*Bìa của bạn!*

# LỜI CẢM ƠN

Suốt quá trình học tập trong trường những năm qua, em đã được các thầy cô cung cấp và truyền đạt tất cả kiến thức quý giá và thiết thực nhất. Ngoài ra, em còn được rèn luyện một tinh thần học tập và rèn luyện tính độc lập, sáng tạo. Đây là tính cách hết sức cần thiết để có thể thành công khi bắt tay vào nghề nghiệp trong trong tương lai.

Em xin chân thành cảm ơn Ban giám hiệu trường, ban chủ nhiệm khoa, cùng các thầy cô giáo đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết trong những năm học tại trường. Và quãng thời gian đó thật hữu ích làm em trưởng thành lên rất nhiều khi chuẩn bị ra trường đó là những hành trang không thể thiếu trong công việc sau này.

Em xin chân thành cảm ơn thầy cô đã tận tình quan tâm, giúp đỡ và hướng dẫn em trong suốt thời gian làm đồ án để em hoàn thành tốt đồ án tốt nghiệp này.

Mặc dù đã cố gắng trong quá trình học tập và nghiên cứu nhưng do kinh nghiệm thực tế và trình độ chuyên môn chưa được nhiều nên em không tránh khỏi những thiếu sót em rất mong được sự chỉ bảo, góp phần chân thành từ các thầy, cô giáo cùng tất cả các bạn.

Em xin chân thành cảm ơn!

Hà Nội, ngày tháng năm 2023

Sinh viên thực hiện

# MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc154962341)

[MỤC LỤC 3](#_Toc154962342)

[MỤC LỤC HÌNH ẢNH 6](#_Toc154962343)

[MỤC LỤC BẢNG BIỂU 9](#_Toc154962344)

[CHƯƠNG 1: GIỚI THIỆU CHUNG 10](#_Toc154962345)

[1.1. Giới thiệu đề tài 10](#_Toc154962346)

[1.2. Lý do, mục tiêu và đối tượng của đề tài 10](#_Toc154962347)

[1.2.1. Lý do chọn đề tài 10](#_Toc154962348)

[1.2.2. Mục tiêu của đề tài 11](#_Toc154962349)

[1.2.3. Đối tượng và phạm vi nghiên cứu 11](#_Toc154962350)

[1.3. Cấu trúc đồ án 12](#_Toc154962351)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 13](#_Toc154962352)

[2.1. Giới thiệu các công cụ 13](#_Toc154962353)

[2.1.1. Visual Studio Code 13](#_Toc154962354)

[2.1.2. Git, Github 14](#_Toc154962355)

[2.1.3. Xampp 18](#_Toc154962356)

[2.2. Ngôn ngữ lập trình & thư viện 19](#_Toc154962357)

[2.2.1. Python 19](#_Toc154962358)

[2.2.2. Django 20](#_Toc154962359)

[2.2.3. Keras 20](#_Toc154962360)

[2.2.4. Numpy 21](#_Toc154962361)

[2.2.5. OpenCV 22](#_Toc154962362)

[2.2.6. TensorFlow 23](#_Toc154962363)

[2.2.7. Xltw 24](#_Toc154962364)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 25](#_Toc154962365)

[3.1. Quy trình nghiệp vụ 25](#_Toc154962366)

[3.2. Mô hình ca sử dụng 25](#_Toc154962367)

[3.2.1. Biểu đồ ca sử dụng mức tổng thể của hệ thống 25](#_Toc154962368)

[3.2.2. Biểu đồ ca sử dụng mức chi tiết 28](#_Toc154962369)

[3.2.3. Đặc tả các ca sử dụng 32](#_Toc154962370)

[3.2.4. Biểu đồ hoạt động đăng nhập 40](#_Toc154962371)

[3.2.5. Biểu đồ hoạt động quản lý sinh viên 42](#_Toc154962372)

[3.2.6. Biểu đồ hoạt động quản lý môn học 47](#_Toc154962373)

[3.2.7. Biểu đồ hoạt động quản lý điểm số 49](#_Toc154962374)

[3.2.8. Biểu đồ hoạt động quản lý điểm danh 52](#_Toc154962375)

[3.2.9. Biểu đồ hoạt động điểm danh 53](#_Toc154962376)

[3.2.10. Biểu đồ hoạt động quản lý phòng học 54](#_Toc154962377)

[3.2.11. Biểu đồ hoạt động xem thông tin môn học 56](#_Toc154962378)

[3.2.12. Biểu đồ hoạt động xem điểm số 57](#_Toc154962379)

[3.2.13. Biểu đồ hoạt động quản lý thông tin cá nhân cho sinh viên 58](#_Toc154962380)

[3.2.14. Biểu đồ hoạt động xem thông tin phòng học 60](#_Toc154962381)

[3.3. Biểu đồ tuần tự 61](#_Toc154962382)

[2.2.8. Biểu đồ tuần tự đăng nhập tài khoản 61](#_Toc154962383)

[2.2.9. Biểu đồ tuần tự quản lý sinh viên 63](#_Toc154962384)

[2.2.10. Biểu đồ tuần tự quản lý môn học 67](#_Toc154962385)

[2.2.11. Biểu đồ tuần tự quản lý điểm số 69](#_Toc154962386)

[2.2.12. Biểu đồ tuần tự quản lý điểm danh 71](#_Toc154962387)

[2.2.13. Biểu đồ tuần tự điểm danh 72](#_Toc154962388)

[2.2.14. Biểu đồ tuần tự quản lý phòng học 73](#_Toc154962389)

[2.2.15. Biểu đồ tuần tự xem thông tin môn học 75](#_Toc154962390)

[2.2.16. Biểu đồ tuần tự xem điểm số 75](#_Toc154962391)

[2.2.17. Biểu đồ tuần tự quản lý thông tin cá nhân cho sinh viên 76](#_Toc154962392)

[2.2.18. Biểu đồ tuần tự xem thông tin phòng học 78](#_Toc154962393)

[3.4. Biểu đồ lớp 78](#_Toc154962394)

[2.2.19. Biểu đồ lớp quản lý sinh viên 78](#_Toc154962395)

[2.2.20. Biểu đồ lớp đăng nhập tài khoản 80](#_Toc154962396)

[2.2.21. Biểu đồ lớp quản lý môn học 81](#_Toc154962397)

[2.2.22. Biểu đồ lớp quản lý điểm số 82](#_Toc154962398)

[2.2.23. Biểu đồ lớp quản lý điểm danh 83](#_Toc154962399)

[2.2.24. Biểu đồ lớp điểm danh sinh viên 83](#_Toc154962400)

[2.2.25. Biểu đồ lớp quản lý phòng học 84](#_Toc154962401)

[2.2.26. Biểu đồ lớp xem thông tin môn học 85](#_Toc154962402)

[2.2.27. Biểu đồ lớp xem điểm số 85](#_Toc154962403)

[2.2.28. Biểu đồ lớp quản lý thông tin cá nhân cho sinh viên 86](#_Toc154962404)

[2.2.29. Biểu đồ lớp xem thông tin phòng học 86](#_Toc154962405)

[CHƯƠNG 4. TRIỂN KHAI VÀ XÂY DỰNG HỆ THỐNG 87](#_Toc154962406)

[4.1. Xây dựng cơ sở dữ liệu 87](#_Toc154962407)

[4.1.1. Biểu đồ lớp tổng quát 87](#_Toc154962408)

[4.1.2. Biểu đồ Diagram toàn hệ thống 88](#_Toc154962409)

[4.1.3. Cấu trúc các bảng 88](#_Toc154962410)

[4.2. Xây dựng giao diện 90](#_Toc154962411)

[4.2.1. Giao diện dành cho giảng viên 90](#_Toc154962412)

[4.2.2. Giao diện dành cho sinh viên 91](#_Toc154962413)

[KẾT LUẬN 93](#_Toc154962414)

[TÀI LIỆU THAM KHẢO 94](#_Toc154962415)

# MỤC LỤC HÌNH ẢNH

[Hình 2.1.1.1. Visual Studio Code 13](#_Toc154962231)

[Hình 2.1.2.1. Logo của công cụ Git 14](#_Toc154962232)

[Hình 2.1.3.1. Logo công cụ Xampp 18](#_Toc154962233)

[Hình 2.2.1.1. Django 20](#_Toc154962234)

[Hình 2.2.2.1. Logo thư viện Keras 21](#_Toc154962235)

[Hình 2.2.3.1. Logo thư viện Numpy 21](#_Toc154962236)

[Hình 2.2.4.1. Logo thư viện OpenCV 22](#_Toc154962237)

[Hình 2.2.5.1. TensorFlow 23](#_Toc154962238)

[Hình 3.2.1.1. Biểu đồ Use Case tổng quát 27](#_Toc154962239)

[Hình 3.2.2.1. Use case đăng nhập (Quản trị viên) 28](#_Toc154962240)

[Hình 3.2.2.2. Use case quản lý sinh viên 28](#_Toc154962241)

[Hình 3.2.2.3. Use case quản lý môn học 29](#_Toc154962242)

[Hình 3.2.2.4. Use case quản lý điểm 29](#_Toc154962243)

[Hình 3.2.2.5. Use case quản lý điểm danh 30](#_Toc154962244)

[Hình 3.2.2.6. Use case quản lý phòng học 30](#_Toc154962245)

[Hình 3.2.2.7. Use case đăng nhập cho sinh viên 31](#_Toc154962246)

[Hình 3.2.2.8. Use case xem danh sách môn học 31](#_Toc154962247)

[Hình 3.2.2.9. Use case xem điểm số 32](#_Toc154962248)

[Hình 3.2.2.10. Use case quản lý thông tin cá nhân 32](#_Toc154962249)

[Hình 3.2.2.11. Use case xem thông tin phòng học 32](#_Toc154962250)

[Hình 3.2.4.1. Biểu đồ hoạt động đăng nhập (giảng viên) 40](#_Toc154962251)

[Hình 3.2.4.2. Biểu đồ hoạt động đăng nhập (sinh viên) 41](#_Toc154962252)

[Hình 3.2.5.1. Biểu đồ hoạt động xuất excel danh sách sinh viên 42](#_Toc154962253)

[Hình 3.2.5.2. Biểu đồ hoạt động thêm sinh viên 43](#_Toc154962254)

[Hình 3.2.5.3. Biểu đồ hoạt động sửa sinh viên 44](#_Toc154962255)

[Hình 3.2.5.4. Biểu đồ hoạt động xoá sinh viên 45](#_Toc154962256)

[Hình 3.2.5.5. Biểu đồ hoạt động tìm kiếm sinh viên 46](#_Toc154962257)

[Hình 3.2.6.1. Biểu đồ hoạt động thêm môn học 47](#_Toc154962258)

[Hình 3.2.6.2. Biểu đồ hoạt động sửa môn học 48](#_Toc154962259)

[Hình 3.2.7.1. Biểu đồ hoạt động quản lý thêm điểm 50](#_Toc154962260)

[Hình 3.2.7.2. Biểu đồ hoạt động quản lý sửa điểm 50](#_Toc154962261)

[Hình 3.2.7.3. Biểu đồ hoạt động quản lý xuất điểm 51](#_Toc154962262)

[Hình 3.2.8.1. Biểu đồ hoạt động quản lý xem thông tin điểm danh 52](#_Toc154962263)

[Hình 3.2.9.1. Biểu đồ hoạt động điểm danh khuôn mặt sinh viên 53](#_Toc154962264)

[Hình 3.2.10.1. Biểu đồ hoạt động thêm phòng học 54](#_Toc154962265)

[Hình 3.2.10.2. Biểu đồ hoạt động sửa phòng học 55](#_Toc154962266)

[Hình 3.2.11.1. Biểu đồ hoạt động quản lý xem thông tin môn học 56](#_Toc154962267)

[Hình 3.2.12.1. Biểu đồ hoạt động xem điểm số 57](#_Toc154962268)

[Hình 3.2.13.1. Biểu đồ hoạt động sửa thông tin cá nhân 59](#_Toc154962269)

[Hình 3.2.13.2. Biểu đồ hoạt động đổi mật khẩu 59](#_Toc154962270)

[Hình 3.2.14.1. Biểu đồ hoạt động xem thông tin phòng học 60](#_Toc154962271)

[Hình 3.3.1.1. Biểu đồ tuần tự đăng nhập tài khoản (giảng viên) 61](#_Toc154962272)

[Hình 3.3.1.2. Biểu đồ tuần tự đăng nhập tài khoản (sinh viên) 62](#_Toc154962273)

[Hình 3.3.2.1. Biểu đồ tuần tự thêm sinh viên 63](#_Toc154962274)

[Hình 3.3.2.2. Biểu đồ tuần tự sửa sinh viên 64](#_Toc154962275)

[Hình 3.3.2.3. Biểu đồ tuần tự xoá sinh viên 65](#_Toc154962276)

[Hình 3.3.2.4. Biểu đồ tuần tự xuất excel danh sách sinh viên 66](#_Toc154962277)

[Hình 3.3.2.5. Biểu đồ tuần tự tìm kiếm sinh viên 66](#_Toc154962278)

[Hình 3.3.3.1. Biểu đồ tuần tự quản lý thêm môn học 67](#_Toc154962279)

[Hình 3.3.3.2. Biểu đồ tuần tự quản lý sửa môn học 68](#_Toc154962280)

[Hình 3.3.4.1. Biểu đồ tuần tự quản lý thêm điểm 69](#_Toc154962281)

[Hình 3.3.4.2. Biểu đồ tuần tự quản lý sửa điểm 70](#_Toc154962282)

[Hình 3.3.4.3. Biểu đồ tuần tự quản lý xuất điểm 71](#_Toc154962283)

[Hình 3.3.5.1. Biểu đồ tuần tự quản lý xem thông tin điểm danh 72](#_Toc154962284)

[Hình 3.3.6.1. Biểu đồ tuần tự điểm danh khuôn mặt sinh viên 72](#_Toc154962285)

[Hình 3.3.7.1. Biểu đồ tuần tự thêm phòng học 73](#_Toc154962286)

[Hình 3.3.7.2. Biểu đồ tuần tự sửa phòng học 74](#_Toc154962287)

[Hình 3.3.8.1. Biểu đồ tuần tự quản lý xem thông tin môn học 75](#_Toc154962288)

[Hình 3.3.9.1. Biểu đồ tuần tự xem điểm số 76](#_Toc154962289)

[Hình 3.3.10.1. Biểu đồ tuần tự sửa thông tin cá nhân 76](#_Toc154962290)

[Hình 3.3.10.2. Biểu đồ tuần tự đổi mật khẩu 77](#_Toc154962291)

[Hình 3.3.11.1. Biểu đồ tuần tự xem thông tin phòng học 78](#_Toc154962292)

[Hình 3.4.2.1. Biểu đồ VOPC quản lý sinh viên 78](#_Toc154962293)

[Hình 3.4.2.2. Lớp đối tượng students\_students 79](#_Toc154962294)

[Hình 3.4.1.1. Biểu đồ VOPC đăng nhập tài khoản 80](#_Toc154962295)

[Hình 3.4.1.3. Lớp đối tượng auth\_user 80](#_Toc154962296)

[Hình 3.4.1.4. Lớp đối tượng user\_infouser 81](#_Toc154962297)

[Hình 3.4.3.1. Biểu đồ VOPC quản lý môn học 81](#_Toc154962298)

[Hình 3.4.3.2. Lớp đối tượng subjects\_subject 81](#_Toc154962299)

[Hình 3.4.4.1. Biểu đồ VOPC quản lý điểm số 82](#_Toc154962300)

[Hình 3.4.4.2. Lớp đối tượng scores\_scores 82](#_Toc154962301)

[Hình 3.4.5.1. Biểu đồ VOPC quản lý điểm danh 83](#_Toc154962302)

[Hình 3.4.5.2. Lớp đối tượng attendance\_attendaceinfo 83](#_Toc154962303)

[Hình 3.4.6.1. Biểu đồ VOPC điểm danh sinh viên 84](#_Toc154962304)

[Hình 3.4.7.1. Biểu đồ VOPC quản lý phòng học 84](#_Toc154962305)

[Hình 3.4.7.2. Lớp đối tượng students\_room 84](#_Toc154962306)

[Hình 3.4.8.1. Biểu đồ VOPC xem thông tin môn học 85](#_Toc154962307)

[Hình 3.4.9.1. Biểu đồ VOPC xem điểm số 85](#_Toc154962308)

[Hình 3.4.10.1. Biểu đồ VOPC quản lý thông tin cá nhân cho sinh viên 86](#_Toc154962309)

[Hình 3.4.11.1. Biểu đồ VOPC xem thông tin phòng học 86](#_Toc154962310)

[Hình 4.1.1.1. Biểu đồ lớp tổng quát 87](#_Toc154962311)

[Hình 4.1.2.1. Biểu đồ Diagram toàn hệ thống 88](#_Toc154962312)

[Hình 4.1.3.1. Cấu trúc bảng auth\_user 88](#_Toc154962313)

[Hình 4.1.3.2. Cấu trúc bảng user\_info 89](#_Toc154962314)

[Hình 4.1.3.3. Cấu trúc bảng students\_students 89](#_Toc154962315)

[Hình 4.1.3.4. Cấu trúc bảng subjects\_subject 89](#_Toc154962316)

[Hình 4.1.3.5. Cấu trúc bảng students\_specialization 89](#_Toc154962317)

[Hình 4.1.3.6. Cấu trúc bảng students\_classname 90](#_Toc154962318)

[Hình 4.1.3.7. Cấu trúc bảng scores\_scores 90](#_Toc154962319)

[Hình 4.1.3.8. Cấu trúc bảng attendance\_attendanceinfo 90](#_Toc154962320)

[Hình 4.2.1.1. Giao diện chức năng đăng nhập 90](#_Toc154962321)

[Hình 4.2.1.2. Giao diện trang chủ 90](#_Toc154962322)

[Hình 4.2.1.3. Giao diện quản lý sinh viên 91](#_Toc154962323)

[Hình 4.2.1.4. Giao diện quản lý môn học 91](#_Toc154962324)

[Hình 4.2.1.5. Giao diện quản lý điểm số 91](#_Toc154962325)

[Hình 4.2.1.6. Giao diện điểm danh 91](#_Toc154962326)

[Hình 4.2.1.7. Giao diện quản lý phòng học 91](#_Toc154962327)

[Hình 4.2.2.1. Giao diện đăng nhập 91](#_Toc154962328)

[Hình 4.2.2.2. Giao diện xem thông tin môn học 91](#_Toc154962329)

[Hình 4.2.2.3. Giao diện xem điểm số 91](#_Toc154962330)

[Hình 4.2.2.4. Giao diện quản lý thông tin cá nhân 91](#_Toc154962331)

[Hình 4.2.2.5. Giao diện xem thông tin phòng học 92](#_Toc154962332)

# MỤC LỤC BẢNG BIỂU

[Bảng 3.2.1.1. Xác định các Use case và Actor 26](#_Toc154962333)

[Bảng 3.2.3.1. Đặc tả Use Case đăng nhập 33](#_Toc154962334)

[Bảng 3.2.3.2. Đặc tả Use Case Quản lý sinh viên 34](#_Toc154962335)

[Bảng 3.2.3.3. Đặc tả Use Case Quản lý môn học 35](#_Toc154962336)

[Bảng 3.2.3.4. Đặc tả Use Case Quản lý điểm số 36](#_Toc154962337)

[Bảng 3.2.3.5. Đặc tả Use Case Quản lý điểm danh 37](#_Toc154962338)

[Bảng 3.2.3.6. Đặc tả Use Case Điểm danh 38](#_Toc154962339)

[Bảng 3.2.3.7. Đặc tả Use Case Quản lý phòng học 39](#_Toc154962340)

# CHƯƠNG 1: GIỚI THIỆU CHUNG

## Giới thiệu đề tài

Đề tài "Xây dựng hệ thống quản lý sinh viên" nhằm trợ giúp việc quản lý các thông tin liên quan đến sinh viên và các thông tin liên quan đến học tập cho trường đại học. Website này sẽ cung cấp một giao diện trực quan và dễ sử dụng cho người dùng để quản lý thông tin. Hệ thống mang đến nhiều chức năng, tiện ích trong việc quản lý thông tin giảng viên, thông tin của sinh viên, quản lý điểm của sinh viên theo phương pháp học tín chỉ, và thực hiện điểm danh cho sinh viên dựa trên khuôn mặt.

Hệ thống quản lý sinh viên trường đại học TMU được sử dụng để quản lý các thông tin liên quan đến giảng viên và sinh viên trong phạm vi nội bộ của trường. Hệ thống này nhằm cung cấp cho giảng viên những chức năng cần thiết để tiến hành kiểm soát và quản lý các thông tin của sinh viên, quản lý hoạt động cũng như tình trạng học tập thông qua việc chấm điểm và chấm công cho sinh viên.

Mục tiêu của đề tài này là xây dựng một hệ thống quản lý sinh viên chuyên nghiệp, giúp quản lý các thông tin liên quan đến giảng viên và sinh viên. Đồng thời, nó cũng giúp tối ưu hóa quá trình quản lý thông tin cho học sinh và giảng viên.

Về Website này sẽ có rất nhiều vấn đề khác nhau. Trong đề tài này, em chỉ tập trung nghiên cứu về xây dựng và phát triển một hệ thống quản lý sinh viên, sử dụng python.

## Lý do, mục tiêu và đối tượng của đề tài

### Lý do chọn đề tài

Đề tài "Xây dựng hệ thống quản lý sinh viên" đã được lựa chọn vì nhiều lý do quan trọng. Công tác quản lý sinh viên (kết quả học tập) của sinh viên đóng vai trò hết sức quan trọng đối với hoạt động của một khoa trong các trường đại học và cao đẳng. Một trường đại học mỗi năm tiếp nhận nhiều sinh viên đến trường nhập học và trong quá trình học tập nhà trường sẽ quản lý hồ sơ sinh viên. Việc quản lý số lượng hồ sơ sinh viên khổng lồ sẽ rất khó khăn nếu chỉ sử dụng giấy tờ. Việc quản lý chỉ bằng giấy tờ sẽ dễ xảy ra sai sót hoặc mất mát. Vì vậy ta cần tới một hệ thống quản lý để giúp việc quản lý sinh viên tốt hơn, tránh việc mất mát dữ liệu hoặc sai sót trong quá trình quản lý. Hệ thống quản lí cửa thông tin sinh viên sẽ chịu trách nhiệm quản lý tất cả các hoạt động giảng dạy và thời gian học tập của giảng viên và sinh viên, thông qua đó sẽ giúp việc nâng cao chất lượng đào tạo cũng như học tập tại trường học.

### Mục tiêu của đề tài

Phần mềm hỗ trợ trong việc quản lý thông tin sinh sinh viên như MSSV, tên sinh viên, giới tính …v.v. Phần mềm quản lý sinh viên giúp cho người sử dụng tiết kiệm được thời gian, nhanh chóng, thao tác đơn giản, dễ dàng sử dụng và quản lý.

Xây dựng một hệ thống mới phù hợp, dễ hiểu, dễ sử dụng cho người dùng phần mềm.

Giúp quản lý hệ thống sinh viên trở nên tối ưu hơn, công tác quản lý dễ dàng và tiện lợi hơn.

Giúp bản thân có thêm khả năng sáng tạo, tư duy thông qua thiết kế giao diện tương tác với người dùng, vận dụng kiến thức bản thân vào thực tiễn.

### Đối tượng và phạm vi nghiên cứu

Đối tượng của đề tài "Xây dựng hệ thống quản lý sinh viên" là các trường đại học. Phạm vi nghiên cứu của đề tài này sẽ tập trung vào việc xây dựng một hệ thống quản lý sinh viên. Nghiên cứu sẽ tập trung vào việc phân tích và thiết kế giao diện, tích hợp các chức năng quản lý tất cả các hoạt động giảng dạy và thời gian học tập của giảng viên và sinh viên.

Phạm vi nghiên cứu cũng sẽ bao gồm việc phát triển các tính năng thông minh, như điểm danh bằng khuôn mặt. Đồng thời, nghiên cứu sẽ tập trung vào việc tạo ra một giao diện thân thiện. Phạm vi nghiên cứu sẽ tập trung vào việc xây dựng một hệ thống quản lý hiệu quả và cung cấp các chức năng quản lý thông tin sinh viên, điểm danh, chấm điểm, thống kê.

Trong phạm vi nghiên cứu, sẽ xem xét các yêu cầu và nhu cầu cụ thể của trường đại học, từ đó xây dựng một hệ thống quản lý sinh viên phù hợp và đáp ứng được các yêu cầu và nhu cầu của cả hai bên.

## Cấu trúc đồ án

CHƯƠNG 1: GIỚI THIỆU CHUNG

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

CHƯƠNG 4: TRIỂN KHAI VÀ XÂY DỰNG HỆ THỐNG

KẾT LUẬN

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## Giới thiệu các công cụ

### Visual Studio Code



Hình 2.1.1.1. Visual Studio Code

**Khái niệm Visual Studio Code:**

Visual Studio Code là một trình biên tập mã được phát hành bởi Microsoft dành cho Windows, Linux, macOS. Là một trong những trình soạn thảo mã nguồn phổ biến nhất được sử dụng bởi các lập trình viên.

Nhanh, nhẹ, hỗ trợ đa nền tảng, nhiều tính năng. Nó hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, và cải tiến mã nguồn. Nó cũng cho phép tùy chỉnh, do đó người dùng có thể thay đổi theme, phím tắt, và các tùy chọn khác. Nó miễn phí và là phần mềm mã nguồn mở.

**Tính năng Visual Studio Code:**

* Hỗ trợ nhiều ngôn ngữ lập trình: VSC hỗ trợ nhiều ngôn ngữ lập trình như C/C++, C#, F#, Visual Basic, HTML, CSS, JavaScript, PHP … Vì vậy, nó dễ dàng phát hiện và đưa ra thông báo nếu chương trình có lỗi.
* Hỗ trợ đa nền tảng: Các trình viết code thông thường chỉ được sử dụng hoặc cho Windows hoặc Linux hoặc Mac Systems. Nhưng Visual Studio Code có thể hoạt động tốt trên cả ba nền tảng trên.
* Cung cấp kho tiện ích mở rộng: Trong trường hợp lập trình viên muốn sử dụng một ngôn ngữ lập trình không nằm trong số các ngôn ngữ Visual Studio hỗ trợ, họ có thể tải xuống tiện ích mở rộng. Điều này vẫn sẽ không làm giảm hiệu năng của phần mềm, bởi vì phần mở rộng này hoạt động như một chương trình độc lập.
* Kho lưu trữ an toàn: Đi kèm với sự phát triển của lập trình là nhu cầu về lưu trữ an toàn. Với Visual Studio Code, người dùng có thể hoàn toàn yên tâm vì nó dễ dàng kết nối với Git hoặc bất kỳ kho lưu trữ hiện có nào.
* Hỗ trợ Web: Visual Studio Code hỗ trợ nhiều ứng dụng web. Ngoài ra, nó cũng có một trình soạn thảo và thiết kế website.
* Lưu trữ dữ liệu dạng phân cấp: Phần lớn tệp lưu trữ đoạn mã đều được đặt trong các thư mục tương tự nhau. Ngoài ra, Visual Studio Code còn cung cấp các thư mục cho một số tệp đặc biệt quan trọng.
* Hỗ trợ viết code: Một số đoạn code có thể thay đổi chút ít để thuận tiện cho người dùng. Visual Studio Code sẽ đề xuất cho lập trình viên các tùy chọn thay thế nếu có.
* Hỗ trợ thiết bị đầu cuối: Visual Studio Code có tích hợp thiết bị đầu cuối, giúp người dùng khỏi phải chuyển đổi giữa hai màn hình hoặc trở về thư mục gốc khi thực hiện các thao tác.
* Màn hình đa nhiệm: Người dùng Visual Studio Code có thể mở cùng lúc nhiều tệp tin và thư mục – mặc dù chúng không hề liên quan với nhau.

### Git, Github



Hình 2.1.2.1. Logo của công cụ Git

**Khái niệm Git:**

Git là một hệ thống kiểm soát phiên bản phân tán mã nguồn mở và miễn phí, được thiết kế để xử lý mọi thứ từ các dự án nhỏ đến rất lớn với tốc độ cao và hiệu quả.

**Cách thức hoạt động của Git:**

Git là một phiên bản của Distributed Version Control System – DVCS hay còn gọi là VCS. Sử dụng Git sẽ đem tới cho các lập trình viên một kho lưu trữ, trong đó chứa đầy đủ các lịch sử thay đổi của hệ thống.

Tuy nhiên, so với các VCS thông thường, Git cũng có một số điểm khác biệt. Trong đó, điểm khác biệt nổi bật nhất của Git đó là cách đọc và lưu trữ dữ liệu của phần mềm này. Hầu hết các hệ thống lưu trữ thông tin bằng cách thay đổi dựa trên file và lưu trữ thông tin dưới dạng danh sách. Hệ thống này sẽ tổng hợp và cập nhật thông tin, sau đó lưu trữ dưới dạng một tập hợp danh sách. Các file có thể thay đổi theo thời gian.

Cách hoạt động và lưu trữ thông tin của Git lại hoàn toàn khác. Để lưu trữ thông tin, Git sẽ coi các thông tin là một tập hơn snapshot. Đây là tập hợp toàn bộ ảnh chụp các nội dung của file thông tin tại thời điểm.

Khi đưa ra các lệnh, git sẽ chụp lại thông tin ở thời điểm đó và tạo ra một snapshot. Để nâng cao tính hiệu quả, Git sẽ không lưu trữ file nếu các tệp không có sự thay đổi. Thay vào đó, phần mềm sẽ liên kết dữ liệu tới một tệp giống với file đã được lưu trữ trước đó.

Nhờ vào điểm khác biệt với các VCS này, Git sẽ không tốn nhiều thời gian xem xét các thông tin, các bản giải mã của phiên bản/hệ thống trước. Thay vào đó, Git nghĩ về dữ liệu khác đi, giúp việc lưu trữ dễ dàng hơn. Với cách lưu trữ thông minh này, Git giống như một hệ thống gồm nhiều tệp nhỏ khác nhau.

**Ưu điểm khi dùng Git:**

Sử dụng Git sẽ không tạo ra xung đột giữa các lập trình viên khi thực hiện code hệ thống. Sử dụng Git sẽ giúp nhà phát triển có thể dễ dàng tìm lại, quay lại các phiên bản cũ hơn. Việc kiểm soát sẽ cực kỳ đơn giản và hiệu quả.

Git cũng có các lợi ích tuyệt vời đối với các dự án đang chạy song song. Đặc biệt là các dự án có cùng một cơ sở code. Việc sử dụng Git sẽ giúp phân nhánh tốt hơn. Qua đó giúp việc thực hiện các dự án dễ dàng và hiệu quả hơn:

* Thao tác thực hiện nhanh, gọn hơn, việc sử dụng an toàn hơn.
* Người dùng có thể dễ dàng kết hợp các nhánh với nhau. Qua đó giúp đơn giản hóa quy trình code theo nhóm.
* Khi sử dụng Git, người dùng chỉ cần sử dụng clone mã nguồn từ khoa chứa/phiên bản nào đó có sẵn trong kho lưu trữ/ một nhánh trong kho lưu trữ. Hệ thống lưu trữ chính xác sẽ cho phép bạn có thể làm việc bất cứ lúc nào, ở bất cứ đâu.
* Việc lập trình các dự án song song sẽ cực kỳ dễ dàng, hiệu quả và chính xác.



Hình 2.1.2.2. Logo công cụ GitHub

**Khái niệm GitHub:**

GitHub là một dịch vụ nổi tiếng cung cấp kho lưu trữ mã nguồn Git cho các dự án phần mềm. Github có đầy đủ những tính năng của Git, ngoài ra nó còn bổ sung những tính năng về social để các developer tương tác với nhau.

Github cung cấp các tính năng social networking như feeds, followers, và network graph để các developer học hỏi kinh nghiệm của nhau thông qua lịch sử commit.

Github trở thành một yếu tố có sức ảnh hưởng lớn trong cộng đồng nguồn mở. Cùng với Linkedin, Github được coi là một sự thay thế cho CV của bạn. Các nhà tuyển dụng cũng rất hay tham khảo Github profile để hiểu về năng lực coding của ứng viên.

**Tính năng của Github:**

GitHub được coi là một mạng xã hội dành cho lập trình viên lớn nhất và dễ dùng nhất với các tính năng cốt lõi như:

* Wiki, issue, thống kê, đổi tên project, project được đặt vào namespace là user.
* Watch project: theo dõi hoạt động của project của người khác. Xem quá trình người ta phát triển phầm mềm thế nào, project phát triển ra sao.
* Follow user: theo dõi hoạt động của người khác.

**Ưu điểm của Github:**

* Quản lý source code dễ dàng

Khi bạn tạo một repo, toàn bộ source code của repo đó được lưu trên GitHub. Tại đây, bạn có thể coi lại quá trình mình đã làm việc thông qua các comment sau mỗi lần commit. Và cái hay ở đây, là nhiều người có thể cùng làm một repo.

Lợi ích đầu tiên, chính là bạn biết được ai đã commit và commit cái gì. Tiếp theo, source của bạn có thể phát triển theo nhiều nhánh. Nguyên tắc làm việc với các nhánh như thế này: Bạn có thể rẽ nhiều nhánh để phát triển project. Nhưng cuối cùng, bạn phải merge lại vào nhánh MASTER để ra được project hoàn chỉnh.

* Tracking sự thay đổi qua các version

Khi có nhiều member cùng thực hiện một dự án thì khá là phức tạp để theo dõi revisons – ai thay đổi cái gì, lúc nào và mấy cái files đó được stored ở đâu. Đừng lo vì GitHub đã tính đến chuyện này giúp bạn, bằng cách luôn lưu lại những thay đổi bạn đã push lên repository. Cũng tương tự với Microsoft Word hay Google Drive, bạn có một lịch sử phiên bản để phòng trường hợp các phiên bản trước đó bị mất hay không được lưu.

* Markdown

Markdown là một cách định dạng text trên web. Bạn có thể chỉnh sửa cách hiển thị của document, format từ như định dạng in đậm hay in nghiêng, thêm hình và tạo list những thứ bạn có thể làm với Markdown. Hầu hết, Markdown chỉ là đoạn text đơn thuần với những ký tự đặc biệt chèn vào, như # hay \*. Trong GitHub thì bạn có thể sử dụng Mardown ở những nơi: Git, Comments tại Issues và Pull Requests, các file có đuôi .md hay .markdown extension.

* Github là một kho tài nguyên tuyệt vời

Với chức năng Explore, bạn có thể theo dõi, tìm kiếm những open source projects theo đúng technology pattern mà bạn ưa thích. Github hỗ trợ code search không kể nó ở dưới dạng một project riêng biệt hay là website. Ngoài ra, nền tảng này cũng có SEO khá tốt nên người dùng có thể tìm kiếm bất kỳ code string nào được chia sẻ public.

### Xampp

XAMPP là một bộ công cụ phát triển web miễn phí và mã nguồn mở. XAMPP bao gồm các thành phần cần thiết để tạo và triển khai các ứng dụng web như Apache Server, MySQL database, PHP và Perl. XAMPP được sử dụng rộng rãi để phát triển và kiểm thử các ứng dụng web trên máy tính cá nhân trước khi triển khai lên môi trường sản xuất.



Hình 2.1.3.1. Logo công cụ Xampp

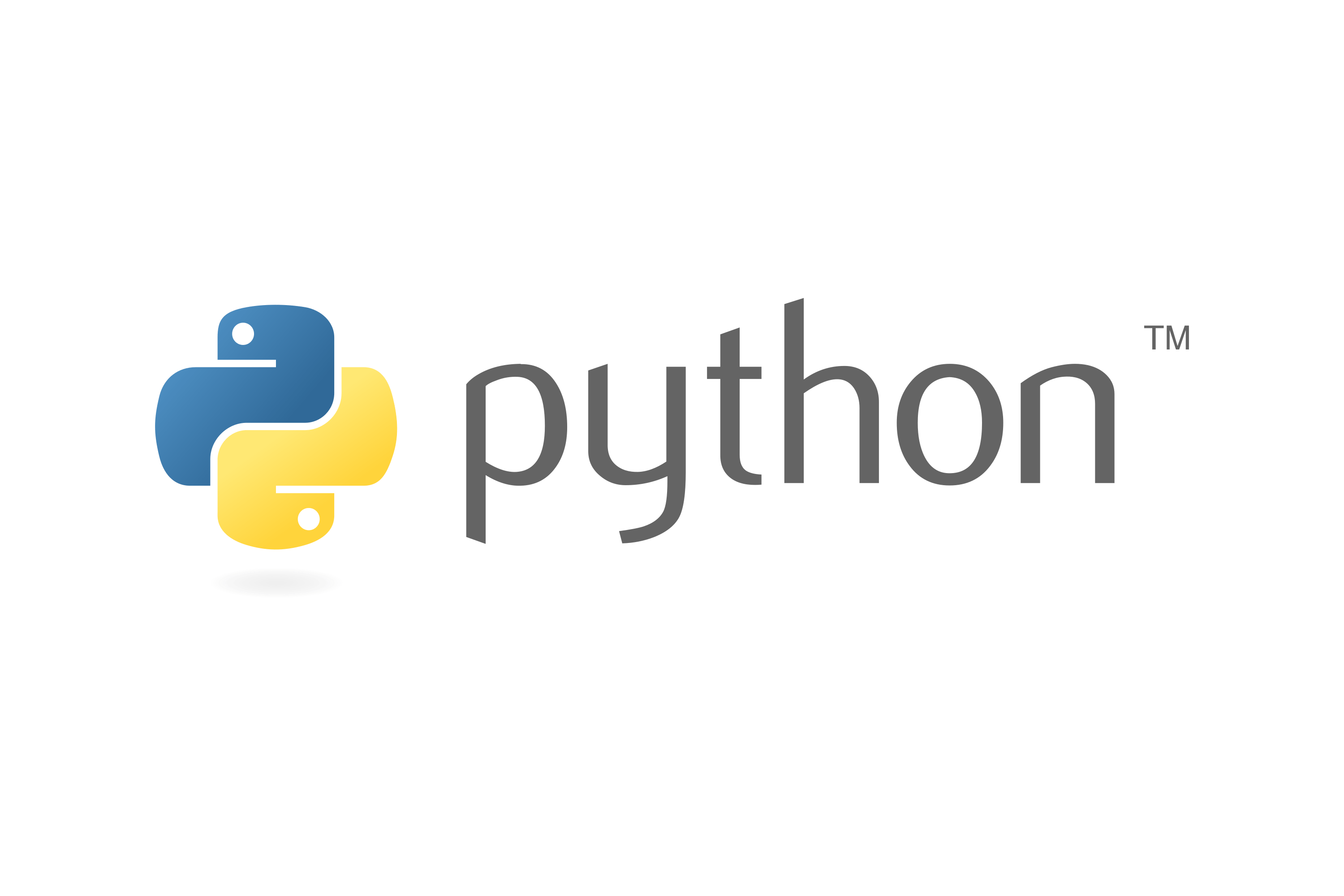
Khi xử dụng Xampp, chúng ta có tận dụng tối ưu các ưu điểm và các chức năng tiện ích mà phần mềm này có thể mang lại.

1. Dễ cài đặt: XAMPP có giao diện cài đặt đơn giản và dễ sử dụng, giúp người dùng nhanh chóng cài đặt và cấu hình các thành phần web như Apache, MySQL, PHP và Perl trên máy tính cá nhân.
2. Đa nền tảng: XAMPP hỗ trợ trên các nền tảng khác nhau như Windows, macOS và Linux, cho phép người dùng phát triển và kiểm thử ứng dụng web trên nhiều môi trường khác nhau.
3. Gọn nhẹ: XAMPP được thiết kế để có kích thước nhỏ gọn và tối ưu, giúp tiết kiệm tài nguyên hệ thống và đảm bảo hiệu suất tốt.
4. Hỗ trợ đa ngôn ngữ: XAMPP hỗ trợ nhiều ngôn ngữ lập trình như PHP, Perl, Python, Ruby, cho phép người dùng phát triển ứng dụng web bằng ngôn ngữ ưa thích của mình.
5. Quản lý cơ sở dữ liệu: XAMPP đi kèm với hệ quản trị cơ sở dữ liệu MySQL, cho phép người dùng quản lý và tương tác với cơ sở dữ liệu một cách dễ dàng.
6. Tiện ích bổ sung: XAMPP cung cấp các công cụ hỗ trợ như phpMyAdmin (quản lý cơ sở dữ liệu MySQL), FileZilla (FTP client/server), Mercury Mail Server (hỗ trợ gửi và nhận email), Tomcat (máy chủ ứng dụng Java), giúp người dùng phát triển và triển khai ứng dụng web một cách tiện lợi.

Tóm lại, XAMPP là một bộ công cụ phát triển web đa nền tảng, dễ sử dụng và hỗ trợ nhiều chức năng quan trọng để phát triển và kiểm thử các ứng dụng web trên máy tính cá nhân. Giúp việc triển khai mã nguồn ở phía máy cá nhân trước khi đưa chúng lên môi trường product thực tế.

## Ngôn ngữ lập trình & thư viện

### Python



Ngôn ngữ lập trình Python là một ngôn ngữ lập trình thông dịch, dễ đọc và dễ hiểu. Nó được phát triển bởi Guido van Rossum và ra mắt lần đầu vào năm 1991. Python nổi tiếng với cú pháp rõ ràng và dễ đọc, giúp người lập trình tập trung vào logic của chương trình mà không cần quá lo lắng về cấu trúc cú pháp phức tạp.

Ngôn ngữ này rất linh hoạt và có thư viện phong phú, hỗ trợ nhiều mục đích khác nhau như phát triển web, khoa học dữ liệu, máy học (machine learning), và automation. Cộng đồng lớn của Python đã tạo ra nhiều công cụ và thư viện mạnh mẽ như NumPy, Pandas, TensorFlow, và Django.

Python cũng nổi tiếng với triết lý "batteries included", có nghĩa là nó đi kèm với một số lượng lớn các thư viện chuẩn để giúp lập trình viên giải quyết các vấn đề phổ biến một cách dễ dàng.

### Django

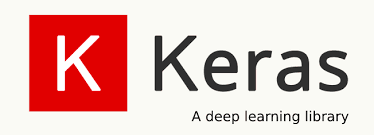


Hình 2.2.1.1. Django

Django là phần mềm bạn có thể sử dụng để phát triển các ứng dụng web một cách nhanh chóng và hiệu quả. Hầu hết các ứng dụng web có một số chức năng phổ biến, như xác thực, truy xuất thông tin từ cơ sở dữ liệu và quản lý cookie. Các nhà phát triển phải viết mã cho chức năng tương tự vào mọi ứng dụng web mà họ viết. Django giúp họ làm việc dễ dàng hơn bằng cách nhóm các chức năng khác nhau thành một tập hợp lớn các mô-đun có thể tái sử dụng, được gọi là một khung ứng dụng web. Các nhà phát triển sử dụng khung web Django để sắp xếp và viết mã của họ hiệu quả hơn và giảm đáng kể thời gian phát triển web.

### Keras

Keras là một open source cho Neural Network được viết bởi ngôn ngữ Python. Nó là một library được phát triển vào năm 2005 bởi Francois Chollet, là một kỹ sư nghiên cứu Deep Learning. Keras có thể sử dụng chung với các thư viện nổi tiếng như Tensorflow, CNTK, Theano. Một số ưu điểm của Keras như:

* Dễ sử dụng, dùng đơn giản hơn Tensor, xây dựng model nhanh.
* Run được trên cả CPU và GPU.
* Hỗ trợ xây dựng CNN, RNN hoặc cả hai. Với những người mới tiếp cận đến Deep như mình thì mình chọn sử dụng Keras để build model vì nó đơn giản, dễ nắm bắt hơn các thư viện khác.
* 

Hình 2.2.2.1. Logo thư viện Keras

### Numpy

Numpy là một thư viện lõi phục vụ cho khoa học máy tính của Python, hỗ trợ cho việc tính toán các mảng nhiều chiều, có kích thước lớn với các hàm đã được tối ưu áp dụng lên các mảng nhiều chiều đó. Numpy đặc biệt hữu ích khi thực hiện các hàm liên quan tới Đại Số Tuyến Tính.



Hình 2.2.3.1. Logo thư viện Numpy

Những tính năng quan trọng của Numpy gồm có:

* Đối tượng mảng N-chiều mạnh mẽ
* Các chức năng broadcasting
* Phép biến đổi Fourier, khả năng số ngẫu nhiên
* Các công cụ để tích hợp mã C / C ++ và Fortran.

Bên cạnh đó, NumPy còn được sử dụng như một nơi chứa dữ liệu chung đa chiều hiệu quả. Các kiểu dữ liệu tùy ý có thể được xác định bằng cách sử dụng NumPy, cho phép NumPy tích hợp liền mạch và nhanh chóng với nhiều loại cơ sở dữ liệu.

### OpenCV

OpenCV là một thư viện mã nguồn mở hàng đầu cho thị giác máy tính (computer vision), xử lý ảnh và máy học, và các tính năng tăng tốc GPU trong hoạt động thời gian thực.

OpenCV được phát hành theo giấy phép BSD, do đó nó hoàn toàn miễn phí cho cả học thuật và thương mại. Nó có các interface C++, C, Python, Java và hỗ trợ Windows, Linux, Mac OS, iOS và Android. OpenCV được thiết kế để tính toán hiệu quả và với sự tập trung nhiều vào các ứng dụng thời gian thực. Được viết bằng tối ưu hóa C/C++, thư viện có thể tận dụng lợi thế của xử lý đa lõi. Được sử dụng trên khắp thế giới, OpenCV có cộng đồng hơn 47 nghìn người dùng và số lượng download vượt quá 6 triệu lần. Phạm vi sử dụng từ nghệ thuật tương tác, cho đến lĩnh vực khai thác mỏ, bản đồ trên web hoặc công nghệ robot.



Hình 2.2.4.1. Logo thư viện OpenCV

OpenCV đang được sử dụng rộng rãi trong các ứng dụng bao gồm:

Hình ảnh street view Kiểm tra và giám sát tự động Robot và xe hơi tự lái Phân tích hình ảnh y tế Tìm kiếm và phục hồi hình ảnh/video Phim - cấu trúc 3D từ chuyển động Nghệ thuật sắp đặt tương tác.

### TensorFlow

Với sự bùng nổ của lĩnh vực Trí Tuệ Nhân Tạo – A.I. trong thập kỷ vừa qua, machine learning và deep learning rõ ràng cũng phát triển theo cùng. Và ở thời điểm hiện tại, TensorFlow chính là thư viện mã nguồn mở cho machine learning nổi tiếng nhất thế giới, được phát triển bởi các nhà nghiên cứu từ Google. Việc hỗ trợ mạnh mẽ các phép toán học để tính toán trong machine learning và deep learning đã giúp việc tiếp cận các bài toán trở nên đơn giản, nhanh chóng và tiện lợi hơn nhiều.

Các hàm được dựng sẵn trong thư viện cho từng bài toán cho phép TensorFlow xây dựng được nhiều neural network. Nó còn cho phép bạn tính toán song song trên nhiều máy tính khác nhau, thậm chí trên nhiều CPU, GPU trong cùng 1 máy hay tạo ra các dataflow graph – đồ thị luồng dữ liệu để dựng nên các model. Nếu bạn muốn chọn con đường sự nghiệp trong lĩnh vực A.I. này, nắm rõ những điều cơ bản của TensorFlow thực sự rất quan trọng.

Được viết bằng C++ và thao tác interface bằng Python nên phần performance của TensorFlow cực kỳ tốt. Đối tượng sử dụng nó cũng đa dạng không kém: từ các nhà nghiên cứu, nhà khoa học dữ liệu và dĩ nhiên không thể thiếu các lập trình viên.



Hình 2.2.5.1. TensorFlow

### Xltw

Xlwt là một API python mã nguồn mở để ghi và định dạng dữ liệu ở các định dạng tệp Microsoft Excel cũ hơn (97/2000 / XP / 2003 XLS). Các nhà phát triển Python có thể dễ dàng thao tác các tệp Microsoft Excel cũ hơn bằng cách sử dụng API Python thuần túy này mà không phụ thuộc vào các mô-đun và gói bên ngoài phân phối Python tiêu chuẩn. API cung cấp các tính năng viết và thao tác phong phú bao gồm, hợp nhất ô, định dạng hàng, thao tác ngày tháng, sử dụng siêu liên kết, chèn hình ảnh, sử dụng ngăn và hơn thế nữa.

# CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Quy trình nghiệp vụ

Phòng quản lý sinh viên cần quản lý toàn bộ sinh viên đang theo học tại trường, mỗi sinh viên trong trường sẽ được gán một mã số sinh viên duy nhất, thông tin của một sinh viên trong trường bao gồm họ tên sinh viên, ngày sinh, địa chỉ, số điện thoại.

Một khoa trong trường sẽ có một hoặc nhiều ngành và một ngành cũng sẽ có một hoặc nhiều lớp, khi sinh viên tiến hành nhập học tại trường thì sinh viên đó sẽ thuộc một lớp dựa vào ngành học của sinh viên đó. Phòng quản lý sinh viên sẽ tiến hành nhập danh sách sinh viên từ phần mềm.

Khi cố vấn học tập cần thống kê danh sách sinh viên trong lớp phòng quản lý sinh viên sẽ in danh sách sinh viên trong lớp đó cho cố vấn học tập.

Mỗi ngành học sẽ học nhiều môn học giống hoặc khác nhau tuỳ theo khung chương trình đào tạo của mỗi ngành.

Trong quá trình học tập, sinh viên sẽ lấy 2 điểm thành phần, điểm kết thúc môn vào cuối học kỳ, phòng quản lý sinh viên sẽ tính điểm trung bình và tiến hành nhập các điểm thi này theo môn học cho toàn bộ sinh viên trong lớp.

Khi có kết quả thi của từng môn học, phòng quản lý sinh viên sẽ xuất bảng điểm cho từng lớp.

Khi sinh viên cần xem lại điểm toàn bộ các môn mình đã học, sinh viên có thể vào xem điểm số của mình trên hệ thống.

## Mô hình ca sử dụng

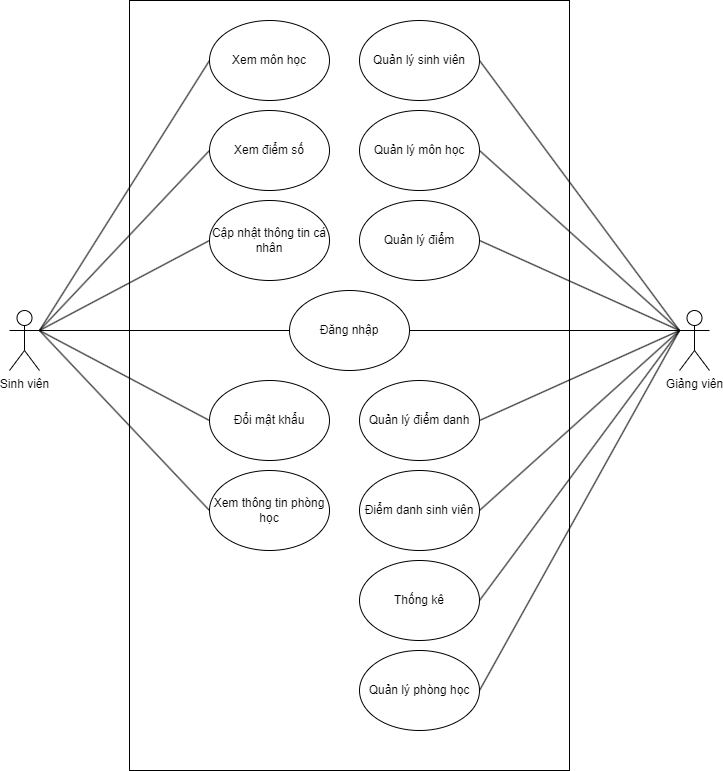
### Biểu đồ ca sử dụng mức tổng thể của hệ thống

- Xác định các Use case và Actor

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Actor** | **Tên Use Case** |
| 1 | Giảng viên | * Đăng nhập * Đăng xuất * Quản lý sinh viên   + Tìm kiếm sinh viên  + Xuất excel danh sách sinh viên  + Xem thông tin chi tiết sinh viên  + Sửa thông tin sinh viên  + Xoá sinh viên   * Quản lý môn học   + Thêm môn học  + Sửa môn học   * Quản lý điểm số   + Xuất excel  + Thêm điểm  + Sửa điểm   * Quản lý thông tin điểm danh   + Xem thông tin điểm danh   * Quản lý điểm danh   + Tạo khuôn mặt điểm danh cho sinh viên   * Điểm danh sinh viên * Thống kê * Quản lý phòng học |
| 2 | Sinh Viên | * Đăng nhập * Đăng xuất * Sửa thông tin cá nhân * Đổi mật khẩu * Xem danh sách môn học * Xem điểm số * Xem thông tin phòng học |

Bảng 3.2.1.1. Xác định các Use case và Actor

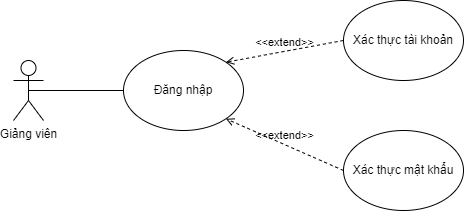
- Biểu đồ Use Case tổng quát



Hình 3.2.1.1. Biểu đồ Use Case tổng quát

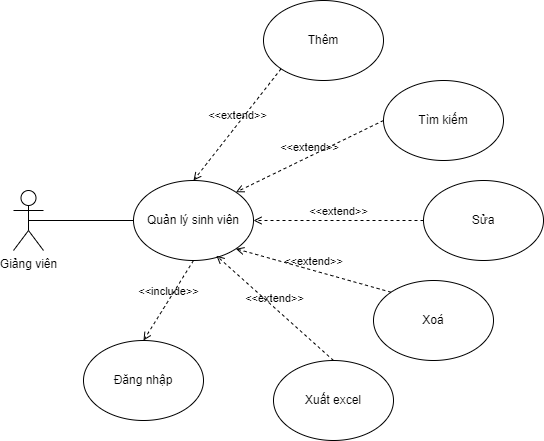
### Biểu đồ ca sử dụng mức chi tiết

* Use case đăng nhập (Giảng viên)



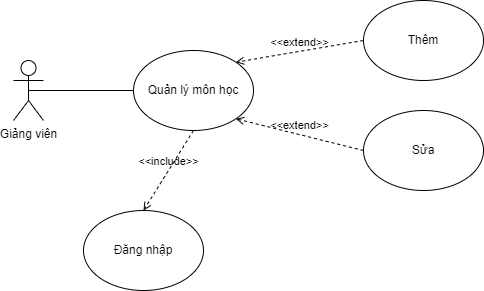
Hình 3.2.2.1. Use case đăng nhập giảng viên

* Use case quản lý sinh viên



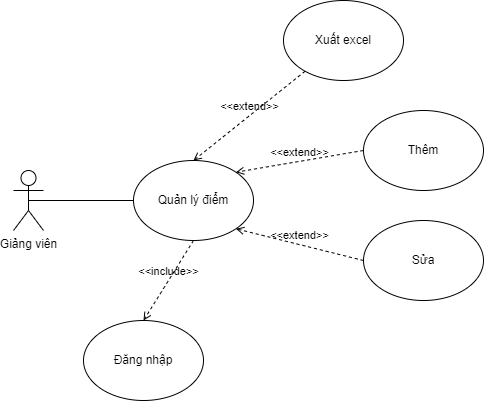
Hình 3.2.2.2. Use case quản lý sinh viên

* Use case quản lý môn học



Hình 3.2.2.3. Use case quản lý môn học

* Use case quản lý điểm



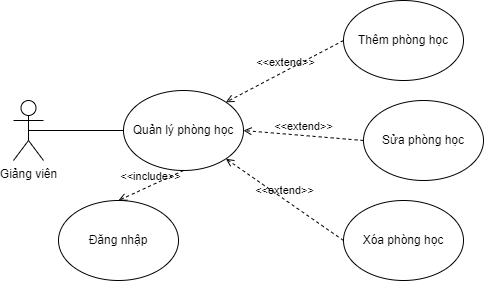
Hình 3.2.2.4. Use case quản lý điểm

* Use case quản lý điểm danh



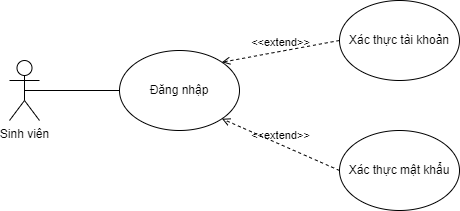
Hình 3.2.2.5. Use case quản lý điểm danh

* Use case quản lý phòng học



Hình 3.2.2.6. Use case quản lý phòng học

* Use case đăng nhập cho sinh viên



Hình 3.2.2.7. Use case đăng nhập cho sinh viên

* Use case xem danh sách môn học



Hình 3.2.2.8. Use case xem danh sách môn học

* Use case xem điểm số



Hình 3.2.2.9. Use case xem điểm số

* Use case quản lý thông tin cá nhân



Hình 3.2.2.10. Use case quản lý thông tin cá nhân

* Use case xem thông tin phòng học



Hình 3.2.2.11. Use case xem thông tin phòng học

### Đặc tả các ca sử dụng

**- Đặc tả Use Case Đăng nhập**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên UC** | **Đăng ký tài khoản** |
| 1 | Tác nhân | Giảng viên, sinh viên |
| 2 | Mô tả | Người dùng đăng nhập vào hệ thống |
| 3 | Điều kiện trước | Người dùng phải có tài khoản của hệ thống được lưu trong CSDL |
| 4 | Luồng sự kiện chính | 1. Hệ thống hiện thị form đăng nhập 2. Nhập thông tin lên form, hệ thống tiến hành kiểm tra thông tin 3. Lưu phiên đăng nhập |
| 5 | Luồng sự kiện phụ | Nhập sai thông tin sẽ thông báo tài khoản hoặc mật khẩu không đúng và yêu cầu nhập lại |
| 6 | Điều kiện sau | Nếu đăng nhập được thì chuyển hướng tới trang chủ, cho phép thực hiện các tác vụ |
| 7 | Yêu cầu đặc biệt | Không có |

Bảng 3.2.3.1. Đặc tả Use Case đăng nhập

**- Đặc tả Use Case Quản lý sinh viên**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên UC** | **Quản lý sinh viên – Giảng viên** |
| 1 | Tác nhân | Quản trị viên |
| 2 | Mô tả | Cho phép người giảng viên có thể kiểm soát thông tin của sinh viên, cho phép xuất thông tin sinh viên, thêm sinh viên, sửa thông tin sinh viên, xóa sinh viên và tìm kiếm thông tin sinh viên. |
| 3 | Điều kiện trước | Giảng viên đã đăng nhập |
| 4 | Luồng sự kiện chính | 1. Sau khi hoàn thành đăng nhập, hệ thống cho phép giảng viên truy cập vào giao diện chính 2. Giảng viên chọn chức năng quản lý sinh viên 3. Hiển thị giao diện quản lý sinh viên 4. Giảng viên thực hiện các chức năng với thông tin sinh viên rồi lưu CSDL 5. Kết thúc use case quản lý sinh viên |
| 5 | Luồng sự kiện phụ | Nếu tác nhân đang trong giao diện quản lý sinh viên mà chọn về trang chủ thì hệ thống sẽ hỏi “Bạn có muốn thoát?”, nếu chọn có thì hệ thống chuyển về trang chủ. Kết thúc use case. |
| 6 | Điều kiện sau | Hiển thị các thông báo thành công sau khi người giảng viên hiện các thao tác với thông tin sinh viên và trả lại giao diện quản lý sinh viên ban đầu. |
| 7 | Yêu cầu đặc biệt | Không có |

Bảng 3.2.3.2. Đặc tả Use Case Quản lý sinh viên

**- Đặc tả Use Case Quản lý môn học**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên UC** | **Quản lý môn học – Giảng viên** |
| 1 | Tác nhân | Giảng viên |
| 2 | Mô tả | Cho phép người giảng viên có thể kiểm soát thông tin của môn học của mình đang phụ trách, thông qua đó cho phép thêm môn học phụ trách, sửa thông tin môn học. |
| 3 | Điều kiện trước | Giảng viên đã đăng nhập |
| 4 | Luồng sự kiện chính | 1. Sau khi hoàn thành đăng nhập, hệ thống cho phép giảng viên truy cập vào giao diện chính 2. Giảng viên chọn chức năng quản lý môn học 3. Hiển thị giao diện quản lý môn học 4. Giảng viên thực hiện các chức năng với thông tin môn học rồi lưu CSDL 5. Kết thúc use case quản lý môn học |
| 5 | Luồng sự kiện phụ | Nếu tác nhân đang trong giao diện quản lý môn học mà chọn về trang chủ thì hệ thống sẽ hỏi “Bạn có muốn thoát?”, nếu chọn có thì hệ thống chuyển về trang chủ. Kết thúc use case. |
| 6 | Điều kiện sau | Hiển thị các thông báo thành công sau khi người giảng viên hiện các thao tác với thông tin môn học và trả lại giao diện quản lý môn học ban đầu. |
| 7 | Yêu cầu đặc biệt | Không có |

Bảng 3.2.3.3. Đặc tả Use Case Quản lý môn học

**- Đặc tả Use Case Quản lý điểm số**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên UC** | **Quản lý điểm số - Giảng viên** |
| 1 | Tác nhân | Quản trị viên |
| 2 | Mô tả | Cho phép người giảng viên có thể kiểm soát thông tin của điểm số của các sinh viên thuộc 1 môn học mà mình đang phụ trách, thông qua đó cho phép giảng viên thêm điểm cho sinh viên, cập nhật điểm cho sinh viên và xuất thông tin điểm số của sinh viên. |
| 3 | Điều kiện trước | Giảng viên |
| 4 | Luồng sự kiện chính | 1. Sau khi hoàn thành đăng nhập, hệ thống cho phép giảng viên truy cập vào giao diện chính 2. Giảng viên chọn chức năng quản lý điểm số 3. Hiển thị giao diện quản lý điểm số 4. Giảng viên thực hiện các chức năng với thông tin điểm số của sinh viên rồi lưu CSDL 5. Kết thúc use case quản lý điểm số |
| 5 | Luồng sự kiện phụ | Nếu tác nhân đang trong giao diện quản lý điểm số mà chọn về trang chủ thì hệ thống sẽ hỏi “Bạn có muốn thoát?”, nếu chọn có thì hệ thống chuyển về trang chủ. Kết thúc use case. |
| 6 | Điều kiện sau | Hiển thị các thông báo thành công sau khi người giảng viên hiện các thao tác với thông tin điểm số và trả lại giao diện quản lý điểm số ban đầu. |
| 7 | Yêu cầu đặc biệt | Không có |

Bảng 3.2.3.4. Đặc tả Use Case Quản lý điểm số

**- Đặc tả Use Case Quản lý điểm danh**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên UC** | **Quản lý điểm danh – Giảng viên** |
| 1 | Tác nhân | Giảng viên |
| 2 | Mô tả | Cho phép người giảng viên có thể kiểm soát thông tin điểm danh của các sinh viên thuộc 1 môn học và lớp học mà mình đang phụ trách, thông qua đó cho phép giảng viên đánh giá được điểm chuyên cần hoặc điểm hệ số 1, hệ số 2 trong quá trình học của sinh viên. |
| 3 | Điều kiện trước | Giảng viên đã đăng nhập |
| 4 | Luồng sự kiện chính | 1. Sau khi hoàn thành đăng nhập, hệ thống cho phép giảng viên truy cập vào giao diện chính 2. Giảng viên chọn chức năng quản lý điểm danh 3. Hiển thị giao diện quản lý điểm danh 4. Giảng viên thực hiện các chức năng với thông tin điểm danh của sinh viên rồi lưu CSDL 5. Kết thúc use case quản lý điểm số |
| 5 | Luồng sự kiện phụ | Nếu tác nhân đang trong giao diện quản lý điểm danh mà chọn về trang chủ thì hệ thống sẽ hỏi “Bạn có muốn thoát?”, nếu chọn có thì hệ thống chuyển về trang chủ. Kết thúc use case. |
| 6 | Điều kiện sau | Hiển thị các thông báo thành công sau khi người giảng viên hiện các thao tác với thông tin điểm danh và trả lại giao diện quản lý điểm danh ban đầu. |
| 7 | Yêu cầu đặc biệt | Không có |

Bảng 3.2.3.5. Đặc tả Use Case Quản lý điểm danh

**- Đặc tả Use Case Điểm danh**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên UC** | **Điểm danh – Giảng viên** |
| 1 | Tác nhân | Giảng viên |
| 2 | Mô tả | Nếu tác nhân đang trong giao diện quản lý điểm danh mà chọn về trang chủ thì hệ thống sẽ hỏi “Bạn có muốn thoát?”, nếu chọn có thì hệ thống chuyển về trang chủ. Kết thúc use case. |
| 3 | Điều kiện trước | Giảng viên đã đăng nhập |
| 4 | Luồng sự kiện chính | Không có |
| 5 | Luồng sự kiện phụ | Nếu tác nhân đang trong giao diện quản lý điểm danh mà chọn về trang chủ thì hệ thống sẽ hỏi “Bạn có muốn thoát?”, nếu chọn có thì hệ thống chuyển về trang chủ. Kết thúc use case. |
| 6 | Điều kiện sau | Hiển thị các thông báo thành công sau khi người giảng viên hiện các thao tác với thông tin điểm danh và trả lại giao diện quản lý điểm danh ban đầu. |
| 7 | Yêu cầu đặc biệt | Không có |

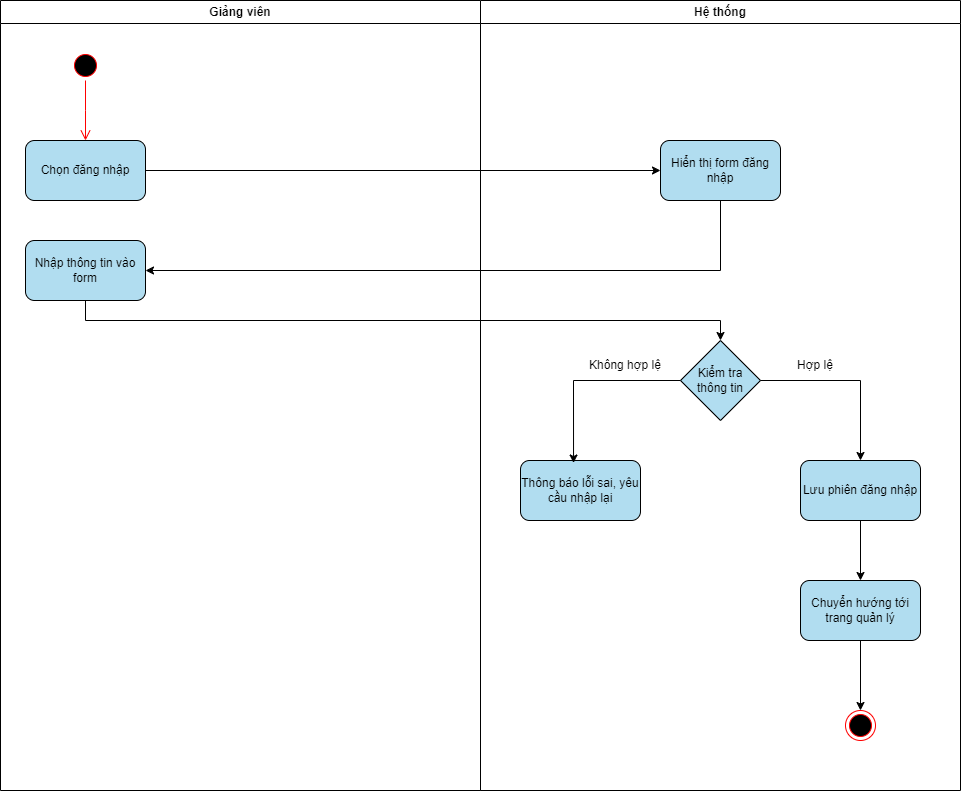
Bảng 3.2.3.6. Đặc tả Use Case Điểm danh

**- Đặc tả Use Case Quản lý phòng học**

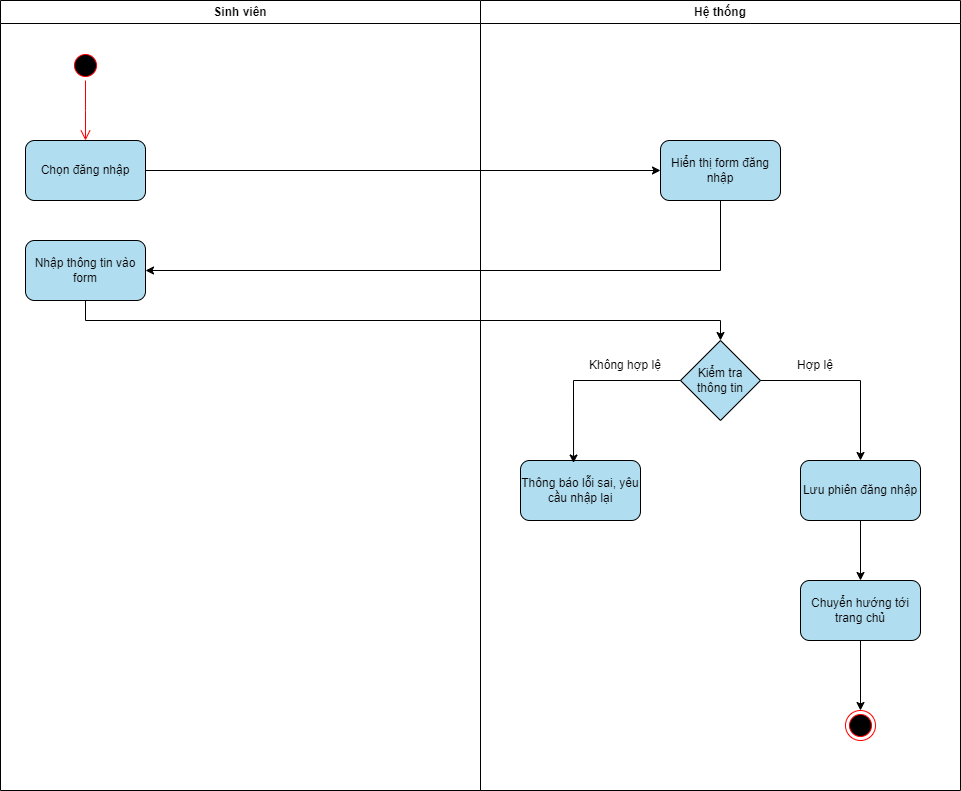
|  |  |  |
| --- | --- | --- |
| **STT** | **Tên UC** | **Quản lý phòng học – Giảng viên** |
| 1 | Tác nhân | Giảng viên |
| 2 | Mô tả | Cho phép người giảng viên có thể kiểm soát thông tin của phòng học, thông qua đó cho phép giảng viên thêm phòng học, cập nhật phòng học. |
| 3 | Điều kiện trước | Giảng viên đã đăng nhập |
| 4 | Luồng sự kiện chính | 1. Sau khi hoàn thành đăng nhập, hệ thống cho phép giảng viên truy cập vào giao diện chính 2. Giảng viên chọn chức năng quản lý phòng học 3. Hiển thị giao diện quản lý phòng học 4. Giảng viên thực hiện các chức năng với thông tin phòng học rồi lưu CSDL 5. Kết thúc use case quản lý môn học |
| 5 | Luồng sự kiện phụ | Nếu tác nhân đang trong giao diện quản lý môn học mà chọn về trang chủ thì hệ thống sẽ hỏi “Bạn có muốn thoát?”, nếu chọn có thì hệ thống chuyển về trang chủ. Kết thúc use case. |
| 6 | Điều kiện sau | Hiển thị các thông báo thành công sau khi người giảng viên hiện các thao tác với thông tin phòng học và trả lại giao diện quản lý phòng học ban đầu. |
| 7 | Yêu cầu đặc biệt | Không có |

Bảng 3.2.3.7. Đặc tả Use Case Quản lý phòng học

### Biểu đồ hoạt động đăng nhập

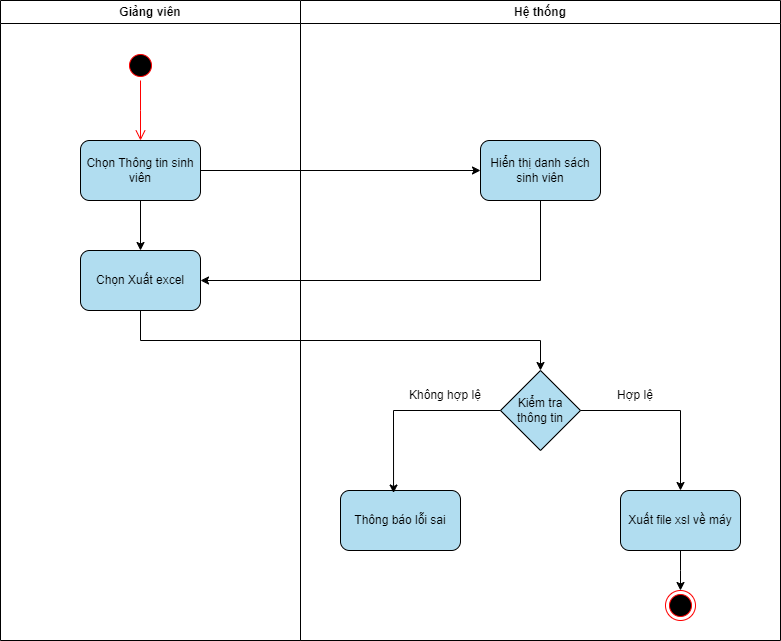


Hình 3.2.4.1. Biểu đồ hoạt động đăng nhập (giảng viên)

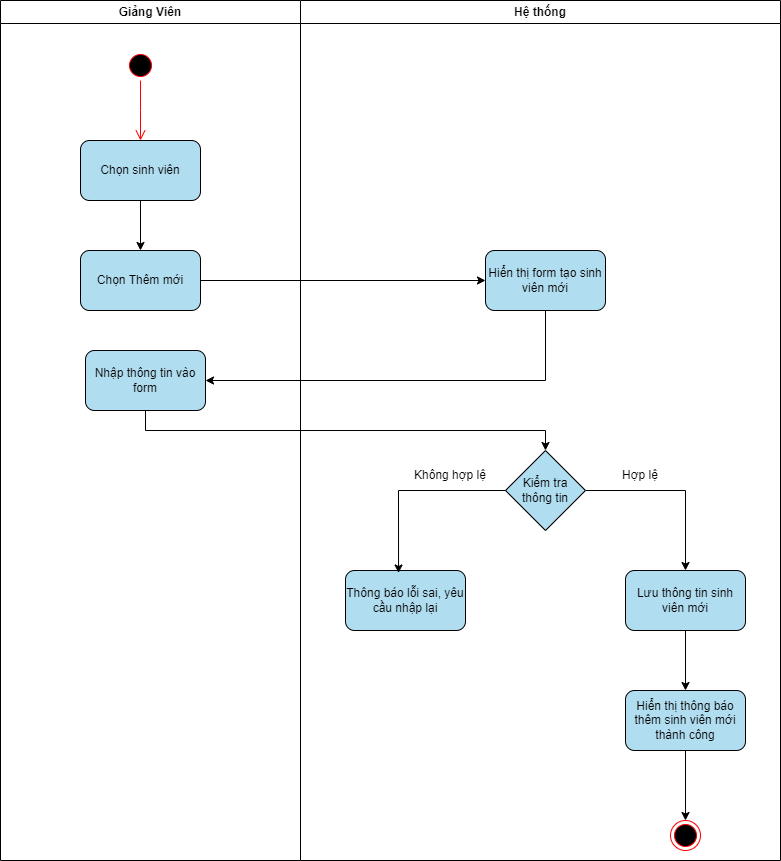


Hình 3.2.4.2. Biểu đồ hoạt động đăng nhập (sinh viên)

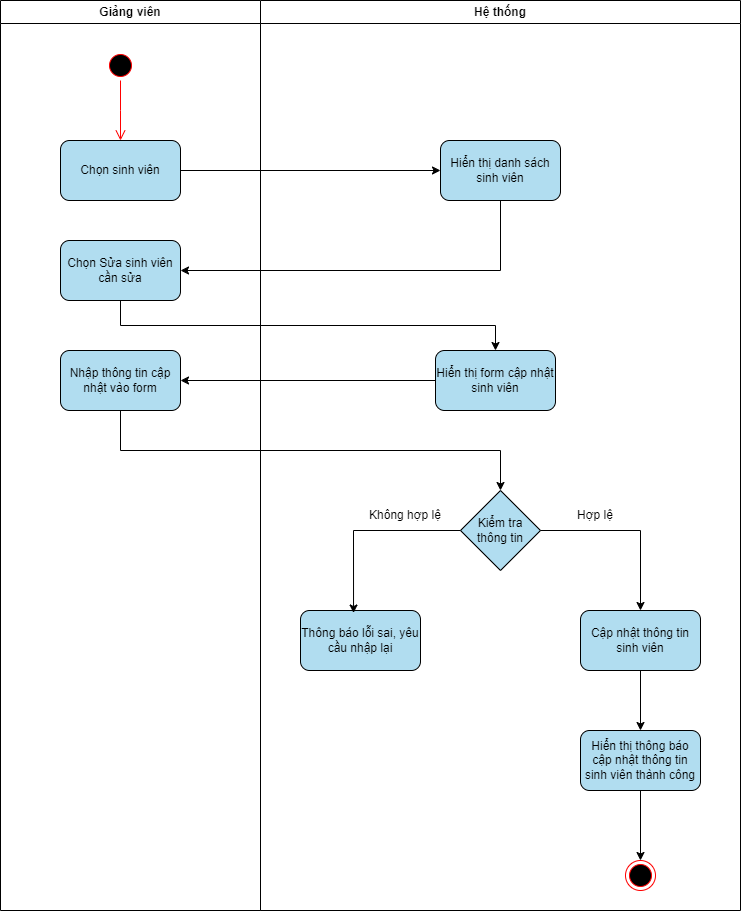
### Biểu đồ hoạt động quản lý sinh viên



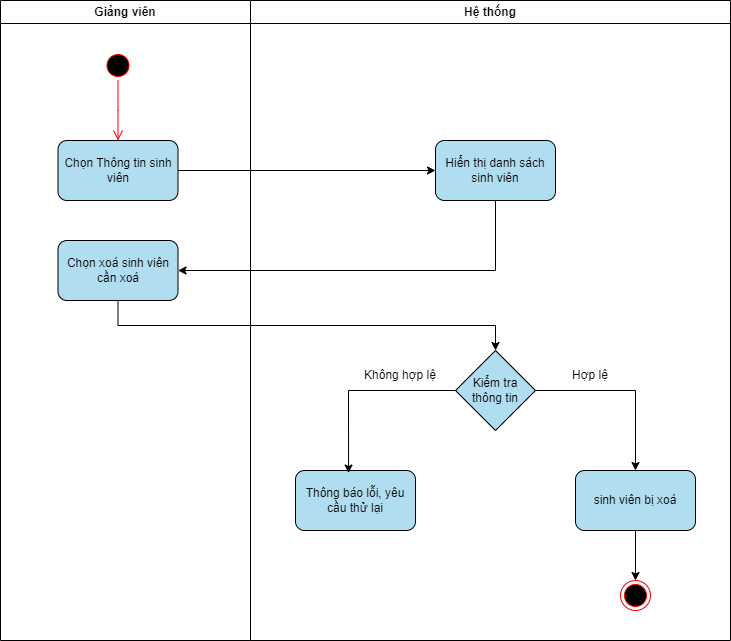
Hình 3.2.5.1. Biểu đồ hoạt động xuất excel danh sách sinh viên



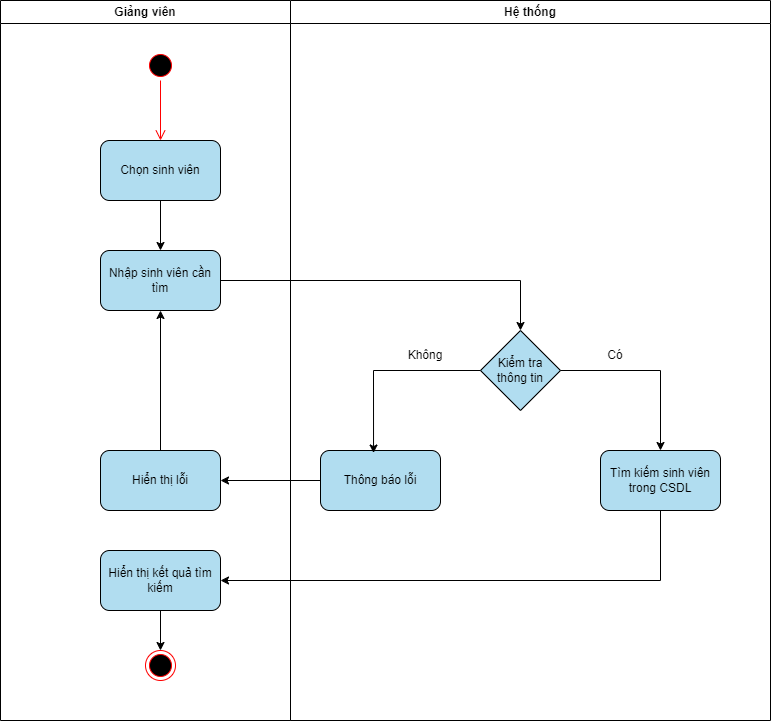
Hình 3.2.5.2. Biểu đồ hoạt động thêm sinh viên



Hình 3.2.5.3. Biểu đồ hoạt động sửa sinh viên

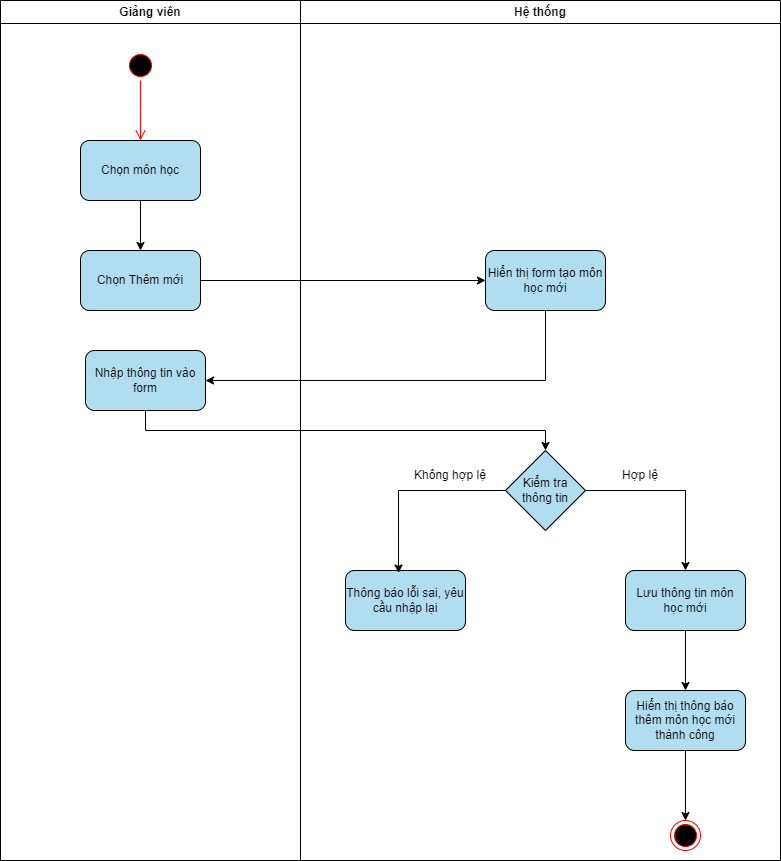


Hình 3.2.5.4. Biểu đồ hoạt động xoá sinh viên

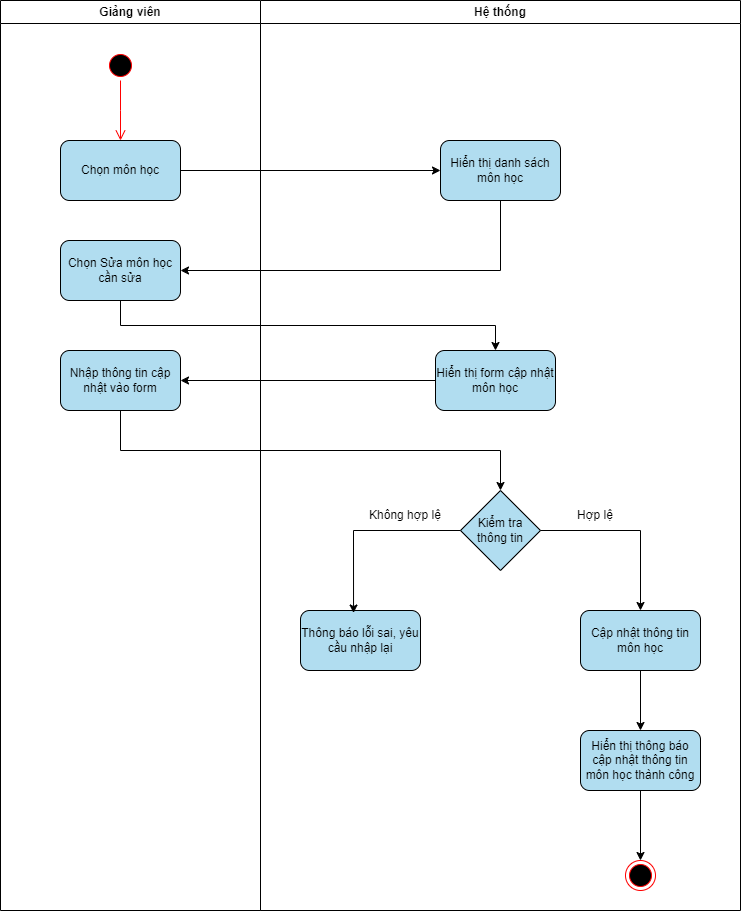


Hình 3.2.5.5. Biểu đồ hoạt động tìm kiếm sinh viên

### Biểu đồ hoạt động quản lý môn học

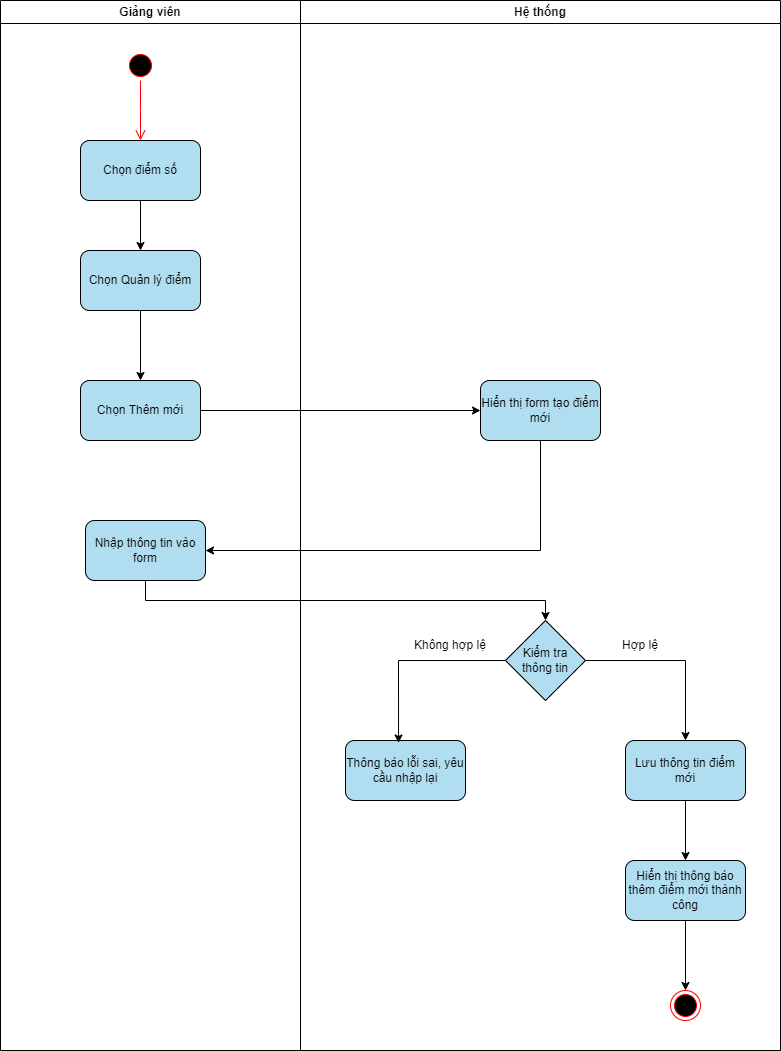


Hình 3.2.6.1. Biểu đồ hoạt động thêm môn học

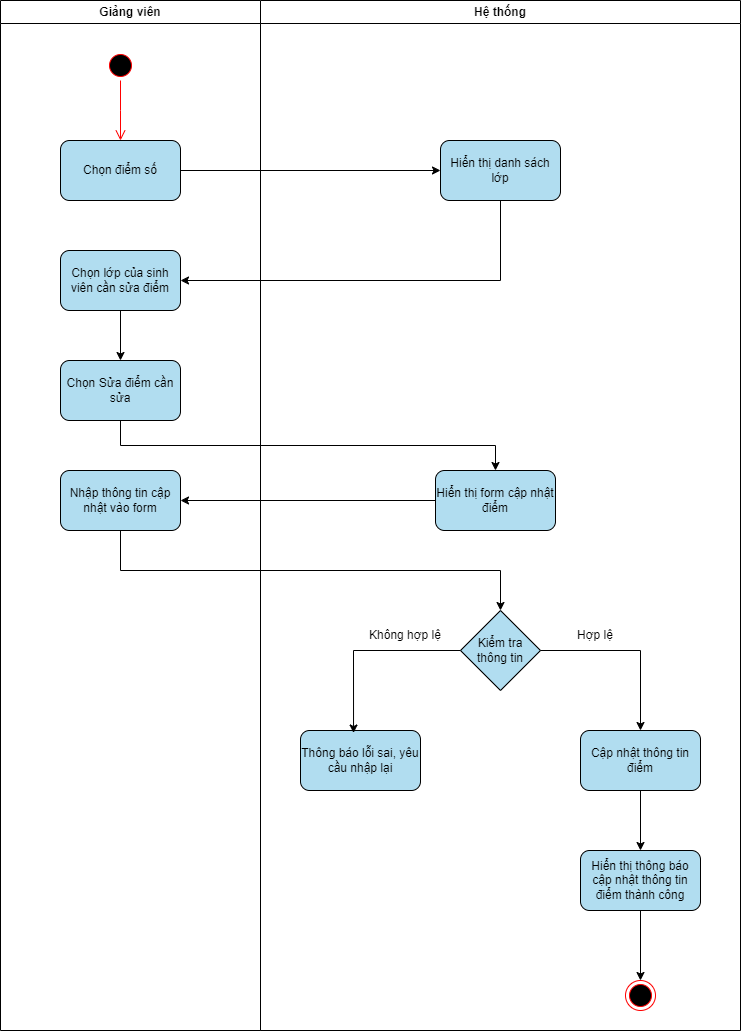


Hình 3.2.6.2. Biểu đồ hoạt động sửa môn học

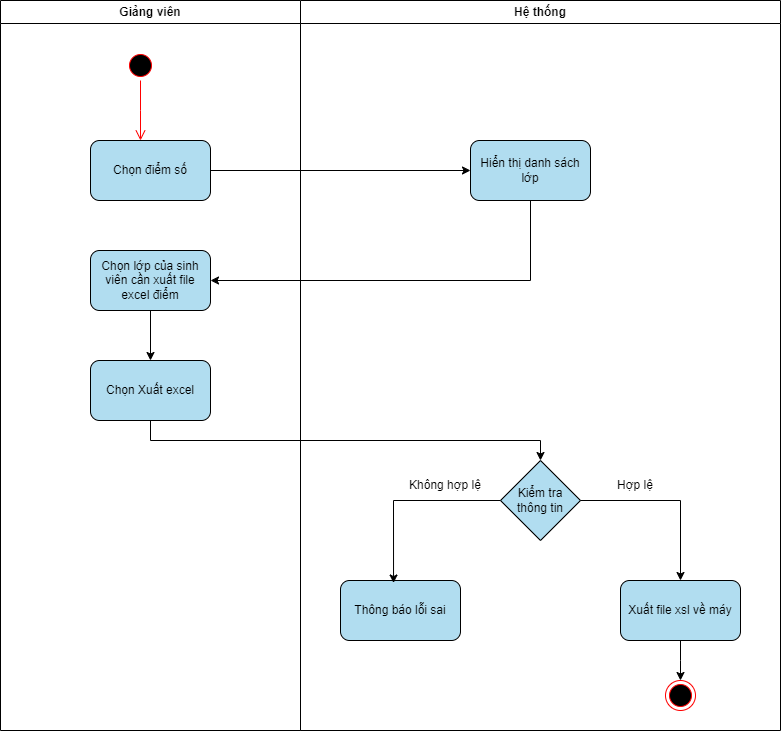
### Biểu đồ hoạt động quản lý điểm số



Hình 3.2.7.1. Biểu đồ hoạt động quản lý thêm điểm

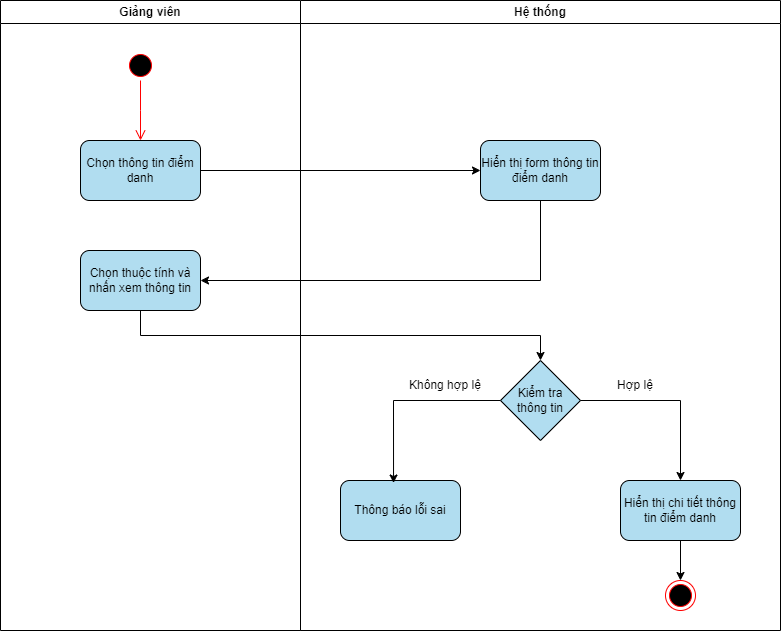
**

Hình 3.2.7.2. Biểu đồ hoạt động quản lý sửa điểm

**

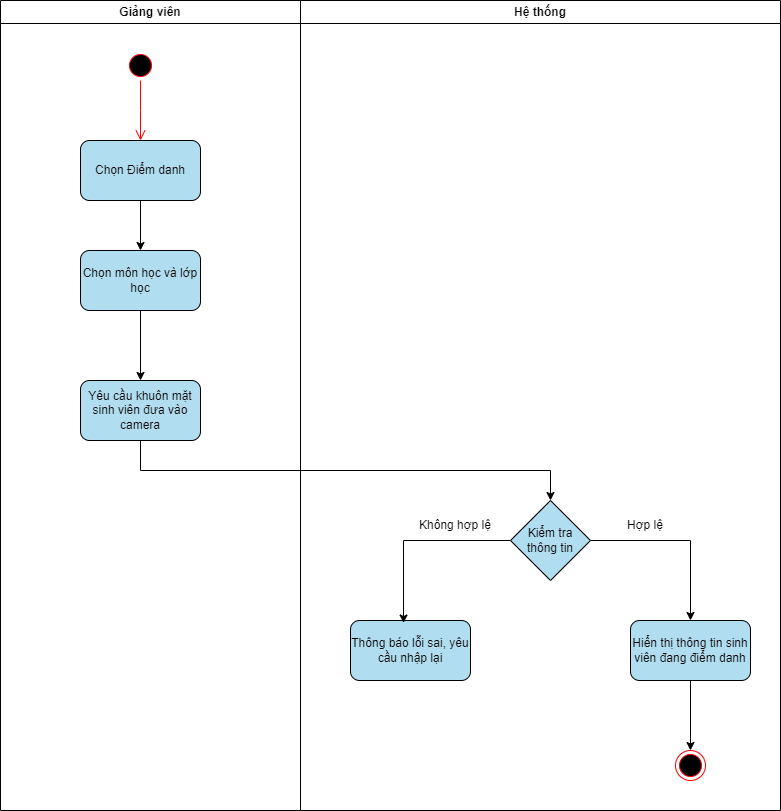
Hình 3.2.7.3. Biểu đồ hoạt động quản lý xuất điểm

### Biểu đồ hoạt động quản lý điểm danh

**

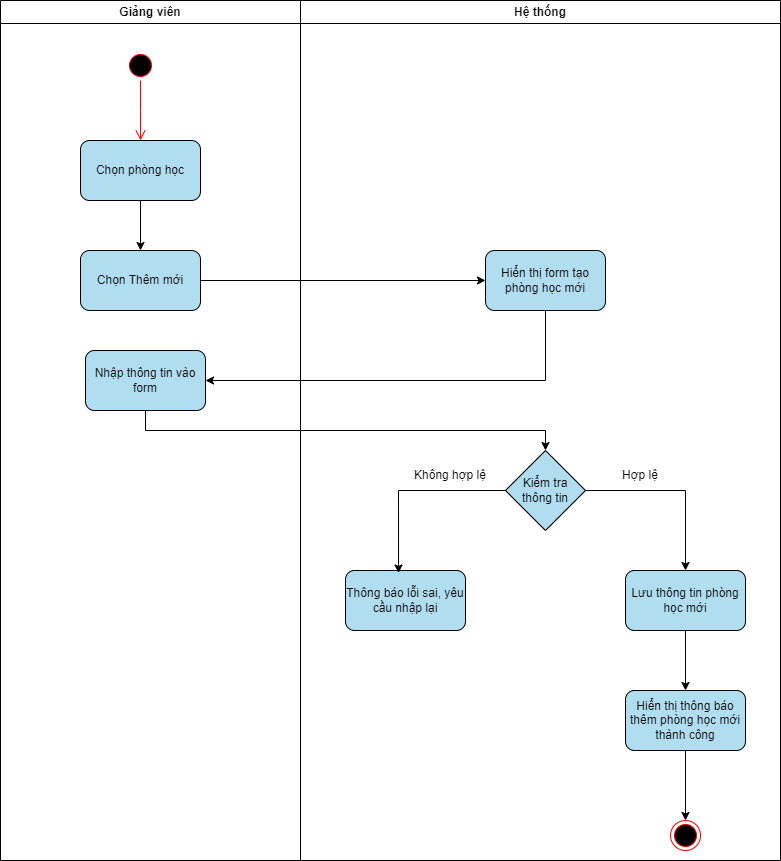
Hình 3.2.8.1. Biểu đồ hoạt động quản lý xem thông tin điểm danh

### Biểu đồ hoạt động điểm danh

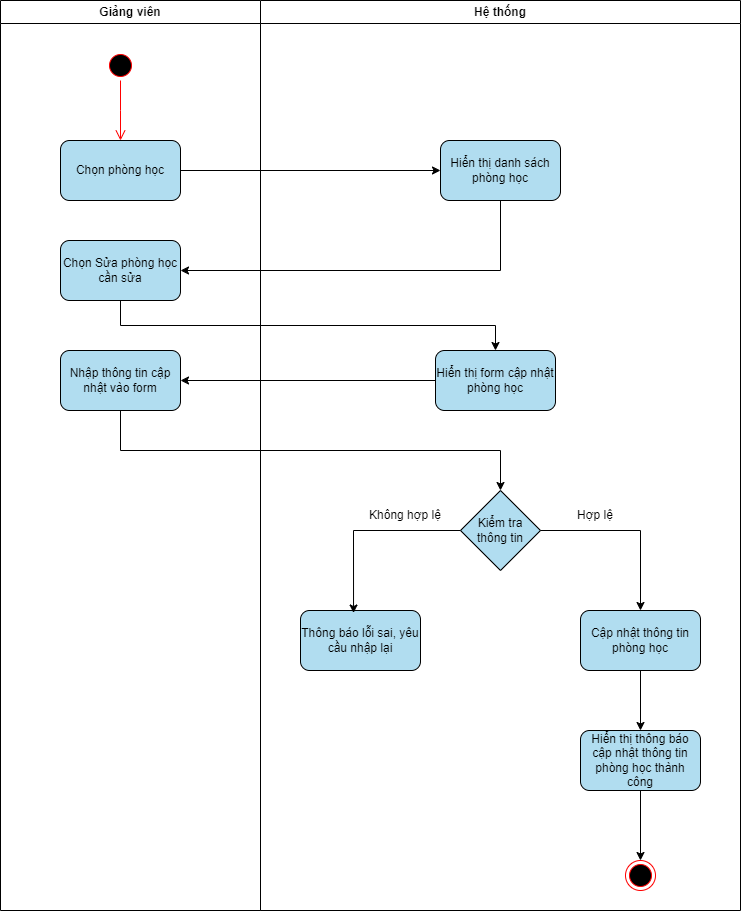


Hình 3.2.9.1. Biểu đồ hoạt động điểm danh khuôn mặt sinh viên

### Biểu đồ hoạt động quản lý phòng học

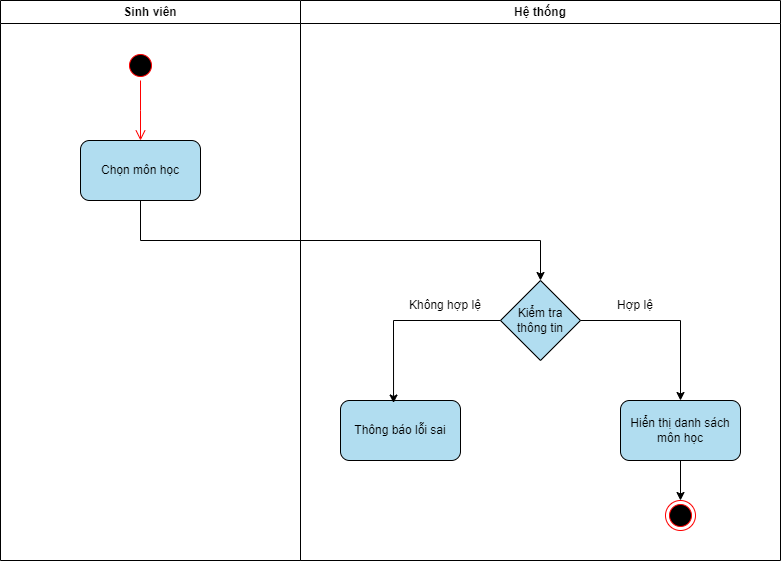


Hình 3.2.10.1. Biểu đồ hoạt động thêm phòng học



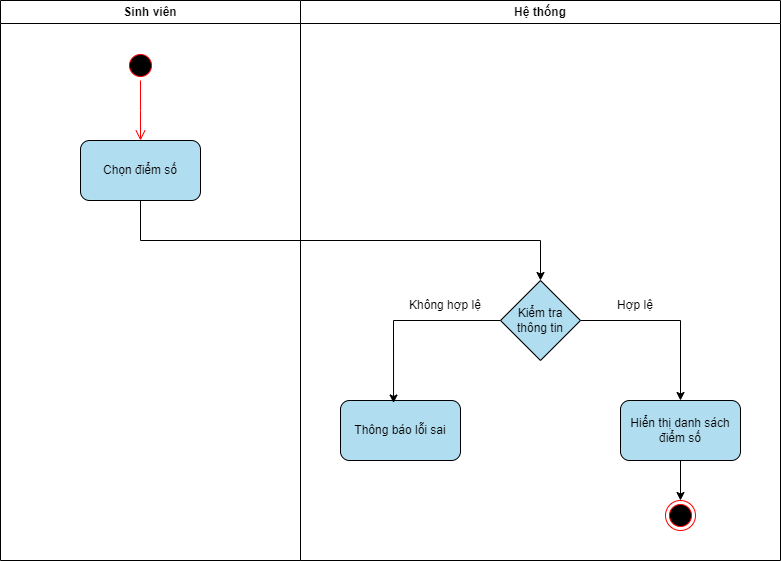
Hình 3.2.10.2. Biểu đồ hoạt động sửa phòng học

### Biểu đồ hoạt động xem thông tin môn học



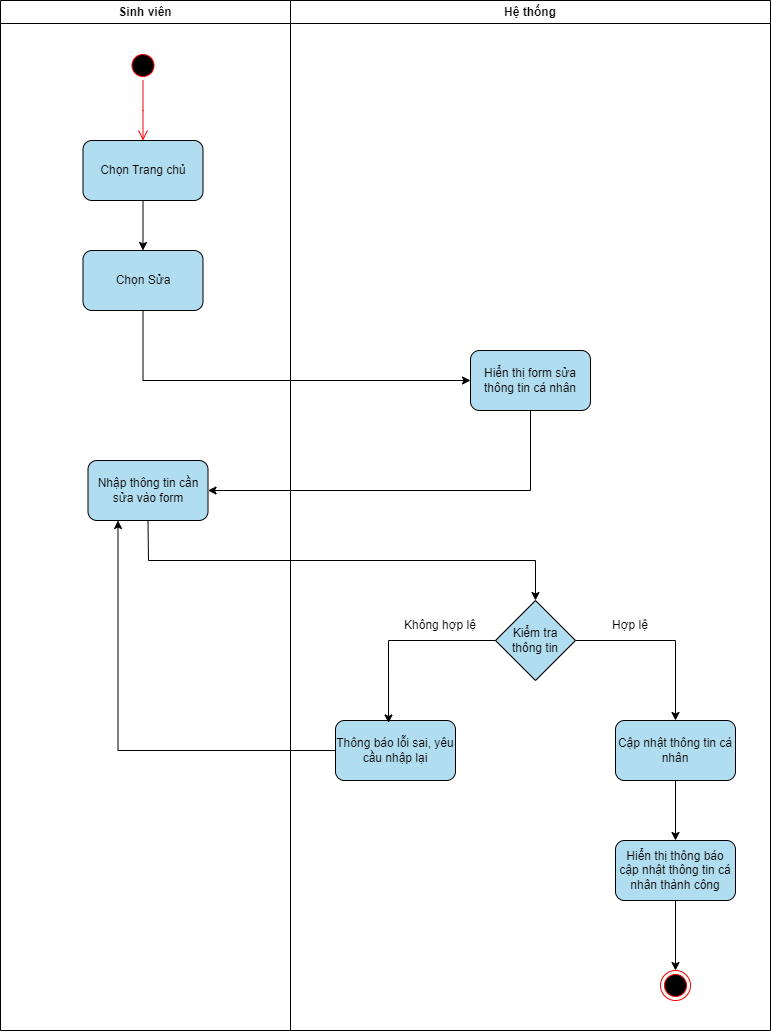
Hình 3.2.11.1. Biểu đồ hoạt động quản lý xem thông tin môn học

### Biểu đồ hoạt động xem điểm số

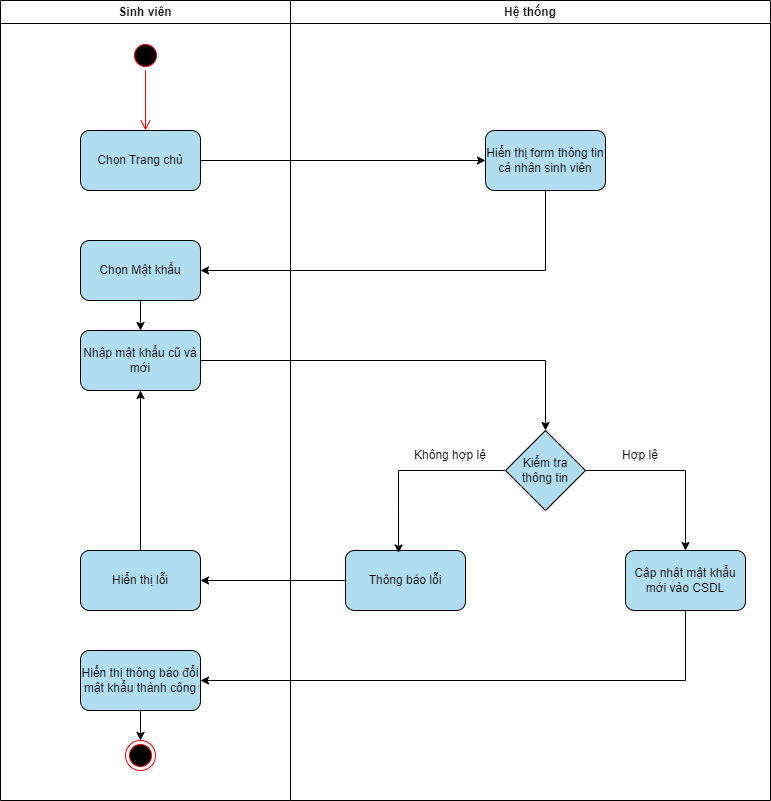


Hình 3.2.12.1. Biểu đồ hoạt động xem điểm số

### Biểu đồ hoạt động quản lý thông tin cá nhân cho sinh viên

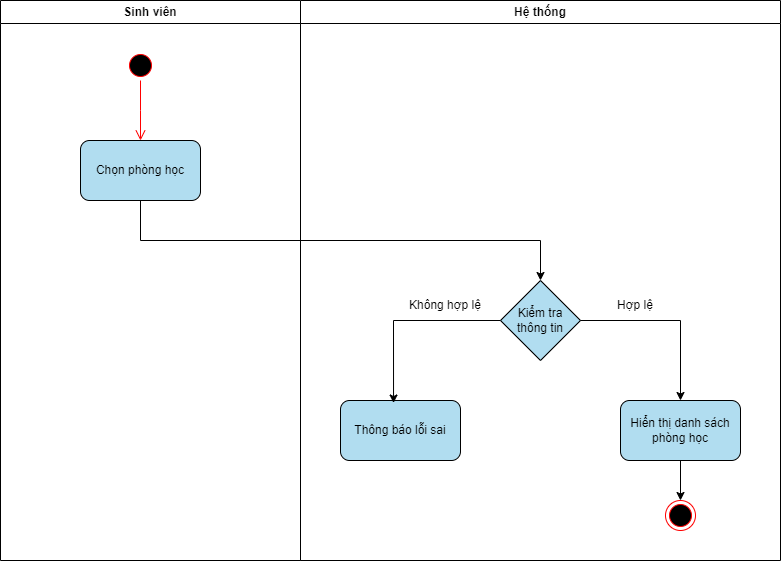


Hình 3.2.13.1. Biểu đồ hoạt động sửa thông tin cá nhân

**

Hình 3.2.13.2. Biểu đồ hoạt động đổi mật khẩu

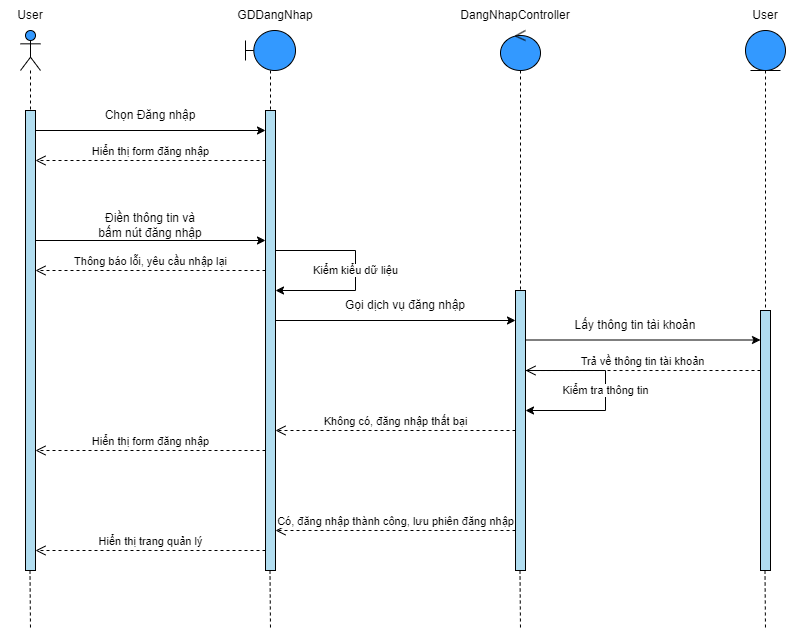
### Biểu đồ hoạt động xem thông tin phòng học



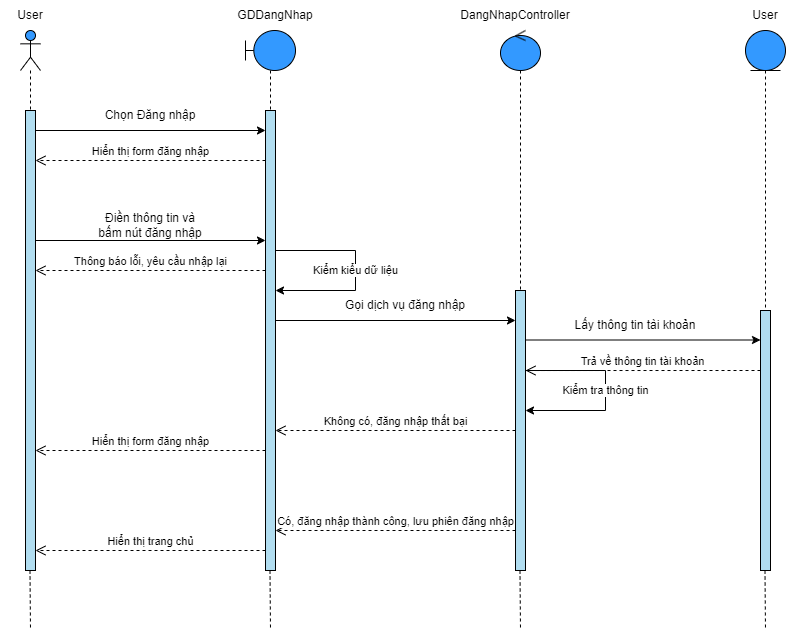
Hình 3.2.14.1. Biểu đồ hoạt động xem thông tin phòng học

## 3.3. Biểu đồ tuần tự

### 3.3.1. Biểu đồ tuần tự đăng nhập tài khoản

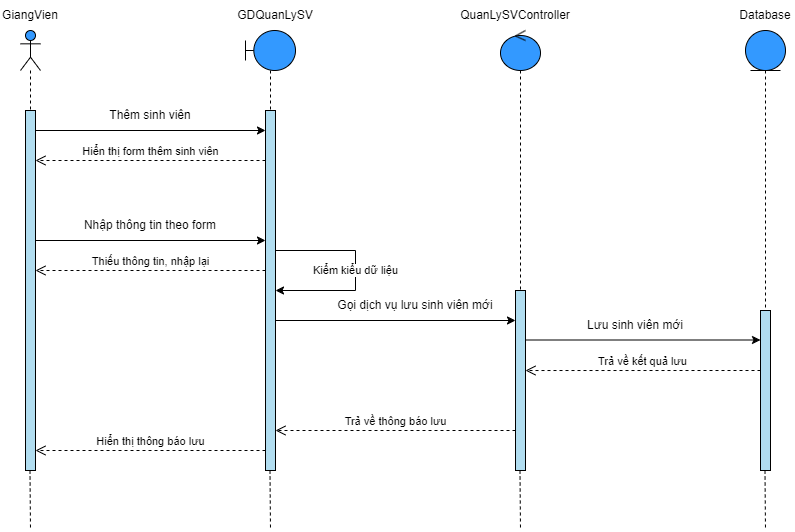


Hình 3.3.1.1. Biểu đồ tuần tự đăng nhập tài khoản (giảng viên)

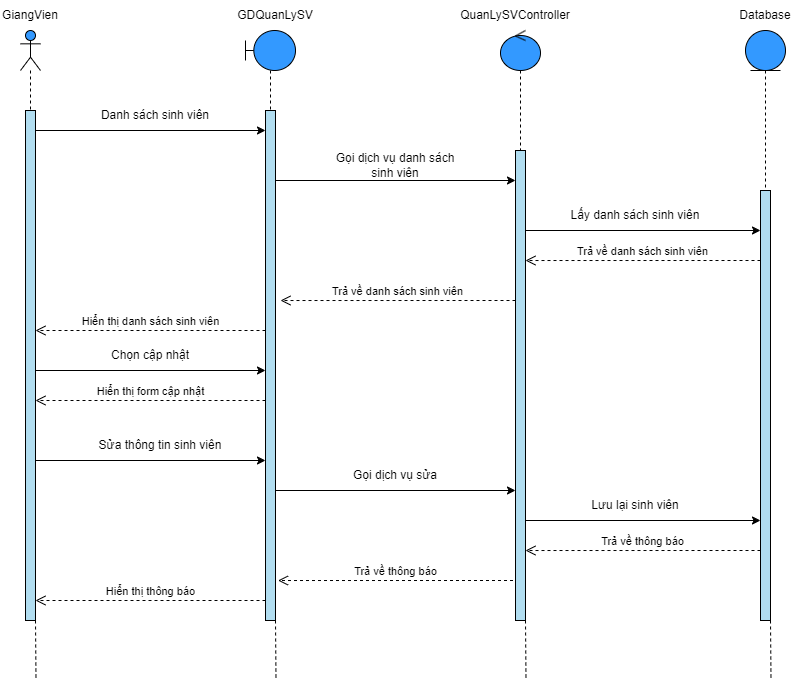
**

Hình 3.3.1.2. Biểu đồ tuần tự đăng nhập tài khoản (sinh viên)

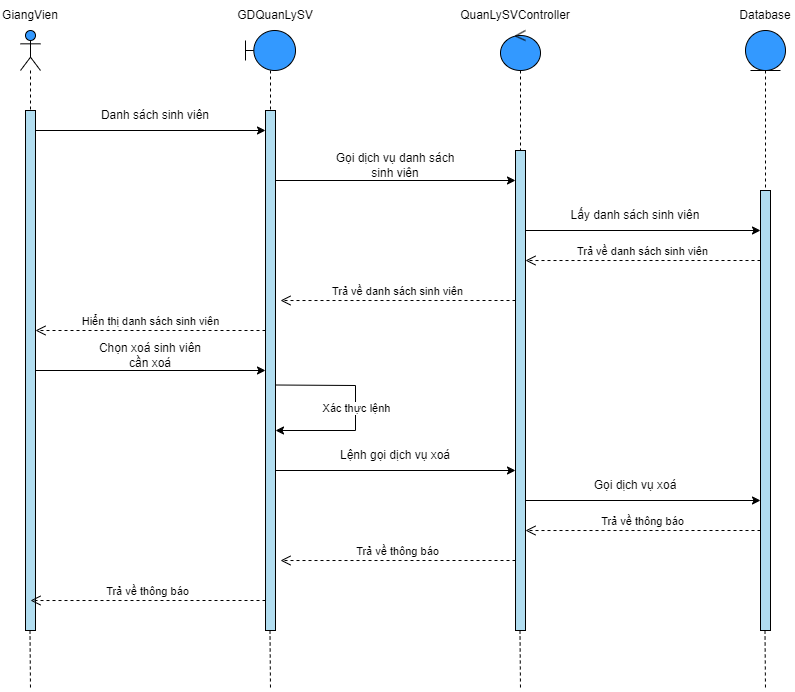
### 3.3.2. Biểu đồ tuần tự quản lý sinh viên



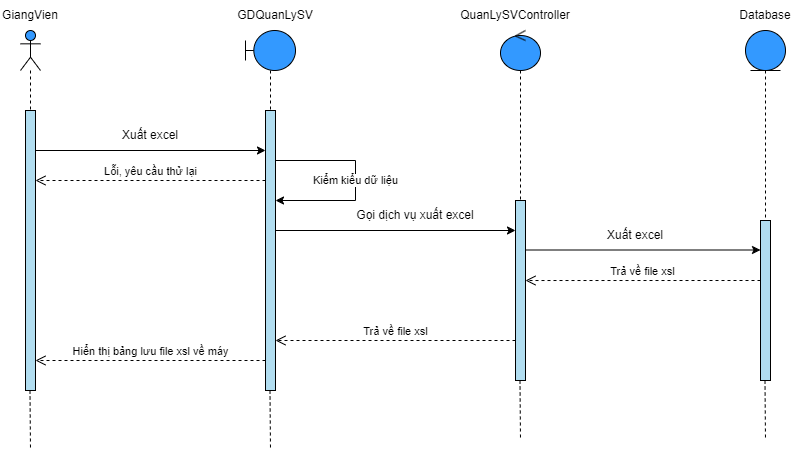
Hình 3.3.2.1. Biểu đồ tuần tự thêm sinh viên

**

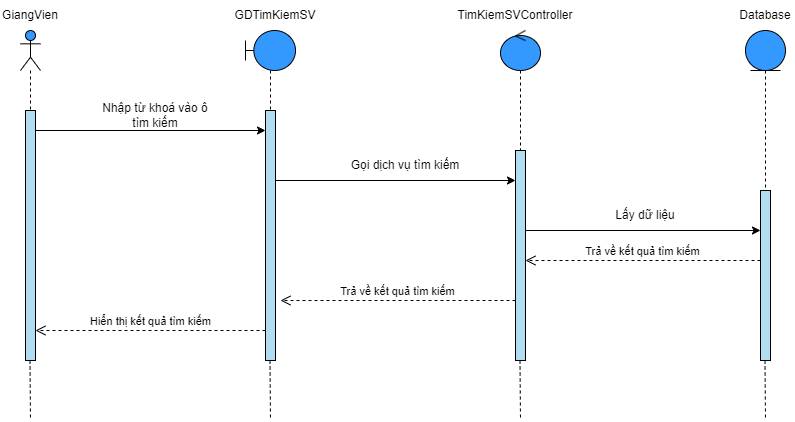
Hình 3.3.2.2. Biểu đồ tuần tự sửa sinh viên

**

Hình 3.3.2.3. Biểu đồ tuần tự xoá sinh viên

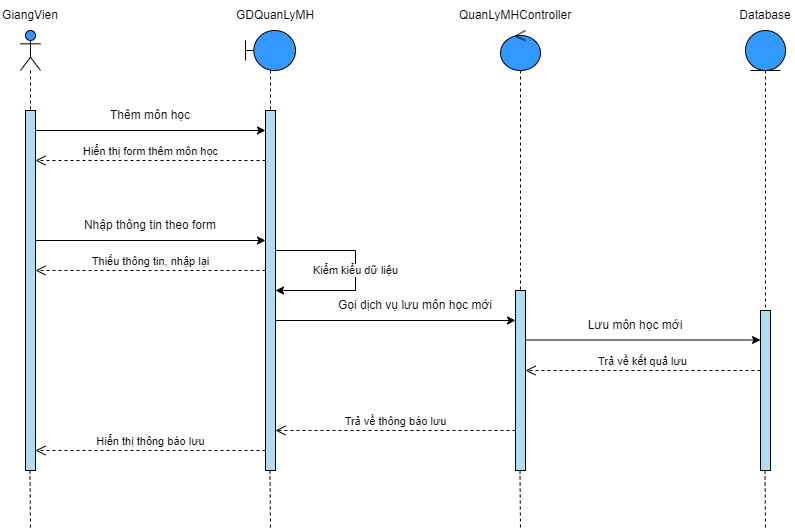


Hình 3.3.2.4. Biểu đồ tuần tự xuất excel danh sách sinh viên

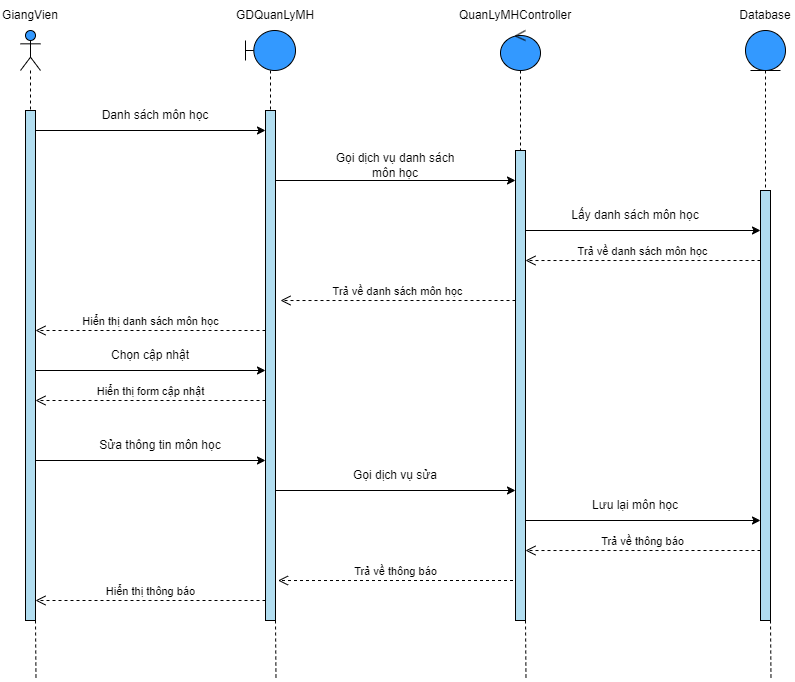


Hình 3.3.2.5. Biểu đồ tuần tự tìm kiếm sinh viên

### 3.3.3. Biểu đồ tuần tự quản lý môn học

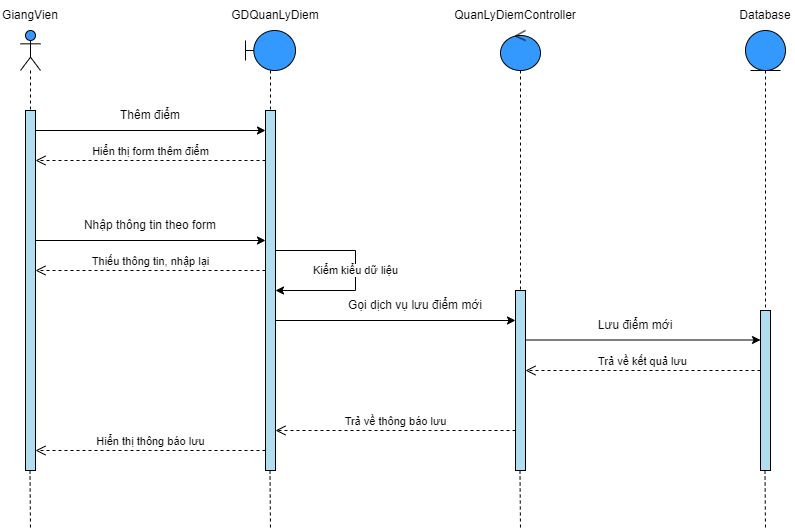
**

Hình 3.3.3.1. Biểu đồ tuần tự quản lý thêm môn học

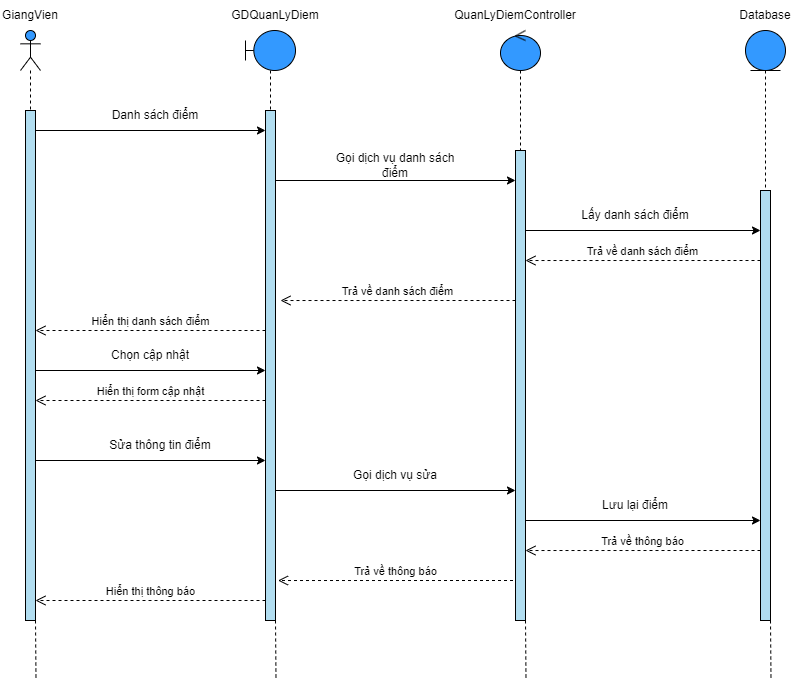
**

Hình 3.3.3.2. Biểu đồ tuần tự quản lý sửa môn học

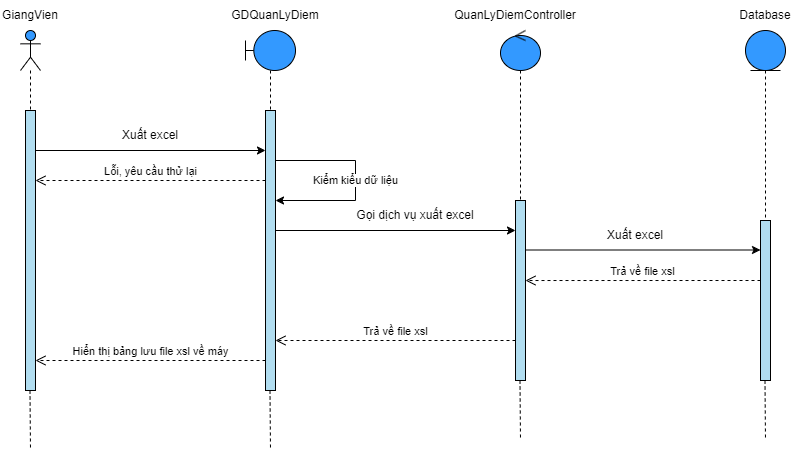
### 3.3.4. Biểu đồ tuần tự quản lý điểm số

**

Hình 3.3.4.1. Biểu đồ tuần tự quản lý thêm điểm

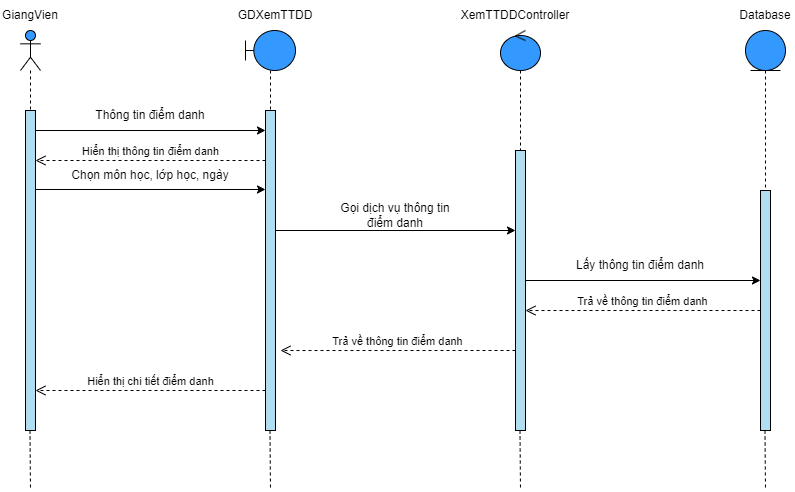
**

Hình 3.3.4.2. Biểu đồ tuần tự quản lý sửa điểm

**

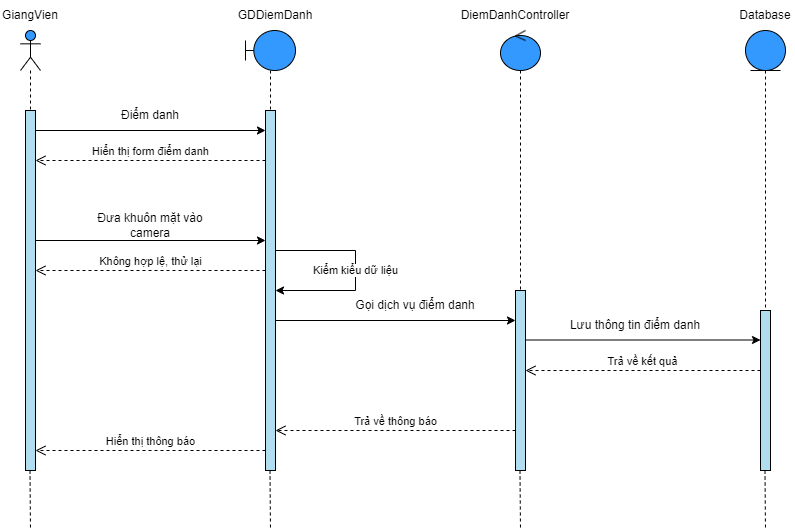
Hình 3.3.4.3. Biểu đồ tuần tự quản lý xuất điểm

### 3.3.5. Biểu đồ tuần tự quản lý điểm danh



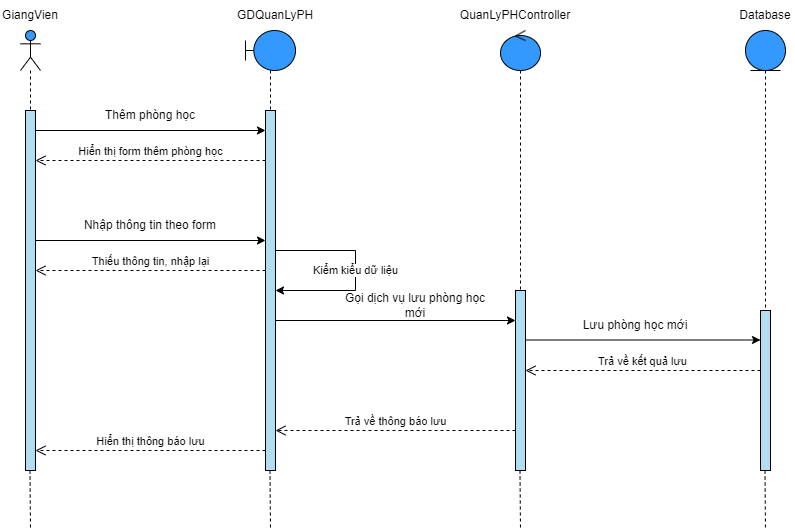
Hình 3.3.5.1. Biểu đồ tuần tự quản lý xem thông tin điểm danh

### 3.3.6. Biểu đồ tuần tự điểm danh

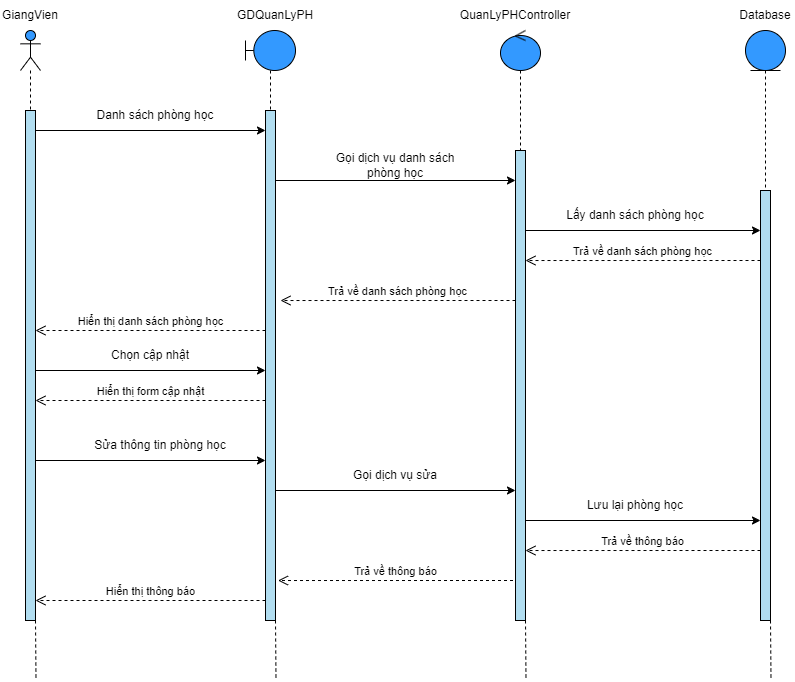


Hình 3.3.6.1. Biểu đồ tuần tự điểm danh khuôn mặt sinh viên

### 3.3.7. Biểu đồ tuần tự quản lý phòng học

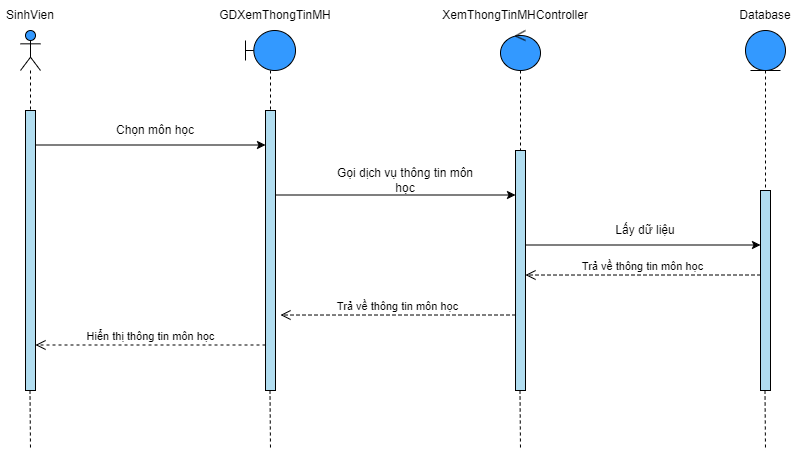


Hình 3.3.7.1. Biểu đồ tuần tự thêm phòng học



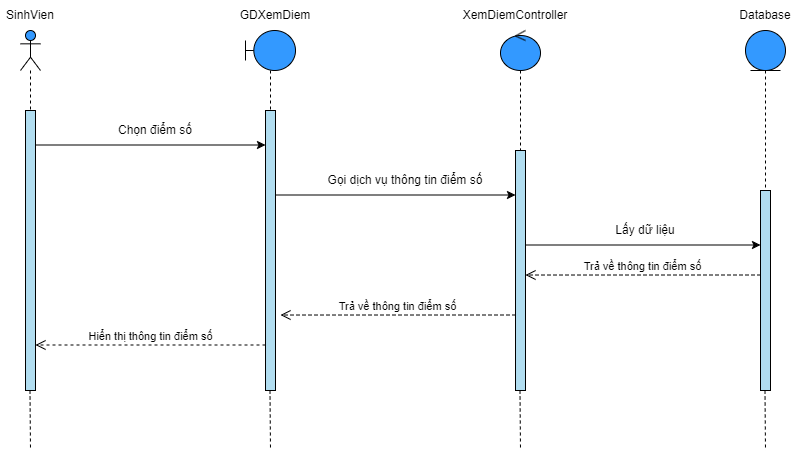
Hình 3.3.7.2. Biểu đồ tuần tự sửa phòng học

### 3.3.8. Biểu đồ tuần tự xem thông tin môn học



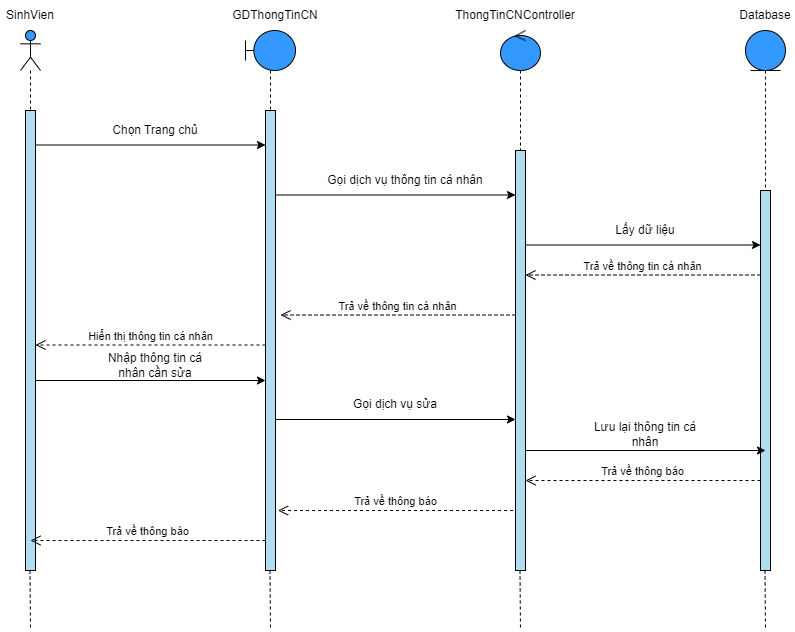
Hình 3.3.8.1. Biểu đồ tuần tự quản lý xem thông tin môn học

### 3.3.9. Biểu đồ tuần tự xem điểm số

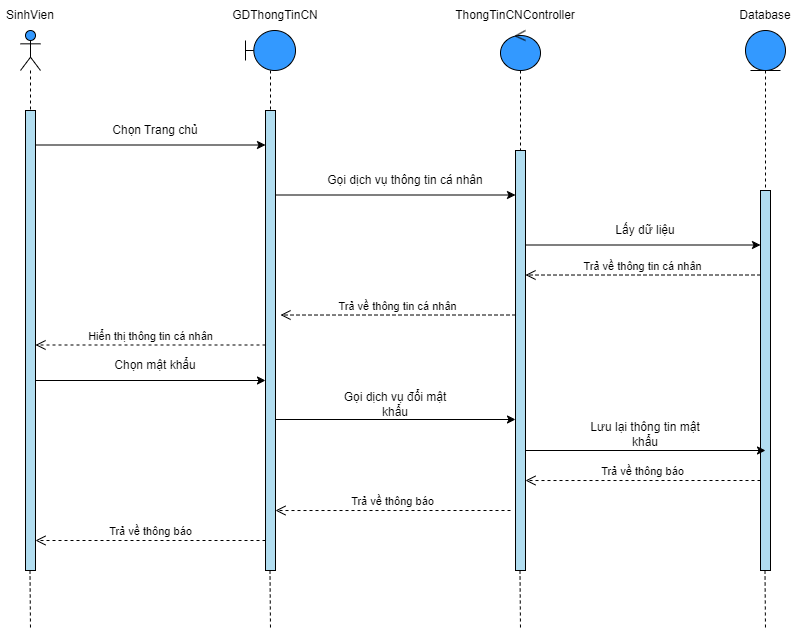


Hình 3.3.9.1. Biểu đồ tuần tự xem điểm số

### 3.3.10. Biểu đồ tuần tự quản lý thông tin cá nhân cho sinh viên

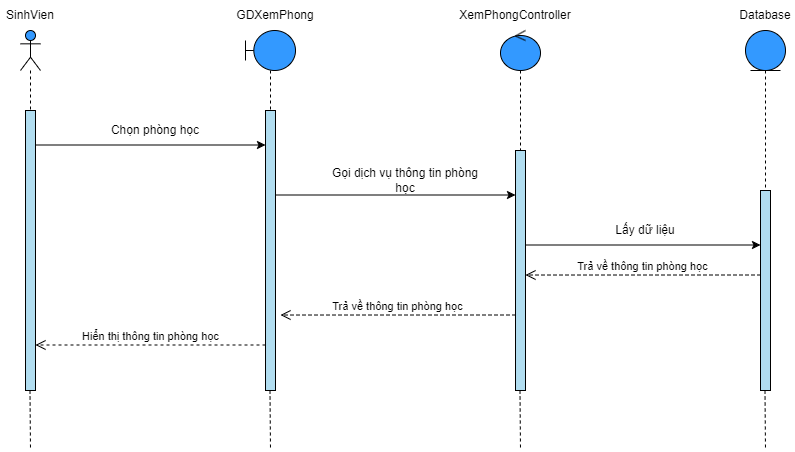


Hình 3.3.10.1. Biểu đồ tuần tự sửa thông tin cá nhân



Hình 3.3.10.2. Biểu đồ tuần tự đổi mật khẩu

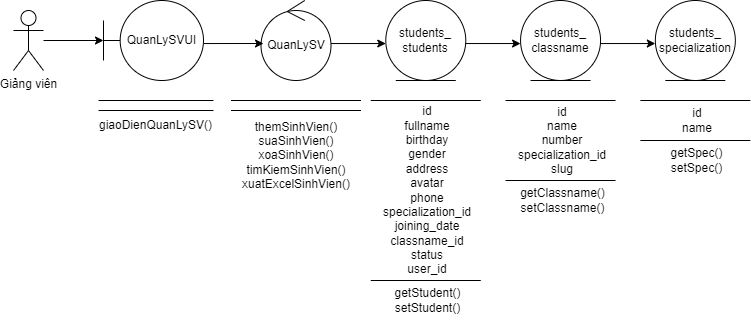
### 3.3.11. Biểu đồ tuần tự xem thông tin phòng học



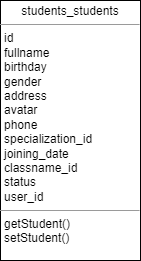
Hình 3.3.11.1. Biểu đồ tuần tự xem thông tin phòng học

## 3.4. Biểu đồ lớp

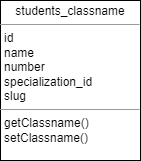
### 3.4.1. Biểu đồ lớp quản lý sinh viên



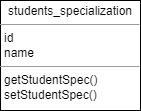
Hình 3.4.2.1. Biểu đồ VOPC quản lý sinh viên



Hình 3.4.2.2. Lớp đối tượng students\_students

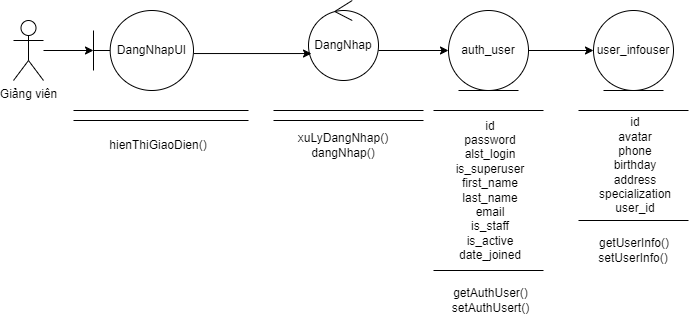


Hình 3.4.2.3. Lớp đối tượng students\_classname

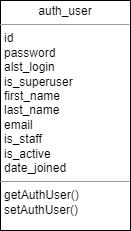


Hình 3.4.2.4. Lớp đối tượng students\_specialization

### 3.4.2. Biểu đồ lớp đăng nhập tài khoản



Hình 3.4.1.1. Biểu đồ VOPC đăng nhập tài khoản

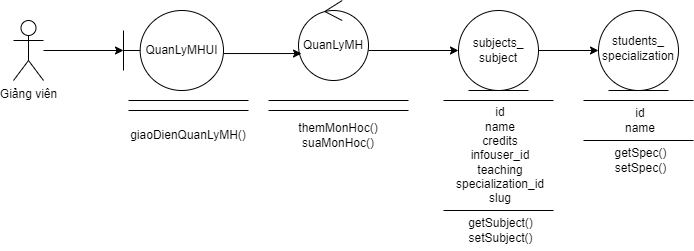


Hình 3.4.1.3. Lớp đối tượng auth\_user

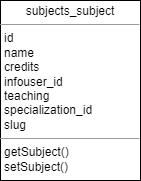


Hình 3.4.1.4. Lớp đối tượng user\_infouser

### 3.4.3. Biểu đồ lớp quản lý môn học

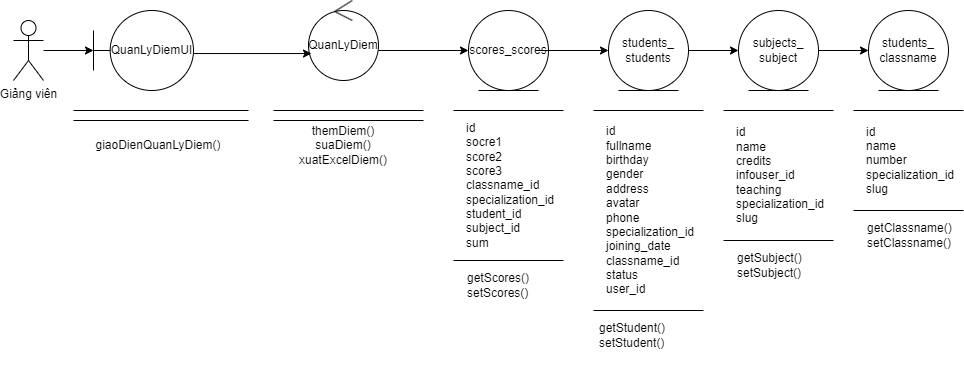


Hình 3.4.3.1. Biểu đồ VOPC quản lý môn học

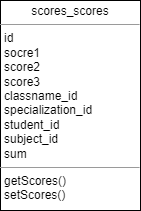


Hình 3.4.3.2. Lớp đối tượng subjects\_subject

### 3.4.4. Biểu đồ lớp quản lý điểm số

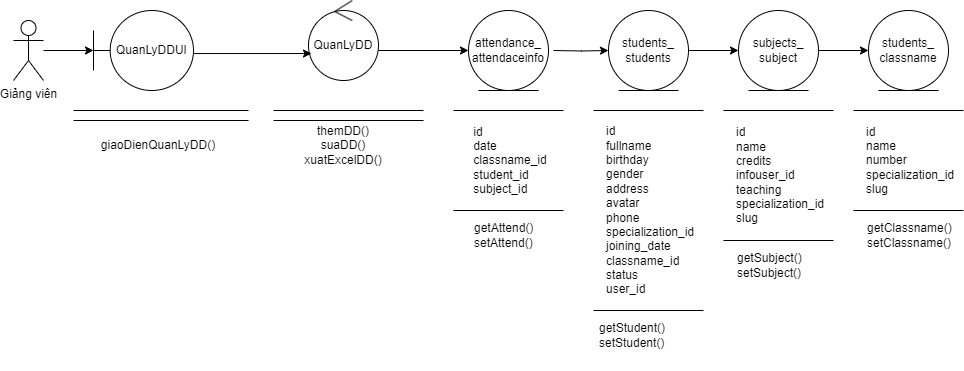


Hình 3.4.4.1. Biểu đồ VOPC quản lý điểm số

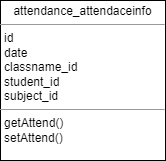


Hình 3.4.4.2. Lớp đối tượng scores\_scores

### 3.4.5. Biểu đồ lớp quản lý điểm danh

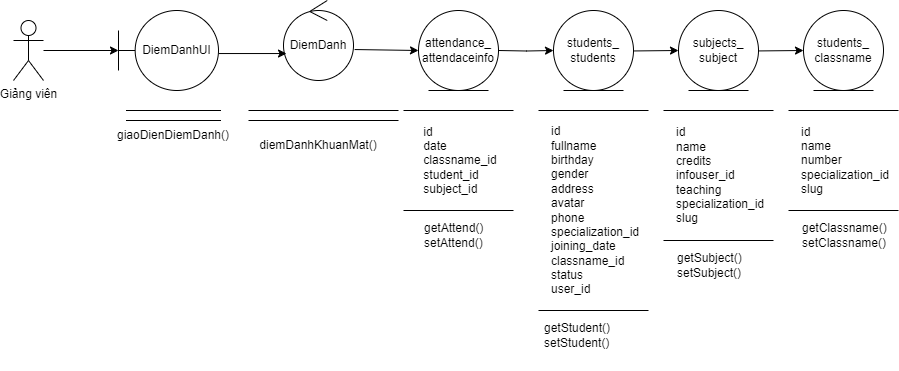


Hình 3.4.5.1. Biểu đồ VOPC quản lý điểm danh



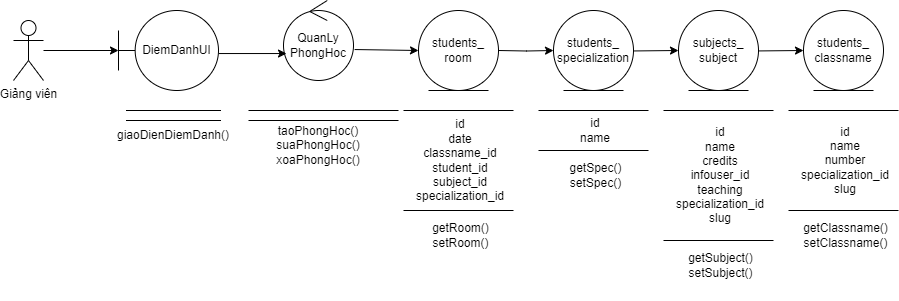
Hình 3.4.5.2. Lớp đối tượng attendance\_attendaceinfo

### 3.4.6. Biểu đồ lớp điểm danh sinh viên

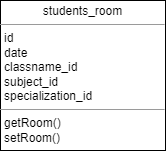


Hình 3.4.6.1. Biểu đồ VOPC điểm danh sinh viên

### 3.4.7. Biểu đồ lớp quản lý phòng học

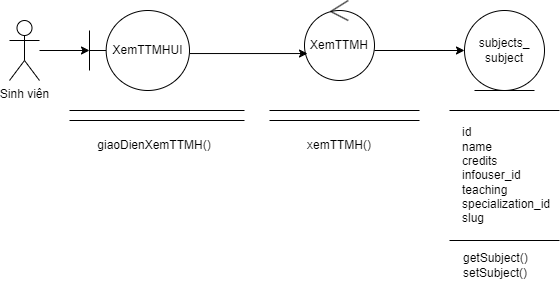


Hình 3.4.7.1. Biểu đồ VOPC quản lý phòng học



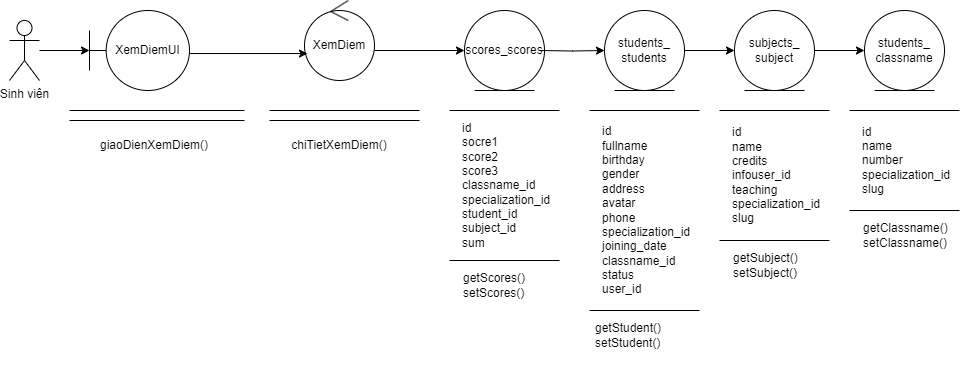
Hình 3.4.7.2. Lớp đối tượng students\_room

### 3.4.8. Biểu đồ lớp xem thông tin môn học



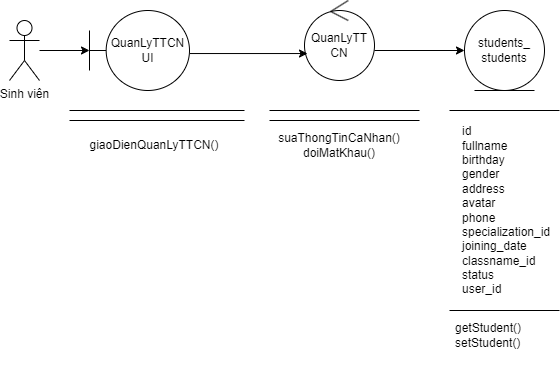
Hình 3.4.8.1. Biểu đồ VOPC xem thông tin môn học

### 3.4.9. Biểu đồ lớp xem điểm số



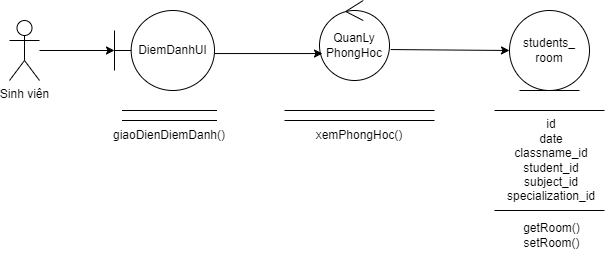
Hình 3.4.9.1. Biểu đồ VOPC xem điểm số

### 3.4.10. Biểu đồ lớp quản lý thông tin cá nhân cho sinh viên



Hình 3.4.10.1. Biểu đồ VOPC quản lý thông tin cá nhân cho sinh viên

### 3.4.11. Biểu đồ lớp xem thông tin phòng học

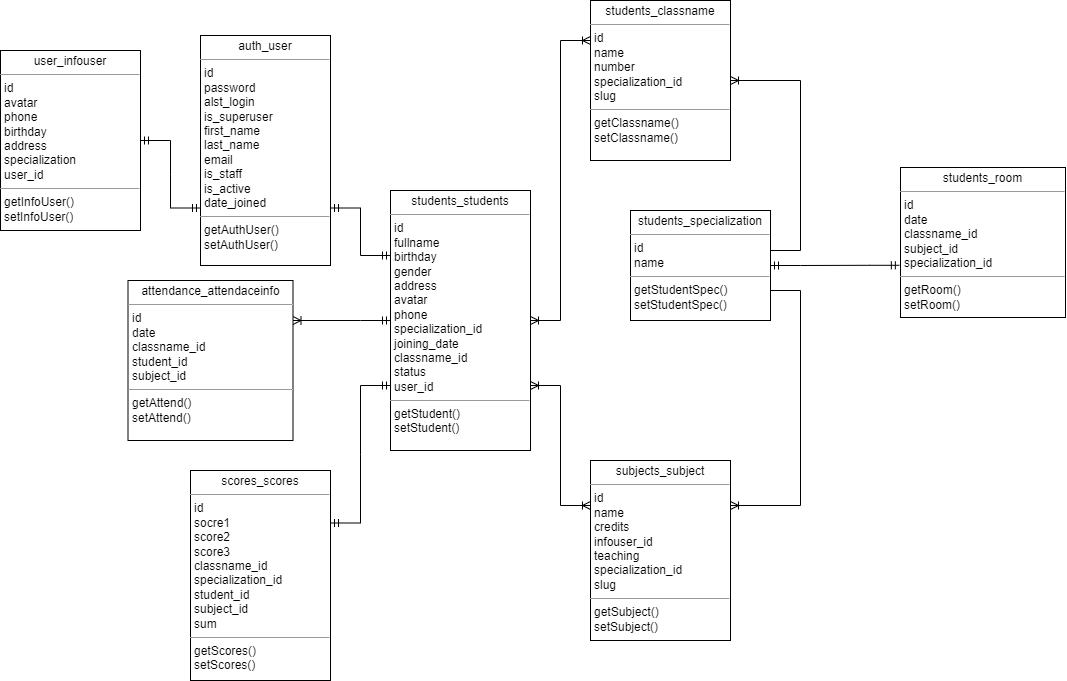


Hình 3.4.11.1. Biểu đồ VOPC xem thông tin phòng học

# CHƯƠNG 4. TRIỂN KHAI VÀ XÂY DỰNG HỆ THỐNG

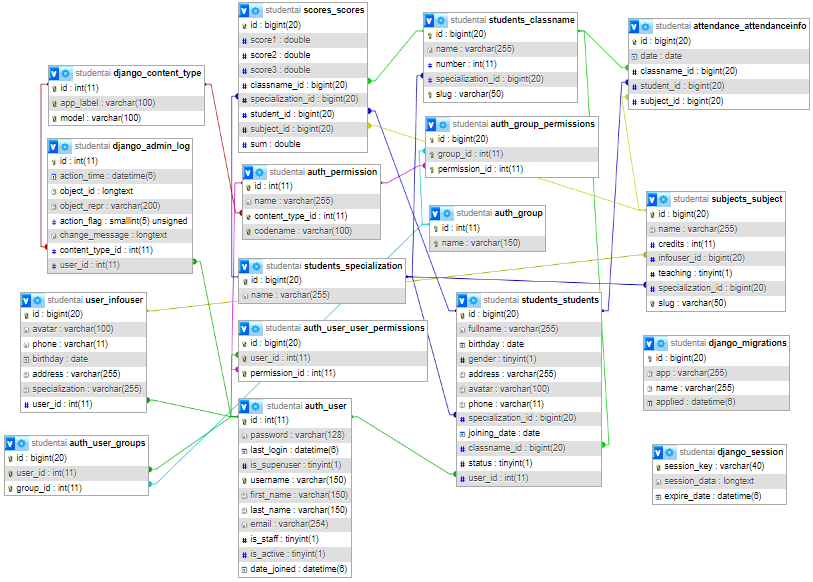
## Xây dựng cơ sở dữ liệu

### Biểu đồ lớp tổng quát



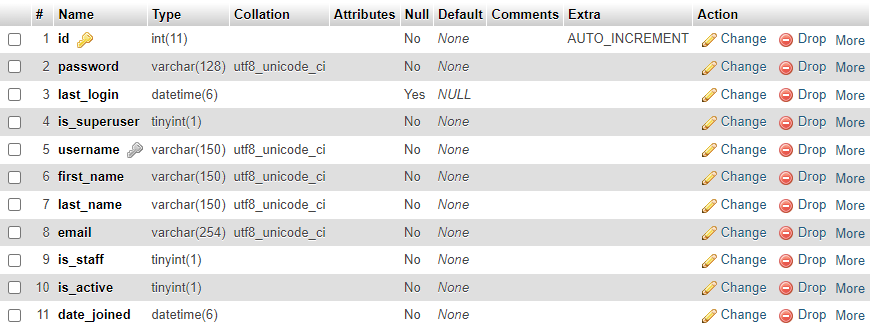
Hình 4.1.1.1. Biểu đồ lớp tổng quát

### Biểu đồ Diagram toàn hệ thống

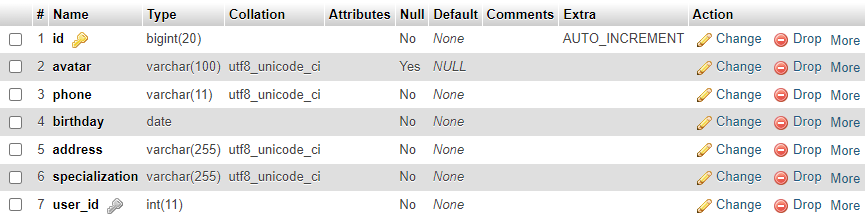


Hình 4.1.2.1. Biểu đồ Diagram toàn hệ thống

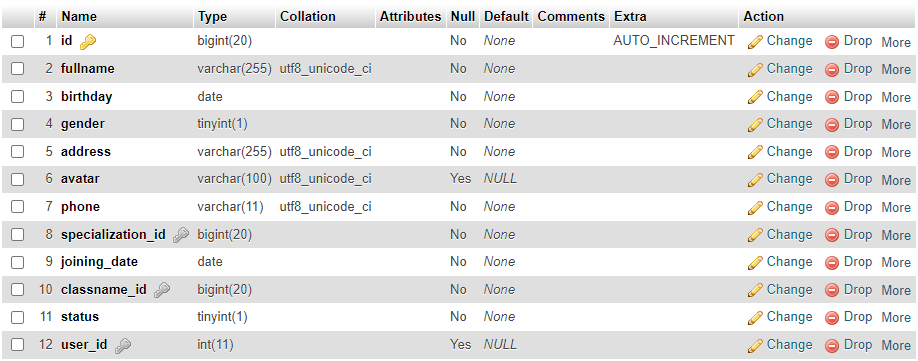
### Cấu trúc các bảng



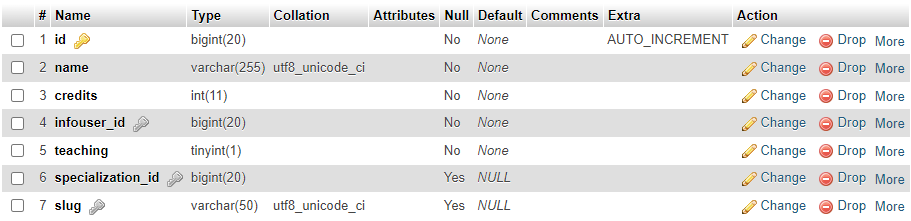
Hình 4.1.3.1. Cấu trúc bảng auth\_user



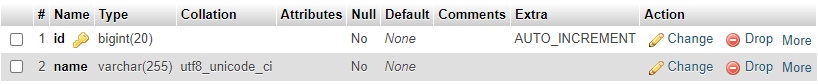
Hình 4.1.3.2. Cấu trúc bảng user\_info



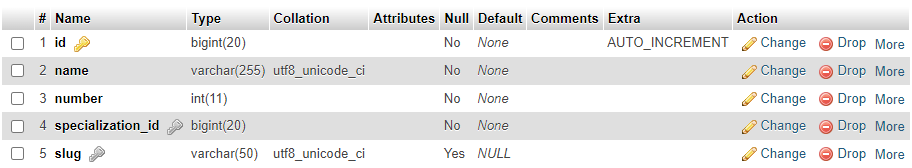
Hình 4.1.3.3. Cấu trúc bảng students\_students



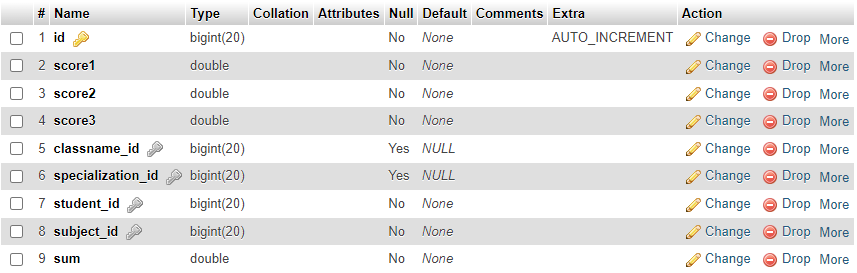
Hình 4.1.3.4. Cấu trúc bảng subjects\_subject



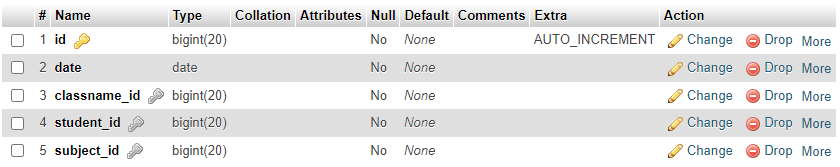
Hình 4.1.3.5. Cấu trúc bảng students\_specialization



Hình 4.1.3.6. Cấu trúc bảng students\_classname



Hình 4.1.3.7. Cấu trúc bảng scores\_scores



Hình 4.1.3.8. Cấu trúc bảng attendance\_attendanceinfo

## Xây dựng giao diện

### Giao diện dành cho giảng viên

Hình 4.2.1.1. Giao diện chức năng đăng nhập

Hình 4.2.1.2. Giao diện trang chủ

Hình 4.2.1.3. Giao diện quản lý sinh viên

Hình 4.2.1.4. Giao diện quản lý môn học

Hình 4.2.1.5. Giao diện quản lý điểm số

Hình 4.2.1.6. Giao diện điểm danh

Hình 4.2.1.7. Giao diện quản lý phòng học

### Giao diện dành cho sinh viên

Hình 4.2.2.1. Giao diện đăng nhập

Hình 4.2.2.2. Giao diện xem thông tin môn học

Hình 4.2.2.3. Giao diện xem điểm số

Hình 4.2.2.4. Giao diện quản lý thông tin cá nhân

Hình 4.2.2.5. Giao diện xem thông tin phòng học

# KẾT LUẬN

**Kết quả đạt được**

Trong quá trình tìm hiểu nội dung, thực hiện bài báo cáo và xây dựng hệ thống website cho đồ án tốt nghiệp của mình, em đã tiếp thu và củng cố thêm được nhiều kiến thức liên quan đến việc xây dựng một hệ thống quản lý thông tin sinh viên trong một trường đại học có sự tích hợp của điểm danh sinh viên dựa trên công nghệ trí tuệ nhân tạo – thông qua đó biết cách triển khai các mô hình học máy và triển khai các mô hình học máy vào trong sản phẩm thực tế.

Ngoài ra, việc thực hiện đồ án giúp em nắm được rõ hơn quy trình khi sản xuất một hệ thống website quản lý, hiểu biết thêm các mô hình khi phát triển các hệ thống website, ngoài ra nắm chắc hơn các kiến thức về cấu hình, kết nối và lập trình trên môi trường máy chủ (server) và kết nối tới hệ quản trị cơ sở dữ liệu!

**Nội dung cần bổ sung**

Với sự cố gắng, nỗ lực, em đã cố gắng hoàn thành yêu cầu đề tài tốt nghiệp được chọn. Do thời gian của đồ án tốt nghiệp có hạn nên phần mềm được xây dựng đang chỉ có các chức năng đơn giản của một phần hệ thống quản lý sinh viên cho trường đại học TMU, với các chức năng cơ bản như: quản lý sinh viên, quản lý môn học, quản lý điểm danh, quản lý điểm số và một chức năng đặc biệt đó là điểm danh cho sinh viên dựa trên khuân mặt của sinh viên viên.

Trong tương lai, em sẽ cần phải phát triển thêm nhiều chức năng hơn để hệ thống có thể đáp ứng cho nhiều trường đại học khác cùng sử dụng và đáp ứng được với nhu cầu thực tế. Các chức năng sẽ cần bổ sung như:

* Chức năng tra cứu thông tin liên quan đến sinh viên dành cho phụ huynh
* Chức năng phòng họp trực tuyến cho sinh viên và giảng viên
* Xây dựng hệ thống gợi ý các phương pháp dạy học dựa vào điểm số của sinh viên
* Xây dựng chức năng chatbot dành cho sinh viên và giảng viên
* Chức năng tin tức để cập nhật các thông báo của trường tới sinh viên

# TÀI LIỆU THAM KHẢO

[1] TTTH ĐH KHTN HCM - Tài liệu Giáo trình Lập trình Python 4 Modules

[2] Tác giả. Chaitanya Baweja - Python and MySQL Database: A Practical Introduction

[3] Tác giả. Bùi Việt Hà - Python Cơ Bản (Tái Bản 2023)

[4] Các trang web, diễn đàn:

* https://www.w3schools.com/python/default.asp
* <https://www.python.org/>
* <https://www.djangoproject.com/>
* <https://toidicode.com/python-co-ban>
* http://sinhvienit.net
* <https://howkteam.vn/course/lap-trinh-python-co-ban-37>