

**TRƯỜNG ĐẠI HỌC ĐIỆN LỰC
KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO
ĐỒ ÁN LẬP TRÌNH WEB NÂNG CAO**

ĐỀ TÀI:

XÂY DỰNG WEBSITE QUẢN LÝ SINH VIÊN

Sinh viên thực hiện : **PHẠM VĂN HÙNG**
Mã sinh viên : **1681310034**
Giảng viên hướng dẫn : **PHƯƠNG VĂN CẢNH**
Ngành : **CÔNG NGHỆ THÔNG TIN**
Chuyên ngành : **CÔNG NGHỆ PHẦN MỀM**
Lớp : **D11 CNPM**

Hà Nội, tháng 07 năm 2020

PHIẾU CHẤM ĐIỂM

Sinh viên thực hiện:

Họ và tên	Chữ ký	Ghi chú
Phạm Văn Hùng		

Giảng viên chấm:

Họ và tên	Chữ ký	Ghi chú
Giảng viên chấm 1 :		
Giảng viên chấm 2 :		

MỤC LỤC

PHIẾU CHẤM ĐIỂM	2
MỤC LỤC	3
LỜI CẢM ƠN	6
CHƯƠNG 1: KHẢO SÁT HỆ THỐNG VÀ CÔNG NGHỆ SỬ DỤNG	1
1.1. Tìm hiểu về website.....	1
1.2. Công nghệ sử dụng.	1
1.2.1 Tổng quan về NodeJS.....	1
1.2.2 Khái niệm về NodeJS	2
1.2.3 Tổng quan về ExpressJS	4
1.2.4 Tổng quan về MySQL	5
1.2.5 Tổng quan về Reactjs	6
CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG	8
2.1. Các tác nhân của hệ thống	8
2.2. Các chức năng của hệ thống	8
2.3. Biểu đồ usecase.....	9
2.3.1. Biểu đồ use case chính.....	9
2.3.2. Biểu đồ use case người dùng	10
2.3.3. Biểu đồ use case đăng nhập.....	10
2.3.4. Biểu đồ use case phân quyền.....	11
2.3.5. Biểu đồ use case tìm kiếm	12
2.3.6. Biểu đồ use điểm danh.....	13
2.3.7. Biểu đồ Use case bảng điểm.....	14
2.3.8. Biểu đồ use case quản lý sinh viên.....	15

2.4. Biểu đồ trình tự	16
2.4.1. Biểu đồ trình tự “Đăng nhập”	16
2.4.2. Biểu đồ trình tự phân quyền	17
2.4.3. Biểu đồ trình tự quản lý điểm danh	18
2.4.4. Biểu đồ trình tự quản lý bảng điểm	19
2.4.5. Biểu đồ trình tự quản lý sinh viên	20
2.4.6. Biểu đồ trình tự “quản lý môn học”	21
2.5. Biểu đồ cộng tác	22
2.5.1. Biểu đồ cộng tác “Đăng nhập”	22
2.5.2. Biểu đồ cộng tác “ Quản lý tài khoản “	22
2.5.3. Biểu đồ cộng tác quản lý điểm danh	23
2.5.4. Biểu đồ cộng tác quản lý bảng điểm	23
2.5.5. Biểu đồ cộng tác quản lý sinh viên.....	24
2.6. Biểu đồ lớp.....	24
2.6.1. Biểu đồ lớp cho ca sử dụng quản lý sinh viên.....	24
2.6.2. Biểu đồ lớp cho ca sử dụng quản lý điểm	25
2.6.3. Biểu đồ lớp cho ca sử dụng “Đăng nhập”	25
2.6.4. Biểu đồ lớp cho ca sử dụng “Tìm kiếm”	26
2.6.5. Biểu đồ lớp cho ca sử dụng quản lý điểm danh.....	26
2.7. Biểu đồ hoạt động	27
2.7.1. Đặc tả biểu đồ hoạt động tìm kiếm.....	27
2.7.2. Đặc tả biểu đồ hoạt động sửa thông tin sinh viên.....	28
2.7.3. Đặc tả biểu đồ hoạt động xóa sinh viên.....	29
2.7.4. Đặc tả biểu đồ hoạt động thêm sinh viên	30
2.7.5. Đặc tả biểu đồ hoạt động sửa thông tin điểm danh	31
2.7.6. Đặc tả biểu đồ hoạt động xóa thông tin điểm danh.	32
2.7.7. Đặc tả biểu đồ hoạt động thêm thông tin điểm danh.....	33

2.7.8. Đặc tả biểu đồ hoạt động sửa thông tin bảng điểm	34
2.7.9. Đặc tả biểu đồ hoạt động xóa thông tin bảng điểm.	35
2.7.10. Đặc tả biểu đồ hoạt động thêm thông tin bảng điểm....	36
CHƯƠNG 3: XÂY DỰNG CƠ SỞ DỮ LIỆU VÀ GIAO DIỆN	38
3.1. Thiết kế cơ sở dữ liệu	38
3.2. Thiết kế giao diện	40
KẾT LUẬN.....	43
TÀI LIỆU THAM KHẢO	44

LỜI CẢM ƠN

Trong lời đầu tiên của báo cáo đồ án lập trình Web nâng cao. Em muốn gửi những lời cảm ơn và biết ơn chân thành nhất của mình tới tất cả những người đã hỗ trợ, giúp đỡ chúng em về kiến thức và tinh thần trong quá trình thực hiện bài làm.

Đặc biệt, em xin gửi lời cảm ơn đến các Giảng viên Khoa Công nghệ thông tin- Trường Đại học Điện Lực, đã tận tình theo sát giúp đỡ, trực tiếp chỉ bảo, hướng dẫn trong suốt quá trình nghiên cứu và học tập của em. Trong thời gian học tập với thầy cô, em không những tiếp thu thêm nhiều kiến thức bổ ích mà còn học tập được tinh thần làm việc, thái độ nghiên cứu khoa học nghiêm túc, hiệu quả.

Do thời gian thực hiện có hạn kiến thức còn nhiều hạn chế nên bài làm của em chắc chắn không tránh khỏi những thiếu sót nhất định. Em rất mong nhận được ý kiến đóng góp của thầy cô giáo và các bạn để em có thêm kinh nghiệm và tiếp tục hoàn thiện báo cáo của mình.

Em xin chân thành cảm ơn!

CHƯƠNG 1: KHẢO SÁT HỆ THỐNG VÀ CÔNG NGHỆ SỬ DỤNG

1.1. Tìm hiểu về website

- Giới thiệu:

“Sinh viên là nền tảng của đất nước.”

Với sự bùng nổ thông tin trên internet, vai trò của việc quản lý thông tin trực tuyến càng trở nên quan trọng. Khác với việc quản lý dữ liệu sổ sách bình thường việc quản lý sinh viên bằng công nghệ thông tin càng giúp cho kết nối mọi mặt trở lên tốt hơn và chính xác hơn. Về phía những người quản lý việc quản lý thông tin một cách chính xác tránh mất mát dữ liệu sinh viên. Do đó việc sử dụng các trang web thông tin trực tuyến vô cùng ý nghĩa và thiết thực trong nền công nghiệp 4.0 hiện nay.

Với mọi mặt quản lý sinh viên từ thông tin kết quả học tập, điểm danh sinh viên từ đầu vào và môn học giúp chúng ta củng cố chi tiết thông tin chính xác cho một sinh viên tránh sự sai sót không đáng có.

Việc bùng nổ công nghệ thông tin làm cho việc quản lý thông tin trở lên vô cùng cần thiết không chỉ quản lý mà giúp người quản lý mọi lúc mọi nơi có thể điều chỉnh thông tin, chỉnh sửa và dễ dàng quản lý thông tin một cách chính xác nhất.

1.2. Công nghệ sử dụng.

1.2.1 Tổng quan về NodeJS

• Giới thiệu về JavaScript

JavaScript là một ngôn ngữ lập trình của HTML và web được sử dụng phổ biến nhất như là một phần của các trang web giúp tạo ra các web động. JavaScript là một chương trình thông dịch dựa trên hướng đối tượng.

Ban đầu, JavaScript được phát triển bởi Brenda Eich, một lập trình viên của Netscape vào khoảng thời gian 1995-1996. Thời điểm đó, ngôn ngữ này được gọi là LiveScript. Sau đó các chuyên gia marketing của công

ty đã đặt lại tên ngôn ngữ này thành JavaScript bởi vì sự phổ biến của Java lúc bấy giờ.

- **Ưu điểm của JavaScript**

Các lợi thế của việc sử dụng JavaScript là:

- **Sự tương tác Server ít hơn:** Ta có thể xác nhận đầu vào người sử dụng trước khi gửi trang tới Server. Điều này làm tiết kiệm lưu lượng tải ở Server, nghĩa là Server của bạn tải ít hơn.
- **Phản hồi ngay lập tức tới khách truy cập:** Họ không phải chờ một trang web tải lại để thấy xem nếu họ quên nhập một thông tin.
- **Tăng khả năng tương tác:** Có thể tạo các giao diện mà có thể phản ứng lại khi người dùng di chuyển chuột qua hoặc kích hoạt thông qua bàn phím.
- **Giao diện phong phú hơn:** Có thể sử dụng JavaScript để tạo các thành phần kéo, thả và các con trượt (Slider) để cung cấp một giao diện giàu tính năng hơn tới người dùng.

- **Hạn chế của JavaScript**

Ta không thể coi JavaScript như một ngôn ngữ chương trình chính thức (full-fieldged) do thiếu các tính năng quan trọng sau:

- Client-side JavaScript không cho phép đọc và ghi file vì lý do bảo mật.
- JavaScript không được sử dụng cho việc kết nối mạng các ứng dụng vì không có những hỗ trợ có sẵn.
- JavaScript không có khả năng xử lý đa luồng hoặc đa xử lý.

1.2.2 Khái niệm về NodeJS

NodeJS là một nền tảng Server side được xây dựng dựa trên JavaScript Engine (V8 Engine) được phát triển bởi Ryan Dahl năm 2009 và phiên bản cuối cùng là v0.10.36. Định nghĩa NodeJS bởi tài liệu chính thức như sau:

Node.js là một nền tảng dựa vào Chrome JavaScript runtime để xây dựng các ứng dụng nhanh, có độ lớn. Node.js sử dụng các phần phát sinh các sự kiện (event-driven), mô hình non-blocking I/O để tạo các ứng dụng

nhẹ và hiệu quả cho các ứng dụng về dữ liệu thời gian thực chạy trên các thiết bị phân tán.

NodeJS là một mã nguồn mở, đa nền tảng cho phát triển các ứng dụng phía Server và các ứng dụng liên quan đến mạng. Ứng dụng NodeJS được viết bằng JavaScript và có thể chạy trong môi trường NodeJS trên hệ điều hành Window, Linux.

NodeJS cũng cung cấp cho chúng ta các module JavaScript đa dạng, có thể đơn giản hoá sự phát triển của các ứng dụng web sử dụng NodeJS với các phần mở rộng.

- **Đặc điểm của NodeJS**

Dưới đây là một số đặc điểm quan trọng khiến NodeJS trở thành sự lựa chọn hàng đầu trong phát triển phần mềm:

Không đồng bộ và phát sinh sự kiện (Event Drivent): Tất cả các APIs của thư viện NodeJS đều đồng bộ, nghĩa là không khoá (blocking). Điều này rất cần thiết vì NodeJS không bao giờ đợi một API trả về dữ liệu. Server chuyển sang một API sau khi gọi và có cơ chế thông báo về sự kiện của NodeJS giúp Server nhận được phản hồi từ các API gọi trước đó.

Tốc độ rất nhanh: Dựa trên V8 JavaScript Engine của Google Chrome khiến NodeJS xử lý rất nhanh trong các quá trình thực hiện code. Đó là một yêu cầu khá quan trọng khi phát triển các startup muốn tạo ra một sản phẩm lớn và muốn đảm bảo có thể mở rộng nhanh chóng, đáp ứng được một lượng lớn người dùng khi trang Web đang phát triển.

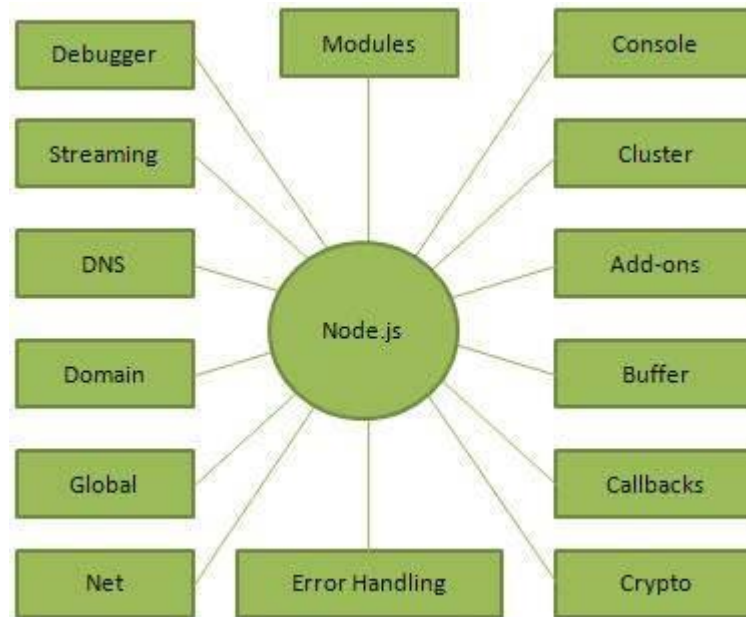
Các tiến trình đơn giản nhưng hiệu năng cao: NodeJS sử dụng một mô hình đơn luồng (single thread) với các sự kiện lặp. Các cơ chế sự kiện giúp Server trả lại các phản hồi một cách không khoá và tạo cho Server hiệu quả cao ngược lại với các cách truyền thống tạo ra một số lượng luồng hữu hạn để quản lý các request. NodeJS sử dụng các chương trình đơn luồng và các chương trình này cung cấp các dịch vụ cho số lượng request nhiều hơn so với các Server truyền thống như Apache HTTP Server.

Không đệm: Ứng dụng NodeJS không lưu trữ các dữ liệu buffer.

Có giấy phép: NodeJS được phát hành dựa vào MIT License.

- **Các thành phần quan trọng trong NodeJS**

Lược đồ dưới đây mô tả các thành phần quan trọng của NodeJS



- **Ứng dụng của NodeJS**

Các lĩnh vực trong thực tế nên sử dụng NodeJS

- Các ứng dụng về I/O
- Các ứng dụng về luồng dữ liệu
- Các ứng dụng về dữ liệu hướng đến thời gian thực
- Các ứng dụng dựa vào JSON APIs
- Các ứng dụng Single Page Application

- **Nhược điểm của NodeJS**

NodeJS không nên sử dụng trong các ứng dụng đòi hỏi về CPU.

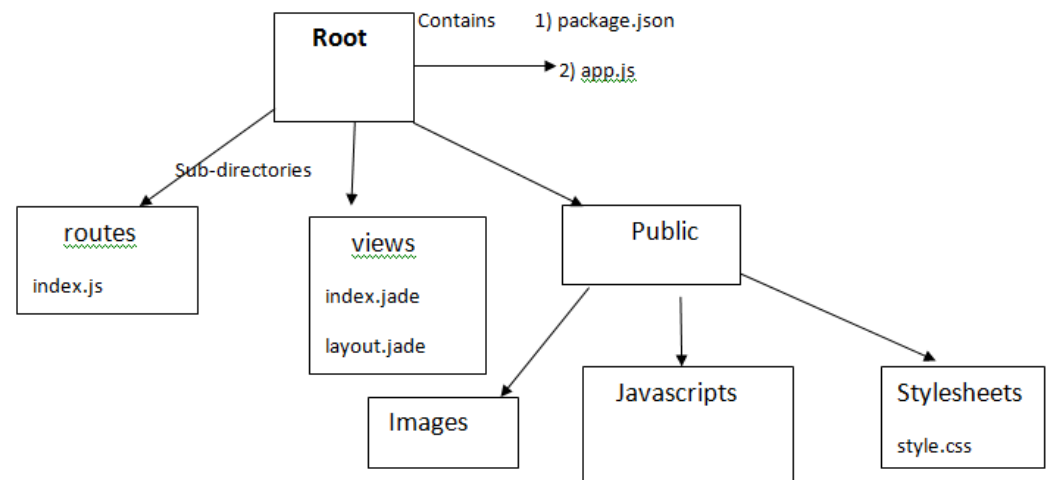
1.2.3 Tổng quan về ExpressJS

- **Giới thiệu về ExpressJS**

Express là một framework nhỏ nhưng linh hoạt được xây dựng trên nền tảng của NodeJS. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile. Express có vô số các package hỗ trợ. Về performance, Express cung cấp thêm về các tính năng để lập trình tốt hơn chứ không làm giảm tốc độ của NodeJS. Hơn hết, các framework nổi tiếng của NodeJS

hiện nay đều sử dụng ExpressJS như một core function, chẳng hạn: SailsJS, MEAN...

- **Cấu trúc của ExpressJS**



Cấu trúc của ExpressJS vô cùng đơn giản

- **Root**
 - app.js chứa các thông tin về cấu hình, khai báo, các định nghĩa... để ứng dụng của chúng ta chạy được.
 - package.json chứa các package cho ứng dụng chạy (tương tự như composer.json hoặc Gemfile của PHP hoặc RoR)
- Folder routes: chứa các route có trong ứng dụng
- Folder view: chứa các view/template cho ứng dụng
- Folder public: chứa các file css, js, images... cho ứng dụng

1.2.4 Tổng quan về MySQL

- **Giới thiệu về MySQL**

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. 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 cơ sở dữ liệu trên Internet. MySQL miễn phí hoàn toàn cho nên bạn có thể tải về 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, MacOSX, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS...

MySQL là một trong những ví dụ rất 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 hỗ 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...

- **Ưu điểm của MySQL**
 - Là hệ quản trị cơ sở dữ liệu quan hệ
 - Hoạt động theo mô hình client/server, làm database server cho các client kết nối đến
 - Hỗ trợ SQL tiêu chuẩn
 - Có nhiều table format với đặc tính khác nhau, ví dụ MyISAM không hỗ trợ transaction, InnoDB hỗ trợ transaction...
 - Có nhiều API, library hỗ trợ cho nhiều ngôn ngữ lập trình
 - Hỗ trợ giao tiếp ODBC
 - Có khả năng replication
 - Có thể chạy trên nhiều platform: Mac OS X, Windows, Linux, Unix (HP-UX, Solaris...)

- **Nhược điểm của MySQL**

Do MySQL hỗ trợ các table format khác nhau, một số table format sẽ không hỗ trợ transaction (row locking), không hỗ trợ hot backup, không hỗ trợ định nghĩa kiểu dữ liệu, không hỗ trợ XML, không hỗ trợ các tính năng OLAP...

1.2.5 Tổng quan về Reactjs

- **Giới thiệu về reactjs**

React.js là một thư viện Javascript để xây dựng giao diện người dùng, được phát triển ban đầu từ Facebook và đóng góp bởi cộng đồng lập trình viên trên thế giới.

React.js có thể sử dụng để viết phần nền cho single-page hoặc ứng dụng di động, giúp trang lấy và xử lý dữ liệu được tối ưu hơn các phương pháp khác.

React.js thích hợp với các ứng dụng lớn, khả năng mở rộng ở tương lai.

- **Điểm mạnh của reactjs.**

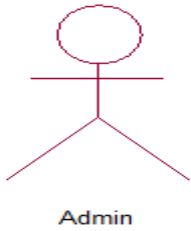

Điểm mạnh của React.js dễ thấy nhất có lẽ là phần **Component**: React.js chia nhỏ các phần của trang thành từng phần riêng biệt để xử lý, gọi là **component**, giúp dễ quản lý, dễ sử dụng ở nhiều nơi. Mỗi khi dữ liệu được cập nhật mới, thay vì thay đổi nguyên trang, thì React.js sẽ giúp thay đổi chỉ **component** liên quan, việc này sẽ tối ưu rất nhiều thời gian làm mới dữ liệu. Đặc biệt là hầu hết các **component** thường dùng đã được phát triển và chia sẻ, chỉ cần cài đặt và sử dụng.

- **Điểm yếu của reactjs**

Khó học đối với một số bạn mới tiếp xúc lần đầu, hoặc với những bạn đã quen với cách viết các thư viện khác như jQuery.

CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

2.1. Các tác nhân của hệ thống

STT	Tác nhân	Ý nghĩa, nhiệm vụ của Actor
1		Tác nhân Admin là tác nhân giữ vai trò chính của hệ thống website, là người giữ vai trò quản lý đăng nhập, phân quyền cho các tài khoản khác sử dụng, sửa điểm, điểm danh sinh viên,...
2		Giảng viên là người đăng nhập vào hệ thống để quản lý việc nhập điểm cũng như điểm danh sinh viên,

2.2. Các chức năng của hệ thống

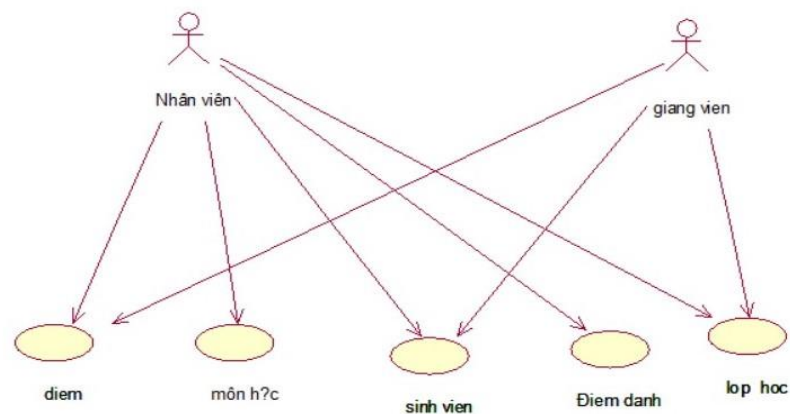
Dựa trên văn bản mô tả bài toán và việc phân tích các tác nhân, ta xác định được các ca sử dụng như sau:

STT	Tên chức năng	Mô tả
1	Đăng nhập	Chức năng này cho phép Admin, giảng viên, nhân viên phòng đào tạo.

2	Phân quyền	Chức năng cho phép người quản lý quản lý các tài khoản đăng nhập.
3	Tìm kiếm	Chức năng này giúp Admin và nhân viên tìm kiếm thông tin về sinh viên lớp.
4	Điểm danh	Chức năng này cho phép giảng viên, người quản lý điểm danh sinh viên theo môn học.
5	Bảng điểm	Chức năng này cho phép quản lý bảng điểm sinh viên.
6	Sinh viên	Chức năng này cho phép quản lý sinh viên.

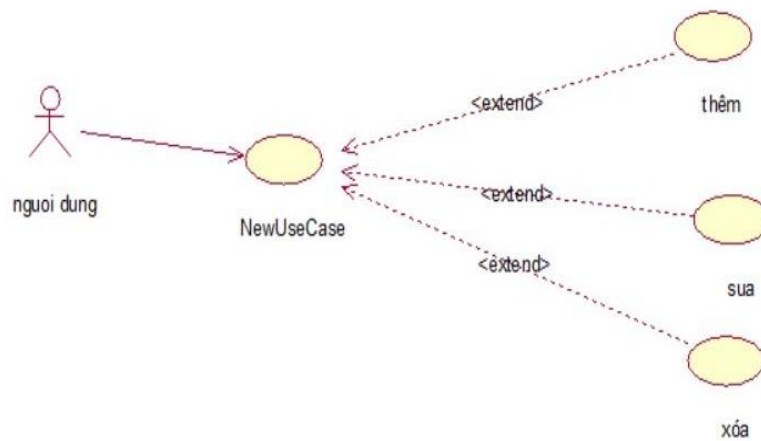
2.3. Biểu đồ usecase

2.3.1. Biểu đồ use case chính



Hình 2.3.1. Biểu đồ usecase chính

2.3.2. Biểu đồ use case người dùng



Hình 2.3.2. Biểu đồ use case người dùng

2.3.3. Biểu đồ use case đăng nhập



Hình 2.3.3 - Biểu đồ Use case đăng nhập

- Tác nhân: Admin, Khách hàng thành viên.

- Mô tả khái quát:

+ PM truy cập vào trang quản trị để quản lý hệ thống của mình.

+ Teacher đăng nhập vào hệ thống với quyền quản lý sinh viên và điểm danh hay nhập điểm.

Điều kiện đầu vào: Ca sử dụng bắt đầu khi người dùng muốn đăng nhập vào hệ thống.

- Dòng sự kiện chính:

+ Hệ thống yêu cầu nhập tên đăng nhập và mật khẩu.

- + Người dùng nhập tên đăng nhập và mật khẩu.
- + Hệ thống kiểm tra tên và mật khẩu vừa nhập, nếu đúng sẽ cho phép vào hệ thống.
- Dòng sự kiện phụ:
 - + Nếu trong kịch bản chính tên hoặc mật khẩu bị sai thì hệ thống không vào được.
 - + Người dùng có thể nhập lại hoặc hủy bỏ việc đăng nhập khi đó ca sử dụng sẽ kết thúc.

2.3.4. Biểu đồ use case phân quyền



Hình 2.3.4 - Biểu đồ use case phân quyền

- Tác nhân: người dùng.
- Mô tả khái quát: cho phép người dùng tạo tài khoản và quyền cho người dùng khác đăng nhập hệ thống.
- Điều kiện đầu vào: Ca sử dụng bắt đầu khi người dùng truy cập vào hệ thống và chọn thêm trong mục tài khoản.

❖ Dòng sự kiện chính:

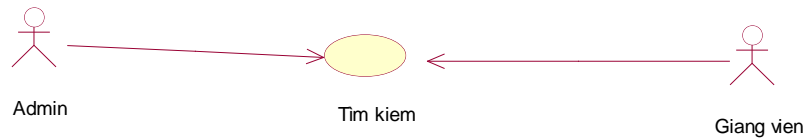
- Người dùng chọn mục Thêm.
- Form thêm hiển thị.
- Người dùng nhập thông tin cá nhân cần thiết vào form.
- Nhấn nút Lưu.
- Hệ thống thông báo kết quả quá trình nhập thông tin cá nhân. Nếu thông tin nhập chính xác thì hệ thống cập nhật thông tin tài khoản vào danh tài khoản.

❖ Dòng sự kiện phụ:

- Quá trình nhập thông tin không chính xác.
- Hệ thống thông báo nhập thông tin không chính xác.

- Hệ thống yêu cầu khách hàng xem lại thông tin.
- Khách hàng đồng ý thì nhập lại thông tin, nếu không đồng ý thì ca sử dụng kết thúc.

2.3.5. Biểu đồ use case tìm kiếm



Hình 2.3.5 - Biểu đồ Use case tìm kiếm

- Tác nhân: Giảng viên, Admin.
- Mô tả khái quát: Thông tin cần tìm kiếm được hiển thị.
- Điều kiện đầu vào: Người dùng truy cập vào hệ thống website nhập thông tin cần tìm kiếm, hệ thống sẽ tự động thực hiện yêu cầu tìm kiếm.

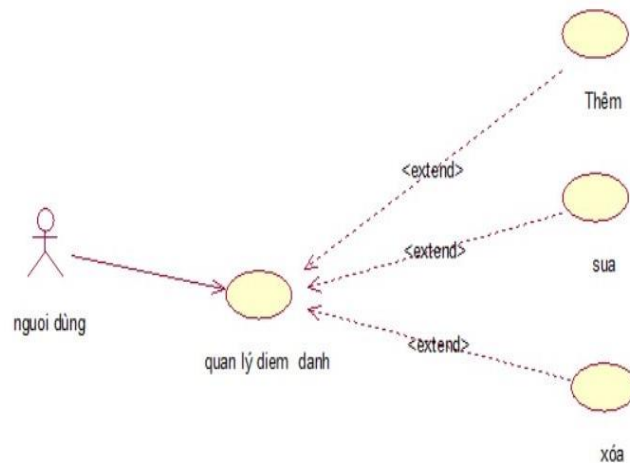
❖ Dòng sự kiện chính:

- Ca sử dụng bắt đầu khi người dùng truy cập vào website của hệ thống và chọn tìm kiếm. Sau đó người dùng điền từ khóa và hệ thống sẽ kiểm tra nếu từ khóa hợp lệ:
- Hệ thống bắt đầu hiển thị thông tin sản phẩm hoặc thông tin bài viết được tìm thấy.
- Người dùng có thể chọn xem thông tin hoặc chọn “Thoát” khi đó ca sử dụng kết thúc.

❖ Dòng sự kiện phụ:

- Từ khóa không hợp lệ:
- Người dùng có thể điền lại từ khóa hoặc hủy bỏ tìm kiếm, khi đó ca sử dụng kết thúc.

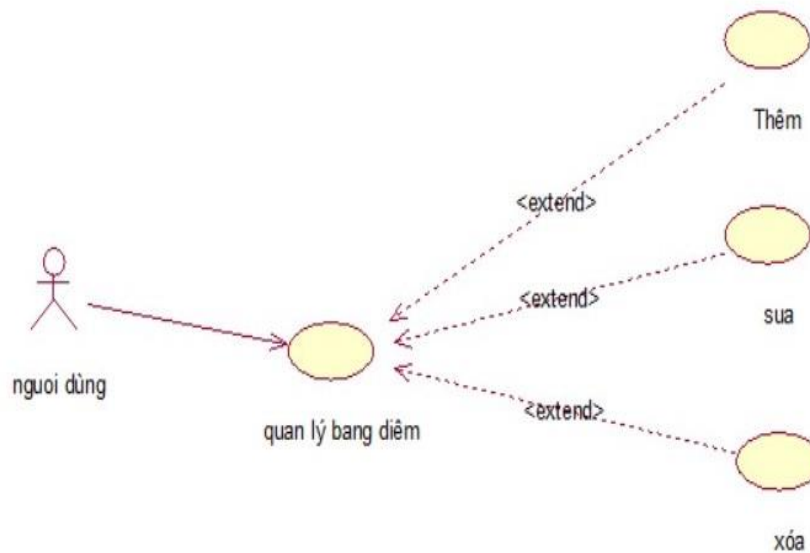
2.3.6. Biểu đồ use điểm danh



Hình 2.3.6 - Biểu đồ Use case điểm danh

- Tác nhân: Người dùng hoặc giảng viên
- Mô tả khái quát người dùng hoặc giảng viên vào điểm danh hoặc sửa điểm danh.
- Điều kiện đầu vào: Ca sử dụng bắt đầu khi người dùng hoặc giảng viên vào mục điểm danh.
- ❖ Dòng sự kiện chính:
 - Người dùng hoặc giảng viên chọn mục bình luận. Hệ thống sẽ hiển thị ra màn hình. Sau đó vào thêm mới để tạo mới điểm danh hoặc chọn vào các phiếu điểm danh cũ đã có.

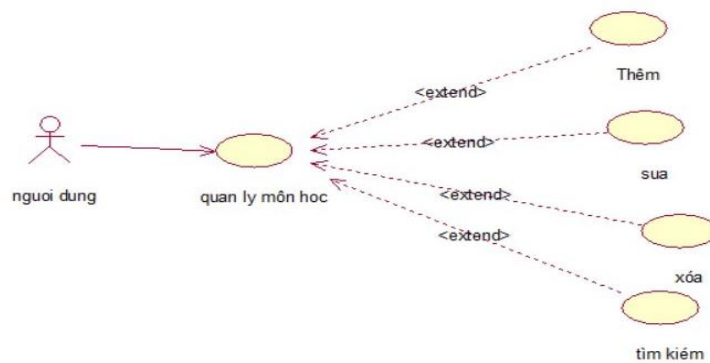
2.3.7. Biểu đồ Use case bảng điểm



Hình 2.3.7 - Biểu đồ Use case bảng điểm

- ❖ Use case chọn tour cần đặt.
 - Tác nhân: người dùng hoặc giảng viên.
 - Mô tả khái quát: Ca sử dụng cho phép cho phép nhập quản lý điểm.
 - Điều kiện đầu vào: Ca sử dụng bắt đầu khi người dùng hoặc giảng viên vào hệ thống.
- ❖ Dòng sự kiện chính:
 - Khách hàng chọn vào điểm hoặc lựa chọn thêm mới bảng điểm.
 - Hệ thống hiển thị các bảng điểm cũ.
 - Người dùng xem có thể lựa chọn để sửa hoặc xóa.
 - Ca sử dụng kết thúc

2.3.8. Biểu đồ use case quản lý sinh viên.



Hình 2.3.8 - Biểu đồ Use case quản lý bài viết

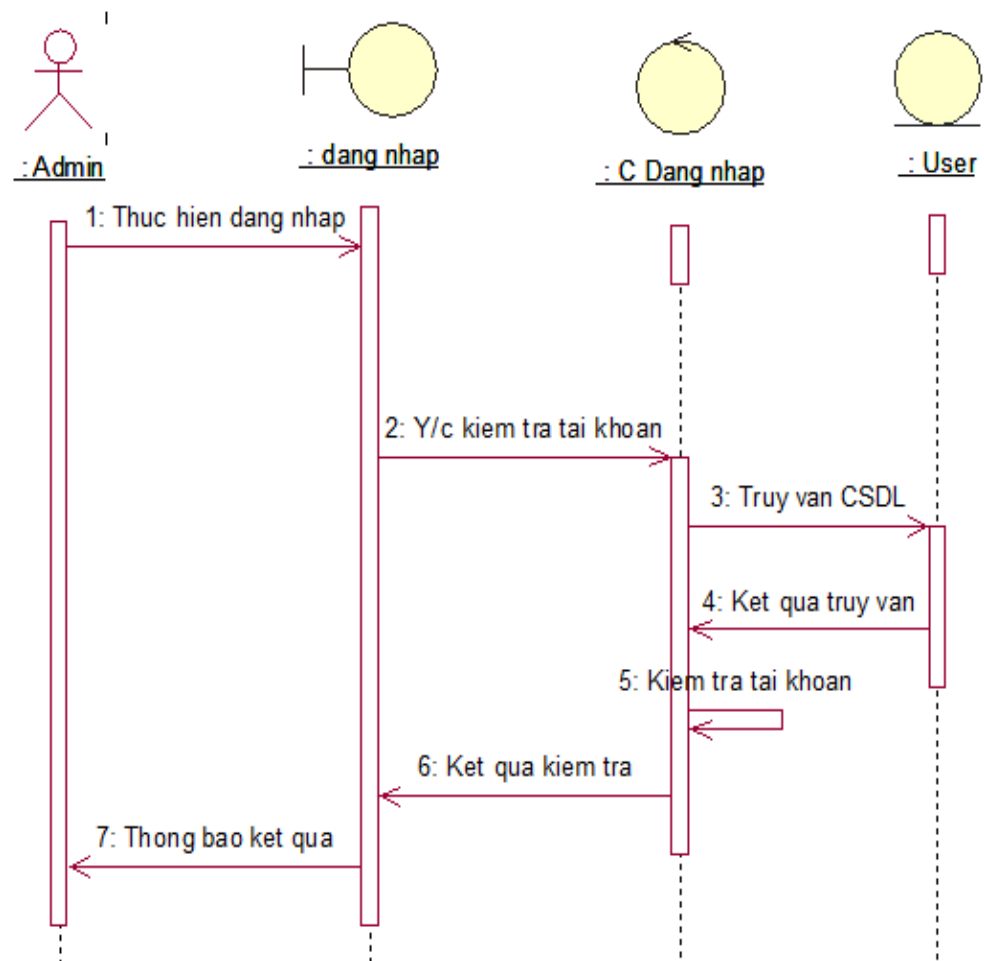
- Tác nhân: Người quản trị
- Mô tả khái quát: Người quản trị vào hệ thống và quản lý thông tin.
- Điều kiện đầu vào: Ca sử dụng bắt đầu khi người dùng đăng nhập thành công vào hệ thống.

❖ Dòng sự kiện chính:

- Ca sử dụng bắt đầu khi người dùng vào hệ thống đăng nhập và nhập mật khẩu của mình.
 - Hệ thống yêu cầu người dùng có thể chọn một số chức năng cần thực hiện: thêm mới, sửa, xóa, thoát.
- + Nếu chọn “Thêm mới” thì sự kiện con được thực hiện.
+ Nếu chọn “Sửa bài viết” thì sự kiện con được thực hiện.
+ Nếu chọn “Xóa bài viết” thì sự kiện con được thực hiện.
+ Nếu chọn “Thoát” thì ca sử dụng kết thúc.

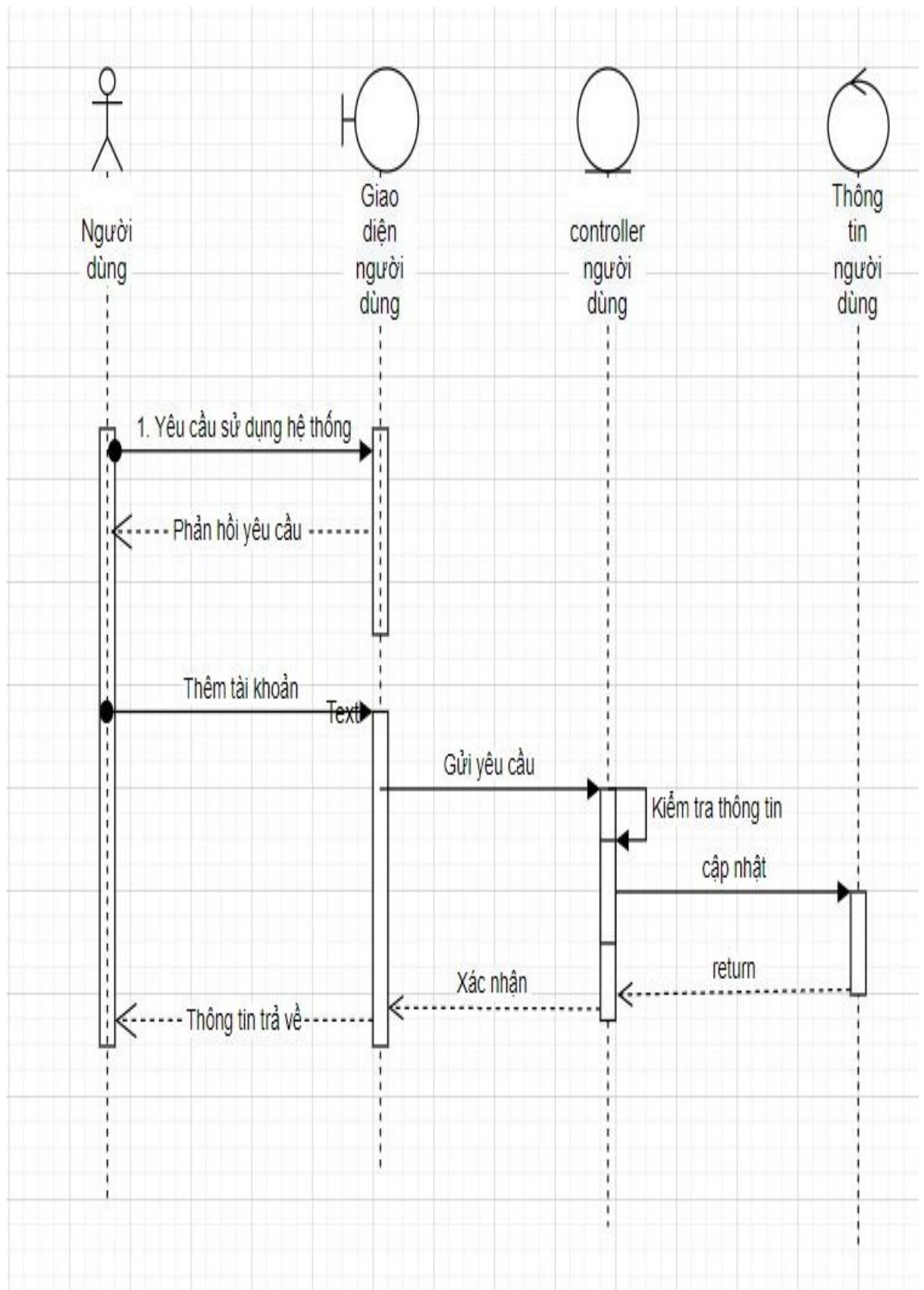
2.4. Biểu đồ trình tự

2.4.1. Biểu đồ trình tự “Đăng nhập”



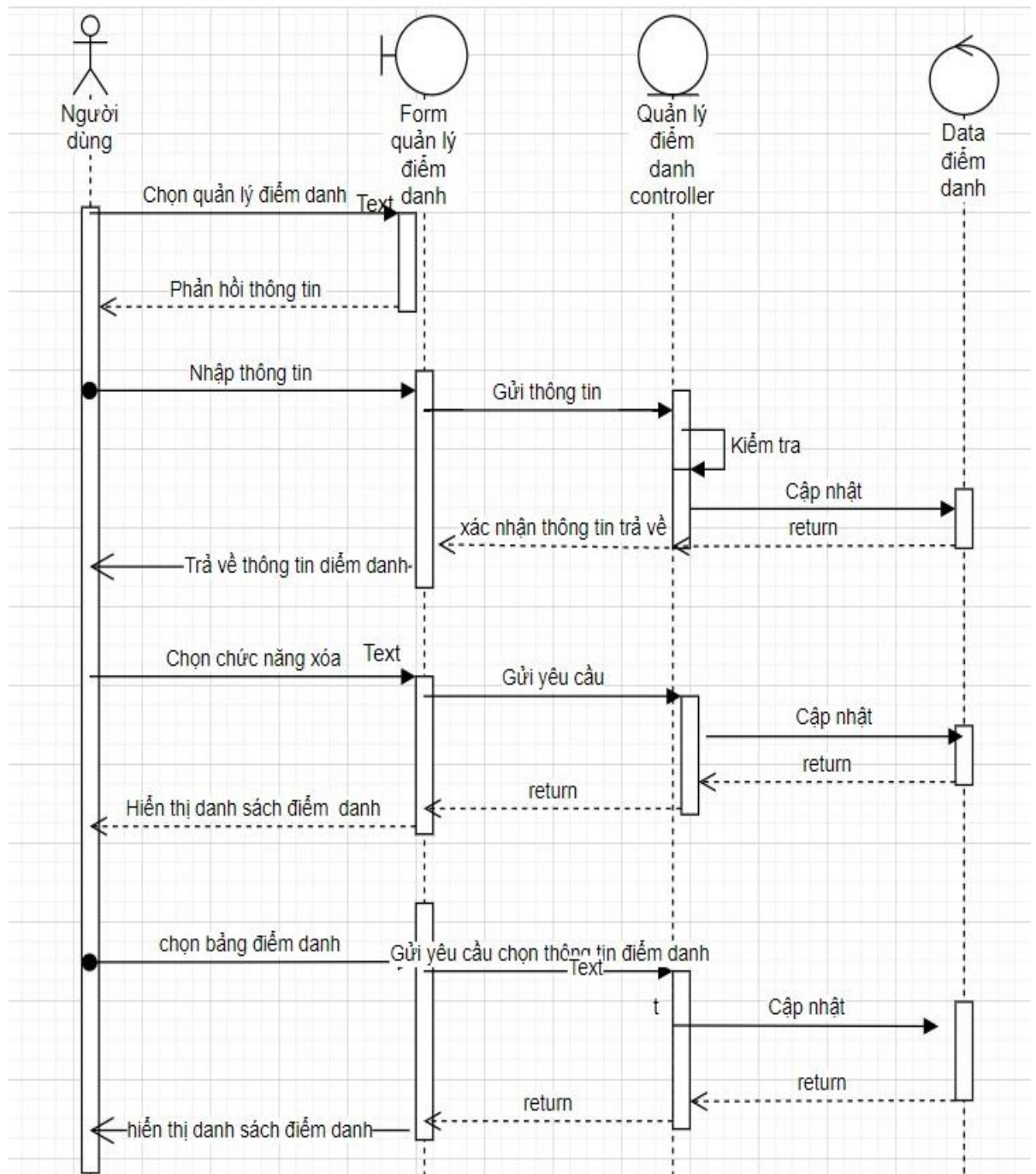
Hình 2.4.1 - Biểu đồ trình tự cho ca sử dụng đăng nhập

2.4.2. Biểu đồ trình tự phân quyền



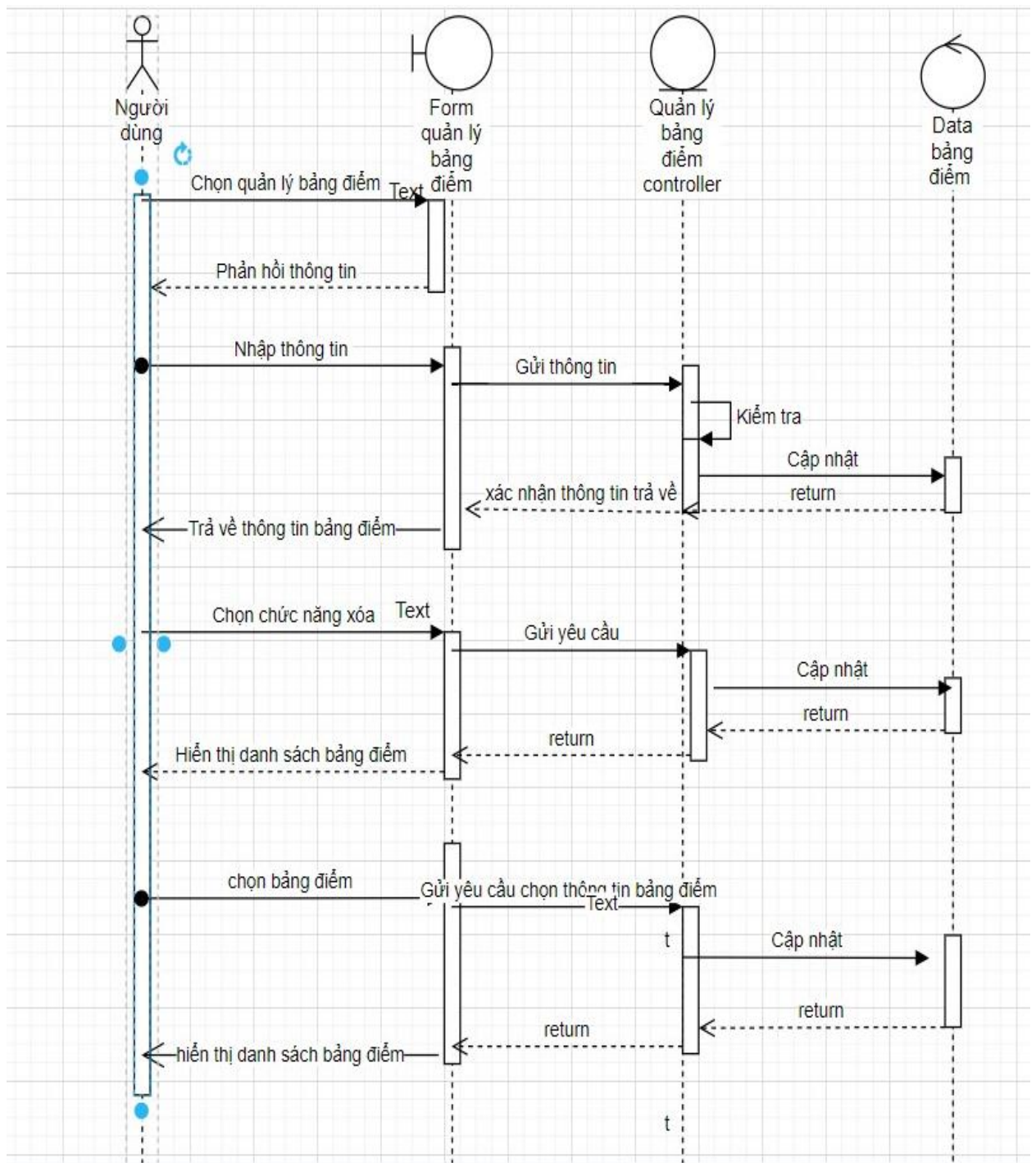
Hình 2.4.2 - Biểu đồ trình tự cho ca sử dụng phân quyền

2.4.3. Biểu đồ trình tự quản lý điểm danh



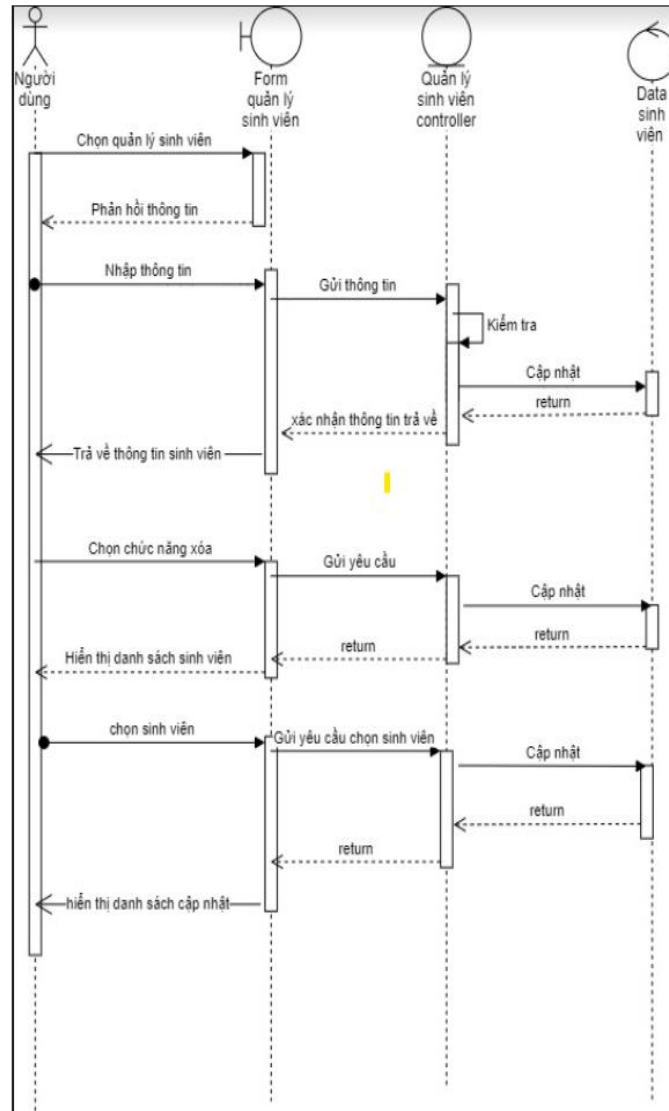
Hình 2.4.3 - Biểu đồ trình tự cho ca sử dụng quản lý điểm danh

2.4.4. Biểu đồ trình tự quản lý bảng điểm



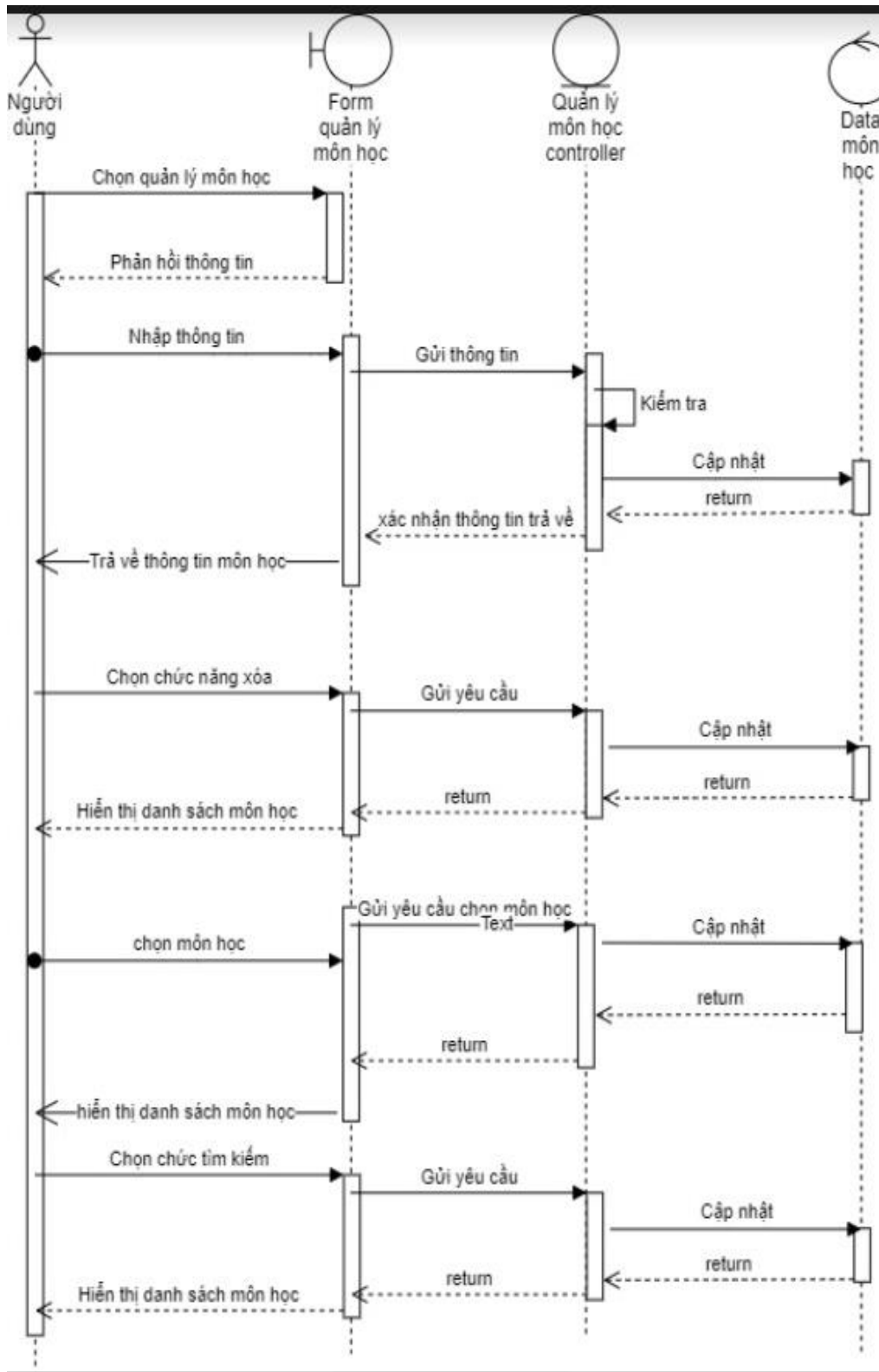
Hình 2.4.4 - Biểu đồ trình tự cho ca sử dụng quản lý bảng điểm.

2.4.5. Biểu đồ trình tự quản lý sinh viên



Hình 2.4.5 - Biểu đồ trình tự cho ca sử dụng quản lý sinh viên.

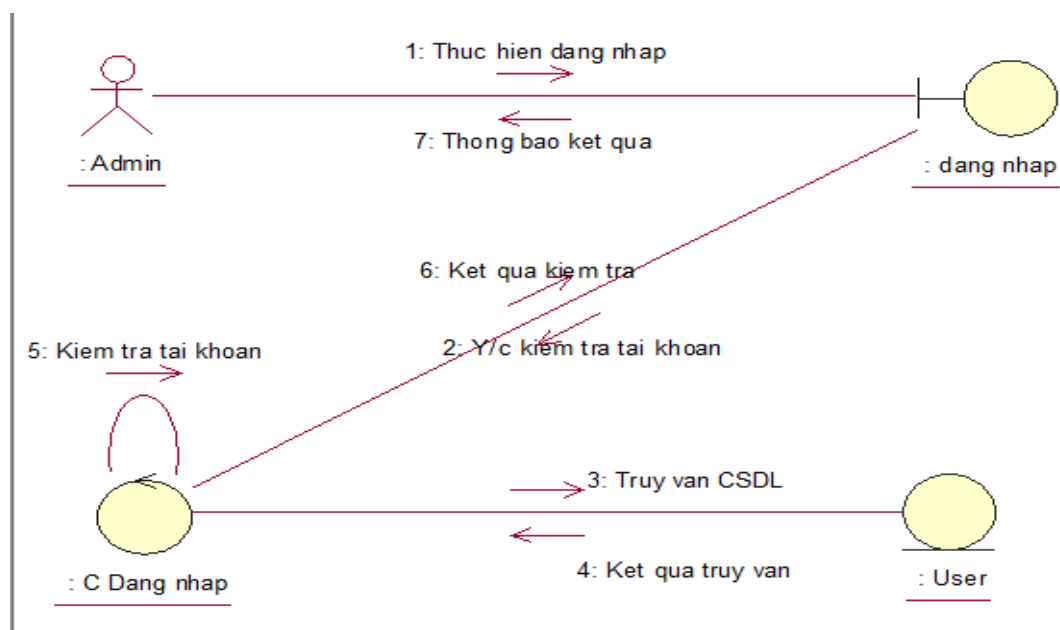
2.4.6. Biểu đồ trình tự “quản lý môn học”



Hình 2.4.6 - Biểu đồ trình tự cho ca sử dụng quản lý môn học

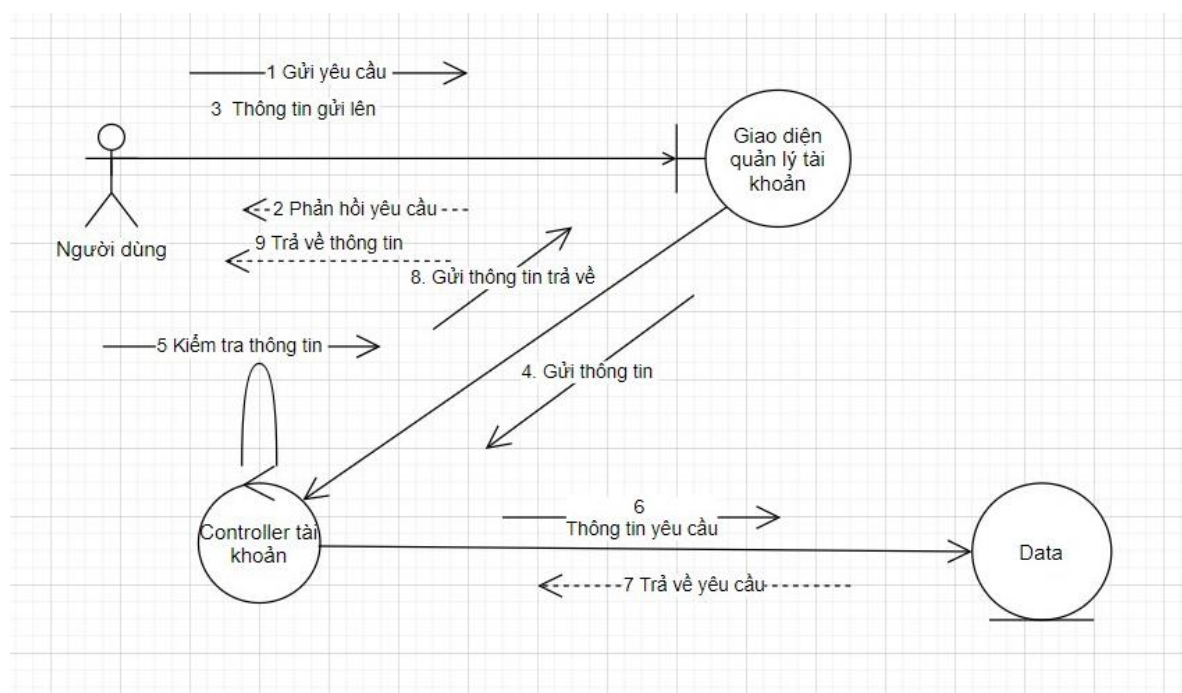
2.5. Biểu đồ cộng tác

2.5.1. Biểu đồ cộng tác “Đăng nhập”



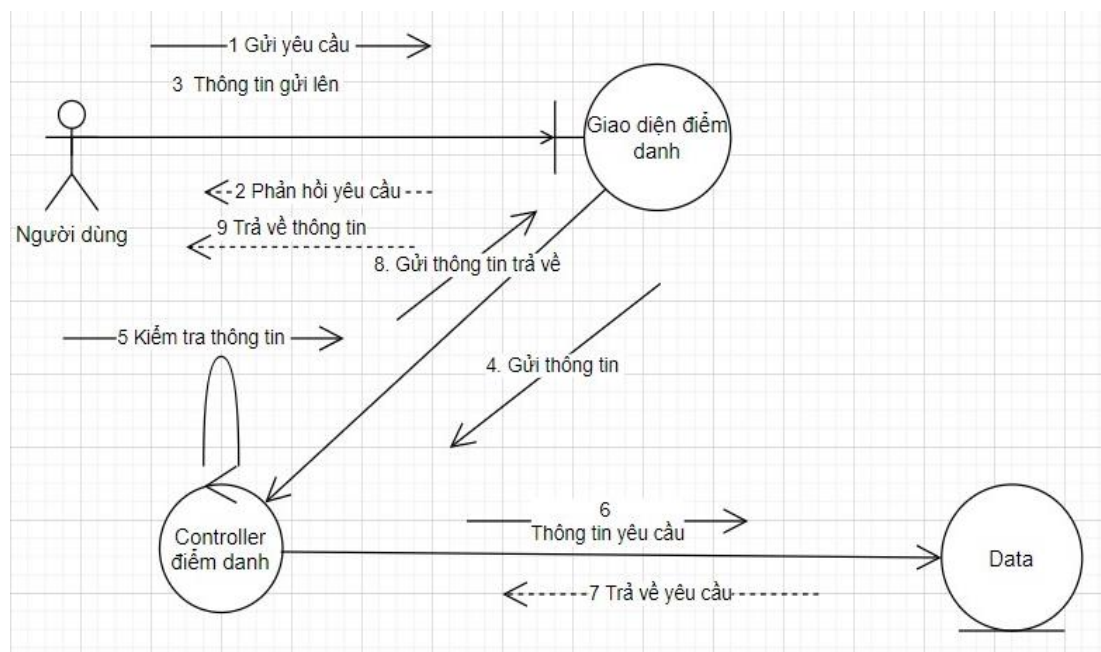
Hình 2.5.1 - Biểu đồ cộng tác cho ca sử dụng đăng nhập

2.5.2. Biểu đồ cộng tác “Quản lý tài khoản”



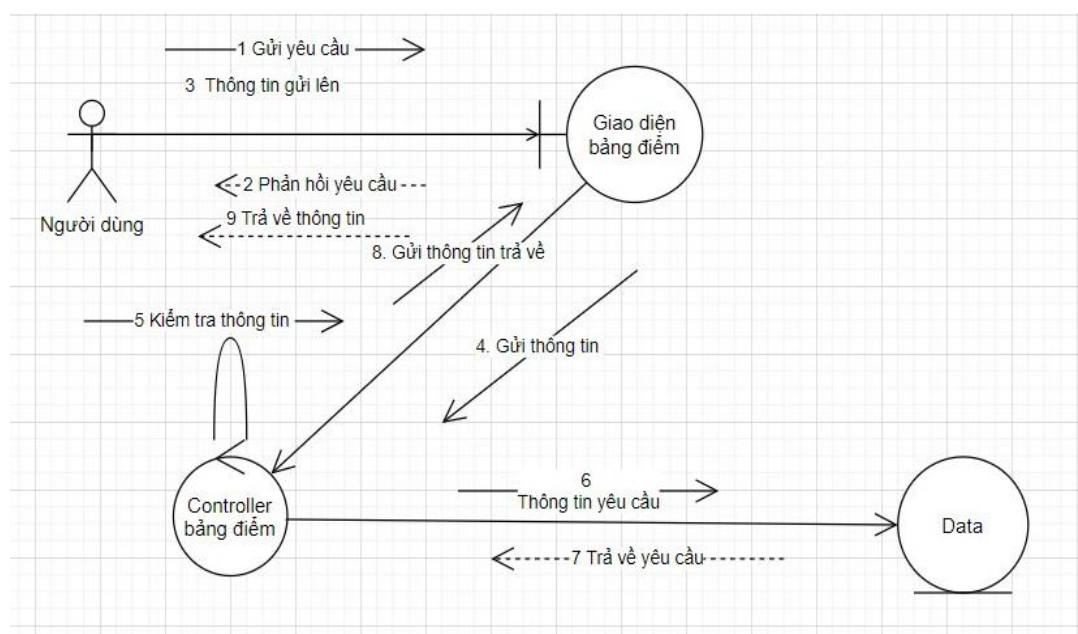
Hình 2.5.2-Biểu đồ cộng tác cho ca sử dụng quản lý tài khoản

2.5.3. Biểu đồ cộng tác quản lý điểm danh .



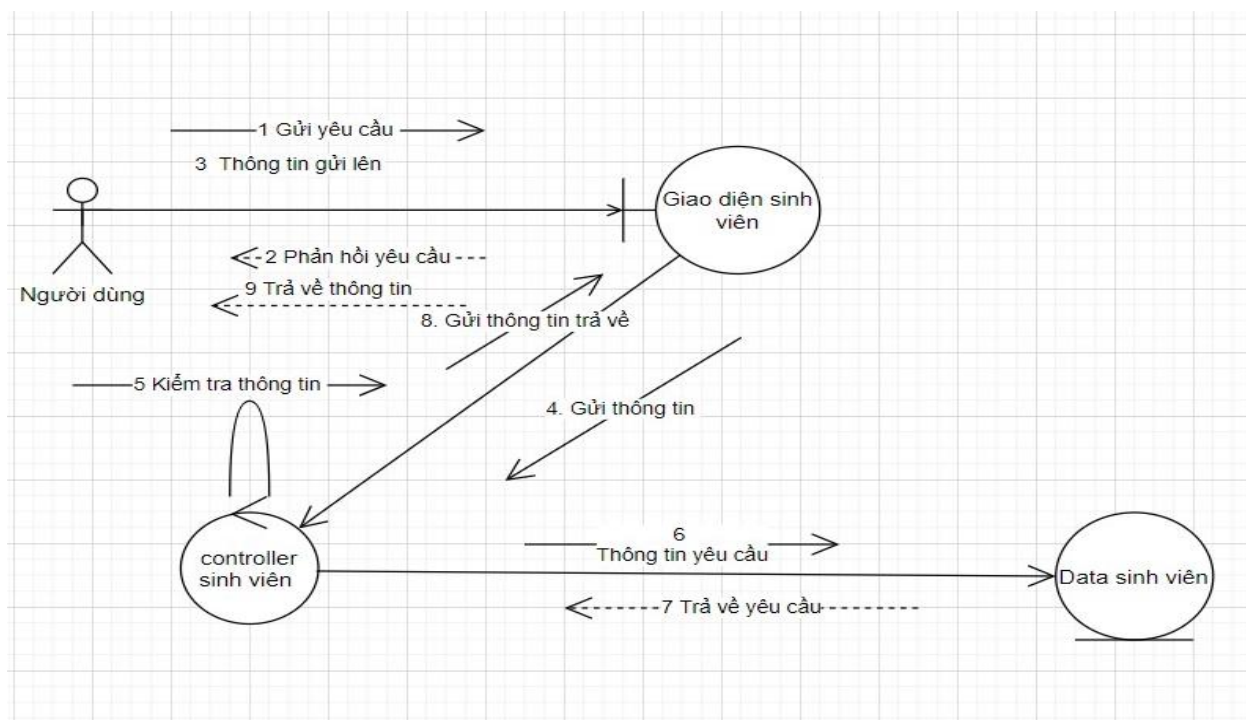
Hình 2.5.3 - Biểu đồ cộng tác cho ca sử dụng quản lý điểm danh

2.5.4. Biểu đồ cộng tác quản lý bảng điểm



Hình 2.5.4 - Biểu đồ cộng tác cho ca sử quản lý bảng điểm

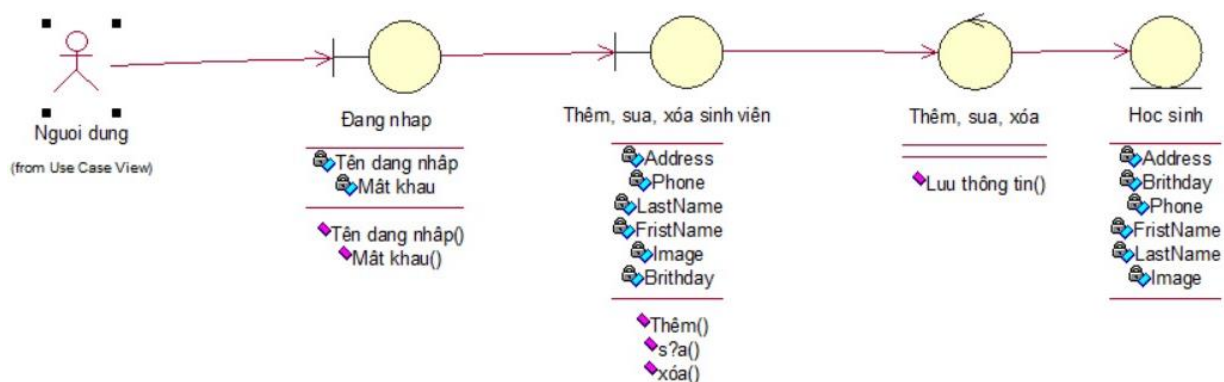
2.5.5. Biểu đồ cộng tác quản lý sinh viên.



Hình 2.5.5 - Biểu đồ cộng tác cho ca sử dụng quản lý sinh viên.

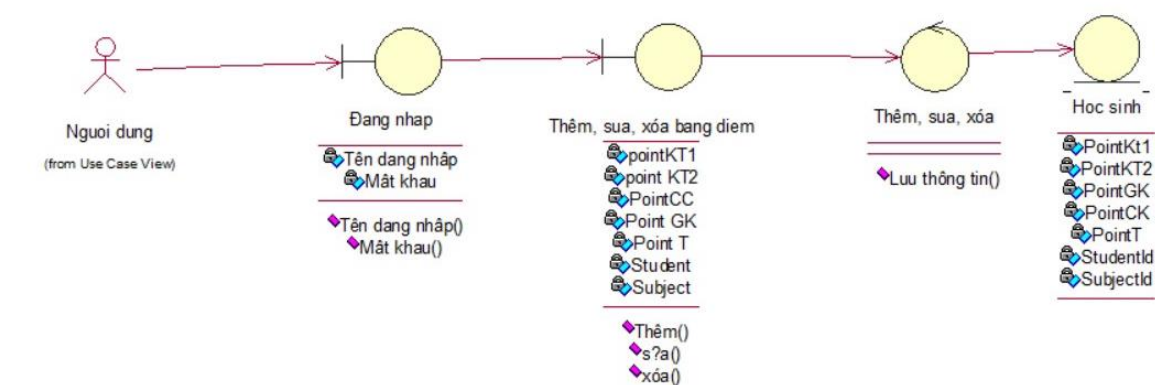
2.6. Biểu đồ lớp

2.6.1. Biểu đồ lớp cho ca sử dụng quản lý sinh viên



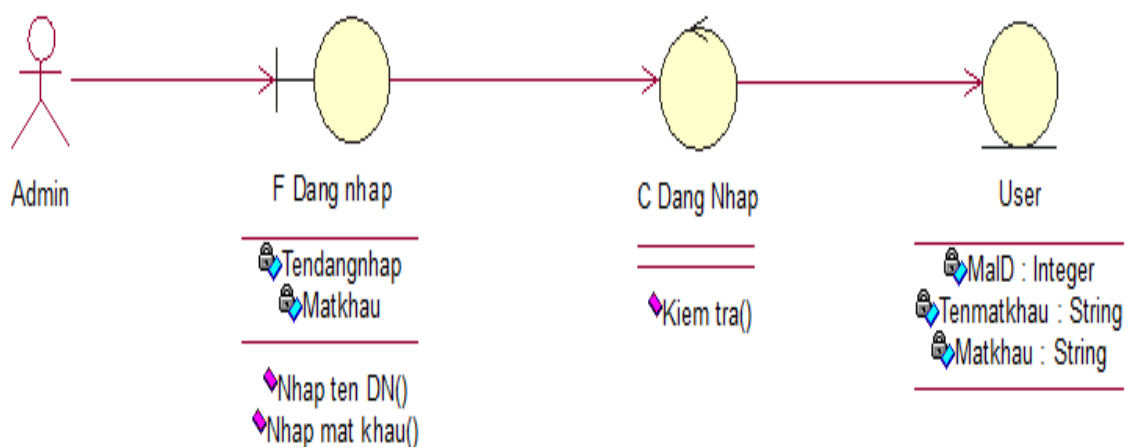
Hình 2.6.1 - Biểu đồ lớp cho ca sử dụng thêm ,sửa, xóa sinh viên

2.6.2. Biểu đồ lớp cho ca sử dụng quản lý điểm



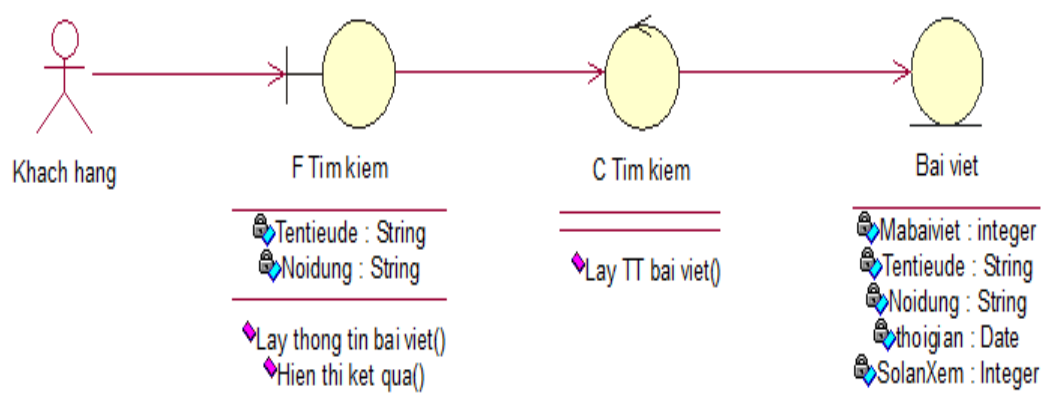
Hình 2.6.2 - Biểu đồ lớp cho ca sử dụng quản lý điểm

2.6.3. Biểu đồ lớp cho ca sử dụng “Đăng nhập”



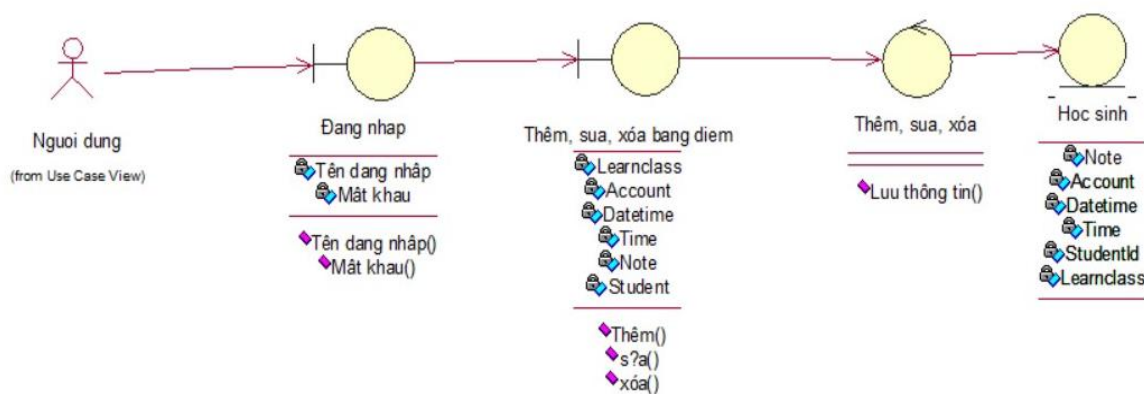
Hình 2.6.3 - Biểu đồ lớp cho ca sử dụng đăng nhập

2.6.4. Biểu đồ lớp cho ca sử dụng “Tìm kiếm”



Hình 2.6.4 - Biểu đồ lớp cho ca sử dụng tìm kiếm

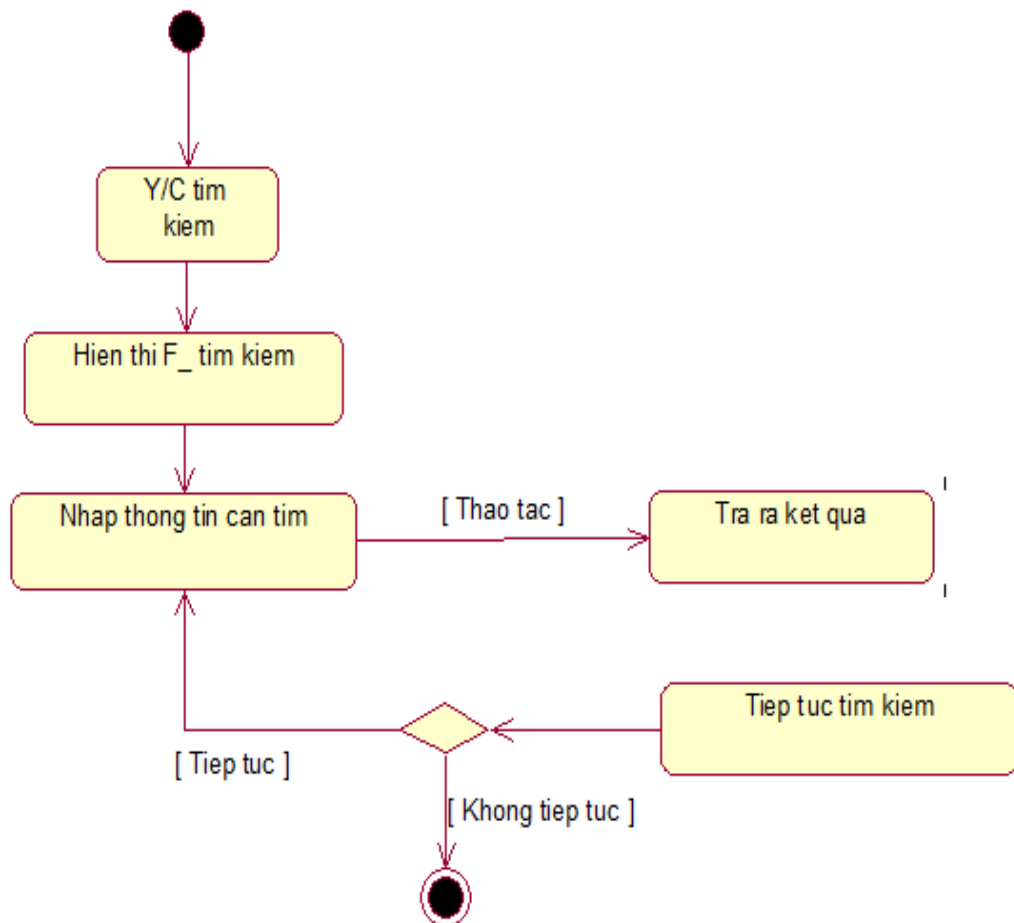
2.6.5. Biểu đồ lớp cho ca sử dụng quản lý điểm danh



Hình 2.6.5 - Biểu đồ lớp cho ca sử dụng cho việc điểm danh

2.7. Biểu đồ hoạt động

2.7.1. Đặc tả biểu đồ hoạt động tìm kiếm



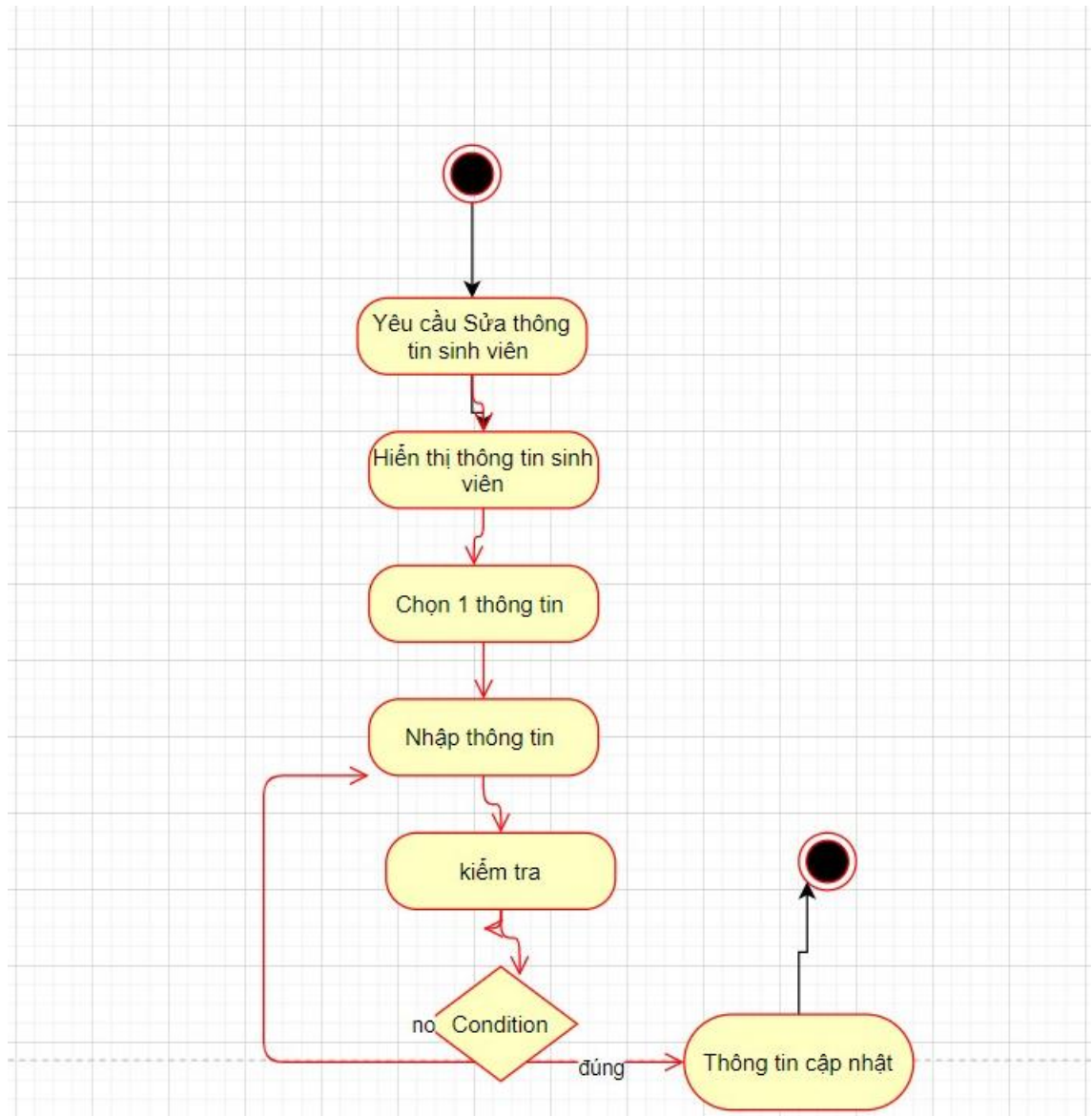
Hình 2.7.1 - Biểu đồ hoạt động tìm kiếm

❖ Đặc tả quy trình hoạt động tìm kiếm sản phẩm:

- Người dùng yêu cầu tìm sản phẩm.
- Lúc này tác nhân sẽ tương tác việc tìm kiếm thông qua giao diện trang chủ của hệ thống.
- Hệ thống hiển thị giao diện tìm kiếm và yêu cầu tác nhân nhập thông tin muốn tìm kiếm.
- Việc tìm kiếm được thực hiện. Hệ thống căn cứ vào từ khóa tìm kiếm của tác nhân để thao tác trong cơ sở dữ liệu.
- Kết quả trả ra theo yêu cầu tìm kiếm của tác nhân qua giao diện trang chủ.

- Nếu tác nhân tiếp tục muốn tìm kiếm thêm thông tin thì thực hiện việc nhập thông tin tìm kiếm như ban đầu và hệ thống tiếp tục thực hiện theo quy trình đó.
- Kết thúc quy trình kiểm kiếm.

2.7.2. Đặc tả biểu đồ hoạt động sửa thông tin sinh viên



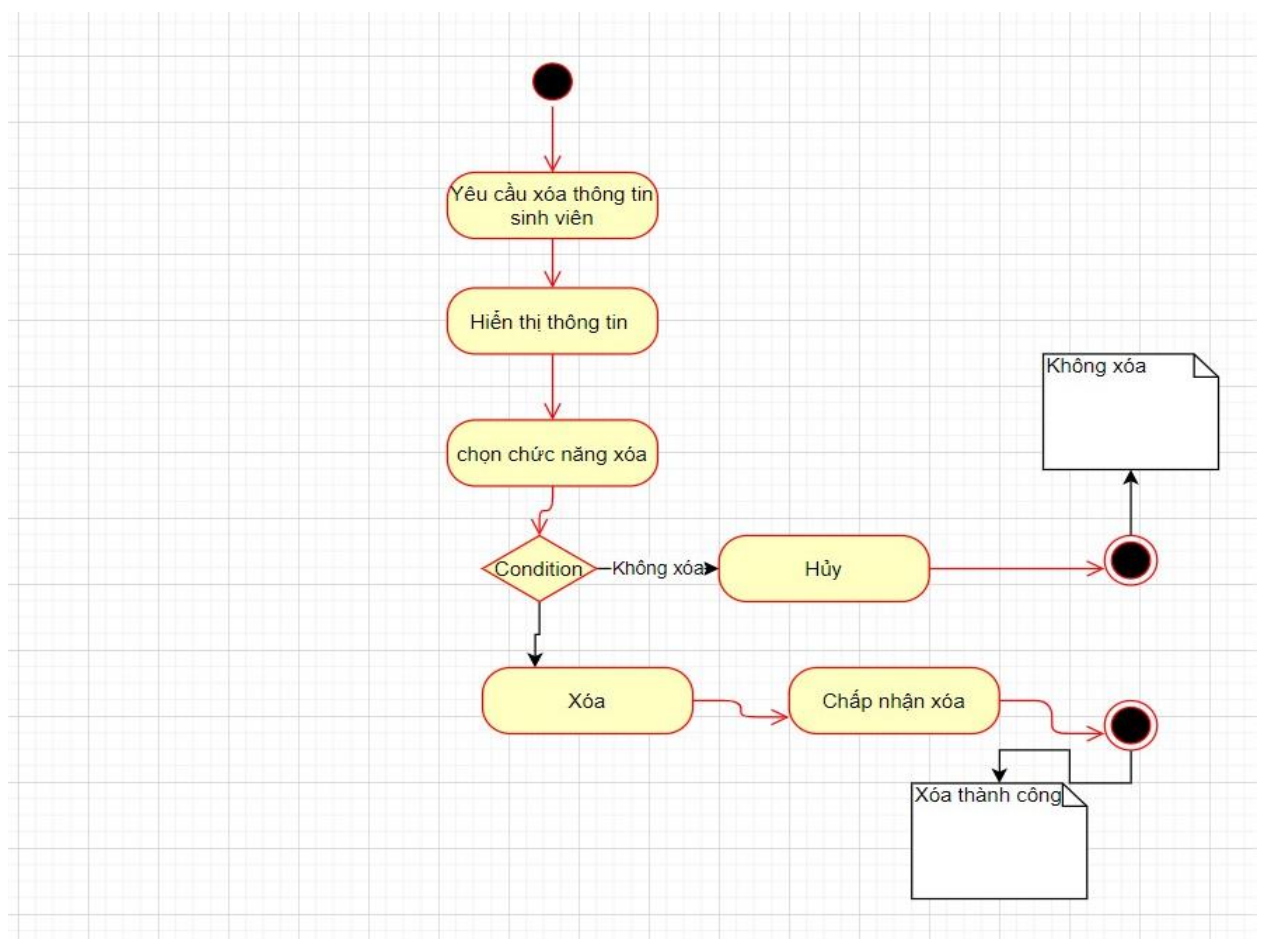
Hình 2.7.2 - Biểu đồ hoạt động sửa sinh viên

❖ Đặc tả quy trình hoạt động sửa thông tin sinh viên:

- Người dùng yêu cầu sửa thông tin sinh viên vào hệ thống hiển thị giao diện sửa thông tin sinh viên.

- Khi hệ thống hỏi người dùng có muốn sửa hay không.
- Hệ thống yêu cầu tác nhân chọn bài cần sửa và nhập thông tin muốn sửa. Sau khi người dùng nhập các thông tin cần sửa thì hệ thống sẽ kiểm tra giá trị đã sửa xem đúng hay không.
- + Nếu dữ liệu vừa nhập vào đúng thì sẽ chấp nhận việc sửa. Lúc này hệ thống cập nhật lại thông tin đã sửa cho người dùng kiểm tra và kết thúc quy trình sửa bản tin.
- + Nếu dữ liệu vừa nhập sai thì hệ thống yêu cầu người dùng nhập lại và trở lại quy trình nhập thông tin ban đầu.

2.7.3. Đặc tả biểu đồ hoạt động xóa sinh viên

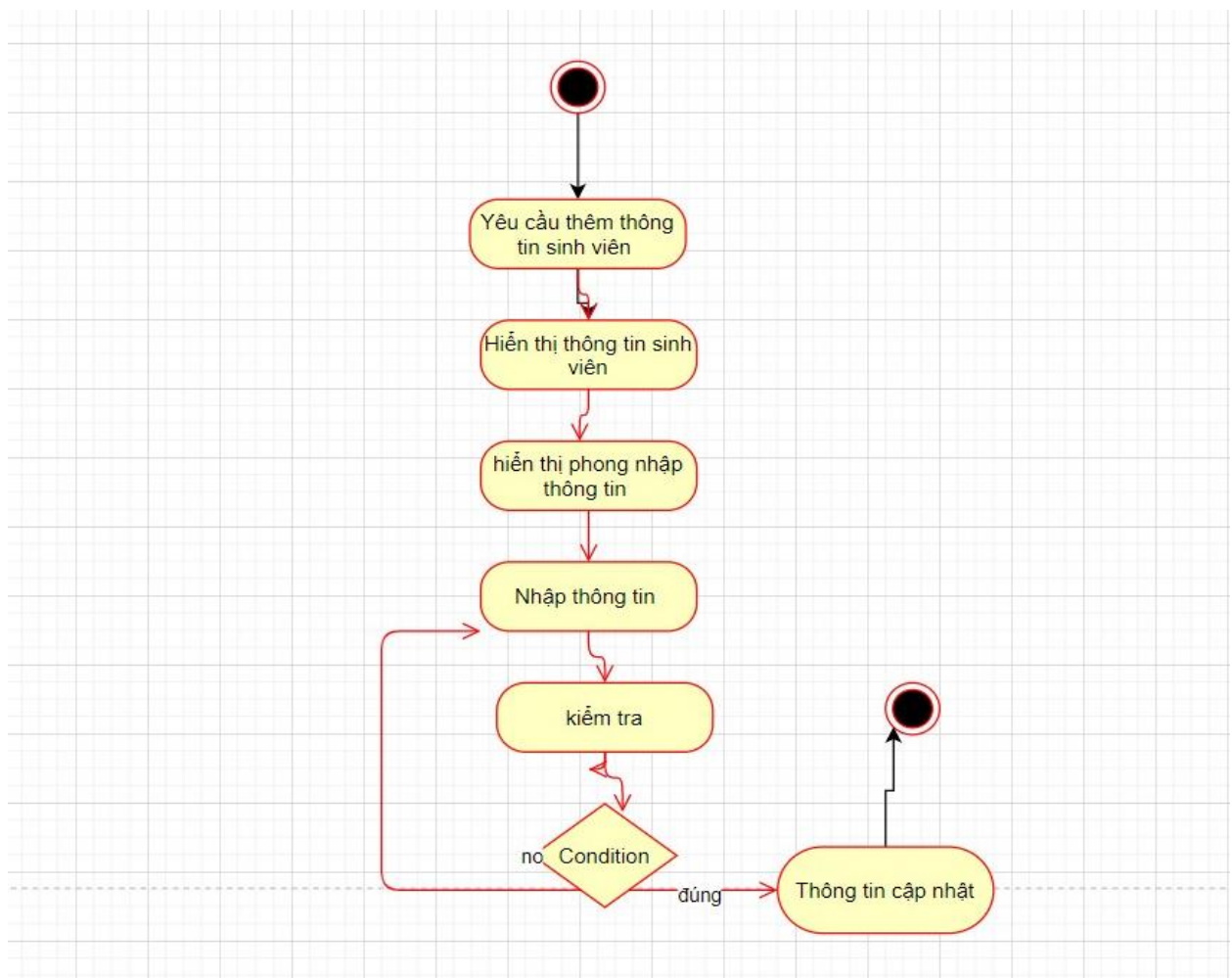


Hình 2.7.3 - Biểu đồ hoạt động xóa sinh viên

- ❖ Đặc tả quy trình hoạt động xóa sinh viên:
 - Người dùng yêu cầu xóa sinh viên.

- Hệ thống sẽ hiển thị giao diện xóa bài viết người dùng có thể xóa một sinh viên.
- Sau khi chọn chức năng xóa bài viết. Hệ thống hỏi lại đã chắc chắn muốn xóa chưa.
- + Nếu không đồng ý thì hệ thống sẽ hủy việc xóa thông tin sinh viên và kết thúc quá trình xóa và trường hợp này việc xóa thông tin sinh viên không thành công.
- + Nếu đồng ý, hệ thống yêu cầu người dùng chọn danh sách các thông tin sinh viên muốn xóa và chấp nhận xóa.
- Kết thúc quy trình xóa sinh viên.

2.7.4. Đặc tả biểu đồ hoạt động thêm sinh viên



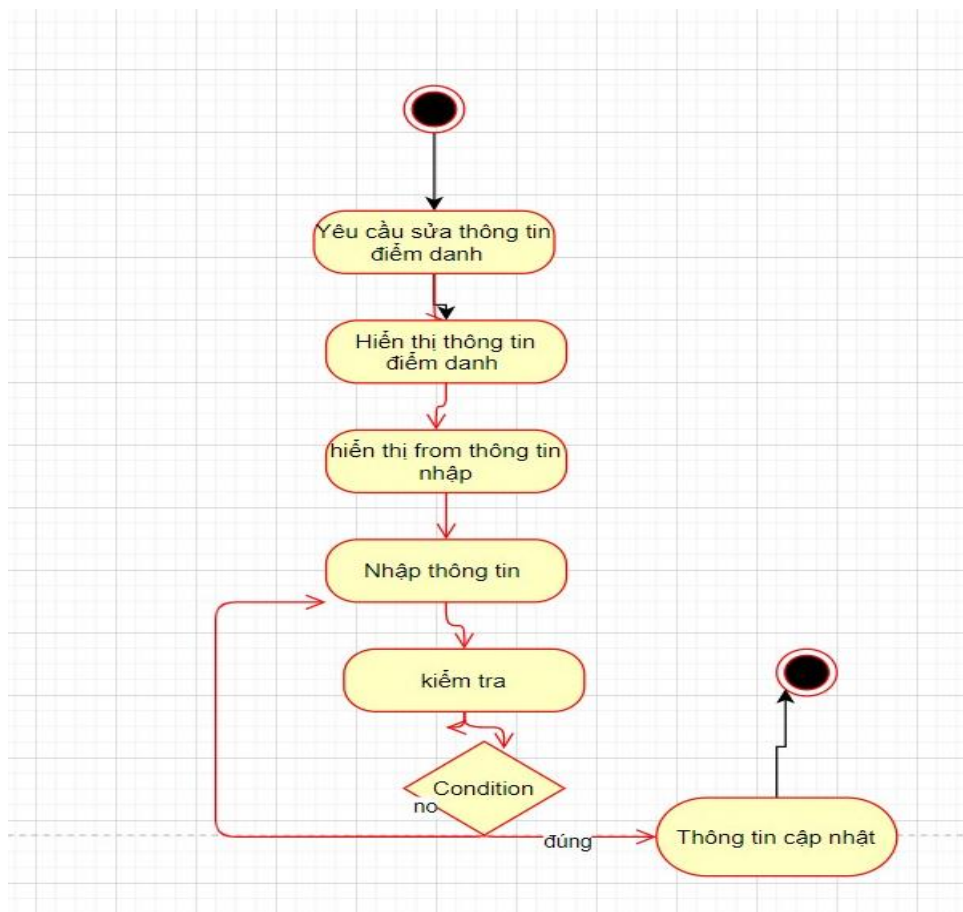
Hình 2.7.4 - Biểu đồ hoạt động thêm sinh viên

❖ Đặc tả quy trình hoạt động thêm sinh viên:

- Người dùng yêu cầu thêm mới sinh viên.

- Hệ thống sẽ hiển thị giao diện thêm mới sinh viên.
- Sau đó hệ thống yêu cầu người dùng nhập thông tin cho sinh viên để hoàn thành quy trình thêm mới bài viết với các thông tin như: tên sinh viên, địa chỉ, số điện thoại lớp,...
- Người dùng nhập các thông tin thì hệ thống sẽ kiểm tra các dữ liệu đầu vào xem đúng hay không. Kết quả lúc này sẽ trả ra 2 giá trị: Đúng hoặc sai.
- + Nếu đúng thì cho phép thêm sinh viên và việc thêm mới sinh viên thành công.
- + Ngược lại nếu sai thì hệ thống yêu cầu nhập lại và quay lại việc nhập thông tin và kiểm tra thông tin nhận.
- Kết thúc quy trình thêm mới sinh viên.

2.7.5. Đặc tả biểu đồ hoạt động sửa thông tin điểm danh

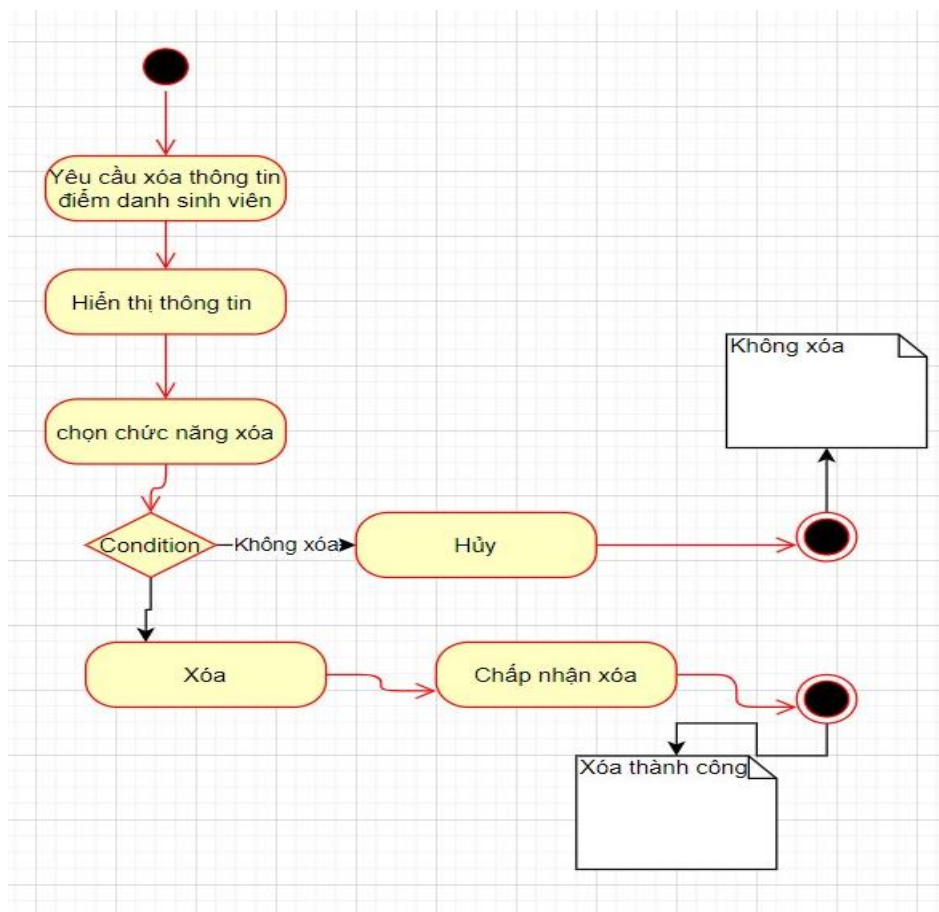


Hình 2.7.5 - Biểu đồ hoạt động sửa thông tin điểm danh

- ❖ Đặc tả quy trình hoạt động sửa thông tin điểm danh sinh viên:
- Người dùng yêu cầu sửa thông tin điểm danh vào hệ thống hiển thị giao diện sửa thông tin điểm danh sinh viên.

- Khi hệ thống hỏi người dùng có muốn sửa hay không.
- Hệ thống yêu cầu tác nhân chọn bài cần sửa và nhập thông tin muốn sửa. Sau khi người dùng nhập các thông tin cần sửa thì hệ thống sẽ kiểm tra giá trị đã sửa xem đúng hay không.
- + Nếu dữ liệu vừa nhập vào đúng thì sẽ chấp nhận việc sửa. Lúc này hệ thống cập nhật lại thông tin đã sửa cho người dùng kiểm tra và kết thúc quy trình sửa bản tin.
- + Nếu dữ liệu vừa nhập sai thì hệ thống yêu cầu người dùng nhập lại và trở lại quy trình nhập thông tin ban đầu.

2.7.6. Đặc tả biểu đồ hoạt động xóa thông tin điểm danh.

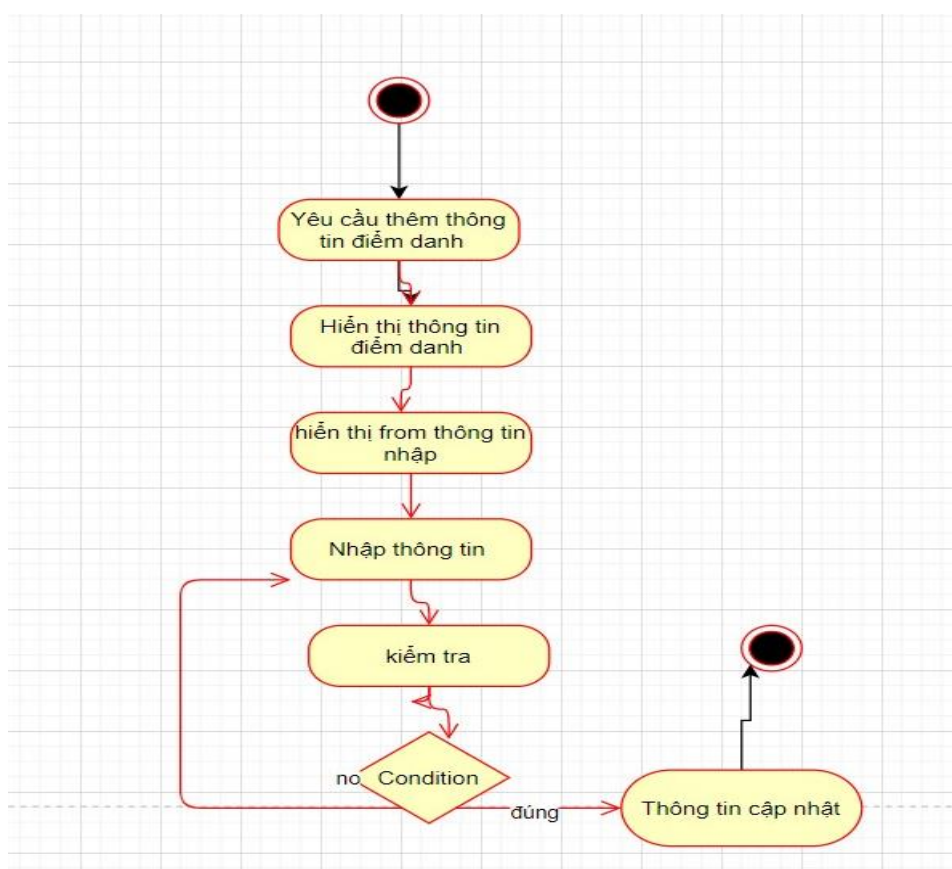


Hình 2.7.6 - Biểu đồ hoạt động xóa thông tin điểm danh

- ❖ Đặc tả quy trình hoạt động xóa thông tin điểm danh:
 - Người dùng yêu cầu xóa thông tin điểm danh sinh viên.

- Hệ thống sẽ hiển thị giao diện xóa bài viết người dùng có thể xóa một thông tin điểm danh của một sinh viên.
- Sau khi chọn chức năng xóa thông tin điểm danh. Hệ thống hỏi lại đã chắc chắn muốn xóa chưa.
- + Nếu không đồng ý thì hệ thống sẽ hủy việc xóa điểm danh và kết thúc quá trình xóa và trường hợp này việc điểm danh sinh viên không thành công.
- + Nếu đồng ý, hệ thống yêu cầu người dùng chọn danh sách thông tin muốn xóa và chấp nhận xóa.
- Kết thúc quy trình xóa thông tin điểm danh sinh viên.

2.7.7. Đặc tả biểu đồ hoạt động thêm thông tin điểm danh

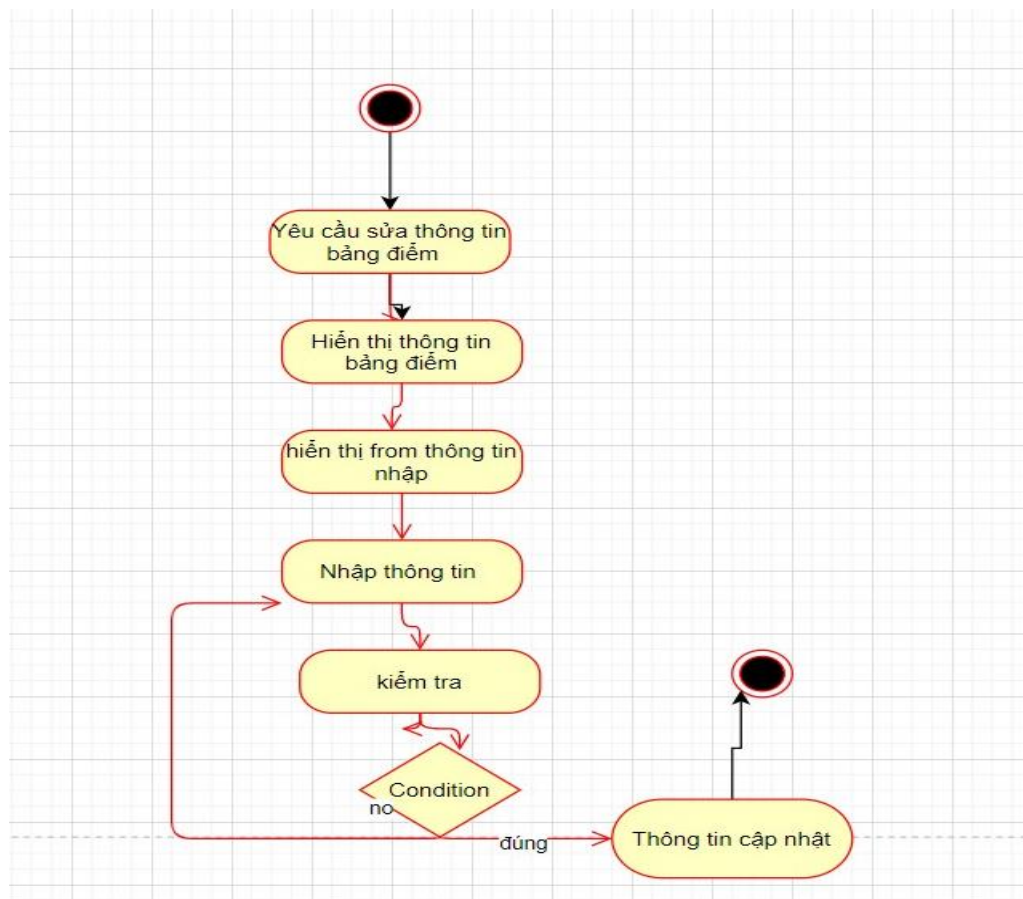


Hình 2.7.7 - Biểu đồ hoạt động thêm điểm danh sinh viên

- ❖ Đặc tả quy trình hoạt động thêm thông tin điểm danh:
 - Người dùng yêu cầu thêm mới thông tin điểm danh.
 - Hệ thống sẽ hiển thị giao diện thêm mới thông tin điểm danh.

- Sau đó hệ thống yêu cầu người dùng nhập thông tin điểm danh cho sinh viên để hoàn thành quy trình thêm mới bài viết với các thông tin như: tên sinh viên, thời gian điểm danh , ghi chú,...
- Người dùng nhập các thông tin thì hệ thống sẽ kiểm tra các dữ liệu đầu vào xem đúng hay không. Kết quả lúc này sẽ trả ra 2 giá trị: Đúng hoặc sai.
- + Nếu đúng thì cho phép thêm thông tin điểm danh và việc thêm mới thông tin điểm danh thành công.
- + Ngược lại nếu sai thì hệ thống yêu cầu nhập lại và quay lại việc nhập thông tin và kiểm tra thông tin nhận.
- Kết thúc quy trình thêm mới thông tin điểm danh.

2.7.8. Đặc tả biểu đồ hoạt động sửa thông tin bảng điểm



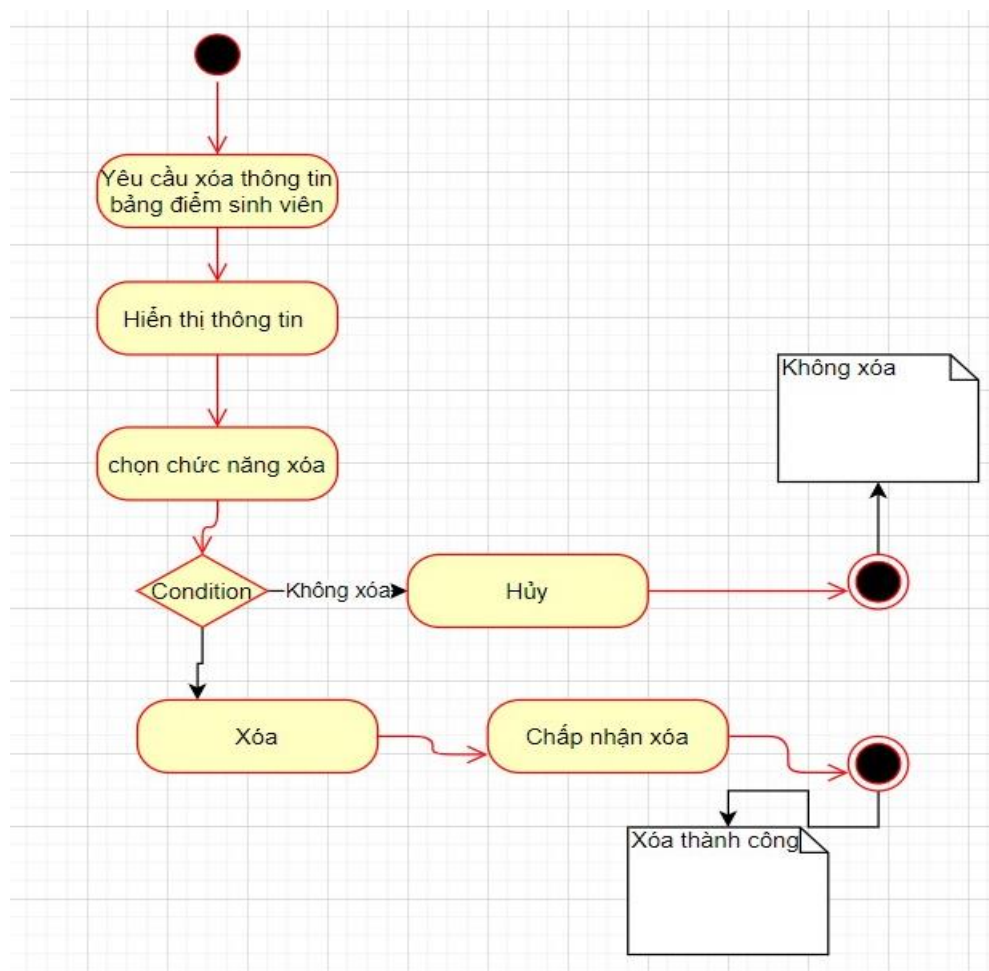
Hình 2.7.8 - Biểu đồ hoạt động sửa thông tin bảng điểm

❖ Đặc tả quy trình hoạt động sửa thông tin bảng điểm:

- Người dùng yêu cầu sửa thông tin bảng điểm vào hệ thống hiển thị giao diện sửa thông tin bảng điểm sinh viên.

- Khi hệ thống hỏi người dùng có muốn sửa hay không.
- Hệ thống yêu cầu tác nhân chọn bài cần sửa và nhập thông tin muốn sửa. Sau khi người dùng nhập các thông tin cần sửa thì hệ thống sẽ kiểm tra giá trị đã sửa xem đúng hay không.
- + Nếu dữ liệu vừa nhập vào đúng thì sẽ chấp nhận việc sửa. Lúc này hệ thống cập nhật lại thông tin đã sửa cho người dùng kiểm tra và kết thúc quy trình sửa bản tin.
- + Nếu dữ liệu vừa nhập sai thì hệ thống yêu cầu người dùng nhập lại và trở lại quy trình nhập thông tin ban đầu.

2.7.9. Đặc tả biểu đồ hoạt động xóa thông tin bảng điểm.

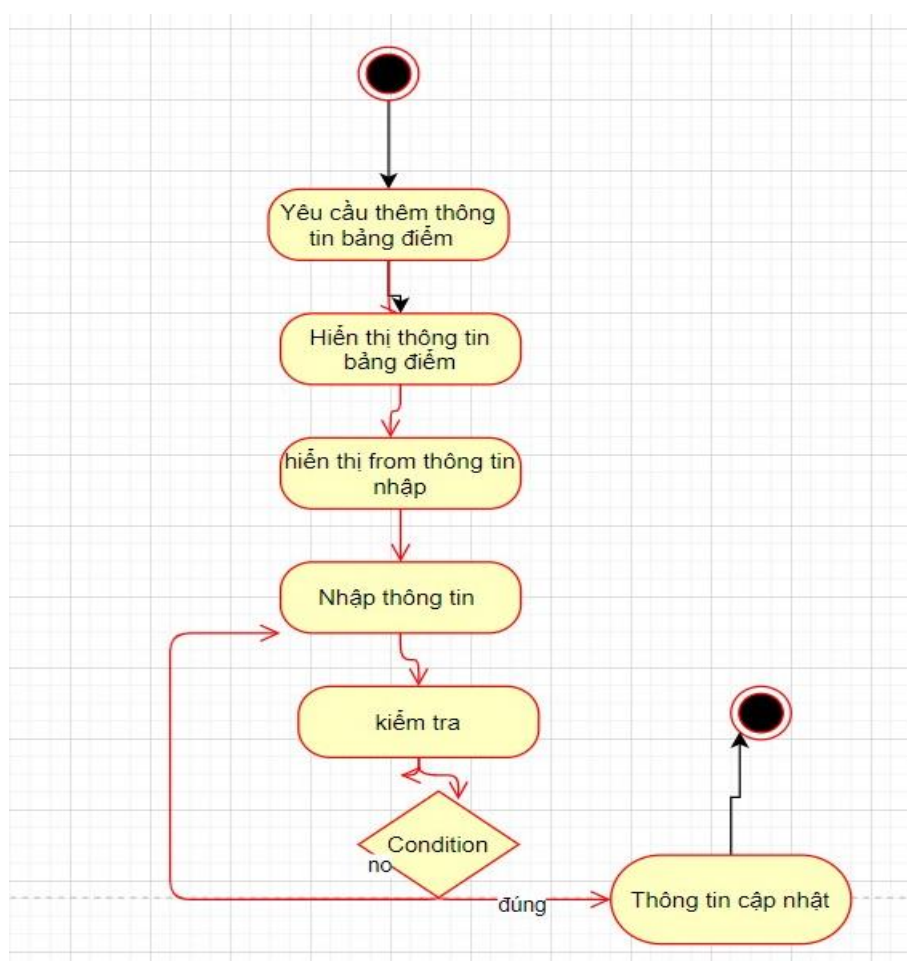


Hình 2.7.9 - Biểu đồ hoạt động xóa thông tin bảng điểm

- ❖ Đặc tả quy trình hoạt động xóa thông tin bảng điểm:
 - Người dùng yêu cầu xóa thông tin bảng điểm sinh viên.

- Hệ thống sẽ hiển thị giao diện xóa bài viết người dùng có thể xóa một thông tin bảng điểm của một sinh viên.
- Sau khi chọn chức năng xóa thông tin bảng điểm. Hệ thống hỏi lại đã chắc chắn muốn xóa chưa.
- + Nếu không đồng ý thì hệ thống sẽ hủy việc bảng điểm và kết thúc quá trình xóa và trường hợp này việc xóa bảng điểm không thành công.
- + Nếu đồng ý, hệ thống yêu cầu người dùng chọn danh sách bảng điểm muốn xóa và chấp nhận xóa.
- Kết thúc quy trình xóa thông tin bảng điểm sinh viên.

2.7.10. Đặc tả biểu đồ hoạt động thêm thông tin bảng điểm




Hình 2.7.10 - Biểu đồ hoạt động động bảng điểm sinh viên


- ❖ Đặc tả quy trình hoạt động thêm thông tin bảng điểm :
 - Người dùng yêu cầu thêm mới thông tin bảng điểm.
 - Hệ thống sẽ hiển thị giao diện thêm mới thông tin bảng điểm.

- Sau đó hệ thống yêu cầu người dùng nhập thông tin bảng điểm cho sinh viên để hoàn thành quy trình thêm mới bài viết với các thông tin như: tên sinh viên, điểm giữa kỳ, môn,...
- Người dùng nhập các thông tin thì hệ thống sẽ kiểm tra các dữ liệu đầu vào xem đúng hay không. Kết quả lúc này sẽ trả ra 2 giá trị: Đúng hoặc sai.
- + Nếu đúng thì cho phép thêm thông tin bảng điểm và việc thêm mới thông tin bảng điểm thành công.
- + Ngược lại nếu sai thì hệ thống yêu cầu nhập lại và quay lại việc nhập thông tin và kiểm tra thông tin nhận.
- Kết thúc quy trình thêm mới thông tin bảng điểm.


- Bảng tài khoản quản trị:

Name	Type	Length	Decimals	Not null	Virtual	Key	Comment
► Id	char	36	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1	
UserName	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Account	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
PassWord	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Image	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Mail	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Address	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
departmentId	char	36	0	<input type="checkbox"/>	<input type="checkbox"/>		
positionId	char	36	0	<input type="checkbox"/>	<input type="checkbox"/>		
createdAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
updatedAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		


- Bảng quyền hạn:

Name	Type	Length	Decimals	Not null	Virtual	Key	Comment
► Id	char	36	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1	
Note	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Title	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
createdAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
updatedAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		


- Bảng thông tin sinh viên:

Fields	Indexes	Foreign Keys	Triggers	Options	Comment	SQL Preview	
Name	Type	Length	Decimals	Not null	Virtual	Key	Comment
► Id	char	36	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1	
Note	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Last_Name	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Frist_Name	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Image	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Address	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
Brithday	datetime	0	0	<input type="checkbox"/>	<input type="checkbox"/>		
Code	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
learnclassId	char	36	0	<input type="checkbox"/>	<input type="checkbox"/>		
createdAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
updatedAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

- Bảng điểm danh:

Name	Type	Length	Decimals	Not null	Virtual	Key	Comment
Id	char	36	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1	
Times	int	11	0	<input type="checkbox"/>	<input type="checkbox"/>		
TimesDate	datetime	0	0	<input type="checkbox"/>	<input type="checkbox"/>		
Note	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		
accountId	char	36	0	<input type="checkbox"/>	<input type="checkbox"/>		
subjectId	char	36	0	<input type="checkbox"/>	<input type="checkbox"/>		
studentId	char	36	0	<input type="checkbox"/>	<input type="checkbox"/>		
createdAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
updatedAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

- Bảng điểm:

Name	Type	Length	Decimals	Not null	Virtual	Key	Comment
Id	char	36	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1	
PointCC	float	0	0	<input type="checkbox"/>	<input type="checkbox"/>		
PointKT1	float	0	0	<input type="checkbox"/>	<input type="checkbox"/>		
PointKT2	float	0	0	<input type="checkbox"/>	<input type="checkbox"/>		
PointGK	float	0	0	<input type="checkbox"/>	<input type="checkbox"/>		
PointT	float	0	0	<input type="checkbox"/>	<input type="checkbox"/>		
PointTK	float	0	0	<input type="checkbox"/>	<input type="checkbox"/>		
subjectId	char	36	0	<input type="checkbox"/>	<input type="checkbox"/>		
studentId	char	36	0	<input type="checkbox"/>	<input type="checkbox"/>		
createdAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
updatedAt	datetime	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

3.2. Thiết kế giao diện

- Giao diện đăng nhập


Login

Account

Account

Password

Enter password

LOGIN 


Copyright © 2020 . All rights reserved.

Hùng Phạm

- Giao diện người đăng nhập:

- Học sinh
- Học kỳ
- Bảng điểm
- Điểm danh

chào mừng hoàng thị cẩm



ảnh người dùng hoàng thị cẩm

Thông tin tài khoản đăng nhập

Tên người dùng : hoàng thị cẩm
 Tên tài khoản : hoàng thị cẩm
 Địa chỉ : hoàng quốc việt
 Mail : hoangtc@gmail.com
 Phòng Ban:
 Chức vụ :
 Quyền hạn đăng nhập : TEACHER

Copyright © 2020 . All rights reserved.

Hùng Phạm

- Giao diện sinh viên:

- Học sinh
- Học kỳ
- Bảng điểm
- Điểm danh

HỒNG THI CẨM

Danh Sách Học sinh

THÊM

STT	Code	Họ	Tên	Ngày sinh	Địa chỉ	Lớp	Chức Năng
1	1681310025	Lương	Nhật Duy	17/07/1998	Nam Định	Công nghệ phần mềm 1	CHỨC NĂNG
2	1681320010	Nguyễn	Hải Yến	09/07/1998	Hà Tây	Quản trị an ninh mạng 1	CHỨC NĂNG
3	1112536	Phạm	Văn Thịnh	11/07/1998	Thái Bình	Quản trị an ninh mạng 1	CHỨC NĂNG
4	1681310066	Đào	Thanh Liêm	01/06/1998	Hà Nam	Công nghệ phần mềm 1	CHỨC NĂNG

- Giao diện bảng điểm:

HỒNG THỊ CẨM

Học sinh

Học kỳ

Bảng điểm

Điểm danh

Bảng Điểm

THÊM

STT	Tên Môn học	Tên Học sinh	Kiểm tra 1	Kiểm tra 2	Giữa kì	Cuối kì	Thi	Tổng kết
1	Lập trình Game	Đào Thanh Liêm	7.5	7.5	7.5	7.5	7	7.15
2	Lập trình Game	Phạm Văn Hùng	7.5	7.5	7.5	7.5	8.5	8.2
3	Lập trình Game	Lương Nhật Duy	7.5	7.5	7.5	7.5	7	7.15
4	Lập trình trên thiết bị di động	Đào Thanh Liêm	2	2	2	2	2	2
5	Lập trình trên thiết bị di động	Lương Nhật Duy	8	8	8	8	9	8.7
6	Ngôn ngữ lập trình python	Lương Nhật Duy	8.5	8.5	8.5	8.5	8.5	8.5
7	Ngôn ngữ lập trình python	Đào Thanh Liêm	3	3	3	3	5	4.4
8	Lập trình web nâng cao	Phạm Văn Hùng	9	9	9	9	8.5	8.65

- Giao diện điểm danh:

CHÀO MỪNG : HỒNG THỊ CẨM

Học sinh

Học kỳ

Bảng điểm

Điểm danh

Danh điểm danh

THÊM

STT	Số tiết vắng	Ngày điểm danh	Ghi chú	Môn học	Tên học sinh	Giảng viên	Chức Năng
1	3	29/05/2020	vắng	Lập trình web nâng cao	Đào Thanh Liêm	hồng thị cẩm	CHỨC NĂNG

Copyright © 2020 . All rights reserved.

Hùng Phạm

KẾT LUẬN

Sau một thời gian tích cực làm việc, nghiên cứu của em với sự hướng dẫn, chỉ bảo nhiệt tình của giảng viên hướng dẫn – thầy cô giảng viên, em đã hoàn thành bản đồ án lập trình Web nâng cao.

Trong quá trình nghiên cứu và phát triển đề tài, dựa trên kinh nghiệm đang có, em đã rất cố gắng nhưng báo cáo này vẫn còn có nhiều thiếu sót. Em rất mong nhận được sự đóng góp, chỉ bảo của thầy, cô và các bạn khác trong lớp. Em xin chân thành cảm ơn!

TÀI LIỆU THAM KHẢO

- 1: Nguyễn Văn Ba, Phân tích và thiết kế hệ thống thông tin, NXB ĐH Quốc gia HN 2003.
- 2: Oracle Kiến trúc và quản trị.
- 3: SQL và PL/SQL cơ bản.
- 4: Enterprise DBA Part 1A: Architecture and Administration.
- 5: Glenn Maslen - Introduction to Oracle: SQL, SQL*PLUS, and PL/SQL Course Notes - 1992 - Oracle Corporation.