BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

------🙢✧🙠*------*



**BÀI BÁO CÁO NHÓM HỌC PHẦN NIÊN LUẬN CƠ SƠ CT271**

**TÊN ĐỀ TÀI:**

**WEBSITE HỖ TRỢ QUẢN LÝ GIẢNG DẠY**

**CÁN BỘ HƯỚNG DẪN:**

**Lê Văn Lâm – MSCB:**

**SINH VIÊN THỰC HIỆN:**

**Đặng Nhật Duy – MSSV: B2105568**

**Khóa: 47**

**Lớp: DI21V7A2**

**Học Kỳ I, năm học: 2024 - 2025**

LỜI CẢM ƠN

T h a n k s

Trước tiên chúng em xin cảm ơn sâu sắc đến cha mẹ đã cổ vũ, động viên, hỗ trợ về tinh thần cũng như tạo mọi điều kiện thuận lợi cho em hoàn thành tốt tiểu luận này.

Chúng em xin chân thành cảm ơn thầy Lê Văn Lâm đã luôn quan tâm chỉ dạy, theo dõi, đã tận tình giúp đỡ, hỗ trợ cho tôi thực hiện bài niên luận này.

Em cũng xin gửi lời cám ơn chân thành đến các Thầy Cô Giảng viên Đại học Cần Thơ, đặc biệt là các Thầy Cô ở Trường CNTT & TT, những người đã truyền đạt những kiến thức quý báu trong thời gian qua.

Đồng cảm ơn đến các tác giả trong các quyển sách báo, internet, anh chị đi trước đã tìm tòi, nghiên cứu đúc kết kinh nghiệm làm tài liệu để em có thể tham khảo trong quá trình thực hiện đề tài.

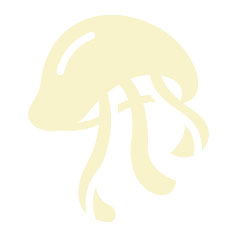
Sinh viên thực hiện

Đặng Nhật Duy – B2105568

# CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI

## **Giới thiệu đề tài**

Chủ đề: Website hỗ trợ quản lý giảng dạy

Tên ứng dụng: Shelly

### **Đặt vấn đề**

**Mục đích, yêu cầu của dự án (trả lời câu hỏi: Làm để làm gì và yêu cầu cần đạt được là gì?):**

**Ở phía trung tâm giảng dạy:**

Trung tâm A đang sở hữu một kho tàng kiến thức phong phú và đội ngũ giảng viên giàu kinh nghiệm. Tuy nhiên, việc tiếp cận của học viên đến các khóa học chất lượng này còn hạn chế. Để có một hệ sinh thái học tập ổn định và lành mạnh, trung tâm cần phát triển một trang web để giải quyết các yêu cầu trên.

**Ở phía giảng viên:**

Bạn là một giảng viên giảng dạy tại một trường Công Nghệ Thông Tin. Bạn có nhiều kinh nghiệm và kiến thức chuyên môn sâu nhưng việc sắp xếp môn học và chấm điểm theo phương pháp truyền thống khá khó khăn, bạn nhận thấy rằng:

* Khó quản lý tài liệu
* Học viên cần một lộ trình rõ ràng
* Khó để tìm kiếm thông tin
* Cần lập biểu đồ điểm để dễ dàng theo dõi tình trạng lớp

Về mặt tổng quan, dự án hướng tới những mục tiêu sau:

* Áp dụng công nghệ có khả năng xử lý trực tuyến, cơ sở dữ liệu tập trung.
* thân thiện, dễ sử dụng.
* Đáp ứng các chức năng quản trị, phân tích.
* Hệ thống báo cáo linh hoạt, có khả năng tùy biến cao.

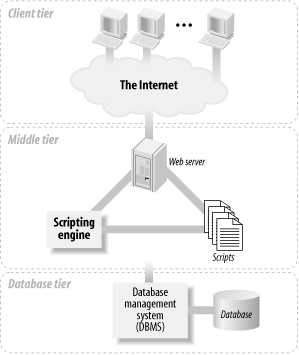
### **Các điểm nổi bật của hệ thống:**

Có thể vận hành tốt trên nhiều trình duyệt

Có khả năng tạo biểu đồ

Có tích hợp lịch từ Google Lịch

## **Kiến trúc hệ thống**

****Hệ thống website được xây dựng theo mô hình Client – Server, đây là một cấu trúc điển hình được sử dụng rộng rãi để xây dựng các ứng dụng web động.

Mô hình Client-Server bao gồm ba tầng chính: Client Tier (tầng khách hàng), Middleware (tầng trung gian), và Server Tier (tầng máy chủ).

**Về tầng Client Tier**

Đây là tầng gần với người dùng nhất và thường là người dùng (UI) mà họ tương tác trực tiếp.

Tầng này có nhiệm vụ:

* Thu thập dữ liệu đầu vào từ người dùng
* Gửi yêu cầu (request) đến tầng máy chủ
* Hiển thị dữ liệu trả về từ máy chủ.

**Về tầng Middleware hay Middle Tier**

Tầng này hoạt động như một cầu nối giữa tầng khách hàng và tầng máy chủ. Có trách nhiệm xử lý các yêu cầu từ Client Tier, quản lý giao tiếp, và thực hiện các logic xử lý cốt lõi trước khi gửi yêu cầu tới tầng máy chủ.

**Về tầng Server Tier**

Đây là tầng chịu trách nhiệm xử lý logic kinh doanh, truy xuất và quản lý dữ liệu từ cơ sở dữ liệu hoặc các nguồn tài nguyên khác.

Server Tier sẽ nhận yêu cầu từ Middleware, thực hiện các thao tác xử lý cần thiết, sau đó trả kết quả về cho tầng trung gian để gửi lại cho Client.

**Sự tương tác giữa các tầng**

1. **Client Tier** gửi yêu cầu đến **Middleware** thông qua các giao thức như HTTP, WebSocket, hoặc gRPC.
2. **Middleware** nhận yêu cầu, thực hiện các kiểm tra ban đầu (như xác thực, kiểm tra cache, nén dữ liệu), và gửi yêu cầu hợp lệ đến **Server Tier**.
3. **Server Tier** xử lý yêu cầu, lấy dữ liệu từ cơ sở dữ liệu, thực hiện logic cần thiết và gửi phản hồi lại cho **Middleware**.
4. **Middleware** sau đó xử lý kết quả, có thể lưu trữ tạm (cache) nếu cần, rồi gửi phản hồi này trở lại **Client Tier**.
5. **Client Tier** nhận và hiển thị kết quả cho người dùng cuối.

### **Phạm vi chức năng**

Hệ thống giải quyết các chức năng nghiệp vụ sau:

* Quản lý danh sách thông tin các giảng viên, các môn học và các học viên.
* Cập nhật và thay đổi thông tin giảng viên, môn học và học viên.
* Quản lý điểm của học viên.
* Xây dựng báo cáo điểm của học viên theo dạng biểu đồ.

### **Phạm vi phi chức năng**

Hệ thống bao gồm các yêu cầu phi chức năng sau:

* Người dùng chưa đăng nhập thì không thể truy cập vào bất kì đâu của trang Web ngoại trừ trang Đăng Nhập.
* Chỉ có quản trị viên mới có thể vào trang Human Resource.
* Đối với một môn học, quản trị viên có thể đánh dấu môn học, thêm, sửa hoặc xóa môn học. Giảng viên có thể đánh dấu môn học, thêm hoặc xóa ghi chú cho môn học, thêm, sửa hoặc xóa học viên.
* Đối với trang liên hệ, quản trị viên chỉ có thể xem các liên hệ do các tài khoản người dùng khác gửi. Giảng viên có thể gửi liên hệ cho quản trị viên nhưng không thể xem bất kì liên hệ nào khác.

### **Phạm vi triển khai**

Chương trình được triển khai trên toàn chi nhánh của trung tâm giảng dạy, cơ sở dữ liệu và ứng dụng được đặt tại máy chủ của trung tâm.

### **Quy trình hoạt động**

Quy trình hoạt động của một trung tâm dạy học sẽ gồm:

* **Lập kế hoạch và phát triển chương trình đào tạo:** Trưởng trung tâm sẽ lập kế hoạch chi tiết cho từng khóa học (phòng học, thiết bị, giảng viên, các chi phí cần thiết,…)
* **Lựa chọn giảng viên:** Tìm kiếm và tuyển chọn các chuyên gia, giảng viên có kinh nghiệm và kiến thức chuyên sâu trong lĩnh vực đào tạo. Giảng viên sẽ là người đề ra chương trình giảng dạy, thời lượng các khóa học và sẽ nộp lại cho trưởng trung tâm.
* **Tuyển sinh và đăng ký:** Quảng bá các khóa học thông qua các kênh truyền thông (website, mạng xã hội, email marketing, sự kiện...). Xây dựng hệ thống đăng ký online hoặc offline, thu thập thông tin cá nhân của học viên.
* **Quản lý lớp học:** Bao gồm việc xậy dựng lớp học, theo dõi tiến trình giảng dạy và hỗ trợ học viên trong vấn đề học tập.
* **Kiểm tra chất lượng học tập:** Có các bài kiểm tra định kỳ giúp đánh giá năng lực của học viên. Từ đó có thể cải thiện chất lượng học viên, đưa ra các đề xuất cải tiến chương trình học tốt hơn sau mỗi khóa học.

# CHƯƠNG II: CƠ SỞ LÝ THUYẾT

## **Công nghệ và thư viện**

### **PHP**

PHP (viết tắt của **PHP: Hypertext Preprocessor**) là một ngôn ngữ lập trình kịch bản phía máy chủ (server-side scripting language), được thiết kế chủ yếu để phát triển các ứng dụng web động.

PHP có thể nhúng trực tiếp vào mã HTML, giúp lập trình viên tạo ra các trang web động và tương tác một cách dễ dàng.

Em quyết định lấy php làm ngôn ngữ chính cho dự án của mình vì một số ưu điểm sau:

* Dễ dàng kết hợp và truy xuất giữa Front-end và Back-end.
* Hỗ trợ cơ sở dữ liệu có quan hệ.
* Thuận tiện kết hợp các công nghệ và thư viện thông qua composer

### **MySQL**

**MySQL** là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở, phổ biến nhất thế giới. MySQL hoạt động dựa trên ngôn ngữ SQL (**Structured Query Language**) và thường được sử dụng trong các ứng dụng web, đặc biệt khi kết hợp với PHP.

### **XAMPP**

**XAMPP** là một phần mềm mã nguồn mở, cung cấp một môi trường máy chủ cục bộ (local server) để phát triển ứng dụng web.

XAMPP được phát triển bởi **Apache Friends**, và nó thường được sử dụng để tạo một môi trường phát triển web cục bộ mà không cần kết nối internet.

Các thành phần chính trong XAMPP:

* + - Apache (Web Server): Là máy chủ web được sử dụng để chạy các ứng dụng web. Apache xử lý các yêu cầu HTTP và cung cấp nội dung trang web cho trình duyệt.
    - MariaDB hoặc MySQL (Database): Cung cấp hệ quản trị cơ sở dữ liệu để lưu trữ và quản lý dữ liệu cho các ứng dụng web.

=> XAMPP là một công cụ hoàn hảo để cung cấp môi trường cho dự án này.

### **Một số công nghệ khác**

**Chart.js**

Là một thư viện JavaScript mã nguồn mở dùng để tạo các biểu đồ trực quan, đẹp mắt trên trang web. Nó rất nhẹ (chỉ khoảng 60KB) và dễ dàng tích hợp với các ứng dụng web sử dụng HTML, CSS, và JavaScript.

**Google Calendar API**

Cho phép bạn truy cập dữ liệu từ Google Calendar (lịch Google) của người dùng, hiển thị và quản lý các sự kiện trên ứng dụng web cá nhân.

**Font Awesome**

Là một thư viện biểu tượng (icons) phổ biến, cung cấp hàng ngàn biểu tượng đẹp mắt và dễ dàng sử dụng cho các trang web và ứng dụng. Biểu tượng được cung cấp ở dạng vector, có thể tùy chỉnh kích thước, màu sắc, và hiệu ứng.

**Composer & Packagist**

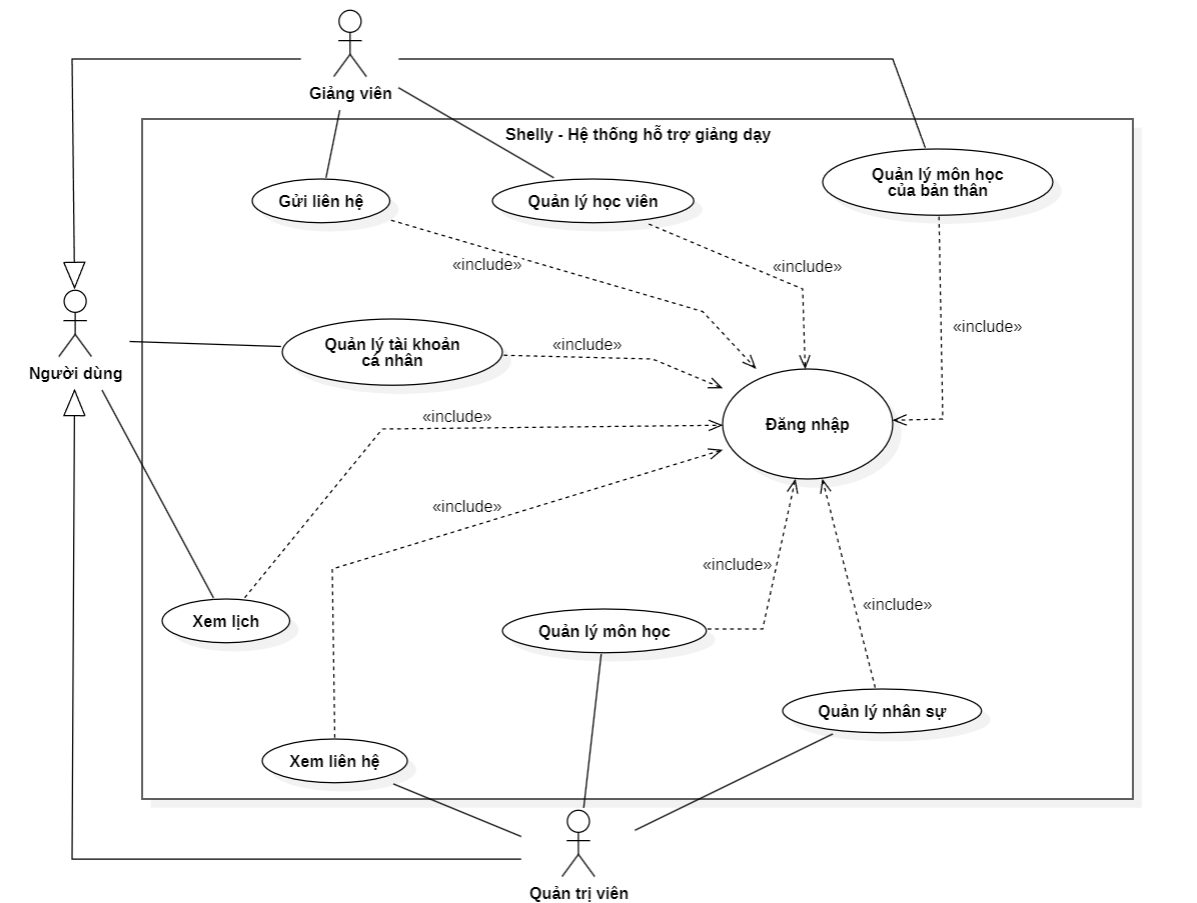
**Composer** là một công cụ quản lý phụ thuộc (dependency manager) dành cho PHP. Nó giúp bạn quản lý các thư viện và gói (packages) cần thiết cho dự án một cách tự động, thay vì phải tải và tích hợp thủ công. Composer tương tự như **npm** cho JavaScript hay **pip** cho Python.

**Packagist** là kho lưu trữ chính thức của Composer, nơi chứa danh sách các gói thư viện PHP có sẵn để sử dụng.

## **Mô hình chức năng**

### **Sơ đồ Use Case tổng quát**

Sơ đồ Use Case tổng quát được thể hiện ở Hình… mô tả về tổng quan các chức năng của toàn bộ hệ thống hỗ trợ giảng dạy.

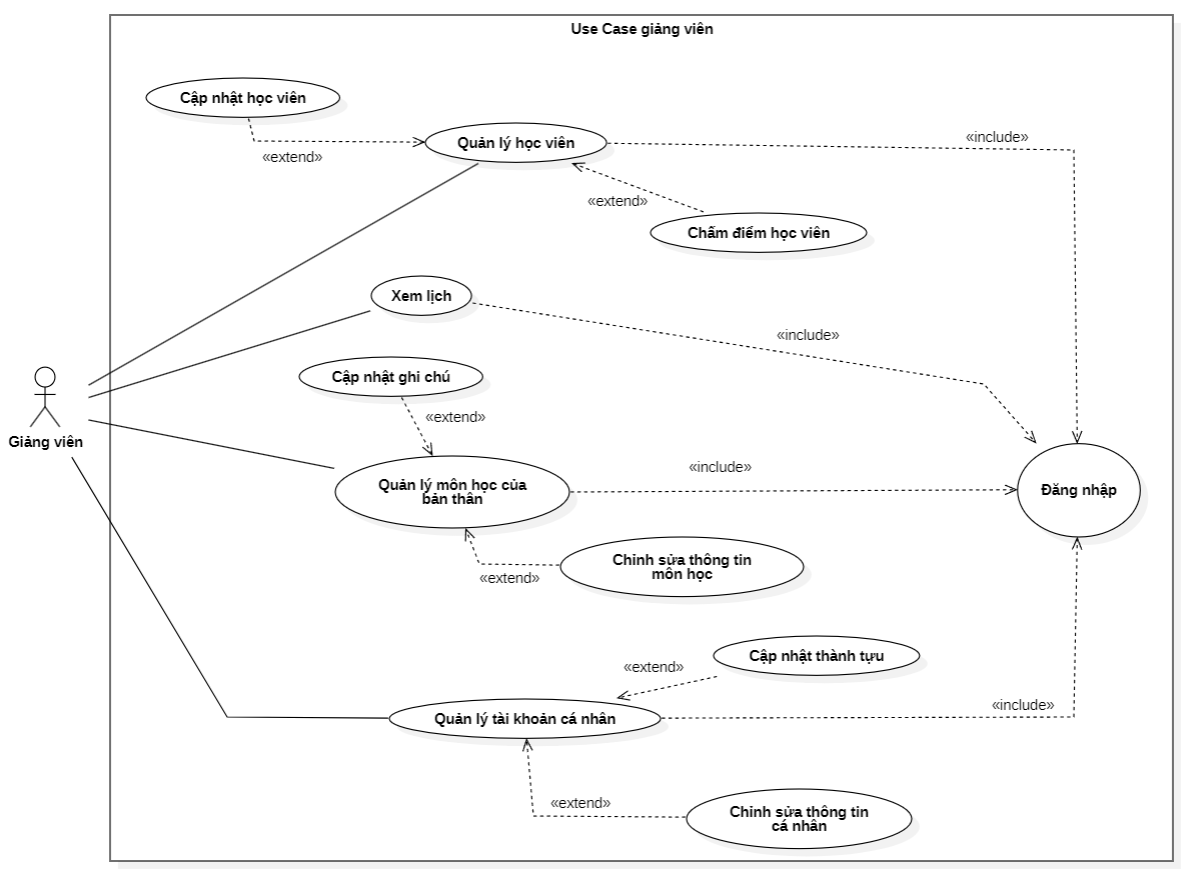


Các nhóm người dùng (actor) của hệ thống: Quản trị viên, Giảng viên. Cả hai tác nhân này đều kế thừa từ tác nhân Người dùng.

Mỗi tác nhân sẽ có một số chức năng và vai trò riêng trong hệ thống. Sẽ được thể hiện chi tiết hơn trong phần Use Case của từng tác nhân.

### **Sơ đồ Use Case giảng viên**

Sơ đồ Use Case giảng viên được thể hiện ở Hình… mô tả về tổng quan các chức năng của Actor Giảng Viên.



**Chức năng đăng nhập:**

Cho phép Giảng Viên đăng nhập vào hệ thống. Để đăng nhập, Giảng Viên cần điền đầy đủ thông tin bao gồm: Tài khoản và mật khẩu (Do quản trị viên cấp).

**Chức năng quản lý tài khoản cá nhân:**

Cho phép Giảng Viên cập nhật một số thông tin cá nhân bao gồm:

* Ảnh đại diện (avatar)
* Họ tên
* Ngày sinh
* Chuyên ngành
* Giới tính
* Mật khẩu

Ngoài ra Giảng Viên còn có thể cập nhật (thêm / xóa) các thành tựu của bản thân.

**Chức năng xem lịch:**

Hệ thống có tích hợp tính năng Google Calendar API nên cho phép người dùng xem lịch và các sự kiện có trên lịch.

**Chức năng quản lý môn học của bản thân:**

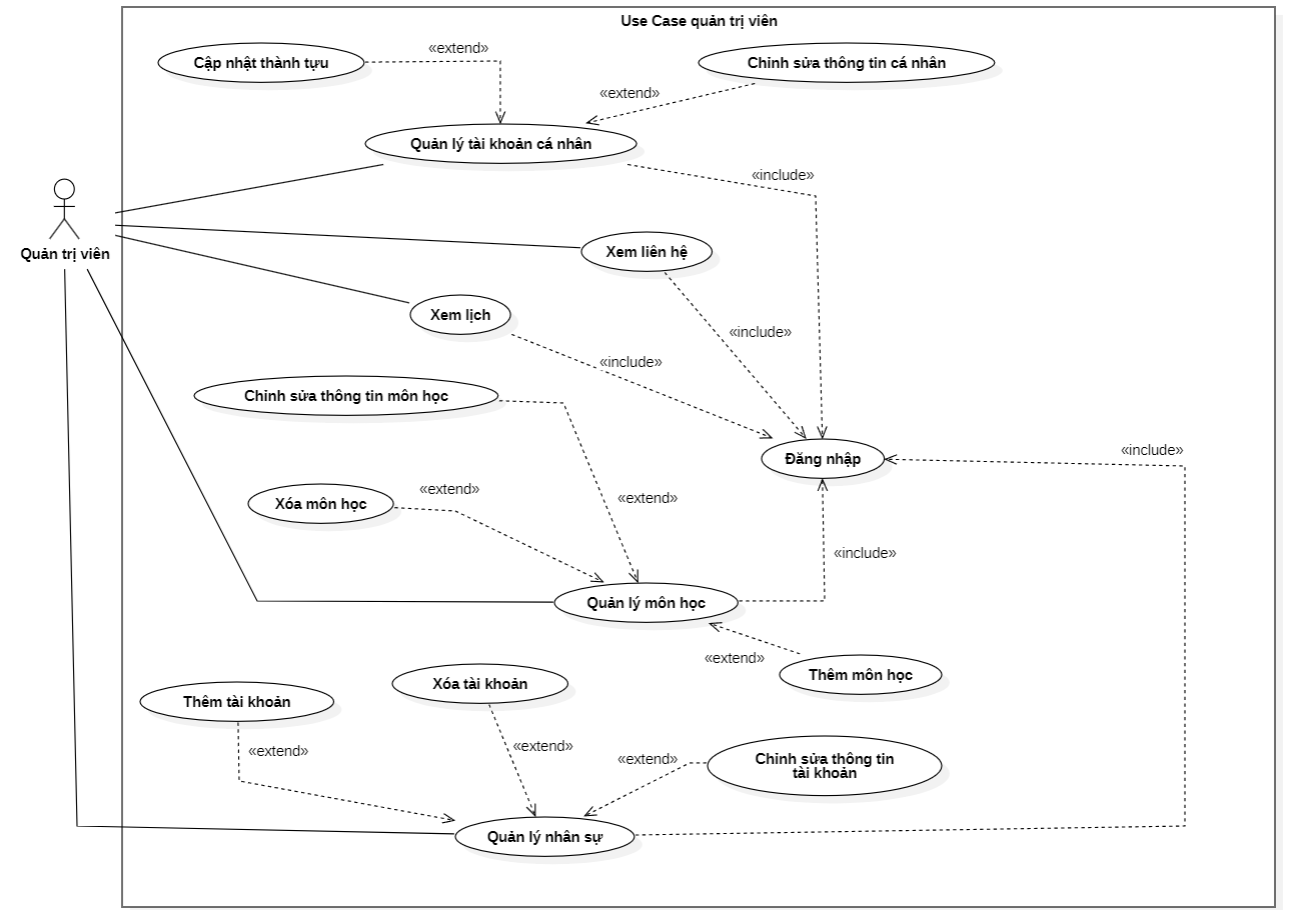
Cho phép Giảng viên thực hiện các thao tác trên môn học mà người đó phụ trách giảng dạy, bao gồm: Chỉnh sửa thông tin môn học, cập nhật (thêm / xóa) ghi chú.

**Chức năng quản lý học viên:**

Chức năng này giúp Giảng viên có thể quản lý các học viên trong môn học mình đang phụ trách. Bao gồm các thao tác: Cập nhật (thêm / sửa / xóa) học viên và chấm điểm cho học viên.

### **Sơ đồ Use Case quản trị viên**

Sơ đồ Use Case quản trị viên được thể hiện ở Hình… mô tả về tổng quan các chức năng của Actor Quản Trị Viên.



**Chức năng đăng nhập:**

Cho phép Quản Trị Viên đăng nhập vào hệ thống. Để đăng nhập, Quản Trị Viên cần điền đầy đủ thông tin bao gồm: Tài khoản và mật khẩu (Do quản trị viên cấp).

**Chức năng xem lịch:**

Hệ thống có tích hợp tính năng Google Calendar API nên cho phép người dùng xem lịch và các sự kiện có trên lịch.

Chức năng xem liên hệ:

Quản Trị Viên có thể xem các liên hệ do Giảng Viên gửi.

**Chức năng quản lý môn học:**

Cho phép Quản Trị Viên thao tác trên tất cả môn học. Các thao tác bao gồm: Đánh dấu môn học, thêm môn học, chỉnh sửa môn học, xóa môn học.

**Chức năng quản lý nhân sự:**

Cho phép Quản Trị Viên thực hiện các thao tác trên tất cả tài khoản. Các thao tác bao gồm: Thêm tài khoản mới, chỉnh sửa thông tin của tài khoản, xóa tài khoản.

**Chức năng quản lý tài khoản cá nhân:**

Cho phép Giảng Viên cập nhật một số thông tin cá nhân bao gồm:

* Ảnh đại diện (avatar)
* Họ tên
* Ngày sinh
* Chuyên ngành
* Giới tính
* Mật khẩu

Ngoài ra Giảng Viên còn có thể cập nhật (thêm / xóa) các thành tựu của bản thân.

## **Các prototype**

Công cụ thiết kế: Figma

Link xem các prototype đã thiết kế: https://www.figma.com/design/lXB8OFQugQqAuQQ7r0znmx/NLCS---Shelly?node-id=9-14&t=cC9xU2lsImNLXszv-1

# CHƯƠNG III: GIAO DIỆN HỆ THỐNG

## **Trang Chủ**

## **Trang Đăng Nhập**

## **Trang Liên Hệ**

## **Trang Danh Sách Môn Học**

## **Trang Chi Tiết Môn Học**

## **Các thành phần (Partials)**

### **Header**

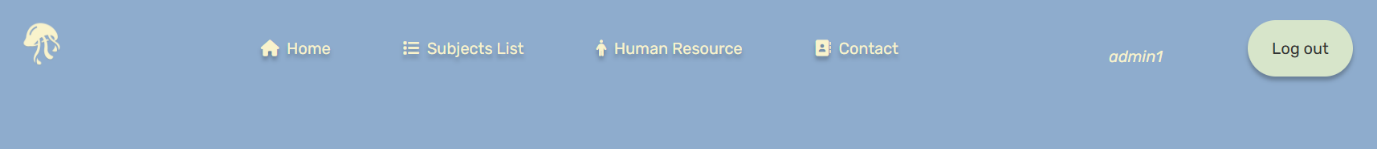
Giao diện Header của hệ thống được xem là một thành phần. Có vị trí trên cùng của gần như tất cả các trang trừ Trang Đăng Nhập.

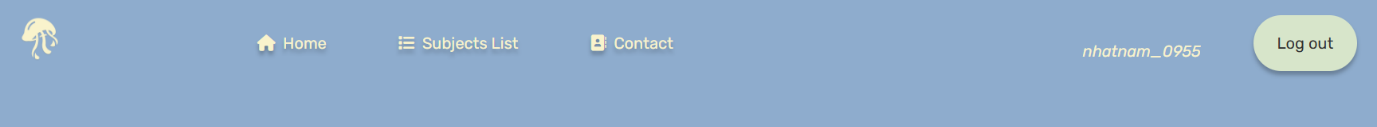
**Chức năng:**

Cho phép người dùng điều hướng sang các trang khác bao gồm:

* Home: Trang chủ.
* Subject List: Trang danh sách môn học.
* Contact: Trang liên hệ.
* Nếu người dùng là quản trị viên (Adminsator) thì có thêm phần điều hướng đến Human Resource: Quản lý nhân viên.

Có thể đăng xuất khỏi tài khoản thông qua nút Log out.

**Hình ảnh giao diện Header:**



### **Footer**

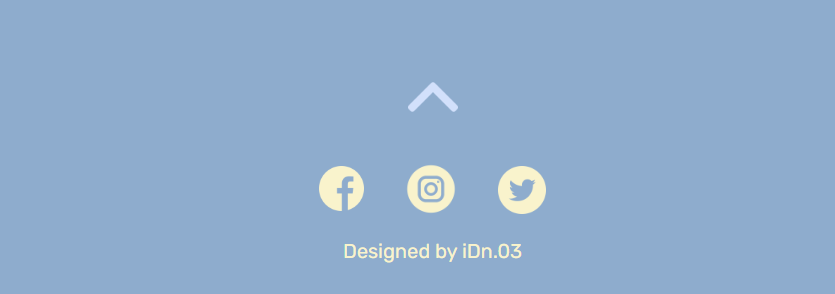
Giao diện Footer của hệ thống được xem là một thành phần. Có vị trí dưới cùng của gần như tất cả các trang trừ Trang Đăng Nhập.

Chức năng:

Cho phép điều hướng lên đầu trang thông qua nút mũi tên hướng lên.

Chứa các nền tảng mạng xã hội của người thiết kế.

Chứa thông tin cơ bản của người thiết kế.

Hình ảnh giao diện:

### **Empty State**

### **Toast Message**

# CHƯƠNG IV: DỮ LIỆU HỆ THỐNG

## **Sơ đồ Cơ Sở Dữ Liệu**

## **Các bảng cơ sở dữ liệu**

# CHƯƠNG V: KẾT LUẬN

## Kết quả đạt được

## Hướng cải thiện và phát triễn

# TÀI LIỆU THAM KHẢO

# **PHỤ LỤC**