**TRƯỜNG ĐẠI HỌC ĐỒNG THÁP**

**KHOA SƯ PHẠM TOÁN – TIN**



0018413035 – ĐẶNG NGỌC BIN

**BÁO CÁO THỰC TẬP TỐT NGHIỆP**

**TÊN ĐỀ TÀI**

**XÂY DỰNG WEBSITE QUẢN LÝ ĐIỂM RÈN LUYỆN SINH VIÊN ĐẠI HỌC ĐỒNG THÁP**

***(Module: Nghiệp vụ xét điểm rèn luyện, xuất báo cáo thống kê)***

NGÀNH: KHOA HỌC MÁY TÍNH

LỚP: ĐHCNTT18A

*Đồng Tháp, 08 tháng 04 năm 2022*

**TRƯỜNG ĐẠI HỌC ĐỒNG THÁP**

**KHOA SƯ PHẠM TOÁN – TIN**



0018413035 – ĐẶNG NGỌC BIN

**BÁO CÁO THỰC TẬP TỐT NGHIỆP**

**TÊN ĐỀ TÀI**

**XÂY DỰNG WEBSITE QUẢN LÝ ĐIỂM RÈN LUYỆN SINH VIÊN ĐẠI HỌC ĐỒNG THÁP**

***(Module: Nghiệp vụ xét điểm rèn luyện chính, xuất báo cáo thống kê)***

NGÀNH: KHOA HỌC MÁY TÍNH

LỚP: ĐHCNTT18A

Cán bộ hướng dẫn: NGUYỄN TUẤN KIỆT

Giảng viên hướng dẫn: Ths. NGUYỄN THỊ THANH THẢO

*Đồng Tháp, 08 tháng 04 năm 2022*

# **LỜI CẢM ƠN**

Để bài báo cáo này đạt kết quả tốt đẹp, em đã nhận được sự hỗ trợ, giúp đỡ của rất nhiều đơn vị. Với tình cảm sâu sắc, chân thành, cho phép em được bày tỏ lòng biết ơn sâu sắc đến tất cả các cá nhân và cơ quan đã tạo điều kiện giúp đỡ trong quá trình học tập và hoàn thiện bài báo cáo tốt nghiệp này.

Trước hết em xin chân thành cảm ơn các thầy cô trong bộ môn Công nghệ thông tin đã giúp đỡ em trong quá trình học tập tại trường, đặc biệt là anh Nguyễn Tuấn Kiệt đã tận tình hướng dẫn trong việc thực hiện bài báo cáo này. Em xin chân thành cảm ơn các anh trong cơ quan Trung tâm Công Nghệ Thông Tin và Truyền thông đã giúp đỡ em trong suốt thời gian học tập vừa qua đồng thời tạo điều kiện cho em học hỏi được nhiều kiến thức mới để hoàn thành tốt bài thực tập.

Vì kiến thức bản thân còn hạn chế, trong quá trình thực tập, hoàn thiện chuyên đề này em không tránh khỏi những lỗi sai sót, kính mong nhận được những ý kiến đóng góp từ thầy, cô cũng như quý lãnh đạo của cơ quan nơi thực tập.

Xin chân thành cảm ơn !

TP. Cao Lãnh, ngày 08 tháng 04 năm 2022 Sinh viên thực tập

Đặng Ngọc Bin

**MỤC LỤC**

[**LỜI CẢM ƠN** 1](#_Toc102491804)

[**MỤC LỤC** 2](#_Toc102491805)

[**DANH MỤC HÌNH** 6](#_Toc102491806)

[**DANH MỤC BẢNG** 9](#_Toc102491807)

[**PHẦN MỘT: MỞ ĐẦU** 10](#_Toc102491808)

[**1.** **Đặt vấn đề** 10](#_Toc102491809)

[**2.** **Ý nghĩa lý thuyết, thực tiễn của đề tài** 11](#_Toc102491810)

[**2.1.** **Ý nghĩa lý thuyết** 11](#_Toc102491811)

[**2.2.** **Ý nghĩa thực tiễn** 11](#_Toc102491812)

[**3.** **Mục tiêu đề tài** 11](#_Toc102491813)

[**4.** **Phạm vi đề tài** 11](#_Toc102491814)

[**5.** **Các phương pháp nghiên cứu** 12](#_Toc102491815)

[**PHẦN HAI: NỘI DUNG** 14](#_Toc102491816)

[**CHƯƠNG I: CƠ SỞ LÝ THUYẾT** 14](#_Toc102491817)

[**1.1.** **Mục đích và yêu cầu** 14](#_Toc102491818)

[**1.2.** **Đánh giá kết quả rèn luyện và khung chấm điểm** 14](#_Toc102491819)

[**1.3.** **Giao diện** 16](#_Toc102491820)

[**1.3.1.** **Giới thiệu Bootstrap** 16](#_Toc102491821)

[**1.3.2.** **Lịch sử của Bootstrap** 17](#_Toc102491822)

[**1.3.3.** **Cấu trúc và tính năng của Bootstrap** 19](#_Toc102491823)

[**1.3.4.** **Cách cài đặt Bootstrap** 19](#_Toc102491824)

[**1.4.** **ASP.NET** 20](#_Toc102491825)

[**1.4.1.** **Giới thiệu ASP.NET** 20](#_Toc102491826)

[**1.4.2.** **Điểm nổi bật của ASP.NET 5x** 21](#_Toc102491827)

[**1.4.3.** **Nền tảng và hỗ trợ của Microsoft** 21](#_Toc102491828)

[**1.5. ASP.NET MVC** 21](#_Toc102491829)

[**1.5.1. Giới thiệu Mô hình ASP.NET MVC** 21](#_Toc102491830)

[**1.5.2. Lợi ích của ASP.NET MVC** 23](#_Toc102491831)

[**1.5.3. Các tính năng của ASP.NET MVC** 23](#_Toc102491832)

[**1.5.4. Ưu điểm của mô hình MVC** 23](#_Toc102491833)

[**1.5.5. Nhược điểm của mô hình MVC** 24](#_Toc102491834)

[**1.5.6. Lịch sử phiên bản ASP.NET MVC** 24](#_Toc102491835)

[**1.5.7. Tổng kết** 25](#_Toc102491836)

[**1.6. Hệ quản trị cơ sở dữ liệu SQL Server** 25](#_Toc102491837)

[**1.6.1. Khái niệm** 25](#_Toc102491838)

[**1.6.2.Tính năng quản lý** 25](#_Toc102491839)

[**1.6.3.Vai trò của SQL server** 26](#_Toc102491840)

[**1.7.Phân tích thiết kế hệ thống hướng đối tượng và UML** 26](#_Toc102491841)

[**1.7.1.Khái niệm OOAD (Object Oriented Analysis and Design)** 26](#_Toc102491842)

[**1.7.2.Khái niệm về UML (Unified Modeling Language)** 26](#_Toc102491843)

[**1.7.3.Tại sao lại là OOAD và UML?** 26](#_Toc102491844)

[**1.7.4.OOAD sử dụng UML** 27](#_Toc102491845)

[**CHƯƠNG II PHÂN TÍCH THIẾT KẾ HỆ THỐNG** 31](#_Toc102491846)

[**2.1. Sơ đồ Usecase** 31](#_Toc102491847)

[**2.1.1. Sơ đồ Usecase tổng quát** 31](#_Toc102491848)

[**2.1.2. Sơ đồ Usecase thành phần** 32](#_Toc102491849)

[**2.1.3.Sơ đồ Usecase chi tiết** 33](#_Toc102491850)

[**2.1.4 Sơ đồ UseCase quản lý khoa** 33](#_Toc102491851)

[**2.1.5 Sơ đồ UseCase quản lý lớp** 34](#_Toc102491852)

[**2.1.6. Sơ đồ UseCase quản lý tài khoản** 34](#_Toc102491853)

[**2.1.7.Sơ đồ UseCase quản lý danh mục (Menu)** 35](#_Toc102491854)

[**2.1.8. Sơ đồ UseCase quản lý quyền** 36](#_Toc102491855)

[**2.2. Cơ sở dữ liệu** 37](#_Toc102491856)

[**2.3. Sơ đồ hoạt động** 43](#_Toc102491857)

[**2.3.1. Sơ đồ đăng nhập** 43](#_Toc102491858)

[**2.3.2. Sơ đồ hoạt động sinh viên** 44](#_Toc102491859)

[**2.3.3. Sơ đồ hoạt động cán bộ lớp** 45](#_Toc102491860)

[**2.3.4. Sơ đồ hoạt động các bộ khoa** 45](#_Toc102491861)

[**2.4. Sơ đồ tuần tự** 46](#_Toc102491862)

[**2.4.1 Sơ đồ tuần tự đăng nhập:** 46](#_Toc102491863)

[**2.4.2.Xét điểm cán bộ lớp** 47](#_Toc102491864)

[**2.4.3. Xét điểm các bộ khoa** 47](#_Toc102491865)

[**2.5. Sơ đồ hợp tác** 48](#_Toc102491866)

[**2.6. Sơ đồ thành phần** 48](#_Toc102491867)

[**2.7. Sơ đồ triển khai** 49](#_Toc102491868)

[**2.8. Kết luận** 49](#_Toc102491869)

[**CHƯƠNG III THIẾT KẾ VÀ CÀI ĐẶT WEBSITE** 51](#_Toc102491870)

[**3.1. Giao diện thông tin rèn luyện sinh viên** 51](#_Toc102491871)

[**3.2.** **Giao diện xét điểm rèn luyện của sinh viên** 51](#_Toc102491872)

[**3.3.** **Giao diện các bộ lớp xét điểm rèn luyện cho các bạn sinh viên của lớp** 52](#_Toc102491873)

[**3.4.** **Giao diện mở và duyệt điểm rèn luyện của cán bộ khoa** 55](#_Toc102491874)

[**3.5.** **Giao diện duyệt điểm rèn luyện cán bộ khoa** 56](#_Toc102491875)

[**3.6.** **Quản trị hệ thống** 60](#_Toc102491876)

[**3.6.1.** **Khoa** 60](#_Toc102491877)

[**3.6.2.** **Lớp** 61](#_Toc102491878)

[**3.7.** **Cài đặt website** 64](#_Toc102491879)

[**3.7.1.** **Môi trường phát triển ứng dụng** 64](#_Toc102491880)

[**3.8**. **Kết luận** 64](#_Toc102491881)

[**PHẦN BA: KẾT LUẬN** 65](#_Toc102491882)

[**1.** **Kết quả đạt được:** 65](#_Toc102491883)

[**2. Hướng phát triển:** 65](#_Toc102491884)

[**TÀI LIỆU THAM KHẢO** 67](#_Toc102491885)

[**1.** **Sách, giáo trình** 67](#_Toc102491886)

[**2.** **Các website tham khảo** 67](#_Toc102491887)

# **DANH MỤC HÌNH**

[Hình 1.1 bootstrap js 18](#_Toc102491888)

[Hình 1.2. Giao diện bootstrap 19](#_Toc102491889)

[Hình 1.3. Giao diện bootstrap 20](#_Toc102491890)

[Hình 1.4 Giao diện mô hình MVC 22](#_Toc102491891)

[Hình 1.5 Giao diện luồng dữ liệu 22](#_Toc102491892)

[Hình 1.6. Các View trong OOAD sử dụng UML 27](#_Toc102491893)

[Hình 1.7. Các bản vẽ trong OOAD sử dụng UML 28](#_Toc102491894)

[Hình 1.8. Ký hiệu về Use Case 29](#_Toc102491895)

[Hình 1.9. Ký hiệu về Class 30](#_Toc102491896)

[Hình 1.10. Ký hiệu về Actor 30](#_Toc102491897)

[Hình 2.1. Sơ đồ tổng quát 31](#_Toc102491898)

[Hình 2.2 Sơ đồ usecase cán bộ khoa 32](#_Toc102491899)

[Hình 2.3. Sơ đồ usecase các bộ lớp 32](#_Toc102491900)

[Hình 2.4. Sơ đồ usecase sinh viên 32](#_Toc102491901)

[Hình 2.5. Sơ đồ chi tiết 33](#_Toc102491902)

[Hình 2.6. Sơ đồ Usecase quản lý khoa 34](#_Toc102491903)

[Hình 2.7. Sơ đồ Usecase quản lý lớp 34](#_Toc102491904)

[Hình 2.8. Sơ đồ Usecase quản lý danh mục 35](#_Toc102491905)

[Hình 2.9. Sơ đồ Usecase quản lý danh mục 36](#_Toc102491906)

[Hình 2.10. Sơ đồ Usecase quản lý danh mục 36](#_Toc102491907)

[Hình 2.11 Sơ đồ diagram 43](#_Toc102491908)

[Hình 2.12. Sơ đồ hoạt động đăng nhập 44](#_Toc102491909)

[Hình 2.13. Sơ đồ hoạt động sinh viên 44](#_Toc102491910)

[Hình 2.14 Sơ đồ hoạt động cán bộ lớp 45](#_Toc102491911)

[Hình 2.15 Sơ đồ hoạt tạo kỳ đánh giá cán bộ khoa 45](#_Toc102491912)

[Hình 2.16 Sơ đồ hoạt động cán bộ khoa 46](#_Toc102491913)

[Hình 2.17. Sơ đồ tuần tự đăng nhập 46](#_Toc102491914)

[Hình 2.18 Sơ đồ tuần tự xét điểm 47](#_Toc102491915)

[Hình 2.19 Sơ đồ tuần tự xét điểm 48](#_Toc102491916)

[Hình 2.20 Sơ đồ hợp tác xét điểm cán bộ lớp 48](#_Toc102491917)

[Hình 2.21. Sơ đồ thành phần 49](#_Toc102491918)

[Hình 2.22 Sơ đồ triển khai 49](#_Toc102491919)

[Hình 3.1 Giao diện sinh viên 51](#_Toc102491920)

[Hình 3.2 Giao diện thông báo tạo phiếu đánh giá 51](#_Toc102491921)

[Hình 3.3 Giao diện đánh giá phiếu đánh giá 52](#_Toc102491922)

[Hình 3.4 Giao diện xem năm học kỳ đánh giá mở hoặc đóng 52](#_Toc102491923)

[Hình 3.5 Giao diện xuất danh sách lớp 53](#_Toc102491924)

[Hình 3.6 Giao diện cán bộ xem danh sách sinh viên chấm điểm 53](#_Toc102491925)

[Hình 3.7 Giao diện cán lớp chấm điểm lại cho sinh viên 54](#_Toc102491926)

[Hình 3.8 Giao diện cán lớp chấm điểm lại cho sinh viên 55](#_Toc102491927)

[Hình 3.9 Giao diện tạo kỳ đánh giá 55](#_Toc102491928)

[Hình 3.10 Giao diện kỳ đánh giá 56](#_Toc102491929)

[Hình 3.11 Giao diện kỳ đánh giá đã tạo 56](#_Toc102491930)

[Hình 3.12 Giao diện kỳ danh sách lớp trong kỳ đánh giá 56](#_Toc102491931)

[Hình 3.13 Giao diện xem lớp mở của kỳ đánh giá 57](#_Toc102491932)

[Hình 3.14 Giao diện xem lớp mở của kỳ đánh giá 57](#_Toc102491933)

[Hình 3.15 Giao diện danh sách sinh viên kỳ đánh giá mở 58](#_Toc102491934)

[Hình 3.16 Giao diện đánh giá phiếu rèn luyện cán bộ khoa 58](#_Toc102491935)

[Hình 3.16 Giao diện đánh giá phiếu rèn luyện cán bộ khoa 59](#_Toc102491936)

[Hình 3.17 Giao diện quản lý khoa 60](#_Toc102491937)

[Hình 3.18 Giao diện thêm khoa mới 60](#_Toc102491938)

[Hình 3.19 Giao diện cập nhật khoa 61](#_Toc102491939)

[Hình 3.20 Giao diện xóa khoa 61](#_Toc102491940)

[Hình 3.21 Giao diện lớp 62](#_Toc102491941)

[Hình 3.22 Giao diện thêm lớp 62](#_Toc102491942)

[Hình 3.23 Giao diện cập nhật lớp 63](#_Toc102491943)

[Hình 3.24 Giao diện xóa lớp 63](#_Toc102491944)

# **DANH MỤC BẢNG**

[Bảng 2.1 Tài khoản 37](#_Toc102491945)

[Bảng 2.2 Tài khoản quyền 37](#_Toc102491946)

[Bảng 2.3 Tài khoản thông báo 38](#_Toc102491947)

[Bảng 2.4 Thông báo 38](#_Toc102491948)

[Bảng 2.5 Tiêu chí 38](#_Toc102491949)

[Bảng 2.6 Chức vụ 39](#_Toc102491950)

[Bảng 2.7 Điểm 39](#_Toc102491951)

[Bảng 2.8 Khoa 39](#_Toc102491952)

[Bảng 2.9 Ký đăng ký 40](#_Toc102491953)

[Bảng 2.10 Lớp 40](#_Toc102491954)

[Bảng 2.11 Lớp đăng ký 40](#_Toc102491955)

[Bảng 2.12 Menu 41](#_Toc102491956)

[Bảng 2.13 Phiếu đánh giá 41](#_Toc102491957)

[Bảng 2.14 Quyền 42](#_Toc102491958)

[Bảng 2.15 Quyền menu 42](#_Toc102491959)

# 

# **PHẦN MỘT: MỞ ĐẦU**

1. **Đặt vấn đề**

Trước đây, khi ngành công nghệ thông tin chưa phát triển, người ta lưu trữ dữ liệu hoàn toàn bằng giấy, điều này đã gây ra những trở ngại rất lớn trong việc cất giữ, tìm kiếm, bảo quản, nghiên cứu và phổ biến kiến thức. Cho nên, cùng với sự phát triển của mạng Internet, các website lần lượt ra đời, đây là một trong những tiện ích quan trọng của công nghệ thông tin, Sự ra đời của các website đã giải quyết cơ bản những trở ngại trên, việc lưu trữ dữ liệu trở nên dễ dàng và an toàn hơn, khả năng tìm kiếm nhanh chóng hơn và việc chia sẽ dữ liệu vô cùng tiện lợi. Chính điều này làm cho việc tiếp thu kiến thức cũng trở nên dễ dàng hơn. Mọi giới, mọi tầng lớp điều có thể tìm kiếm được những thông tin bổ ích cho mình với chi phí vừa phải. Nhằm đáp ứng nhu cầu tìm kiếm, quản lý và tra cứu thông tin ngày càng cao của người sử dụng mạng Internet, hàng loạt các website với nhiều đề tài và mục đích khsc nhau được ra đời.

Ngoài ra, nhu cầu tin học khóa các chương trình quản lý trong các lĩnh vực hiện đang rất lớn trong đó có trường Đại học Đồng Tháp. Ứng dụng tin học trong công tác quản lý đã phát triển mạnh mẽ giúp cho công tác quản lý ngày càng trở nên hiệu quả hơn như nâng cao hiệu suất trong công việc, đưa ra các báo cáo , các số liệu thống kê một cách nhanh chóng chính xác và kịp thời. Đồng thời nhờ có biệc ứng dụng tịn học đã tiết kiệm được rất nhiều thời gian, công sức của con người, nó làm giảm nhẹ bộ máy quản lý cốn rất cồng kềnh từ trước tới nay.

Hiện nay, công tác chấm điểm rèn luyện không còn xa lạ với hầu hết các sinh viên trong trường. Việc chấm điểm rèn luyện cho sinh viên nhằm đánh giá phẩm chất chính trị, đạo đức, lối sống của từng sinh viên theo các mức điểm đạt được trong khung điểm được đưa ra trước, điểm rèn luyện của sinh viên cũng làm cơ sở để xét các công tác khác như: công tác Đoàn, Đảng… Tuy nhiên, hiện tại công tác xét điểm rèn luyện của trường Đại học Đồng Tháp phải làm thủ công, nghĩa là trường phải in phiếu điểm rèn luyện ra phát cho tất cả sinh viên, sinh viên chấm điểm và nộp lại sau đó cán bộ lớp chấm điểm lại cho sinh viên, cuối cùng cán bộ Khoa duyệt và thống kê lại kết quả. Việc làm thủ công như thế mất nhiều thời gian cho giảng viên, sinh viên cũng như các bộ phận chức năng có liên quan, việc thông kê kết quả mất rất nhiều thời gian, chi phí cho nhà trường và đi ngược với xu thế tin học hóa ngày nay.

Vì vậy việc xây dựng một ứng dụng web để giải quyết những tồn tại trên là hết sức cần thiết, giúp nhà trường tiết kiệm cho phí, giúp cho giảng viên, sinh viên có nhiều thời gian hơn để giảng dạy, học tập hay giải quyết các công việc khác.

Do đó, xây dựng website Quản lý điểm rèn luyện của sinh viên trường Đại học Đồng Tháp là mục tiêu của đề tài này.

Với những cách tiếp cận trên, chúng ta thấy quản lý đánh giá điểm rèn luyện đối với sinh viên tại trường đại học đã trở thành một nhiệm vụ có tính thời sự và tính thực tiễn cao. Xuất phát từ những lí do trên em lựa chọn đề tài: **“Xét điểm rèn luyện sinh viên trường Đại học Đồng Tháp"** do cô **Nguyễn Thị Thanh Thảo** và các anh trong cơ quan Trung tâm Công Nghệ và Thông tin đặc biệt là người hướng dẫn thực tập tụi em là Anh **Nguyễn Tuấn Kiệt** người hướng dẫn trực tiếp. Trong quá trình nghiên cứu có một số vấn đề chưa tối ưu mong thầy góp ý để em có thể hoàn thiện một cách tối ưu nhất.

1. **Ý nghĩa lý thuyết, thực tiễn của đề tài**
   1. **Ý nghĩa lý thuyết**

Trong đề tài này sẽ nghiên cứu về các thành phần lý thuyết có liên quan để xây dựng một hệ thống website cụ thể.

* Tìm hiểu về quy chế đánh giá kết quả điểm rèn luyện.
* Quy trình chấm điểm rèn luyện của sinh viên.
* Phân loại kết quả điểm rèn luyện
* Mục đích của việc đánh giá điểm rèn luyện
  1. **Ý nghĩa thực tiễn**

Đề tài này thành công sẽ mang lại giá trị như sau:

* Sinh viên có thể kiểm tra kết quả của mình trước khi có thông báo từ trường.
* Có hệ thống giúp sinh viên đánh giá và lưu điểm rèn luyện của mình
* Giúp khoa quản lý phiếu rèn luyện của sinh viên một cách dễ dàng và thuận tiện.

1. **Mục tiêu đề tài**

Trên cơ sở nghiên cứu lý luận và đánh giá thực trạng quản lý đánh giá kết quả rèn luyện cho sinh viên khoa đề xuất một số giải pháp quản lý đánh giá kết quả rèn luyện cho sinh viên đáp ứng yêu cầu đổi mới căn bản và toàn diện giáo dục nước ta trong giai đoạn hiện nay.

1. **Phạm vi đề tài**

Trong khuôn khổ có giới hạn của một Đồ án tốt nghiệp cũng như hạn chế về thời gian thực hiện, đề tài này chỉ tập trung nghiên cứu việc tin học hóa khâu chấm, duyệt điểm rèn luyện của sinh viên. Và hệ thống này được áp dụng đối với sinh viên trường Đại học Đồng Tháp

1. **Các phương pháp nghiên cứu**

- Nghiên cứu quản lý đánh giá kết quả rèn luyện cho sinh viên của khoa đạt hiệu quả tốt nhất khách quan và minh bạch.

*Đọc tài liệu*

* Nghiên cứu những tài liệu có liên quan đến những gì cần thực hiện.

*Học hỏi từ người hướng dẫn thực tập, bạn bè*

* Học cách định hướng.
* Sắp xếp thời gian.
* Phương pháp giải quyết vấn đề.

**6. Kế hoạch thực hiện đề tài**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thời gian**  **(từ ngày 11/01/2022 đến ngày 08/04/2022)** | **Nội dung thực hiện** | **Người thực hiện** | **Ghi chú** |
| 1 | 17/01/2022 -21/01/2022 | Tìm tài liệu liên quan | ĐẶNG NGỌC BIN |  |
| 2 | 24/01/2022 -28/01/2022 | Tìm hiểu về .Net 5 |  |
| 3 | 07/02/2022 -11/02/2022 | Tìm hiểu Mô hình .Net MVC |  |
| 4 | 14/02/2022 -18/02/2022 | Tìm hiểu về Bootstrap 4,5 |  |
| 5 | 21/02/2022 -25/02/2022 | Xây dựng Nghiệp vụ Website |  |
| 6 | 28/02/2022 -04/03/2022 | Hoàn thành Nghiệp vụ |  |
| 7 | 07/03/2022 -11/03/2022 | Thiết kế UI/UX và CSDL |  |
| 8 | 14/03/2022 -18/03/2022 | Hoàn thiện UI/UX và CSDL |  |
| 9 | 21/03/2022 -25/03/2022 | Xây dựng các chức năng cơ bản |  |  |
| 10 | 28/03/2022 -01/04/2022 | Triển khai hệ thống  vào thực tế |  |  |
| 11 | 28/03/2022 -01/04/2022 | Kiểm thử toàn hệ thống |  |  |
| 12 | 04/04/2022 -08/04/2022 | Hoàn thành  báo cáo thực tập |  |  |

# **PHẦN HAI: NỘI DUNG**

# **CHƯƠNG I:** **CƠ SỞ LÝ THUYẾT**

* 1. **Mục đích và yêu cầu**

- Mục đích của việc đánh giá kết quả rèn luyện của sinh viên Trường nhằm:

+ Góp phần thực hiện mục tiêu giáo dục là đào tạo con người Việt Nam phát triển toàn diện, có đạo đức, tri thức, sức khỏe, thẩm mỹ và nghề nghiệp, trung thành với lý tưởng độc lập dẫn tộc và Chủ nghĩa Xã hội.

+ Đưa ra được những định hướng, nội dung rèn luyện cụ thể, tạo điều kiện cho sinh viên có môi trường rèn luyện chất lượng.

- Yêu cầu

+ Việc đánh giá kết quả rèn luyện của sinh viên trường là việc làm thường xuyên ở mỗi học kỳ của Trường.

+ Quá trình đánh giá phải đảm bảo chất lượng chính xác, công bằng, công khai và dân chủ.

* 1. **Đánh giá kết quả rèn luyện và khung chấm điểm**

Xem xét, đánh giá kết quả rèn luyện của học sinh, sinh viên là đánh giá phẩm chất chính trị, đạo đức, lối sống của từng học sinh, sinh viên. Điểm rèn luyện được đánh giá bằng thang điểm 100.

* + - **Đánh giá về ý thức học tập** ***( Điểm tối đa 20 điểm)***

- Căn cứ để xác định điểm là tinh thần vượt khó, phấn đấu vươn lên trong học tập, tích cực tham gia nghiên cứu khoa học, tham gia dự thi học sinh, sinh viên giỏi ở các cấp.

- Đi học đầy đủ, đúng giờ, chuẩn bị bài tập đầy đủ, tích cực phát biểu trong giờ học (*đi học muộn, nghỉ học không có lý do, bỏ giờ: trừ 1 điểm/ lần* )

* + Ý thức, thái độ tham gia
* Các hoạt động học thuật
* Hoạt động ngoại khóa
* Rèn luyện kỹ năng mềm
* Hoạt động nghiêm cứu khoa học
* Các cuộc thi do Nhà trường, Đoàn thanh niên, Hội Sinh viên phát động
  + Tinh thần vượt khó, phấn đấu vươn lên trong học tập
  + Tích cực đánh giá hoạt động giảng dạy của giảng viên, khóa đào tạo
  + Kết quả học tập
* Điểm TBCHT: 2,00 đến 2,49
* Điểm TBCHT: 2,50 đến 3,19
* Điểm TBCHT: 3,20 đến 3,59
* Điểm TBCHT: 3,60 đến 4,00
* *Điểm thưởng (được cộng nhưng tổng số điểm của tiêu chí này không vượt quá 10 điểm)*
  + - **Đánh giá ý thức chấp hành nội quy, quy chế, quy định trong Nhà trường *(Điểm tối đa 25 điểm)***

-Căn cứ để xác định điểm là ý thức và kết quả việc chấp hành các nội quy,quy chế và các quy định khác được áp dụng trong trường.

Thực hiện tốt nội quy lớp, quy định của Nhà trường.

Thực hiện tốt Quy chế học sinh, sinh viên; Quy định của Nhà trường đối với sinh viên ngoại trú.

Tham gia đầy đủ bảo hiểm y tế, bảo hiểm tai nạn

* + - **Đánh giá về ý thức và kết quả tham gia các hoạt động chính trị, xã hội, văn hóa, văn nghệ, thể thao, phòng chống tội phạm và các tệ nạn xã hội. (Điểm tối đa 20 điểm)**

Căn cứ để xác định điểm là kết quả tham gia các hoạt động rèn luyện về chính trị –xã hội, văn hoá, văn nghệ, thể thao và phòng chống các tệ nạn xã hội

Tham gia nhiệt tình, đầy đủ và có ý thức các hoạt động tập trung do Nhà trường, Khoa, Đoàn Thanh niên, Hội Sinh viên tổ chức (Trừ 05 điểm/1 lần vắng mặt)

Có ý thức và tích cực tham gia tuyên truyền công tác phòng chống tệ nạn xã hội, hoạt động công ích, tình nguyện, từ thiện, nhân đạo, công tác xã hội.

Kết quả phân loại đoàn viên:

- Khá

- Xuất sắc

* + - **Đánh giá về ý thức công dân trong quan hệ cộng đồng *(Điểm tối đa 25 điểm)***

Căn cứ để xác định điểm là kết quả chấp hành các chủ trương của Đảng,chính sách, pháp luật của Nhà nước, thành tích trong công tác xã hội và giữ gìn anninh chính trị, trật tự an toàn xã hội; mối quan hệ cộng đồng, tinh thần giúp đỡ bạnbè, cưu mang người gặp khó khăn.

Chấp hành tốt và tích cực tham gia tuyên truyền các chủ trương của Đảng, chính sách và pháp luật của nhà nước trong cộng đồng.

Tham gia đầy đủ các buổi học tập, tìm hiểu pháp luật do Nhà trường tổ chức.

Tích cực tham gia các hoạt động xã hội do Nhà trường hoặc địa phương nơi cư trú tổ chức.

Có mối quan hệ tốt trong tập thể, không gây mất đoàn kết, bản thân có tác dụng tích cực đối với tập thể.

Thực hiện tốt tinh thần tương thân, tương ái trong cuộc sống.

* + - **Đánh giá về ý thức và kết quả khi tham gia công tác cán bộ lớp, các đoàn thể, tổ chức trong trường hoặc người học đạt được thành tích đặc biệt trong học tập, rèn luyện. (Điểm tối đa 10 điểm)**

Căn cứ để xác định điểm là ý thức, mức độ hoàn thành nhiệm vụ đối với học sinh, sinh viên được phân công quản lý lớp, các tổ chức Đảng, Đoàn thanh niên,Hội sinh viên và các tổ chức khác trong nhà trường; và những học sinh, sinh viênkhác có thành tích đặc biệt trong học tập, rèn luyện.

Không là cán bộ lớp, cán bộ Đoàn, Hội nhưng thực hiện tốt nhiệm vụ được giao.

Là cán bộ lớp, cán bộ Đoàn, Hội nhưng không thực hiện tốt nhiệm vụ được giao, không gương mẫu trước tập thể.

Nếu là cán bộ lớp, cán bộ Đoàn, Hội thì căn cứ vào kết quả thi đua của tập thể lớp quy định như sau:

Chức vụ Xếp loại

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Chức vụ** | **Xếp loại** | | | |
| XS | Tốt | Khá | TBK |
| Cấp trưởng | 10 | 9 | 8 | 6 |
| Cấp phó | 8 | 7 | 6 | 4 |

**\* Ghi chú:**

1. Nếu sinh viên vi phạm quy chế thi, kết quả đánh giá rèn luyện trong học kỳ không vượt quá loại Trung bình .

2. Nếu sinh viên vi phạm quy chế học sinh, sinh viên bị xử lý từ cảnh cáo trở lên khi đánh giá kết quả rèn luyện không được vượt quá loại Trung bình.

* 1. **Giao diện**
     1. **Giới thiệu Bootstrap**

Bootstrap là một framework bao gồm các HTML, CSS và JavaScript template dùng để phát triển website chuẩn responsive.

Bootstrap là một bộ sưu tập miễn phí của các mã nguồn mở và công cụ dùng để tạo ra một mẫu webiste hoàn chỉnh. Với các thuộc tính về giao diện được quy định sẵn như kích thước, màu sắc, độ cao, độ rộng…, các designer có thể sáng tạo nhiều sản phẩm mới mẻ nhưng vẫn tiết kiệm thời gian khi làm việc với framework này trong quá trình thiết kế giao diện website.

* + 1. **Lịch sử của Bootstrap**

Bootstrap là sản phẩm của Mark Otto và Jacob Thornton tại Twitter. Nó được xuất bản như là một mã nguồn mở vào ngày 19/8/2011 trên GitHub. Tên gọi ban đầu là Twitter Blueprint.

Đến ngày 31/1/2012, Bootstrap phiên bản 2 đã được phát hành. Bootstrap 2 được bổ sung bố cục lưới 12 cột với thiết kế tùy chỉnh đáp ứng cho nhiều màn hình kích thước. Tiếp nối sự thành công của phiên bản 2, Bootstrap 3 ra đời vào ngày 19/8/2013 với giao diện tương thích với smartphone.

Chỉ 3 năm sau ngày ra mắt, Bootstrap đã trở thành No.1 project trên GitHub. Vào tháng 10/2014, Mark Otto công bố phiên bản Bootstrap 4 đang được phát triển. Phiên bản alpha đầu tiên của Bootstrap 4 đã được triển khai vào tháng 8/2015. Phiên bản mới nhất của Bootstrap được giới thiệu đến người dùng là Bootstrap 4.3.1. Cho đến nay, Bootstrap vẫn là một trong những framework thiết kế website có lượng người dùng “khủng” nhất.

* **3 file chính của Bootstrap:**

Dưới đây là 3 File chính giúp quản lý các chức năng của Website và giao diện người dùng:

* **Bootstrap.CSS**
* Nó là một Framework CSS có chức năng quản lý và sắp xếp bố cục của các trang Web. HTML có nhiệm vụ quản lý cấu trúc và nội dung thì CSS sẽ xử lý bố cục của Website. Vì vậy mà hai cấu trúc này phải tồn tại cùng nhau để thực hiện được hành động cụ thể.
* Giờ bạn sẽ không phải tốn thời gian để chỉnh sửa thủ công chỉ để thay đổi một thiết kế nhỏ. Thay vào đó, bạn có thể dùng CSS để tạo giao diện thống nhất trên nhiều Website mà không bị giới hạn.
* Hàm CSS không chỉ giới hạn ở kiểu văn bản mà còn dùng định dạng như bảng, bố cục hình. Bạn cần mất một ít thời gian để CSS ghi nhớ hết tất cả các khai báo và bộ chọn.
* **Bootstrap.JS**

****

Hình 1.1 bootstrap js

* Đây là phần cốt lõi vì chứa các File JavaScript, nó chịu trách nhiệm cho việc tương tác của Website. Để tiết kiệm thời gian viết cú pháp JavaScript mà nhiều nhà phát triển sẽ sử dụng jQuery. Đây là thư viện JavaScript mã nguồn mở, đa nền tảng giúp bạn thêm nhiều chức năng vào trang Web.
* Theo kinh nghiệm của tôi, jQuery sẽ thực hiện một số chức năng như:
* Thực hiện các yêu cầu của Ajax như: loại trừ dữ liệu đến từ một vị trí khác một cách linh hoạt.
* Giúp tạo các tiện ích bằng bộ sưu tập Plugin JavaScript.
* jQuery sẽ tạo hình động tùy chỉnh bằng các thuộc tính của CSS.
* Thêm một số tính năng động cho nội dung các trang Web của bạn.
* Tuy Bootstrap với các thuộc tính Element HTML và CSS có thể hoạt động tốt, nhưng vẫn cần jQuery để tạo thiết kế Responsive. Nếu thiếu đi phần này, bạn chỉ có thể dùng các phần tĩnh của CSS mà thôi.
* **Glyphicons**
* Trong giao diện trang Web, phần không thể thiếu chính là Icons. Chúng thường được liên kết với các dữ liệu nhất định và các hành động trong giao diện người dùng.
* Bootstrap dùng Glyphicons để đáp ứng nhu cầu trên và nó đã mở khóa bộ Halflings Glyphicons để bạn sử dụng miễn phí. Tại bản miễn phí, tuy chỉ có giao diện chuẩn nhưng phù hợp với các chức năng thiết yếu.
* Nếu như bạn muốn có các Icon phong cách hơn thì có thể mua bộ bộ Icon Premium khác nhau. Chắc chắn các Icon tại Glyphicons sẽ giúp cho trang Web của bạn trông đẹp và nổi bật hơn.
  + 1. **Cấu trúc và tính năng của Bootstrap**



Hình 1.2. Giao diện bootstrap

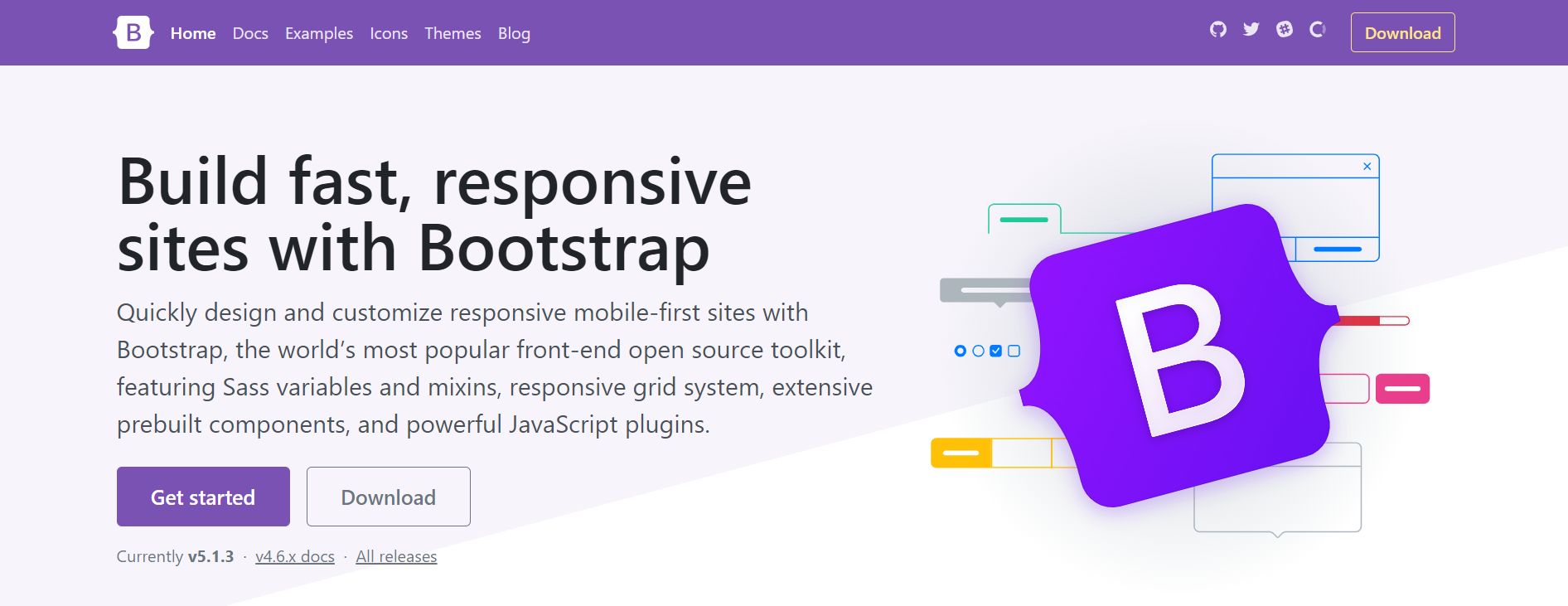
Bootstrap chứa các tập tin JavaScript, CSS và fonts đã được biên dịch và nén lại. Ngoài ra, Bootstrap được thiết kế dưới dạng các mô-đun. Do đó, dễ dàng tích hợp với hầu hết các mã nguồn mở như WordPress, Joomla, Magento, …Trong đó, Bootstrap mang đến nhiều chức năng nổi bật.

* Bootstrap cho phép người dùng truy cập vào thư viện “khổng lồ” các thành tố dùng để tạo nên giao diện của một website hoàn chỉnh như font, typography, form, table, grid…
* Bootstrap cho phép bạn tùy chỉnh framework của website trước khi tải xuống và sử dụng nó tại trang web của khung.
* Tái sử dụng các thành phần lặp đi lặp lại trên trang web.
* Bootstrap được tích hợp jQuery. Bạn chỉ cần khai báo chính xác các tính năng trong quá trình lập trình web của bạn.
* Định nghĩa glyphicons nhằm giảm thiểu việc sử dụng hình ảnh làm biểu tượng và tăng tốc độ tải trang.
  + 1. **Cách cài đặt Bootstrap**

Biết về Bootstrap là gì vẫn chưa đủ. Bạn cần phải biết cách cài đặt và sử dụng Bootstrap như thế nào? Có hai cách phổ biến để tải Bootstrap về web hosting của bạn:

* Tải trực tiếp từ trang cung cấp Bootstrap
* Thông qua CDN Bootstrap

***Tải về từ trang Bootstrap***

******

Hình 1.3. Giao diện bootstrap

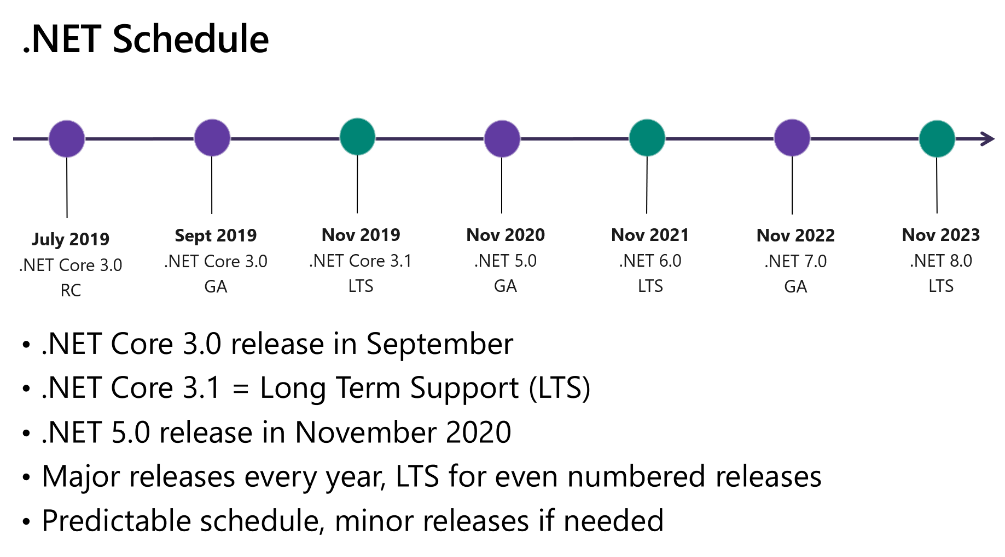
* 1. **ASP.NET**
     1. **Giới thiệu ASP.NET**

.NET Microsoft công bố tại hội nghị Build 2019. NET dựa trên Net Standard, tuy nhiên không phải mọi tính năng của framework sẽ xuất hiện trong .NET 5. Nền tảng .NET sẽ là sự hợp nhất của tất cả các nền tảng .NET hiện tại, không còn các “biệt danh” đi kèm để phân biệt như hiện nay như Core hay Framework. Tất cả thống nhất về một tên gọi chung: .NET.

Để khẳng định đây không phải là một nền tảng xây dựng mới (như khi .NET Core ra đời), nhưng cũng không phải đơn giản chỉ là một sự phát triển từ một nền tảng đang có sẵn, nhóm lựa chọn tên gọi đơn giản là .NET, và định chỉ số phiên bản là 5. Nó thể hiện sự kế thừa và hợp nhất.

Theo đó, sẽ không có .NET Core 4 nữa. Phiên bản 3 là phiên bản cuối cùng của .NET Core. Cũng sẽ không có các phiên bản. NET Framework kế tiếp nữa. 4.8 là phiên bản cuối cùng của .NET Framework.

Sau đó sẽ chỉ còn một .NET, và bắt đầu từ phiên bản 5.



*Hình 1.4. Sơ đồ phát triển .NET*

* + 1. **Điểm nổi bật của ASP.NET 5x**

Có nhiều cải tiến quan trọng trong .NET 5.0

* Hỗ trợ Windows ARM64.
* Cải tiến phát triển máy tính để bàn Windows.
* Các API trình tuần tự JSON được cải tiến.
* Chú thích kiểu tham chiếu không thể xóa.
* Đầu tư web và đám mây(cloud).
* Các ứng dụng tệp tin đơn và hình ảnh vùng chứa nhỏ hơn.
* Cải thiện hiệu suất, với hiệu suất gRPC được cho là vượt qua Go, C++.
* Các ứng dụng .NET đầy đủ với framework giao diện người dùng web Blazor, hỗ trợ Blazor Server và Blazor WebAssembly, hỗ trợ các thư viện khung .NET Core và đã được thực hiện nhanh hơn trong .NET 5.
* Một mô hình mới được đặc trưng để hỗ trợ các API WinRT, bao gồm gọi các API, Hệ thống tương tác WinRT hiện có đã bị xóa khỏi thời gian chạy .NET.
* C# 8 và F# 5 cung cấp các cải tiến ngôn ngữ mới như các chương trình và bản ghi cấp cao nhất cho C# 9, trong khi F# 5 cung cấp lập trình tương tác và tăng hiệu suất cho lập trình chức năng trên .NET.
* Thư viện .NET đã nâng cao hiệu suất cho tuần tự hóa Json, biểu thức chính quy (regular expression) và HTTP (HTTP 1.1, HTTP /2). Hiện tại chúng cũng được chú thích hoàn toàn về tính vô hiệu.
* Độ trễ P95 đã giảm.
* Các tùy chọn triển khai ứng dụng tốt hơn.
  + 1. **Nền tảng và hỗ trợ của Microsoft**

.NET 5.0 hỗ trợ đa nền tảng gần giống như .NET Core 3.1, dành cho Windows, macOS và Linux. Nếu đang sử dụng .NET Core 3.1 trên hệ điều hành được hỗ trợ, có thể sử dụng .NET 5.0 trên cùng phiên bản hệ điều hành đó. Bổ sung quan trọng nhất cho .NET 5.0 là Windows Arm64.

.NET 5.0 là bản phát hành hiện tại. Điều đó có nghĩa là nó sẽ được hỗ trợ cho đến khi .NET 6.0 được phát hành, .NET 6.0 sẽ được phát hành LTS và sẽ được hỗ trợ giống như .NET Core 3.1.

**1.5. ASP.NET MVC**

**1.5.1. Giới thiệu Mô hình ASP.NET MVC**

**ASP.NET MVC** là một framework web được phát triển bởi Microsoft, dựa trên mô hình MVC (model-view-controller). ASP.NET MVC được Microsoft giới thiệu lần đầu tiên tại phiên bản .Net Framework 3.5, bây giờ đã trờ thành mã nguồn mở, tách rời với thành phần độc quyền ASP.NET Web Forms.

ASP.NET MVC đang nổi lên là phương pháp triển web mạnh nhất và phổ biến nhất trên nền ASP.NET hiện tại.

MVC là viết tắt của Model, View và Controller. MVC tách ứng dụng thành ba thành phần: Model, View và Controller.

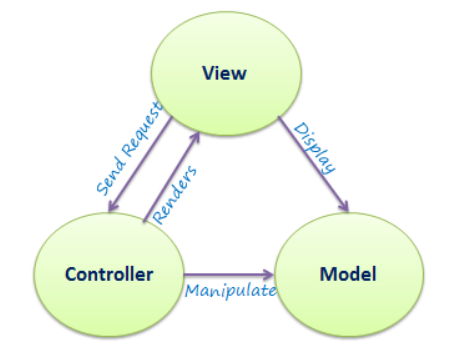
**Model:** đại diện cho hình dạng của dữ liệu. Nó duy trì dữ liệu của ứng dụng. Các đối tượng model lấy và lưu trữ trạng thái mô hình trong cơ sở dữ liệu.

*“Model đại diện cho dữ liệu.”*

**View:** là giao diện người dùng. View hiển thị dữ liệu của model cho người dùng và cũng cho phép họ sửa đổi dữ liệu.

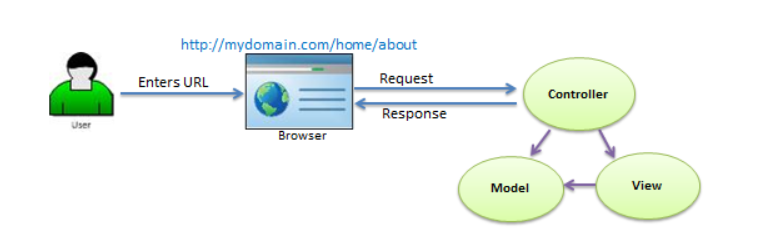
*“View là giao diện người dùng.”*

**Controller:** Xử lý yêu cầu của người dùng. Thông thường, người dùng tương tác với view, những tương tác này sẽ tạo ra yêu cầu tương ứng và sẽ được xử lý bởi controller. Controller trả về view thích hợp kèm theo dữ liệu model dưới dạng phản hồi.



Hình 1.4 Giao diện mô hình MVC

Dưới đây minh họa luồng yêu cầu của người dùng trong ASP.NET MVC.



Hình 1.5 Giao diện luồng dữ liệu

Theo hình trên, khi người dùng nhập URL vào trình duyệt, nó sẽ đến máy chủ và gọi controller thích hợp.

Sau đó, controller sử dụng view với model phù hợp để tạo phản hồi và gửi lại cho người dùng. Chúng ta sẽ thấy chi tiết của sự tương tác trong một vài phần tiếp theo.

**1.5.2. Lợi ích của ASP.NET MVC**

Nền tảng ASP.NET MVC mang lại những lợi ích sau:

* Dễ dàng quản lý sự phức tạp của ứng dụng bằng cách chia ứng dụng thành ba phần model, view, controller
* Nó không sử dụng view state hoặc sever-based form. Điều này tốt cho những lập trình viên muốn quản lý hết các khía cạnh của một ứng dụng.
* Nó sử dựng mẫu Front Controller, mẫu này giúp quản lý các requests chỉ thông qua một Controller. Nhờ đó bạn có thể thiết kế một hạ tầng quản lý định tuyến.
* Hỗ trợ tốt hơn cho mô hình phát triển ứng dụng hướng kiểm thử (TDD)
* Nó hỗ trợ tốt cho các ứng dụng được xây dựng bởi những đội có nhiều lập trình viên và thiết kế mà vẫn quản lý được tính năng của ứng dụng.

**1.5.3. Các tính năng của ASP.NET MVC**

Tách bạch các tác vụ của ứng dụng (Login nhập liệu, business logic, và logic giao diện), dễ dàng kiểm thử và mặc định áp dụng hướng phát triển TDD. Tất cả các tính năng chính của mô hình MVC được cài đặt trên interface và kiểm thử bằng cách sử dụng các đối tượng mocks, mock object là các đối tượng mô phỏng các tính năng của những đối tượng thực sự trong ứng dụng.

MVC là nền tảng có khả năng mở rộng (extensible) & nhúng (pluggable). Các thành phần của ASP.NET MVC được thiết kế để chúng có thể được thay thế một cách dễ dàng hoặc dễ dàng tùy chỉnh. Bạn có thể nhúng thêm view engine, cơ chế định tuyến cho URL, cách kết xuất tham số của action- method và các thành phần khác. ASP.NET MVC cũng hỗ trợ việc sử dụng Dependency Injection(DI) và Inversion of Control(IoC).

ASP.NET MVC có thành phần ánh xạ URL mạnh mẽ cho phép bạn xây dựng những ứng dụng có các địa chỉ URL xúc tích và dễ tìm kiếm. Các địa chỉ URL không cần phải có phần mở rộng của tên tệp tin và được thiết kế để hỡ trợ các mẫu định dạng tên phù hợp với việc tối ưu tìm kiếm (URL) và phù hợp với lập địa chỉ theo kiểu REST.

Hỗ trợ các tính năng có sẵn của ASP.NET như cơ chế xác thực người dùng, quản lý thành viên, quyền, output caching và data caching, session và profile, quản lý trạng thái ứng dụng, hệ thống cấu hình…

**1.5.4. Ưu điểm của mô hình MVC**

Kiểm tra dễ dàng: Với MVC, bạn có thể dễ dàng kiểm tra, rà soát lỗi phần mềm trước khi tới tay người tiêu dùng, đảm bảo chất lượng và độ uy tín cao hơn.

Chức năng control: Trên các nền website thì ngôn ngữ lập trình như CSS, HTML, Javascript có một vai trò vô cùng quan trọng. Việc sử dụng mô hình MVC sẽ giúp bạn có một bộ control ưu việt trên nền tảng các ngôn ngữ hiện đại với nhiều hình thức khác nhau.

View và size: View sẽ là nơi lưu trữ các dữ liệu. Càng nhiều yêu cầu được thực hiện thì kích thước càng tệp càng lớn. Khi đó, đường truyền mạng cũng giảm tốc độ load. Việc sử dụng mô hình MVC sẽ giúp bạn tiết kiệm được diện tích băng thông một cách tối ưu.

Chức năng Soc (Separation of Concern): Chức năng này cho phép bạn phân tách rõ ràng các phần như Model, giao diện, data, nghiệp vụ.

Tính kết hợp: Việc tích hợp ở mô hình MVC cho phép bạn thoải mái viết code trên nền tảng website. Khi đó, server của bạn sẽ được giảm tải khá nhiều.

Đơn giản: Đây là một mô hình với kết cấu tương đối đơn giản. Dù bạn không có quá nhiều chuyên môn cũng có thể sử dụng được.

**1.5.5. Nhược điểm của mô hình MVC**

MVC thường được sử dụng vào những dự án lớn. Do đó, với các dự án nhỏ, mô hình MVC có thể gây cồng kềnh, tốn thời gian trong quá trình phát triển cũng như thời gian trung chuyển dữ liệu.

**1.5.6. Lịch sử phiên bản ASP.NET MVC**

|  |  |  |  |
| --- | --- | --- | --- |
| **MVC Version** | **.Net Version** | **Release date** | **Features** |
| MVC 1.0 | .NET 3.5 | 13/03/2009 | * MVC architecture with webform engine * Routing * HTML Helpers * Ajax Helpers * Auto binding |
| MVC 2.0 | .NET 3.5/4.0 | 10/03/20010 | * Area * Asynchronous controller * Html helper * Methods with lambda expression * DataAnnotations attributes * Client side validation * Custon template * Scaffolding |
| MVC 3.0 | .NET 4.0 | 13/01/2011 | * Unobtrusive javascript validation * Razor view engine * Global filters * Remote validation * Dependency resolver for loC * ViewBag |
| MVC 4.0 | .NET 4.0/4.5 | 15/08/2012 | * Mobile project template * Bundling and minification * Support for Windows Azure SDK |
| MVC 5.0 | .NET 4.5 | 17/10/2013 | * Authentication filters * Bootstrap support * New scaffolding items * ASP.Net Identity |
| MVC 5.2.7 – Cureent | .Net 4.8 | 18/-4/2019 | * Attribute based routinh * Bug fixes and minor features update |

**1.5.7. Tổng kết**

Mô hình MVC được ứng dụng trong nhiều ngôn ngữ lập trình khác nhau, nhưng phổ biến nhất là ứng dụng ASP.NET MVC hay PHP MVC.MVC đang là mô hình được ứng dụng rất nhiều trong lập trình.Việc sử dụng mô hình tương đối đơn giản. Chỉ cần hiểu rõ quy trình vận hành, nắm được các chức năng của từng bộ phận thì việc triển khai mô hình MVC tương đối dễ dàng.

**1.6. Hệ quản trị cơ sở dữ liệu SQL Server**

**1.6.1. Khái niệm**

SQL Server (viết tắt của cụm từ Structured Query Language) là một hệ thống quản lý cơ sở dữ liệu quan hệ (Relational Database Management System, viết tắt là RDBMS). SQL Server có khả năng hỗ trợ một số lượng lớn các quy trình xử lý giao dịch, ứng dụng doanh nghiệp và ứng dụng phân tích trong các công ty hoạt động trong lĩnh vực IT.

**1.6.2.Tính năng quản lý**

Về phương diện quản trị, Microsoft SQL Server gồm các dịch vụ tích hợp SQL Server, dịch vụ SQL Server Data Quality và dịch vụ SQL Server Master. Hai bộ công cụ dành riêng cho quản trị viên cơ sở dữ liệu và lập trình viên (SQL Server Data Tools) sử dụng trong việc phát triển hệ thống cơ sở dữ liệu cho SQL Server Management Studio, thực hiện nhiệm vụ triển khai, giám sát và quản lý các cơ sở dữ liệu.

**1.6.3.Vai trò của SQL server**

SQL Server không phải là một hệ quản trị cơ sở dữ liệu độc lập. Nó là thành phần với vai trò ngôn ngữ làm công cụ giao tiếp của cơ sở dữ liệu và người dùng.

SQL là một ngôn ngữ đòi hỏi sự tương tác cao. Người dùng có thể dễ dàng trao đổi với tiện ích bằng câu lệnh SQL đến cơ sở dữ liệu và nhận lại kết quả từ đó.

SQL là một ngôn ngữ lập trình cơ sở dữ liệu. Bằng cách nhúng các câu lệnh SQL trong ngôn ngữ lập trình, các lập trình viên có thể xây dựng được các chương trình ứng dụng giao tiếp với cơ sở dữ liệu

SQL là một ngôn ngữ lập trình quản trị cơ sở dữ liệu. Quản trị viên cơ sở dữ liệu có thể quản lý và điều khiển các truy cập tới cơ sở dữ liệu thông qua SQL

SQL được sử dụng như một công cụ giao tiếp với các trình ứng dụng trong hệ thống cơ sở dữ liệu khách – chủ.

## **1.7.Phân tích thiết kế hệ thống hướng đối tượng và UML**

**1.7.1.Khái niệm OOAD (Object Oriented Analysis and Design)**

Trong kỹ nghệ phần mềm để sản xuất được một sản phẩm phần mềm người ta chia quá trình phát triển sản phẩm ra nhiều giai đoạn như thu thập và phân tích yêu cầu, phân tích và thiết kế hệ thống, phát triển (coding), kiểm thử, triển khai và bảo trì. Trong đó, giai đoạn phân tích, thiết kế bao giờ cũng là giai đoạn khó khăn và phức tạp nhất. Giai đoạn này giúp chúng ta hiểu rõ yêu cầu đặt ra, xác định giải pháp, mô tả chi tiết giải pháp. Nó trả lời 2 câu hỏi What (phần mềm này làm cái gì?) và How (làm nó như thế nào?).

Để phân tích và thiết kế một phần mềm thì có nhiều cách làm, một trong những cách làm đó là xem hệ thống gồm những đối tượng sống trong đó và tương tác với nhau. Việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ hệ thống và cài đặt được nó. Phương thức này gọi là Phân tích thiết kế hướng đối tượng (OOAD)

**1.7.2.Khái niệm về UML (Unified Modeling Language)**

UML là ngôn ngữ mô hình hóa hợp nhất dùng để biểu diễn hệ thống. Nói một cách đơn giản là nó dùng để tạo ra các bản vẽ nhằm mô tả thiết kế hệ thống. Các bản vẽ này được sử dụng để các nhóm thiết kế trao đổi với nhau cũng như dùng để thi công hệ thống (phát triển), thuyết phục khách hàng, các nhà đầu tư...

**1.7.3.Tại sao lại là OOAD và UML?**

OOAD cần các bản vẽ để mô tả hệ thống được thiết kế, còn UML là ngôn ngữ mô tả các bản vẽ nên cần nội dung thể hiện. Do vậy, chúng ta phân tích và thiết kế theo hướng đối tượng và sử dụng UML để biểu diễn các thiết kế đó nên chúng thường đi đôi với nhau.

**1.7.4.OOAD sử dụng UML**

UML sử dụng để vẽ cho nhiều lĩnh vực khác nhau như phần mềm, cơ khí, xây dựng,…trong phạm vi các bài viết này chúng ta chỉ nghiên cứu cách sử dụng UML cho phân tích và thiết kế hướng đối tượng trong ngành phần mềm. OOAD sử dụng UML bao gồm các thành phần sau:

* ***View (góc nhìn)***

Mỗi góc nhìn như thầy bói xem voi, nó không thể hiện hết hệ thống nhưng thể hiện rõ hệ thống ở một khía cạnh. Chính vì thế trong xây dựng có bản vẽ kiến trúc (nhìn về mặt kiến trúc), bản vẽ kết cấu (nhìn về mặt kết cấu), bản vẽ thi công (nhìn về mặt thi công).



Hình 1.6. Các View trong OOAD sử dụng UML

Trong đó:

* + UseCase View: cung cấp góc nhìn về các ca sử dụng giúp chúng ta hiểu hệ thống có gì? ai dùng và dùng nó như thế nào.
  + Logical View: cung cấp góc nhìn về cấu trúc hệ thống, xem nó được tổ chức như thế nào. Bên trong nó có gì.
  + Process View: cung cấp góc nhìn động về hệ thống, xem các thành phần trong hệ thống tương tác với nhau như thế nào.
  + Component View: Cũng là một góc nhìn về cấu trúc giúp chúng ta hiểu cách phân bổ và sử dụng lại các thành phần trong hệ thống ra sao.
  + Deployment View: cung cấp góc nhìn về triển khai hệ thống, nó cũng ảnh hưởng lớn đến kiến trúc hệ thống.

Tập hợp các góc nhìn này sẽ giúp chúng ta hiểu rõ hệ thống cần phân tích, thiết kế. Trong hình trên chúng ta thấy góc nhìn Use Case View nằm ở giữa và chi phối tất cả các góc nhìn còn lại. Chính vì thế chúng ta thường thấy các tài liệu nói về 4 view + 1 chứ không phải 5 view nhằm nhấn mạnh vai trò của UseCase View.

* ***Diagram (Bản vẽ)***

Diagram các bạn có thể dịch là sơ đồ. Tuy nhiên ở đây chúng ta sử dụng từ bản vẽ cho dễ hình dung. Các bản vẽ được dùng để thể hiện các góc nhìn của hệ thống.



Hình 1.7. Các bản vẽ trong OOAD sử dụng UML

Trong đó:

* Use Case Diagram: bản vẽ mô tả về ca sử dụng của hệ thống. Bản vẽ này sẽ giúp chúng ta biết được ai sử dụng hệ thống, hệ thống có những chức năng gì. Lập được bản vẽ này bạn sẽ hiểu được yêu cầu của hệ thống cần xây dựng.
* Class Diagram: bản vẽ này mô tả cấu trúc của hệ thống, tức hệ thống được cấu tạo từ những thành phần nào. Nó mô tả khía cạnh tĩnh của hệ thống.
* Object Diagram: Tương tự như Class Diagram nhưng nó mô tả đến đối tượng thay vì lớp (Class).
* Sequence Diagarm: là bản vẽ mô tả sự tương tác của các đối tượng trong hệ thống với nhau được mô tả tuần tự các bước tương tác theo thời gian.
* Collaboration Diagram: tương tự như sequence Diagram nhưng nhấn mạnh về sự tương tác thay vì tuần tự theo thời gian.
* State Diagram: bản vẽ mô tả sự thay đổi trạng thái của một đối tượng. Nó được dùng để theo dõi các đối tượng có trạng thái thay đổi nhiều trong hệ thống.
* Activity Diagram: bản vẽ mô tả các hoạt động của đối tượng, thường được sử dụng để hiểu về nghiệp vụ của hệ thống.
* Component Diagram: bản vẽ mô tả về việc bố trí các thành phần của hệ thống cũng như việc sử dụng các thành phần đó.
* Deployment Diagram: bản vẽ mô tả việc triển khai của hệ thống như việc kết nối, cài đặt, hiệu năng của hệ thống v.v…
* ***Notations (các ký hiệu)***

Notations là các ký hiệu để vẽ, nó như từ vựng trong ngôn ngữ tự nhiên. Bạn phải biết từ vựng thì mới ghép thành câu, thành bài được. Chúng ta sẽ tìm hiểu kỹ các notations trong từng bản vẽ sau này.



Hình 1.8. Ký hiệu về Use Case



Hình 1.9. Ký hiệu về Class



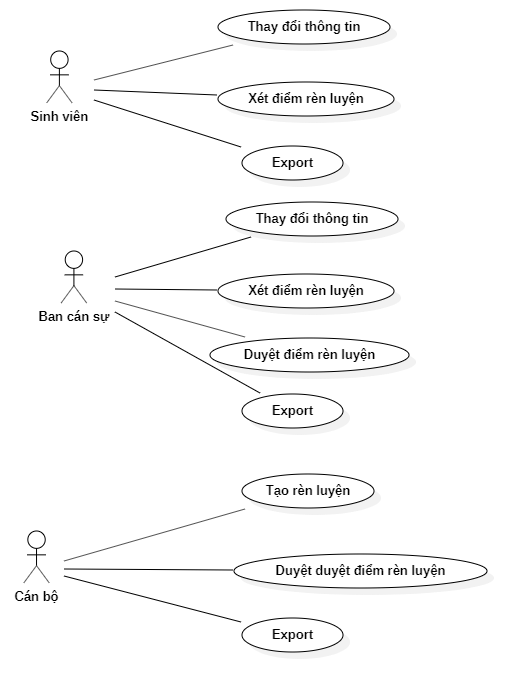
Hình 1.10. Ký hiệu về Actor

**CHƯƠNG II PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

## **2.1. Sơ đồ Usecase**

### **2.1.1. Sơ đồ Usecase tổng quát**

Khái quát chức năng chính của hệ thống. Các chức năng này có tính tổng quát dễ dàng nhìn thấy được trên quan điểm của các tác nhân. Dựa vào yêu cầu của bài toán ta có use case tổng quát như sau:



Hình 2.1. Sơ đồ tổng quát

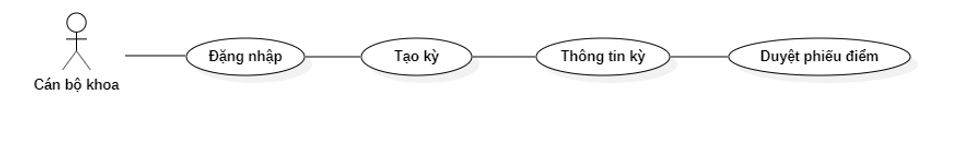
**2.1.2. Sơ đồ Usecase thành phần**

***a. UseCase cán bộ khoa***

**Tác nhân:** Người quản trị

**Điều kiện:** Phải đăng nhập vào hệ thống

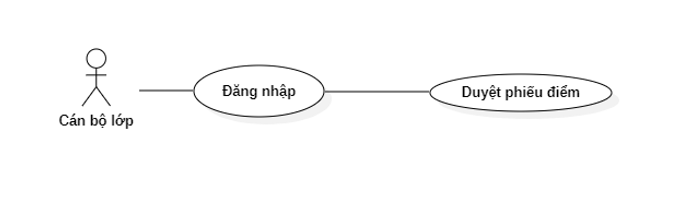
**Mô tả:** Người quản trị sau khi đăng nhập được, chọn vào chức năng tạo điểm rèn luyện, người quản trị duyệt các đánh giá sau khi được thông báo



Hình 2.2 Sơ đồ usecase cán bộ khoa

***b. Usecase cán bộ lớp***

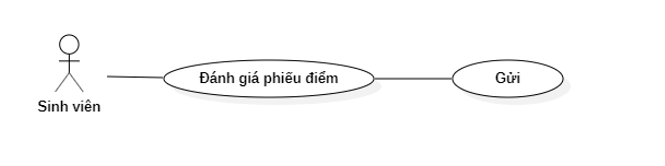
Ban cán sự lớp (lớp trưởng) nhận thông báo từ hệ thống sau đó duyệt các đánh giá của sinh viên trong lớp và gửi thông báo duyệt xong cho hệ thống. Hệ thống sẽ trả về thông báo cho người quản trị, thông tin đó sẽ được người quản trị duyệt một lần nữa và xác nhận hoàn thành.



Hình 2.3. Sơ đồ usecase các bộ lớp

***c. Usecase sinh viên***

Sau khi cán bộ khoa tạo kỳ đánh giá thì sinh viên nhận được thông báo đánh giá phiếu rèn luyện của mình và gửi lên cán bộ lớp, cán bộ khoa duyệt.



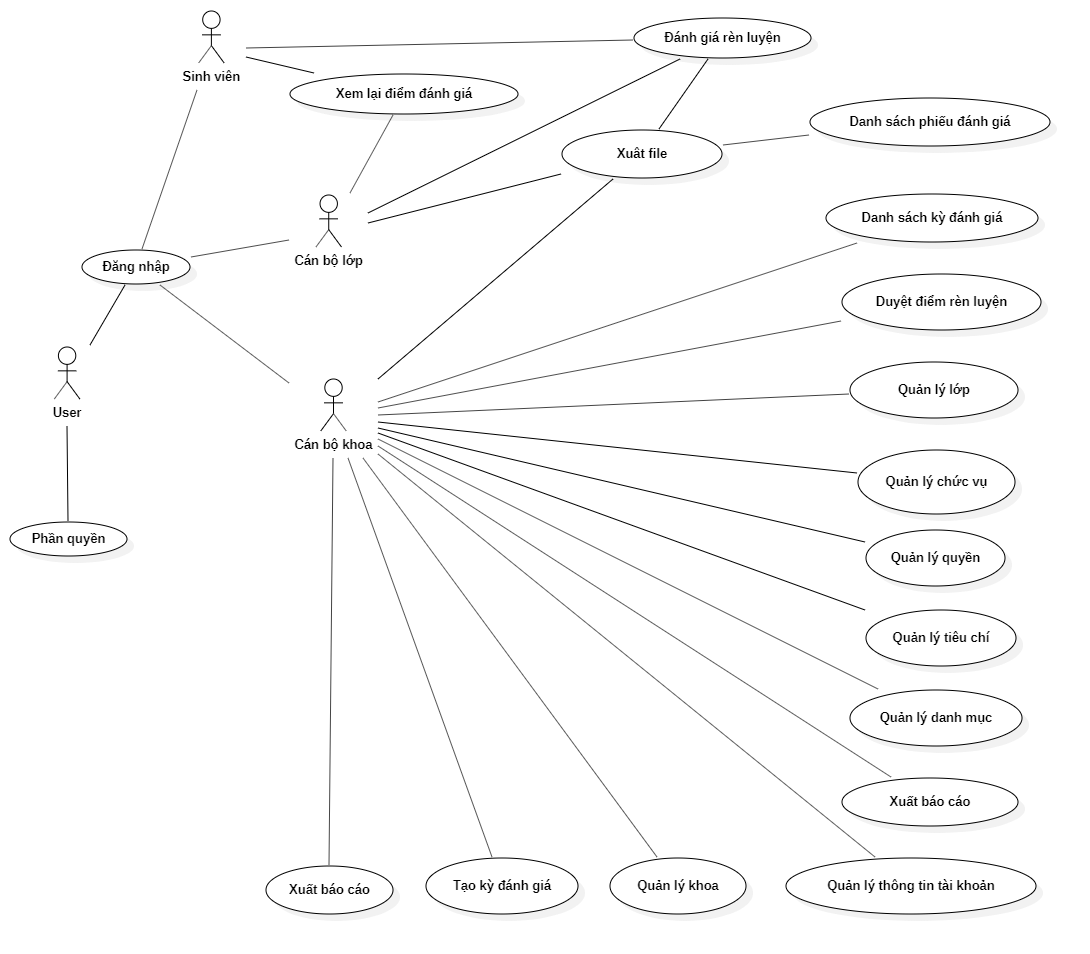
Hình 2.4. Sơ đồ usecase sinh viên

### **2.1.3.Sơ đồ Usecase chi tiết**

Cán bộ yêu cầu tạo mới điểm rèn luyện, hệ thống gửi thông báo cho từng lớp để cập nhật đánh giá điểm số rèn luyện sau đó chờ kết quả và duyệt.

Lớp trưởng chờ thông báo từ hệ thống gửi xuống để duyệt thông tin đánh giá từ lớp.

Sinh viên vào hệ thống đăng nhập bằng tài khoản của mình để đánh giá điểm rèn luyện của mình, sau khi được đánh giá và được lớp trưởng và cán bộ duyệt thì sinh viên có thể xem chi tiết về điểm rèn luyện của mình.



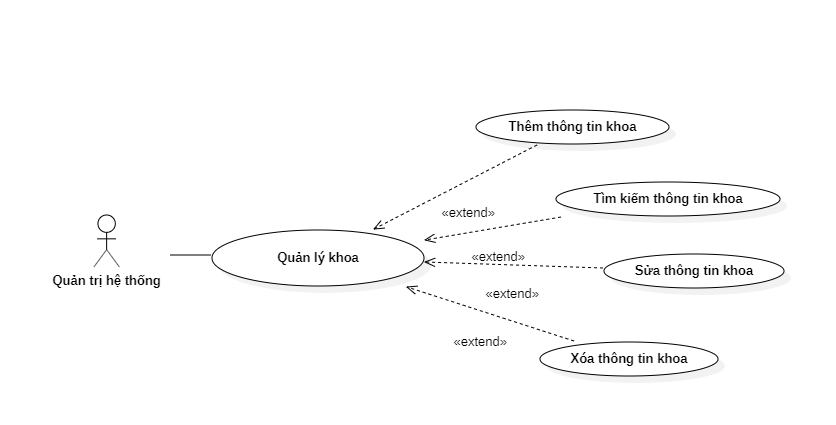
Hình 2.5. Sơ đồ chi tiết

### **2.1.4 Sơ đồ UseCase quản lý khoa**

**Tác nhân:** Người quản trị

**Điều kiện:** Phải đăng nhập vào hệ thống

**Mô tả:** Người quản trị sau khi đăng nhập vào hệ thống, chọn vào chức năng quản trị hệ thống, người quản trị có thể thực hiện việc thêm, sửa, tìm kiếm hoặc xóa thông tin khoa.



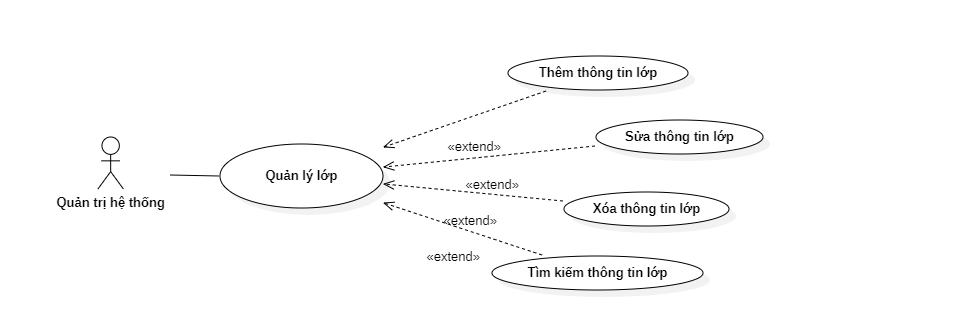
Hình 2.6. Sơ đồ Usecase quản lý khoa

### **2.1.5 Sơ đồ UseCase quản lý lớp**

**Tác nhân:** Người quản trị

**Điều kiện:** Phải đăng nhập vào hệ thống

**Mô tả:** Người quản trị sau khi đăng nhập vào hệ thống, chọn vào chức năng quản trị hệ thống quản lý lớp, người quản trị có thể thực hiện việc thêm, sửa, tìm kiếm hoặc xóa thông tin lớp.



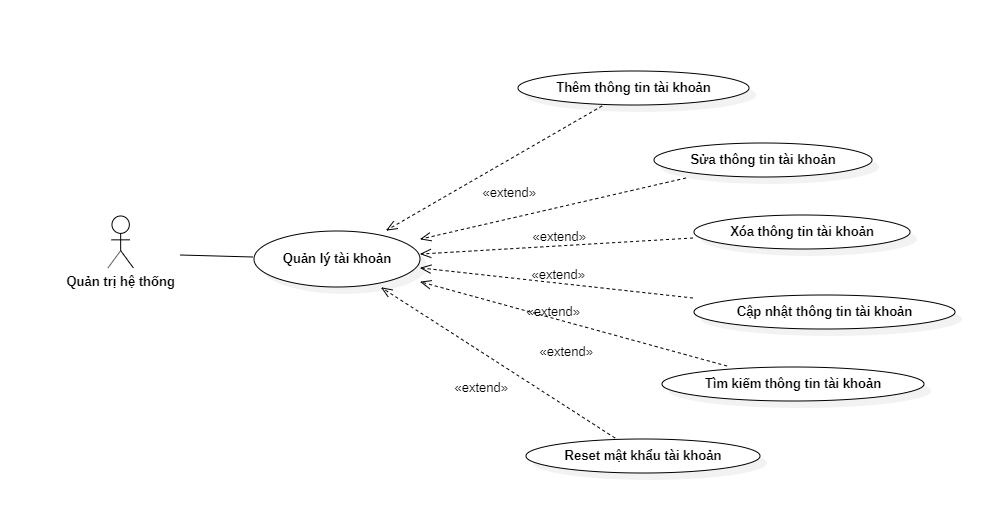
Hình 2.7. Sơ đồ Usecase quản lý lớp

### **2.1.6. Sơ đồ UseCase quản lý tài khoản**

**Tác nhân:** Người quản trị

**Điều kiện:** Phải đăng nhập vào hệ thống

**Mô tả:** Người quản trị sau khi đăng nhập vào hệ thống, chọn vào chức năng quản trị hệ thống quản lý tài khoản, người quản trị có thể thực hiện việc thêm, sửa, tìm kiếm hoặc xóa thông tin tài khoản. Ngoài ra hệ thống còn có thể cập nhật quyền của tài khoản: quyền sinh viên, quyền cán bộ lớp và cán bộ khoa. Cuối cùng là chức năng reset mật khẩu tài khoản “123456”.

******

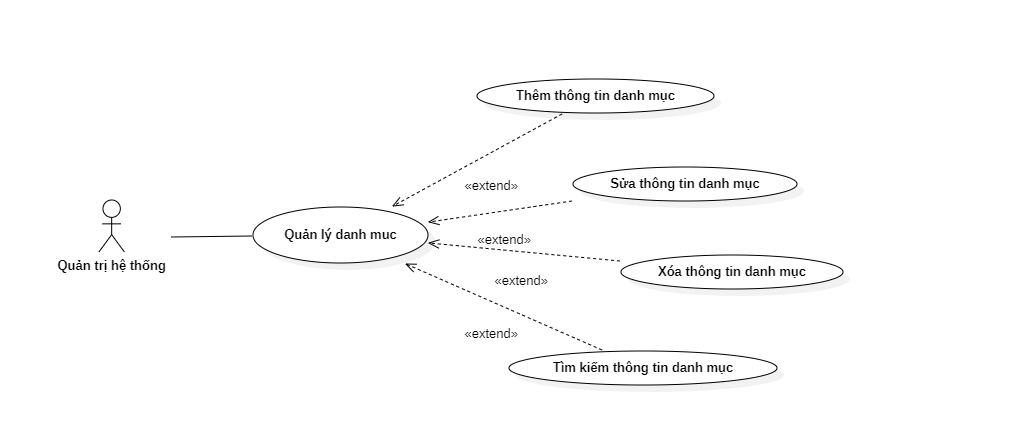
Hình 2.8. Sơ đồ Usecase quản lý danh mục

### **2.1.7.Sơ đồ UseCase quản lý danh mục (Menu)**

**Tác nhân:** Người quản trị

**Điều kiện:** Phải đăng nhập vào hệ thống

**Mô tả:** Người quản trị sau khi đăng nhập vào hệ thống, chọn vào chức năng quản trị hệ thống quản lý danh mục, người quản trị có thể thực hiện việc thêm, sửa, tìm kiếm hoặc xóa thông tin danh mục.

******

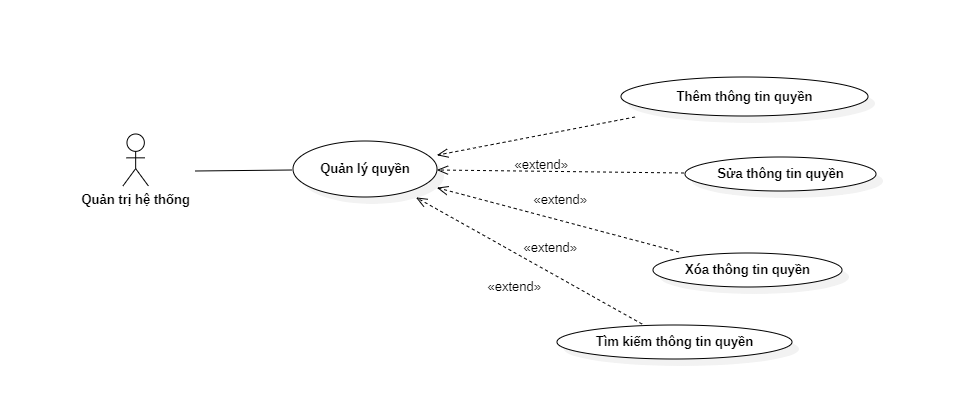
Hình 2.9. Sơ đồ Usecase quản lý danh mục

### **2.1.8. Sơ đồ UseCase quản lý quyền**

**Tác nhân:** Người quản trị

**Điều kiện:** Phải đăng nhập vào hệ thống

**Mô tả:** Người quản trị sau khi đăng nhập vào hệ thống, chọn vào chức năng quản trị hệ thống quản lý phân quyền , người quản trị có thể thực hiện việc thêm, sửa, tìm kiếm hoặc xóa thông tin quyền.

******

Hình 2.10. Sơ đồ Usecase quản lý danh mục

**2.2. Cơ sở dữ liệu**

**Bảng Tài khoản:** dùng để lưu trữ dữ liệu *tài khoản sinh viên*.

* Mô tả thuộc tính

Bảng 2.1 Tài khoản

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | TaiKhoanID | int | Khóa chính |
| 2 | TenTaiKhoan | nvarchar(250) |  |
| 3 | MatKhau | nvarchar(500) |  |
| 4 | isDelete | bit |  |
| 5 | NgayTao | date |  |
| 6 | HoTen | nvarchar(500) |  |
| 7 | SoDienThoai | nvarchar(50) |  |
| 8 | Email | nvarchar(500) |  |
| 9 | DiaChi | ntext |  |
| 10 | HinhAnh | nvarchar(500) |  |
| 11 | MaTaiKhoan | NgaySinh |  |
| 12 | ChucVuID | int |  |
| 13 | MaLop | int |  |
| 14 | MaKhoa | int |  |
| 15 | NgaySinh | date |  |

**Bảng Tài khoản quyền:** dùng để lưu trữ dữ liệu *Tài khoản quyền*.

* Mô tả thuộc tính

Bảng 2.2 Tài khoản quyền

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID | int |  |
| 2 | TaiKhoanID | int |  |
| 3 | MaQuyen | int |  |
| 4 | isDelete | bit |  |

**Bảng Tài khoản thông báo:** dùng để lưu trữ dữ liệu *Tài khoản thông báo*.

* Mô tả thuộc tính

Bảng 2.3 Tài khoản thông báo

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID | int | Khóa chính |
| 2 | TaiKhoanID | int |  |
| 3 | ThongBaoID | int |  |
| 4 | isDelete | bit |  |
| 5 | TrangThai | int |  |

**Bảng Thông báo:** dùng để lưu trữ dữ liệu *Thông báo*.

* Mô tả thuộc tính

Bảng 2.4 Thông báo

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ThongBaoID | int | Khóa chính |
| 2 | NgayThongBao | date |  |
| 3 | NoiDungTB | ntext |  |
| 4 | TieuDe | nvarchar(500) |  |
| 5 | KyDKID | int |  |
| 6 | IsDelete | bit |  |

**Bảng Tiêu chí:** dùng để lưu trữ dữ liệu *Tiêu chí*.

* Mô tả thuộc tính

Bảng 2.5 Tiêu chí

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | TieuChiID | int | Khóa chính |
| 2 | TenTieuChi | date |  |
| 3 | DiemTieuChi | ntext |  |
| 4 | OrderKey | nvarchar(500) |  |
| 5 | isDelete | int |  |

**Bảng Chức vụ:** dùng để lưu trữ dữ liệu *Chức vụ*.

* Mô tả thuộc tính

Bảng 2.6 Chức vụ

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ChucVuID | int | Khóa chính |
| 2 | TenChucVu | nvarchar(500) |  |
| 3 | isDelete | bit |  |
| 4 | Loai | bit |  |

**Bảng Điểm:** dùng để lưu trữ dữ liệu *Điểm*.

* Mô tả thuộc tính

Bảng 2.7 Điểm

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | DiemID | int | Khóa chính |
| 2 | PhieuID | int |  |
| 3 | DiemSV | int |  |
| 4 | DiemCBL | int |  |
| 5 | DiemCBK | int |  |
| 6 | TieuChiID | int |  |
| 7 | isDelete | bit |  |

**Bảng Khoa:** dùng để lưu trữ dữ liệu *Khoa*.

* Mô tả thuộc tính

Bảng 2.8 Khoa

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | MaKhoa | int | Khóa chính |
| 2 | TenKhoa | nvarchar(500) |  |
| 3 | isDelete | bit |  |

**Bảng Ký đăng ký:** dùng để lưu trữ dữ liệu *Ký đăng ký*.

* Mô tả thuộc tính

Bảng 2.9 Ký đăng ký

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | KyDKID | int | Khóa chính |
| 2 | HocKyXet | nvarchar(500) |  |
| 3 | NgayBD | date |  |
| 4 | NgayKT | date |  |
| 5 | NamXet | nvarchar(500) |  |
| 6 | TieuDeDK | nvarchar(500) |  |
| 7 | NoiDungDK | ntext |  |
| 8 | IsDelete | bit |  |
| 9 | TrangThai | int |  |

**Bảng Lớp:** dùng để lưu trữ dữ liệu *Lớp*.

* Mô tả thuộc tính

Bảng 2.10 Lớp

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | MaLop | int | Khóa chính |
| 2 | TenLop | nvarchar(500) |  |
| 3 | MaKhoa | date |  |
| 4 | isDelete | date |  |

**Bảng Lớp đăng ký:** dùng để lưu trữ dữ liệu *Lớp đăng ký*.

* Mô tả thuộc tính

Bảng 2.11 Lớp đăng ký

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID | int | Khóa chính |
| 2 | MaLop | int |  |
| 3 | KyDKID | int |  |
| 4 | MaKhoa | int |  |
| 5 | IsDelete | bit |  |

**Bảng Menu:** dùng để lưu trữ dữ liệu *Menu*.

* Mô tả thuộc tính

Bảng 2.12 Menu

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | MenuID | int | Khóa chính |
| 2 | TenMenu | nvarchar(500) |  |
| 3 | ParentID | int |  |
| 4 | OrderKey | int |  |
| 5 | isDelete | bit |  |
| 6 | Path | nvarchar(500) |  |

**Bảng Phiếu đánh giá:** dùng để lưu trữ dữ liệu *Phiếu đánh giá*.

* Mô tả thuộc tính

Bảng 2.13 Phiếu đánh giá

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | PhieuID | int | Khóa chính |
| 2 | SinhVienID | int |  |
| 3 | CanBoLopID | int |  |
| 4 | CanBoID | int |  |
| 5 | NgayDanhGia | date |  |
| 6 | KyDKID | int |  |
| 7 | TenPhieu | nvarchar(500) |  |
| 8 | isDelete | bit |  |
| 9 | DiemTong | int |  |
| 10 | DiemTongSV | int |  |
| 11 | DiemTongCBL |  |  |

**Bảng Quyền:** dùng để lưu trữ dữ liệu *Quyền.*

* Mô tả thuộc tính

Bảng 2.14 Quyền

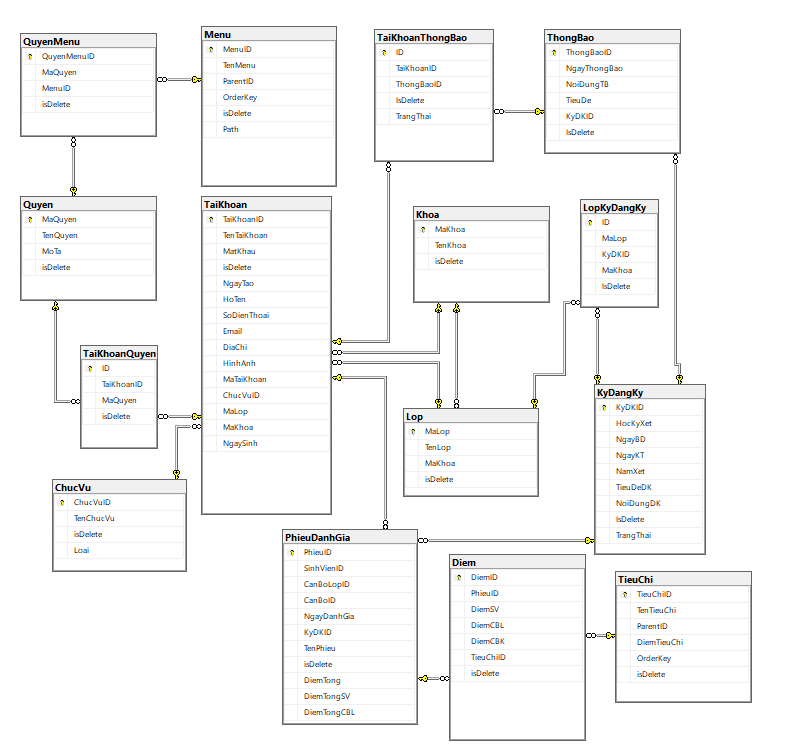
|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | MaQuyen | int | Khóa chính |
| 2 | TenQuyen | nvarchar(500) |  |
| 3 | MoTa | nvarchar(500) |  |
| 4 | isDelete | bit |  |

**Bảng Quyền menu:** dùng để lưu trữ dữ liệu *Quyền Menu.*

* Mô tả thuộc tính

Bảng 2.15 Quyền menu

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | QuyenMenuID | int | Khóa chính |
| 2 | MaQuyen | int |  |
| 3 | MenuID | int |  |
| 4 | isDelete | bit |  |

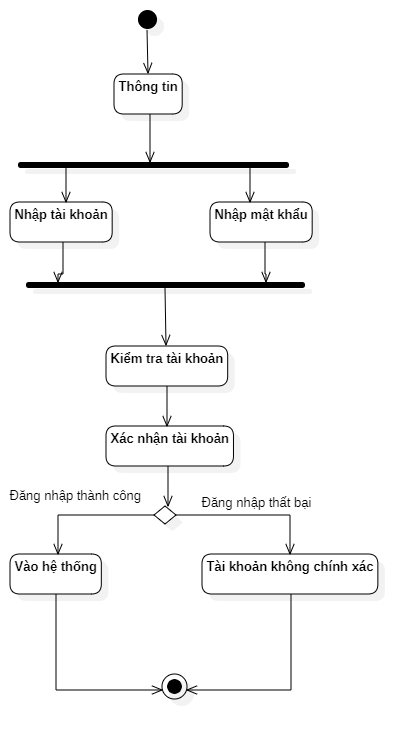


Hình 2.11 Sơ đồ diagram

## **2.3. Sơ đồ hoạt động**

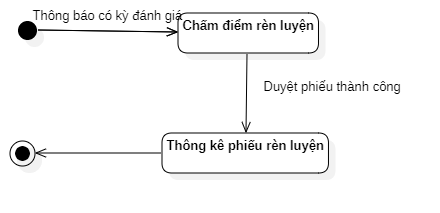
**2.3.1. Sơ đồ đăng nhập**

Người quản trị nhập tài khoản và nhập mật khẩu hệ thống kiểm tra thông tin và xác nhận, nếu tài khoản đúng thông báo đăng nhập thành công và kết thúc. Ngược lại, thông báo thông báo tài khoản và mật khẩu không đúng và kết thúc. Nếu chưa có tài khoản thông báo tài khoản không tồn tại. Tài khoản có quyền khác nhau: quyền sinh viên đánh giá rèn luyện khi cán bộ khoa mở cho lớp và xem thông danh sách phiếu đánh giá từng học kỳ, quyền cán bộ lớp xét điểm rèn luyện của sinh viên đã đánh giá và tự đánh giá phiếu rèn luyện của mình, quyền cán bộ khoa tạo đánh kỳ đánh giá, xem lại kỳ đánh giá, duyệt điểm rèn luyện của sinh viên khi cán bộ lớp đã duyệt.



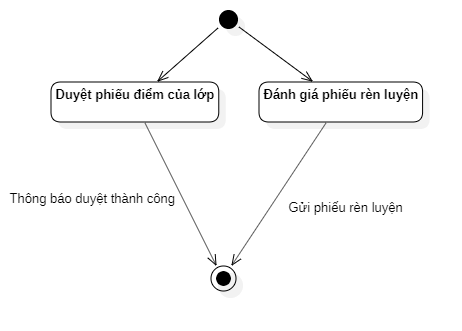
Hình 2.12. Sơ đồ hoạt động đăng nhập

**2.3.2. Sơ đồ hoạt động sinh viên**

****

Hình 2.13. Sơ đồ hoạt động sinh viên

**2.3.3. Sơ đồ hoạt động cán bộ lớp**

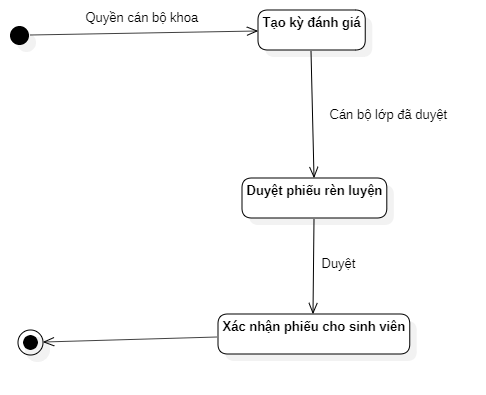
****

Hình 2.14 Sơ đồ hoạt động cán bộ lớp

**2.3.4. Sơ đồ hoạt động các bộ khoa**



Hình 2.15 Sơ đồ hoạt tạo kỳ đánh giá cán bộ khoa

****

Hình 2.16 Sơ đồ hoạt động cán bộ khoa

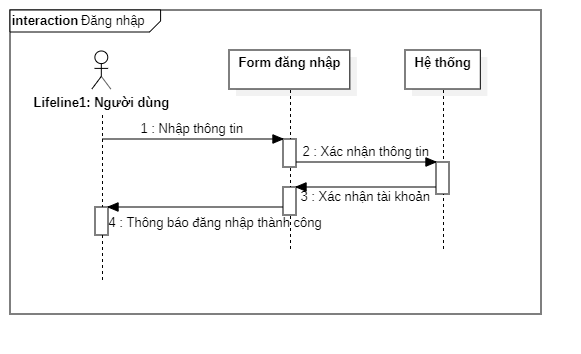
**2.4. Sơ đồ tuần tự**

**2.4.1 Sơ đồ tuần tự đăng nhập:**

**Tác nhân:** Người dùng

**Điều kiện:** Phải đăng nhập form đăng nhập

**Mô tả:** Người quản trị yêu cầu đăng nhập tại form đăng nhập, sau đó kiểm tra điều kiện đăng nhập. Hệ thống sẽ xác nhận thông tin và thông báo thành công cho người dùng

****

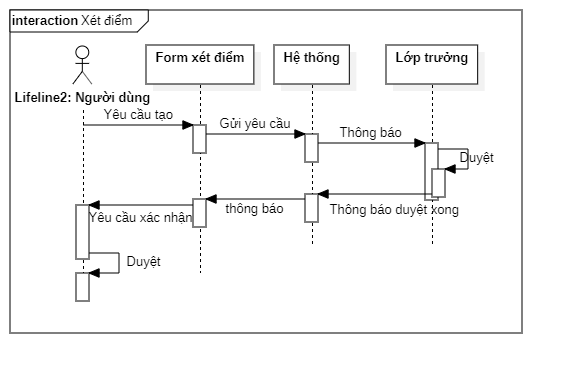
Hình 2.17. Sơ đồ tuần tự đăng nhập

**2.4.2.Xét điểm cán bộ lớp**

**Tác nhân:** Người dùng

**Điều kiện:** Phải đăng nhập form đăng nhập

**Mô tả:** Sinh viên được yêu cầu xét điểm từ hệ thống sau khi cán bộ khoa tạo rèn luyện, sau khi sinh viên duyệt xong, lớp trưởng sẽ kiểm tra trước khi đưa lên cán bộ khoa.

**

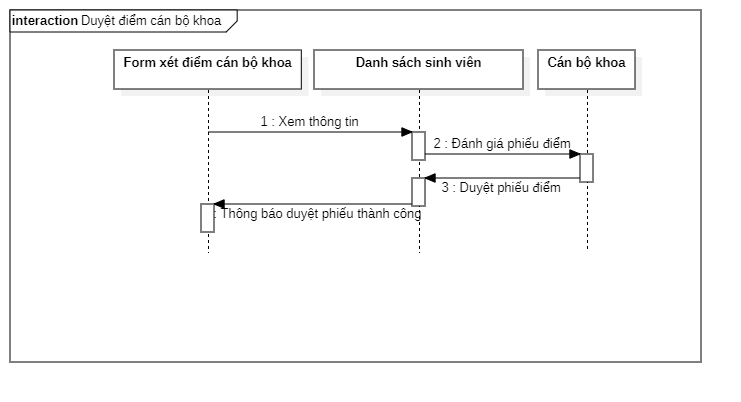
Hình 2.18 Sơ đồ tuần tự xét điểm

### **2.4.3. Xét điểm các bộ khoa**

**Tác nhân:** Người dùng

**Điều kiện:** Phải đăng nhập form đăng nhập

**Mô tả:** Xét điểm từ hệ thống sau khi cán bộ khoa xét điểm rèn luyện, sau khi cán bộ lớp gửi.

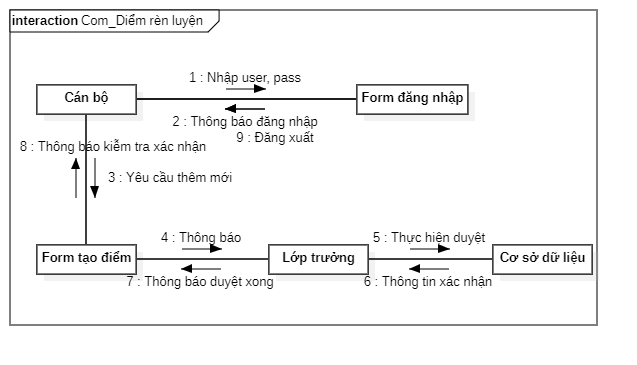


Hình 2.19 Sơ đồ tuần tự xét điểm

## **2.5. Sơ đồ hợp tác**

#### **2.5.1. Điểm rèn luyện cán bộ lớp**

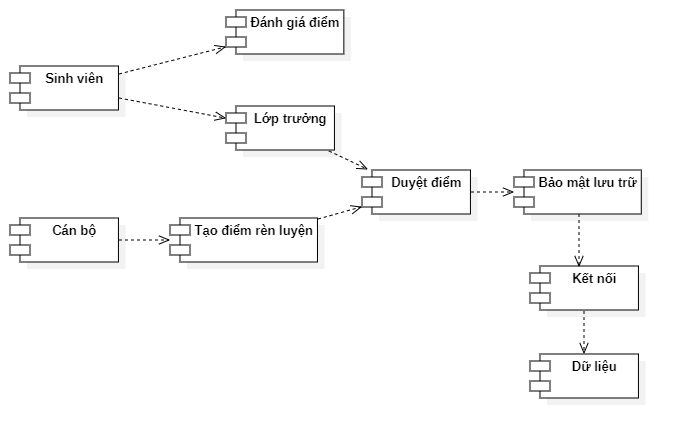
**Mô tả:** Cán bộ lớp đặng nhập vào giao diện chấm điểm sinh viên

****

Hình 2.20 Sơ đồ hợp tác xét điểm cán bộ lớp

**2.6. Sơ đồ thành phần**

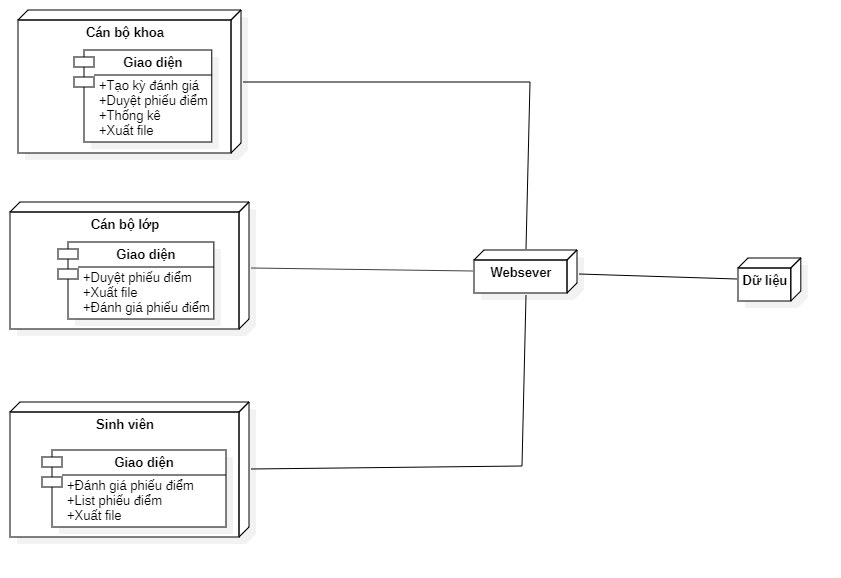
**Mô tả:** Sinh viên đánh giá điểm rèn luyện, lớp trưởng duyệt điểm rèn luyện, cán bộ khoa tạo kỳ xét điểm rèn luyện và duyệt điểm. Dữ liệu thành phần sẽ được lưu trữ vào cơ sở dữ liệu.

****

Hình 2.21. Sơ đồ thành phần

**2.7. Sơ đồ triển khai**

**Mô tả:** Giao diện triển khai giao diện cán bộ khoa tạo kỳ, duyệt phiếu điểm, xuất file danh sách kỳ xét điểm, thống kê số lượng sinh viên toàn khoa, số phiếu chưa duyệt và đã duyệt.

****

Hình 2.22 Sơ đồ triển khai

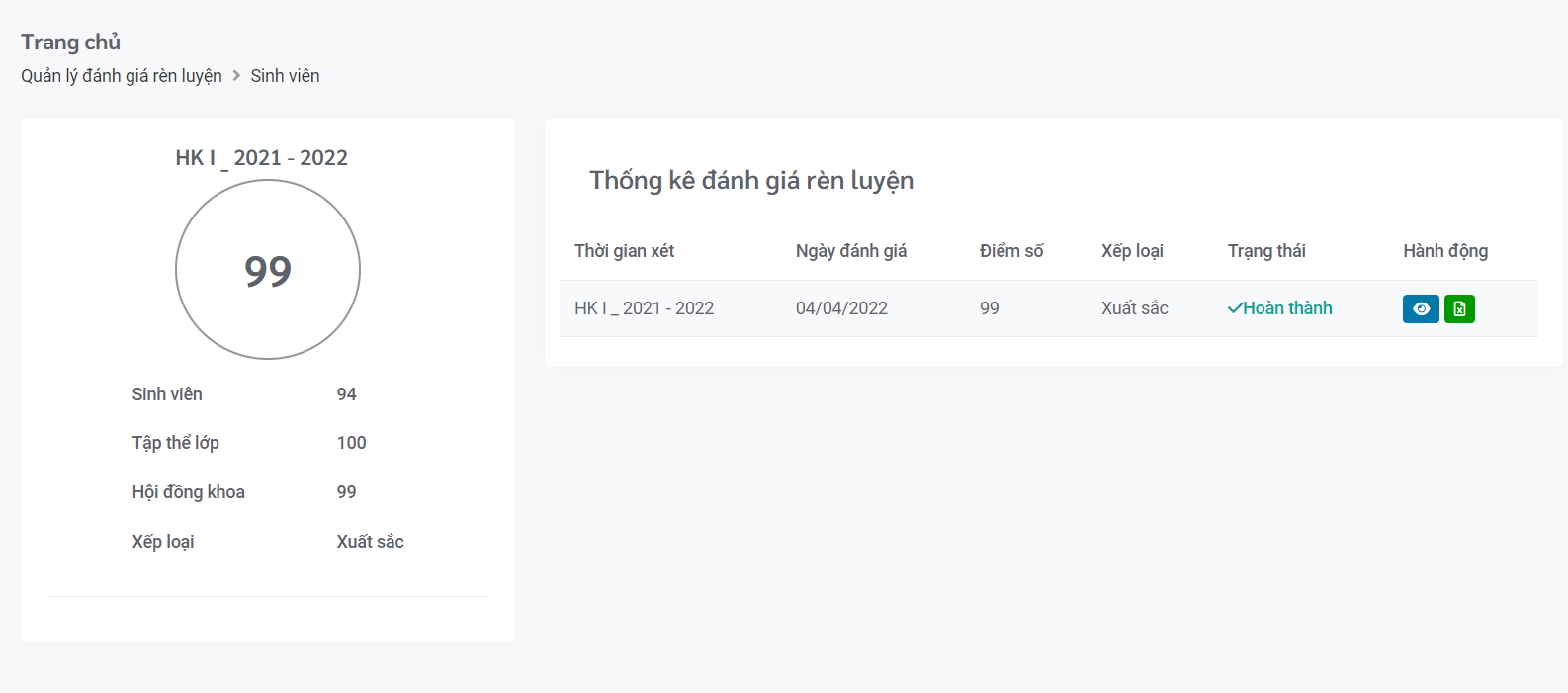
**2.8. Kết luận**

Thông qua các phần phân tích và thiết kế trên cho ta cái nhìn tổng quan về cách lưu trữ dữ liệu, hiểu rõ các chức năng và tiến trình xử lý cho từng chức năng cụ thể, những nhóm người dùng khác nhau sẽ sử dụng các chức năng riêng biệt cho từng nhóm, có cái nhìn trực quan về giao diện của của ứng dụng. Bên cạnh đó, kết quả của việc phân tích và thiết kế giúp chúng ta có thể lựa chọn bất kỳ môi trường và công cụ nào để phát triển hệ thống. Vấn đề cài đặt ứng dụng được thực hiện dễ dàng, một cách có hệ thống. Hạn chế dẫn đến sai sót trong quá trình xây dựng và cài đặt ứng dụng.

**CHƯƠNG III THIẾT KẾ VÀ CÀI ĐẶT WEBSITE**

**3.1. Giao diện thông tin rèn luyện sinh viên**

Sau khi tài khoản sinh viên đăng nhập thành công sẽ hiển thị những thông tin cần thiết cho người sử dụng Giao diện điểm sinh viên, sinh viên có thể xem điểm qua từng học kỳ của từng năm học và kết xuất dữ liệu.

**

Hình 3.1 Giao diện sinh viên

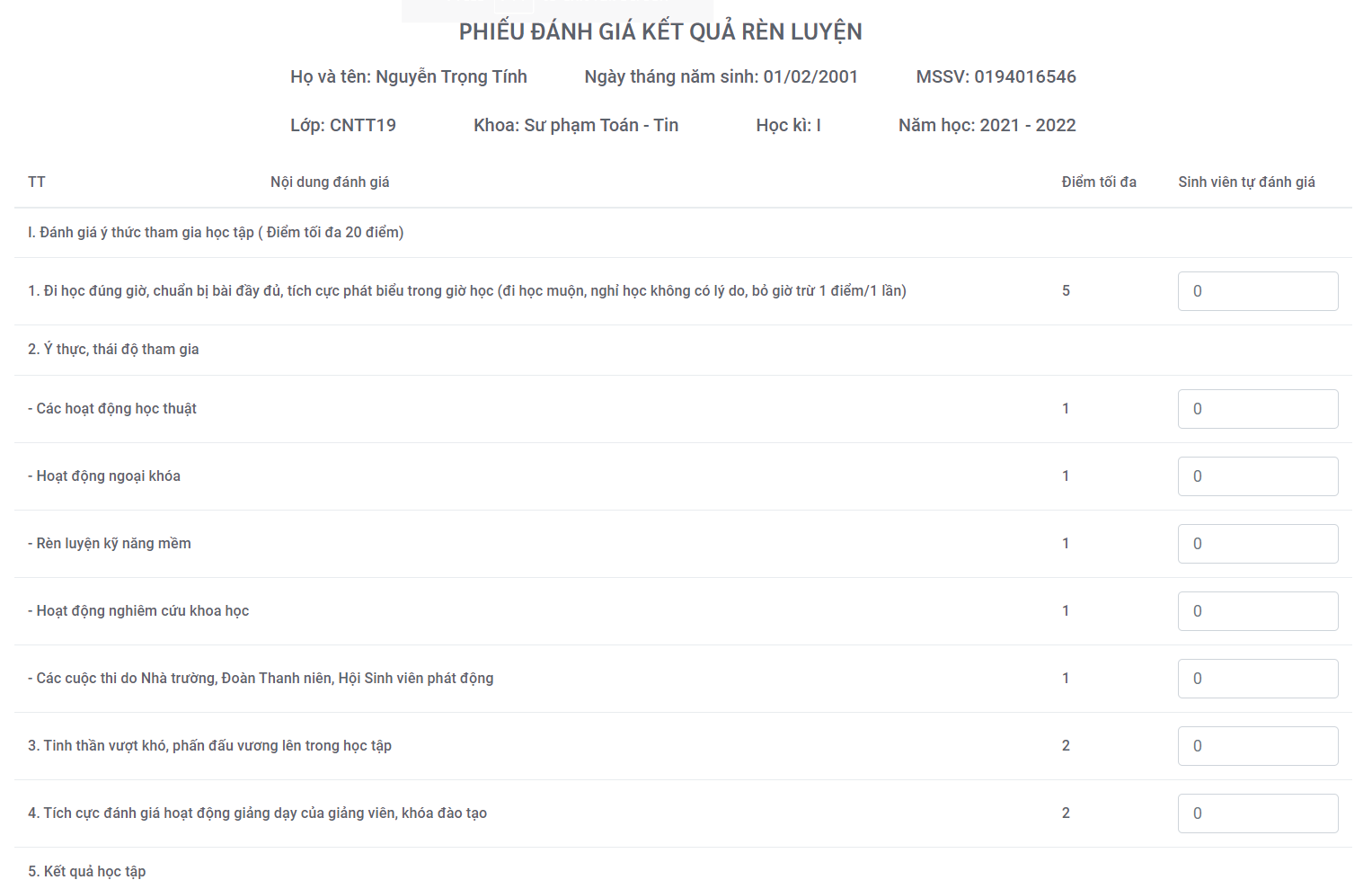
* 1. **Giao diện xét điểm rèn luyện của sinh viên**

Sau khi sinh viên đăng nhập vào hệ thống nếu cán bộ khoa đã tạo kỳ đánh giá cho lớp của tài khoản sinh viên thì sẽ có thông báo đăng ký xét điểm rèn luyện và khởi tạo phiếu đánh giá điểm rèn luyện, đánh giá phiếu rèn luyện.

****

Hình 3.2 Giao diện thông báo tạo phiếu đánh giá

Sau khi sinh viên đã tạo phiếu xét điểm rèn luyện hệ thống sẽ tạo ra một phiếu đánh giá kết quả rèn luyện cho sinh viên tự đánh giá  và gửi lên cán bộ lớp duyệt.

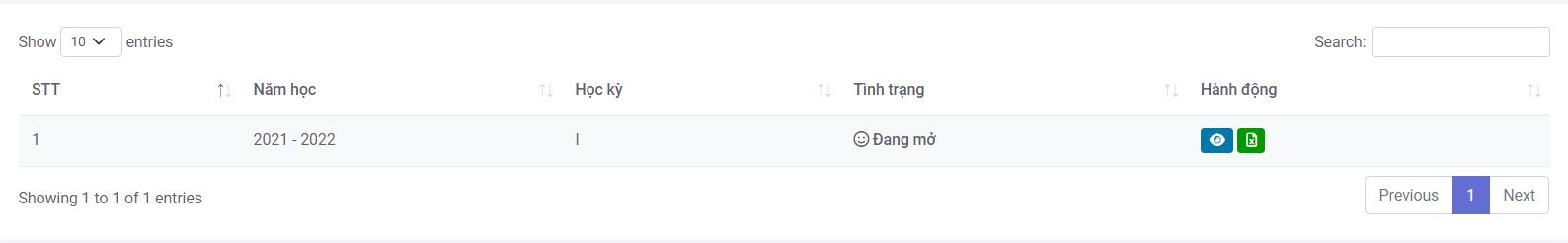
****

Hình 3.3 Giao diện đánh giá phiếu đánh giá

* 1. **Giao diện các bộ lớp xét điểm rèn luyện cho các bạn sinh viên của lớp**

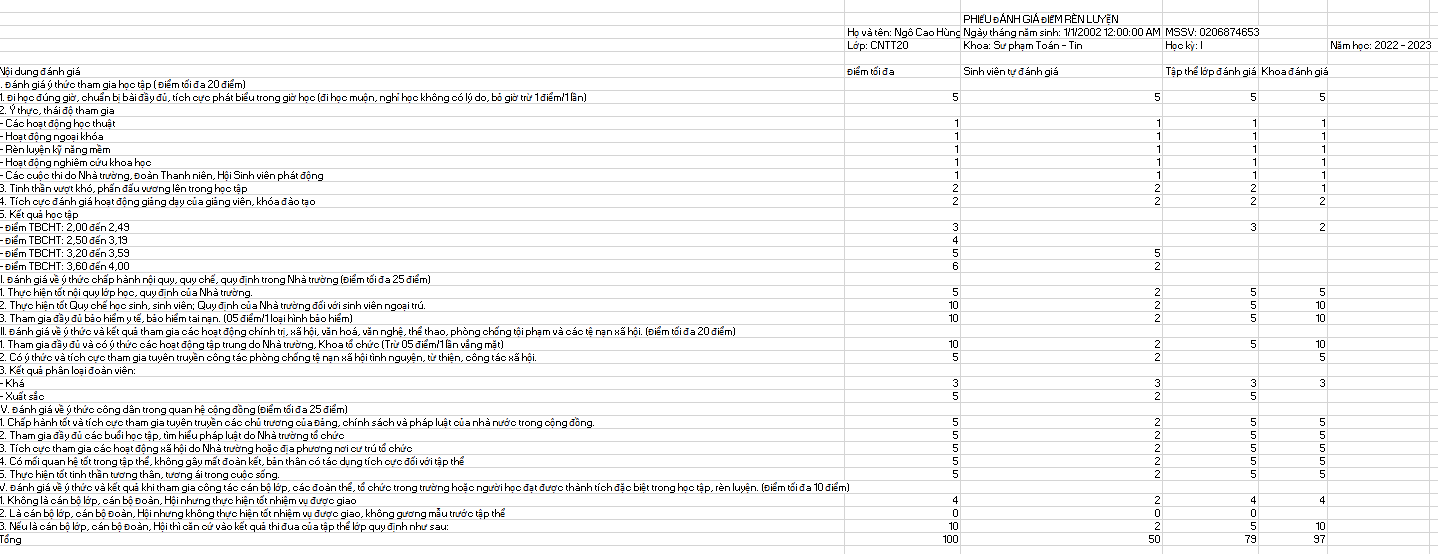
Giao diện các bộ lớp sẽ có chức năng tương tự như của sinh viên và sẽ có một nghiệp vụ quản lý điểm rèn luyện của lớp mình.

Cán bộ lớp xuất báo cáo  thống kê danh sách lớp mình thông kê các thông tin của sinh viên như: Tên sinh viên, mã sinh viên, điểm và xếp loại rèn luyện.



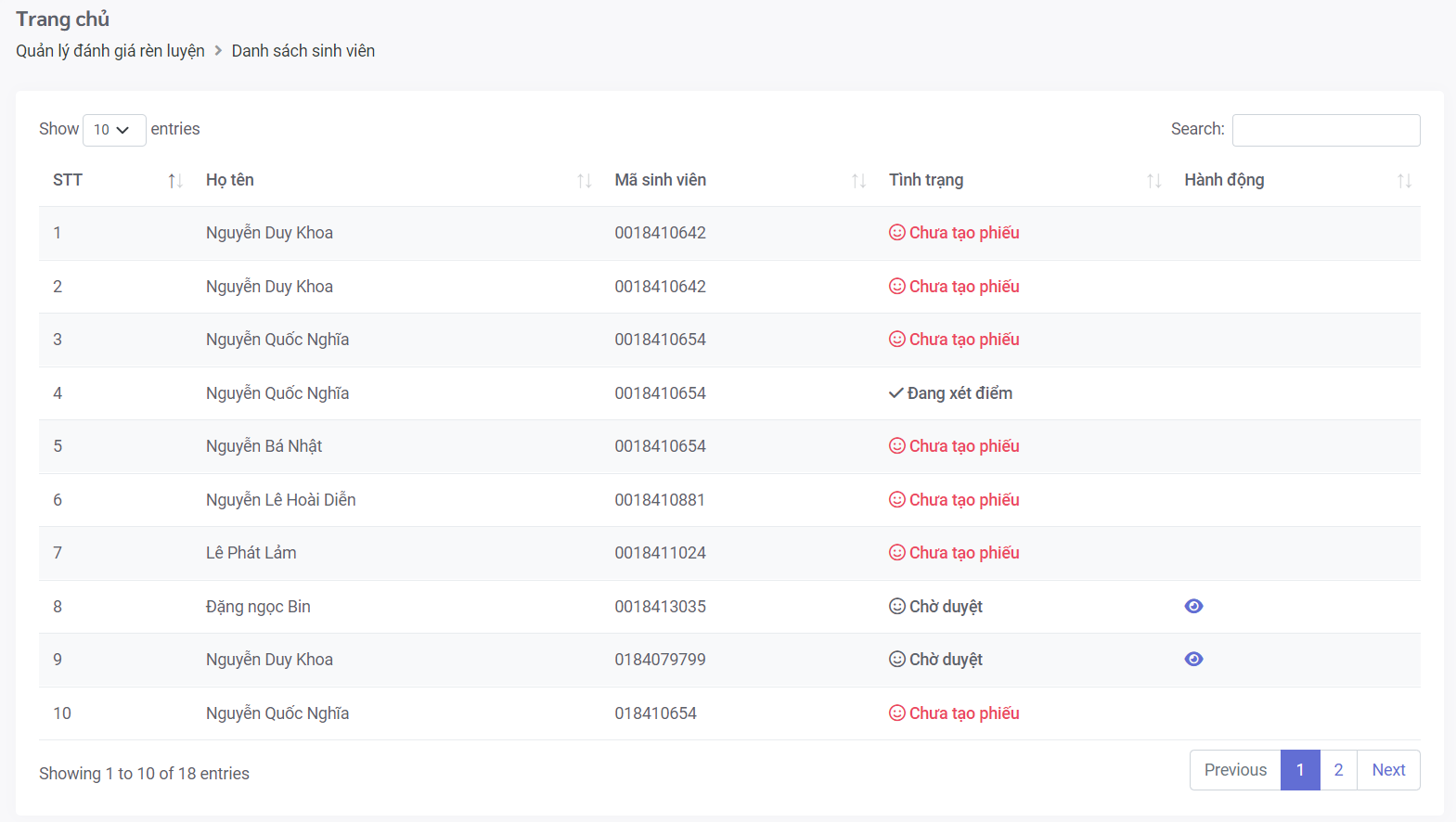
Hình 3.4 Giao diện xem năm học kỳ đánh giá mở hoặc đóng

Sinh viên có thể xuất file  thông tin phiếu chấm điểm, cựu thể là xuất phiếu phiếu đánh giá khi sinh viên vừa đánh giá xong hoặc cán bộ lớp, cán bộ khoa duyệt phiếu.

**

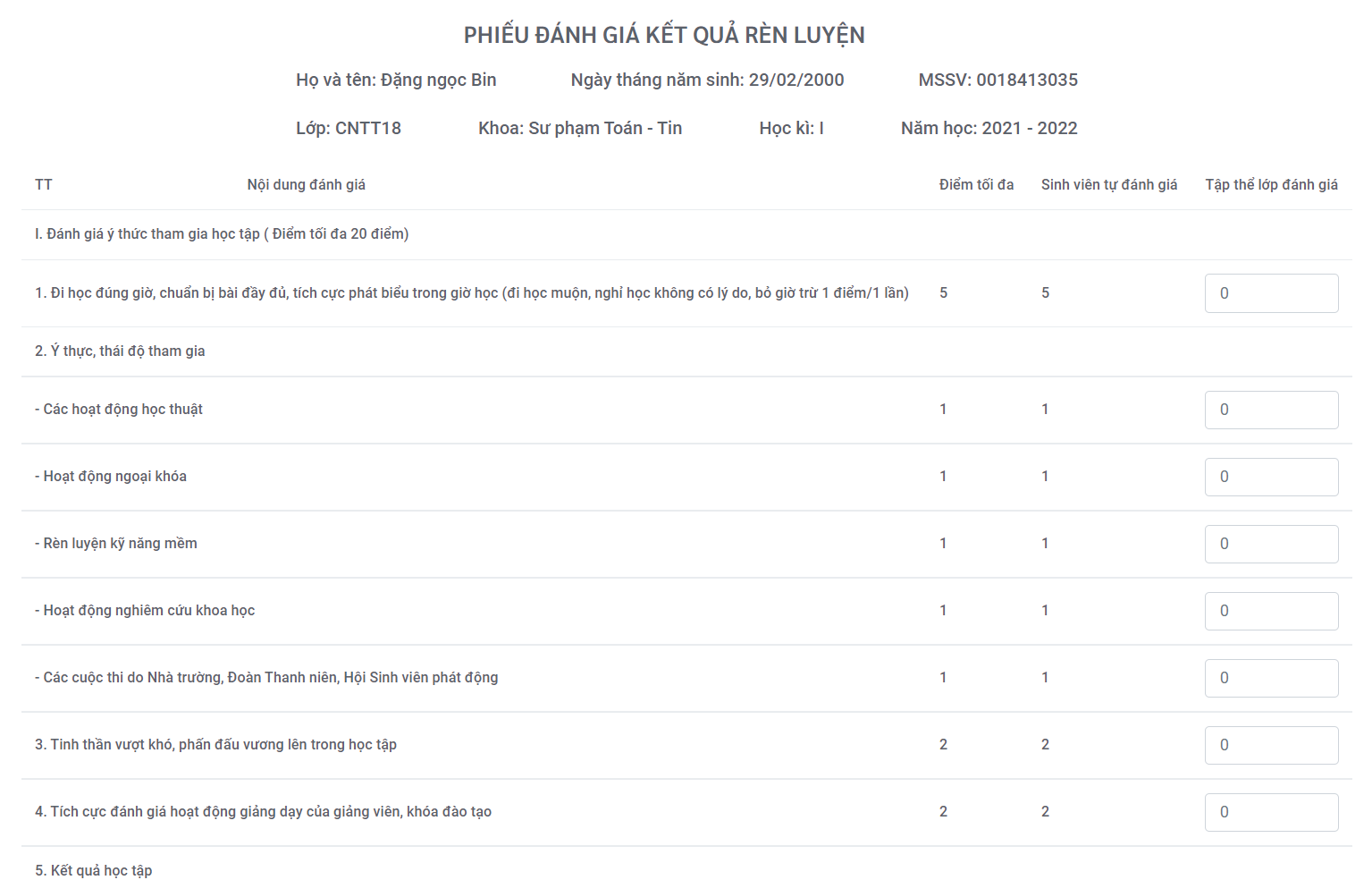
Hình 3.5 Giao diện xuất danh sách lớp

Sau khi thao tác và hành động duyệt điểm rèn luyện năm học và kỳ đánh giá cán bộ sẽ biết được những sinh viên đã tạo phiếu, chưa tạo phiếu, chờ duyệt, và chờ khoa duyệt cuối cùng là trạng thái hoàn thành là khi cán bộ khoa đã duyệt phiếu. Cán bộ khoa sẽ xem được phiếu đã hoàn thành và chờ duyệt, trạng thái chờ duyệt là trạng thái chờ cán bộ lớp duyệt. Cán bộ lớp thao tác vào hành động với tình trạng *chờ duyệt* cán bộ sẽ đánh giá lại điểm của sinh viên đã gửi lên và duyệt chuyển sang trạng thái *Chờ khoa duyệt.*



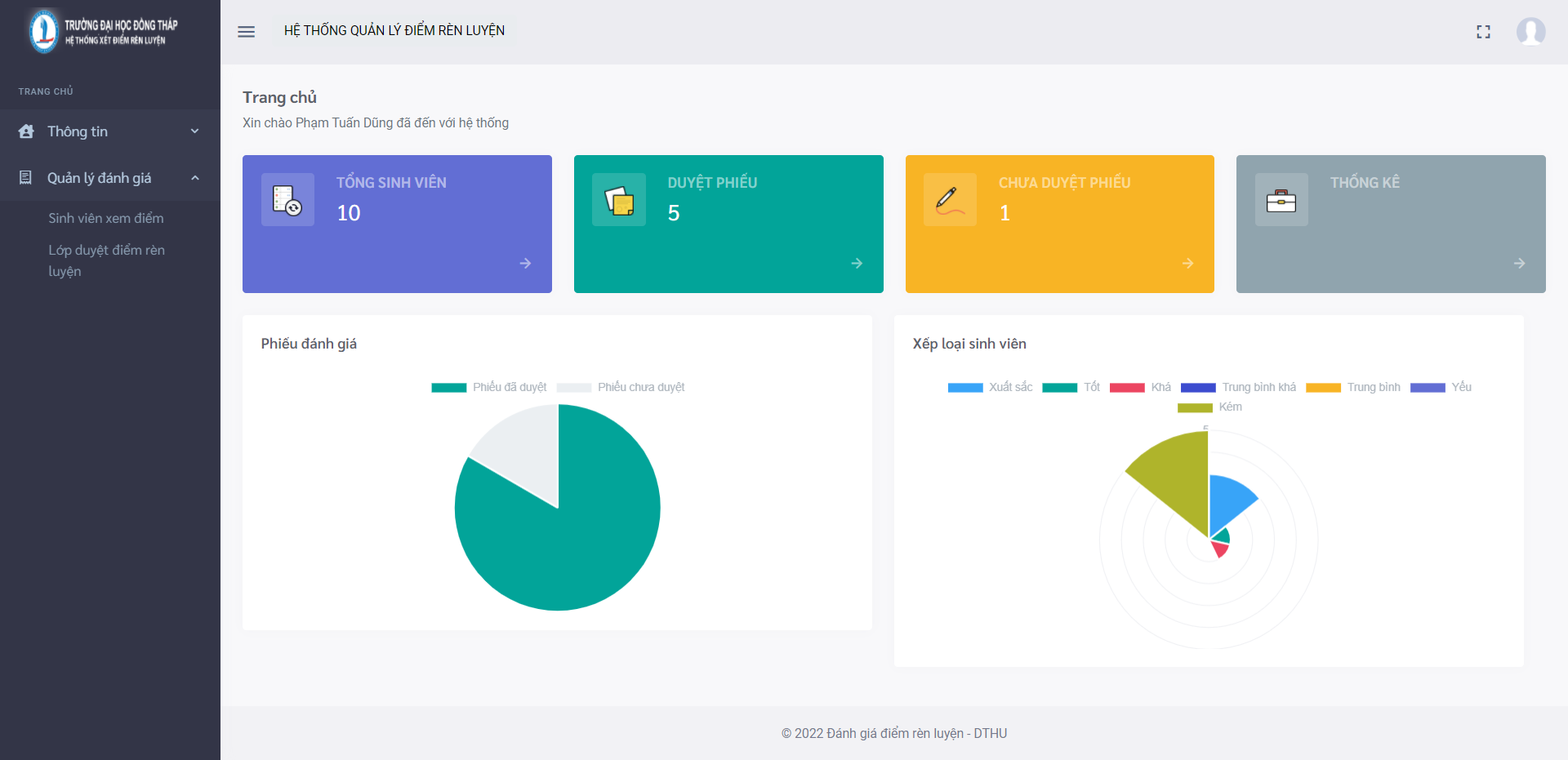
Hình 3.6 Giao diện cán bộ xem danh sách sinh viên chấm điểm

Các bộ lớp xem tình trạng xét điểm của sinh viên lớp mình. Nếu sinh viên đã tạo phiếu và tự đánh giá phiếu điểm rèn luyện của mình thì tình trạng sẽ chuyển sang chờ các bộ lớp duyệt, lúc này các bộ lớp sẽ thấy được hành động với tình trang là *chờ duyệt* Sau khi duyệt cho sinh viên hệ thống sẽ thông báo và gửi lên cán bộ khoa đánh giá và xét điểm rèn luyện lại một lần nữa *hoàn thành công việc đánh giá rèn luyện.*



Hình 3.7 Giao diện cán lớp chấm điểm lại cho sinh viên

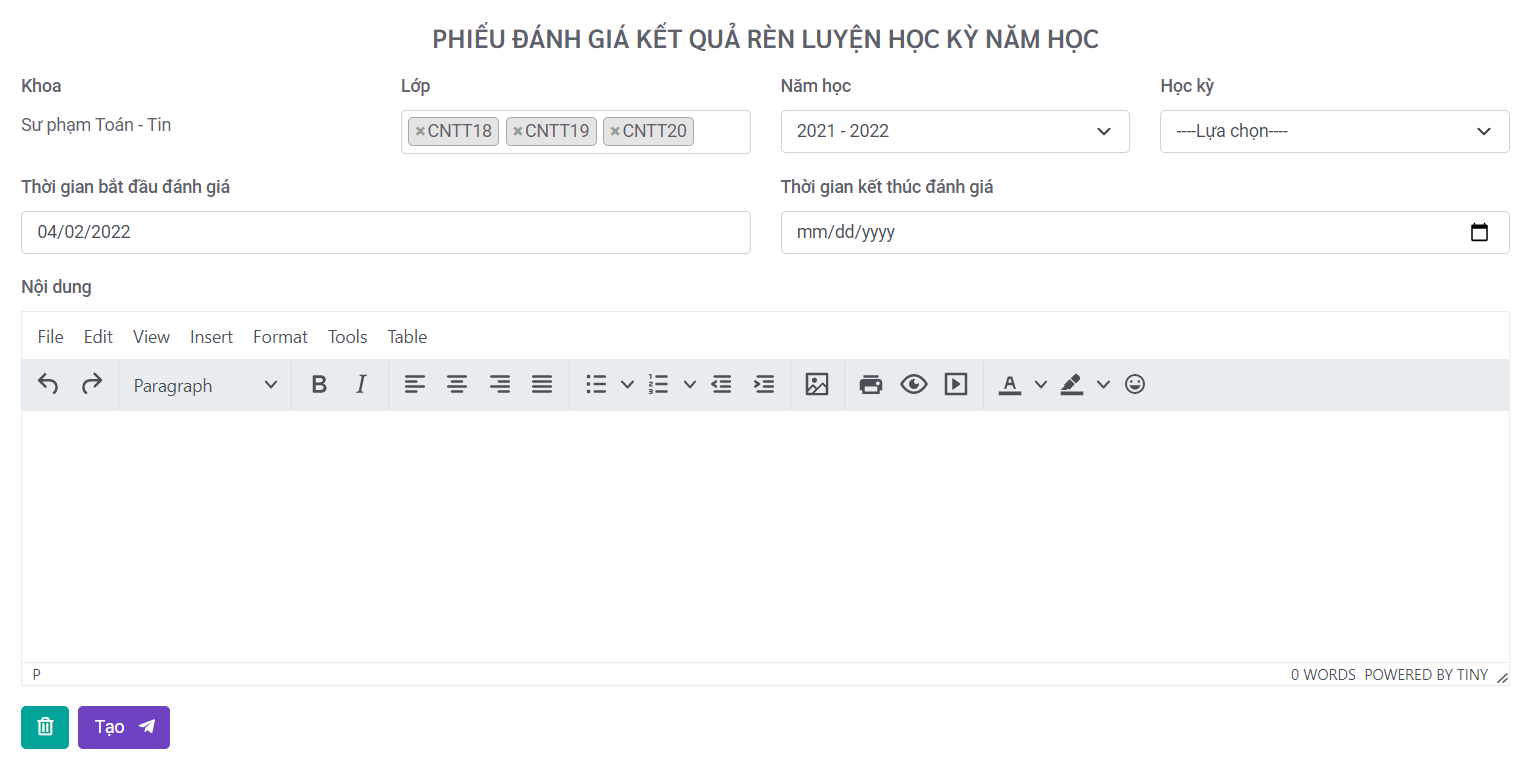
Cán bộ lớp xem thống kê tổng sinh viên lớp mình và phiếu đánh giá chưa duyệt và đã duyệt thông kế được bao nhiêu bạn xuất sắc, giỏi, khá, trung bình khá, khá, tốt, kém.



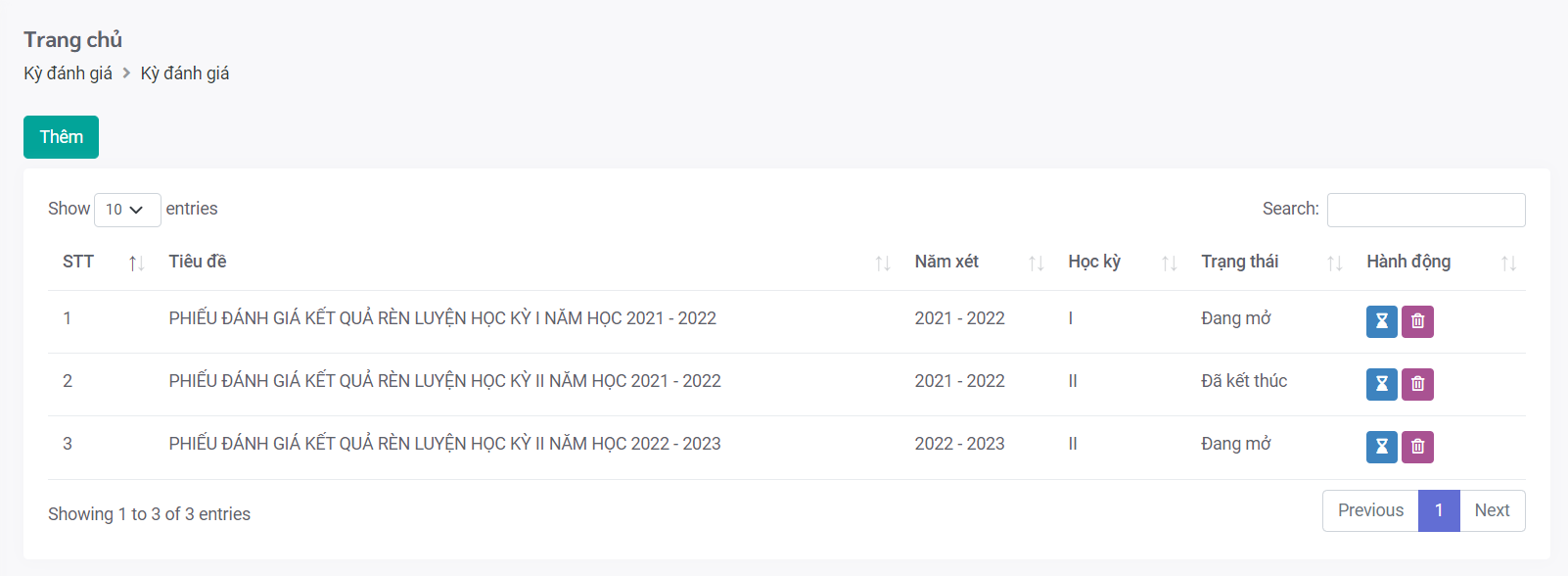
Hình 3.8 Giao diện cán lớp chấm điểm lại cho sinh viên

* 1. **Giao diện mở và duyệt điểm rèn luyện của cán bộ khoa**

Cán bộ khoa tạo kỳ giá rèn luyện cho các lớp của khoa mình và chỉ tạo được một kỳ đánh giá trong mỗi học kỳ của năm học. Ví dụ trong 1 năm học có 2 học kỳ thì chỉ được tạo 2 kỳ đánh giá trong năm đó.

**

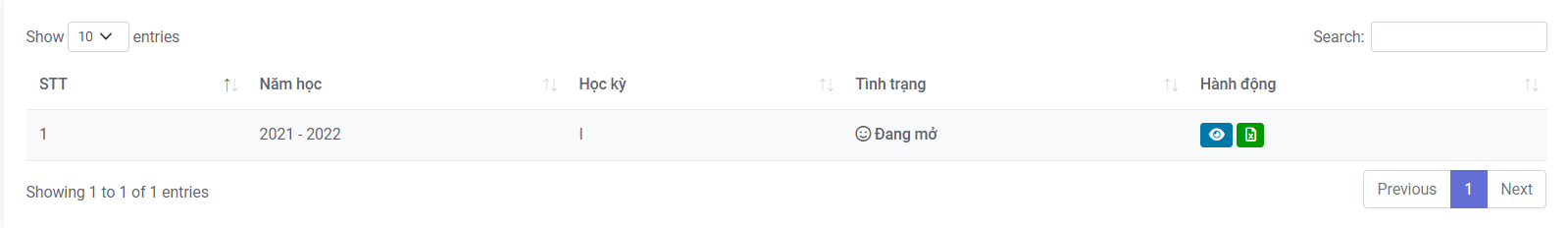
Hình 3.9 Giao diện tạo kỳ đánh giá



Hình 3.10 Giao diện kỳ đánh giá

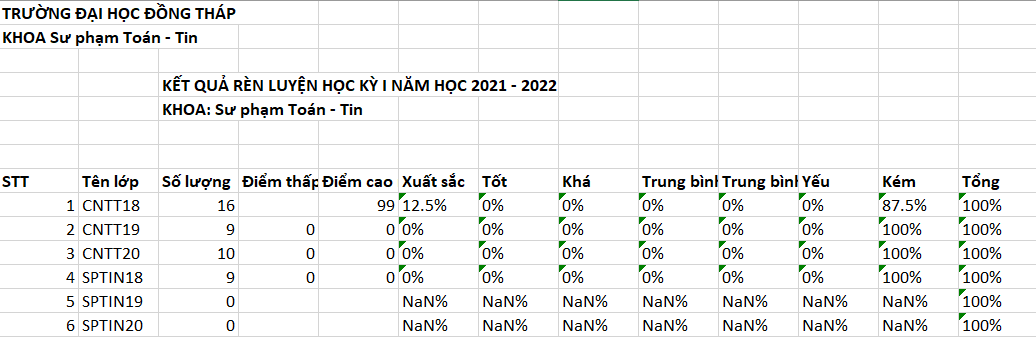
* 1. **Giao diện duyệt điểm rèn luyện cán bộ khoa**

Sau khi sinh viên đánh giá rèn luyện xong gửi lên cán bộ lớp duyệt và chấm điểm lại thì cán bộ khoa phải chịu trách nhiệm kiểm tra lại số liệu và duyệt cho sinh viên để hoàn thành việc đánh giá phiếu đánh giá của sinh viên. Sau khi thao tác vào hành động kỳ đáng mở hệ thống sẽ chuyển hướng xem kỳ có bao nhiêu lớp được khoa tạo để đánh giá phiếu đánh giá.

**

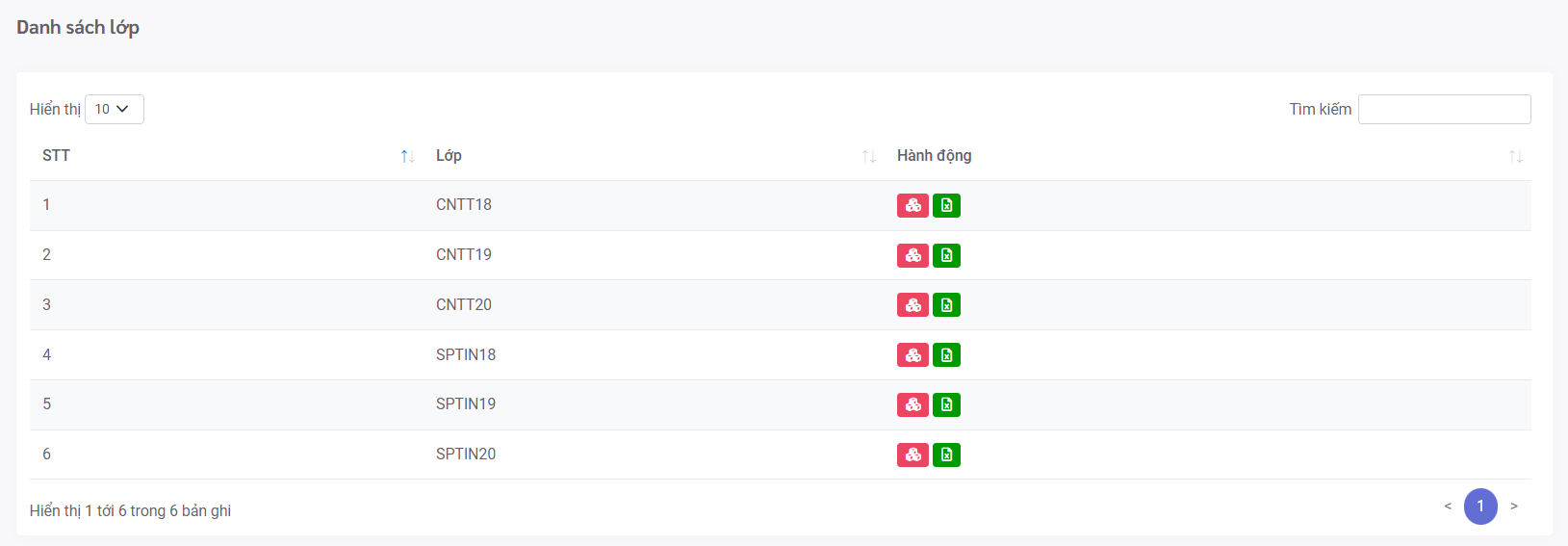
Hình 3.11 Giao diện kỳ đánh giá đã tạo

*Xuất báo cáo  thông kê* Danh sách các lớp được cán bộ khoa mở trong kỳ đánh giá.



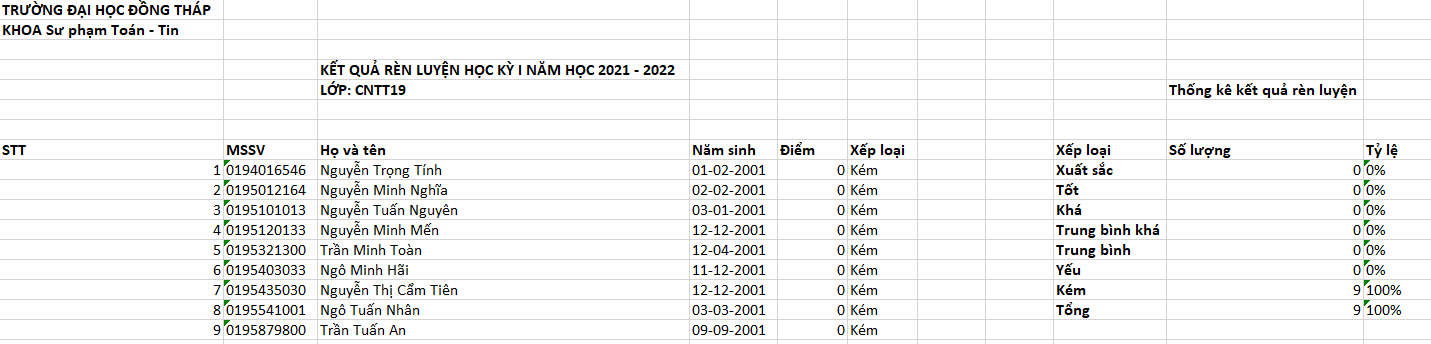
Hình 3.12 Giao diện kỳ danh sách lớp trong kỳ đánh giá

Thao tác  hiện danh sách trình trạng sinh viên đánh giá phiếu rèn luyện.

**

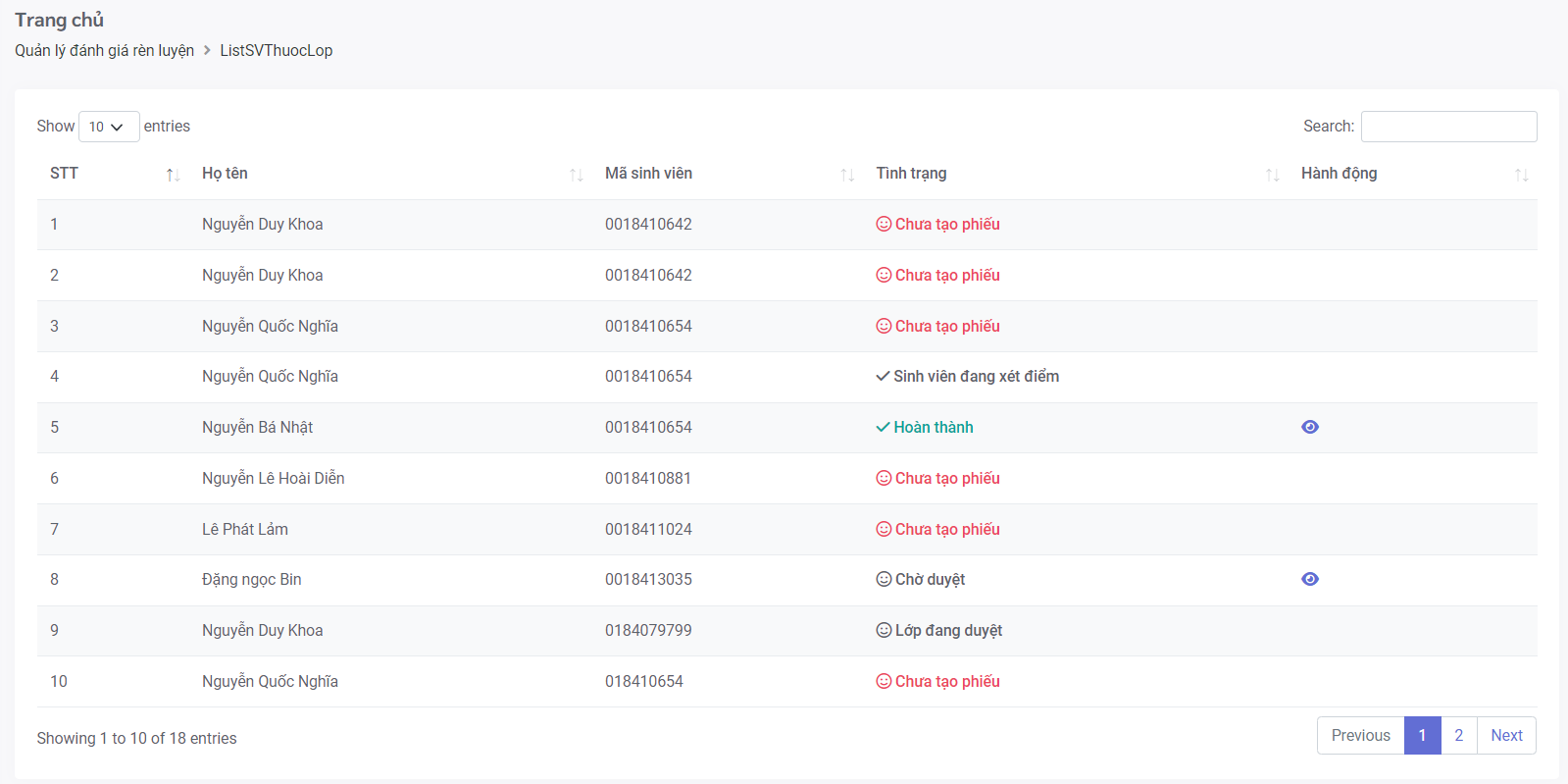
Hình 3.13 Giao diện xem lớp mở của kỳ đánh giá

Xuất báo cáo danh sách sinh viên của lớp đánh giá phiếu rèn luyện thống kê các thông tin như: Tên sinh viên, mã sinh viên, điểm xét rèn luyện, xếp loại rèn luyện. Nếu như cán bộ khoa đã kết thúc kỳ đánh giá mới có thể xuất được báo cáo ngược lại sẽ không xuất được khi kỳ đánh giá vẫn đang mở.



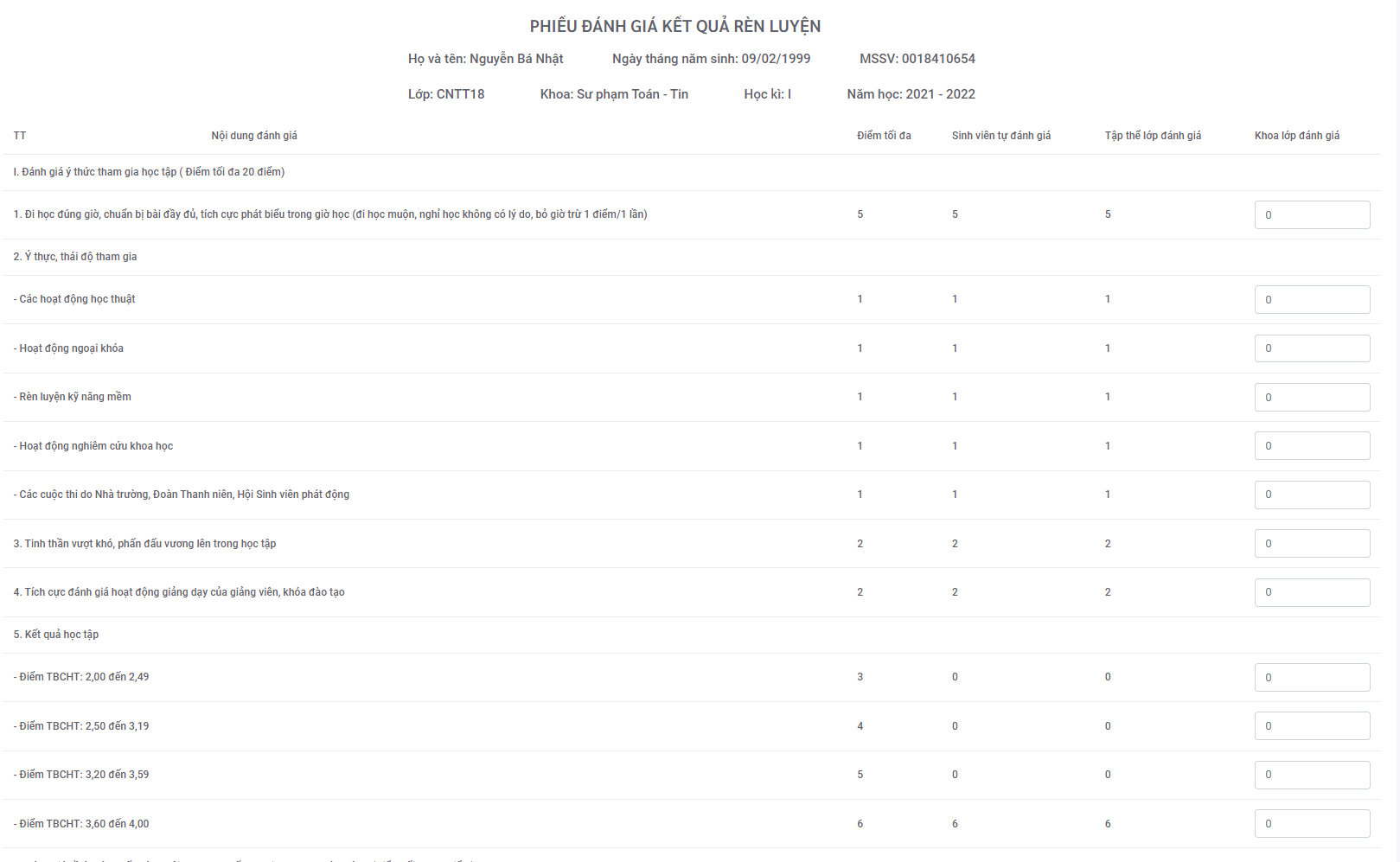
Hình 3.14 Giao diện xem lớp mở của kỳ đánh giá

Cán bộ khoa có thể xem những sinh viên đã tạo phiếu và chấm điểm được cán bộ lớp duyệt , cuối cùng sẽ được khoa đánh giá chấm điểm lại nghiệp vụ chấm điểm rèn luyện sinh viên và hoàn thành việc chấm điểm rèn luyện của sinh viên.

****

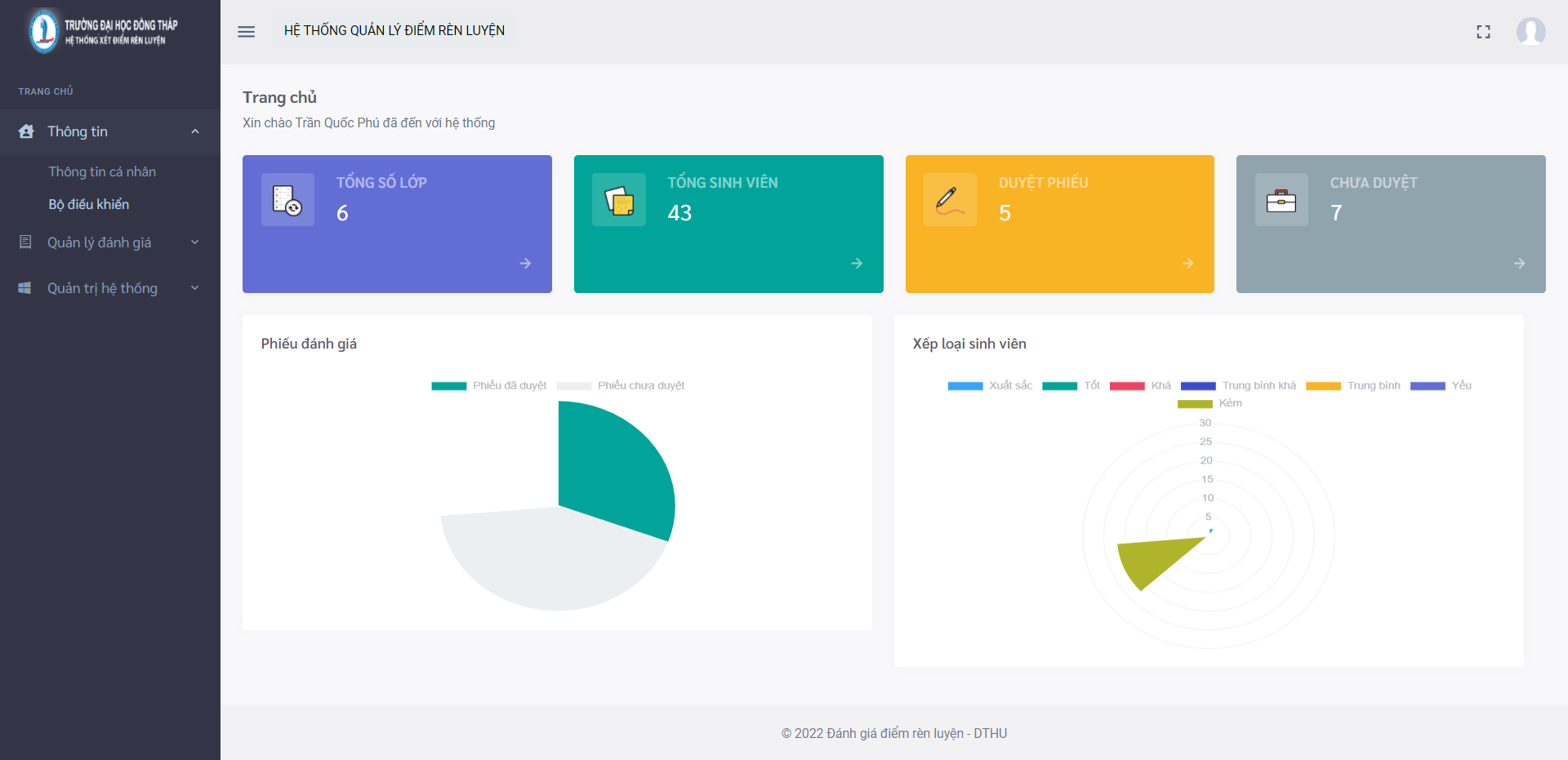
Hình 3.15 Giao diện danh sách sinh viên kỳ đánh giá mở

Cán bộ khoa đánh giá lại điểm của từng sinh viên thông khoa các tiêu chí có sẵn sau khi việc đánh giá phiếu rèn luyện hoàn tất cán bộ khoa *Duyệt* phiếu và chã kết quả *hoàn thành* về cho sinh viên.

****

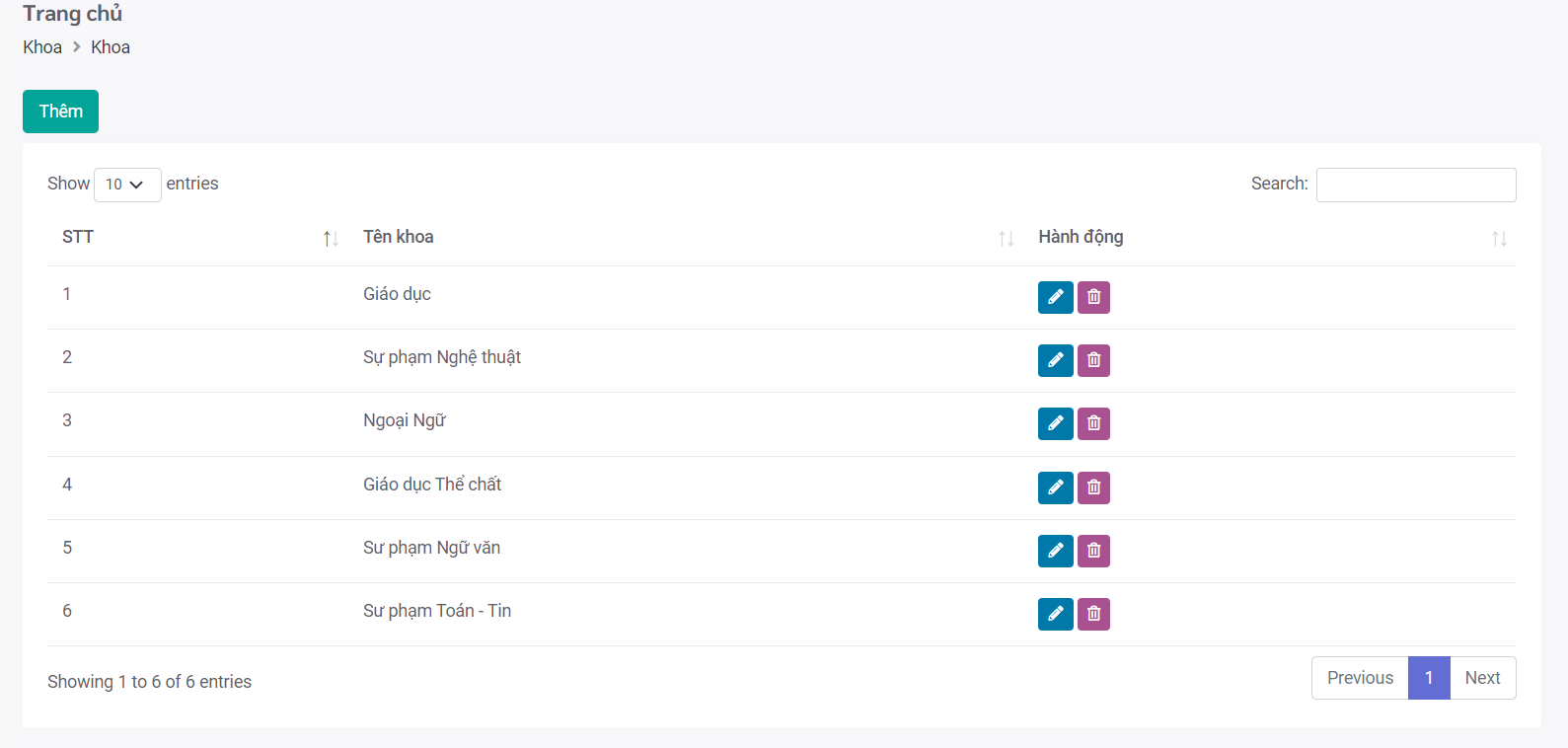
Hình 3.16 Giao diện đánh giá phiếu rèn luyện cán bộ khoa

Cán bộ khoa thống kê tổng số lớp, tổng số sinh viên của khoa thuộc cái lớp của khoa, phiếu điểm đã duyệt và chưa duyệt. Thông kế số lượng sinh viên xuất sắc, tốt, khá, trung bình khá, trung bình, yếu, kém.



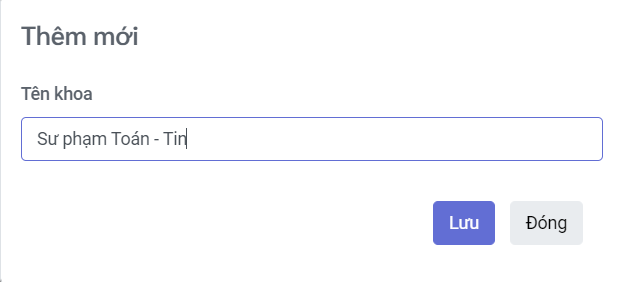
Hình 3.16 Giao diện đánh giá phiếu rèn luyện cán bộ khoa

* 1. **Quản trị hệ thống** 
     1. **Khoa**

Giao diện quản lý khoa quản lý thông tin: Tên khoa và các thao tác nghiệp vụ thêm, sửa, xóa, tìm kiếm, phân trang.****

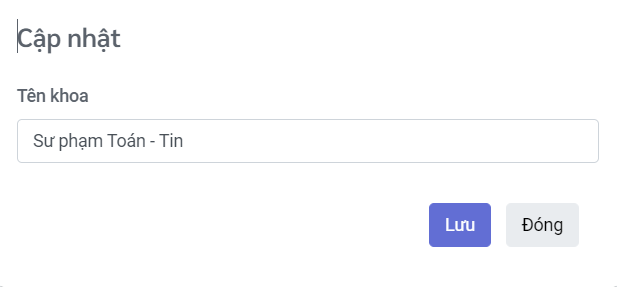
Hình 3.17 Giao diện quản lý khoa

* Thêm mới **** khoa

****

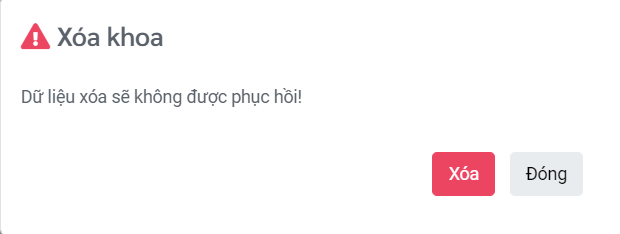
Hình 3.18 Giao diện thêm khoa mới

* Cập nhật  khoa sửa đổi thông tin khoa

****

Hình 3.19 Giao diện cập nhật khoa

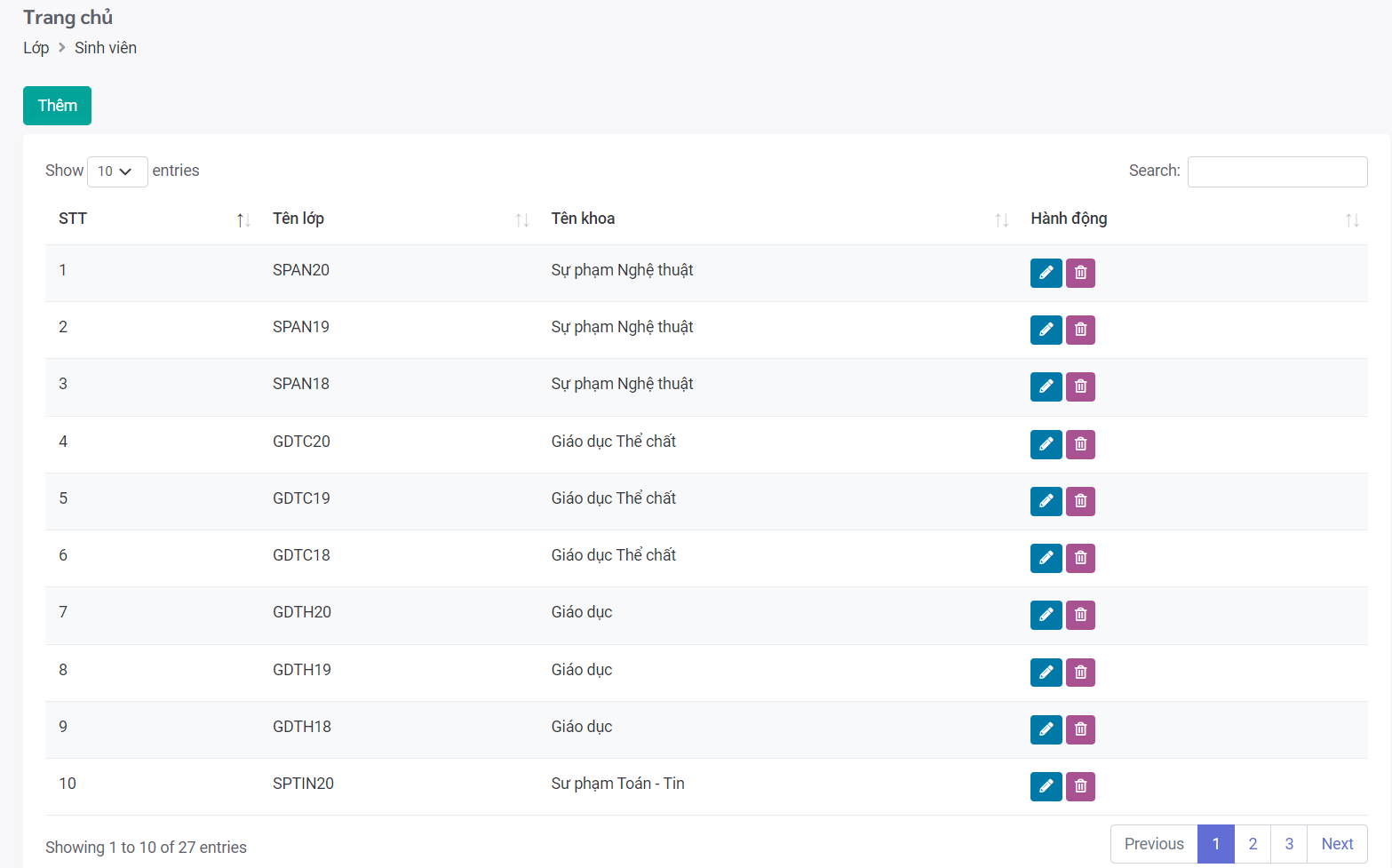
* Xóa  khoa hệ thống sẽ thông báo bạn có chắc chắn xóa khoa không nếu xóa *dữ liệu xóa sẽ không được phục hồi !*

****

Hình 3.20 Giao diện xóa khoa

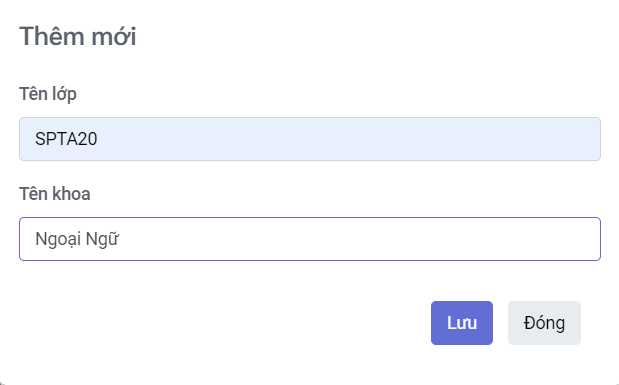
* + 1. **Lớp**

Giao diện quản lý lớp quản lý các thông tin như: tên lớp thuộc khoa nào và các thao tác nghiệp vụ thêm, sửa, xóa, tìm kiếm, phân trang.

****

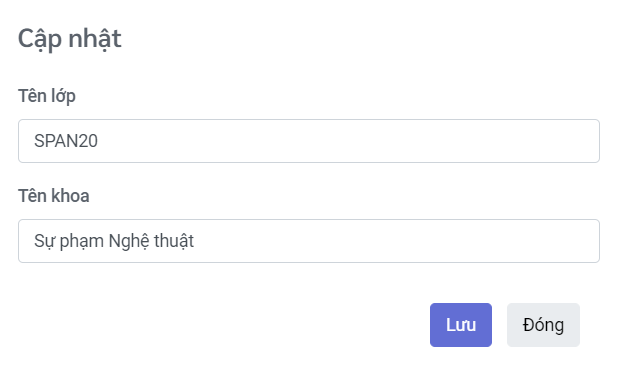
Hình 3.21 Giao diện lớp

* Thêm mới **** lớp và chọn khoa đạo tạo phù hợp với lớp vừa mới lên

****

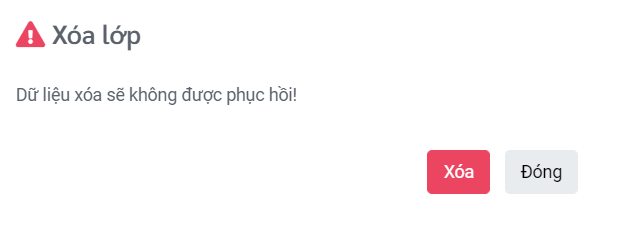
Hình 3.22 Giao diện thêm lớp

* Cập nhật  khoa sửa thông tin lớp và thuộc khoa nào.

****

Hình 3.23 Giao diện cập nhật lớp

* Xóa  lớp hệ thống sẽ thông báo bạn có chắc chắn xóa khoa không nếu xóa *dữ liệu xóa sẽ không được phục hồi !*

****

Hình 3.24 Giao diện xóa lớp

## **Cài đặt website**

* + 1. **Môi trường phát triển ứng dụng**
* **Hệ thống được phát triển trên nền C# .Net 4.6.1**

Ứng dụng được phát triển bằng ngôn ngữ lập trình C# là ngôn ngữ hướng đối tượng, hỗ trợ mô hình lập trình hướng đối tượng khá tốt, giúp cho việc xử lý được dễ dàng, có khả năng tái sử dụng và phát triển mã nguồn, tính đóng gói của ngôn ngữ giúp linh hoạt hơn trong việc sử dụng và hạn chế sự can thiệp từ bên ngoài.

Entity Framework phát hành đầu tiên vào năm 2008 nhằm hỗ trợ sự tương tác giữa các ứng dụng trên nền tảng . NET với các cơ sở dữ liệu quan hệ. Hay nói cách khác, nó là một công cụ giúp ánh xạ giữa các đối tượng trong phần mềm của bạn với các bảng của một cơ sở dữ liệu quan hệ.

* **Công cụ phát triển ứng dụng bao gồm:**

Microsoft Visual Studio 2010 (hoặc Visual Studio 2008) là công cụ phát triển chính của ứng dụng.

Visual Studio Code 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 và là mã nguồn mở chính là những ưu điểm vượt trội khiến Visual Studio Code ngày càng được ứng dụng rộng rãi.

Hệ quản trị cơ sở dữ liệu SQL Server 2019 (hoặc SQL Server 2019).

**3.8**. **Kết luận**

Thông qua việc cài đặt và phát triển trên chúng ta có cái nhìn tổng thể về phương thức hoạt động cách chấm điểm rèn luyện và quản lý của toàn hệ thống.

# **PHẦN BA: KẾT LUẬN**

1. **Kết quả đạt được:**

Qua thời gian nghiên cứu tìm hiểu công nghệ và viết ứng dụng, em đã đạt được một số kết quả nhất định. Hoàn thành đề tài đúng thời hạn đã giao. Các chức năng quản lý cơ bản đã đáp ứng được đầy đủ các yêu cầu nghiệp vụ đặt ra ban đầu. Những chức năng cụ thể mà một website quản lý điểm rèn luyện đã thực hiện được như sau:

* Nhóm chức năng tài khoản sinh viên
* Xét điểm rèn luyện
* Xem lại điểm các kỳ xét điểm
* Xuất excel phiếu đánh giá
* Nhóm chức năng tài khoản cán bộ lớp
* Xét điểm rèn luyện
* Xuất excel phiếu đánh giá
* Xem lại điểm các kỳ xét điểm
* Xuất excel danh sách đánh giá
* Thông kê
* Duyệt phiếu rèn luyện
* Nhóm chức năng cán bộ khoa
* Tạo kỳ xét điểm
* Danh sách kỳ xét điểm
* Duyệt rèn luyện
* Xuất danh sách lớp trong kỳ đánh giá
* Xuất excel danh sách đánh giá
* Thông kê
* Quản lý khoa
* Quản lý lớp
* Quản lý chức vụ
* Quản lý tài khoản
* Quản lý danh mục
* Quản lý quyền
* Tiêu chí

**2. Hướng phát triển:**

Em xin đưa ra một số hướng phát triển cho đề tài như sau:

- Do đề tài mang tính thực tiễn cao, nên nhóm nghiên cứu cần tiến hành hoàn thiện hơn trong việc phân tích thiết kế cơ sở dữ liệu, để đảm bảo tính đúng đắn, tối ưu và khả năng phát triển.

- Sẽ tiếp tục tối ưu hiệu suất của website.

- Chỉnh sửa những lỗi hiện tại của website, tối ưu hóa tốc độ và cập nhật giao diện đẹp hơn để tăng trải nghiệm người dùng.

# **TÀI LIỆU THAM KHẢO**

1. **Sách, giáo trình**

[1]. Tài liệu lập trình C# – Trường Đại Học Khoa Học Tự Nhiên - Đại học quốc gia TP. HCM.

[2]. Phân tích thiết kế hệ thống thông tin – Trần Đình Quế.

1. **Các website tham khảo**

[3]. <https://getbootstrap.com/>

[4].<https://docs.microsoft.com/en-us/aspnet/mvc/overview/getting-started/introduction/getting-started>

[5].[https://docs.microsoft.com/en-us/aspnet/core/data/ef-mvc/intro?view=aspnetcore-6.0](https://docs.microsoft.com/en-us/aspnet/core/data/ef-mvc/intro?view=aspnetcore-6.0%20%20)