****

VIETNAM-KOREA UNIVERSITY OF IINFORMATION AND COMMUNICATION TECHNOLOGY

**FACULTY OF COMPUTER SCIENCE**

**GRADUATION THESIS REPORT**

**PROJECT NAME**

**Student name : Le Thanh An**

**Le Van Hieu**

**Class : 19IT2**

**Instructor : Ph.D Nguyen Huu Nhat Minh**

**School year : 2019 - 2024**

***Da Nang – December 2023***

ADJUSTMENT

(this is some adjustment of instructor)

∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙∙

*Da Nang, December 20, 2023*

**Instructor signature**

SINCERELY THANKS

Để hoàn thành bài báo cáo thực tập với đề tài: “Xây dựng hệ thống quản lý đồ án” cho em gửi lời cảm ơn đến thầy cô tại Phòng Đào tạo - Trường đại học Công nghệ thông tin và Truyền thông Việt-Hàn đã tạo điều kiện, gợi ý cơ quan thực tập để em có điều kiện tiếp cận doanh nghiệp và nghiên cứu thực tiễn. Cảm ơn cô Ths. Dương Thị Mai Nga đã hướng dẫn, hỗ trợ em trong quá trình hoàn thiện bài báo cáo thực tập.

Trong đợt thực tập tại Phòng đào tạo – Trường đại học Công nghệ thông tin và Truyền thông Việt - Hàn, chân thành cảm ơn tới các thầy/cô tại phòng đã giúp đỡ nhiệt tình và đặc biệt em gửi lời cảm ơn đễn người hướng dẫn thầy Ths. Lê Song Toàn đã tận tình hướng dẫn, cung cấp số liệu, đặc ra các yêu cầu thực tế giúp em hoàn thành đợt thực tập doanh nghiệp này.

Trong quá trình tham gia thực tập tại doanh nghiệp do kinh nghiệm còn hạn chế nên dù cố gắng thế nào cũng không thể tránh khỏi sai sót. Những ý kiến quan tâm, góp ý của thầy cô là kiến thức quý báu giúp em nhận ra những hạn chế trong bài báo cáo và giúp bài báo cáo hoàn chỉnh hơn.

Em xin chân thành cảm ơn.

MỤC LỤC

[PHIẾU NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP i](#_Toc145564612)

[NHẬN XÉT ii](#_Toc145564613)

[LỜI CẢM ƠN iii](#_Toc145564614)

[MỤC LỤC iv](#_Toc145564615)

[DANH MỤC HÌNH ẢNH vi](#_Toc145564616)

[DANH MỤC BẢNG vii](#_Toc145564617)

[MỞ ĐẦU 1](#_Toc145564618)

[CHƯƠNG 1: GIỚI THIỆU VỀ PHÒNG ĐÀO TẠO 3](#_Toc145564619)

[1.1. Thông tin liên hệ 3](#_Toc145564620)

[1.2. Giới thiệu chung 3](#_Toc145564621)

[1.3. Chức năng, nhiệm vụ 3](#_Toc145564622)

[1.4. Cơ cấu, tổ chức 4](#_Toc145564623)

[1.5. Mục tiêu hoạt động 4](#_Toc145564624)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT LIÊN QUAN ĐẾN ĐỀ TÀI 6](#_Toc145564625)

[2.1. Ngôn ngữ lập trình PHP 6](#_Toc145564626)

[2.1.1. Khái niệm ngôn ngữ PHP 6](#_Toc145564627)

[2.1.2. Tại sao nên dùng PHP 6](#_Toc145564628)

[2.1.3. Ưu và nhược điểm của ngôn ngữ PHP 7](#_Toc145564629)

[2.1.4. Kiểu dữ liệu trong PHP 7](#_Toc145564630)

[2.2. Lý thuyết về mô hình MVC và Framework Laravel 7](#_Toc145564631)

[2.2.1. Framework là gì? PHP Framework là gì ? 7](#_Toc145564632)

[2.2.2. Khái niệm Framework Laravel 8](#_Toc145564633)

[2.2.3. Mô hình MVC trong Laravel 8](#_Toc145564634)

[2.2.4. Sơ lược các tính năng cơ bản trong Laravel 9](#_Toc145564635)

[2.3. Hệ quản trị cơ sở dữ liệu MySQL 10](#_Toc145564636)

[2.3.1. Khái niệm 10](#_Toc145564637)

[2.3.2. Tại sao sử dụng MySQL 10](#_Toc145564638)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 12](#_Toc145564639)

[3.1. Tổng quan về hệ thống quản lý đào tạo 12](#_Toc145564640)

[3.1.1. Mô tả hệ thống hiện tại 12](#_Toc145564641)

[3.1.2. Những hạn chế của hệ thống 12](#_Toc145564642)

[3.2. Phân tích và thiết kế hệ thống quản lý đồ án 12](#_Toc145564643)

[3.2.1. Yêu cầu chức năng 12](#_Toc145564644)

[1.1.1. Yêu cầu phi chức năng 13](#_Toc145564645)

[3.3. Biểu đồ mô hình hóa UML 15](#_Toc145564646)

[3.3.1. Biểu đồ ca sử dụng 15](#_Toc145564647)

[3.3.2. Biểu đồ lớp 22](#_Toc145564648)

[3.3.3. Biểu đồ hoạt động 23](#_Toc145564649)

[3.3.4. Thiết kế cơ sở dữ liệu 27](#_Toc145564650)

[CHƯƠNG 4: TRIỂN KHAI XÂY DỰNG HỆ THỐNG 28](#_Toc145564651)

[4.1. Khởi tạo môi trường xây dựng 28](#_Toc145564652)

[4.2. Kết quả xây dựng chương trình 29](#_Toc145564653)

[4.2.1. Chức năng đăng nhập 29](#_Toc145564654)

[4.2.2. Các chức năng của Khoa 30](#_Toc145564655)

[4.2.3. Các chức năng của Giảng viên 34](#_Toc145564656)

[4.2.4. Các chức năng của Sinh viên 35](#_Toc145564657)

[KIẾN NGHỊ VÀ KẾT LUẬN 37](#_Toc145564658)

[TÀI LIỆU THAM KHẢO 38](#_Toc145564659)

DANH MỤC HÌNH ẢNH

[Hình 2.1. Mô hình MVC trong Laravel 8](#_Toc144713233)

[Hình 3.1. Biểu đồ use-case tổng quát 15](#_Toc144713234)

[Hình 3.2. Biểu đồ use-case chức năng đăng nhập 15](#_Toc144713235)

[Bảng 3.1. Mô tả use-case đăng nhập 15](#_Toc144713236)

[Hình 3.3. Biểu đồ use-case quản lý lớp đồ án của Khoa 16](#_Toc144713237)

[Bảng 3.2. Mô tả use-case quản lý lớp đồ án 16](#_Toc144713238)

[Hình 3.4. Biểu đồ use-case quản lý nhóm lớp đồ án của Khoa 17](#_Toc144713239)

[Bảng 3.3. Mô tả use-case quản lý nhóm lớp đồ án 18](#_Toc144713240)

[Hình 3.5. Biểu đồ use-case quản lý đợt bảo vệ của Khoa 18](#_Toc144713241)

[Bảng 3.4. Mô tả use-case quản lý đợt bảo vệ 19](#_Toc144713242)

[Hình 3.6. Biểu đồ use-case quản lý hội đồng chấm của Khoa 19](#_Toc144713243)

[Bảng 3.5. Mô tả use-case quản lý hội dồng chấm 19](#_Toc144713244)

[Hình 3.7. Biểu đồ use-case chi tiết các chức năng của Giảng Viên 20](#_Toc144713245)

[Bảng 3.6. Mô tả use-case các chức năng của Giảng viên 20](#_Toc144713246)

[Hình 3.8. Biểu đồ use-case chi tiết các chức năng của sinh viên 21](#_Toc144713247)

[Hình 3.9. Biểu đồ lớp 22](#_Toc144713248)

[Hình 3.10. Biểu đồ hoạt động chức năng đăng nhập 23](#_Toc144713249)

[Hình 3.11. Biểu đồ hoạt động chức năng của Khoa 24](#_Toc144713250)

[Hình 3.12. Biểu đồ hoạt động chức năng của Giảng viên 25](#_Toc144713251)

[Hình 3.13. Biểu đồ hoạt dộng các chức năng của Sinh viên 26](#_Toc144713252)

[Hình 3.14. Mô hình các bảng cơ sở dữ liệu 27](#_Toc144713253)

[Hình 4.1. Giao diện đăng nhập 29](#_Toc144713254)

[Hình 4.2. Giao diện xem danh sách lớp đồ án 30](#_Toc144713255)

[Hình 4.3.Giao diện cài đặt lớp đồ án 30](#_Toc144713256)

[Hình 4.4. Giao diện quản lý nhóm lớp đồ án 31](#_Toc144713257)

[Hình 4.5. Giao diện phân sinh viên vào hội đồng bảo vệ 31](#_Toc144713258)

[Hình 4.6. Giao diện cài đặt lớp đồ án 32](#_Toc144713259)

[Hình 4.7. Giao diện phân nhóm lớp đồ án 32](#_Toc144713260)

[Hình 4.8. Giao diện quản lý đợt bảo vệ 33](#_Toc144713261)

[Hình 4.9. Giao diện quản lý hội đồng chấm 33](#_Toc144713262)

[Hình 4.10. Giao diện xem danh sách lớp đồ án của Giảng viên 34](#_Toc144713263)

[Hình 4.11. Giao diện xem chi tiết nhóm sinh viên thực hiện đồ án 34](#_Toc144713264)

[Hình 4.12. Giao diện quản lý nhóm sinh viên thực hiện đồ án 35](#_Toc144713265)

[Hình 4.13. Giao diện xem chi tiết lớp đồ án của Sinh viên 35](#_Toc144713266)

DANH MỤC BẢNG

[Bảng 1.1. Cơ cấu, tổ chức Phòng Đào tạo 4](#_Toc144713650)

[Bảng 3.1. Mô tả use-case đăng nhập 15](#_Toc144713651)

[Bảng 3.2. Mô tả use-case quản lý lớp đồ án 16](#_Toc144713652)

[Bảng 3.3. Mô tả use-case quản lý nhóm lớp đồ án 18](#_Toc144713653)

[Bảng 3.4. Mô tả use-case quản lý đợt bảo vệ 19](#_Toc144713654)

[Bảng 3.5. Mô tả use-case quản lý hội dồng chấm 19](#_Toc144713655)

[Bảng 3.6. Mô tả use-case các chức năng của Giảng viên 20](#_Toc144713656)

OVERVIEW

* + - * 1. Lý do thực tập
* Thực tập tốt nghiệp là cơ hội trải nghiệm tuyệt vời giúp sinh viên làm quen môi trường thực tế, vận dụng và hoàn thiện kiến thức, trau dồi kỹ năng, nghiệp vụ… và chuẩn bị sẵn sàng để bắt đầu công việc trong tương lai.
* Vì vậy, Trường Đại học Công nghệ Thông tin và Truyền thông Việt - Hàn đã phối hợp với doanh nghiệp tạo điều kiện tốt nhất cho các sinh viên năm hoàn thành học phần quan trọng này.
  + - * 1. Mục tiêu thực tập

Thông qua kỳ thực tập doanh nghiệp, em hướng đến các mục tiêu chính sau:

* Hoàn thành một cách tốt nhất học phần “Thực tập tốt nghiệp”.
* Tiếp cận công việc thực tế sẽ làm trong tương lai.
* Có thêm nhiều mối quan hệ với các đồng nghiệp tại đơn vị thực tập.
* Trau dồi, học hỏi các kỹ năng mềm(giao tiếp, ngoại ngữ, làm việc nhóm, ...).
* Phấn đấu hoàn thành tốt các công việc, dự án được giao tại đơn vị thực tập để có thể được giữ lại làm nhân viên chính thức.
  + - * 1. Phạm vi thực tập
* Nơi thực tập: Phòng Đào tạo Trường Đại học Côg nghệ Thông tin và Truyền Thông Việt – Hàn
* Thời gian bắt đầu thực tập: Ngày 26 tháng 6 năm 2023.
* Thời gian kết thúc thực tập: Ngày 04 tháng 09 năm 2023.
  + - * 1. Phương pháp nghiên cứu

Trong quá trình thực tập, để hoàn thành tốt các công việc được giao, em tiếp cận các

công việc theo các phương pháp sau:

* Tự tìm hiểu (học hỏi, quan sát, tham khảo ý kiến).
* Vận dụng kiến thức đã học tại trường và qua các khóa học trực tuyến để tiến hành tham gia thực hiện dự án.
* Báo cáo và nhận phản hồi từ người hướng dẫn(Test Leader).
* Học hỏi từ đồng nghiệp bằng cách giao tiếp trực tiếp và trực tuyến(email, mạng xã hội, ...).
  + - * 1. Nội dung nghiên cứu

Để hoàn thành đề tài thực tập doanh nghiệp lần này, em tập trung nghiên cứu các nội dung chính sau:

* Làm quen với môi trường làm việc, nội quy và quy định.
* Tìm hiểu về quy trình trình làm việc
* Học và tìm hiểu cách tạo ra mô hình.
* Thiết kế mô hình thực dụng với thực tiễn.
  + - * 1. Bố cục bài thực tập

Mở đầu: Trình bày lý do, mục tiêu, phạm vi, phương pháp và nội dung nghiên cứu và bố cục tổng quan bài báo cáo.

Chương 1: Giới thiệu về phòng đào tạo.

Chương 2: Cơ sở lý thuyết liên quan đến đề tài

Chương 3: Phân tích và thiết kế hệ thống

Chương 4: Triển khai xây dựng

Kiến nghị và kết luận: Nêu kết quả đạt được và các đề xuất, giải pháp.

# THEORETICAL BASIS

## Blockchain and smart contract overview

## Technology is used in front-end

## Technology is used in back-end

# CƠ SỞ LÝ THUYẾT LIÊN QUAN ĐẾN ĐỀ TÀI

## Ngôn ngữ lập trình PHP

### Khái niệm ngôn ngữ PHP

Ngôn ngữ, các thư viện, tài liệu gốc của PHP được xây dựng bởi cộng đồng và có sự đóng góp rất lớn của Zend Inc., công ty do các nhà phát triển cốt lõi của PHP lập nên nhằm tạo ra một môi trường chuyên nghiệp để đưa PHP phát triển ở quy mô doanh nghiệp.

### Tại sao nên dùng PHP

80% trong số 10 triệu trang web hàng đầu sử dụng ngôn ngữ lập trình PHP. PHP rất phổ biến đối với công việc tự do, bởi vì nhiều doanh nghiệp nhỏ hoặc những người không có kỹ thuật muốn sử dụng WordPress, Wix hoặc các hệ thống quản lý nội dung phổ biến khác để thiết lập trang web hoặc website bán hàng. PHP được thiết kế cho web và nó hoạt động tốt. Những “gã” khổng lồ trong giới lập trình như Flickr, Wikipedia, Tumblr và Facebook đã sử dụng kết hợp framework PHP cho hệ thống websites của mình.

PHP đã được sử dụng để phát triển các trang web thương mại điện tử, ứng dụng GUI, ứng dụng Facebook, công cụ quản lý dự án, trang web CMS và nhiều ứng dụng khác. Với các ứng dụng đa dạng và sự thống trị của nó trên thị trường, PHP nắm giữ một tương lai đầy hứa hẹn.

PHP không bị ràng buộc bởi các quy tắc; nó giúp cho lập trình viên được tự do sáng tạo để xây dựng mọi thứ mình muốn. Sự thân thiện với người dùng và tính linh hoạt chính là lợi thế lớn nhất của PHP.

Ngoài ra, khi nói đến việc xây dựng các ứng dụng dành riêng cho dự án, các lập trình viên sẽ không cần lo lắng về việc xây dựng từ đầu. PHP có nhiều frameworks kết hợp được sử dụng để thêm các chức năng cụ thể. Một số Framework phổ biến như: Laravel, CodeIgniter, Yii 2, Phalcon, CakePHP,…

Trước tiên, quy mô của cộng đồng rất quan trọng, bởi vì cộng đồng ngôn ngữ lập trình càng lớn thì bạn càng có nhiều cơ hội nhận được hỗ trợ. Khi bước chân vào thế giới lập trình, bạn sẽ sớm hiểu được tầm quan trọng của việc hỗ trợ bởi vì cộng đồng các nhà phát triển là nơi nhận và đưa ra trợ giúp. Ngoài ra, trong một cộng đồng lớn sẽ có nhiều người xây dựng được các công cụ hữu ích giúp giảm bớt quá trình phát triển ngôn ngữ cụ thể. Hiện tại, có hơn 600 ngôn ngữ lập trình nổi tiếng trên toàn thế giới.

Ngôn ngữ lập trình PHP được coi là một trong những ngôn ngữ lập trình dễ học nhất. Những người mới bắt đầu thường thấy rất dễ để tham gia vào việc phát triển website bằng cách học PHP.

Các bản vá lỗi cho PHP cũng như các sản phẩm từ PHP cũng rất nhanh chóng do có nguồn cộng đồng hỗ trợ đông đảo, mạnh mẽ. Tính chất mã nguồn mở giúp cho cộng đồng có thể sớm phát hiện các lỗi trong mã nguồn PHP để hoàn thiện và khắc phục.

### Ưu và nhược điểm của ngôn ngữ PHP

* Ưu điểm của ngôn ngữ PHP :
* Là mã nguồn mở nên người dùng có thể cài đặt, sử dụng PHP dễ dàng
* PHP là ngôn ngữ lập trình thân thiện, thích hợp với website
* Nhúng dễ dàng vào trang HTML
* Tương thích với nhiều ngôn ngữ/ trình duyệt web.
* Nhược điểm của ngôn ngữ PHP
* Chỉ hoạt động trên các ứng dụng web.
* Cấu trúc thường khá phức tạp và không đẹp mắt như ngôn ngữ lập trình khác.

### Kiểu dữ liệu trong PHP

Các kiểu dữ liệu được sử dụng để chứa các loại dữ liệu hoặc giá trị khác nhau của một biến nào đó. Các biến có thể lưu trữ dữ liệu thuộc các kiểu khác nhau và các kiểu dữ liệu khác nhau có các chức năng và nhiệm vụ khác nhau.

PHP hỗ trợ các kiểu dữ liệu sau:

* String
* Integer
* Float (double)
* Boolean
* Array
* Object
* NULL
* Resource

## Lý thuyết về mô hình MVC và Framework Laravel

### Framework là gì? PHP Framework là gì ?

Framework chính là một thư viện với các tài nguyên có sẵn cho từng lĩnh vực để lập trình viên sử dụng thay vì phải tự thiết kế. Có Framework, lập trình viên chỉ tìm hiểu và khai thác những tài nguyên đó, gắn kết chúng lại với nhau và hoàn chỉnh sản phẩm của mình. Đối với lập trình viên trong mỗi một lĩnh vực, họ cần phải xây dựng các lớp chương trình để xây dựng nên những phần mềm, ứng dụng thành phẩm.

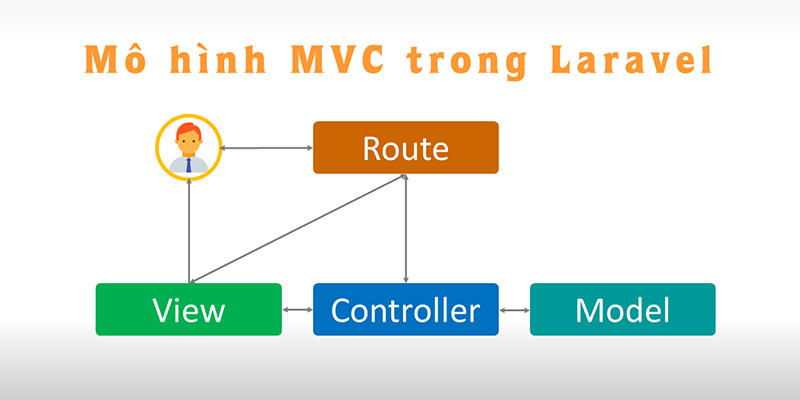
PHP framework là thư viện làm cho sự phát triển của những ứng dụng web viết bằng ngôn ngữ PHP trở nên trôi chảy hơn. Bằng cách cung cấp 1 cấu trúc cơ bản để xây dựng những ứng dụng đó. Hay nói cách khác, PHP framework giúp bạn thúc đẩy nhanh chóng quá trình phát triển ứng dụng. Giúp bạn tiết kiệm được thời gian, tăng sự ổn định cho ứng dụng. Giảm thiểu số lần phải viết lại code cho lập trình viên.

### Khái niệm Framework Laravel

Laravel là một PHP framework, có mã nguồn mở và miễn phí, được xây dựng nhằm hỗ trợ phát triển các phần mềm, ứng dụng, theo kiến trúc MVC. Hiện nay, Laravel đang là PHP framework phổ biến nhất và tốt nhất.

### Mô hình MVC trong Laravel

MVC (Model-View-Controller) là mẫu kiến trúc phần mềm trên máy tính nhằm mục đích tạo lập giao diện cho người dùng. Theo đó, hệ thống MVC được chia thành ba phần có khả năng tương tác với nhau và tách biệt các nguyên tắc nghiệp vụ với giao diện người dùng.



Hình 2.1. Mô hình MVC trong Laravel

Ba thành phần ấy bao gồm:

* Controller: Giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng.
* Model: Là thành phần chứa tất cả các nghiệp vụ logic, phương thức xử lý, truy xuất database, đối tượng mô tả dữ liệu như các Class, hàm xử lý…
* View: Đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images…

Bằng cách này, thông tin nội hàm được xử lý tách biệt với phần thông tin xuất hiện trong giao diện người dùng. Bảo đảm các nguyên tắc nghề nghiệp của lập trình viên.

### Sơ lược các tính năng cơ bản trong Laravel

Bundles : Ở laravel phiên bản 3.x, cung cấp một hệ thống đóng gói các module, với rất nhiều tính năng đi kèm.

Composer : Ở laravel phiên bản 4.x, được sử dụng như một công cụ quản lý với tính năng như thêm các gói cài đặt, các chức năng PHP phụ trợ cho Laravel có trong kho Packagist.

Eloquent ORM (object relation mapping) : ánh xạ các đối tượng và quan hệ cơ sở dữ liệu, cung cấp các phương thức nội bộ để thực thi đồng thời cũng bổ sung các tính năng hạn chế về mối quan hệ giữa các đối tượng cơ sở dữ liệu. Eloquent ORM trình bày các bảng trong cơ sở dữ liệu dưới dạng các lớp, cung cấp thêm lựa chọn truy cập cơ sở dữ liệu trực tiếp mới mẻ hơn, chuyên nghiệp hơn.

Application logic : Là một phần của phát triển ứng dụng, được sử dụng bởi bộ điều khiển controllers.

Routes : Định nghĩa mối quan hệ giữa các đường dẫn (url), các liên kết (link) . Khi một liên kết được tạo ra bằng cách sử dụng tên của routes, thì một định danh liên kết thống nhất sẽ được tạo ra bởi laravel.

Restful Controller : cung cấp các tùy chọn để tách các logic phía sau các request HTTP POST, GET

Class auto loading : cung cấp việc tải tự động các class trong PHP, mà không cần include các class vào. Tùy thuộc vào yêu cầu các class cần thiết sẽ được nạp vào, hạn chế các class không cần thiết.

View : chưa các mã html, hiển thị dữ liệu được chỉ định bởi controller

Migrations : cung cấp một hệ thống kiểm soát các phiên bản lược đồ cơ sở dữ liệu (database cheme), làm cho web ứng dụng có khả năng tương tác phù hợp những thay đổi logic, các đoạn mã code của ứng dụng và những thay đổi cần thiết trong việc bố trí cơ sở dữ liệu, triển khai nới lỏng và cập nhật các ứng dụng.

Unit Testing : đóng một vai trò quan trọng trong Laravel, Unit testting chứa rất nhiều các hệ thống unit testing, giúp phát hiện và ngăn chặn lỗi trong khuôn khổ nhất định. Unit Testing có thể được chạy thông qua tiện ích command-line.

Automatic pagination : Tính năng tự động phân trang được tích hợp vào Laravel giúp đơn giản hóa các nhiệm vụ thực hiện phân trang so với các phương pháp thông thường.

## Hệ quản trị cơ sở dữ liệu MySQL

### Khái niệm

MySQL là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. MySQL miễn phí hoàn toàn cho nên có thể tải về MySQL từ trang chủ. Nó có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,…

MySQL là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

MySQL được sử dụng cho việc bổ trợ PHP, Perl, và nhiều ngôn ngữ khác, nó làm nơi lưu trữ những thông tin trên các trang web viết bằng PHP hay Perl,…

### Tại sao sử dụng MySQL

MySQL là một hệ quản trị cơ sở dữ liệu thường dùng với ngôn ngữ PHP.

Tốc độ: MySQL rất nhanh. Những nhà phát triển cho rằng MySQL là cơ sở dữ liệu nhanh nhất có thể có.

Dễ sử dụng: MySQL tuy có tính năng cao nhưng thực sự là một hệ thống cơ sở dữ liệu rất đơn giản và ít phức tạp khi cài đặt và quản trị hơn các hệ thống lớn .

Giá thành: MySQL là miễn phí cho hầu hết các việc sử dụng trong một tổ chức.

Hỗ trợ ngôn ngữ truy vấn: MySQL hiểu SQL, là ngôn ngữ của sự chọn lựa cho tất cả các hệ thống cơ sở dữ liệu hiện đại. Có thể truy cập MySQL bằng cách sử dụng các ứng dụng mà hỗ trợ ODBC (Open Database Connectivity -một giao thức giao tiếp cơ sở dữ liệu được phát triển bởi Microsoft).

Năng lực: Nhiều client có thể truy cập đến server trong cùng một thời gian. Các client có thể sử dụng nhiều cơ sở dữ liệu một cách đồng thời. Có thể truy cập MySQL tương tác với sử dụng một vài giao diện để có thể đưa vào các truy vấn và xem các kết quả: các dòng yêu cầu của khách hàng, các trình duyệt Web…

Kết nối và bảo mật: MySQL được nối mạng một cách đầy đủ, các cơ sở dữ liệu có thể được truy cập từ bất kỳ nơi nào trên Internet do đó có thể chia sẽ dữ liệu với bất kỳ ai, bất kỳ nơi nào. Nhưng MySQL kiểm soát quyền truy cập cho nên người mà không nên nhìn thấy dữ liệu thì không thể nhìn được.

Tính linh động: MySQL chạy trên nhiều hệ thống UNIX cũng như không phải UNIX chẳng hạn như Windows hay OS/2. MySQL chạy được các với mọi phần cứng từ các máy PC ở nhà cho đến các máy server.

Sự phân phối rộng: MySQL rất dễ dàng đạt được, chỉ cần sử dụng trình duyệt web . Nếu không hiểu làm thế nào mà nó làm việc hay tò mò về thuật toán, có thể lấy mã nguồn và tìm tòi nó.

Sự hỗ trợ: Có thể tìm thấy các tài nguyên có sẵn mà MySQL hỗ trợ. Cộng đồng MySQL rất có trách nhiệm. Họ trả lời các câu hỏi trên mailing list thường chỉ trong vài phút. Khi lỗi được phát hiện, các nhà phát triển sẽ đưa ra cách khắc phục trong vài ngày, thậm chí có khi trong vài giờ và cách khắc phục đó sẽ ngay lập tức có sẵn trên Internet.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Tổng quan về hệ thống quản lý đào tạo

### Mô tả hệ thống hiện tại

Hệ thống quản lý đào tạo của trường là một ứng dụng web được xây dựng bằng Laravel, với mục tiêu chính là quản lý các hoạt động liên quan đến quá trình đào tạo và quản lý của phòng đào tạo. Hệ thống này tích hợp nhiều phân hệ quan trọng như quản lý đồ án, thanh tra-pháp chế, công tác sinh viên và cơ sở vật chất.

### Những hạn chế của hệ thống

Các hạn chế và vấn đề của hệ thống hiện tại bao gồm:

* Không phân tách rõ ràng: Hệ thống quản lý đồ án đang được tích hợp chung với các phân hệ khác, gây khó khăn trong việc quản lý và phát triển các phân hệ độc lập. Điều này có thể dẫn đến sự phức tạp và không linh hoạt trong việc mở rộng và cải thiện từng phân hệ riêng biệt.
* Hiệu suất và tốc độ: Do sự kết hợp của nhiều phân hệ trên cùng một website, hệ thống có thể gặp vấn đề về hiệu suất và tốc độ xử lý. Sự tải trọng lớn từ các phân hệ khác nhau có thể ảnh hưởng đến trải nghiệm sử dụng và thời gian phản hồi của hệ thống.
* Quản lý dữ liệu không hiệu quả: Việc quản lý dữ liệu liên quan đến đồ án, thanh tra-pháp chế, công tác sinh viên và cơ sở vật chất trên cùng một hệ thống có thể gây khó khăn trong việc tra cứu, tìm kiếm và xử lý thông tin. Các dữ liệu có thể bị trùng lặp hoặc không được tổ chức một cách hiệu quả.
* Thiếu tính linh hoạt và tuỳ chỉnh: Hệ thống hiện tại có thể không cung cấp đủ tính linh hoạt và khả năng tuỳ chỉnh cho từng phân hệ độc lập. Việc thay đổi và cải thiện các chức năng và quy trình riêng của từng phân hệ có thể gặp khó khăn.

## Phân tích và thiết kế hệ thống quản lý đồ án

### Yêu cầu chức năng

Theo yêu cầu, hệ thống quản lý đồ án có thể bao gồm các chức năng chính sau :

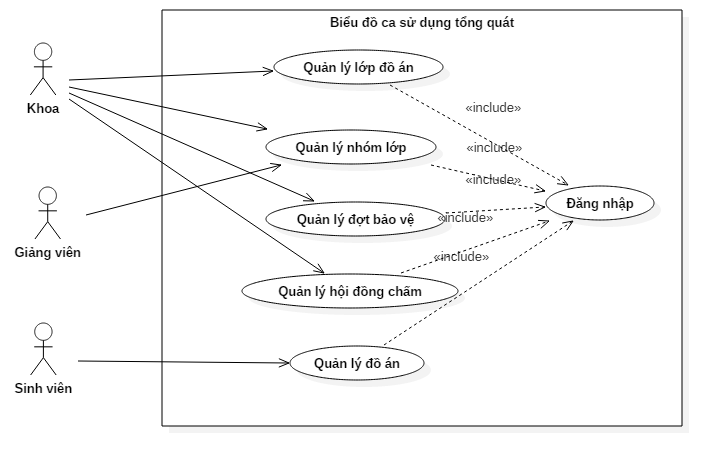
* Vai trò quản trị:
  + Tạo lớp con: Quản trị viên có khả năng tạo lớp con để chia nhỏ các khóa học hoặc kỳ học thành các phần nhỏ hơn để dễ dàng quản lý.
  + Thiết lập thời gian nộp đề cương và báo cáo: Quản trị viên có thể xác định thời gian chính xác cho việc nộp đề cương và báo cáo của sinh viên, đảm bảo sự tuân thủ thời hạn.
  + Upload file hướng dẫn: Quản trị viên có thể tải lên các tài liệu hướng dẫn liên quan đến quá trình thực hiện đồ án, để hỗ trợ cả giảng viên và sinh viên.
  + Thêm định hướng đề tài: Quản trị viên có thể cung cấp các định hướng cho đề tài để hỗ trợ sinh viên trong việc lựa chọn và định hướng đồ án của họ.
  + Thiết lập thời gian tạo nhóm: Quản trị viên quyết định thời gian cho phép sinh viên tạo nhóm, giúp đảm bảo việc hình thành nhóm diễn ra đúng thời hạn.
  + Quản lý đợt bảo vệ: Quản trị viên tổ chức và quản lý các đợt bảo vệ đồ án, đảm bảo việc tổ chức diễn ra suôn sẻ và hiệu quả.
  + Quản lý hội đồng chấm: Quản trị viên cân nhắc và phân công thành viên cho hội đồng chấm, đảm bảo việc chấm điểm được thực hiện một cách công bằng và chính xác.
  + Phân sinh viên vào hội đồng: Quản trị viên phân công sinh viên vào các hội đồng chấm, đảm bảo sự đa dạng và chất lượng trong hội đồng.
* Vai trò giảng viên:
  + Vai trò "Giảng viên" trong hệ thống đồ án có nhiệm vụ hướng dẫn và đánh giá tiến độ của sinh viên trong quá trình thực hiện đồ án. Giảng viên có khả năng thực hiện những công việc sau:
  + Xem danh sách sinh viên: Giảng viên có thể xem danh sách sinh viên được giao hướng dẫn trong lớp đồ án của mình.
  + Phê duyệt đề cương và báo cáo: Giảng viên xem và phê duyệt các đề cương và báo cáo mà sinh viên của mình nộp, đảm bảo chất lượng và đáp ứng yêu cầu.
  + Phân công hội đồng bảo vệ: Giảng viên thực hiện việc phân công thành viên cho hội đồng bảo vệ đồ án, để bảo đảm sự đa dạng và uy tín trong quá trình chấm điểm.
  + Cho điểm và đánh giá: Giảng viên gán điểm và đánh giá bài thuyết trình của sinh viên trong quá trình bảo vệ đồ án, dựa trên kiến thức và hiểu biết cá nhân.
  + Gửi mail cho nhóm sinh viên: Giảng viên có khả năng gửi email thông báo cho toàn bộ lớp hoặc gửi email cho từng nhóm sinh viên để thông báo về các vấn đề quan trọng.

### Yêu cầu phi chức năng

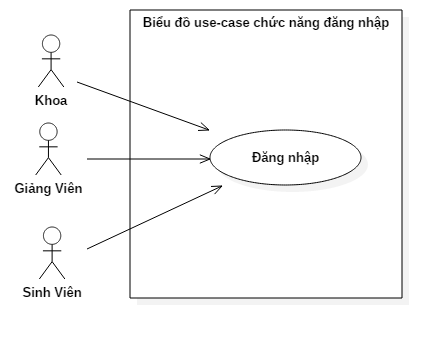
* Độ bảo mật cao: hệ thống cần đảm bảo tính bảo mật cao đối với dữ liệu cá nhân của sinh viên và giảng viên, đồ án, điểm số và thông tin quản lý.
* Yêu cầu cơ chế xác thực và ủy quyền mạnh mẽ để đảm bảo chỉ những người có quyền truy cập mới có thể truy cập vào các chức năng cụ thể.
* Khả năng mở rộng: hệ thống cần thiết kế sao cho dễ dàng mở rộng, có thể tích hợp thêm các phân hệ và chức năng mới trong tương lai mà không gây ra sự cố về hiệu suất hoặc tương tác không mong muốn.
* Hiệu suất ổn định: hệ thống cần đảm bảo hiệu suất ổn định, đặc biệt trong các tình huống tải cao như khi nhiều sinh viên truy cập cùng lúc hoặc khi thực hiện bảo vệ đồ án đồng thời.
* Giao diện người dùng thân thiện: Giao diện người dùng cần được thiết kế đơn giản, dễ sử dụng và thân thiện với người dùng, đảm bảo các vai trò khác nhau có thể tương tác một cách trực quan và hiệu quả.
* Hỗ trợ đa nền tảng: Hệ thống cần tương thích với nhiều nền tảng khác nhau như máy tính, điện thoại di động và máy tính bảng để người dùng có thể truy cập mọi lúc, mọi nơi.
* Bảo trì và quản lý dễ dàng: Hệ thống cần cung cấp các công cụ quản lý và bảo trì để giúp quản trị viên dễ dàng giám sát, sửa chữa và cập nhật hệ thống.
* Tích hợp email thông báo: Hệ thống cần có khả năng gửi email thông báo cho người dùng về các sự kiện quan trọng như thời hạn nộp bài, lịch bảo vệ và các thông báo quan trọng khác.
* Sự ổn định và sẵn sàng hoạt động liên tục: Hệ thống cần hoạt động liên tục 24/7 mà không bị gián đoạn quá lâu, đảm bảo người dùng có thể truy cập và thực hiện công việc bất kỳ lúc nào.
* Hỗ trợ kỹ thuật và hướng dẫn sử dụng: Hệ thống cần cung cấp kênh hỗ trợ kỹ thuật và tài liệu hướng dẫn chi tiết để người dùng có thể giải quyết vấn đề một cách dễ dàng và nhanh chóng.
* Sự bảo đảm về dữ liệu: Hệ thống cần đảm bảo tính toàn vẹn và an toàn cho dữ liệu, bao gồm việc thực hiện các sao lưu định kỳ và có kế hoạch phục hồi dữ liệu khi cần thiết.

## Biểu đồ mô hình hóa UML

### Biểu đồ ca sử dụng



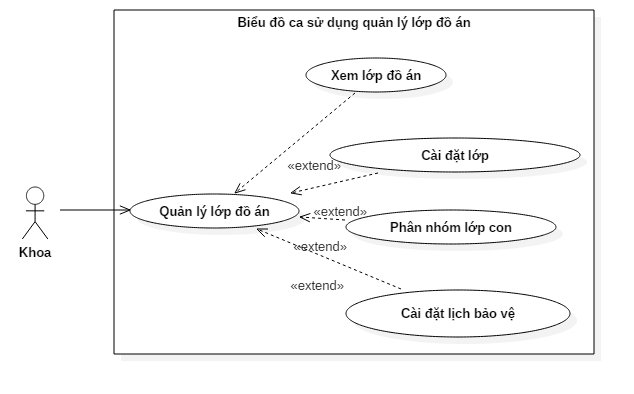
Hình 3.1. Biểu đồ use-case tổng quát



Hình 3.2. Biểu đồ use-case chức năng đăng nhập

Bảng 3.1. Mô tả use-case đăng nhập

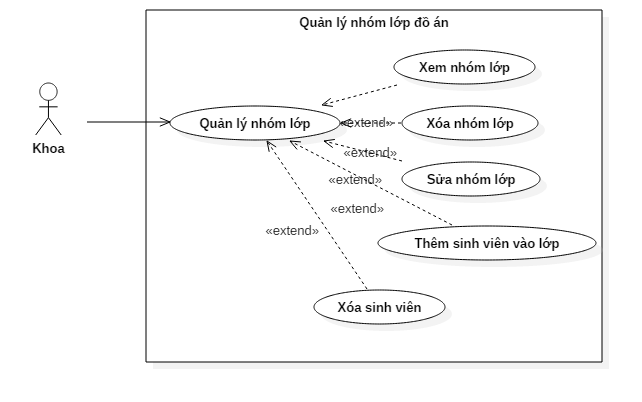
|  |  |
| --- | --- |
| Tên use-case | Đăng nhập |
| Actor | Khoa, Giảng viên, Sinh viên |
| Mô tả | Use-case cho phép người dùng đăng nhập tài khoản trên hệ thống |
| Điều kiện kích hoạt | Khi người dùng muốn đăng nhập vào hệ thống |
| Tiền điều kiện | Tài khoản email của tác nhân đăng nhập tồn tại trong hệ thống |
| Hậu điều kiện | Thông báo đăng nhập thành công và chuyển hướng đến trang chức năng tương ứng |
| Luồng sự kiện chính | 1. Hệ thống yêu cầu cung cấp email đăng nhập 2. Người dùng nhập email và nhấn nút lấy mã đăng nhập 3. Hệ thống kiểm tra email và gửi mã đăng nhập đến cho người dùng 4. Người dùng nhập mã đăng nhập và nhấn đăng nhập 5. Hệ thống kiểm tra mã đăng nhập và chuyển hướng người dùng đến trang phù hợp dựa vào vai trò |
| Luồng sự kiện rẽ nhánh | Nếu email không tồn tại trong hệ thống hoặc mã đăng nhập không hợp lệ thì hiển thị thông báo lỗi và yêu cầu người dùng nhập lại. |



Hình 3.3. Biểu đồ use-case quản lý lớp đồ án của Khoa

Bảng 3.2. Mô tả use-case quản lý lớp đồ án

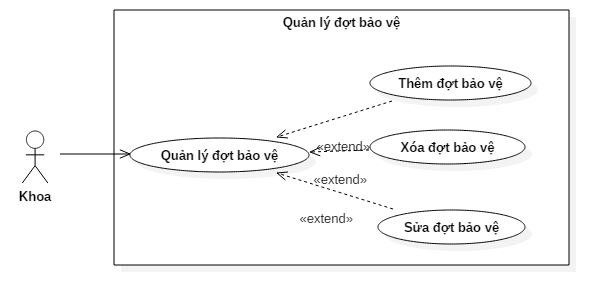
|  |  |
| --- | --- |
| Tên use-case | Quản lý lớp đồ án |
| Actor | Khoa |
| Mô tả | Use-case cho phép Khoa thực hiện quản lý lớp đồ án đã được tạo bởi Phòng Đào tạo |
| Điều kiện kích hoạt | Khoa muốn quản lý lớp đồ án |
| Tiền điều kiện | Khoa đã đăng nhập vào hệ thống |
| Hậu điều kiện | Dữ liệu lớp đồ án được cập nhật |
| Luồng sự kiện chính | 1. Khoa chọn chức năng quản lý lớp đồ án/đề án 2. Hệ thống hiển thị danh sách lớp đồ án 3. Khoa chọn chức năng cài đặt, phân nhóm lớp con hoặc cài đặt lịch bảo vệ 4. Hệ thống hiển thị giao diện quản lý tương ứng 5. Khoa thực hiện cung cấp thông tin về lớp đồ án 6. Hệ thống kiểm tra thông tin và cập nhật dữ liệu |
| Luồng sự kiện rẽ nhánh | Nếu thông tin cung cấp không hợp lệ thì hiển thị thông báo lỗi |



Hình 3.4. Biểu đồ use-case quản lý nhóm lớp đồ án của Khoa

Bảng 3.3. Mô tả use-case quản lý nhóm lớp đồ án

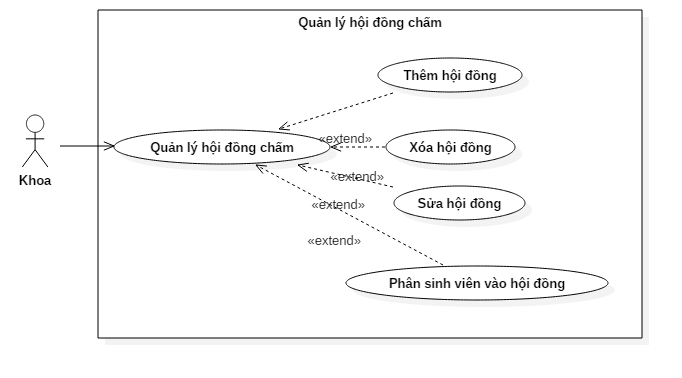
|  |  |
| --- | --- |
| Tên use-case | Quản lý nhóm lớp đồ án |
| Actor | Khoa |
| Mô tả | Use-case cho phép Khoa quản lý nhóm lớp được phân từ lớp đồ án chính |
| Điều kiện kích hoạt | Khoa muốn quản lý nhóm lớp đồ án |
| Tiền điều kiện | Khoa đã đăng nhập vào hệ thống |
| Hậu điều kiện | Dữ liệu nhóm lớp đồ án được cập nhật |
| Luồng sự kiện chính | 1. Khoa chọn chức năng quản lý lớp đồ án/đề án 2. Hệ thống hiển thị danh sách lớp đồ án 3. Tại mục thao tác trên nhóm lớp gốc, Khoa chọn xem nhóm lớp 4. Hệ thống hiển thị danh sách nhóm lớp theo lớp đồ án chính tương ứng 5. Khoa lựa chọn chức năng quản lý nhóm lớp mong muốn và cung cấp thông tin 6. Hệ thống kiểm tra thông tin và cập nhật dữ liệu |
| Luồng sự kiện rẽ nhánh | Nếu thông tin cung cấp không hợp lệ thì hiển thị thông báo lỗi |



Hình 3.5. Biểu đồ use-case quản lý đợt bảo vệ của Khoa

Bảng 3.4. Mô tả use-case quản lý đợt bảo vệ

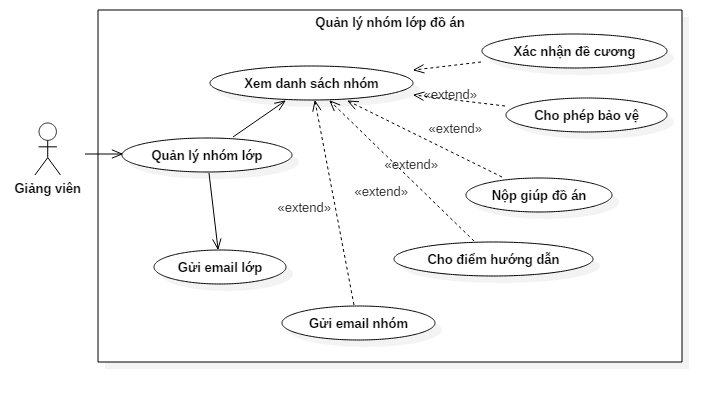
|  |  |
| --- | --- |
| Tên use-case | Quản lý đợt bảo vệ |
| Actor | Khoa |
| Mô tả | Use-case cho phép Khoa quản lý đợt bảo vệ |
| Điều kiện kích hoạt | Khoa muốn quản lý đợt bảo vệ |
| Tiền điều kiện | Khoa đã đăng nhập vào hệ thống |
| Hậu điều kiện | Dữ liệu nhóm lớp đồ án được cập nhật |
| Luồng sự kiện chính | 1. Khoa chọn chức năng quản lý đợt bảo vệ 2. Hệ thống hiển thị danh sách đợt bảo vệ 3. Khoa chọn chức năng thêm, xóa, sửa đợt bảo vệ và cung cấp thông tin 4. Hệ thống kiểm tra và thực hiện cấp nhật |
| Luồng sự kiện rẽ nhánh | Nếu thông tin cung cấp không hợp lệ thì hiển thị thông báo lỗi |



Hình 3.6. Biểu đồ use-case quản lý hội đồng chấm của Khoa

Bảng 3.5. Mô tả use-case quản lý hội dồng chấm

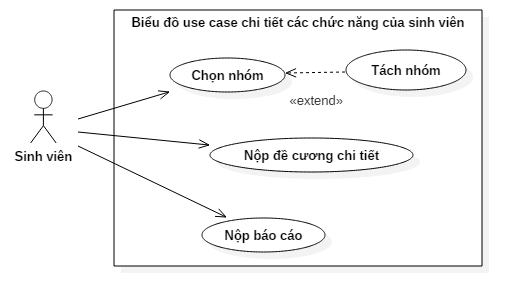
|  |  |
| --- | --- |
| Tên use-case | Quản lý hội đồng chấm |
| Actor | Khoa |
| Mô tả | Use-case cho phép Khoa quản lý hội đồng chấm |
| Điều kiện kích hoạt | Khoa muốn quản lý đợt bảo vệ |
| Tiền điều kiện | Khoa đã đăng nhập vào hệ thống |
| Hậu điều kiện | Dữ liệu nhóm lớp đồ án được cập nhật |
| Luồng sự kiện chính | 1. Khoa chọn chức năng quản lý đợt bảo vệ 2. Hệ thống hiển thị danh sách đợt bảo vệ 3. Khoa chọn chức năng thêm, xóa, sửa đợt bảo vệ và cung cấp thông tin 4. Hệ thống kiểm tra và thực hiện cấp nhật |
| Luồng sự kiện rẽ nhánh | Nếu thông tin cung cấp không hợp lệ thì hiển thị thông báo lỗi |

****

Hình 3.7. Biểu đồ use-case chi tiết các chức năng của Giảng Viên

Bảng 3.6. Mô tả use-case các chức năng của Giảng viên

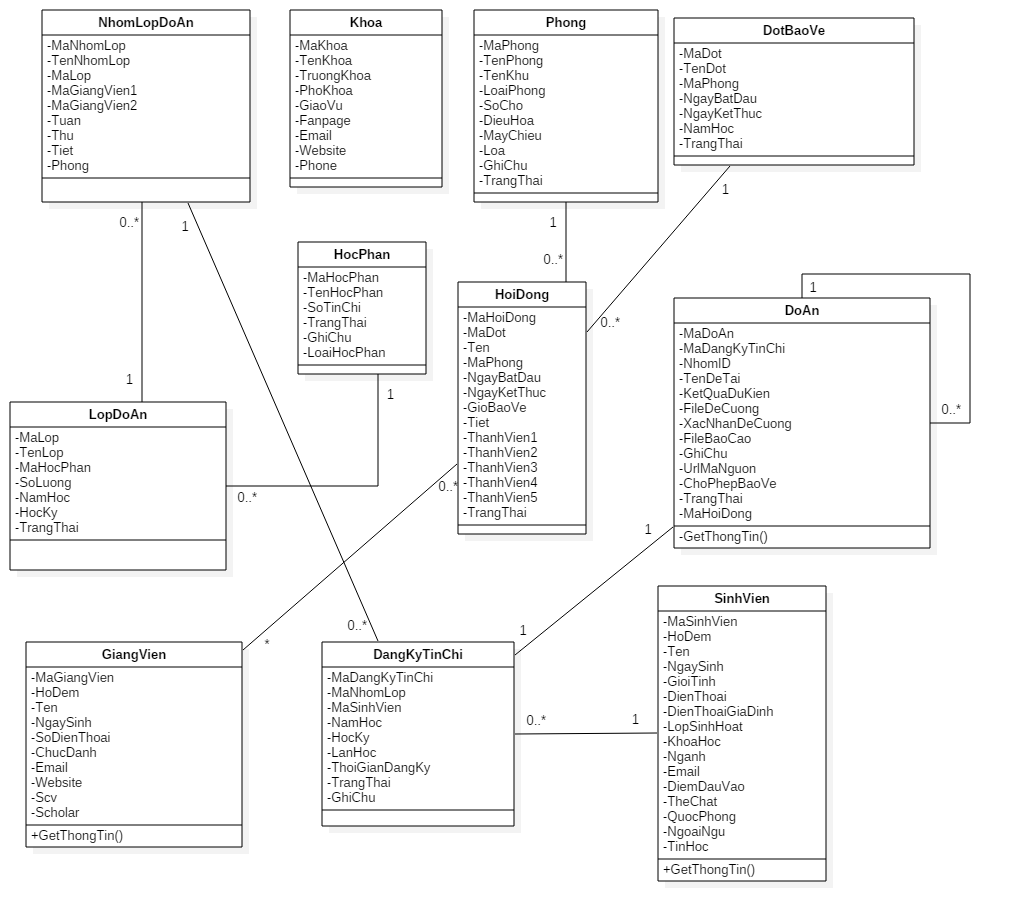
|  |  |
| --- | --- |
| Tên use-case | Các chức năng của Giảng viên |
| Actor | Giảng viên |
| Mô tả | Use-case cho phép thực hiện các chức năng của Giảng viên |
| Điều kiện kích hoạt | Giảng viên muốn thực hiện các chức năng của mình |
| Tiền điều kiện | Giảng viên đã đăng nhập vào hệ thống |
| Hậu điều kiện | Dữ liệu nhóm lớp đồ án được cập nhật |
| Luồng sự kiện chính | 1. Giảng viên chọn chức năng quản lý nhóm lớp đồ án 2. Hệ thống hiển thị danh sách nhóm đồ án 3. Giảng viên chọn chức năng cần thao tác và cung cấp thông tin phù hợp 4. Hệ thống kiểm tra và thực hiện cấp nhật |
| Luồng sự kiện rẽ nhánh | Nếu thông tin cung cấp không hợp lệ thì hiển thị thông báo lỗi |

****

Hình 3.8. Biểu đồ use-case chi tiết các chức năng của sinh viên

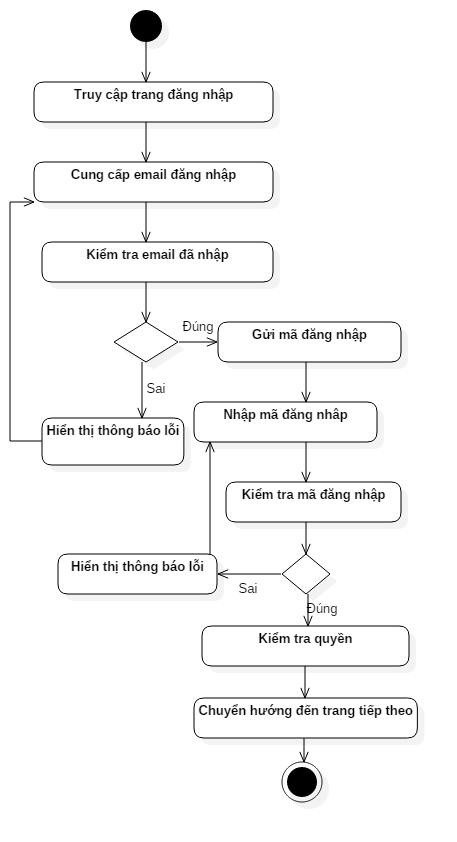
|  |  |
| --- | --- |
| Tên use-case | Các chức năng của Sinh viên |
| Actor | Sinh viên |
| Mô tả | Use-case cho phép thực hiện các chức năng của Sinh viên |
| Điều kiện kích hoạt | Sinh viên muốn thực hiện các chức năng của mình |
| Tiền điều kiện | Sinh viên đã đăng nhập vào hệ thống |
| Hậu điều kiện | Dữ liệu nhóm lớp đồ án được cập nhật |
| Luồng sự kiện chính | 1. Sinh viên truy cập trang danh sách đồ án 2. Hệ thống hiển thị danh sách đồ án của sinh viên 3. Sinh viên chọn chức năng cần thao tác như chọn nhóm, nộp đề cương chi tiết, nộp báo cáo 4. Hệ thống kiểm tra và thực hiện cấp nhật |
| Luồng sự kiện rẽ nhánh | Nếu thông tin cung cấp không hợp lệ thì hiển thị thông báo lỗi |

### Biểu đồ lớp

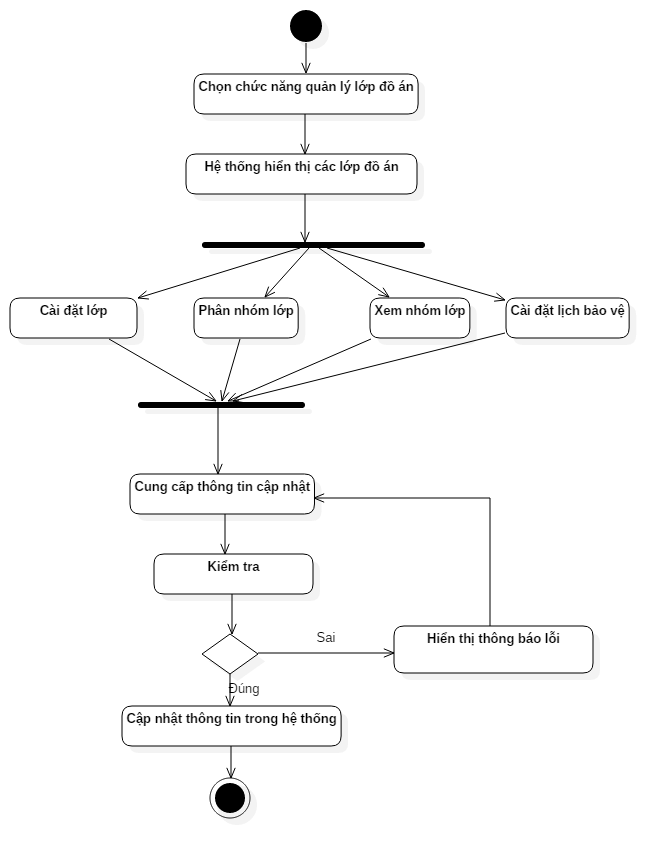


Hình 3.9. Biểu đồ lớp

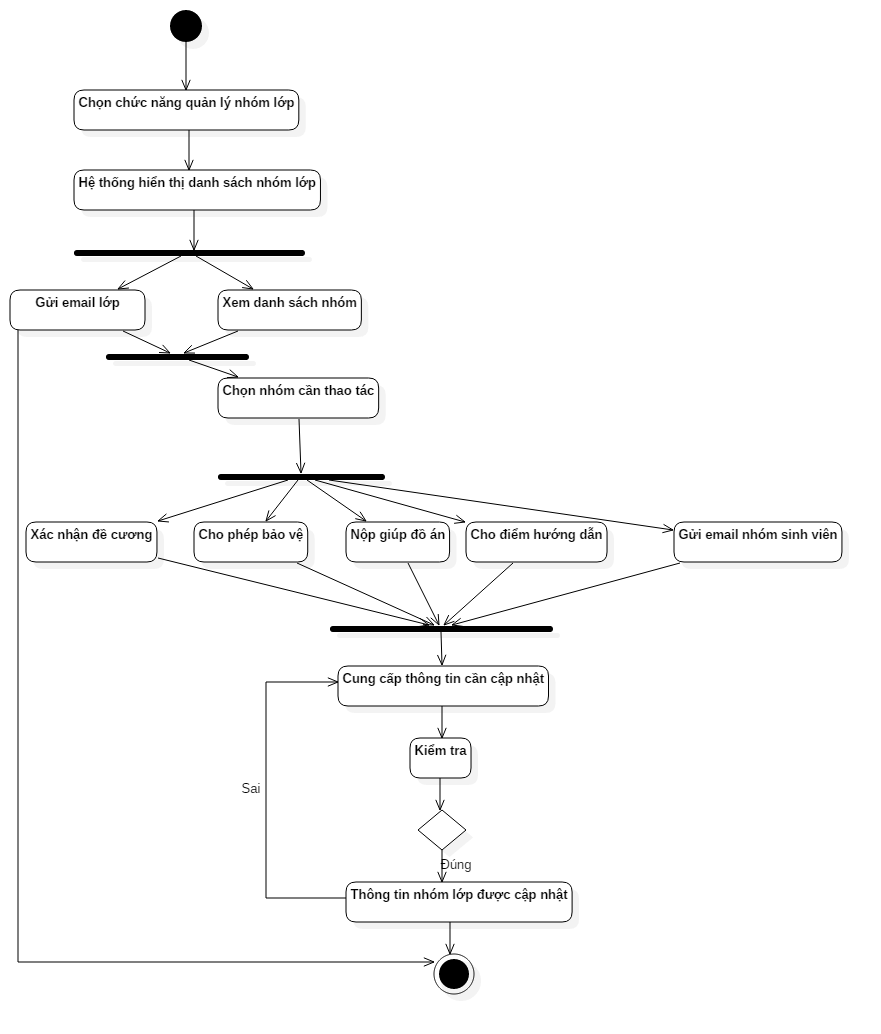
### Biểu đồ hoạt động



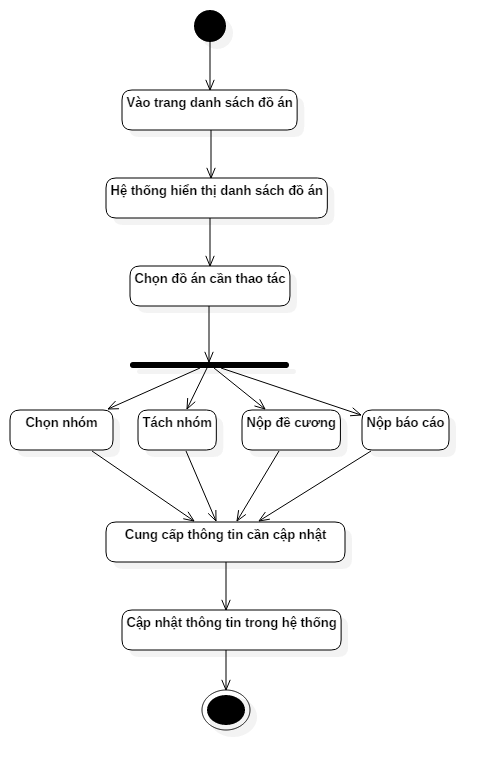
Hình 3.10. Biểu đồ hoạt động chức năng đăng nhập



Hình 3.11. Biểu đồ hoạt động chức năng của Khoa

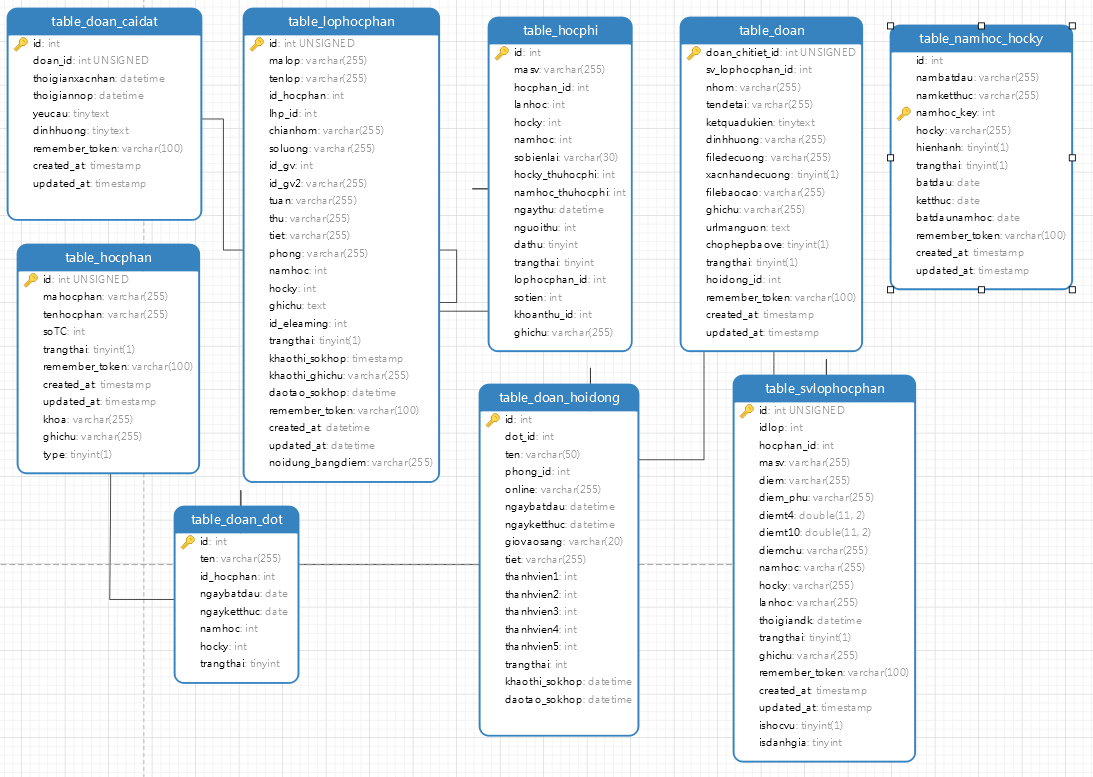


Hình 3.12. Biểu đồ hoạt động chức năng của Giảng viên



Hình 3.13. Biểu đồ hoạt dộng các chức năng của Sinh viên

### Thiết kế cơ sở dữ liệu



Hình 3.14. Mô hình các bảng cơ sở dữ liệu

# TRIỂN KHAI XÂY DỰNG HỆ THỐNG

## Khởi tạo môi trường xây dựng

Trước khi bắt đầu xây dựng hệ thống quản lý đồ án bằng Laravel, MySQL và Visual Studio Code, cần thực hiện một số bước để khởi tạo môi trường phát triển.

Bước 1: Cài Đặt Composer

Composer là một công cụ quản lý gói PHP cần thiết để cài đặt Laravel. Thực hiện các bước sau để cài đặt Composer:

Truy cập trang web chính thức của Composer tại https://getcomposer.org/.

Tải Composer và làm theo hướng dẫn cài đặt cho hệ điều hành của bạn.

Bước 2: Cài Đặt Laravel

Sau khi cài đặt Composer, bạn có thể cài đặt Laravel Framework bằng cách thực hiện các bước sau:

Mở cửa sổ terminal hoặc dòng lệnh.

Chạy lệnh sau để cài đặt Laravel:

composer global require laravel/installer

Bước 3: Tạo Dự Án Laravel Mới

Sau khi cài đặt Laravel, bạn có thể tạo một dự án Laravel mới bằng lệnh sau (thay <project-name> bằng tên dự án của bạn):

laravel new <project-name>

Cài Đặt MySQL và Sử Dụng Navicat

Bước 4: Cài Đặt MySQL

Truy cập trang web chính thức của MySQL tại

https://dev.mysql.com/downloads/installer/.

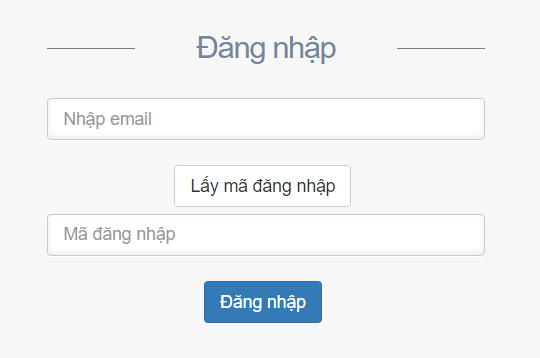
Tải và cài đặt MySQL theo hướng dẫn của trang web.

Bước 5: Sử Dụng Navicat

Navicat là một công cụ quản lý cơ sở dữ liệu MySQL mạnh mẽ và dễ sử dụng. Có thể tải và cài đặt Navicat từ trang web chính thức. Sau khi cài đặt xong, có thể sử dụng Navicat để kết nối và quản lý cơ sở dữ liệu MySQL của dự án Laravel. Navicat cung cấp giao diện đồ họa và nhiều công cụ hữu ích để thực hiện các tác vụ quản lý cơ sở dữ liệu.

## Kết quả xây dựng chương trình

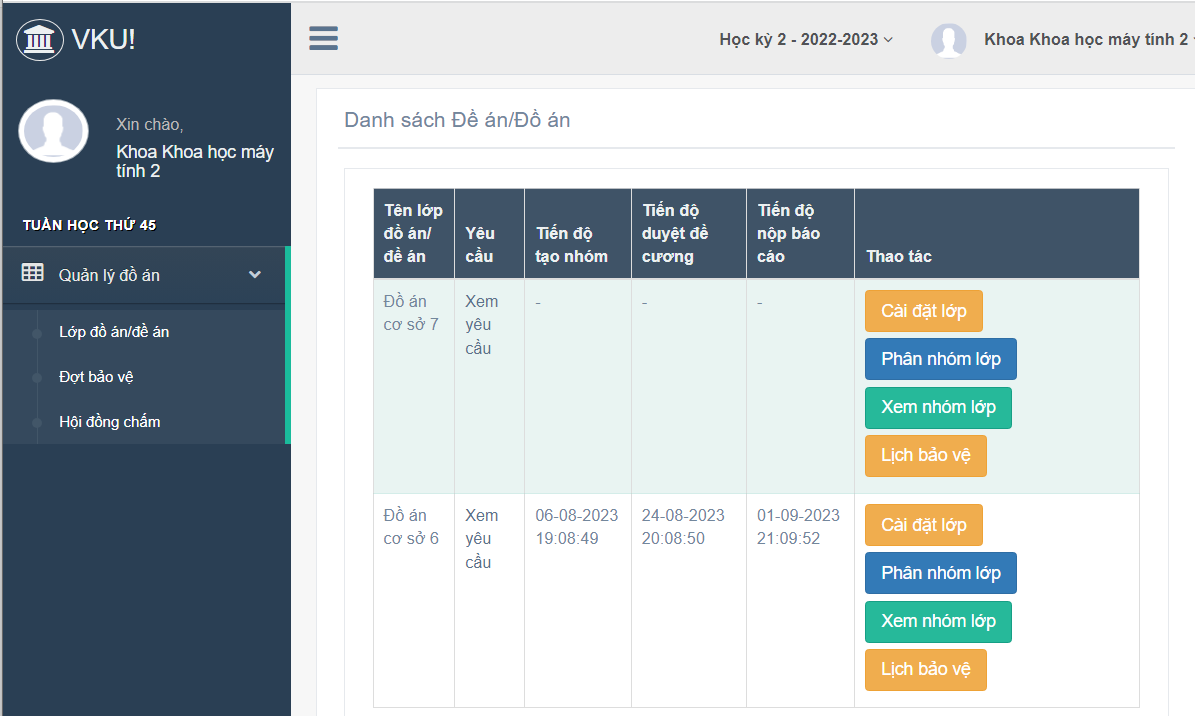
### Chức năng đăng nhập



Hình 4.1. Giao diện đăng nhập

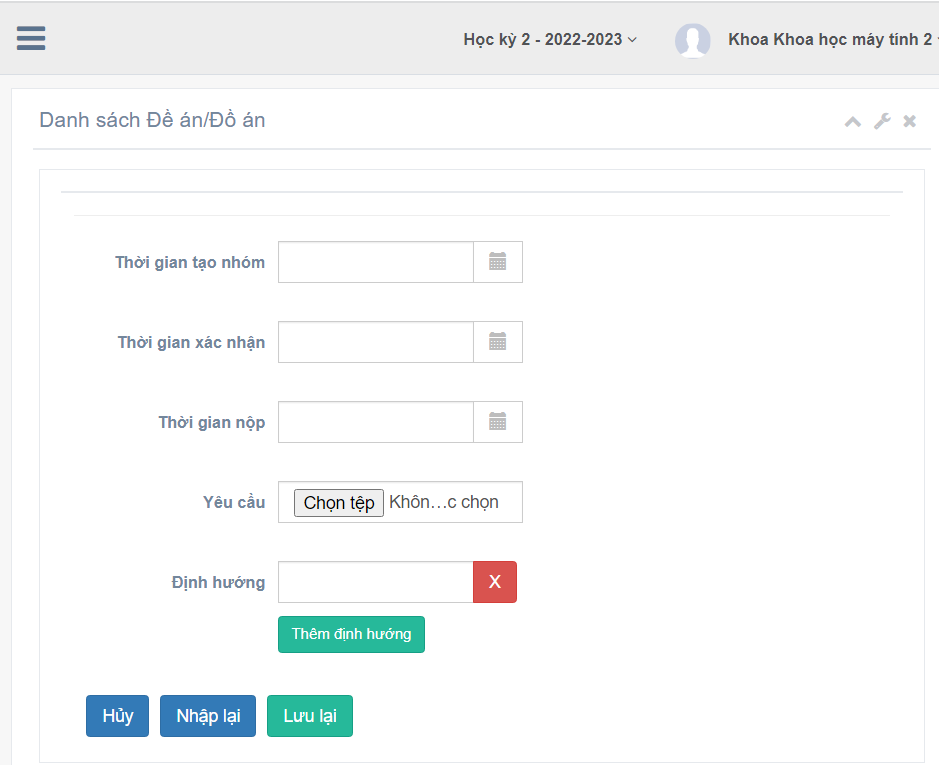
* Trên màn hình chính của ứng dụng, hiển thị "Email" dành cho người dùng nhập địa chỉ email của họ. Điều này giúp hệ thống xác định người dùng cụ thể.
* Sau khi người dùng nhập email của mình vào trường "Email" và nhấn Enter hoặc nút "Gửi mã đăng nhập", hệ thống sẽ bắt đầu quá trình xác minh.
* Hệ thống sau đó sẽ tự động tạo một Mã Đăng nhập (OTP - One-Time Password) ngẫu nhiên.
* Mã Đăng nhập này sẽ được gửi đến địa chỉ email mà người dùng đã cung cấp trước đó. Hệ thống sẽ gửi một email thông báo đến địa chỉ đó với mã này, cùng với hướng dẫn cho người dùng.
* Người dùng sau đó sẽ truy cập hộp thư email của họ, kiểm tra email từ hệ thống, và lấy Mã Đăng nhập được gửi trong email đó.
* Người dùng quay lại màn hình Đăng nhập của ứng dụng và nhập Mã Đăng nhập vào một trường "Mã Đăng nhập" hoặc "Mã OTP".
* Sau khi nhập mã, người dùng nhấn nút "Đăng nhập" hoặc "Xác minh" để hoàn tất quá trình đăng nhập.
* Hệ thống sẽ kiểm tra mã đăng nhập nhập vào và đối chiếu nó với mã đã gửi đi. Nếu mã đúng, người dùng sẽ được chuyển đến màn hình chính hoặc màn hình làm việc trong hệ
* thống quản lý đồ án theo từng quyền Khoa, Giảng Viên hoặc Sinh Viên dựa vào email đã cung cấp.

### Các chức năng của Khoa



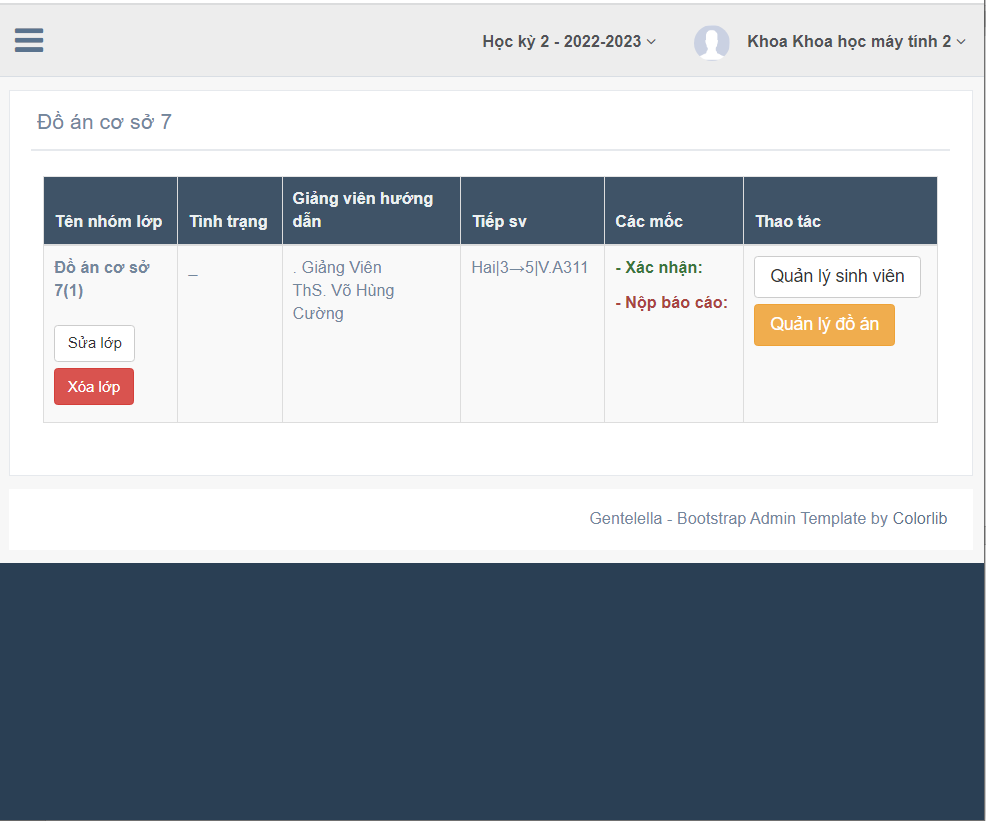
Hình 4.2. Giao diện xem danh sách lớp đồ án

Giao diện hiển thị thông tin về danh sách các lớp đồ án do phòng đào tạo đã phân chia. Dựa vào các lớp đồ án gốc, Khoa thực hiện cài đặt lớp, phân nhóm lớp, xem chi tiết nhóm lớp và cài đặt lịch bảo vệ đồ án.



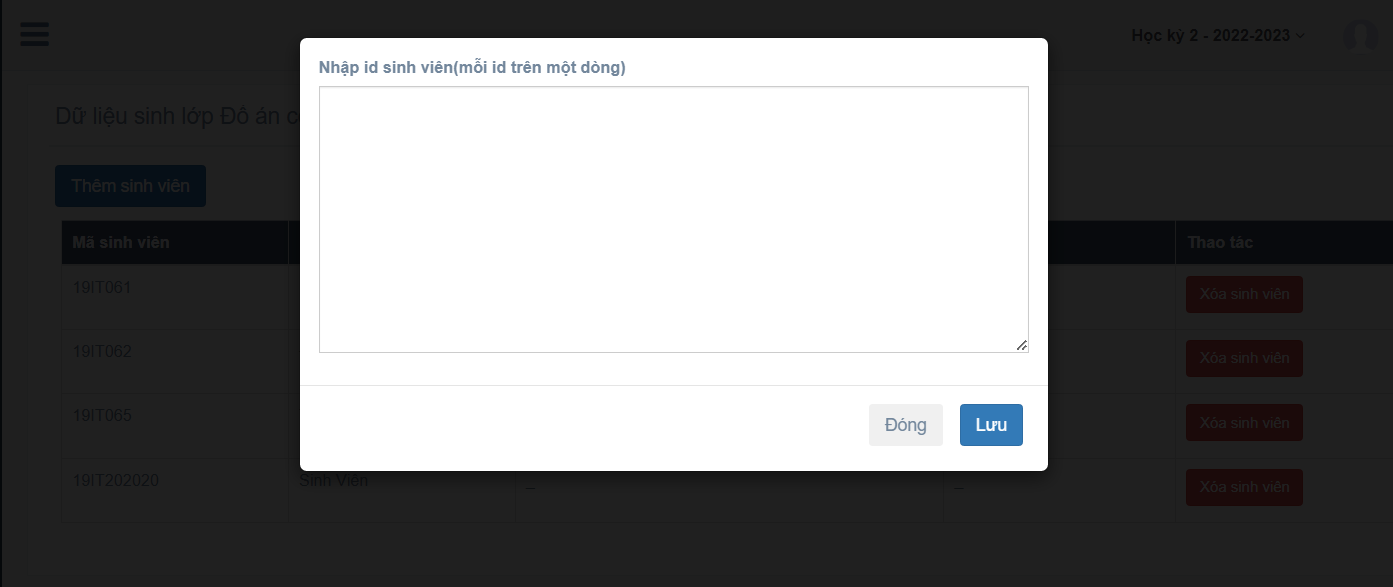
Hình 4.3.Giao diện cài đặt lớp đồ án

Giao diện cài đặt lớp đồ án hiển thị form giúp Khoa cập nhật thông tin thời gian tạo nhóm, thời gian xác nhận đề cương chi tiết, thời gian nộp báo cáo, cập nhật file yêu cầu đồ án, định hướng đề tài.



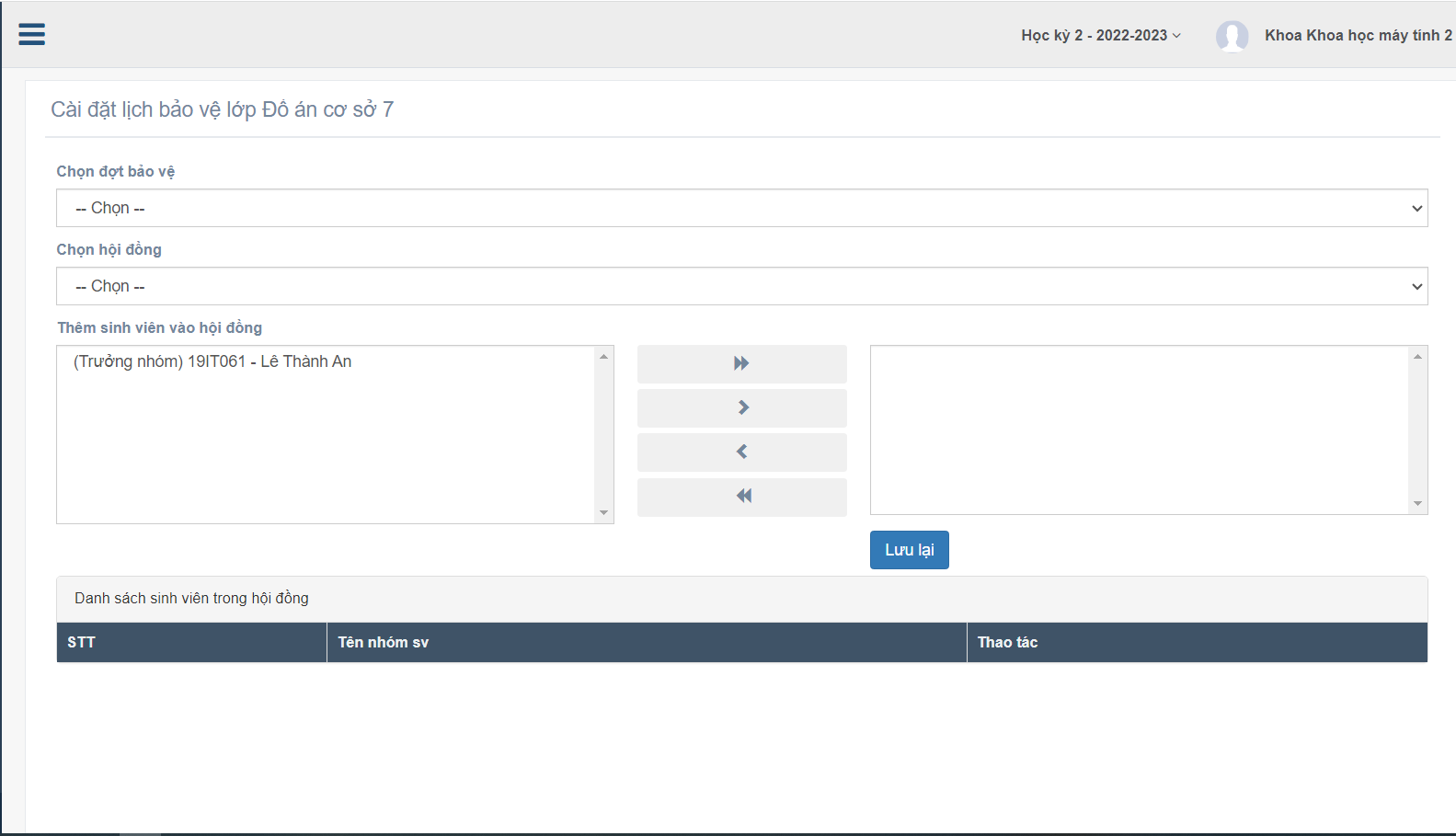
Hình 4.4. Giao diện quản lý nhóm lớp đồ án

Giao diện quản lý nhóm lớp đồ án hiển thị danh sách thông tin các nhóm lớp đã được phân chia từ lớp gốc và Khoa có thể lựa chọn xóa lớp, sửa lớp hoặc quản lý danh sách sinh viên thuộc lớp.



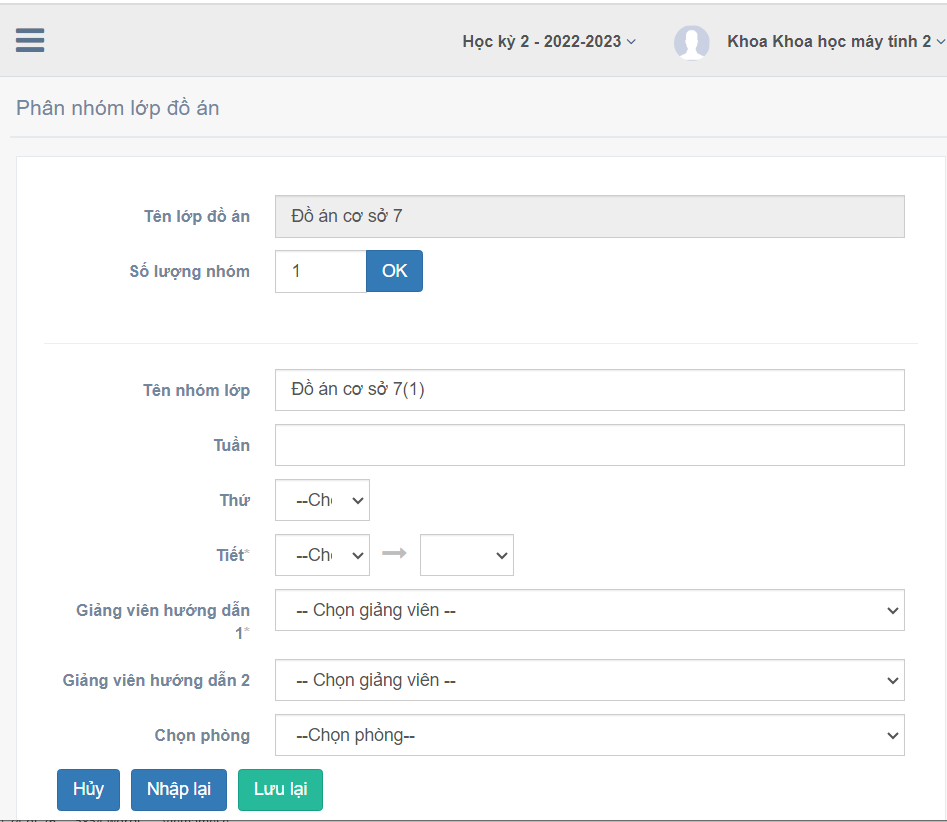
Hình 4.5. Giao diện phân sinh viên vào hội đồng bảo vệ

Ở giao diện phân sinh viên vào hội đồng bảo vệ, Khoa thực hiện nhập danh sách mã sinh viên nằm trên mỗi dòng dựa vào danh sách sinh viên đã đăng ký đồ án.



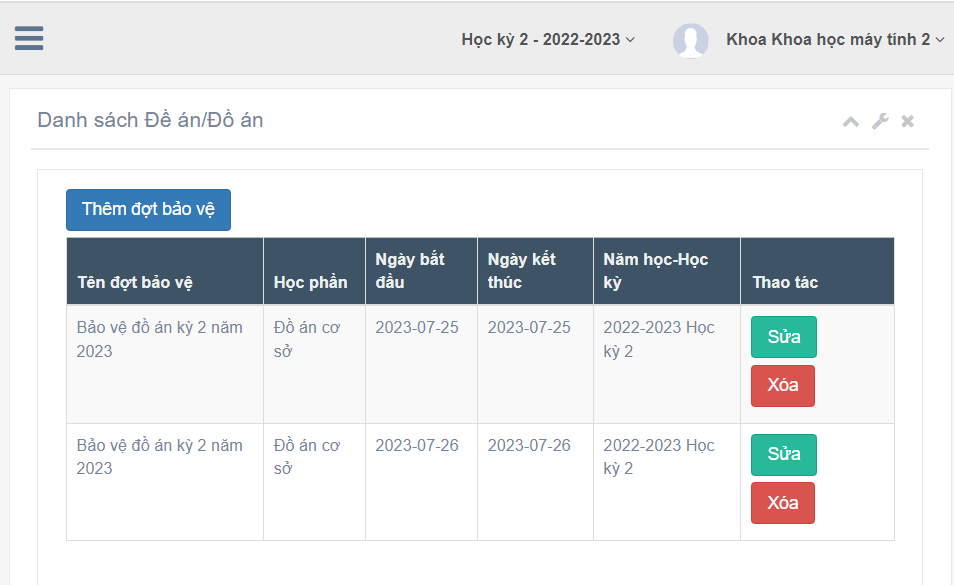
Hình 4.6. Giao diện cài đặt lớp đồ án

Giao diện cài đặt lớp đồ án giúp Khoa chọn đợt bảo vệ, hội đồng bảo vệ cho lớp đồ án cụ thể, phân sinh viên được phép bảo vệ vào hội đồng.



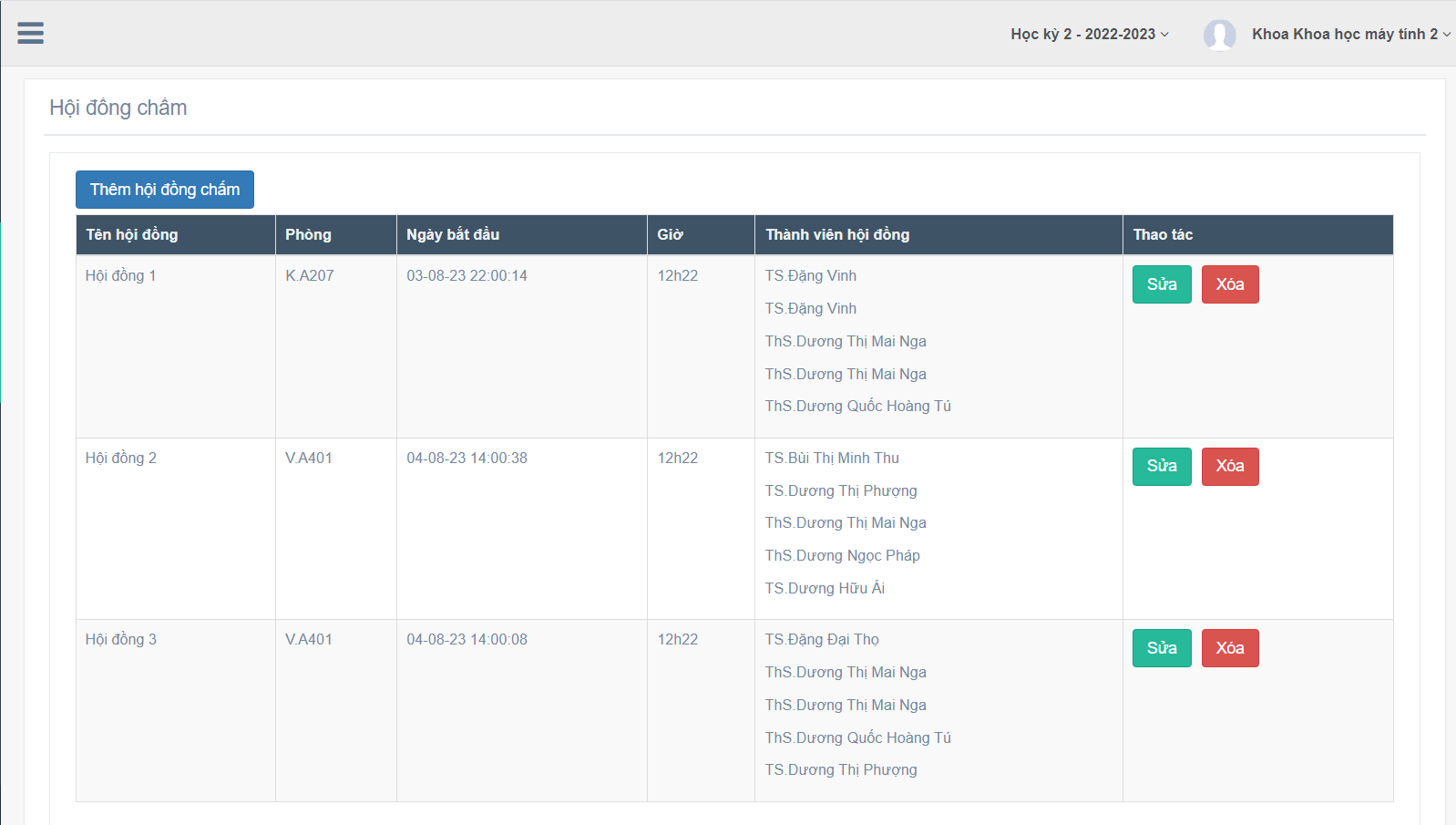
Hình 4.7. Giao diện phân nhóm lớp đồ án

Giao diện phân nhóm lớp đồ án giúp khoa phân tách lớp đồ án gốc được Phòng Đào tạo thêm thành các lớp đồ án con. Khoa thực hiện chọn số lượng nhóm cần phân chia và điền thông tin cho từng nhóm lớp.



Hình 4.8. Giao diện quản lý đợt bảo vệ

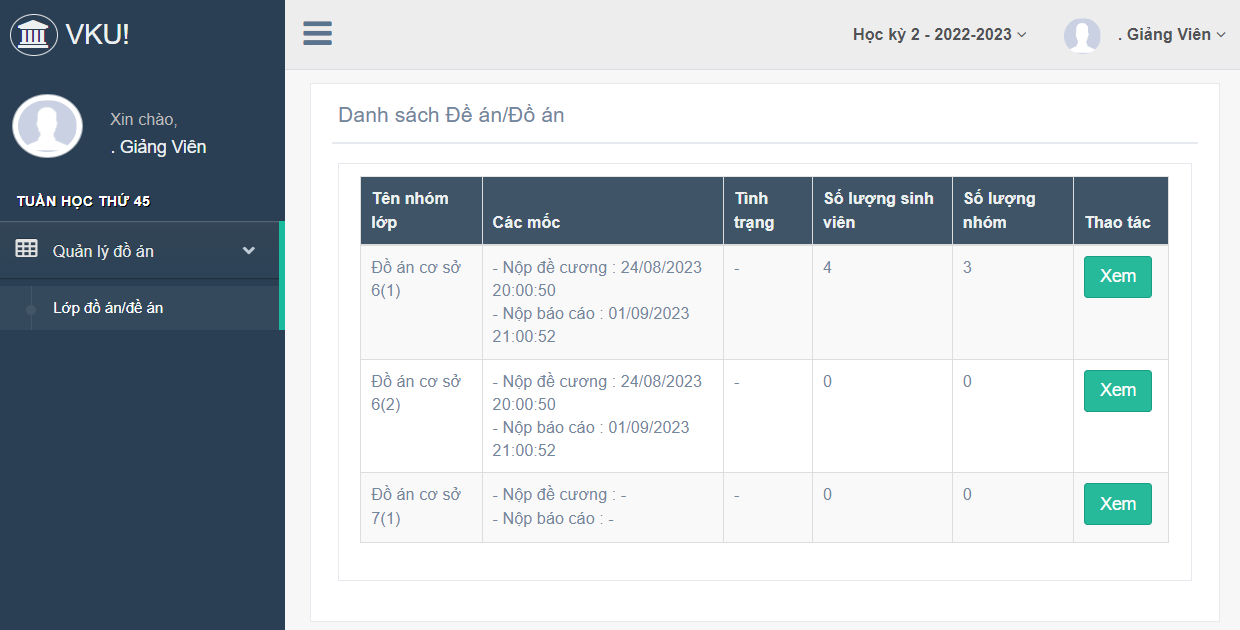
Giao diện quản lý đợt bảo vệ hiển thị danh sách đợt bảo vệ và Khoa thực hiện thêm, xóa, sửa đợt bảo vệ.



Hình 4.9. Giao diện quản lý hội đồng chấm

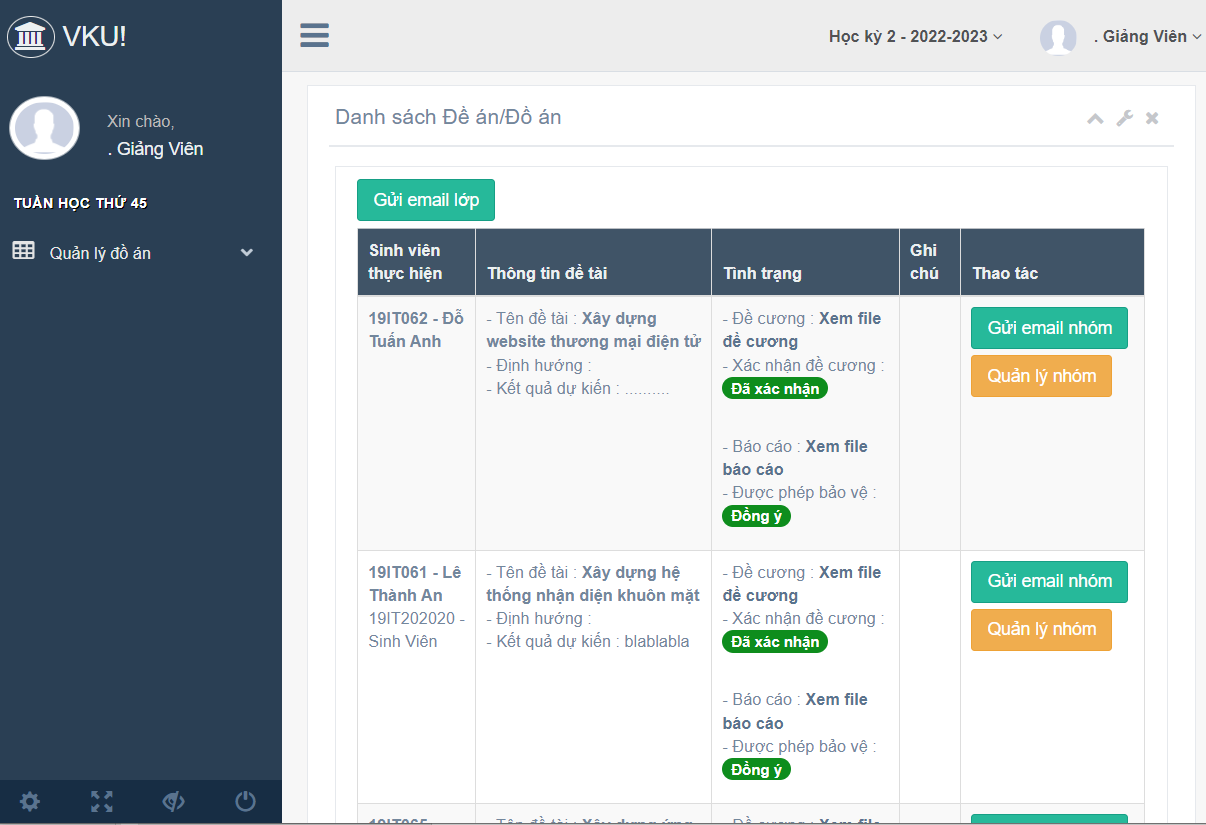
Giao diện quản lý hội đồng chấm hiển thị danh sách hội đồng chấm trong hệ thống, cho phép Khoa thực hiện thêm, xóa, sửa hội đồng

### Các chức năng của Giảng viên



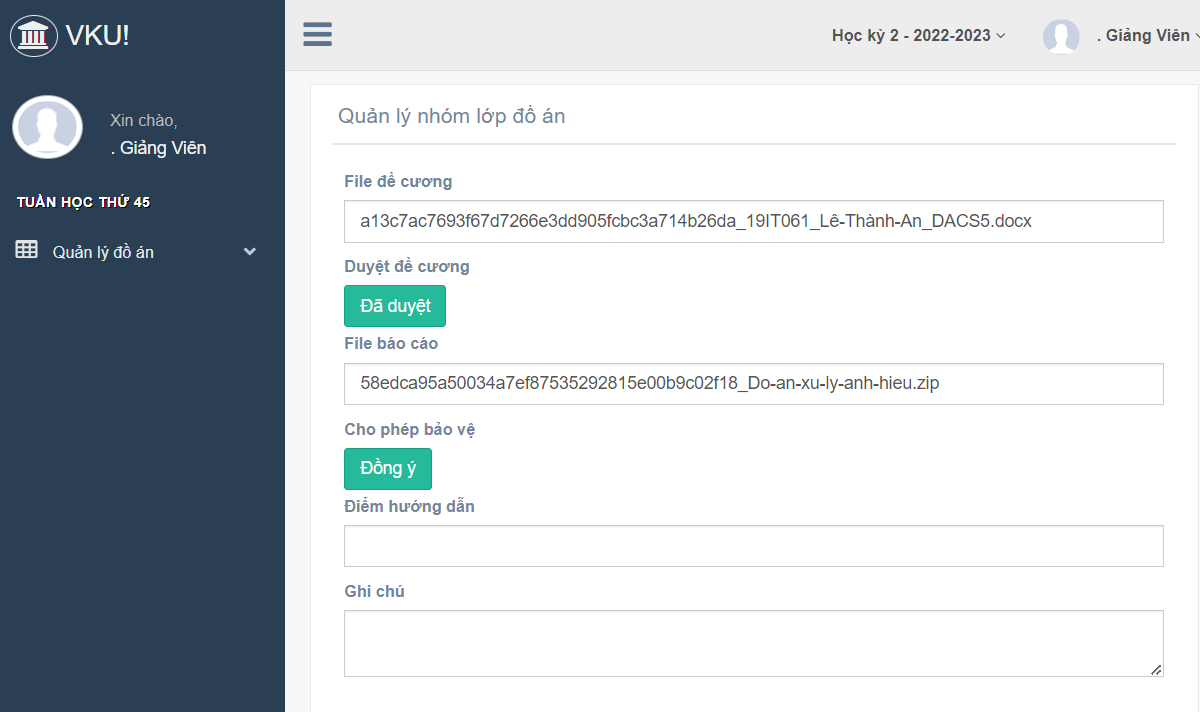
Hình 4.10. Giao diện xem danh sách lớp đồ án của Giảng viên

Giao diện xem danh sách lớp đồ án của Giảng viên hiển thị danh sách các đồ án mà giảng viên được phân công quản lý. Giảng viên có thể chọn Xem ở nhóm đồ án bất kỳ để thực hiện các thao tác cập nhật trên đồ án đã chọn.



Hình 4.11. Giao diện xem chi tiết nhóm sinh viên thực hiện đồ án

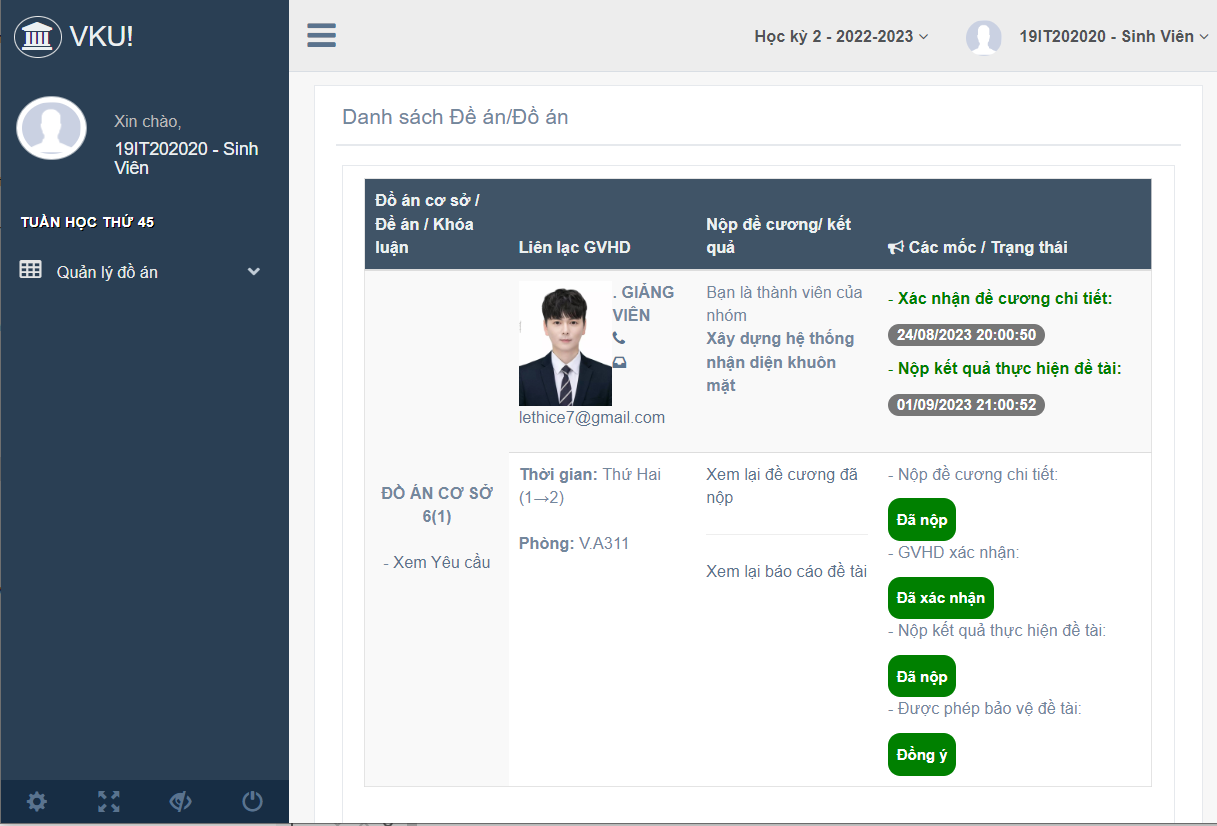
Giảng viên xem chi tiết nhóm đồ án bất kỳ bằng cách chọn xem nhóm đồ án đó. Sau đó, hệ thống sẽ hiển thị danh sách sinh viên đang thực hiện đồ án, sinh viên làm theo nhóm sẽ hiển thị chung trong cột Sinh viên thực hiên. Ngoài ra Giảng viên có thể gửi email đến lớp đồ án để tất cả sinh viên trong nhóm nắm bắt thông tin về việc thực hiện đồ án hoặc chọn chỉ gửi đến một sinh viên/nhóm sinh viên cụ thể.



Hình 4.12. Giao diện quản lý nhóm sinh viên thực hiện đồ án

Giao diện quản lý nhóm sinh viên thực hiện đồ án giúp Giảng viên tùy chỉnh các thiết lập về đồ án của nhóm/sinh viên thực hiện, duyệt đề cương, cho điểm hướng dẫn, đồng ý cho bảo vệ.

### Các chức năng của Sinh viên



Hình 4.13. Giao diện xem chi tiết lớp đồ án của Sinh viên

Giao diện xem chi tiết lớp đồ án của Sinh viên hiển thị danh sách nhóm đồ án của Sinh viên trong học kỳ và thông tin các trạng thái hiện tại của đồ án. Sinh viên có thể chọn nhóm, tách nhóm, nộp đề cương, nộp báo cáo trên đồ án bất kỳ.

KIẾN NGHỊ VÀ KẾT LUẬN

* + - * 1. Về mặt lý thuyếts
* Tìm hiểu về framework Laravel trong việc phát triển website của Phòng đào tạo
* Phân tích cấu trúc cơ sở dữ liệu của hệ thống hiện có
* Lên kế hoạch thực hiện phân tách hệ thống quản lý đồ án ra thành website riêng từ hệ thống đào tạo
  + - * 1. Về mặt thực tiễn
* Dựa trên yêu cầu đã phân tích thực hiện mô hình hóa bằng biểu đồ UML
* Các chức năng tương tự hệ thống cũ đã phân tách ra hệ thống riêng cơ bản đã hoàn thành.
  + - * 1. Hướng phát triển
* Tiếp tục sửa các lỗi còn tồn tại trong hệ thống đã xây dựng và khắc phục các lỗ hổng bảo mật.
* Thực hiện phân tách hệ thống còn lại như Công tác sinh viên, Khảo thí, Cơ sở vật chất khỏi hệ thống đào tạo.

TÀI LIỆU THAM KHẢO

1. HiepSiIT. *https://hiepsiit.com/detail/php/ngonnguphp/gioithieu.* [Online]

2. Admin. *Wikiscret.com.* [Online] 08 16, 2021. https://wikisecret.com/php-la-gi-uu-diem-va-nhuoc-diem-cua-ngon-ngu-lap-trinh-php.html.

3. —. PHP là gì? Ưu điểm và nhược điểm của ngôn ngữ lập trình PHP. *Wikiscret.com.* [Trực tuyến] 16 08 2021. https://wikisecret.com/php-la-gi-uu-diem-va-nhuoc-diem-cua-ngon-ngu-lap-trinh-php.html.

4. Giới thiệu cơ bản về hệ quản trị cơ sở dữ liệu MySQL. *Hour Of Code.* [Online] https://hourofcode.vn/gioi-thieu-co-ban-ve-quan-tri-co-du-lieu-mysql/.

5. Laravel. Tài liệu hướng dẫn laravel. *https://laravel.com.* [Online] https://laravel.com/docs/8.x.