffID và month tương ứng

```
@Repository
public interface TimesheetsRepository extends CrudRepository<Timesheets,String> {
    Iterable<Timesheets> findByStaffID(String staffID);
    @Query(value = "select top(1) with ties * from Timesheets where staffID = :staffID and mon = :mon order by yea DESC",
        nativeQuery = true)
    Optional<Timesheets> findByStaffIDAndMon(@Param("staffID") String staffID,
                                              @Param("mon") int mon);
}
```

Hình 2.5. Ví dụ custom một câu query

2.3.6 Sử dụng Spring Data JPA trong Spring Boot

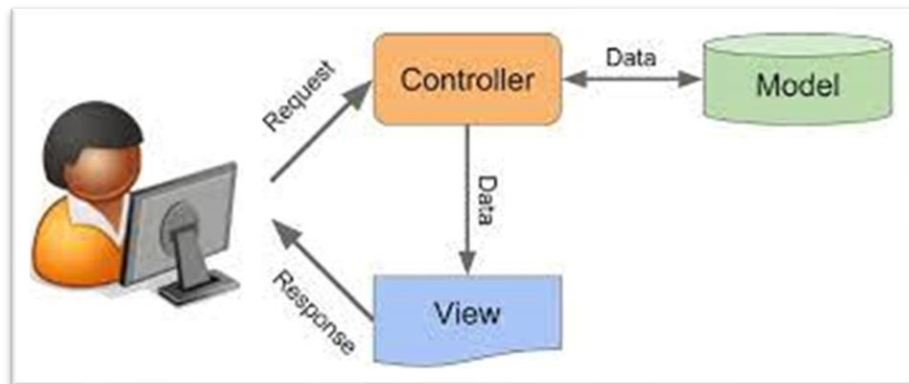
Để sử dụng Spring Data JPA trong Spring boot chúng ta cần thêm spring-boot-starter-data-jpa dependency và jdbc driver tương ứng với database đang sử dụng tương ứng.

2.4. TÌM HIỂU SPRING MVC

2.4.1 Giới Thiệu

Spring MVC là một module của Spring Framework tuân theo mô hình Model-View-Controller hoặc MVC, được thiết kế để xây dựng các nền tảng Web

Cách hoạt động



Hình 2.6. Mô hình MVC

Model: là các file POJO, Service, DAO thực hiện truy cập database, xử lý business

View: là các file JSP, html...

Controller: là Dispatcher Controller, Handler Mapping, Controller – thực hiện điều hướng các request.

Controller sẽ tiếp nhận các request phía user , xử lý request xem đó là url request hay sự kiện, sau đó controller xử lý input của user xem nội dung của request là gì

rồi giao tiếp với Model trong MVC. Model sẽ chuẩn bị các data cần thiết và gửi đi lại đến controller. Cuối cùng khi xử lý xong yêu cầu thì controller gửi dữ liệu trở lại View và hiển thị lên cho User trên trình duyệt

2.5. TÌM HIỂU THYMELEAF

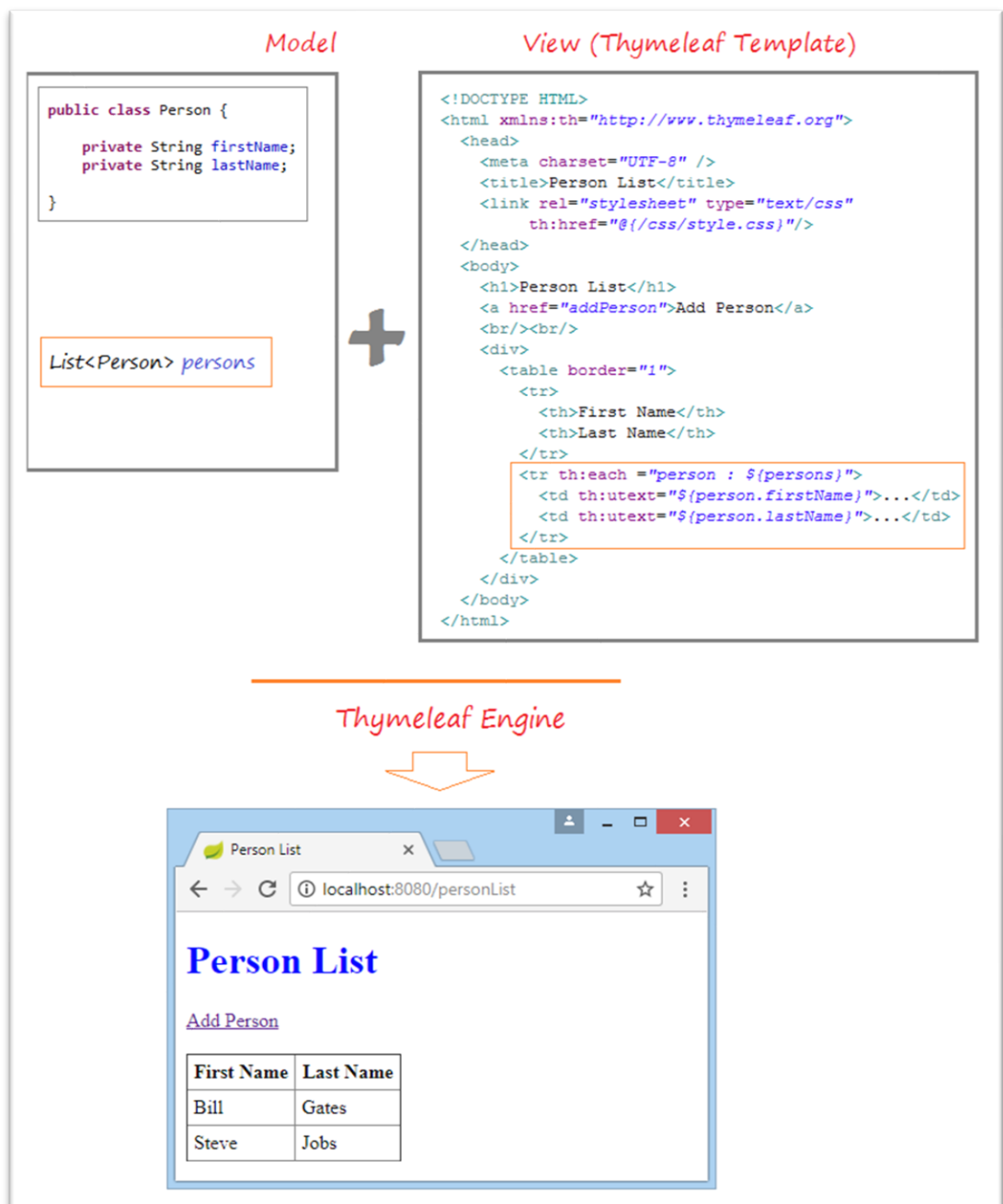
Trong các dự án thực tế của Java, việc lựa chọn các view phù hợp với các dự án là hoàn toàn cần thiết và quan trọng. Một trong những view hay sử dụng trong Java như jsp, jsf, thymeleaf. Một trong số đó thymeleaf được Spring Framework khuyến cáo nên sử dụng kết hợp với ứng dụng khi sử dụng Spring-Boot.

2.5.1. Giới Thiệu Thymeleaf

Thymeleaf là một Java template engine phía máy chủ, nó có thể làm việc với cả hai môi trường, web và non-web, nó có khả năng xử lý HTML, XML, JavaScript, CSS và thậm chí cả văn bản thuần túy.

Mục tiêu chính của Thymeleaf là mang lại các template tự nhiên, đồng nhất, đơn giản (nature templates) và dễ dàng bảo trì. Để đạt được điều này, Thymeleaf xây dựng trên khái niệm về các mẫu tự nhiên (natural templates) để chèn logic của nó vào các file template mà không ảnh hưởng đến template. Điều này giúp thu hẹp khoảng cách giữa các nhóm thiết kế và phát triển.

Ngay từ đầu, Thymeleaf đã được thiết kế với tiêu chuẩn Web – đặc biệt là HTML5 – cho phép bạn tạo các mẫu xác nhận (validating templates) đầy đủ.



Hình 2.7. Lấy và hiển thị dữ liệu lên file HTML bằng Thymeleaf

(<https://openplanning.net/12343/gioi-thieu-ve-thymeleaf>)

2.4.1 Lợi Ích Khi Sử Dụng Thymeleaf

a. Dễ dàng học

Với Thymeleaf, trang chủ của nó đã viết sẵn một documentation Thymeleaf từ a- z giúp các developer có thể dễ dàng học nó.

b. Thân thiện và gần gũi với các developer

Vì nó có thể sử dụng HTML làm view, với html thì bất kỳ các lập trình viên nào cũng biết, và tất nhiên ta chỉ sử dụng các file HTML làm view, thymeleaf sẽ tham gia vào file HTML dưới dạng các thuộc tính của các thẻ HTML. Ta không cần phải thêm bất cứ loại thẻ non-HTML nào cả. (Ví dụ như jsp thì muốn sử dụng các thẻ chuẩn của jsp như JSTL thì cần có taglib, với thymeleaf thì không cần sử dụng bất kỳ loại thẻ nào khác html nên được gọi là NON-HTML)

c. Không cần phải restart lại web

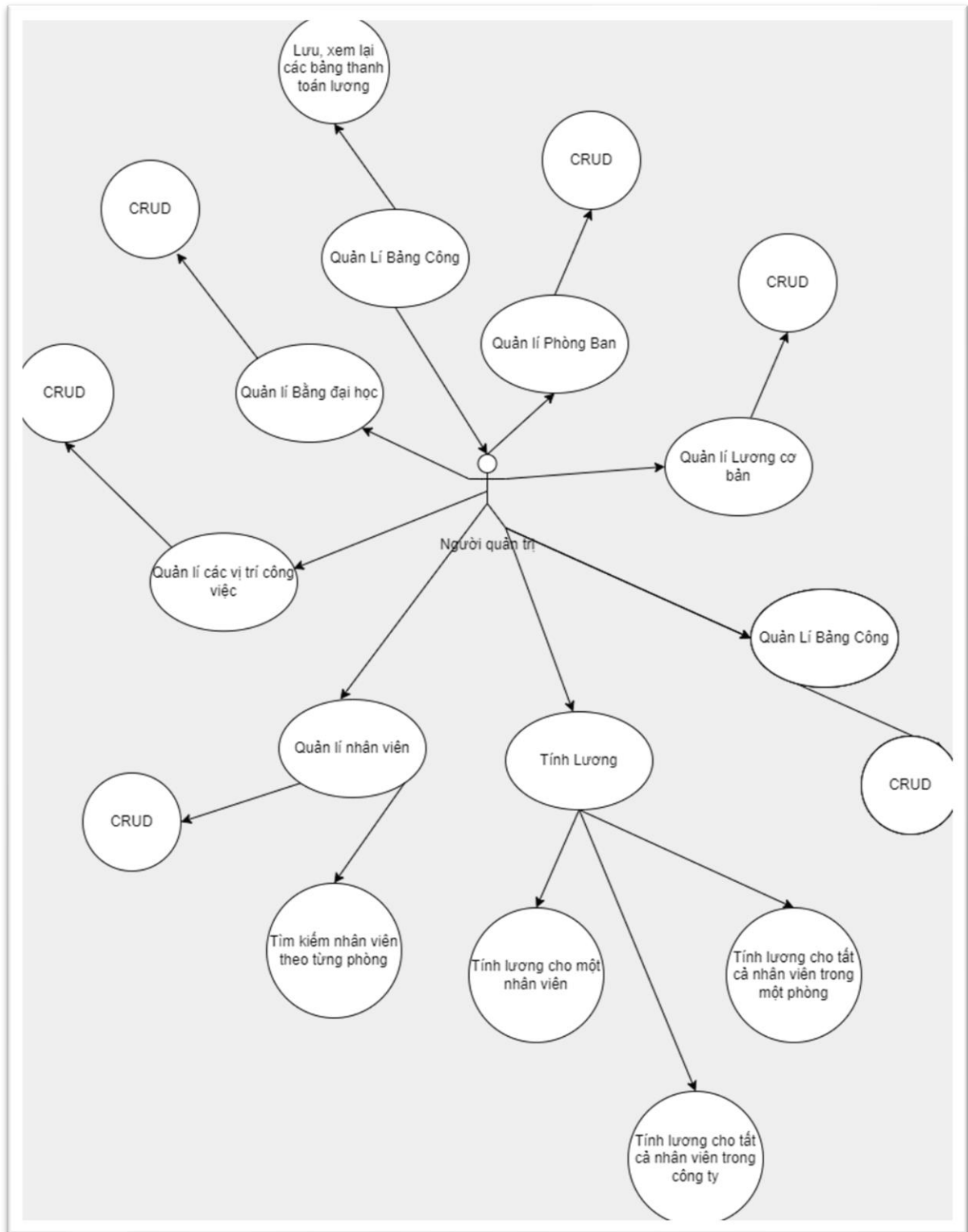
Thông thường với các view chúng ta sử dụng với Java như jsp, jsf muốn nhìn thấy được sự thay đổi của giao diện ta phải restart lại trang web. Tuy nhiên với thymeleaf ta không cần F5 (tải lại) lại trang cũng có thể xem các thay đổi trên giao diện, mặc định với thymeleaf nó không thể xử lý được vấn đề này, tuy nhiên cũng ta config nó rất đơn giản trong file cấu hình bằng 1 dòng lệnh config duy nhất (nó được gọi là cache, mình sẽ hướng dẫn và nói sâu vấn đề này trong loạt bài hướng dẫn về thymeleaf sắp tới.

Ví dụ: Khi các bạn viết css cho một thẻ html nào đó, muốn nhìn thấy thay đổi của giao diện, các bạn cần phải load lại trang, tuy nhiên với thymeleaf không cần load hay F5 lại trang, ta vẫn nhìn thấy được sự thay đổi đó.

Chương 3. XÂY DỰNG HỆ THỐNG

3.1. Xây dựng kiến trúc hệ thống

3.1.1. Sơ đồ use case



Hình 3.1. Sơ đồ use case

3.1.2.1. Danh sách Actor

Bảng 3.1. Bảng danh sách Actor

STT	Tên Actor	Ý Nghĩa
1	Người quản trị	Quản trị viên, nhân viên quản lý nhân sự, kế toán trong công ty

3.1.2.2. Danh sách Use case

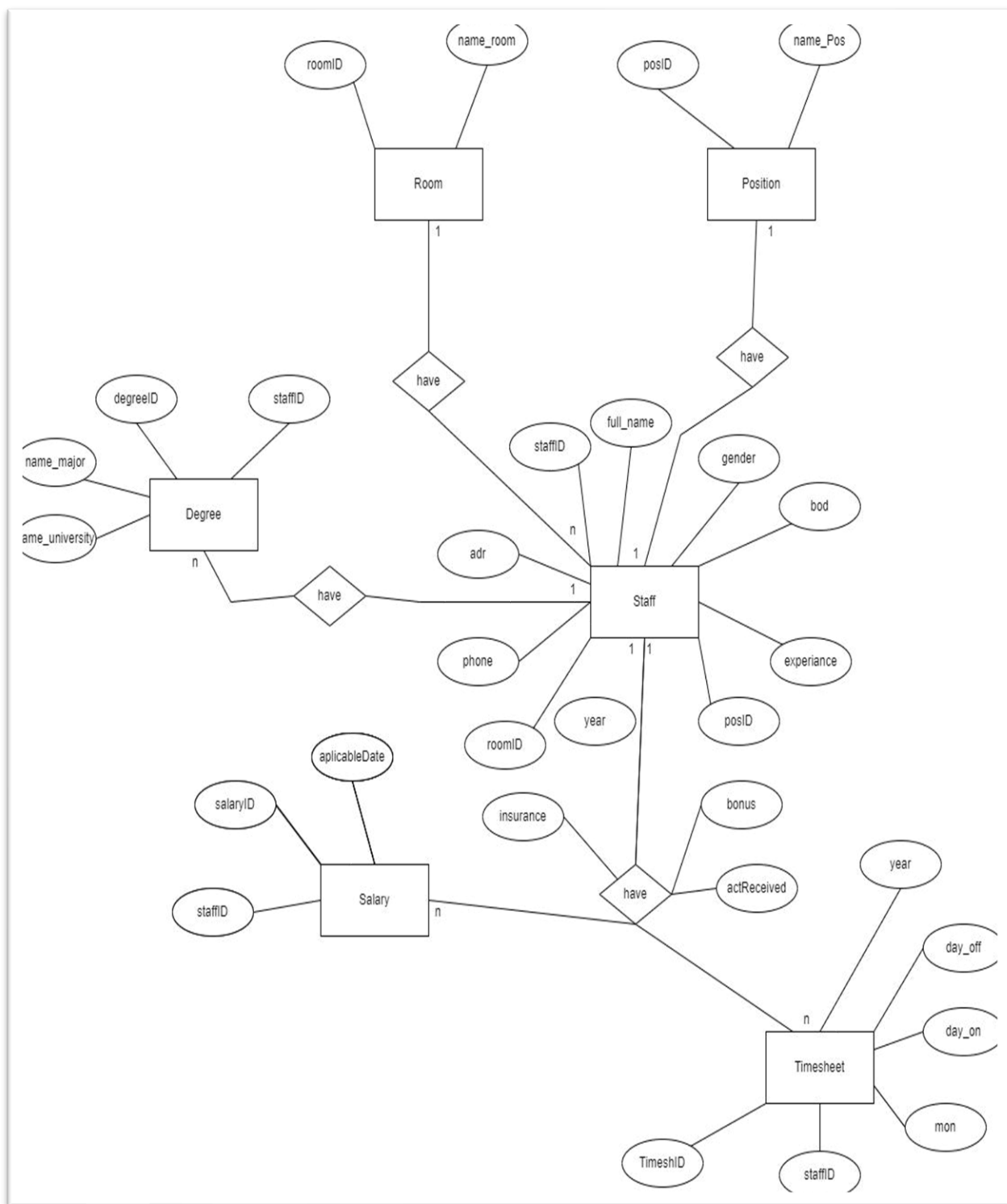
Bảng 3.2. Bảng danh sách use case

STT	Tên use case	Ý Nghĩa
1	Quản lý nhân viên	Thêm, xóa, sửa thông tin nhân viên
2	Quản lý bằng đại học	Thêm, xóa, sửa thông tin các bằng đại học trong công ty
3	Quản lý phòng ban	Thêm, xóa, sửa thông tin các phòng ban trong công ty
4	Quản lý các vị trí công việc	Thêm, xóa, sửa thông tin các vị trí công việc trong công ty
5	Quản lý lương cơ bản	Thêm, xóa, sửa thông tin lương cơ bản của các nhân viên trong công ty
6	Quản lý bản công	Thêm, xóa, sửa thông tin

		bảng công của các nhân viên trong công ty
7	Xem lại bảng thanh toán lương	Xem lại tất cả bảng thanh toán lương trong công ty
8	Tìm kiếm nhân viên theo phòng	Tìm tất cả nhân viên chung một phòng theo id của phòng
9	Tính lương cần trả cho một nhân viên theo tháng	Tính tiền lương phải trả cho nhân viên của một tháng nào đó
10	Tính lương cần trả cho tất cả nhân viên	Tính tiền lương phải trả cho tất cả nhân viên trong công ty của một tháng nào đó
11	Tính lương cần trả cho tất cả nhân viên theo phòng	Tính tiền lương phải trả cho tất cả nhân viên trong một phòng theo id của phòng

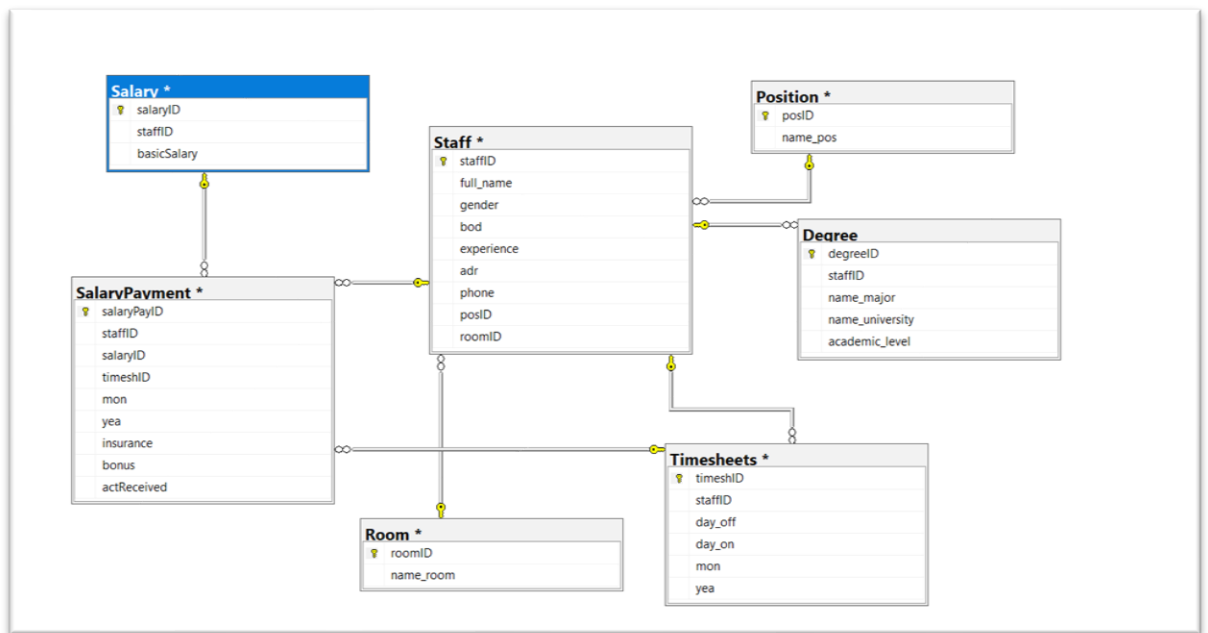
3.1.2. Cơ sở dữ liệu

1. Mô hình liên kết thực thể



Hình 3.2. Sơ đồ liên kết thực thể

2. Mô hình quan hệ



Hình 3.3. Mô hình quan hệ

3. CSDL vật lý

Bảng 3.3. Bảng Degree

Tên trường	Kiểu dữ liệu	Khóa	Giải thích
degreeID	varchar(6)	PK	Mã bảng
staffID	varchar(6)	FK	Mã nhân viên
name_major	varchar(100)		Tên chuyên ngành
name_university	varchar(100)		Tên trường đại học
academic_level	int		Trình độ học vấn
Bảng: Degree			

Bảng 3.4. Bảng Position

Tên trường	Kiểu dữ liệu	Khóa	Giải thích
posID	varchar(6)	PK	Mã chức vụ
name_pos	varchar(100)		Tên chức vụ
Bảng: Position			

Bảng 3.5. Bảng Room

Tên trường	Kiểu dữ liệu	Khóa	Giải thích
roomID	varchar(6)	PK	Mã phòng
name_room	varchar(50)		Tên phòng
Bảng: Room			

Bảng 3.6. Bảng Staff

Tên trường	Kiểu dữ liệu	Khóa	Giải thích
staffID	varchar(6)	PK	Mã nhân viên
full_name	varchar(100)		Tên của nhân viên
gender	bit		Giới tính
bod	date		Năm sinh
experience	int		Kinh nghiệm làm việc
adr	varchar(100)		Địa chỉ nhân viên
phone	varchar(11)		Số điện thoại nhân viên

posID	varchar(6)	FK	Mã chức vụ
roomID	varchar(6)	FK	Mã phòng
Bảng: Staff			

Bảng 3.7. Bảng Timesheets

Tên trường	Kiểu dữ liệu	Khóa	Giải thích
timeshID	varchar(6)	PK	Mã bảng chấm công
staffID	varchar(6)	FK	Mã nhân viên
day_off	int		Ngày nghỉ
day_on	int		Ngày làm
mon	int		Tháng
yea	int		Năm
Bảng: Timesheets			

Bảng 3.8. Bảng Salary

Tên trường	Kiểu dữ liệu	Khóa	Giải thích
salaryID	varchar(6)	PK	Mã lương
staffID	varchar(6)	FK	Mã nhân viên
basicSalary	float		Lương cơ bản
Bảng: Salary			

Bảng 3.9. Bảng SalaryPayment

Tên trường	Kiểu dữ liệu	Khóa	Giải thích
salaryPayID	varchar(6)	PK	Mã trả lương
staffID	varchar(6)	FK	Mã nhân viên
salaryID	varchar(6)	FK	Mã lương
timeshID	varchar(6)	FK	Mã bảng chấm công
mon	int		Tháng
yea	int		Năm
insurance	float		Bảo hiểm
bonus	float		Thưởng thêm
actReceived	float		Hoạt động nhận được
Bảng: SalaryPayment			

3.2. Thiết kế giao diện và mô tả xử lý

3.2.1 Menu Chính



Hình 3.4. Giao diện menu chính

3.2.2 Quản lý nhân viên

a. Trang chính

Staff list										
ID	Name	Gender	Birthday	Epx	Address	Phone	PosID	RoomID		
NV0001	Nguyen Van A	Male	2001-02-12	2	69 Nguyen Dinh Chieu	0978987678	POS001	ROO001	Edit Staff	<button>Delete</button>
NV0002	Nguyen Van B	Male	2001-04-22	3	79 Pham Van Dong	0978123745	POS001	12a45b	Edit Staff	<button>Delete</button>
NV0003	Nguyen Van C	Male	1999-04-22	5	55 Dien Bien Phu	01387123745	POS001	12a45b	Edit Staff	<button>Delete</button>
Insert new Staff										

Hình 3.5. Màn hình giao diện hiện thị thông tin staff

Có thể xóa hồ sơ thông tin nhân viên bằng cách nhấn vào nút Delete ở ngoài cùng bên phải hàng thông tin nhân viên muốn xóa, web sẽ reload lại và hàng thông tin nhân viên bị xóa sẽ biến mất

Khi nhấn vào link “Edit Staff” thì sẽ đi tới trang có giao diện dùng để update lại thông tin của nhân viên đó

b. Trang thêm mới một nhân viên

Insert new Staff

ID	Name	Gender	Birthday	Epx	Address	Phone	Position	Room	Status
		Male ▾	dd/mm/yyyy <input type="checkbox"/>	0			Manager ▾	Phòng nhân sự ▾	

[Back to Staffs list Page](#)

Hình 3.6. Màn hình giao diện trang thêm mới nhân viên

Đây là giao diện thêm mới một thông tin nhân viên, ID của nhân viên bắt buộc là 6 ký tự, các vị trí công việc và phòng ban chọn theo cái đã những option có sẵn trong công ty, có thể thêm vị trí công việc và phòng ban mới ở giao diện quản lý phòng ban và quản lý vị trí công việc

Khi nhập vào ID nhân viên đã tồn tại thì web sẽ hiện thị lỗi “staffID already exist”

c. Trang chỉnh sửa thông tin của một nhân viên

Update attributes of Staff : NV0001

ID	Name	Gender	Birthday	Epx	Address	Phone	PosID	RoomID	Status
NV0001	Nguyen Van A	Female ▾	02/12/2001 <input type="checkbox"/>	4	70 Nguyen Dinh Chau	0978987678	POS001	ROOM001	New edit

[Back to Staffs list Page](#)

Hình 3.7. Màn hình giao diện trang update thông tin nhân viên

Đây là giao diện update thông tin nhân viên khi mà ta đã nói khi này Ở đây các trường thông tin được giữ nguyên hoặc là để trống thì sẽ không update, chỉ update những trường đã bị thay đổi

3.2.3 Quản lý phòng ban

Có các chức năng tương tự như quản lý nhân viên

a. Trang chính

Room list

ID	Name	
124asf	Phòng nhân sự	<button>Delete</button>
12a45b	Phòng kỹ thuật	<button>Delete</button>
ROO001	DevOps	<button>Delete</button>

[Insert a new Room](#)

[Back to menu](#)

Hình 3.8. Màn hình giao diện hiện thị thông tin phòng ban

- b. Trang thêm mới một phòng ban

Insert new Room

RoomID	Name Room

Create

[Back to Room list Page](#)

Hình 3.9. Màn hình giao diện hiện thị thêm một phòng ban mới

3.2.4 Quản lý vị trí công việc

Có các chức năng thêm sửa xóa tương tự

- a. Trang chính

Position list

PosID	Name Position	
		Delete

Insert new Position

Position ID	Name Position
Create	

[Back to menu](#)

Hình 3.10. Màn hình giao diện hiển thị thông tin Vị trí công việc và thêm vị trí công việc mới

3.2.5 Quản lý bảng công

a. Trang chính

TimeSheets list

TimeshID	StaffID	Day off	Day on	Month	Year	
						Delete

Insert new Timesheet

TimeshID	StaffID	Day off	Day on	Month	Year
	▼				
Create					

[Back to menu](#)

Hình 3.11. Màn hình hiển thị thông tin bảng công và thêm mới một bảng công

Hình 3. : Màn hình hiển thị thông tin TimeSheets và Insert new TimeSheet

b. Tìm bảng công của nhân viên theo staffID

TimeSheets list by StaffID					
ID	StaffID	Day OFF	Day ON	Month	Year

Hình 3.12. Màn hình hiển thị thông tin các bảng công được tìm bằng id của nhân viên

3.2.6 Quản lý lương cơ bản

Có các chức năng thêm sửa xóa tương tự

a. Trang chính

Salary list

SalaryID	StaffID	Basic Salary	Applicable Date	
SAL001	NV0001	10000000.000 VND	2022-01-01	<button>Delete</button>

Insert new Salary

SalaryID	StaffID	Basic Salary	Applicable Date	Action
<input type="text"/>	<input type="text" value="NV0001"/>	<input type="text"/>	<input type="text" value="mm / dd / yyyy"/>	<input type="button" value="Create"/>

[Back to menu](#)

Hình 3.13. Màn hình hiển thị tất cả lương cơ bản và thêm mới lương cơ bản cho nhân viên trong công ty

3.2.7 Tính lương

a. Trang chính

Salary Menu

Choose calculation method

Get by StaffID

[Submit](#)

Ở giao diện tính lương sẽ có 3 option lựa chọn :

- + Tính lương một nhân viên theo staffID
- + Tính lương các nhân viên trong phòng theo roomID
- + Tính lương tất cả nhân viên

Hình 3.14. Màn hình hiển thị giao diện tính lương cho nhân viên

b. Trang chi tiết bảng tính lương nhân viên theo id

Detail Salary of Staff NV0001

Salary Payment ID	StaffID	Month	Year	Day off	Day on	Insurance	Bonus	Actually Received
10	NV0001	6	2022	0	30	500000.0	2000000.0	11.500.000,00

[Back to menu](#)

Hình 3.15. Chi tiết lương của một nhân viên

- c. Trang Not found data nếu không tìm thấy dữ liệu phù hợp để tính lương cho nhân viên



Hình 3.16. Trang hiển thị khi không tìm thấy dữ liệu phù hợp để tính lương

Khi không tìm thấy dữ liệu phù hợp (không có dữ liệu bảng công hoặc lương cơ bản) sẽ đi tới trang báo lỗi Not Found Data

Chương 4. KẾT LUẬN

4.1. Kết quả đạt được

Trong đồ án này, nhóm em đã vận dụng phương pháp phân tích thiết kế hướng cấu trúc để xây dựng phần mềm quản lý nhân viên và tính lương. Nhóm em đã đạt được những kết quả sau:

- Làm rõ được các yêu cầu của bài toán đặt ra.
- Áp dụng phương pháp thiết kế hướng cấu trúc để phân tích và thiết kế phần mềm dựa trên các thông tin thu được, bám sát vào yêu cầu của bài toán.
- Chương trình lưu trữ được những thông tin cần thiết của hệ thống.

4.2. Ưu điểm

- Giao diện đơn giản, người sử dụng dễ dàng sử dụng.
- Các thao tác xử lý đơn giản, không quá cầu kỳ.
- Hiện thị nội dung tương đối đầy đủ, người sử dụng dễ xem và hiểu được các chỉ số hiển thị.

4.3. Nhược điểm và giải pháp

Do thời gian hạn chế và kinh nghiệm thực tế chưa nhiều, nên kết quả còn hạn chế. Việc phân tích bài toán chưa mô tả hết được mọi khía cạnh của bài toán. Các giao diện chương trình chưa đạt được tính thân thiện cao. Hướng phát triển:

- Bổ sung và hoàn thiện các chức năng của chương trình để hoàn thiện tính tiện dụng với người dùng.
- Hoàn thiện giao diện chương trình để chương trình thân thiện, dễ sử dụng với người dùng.

TÀI LIỆU THAM KHẢO

[1] What is Spring Data JPA? And why should you use it?

<https://thorben-janssen.com/what-is-spring-data-jpa-and-why-should-you-use-it/>

[2] Hướng dẫn annotation @Query / NativeQuery trong Spring data

<https://hocspringboot.net/2021/05/25/huong-dan-annotation-query-nativequery-trong-spring-data/>

[3] Luồng đi của ứng dụng Spring MVC

<https://levunguyen.com/laptrinhspring/2020/02/01/springmvc/>

[4] Spring boot và Thymeleaf

<https://openplanning.net/11545/spring-boot-va-thymeleaf>

[5] Lập trình Spring MVC

<https://levunguyen.com/laptrinhspring/2020/02/01/springmvc/>

[6] Giới thiệu Spring Boot

<https://topdev.vn/blog/gioi-thieu-ve-spring-boot-spring-boot-la-gi/#:~:text=Spring%20Boot%20l%C3%A0%20m%E1%BB%99t%20d%E1%BB%B1,tri%E1%BB%83n%20business%20cho%20%E1%BB%A9ng%20d%E1%BB%A5ng.>

[7] Introduction to Spring Data JPA

<https://www.baeldung.com/the-persistence-layer-with-spring-data-jpa>

[8] Tìm hiểu mô hình Spring MVC

<https://vietnix.vn/tim-hieu-mo-hinh-mvc-la-gi/>

[9] Website học Spring Boot từ zero đến hero

<https://loda.me/courses/spring-boot>

[10] Khóa học Spring Boot trong vòng 2 giờ

<https://www.youtube.com/watch?v=UMePnyjr6FM>

[11] Viết ứng dụng CRUD với Java Spring MVC MySQL JPA jsp - phần 1&2

<https://www.youtube.com/watch?v=CX6Z12GwuTM&list=PLWBrqglNjN12xQqLFXnu5H2LzDuxpbutD&index=2>