

**TRƯỜNG ĐẠI HỌC THỦY LỢI**

**CƠ SỞ 2**

-----🙞🙜🕮🙞🙜-----



**ĐỀ TÀI: PHẦN MỀM QUẢN LÝ ĐỒ ÁN TỐT NGHIỆP**

**GIẢNG VIÊN: Th.S VIÊN THANH NHÃ**

**SINH VIÊN: MAI VIỆT THÚC**

**MSSV: 1751060431**

**LỚP: S20-K59TH**

**MỤC LỤC:**

[**CHƯƠNG 1. ĐẶT VẤN ĐỀ** 4](#_Toc21335433)

[**CHƯƠNG 2. CÁC TOOL, PHẦN MỀM ĐƯỢC SỬ DỤNG** 5](#_Toc21335434)

[**1.** **Hệ quản trị cơ sở dữ liệu SQL Server 2008** 5](#_Toc21335435)

[**2.** **ArgoUML** 5](#_Toc21335436)

[**3.** **PowerDesigner 16** 6](#_Toc21335437)

[**4.** **NetBeans IDE 8.0.2** 6](#_Toc21335438)

[**CHƯƠNG 3. DANH MỤC CÁC TỪ KHÓA** 7](#_Toc21335439)

[**Danh mục các thuật ngữ:** 7](#_Toc21335440)

[**CHƯƠNG 4. TÀI LIỆU THAM KHẢO** 8](#_Toc21335441)

[**CHƯƠNG 5: TÓM TẮT PHẦN MỀM** 9](#_Toc21335442)

[**CHƯƠNG 6: YÊU CẦU CHỨC NĂNG** 11](#_Toc21335443)

[**1.** **Yêu cầu chức năng của hệ thống** 11](#_Toc21335444)

[**2.** **Phi chức năng** 12](#_Toc21335445)

[**CHƯƠNG 7: MÔ HÌNH ER** 13](#_Toc21335446)

[**CHƯƠNG 8:** 14](#_Toc21335447)

[**1.** **MÔ HÌNH LOGIC** 14](#_Toc21335448)

[**2.** **MÔ HÌNH VẬT LÝ** 15](#_Toc21335449)

[**CHƯƠNG 9: MÔ HÌNH QUAN HỆ** 16](#_Toc21335450)

[**CHƯƠNG 10: USE CASE CÁC TÍNH NĂNG CƠ BẢN** 16](#_Toc21335451)

[**1.** **USE CASE “QUẢN LÝ ĐỒ ÁN TỐT NGHIỆP”** 16](#_Toc21335452)

[**1.1.** **USE CASE CON “QUẢN LÝ ĐỀ TÀI”** 17](#_Toc21335453)

[**1.2.** **USE CASE “QUẢN LÝ THÔNG TIN GIẢNG VIÊN”** 18](#_Toc21335454)

[**1.3.** **USE CASE “QUẢN LÝ GIẢNG VIÊN PHỤ TRÁCH HD PB UV”** 19](#_Toc21335455)

[**1.4** **USE CASE “QUẢN LÝ HỘI ĐỒNG”** 20](#_Toc21335456)

[**1.5** **USE CASE “QUẢN LÝ SINH VIÊN VÀ ĐỀ TÀI THỰC HIỆN”** 21](#_Toc21335457)

[**CHƯƠNG 11: CLASS DIAGRAM** 23](#_Toc21335458)

[**CHƯƠNG 12: FROM GIAO DIỆN** 24](#_Toc21335459)

[**CHƯƠNG 13: CODE CHƯƠNG TRÌNH** 29](#_Toc21335460)

# **CHƯƠNG 1. ĐẶT VẤN ĐỀ**

Quản lý điểm và đồ án tốt nghiệp là một công việc rất quan trọng và không thể thiếu trong các trường đại học mà điển hình là khoa CNTT trường Đại học Thủy Lợi Cơ Sở 2.

Thay vì lưu dữ liệu của đồ án tốt nghiệp và các thông tin cần thiết và liên quan vào các filel excel hay access như truyền thống vốn khó trong việc quản lý và lưu trữ, không có tính liên kết logic, dễ dàng gặp lỗi và xung đột dữ liệu khi cập nhật hay sửa chửa thông tin, nhất là mỗi khi muốn tìm kiếm phải làm thủ công mất rất nhiều thời gian . Phần mềm quản lý điểm và đồ án tốt nghiệp không chỉ giúp quản lý điểm và đồ án một cách vô cùng trực quan dễ dàng sử dụng, cập nhật, thay đổi, xóa thông tin mang tính logic cao mà còn giúp người dùng dễ dàng tìm kiếm được thông tin một cách vô cùng tiện lợi, nhanh chóng.

Bên cạnh đó, hàng năm sau mỗi kỳ làm và bảo vệ đề tài tốt nghiệp của sinh viên người, quản lý đồ án tốt nghiệp sinh viên phải lưu lại các tài liệu liên quan và thông tin của các đồ án tốt nghiệp để làm tài liệu tham khảo cho các sinh viên khóa sau. Điều này sẽ trở nên dễ dàng hơn đối với phần mềm quản lý điểm và đồ án tốt nghiệp khi phần mềm có kèm chức năng xuất ra file excel để đễ dàng chia sẻ, lưu trữ…

# **CHƯƠNG 2. CÁC TOOL, PHẦN MỀM ĐƯỢC SỬ DỤNG**

1. **Hệ quản trị cơ sở dữ liệu SQL Server 2008**

SQL Server là một hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS) hay còn được gọi là Realational Database Mannagement Systems. Cơ sở dữ liệu quan hệ là cơ sở dữ liệu, mà dữ liệu bên trong nó được tổ chức thành các bằng. Các bảng được tổ chức bằng các nhóm dữ liệu theo cùng chủ đề và có chứa các cột và các hàng thông tin. Sau đó các bảng này được liên kết với nhau bởi bộ Database Engine khi có yêu cầu. Cơ sở dữ liệu quan hệ là một trong những mô hình cơ sở dữ liệu thông dụng nhất hiện nay.

Hệ quản trị cơ sở dữ liệu “SQL Server” là một trong những phần mềm tiện lợi và hiệu quả trong việc phát triển các ứng dụng cơ sở dữ liệu lớn, phân tách tích hợp cho cơ quan, tổ chức, địa phương.

SQL Server hỗ trợ tốt trong quá trình quản lý xử lý đồng nhất, bảo mật dữ liệu theo mô hình Client/Server trên mạng.

SQL Server là một hệ cơ sở dữ liệu quan hệ đủ mạnh, phù hợp với ASP.NET MVC. Thông qua đó có thể dễ dàng lưu trữ, sắp xếp, tìm kiếm và hiển thị dữ liệu trên SQL Server. Máy chủ SQL Server quản lý việc truy cập dữ liệu sẽ đảm bảo cho việc truy cập và sử dụng dữ liệu đồng thời bởi nhiều người, đảm bảo rằng chỉ có những người dùng hợp lệ mới có quyền truy cập cơ sở dữ liệu và làm tăng tốc độ truy cập cơ sở dữ liệu. Căn cứ vào các ưu điểm đã nêu ở trên SQL Server được chon làm hệ quản trị cơ sở của hệ thống.

SQL Server 2008 có tác dụng đòn bẩy cho công nghệ .NET 3.0 (Dot Net Framework 3.0) với LINQ (Language Integrated Query – ngôn ngữ truy vấn tích hợp). Thêm vào đó là sự hỗ trợ hiệu quả hơn cho các thực thể dữ liệu doanh nghiệp cùng với các tùy chọn đồng bộ dữ liệu.

SQL Server được dùng để lưu trữ và truy xuất dữ liệu cho phần mềm quản lý đồ án.

1. **ArgoUML**

ArgoUML là một ứng dụng lập biểu đồ UML được viết bằng java và được phát hành theo mã nguồn mở Eclipse. Nhờ có ứng dụng Java, nó có sẵn trên mọi nền tảng được java SE hỗ trợ.

ArgoUML được dùng để vẽ mô hình Use Case và Class Diagram nhằm giúp xác định được mục đích và yêu cầu của phần mềm quản lý đồ án.

1. **PowerDesigner 16**

SAP PowerDesigner là một công cụ mo hình hóa doanh nghiệp hợp tác được sản xuất bởi Sybase, hiện thuộc sở hữu của SAP. Nó có thể chạy trong Microsoft Windows dưới dạng một ứng dụng gốc hoặc trong môi trường Eclipse thông qua một plugin. Nó hỗ trợ thiết kế phần mềm kiến trúc hướng mô hình và lưu trữ các mô hình bằng nhiều phần mở rộng tệp, chẳng hạn như .bpm, .cdm và .pdm. cấu trúc tệp bên trong có thể là XML hoặc định dạng tệp nhị phân nén. Nó cũng có thể lưu trữ các mô hình trrong một kho lưu trữ cơ sở dữ liệu.

PowerDesigner được dùng để vẽ các mô hình quan hệ, mô hình logic, vật lý phục vụ cho quá trình phân tích và thiết kế chương trình.

1. **NetBeans IDE 8.0.2**

**NetBeans** là một môi trường phát triển tích hợp (IDE) cho java. Netbeans cho phép các ứng dụng được phát triển từ tập hợp các thành phần mềm mô-đun được gọi là mô-đun. NetBeans chạy trên windows, macos, linux và Solaris. Ngoài việc phát triển cho java, nó còn có các phần mở rộng cho ngôn ngữ khác nhau nhu PHP, C, C++, HTML5 và javaScript. Các ứng dụng dựa trên NetBeans, bao gồm NetBeans IDE, có thể được các nhà phát triển bên thứ ba mở rộng.

Netbean được dùng để thiết kế giao diện người dùng và source code trên nề tảng java, phục vụ cho việc built giao diện và các thuật toán phục vụ cho các chức năng trong phần mềm.

# **CHƯƠNG 3. DANH MỤC CÁC TỪ KHÓA**

**Danh mục các thuật ngữ:**

CNTT: công nghệ thông tin.

CSDL: cơ sở dữ liệu.

SQL: Structured Query Language- ngôn ngữ truy vấn mang tính cấu trúc.

ERD: Entity Relationship Diagram.

CDM: Conceptual Data Model.

LDM: Logical Data Model.

PDM: Physical Data Model.

UML: **Ngôn ngữ mô hình hóa thống nhất** ([tiếng Anh](https://www.wikiwand.com/vi/Ti%E1%BA%BFng_Anh): *Unified Modeling Language*, viết tắt thành **UML**) là một ngôn ngữ mô hình gồm các ký hiệu đồ họa mà các phương pháp hướng đối tượng sử dụng để thiết kế các hệ thống thông tin một cách nhanh chóng.

Eclipse: là một môi trường phát triển tích hợp cho java, PHP, C, C++, C#, Python, HTML, XML, JavaScript. Được phát triển ban đầu bởi IBM và hiện nay bởi tổ chức Eclipse.

XML: ngôn ngữ đánh dấu mở rộng (extensible markup language) là ngôn ngữ đánh dấu với mục đích chung do w3c đề nghị, để tạo ra các ngôn ngữ đánh dấu khác. Đây là một tập hợp con đơn giản của SGML, có khả năng mô tả nhiều loại dữ liệu khác nhau nhằm đơn giản hóa việc chia sẻ dữ liệu giữa các hệ thống khác nhau, đặc biệt là các hệ thống được kết nối với internet.

# **CHƯƠNG 4. TÀI LIỆU THAM KHẢO**

* + SQL Server 2008: <https://vi.wikipedia.org/wiki/Microsoft_SQL_Server>
  + ArgoUML: <https://en.wikipedia.org/wiki/ArgoUML>
  + Khái niêm XML: <https://vi.wikipedia.org/wiki/XML>
  + PowerDesigner 16: <https://en.wikipedia.org/wiki/PowerDesigner>
  + NetBeans IDE: <https://en.wikipedia.org/wiki/NetBeans>
  + Khái niệm Eclipse: <https://en.wikipedia.org/wiki/Eclipse_(software)>
  + Giáo trình hệ quản trị cơ sở dữ liệu SQL- Đại học Thủy Lợi.
  + Giáo trình lập trình JAVA- Đại học Thủy Lợi.
  + Bài giảng môn Hệ quản trị cơ sở dữ liệu- Th.s Viên Thanh Nhã.
  + Bài giảng môn Phân tích thiết kế hệ thống thông tin- Th.S Viên Thanh Nhã.
  + Bài giảng môn JAVA- Th.S Viên Thanh Nhã.

# **CHƯƠNG 5: TÓM TẮT PHẦN MỀM**

Phần mềm quản lý đề tài sinh viên được xây dựng trên nhu cầu quản lý đề tài tốt nghiệp cho sinh viên trường đại học. Có thể đáp ứng đầy đủ các thao tác hay tác vụ về những vấn đề liên quan đến việc làm đề tài tốt nghiệp của sinh viên ở các trường đại học. Đảm bảo tiết kiệm thời gian chi phí quản lý hiệu quả các đề tài tốt nghiệp.

Về chức năng của phần mềm quản lý đồ án có thể đáp ứng các nghiệp vụ như xem thông tin của sinh viên như mã số sinh viên, tên sinh viên, địa chỉ, số điện thoại,.. của sinh viên đó hoặc tra cứu thông tin về các giảng viên hướng dẫn đề tài hay phản biên đề tài.

Phần mềm cho phép người sử dụng/ quản lý có thể tra cứu thông tin của bản thân như mã số sinh viên, mã số đề tài, thông tin cá nhân và điểm để đảm bảo không có sai sót và tính minh bạch trong công tác đánh giá, quản lý.

Ngoài ra, người dùng/ quản lý cũng có thể tra cứu được giảng viên và các đề tài mà họ hướng dẫn và đề tài mà họ phản biện, cập nhật điểm cho sinh viên và tra cứu học vị, học hàm tương ứng.

Phần mềm sẽ được điều hành bởi một người quản lý hoặc nhà trường đề đảm bảo tính an toàn cho thông tin, thay đổi thông tin nếu có sai sót.

Phần mềm quản lý đồ án có nhiều ưu điểm nổi bật: với phần mềm quản lý đồ án, người quản lý đồ án có thể cập nhật thông tin đồ án môt cách đầy đủ, nhanh chóng, tiết kiệm thời gian. Người dùng có thể truy xuất, tra cứu thông tin đồ án theo các tiêu chí mà mình quan tâm. Mỗi đồ án có rất nhiều thông tin liên quan như sinh viên thực hiện đồ án, giáo viên hướng dẫn, giáo viên phản biện, hội đồng, điểm cho đồ án… Khi đó, nhờ sự hỗ trợ của hệ thống, người quản lý sẽ rất dễ dàng trong việc quản lý, theo dõi, cập nhật những thông tin đó mà khôn cần phải tốn quá nhiều thời gian tìm kiếm như công việc quản lý thủ công trước kia.

Bên cạnh những ưu điểm kể trên, phần mềm quản lý đồ án vẫn tồn tại một số nhược điểm như chưa có chức năng quản lý đánh giá điểm cho giảng viên theo đồ án sinh viên mà mình hướng dẫn. Chưa thể chạy một lúc nhiều tính năng khi cần vì phần mềm không chia ra các cửa sổ. Những nhược điểm này có thể được cập nhật thêm trong các phiên bản sau.

# **CHƯƠNG 6: YÊU CẦU CHỨC NĂNG**

1. **Yêu cầu chức năng của hệ thống**

Hệ thống cho phép lưu trữ, cập nhật, tìm kiếm, thống kê, xóa, xuất dữ liệu ra các định dạng file khác nhau các thông tin liên quan đến đồ án tốt nghiệp sinh viên.

**Các chức năng cần đạt được:**

* Quản lý đề tài: chức năng này cho phép người quản lý được thêm mới các đề tài, sửa, xóa thông tin của đề tài như tên đề tài, mã số đề tài, số sinh viên tham gia,.. và lưu lại trong Cơ Sở Dữ Liệu.
* Quản lý sinh viên thực hiện đề tài: chức năng này cho phép người quản lý thêm, sửa, xóa các thông tin như tên sinh viên, số điện thoại, địa chỉ, lớp hay đề tài thực hiện.
* Quản lý giảng viên hướng dẫn: chức năng này cho phép người quản lý thêm, sửa, xóa đề tài phụ trách, điểm cho đề tài của sinh viên.
* Quản lý giảng viên phản biện: chức năng này cho phép người quản lý thêm, sửa, xóa đề tài phản biện, điểm cho đề tài của sinh viên.
* Quản lý hội đồng: chức năng này cho phép người quản lý thêm, sửa, xóa các thông tin của hội đồng như phòng hội đồng, ngày hướng dẫn, tình trạng của hội đồng, mã số của giáo viên chủ tịch hồng, quyết định của hội đồng đối với đề tài của sinh viên.
* Thống kê điểm: chức năng này cho phép người quản lý có thể thống kê điểm và xuất ra file excel cho sinh viên làm đồ án theo lớp.

**Chức năng quản lý giáo viên:**

* Quản lý thông tin giáo viên: chức năng này cho phép người quản lý được thêm mới, sửa, xóa các thông tin của giảng viên trong danh sách các giảng viên của khoa sẽ tham gia hướng dẫn, phản biện đề tài.
* Quản lý hướng nghiên cứu của giáo viên: chức năng này cho phép người quản lý thêm mới, xem, sửa, xóa các thông tin liên quan đến hướng nghiên cứu của các giảng viên tham gia hướng dẫn, phản biện đề tài.
* Quản lý thực hiện nghiên cứu của giảng viên: chức năng này cho phép người quản lý thêm mới, xem, sửa, xóa thông tin liên quan đến việc thực hiện nghiên cứu của các giảng viên tham gia hướng dẫn, phản biện đề tài đã hướng dẫn, phản biên.

**Chức năng xem thông tin:**

* Chức năng xem thông tin đồ án: chức năng này cho phép người quản lý, người dùng có thể xem các thông tin về đồ án, giảng viên hướng dẫn, giảng viên phản biện.
* Chức năng tìm kiếm giảng viên hướng dẫn: chức năng này cho phép người quản lý có thể tìm kiếm các thông tin về tất cả các đề tài mà giảng viên đó đã hướng dẫn.
* Chức năng tìm kiếm sinh viên: chức năng này cho phép người quản lý có thể tìm kiếm thông tin về sinh viên thực hiện đề tài theo lớp hoặc theo đề tài.

1. **Phi chức năng**

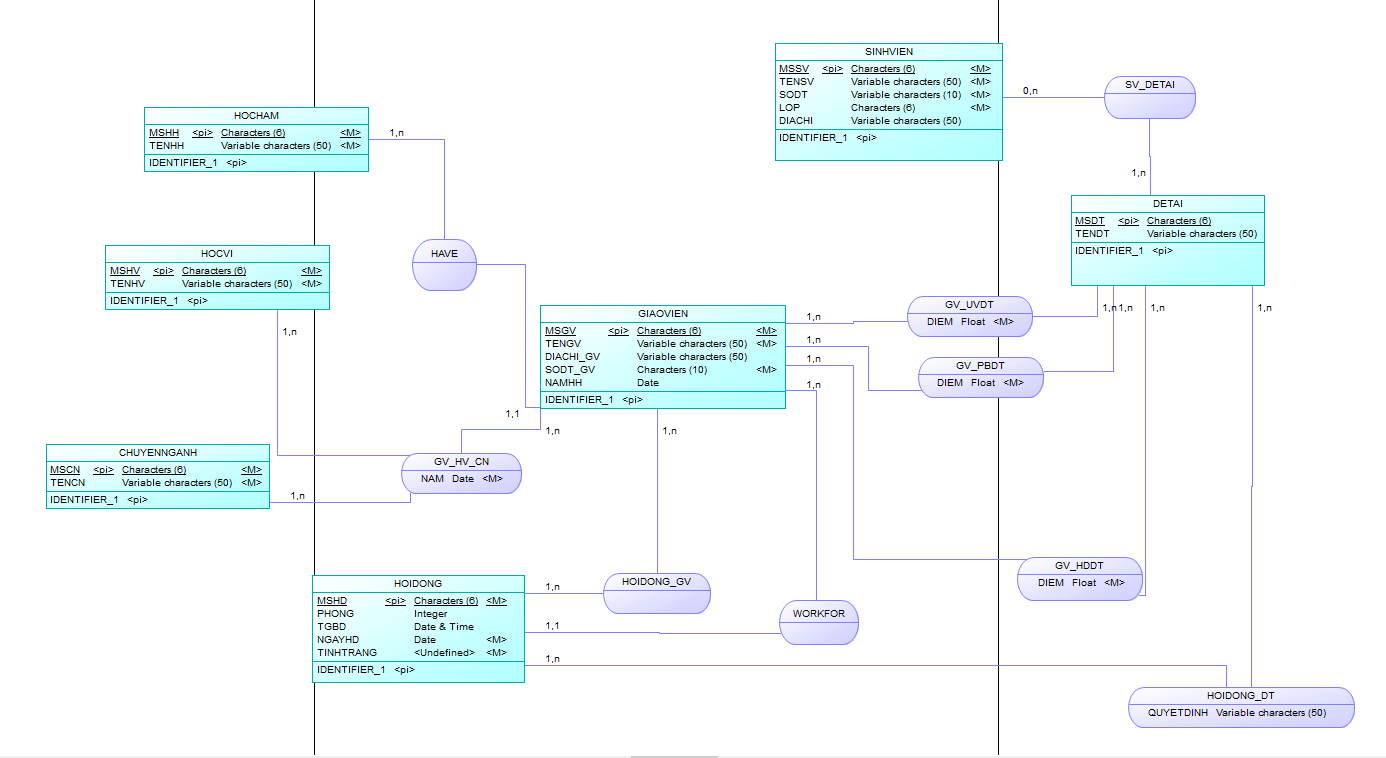
- Giao diện đẹp, bắt mắt.

- Trực quan.

- Dễ làm quen, dễ thao tác trên phần mềm.

- Phần mềm tối ưu cấu hình, có thể chạy trên máy có cấu hình thấp.

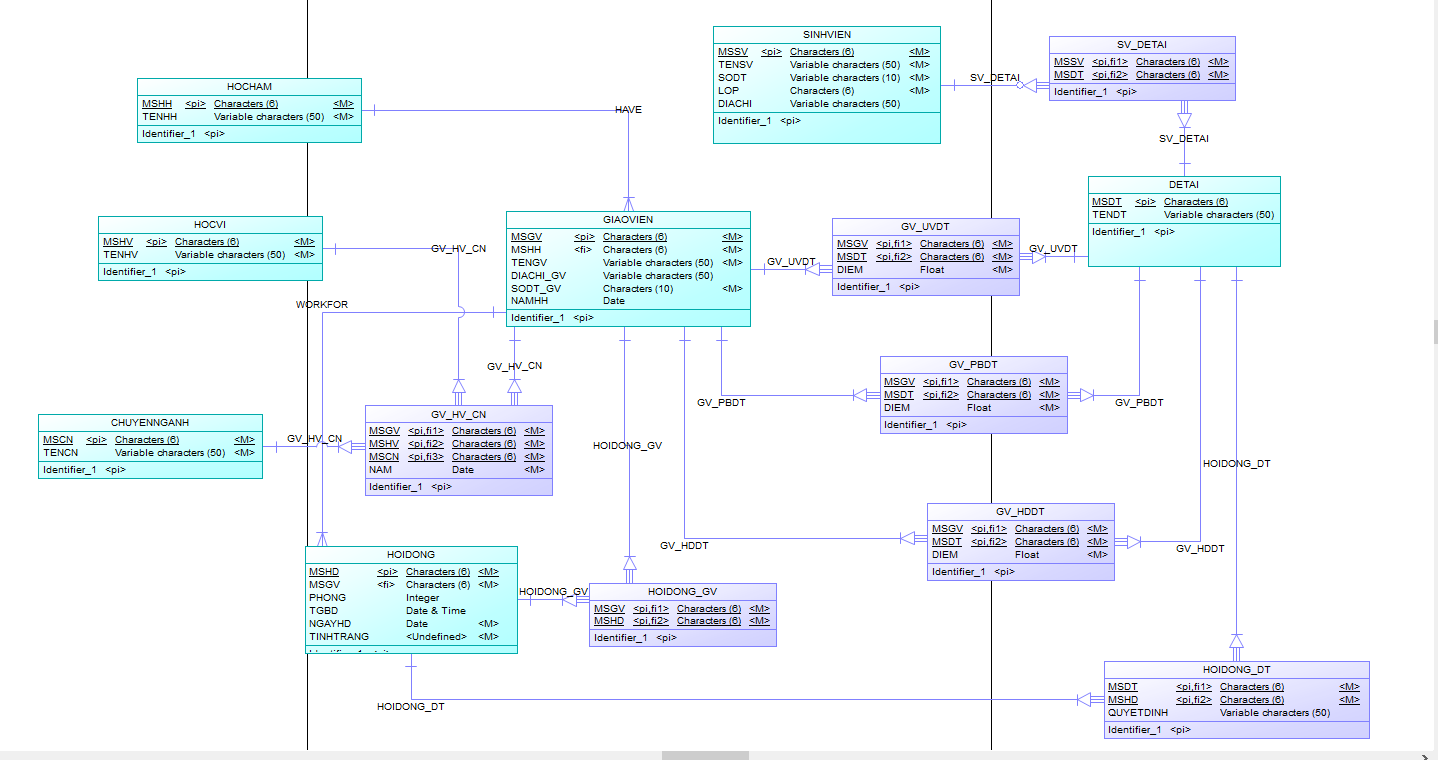
# **CHƯƠNG 7: MÔ HÌNH ER**

****

Hình: Mô hình mức quan niêm ER (CMD)

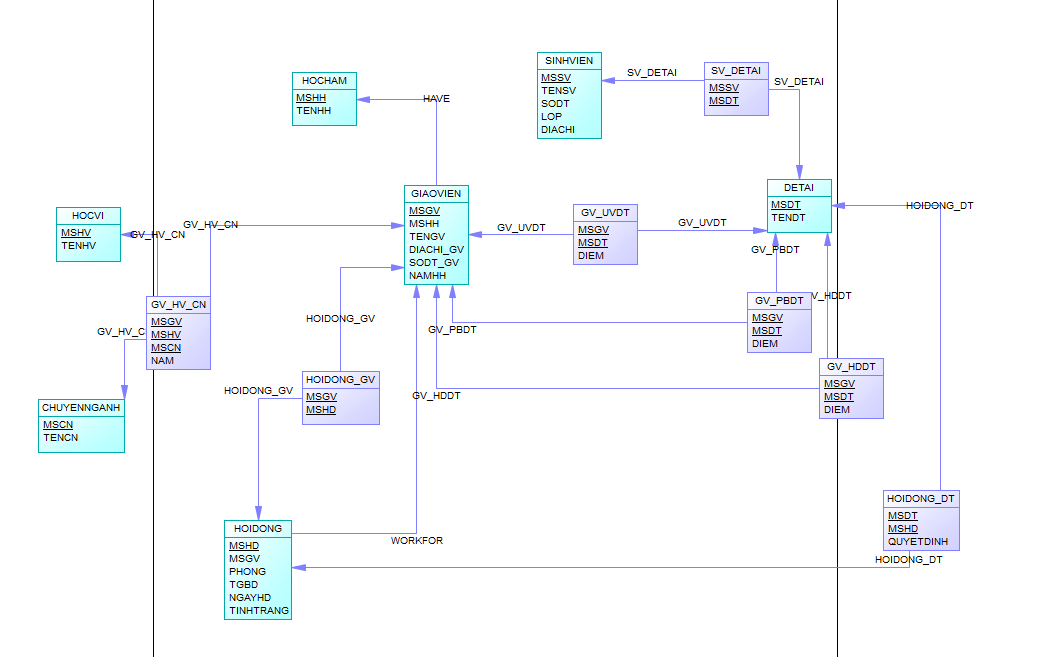
# **CHƯƠNG 8:**

1. **MÔ HÌNH LOGIC**

****

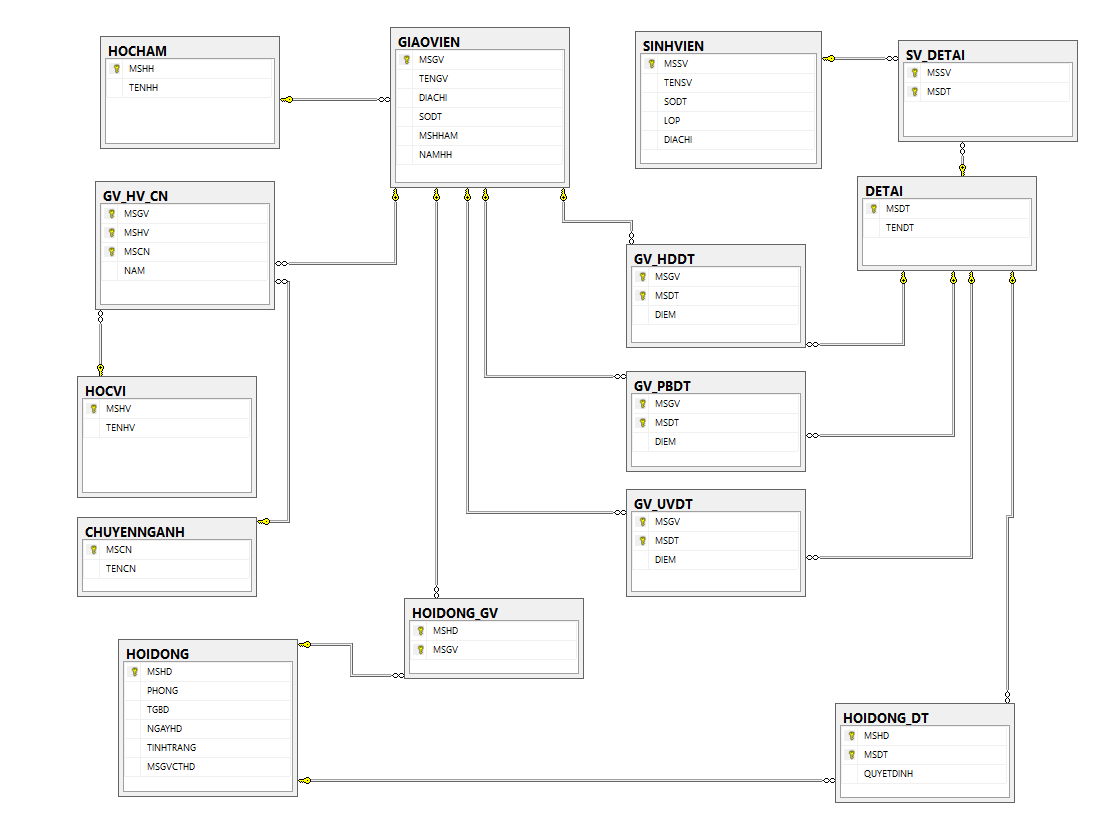
Hình: Mô hình Logic (LDM)

1. **MÔ HÌNH VẬT LÝ**

****

Hình: 3 Mô hình vật lý (PDM)

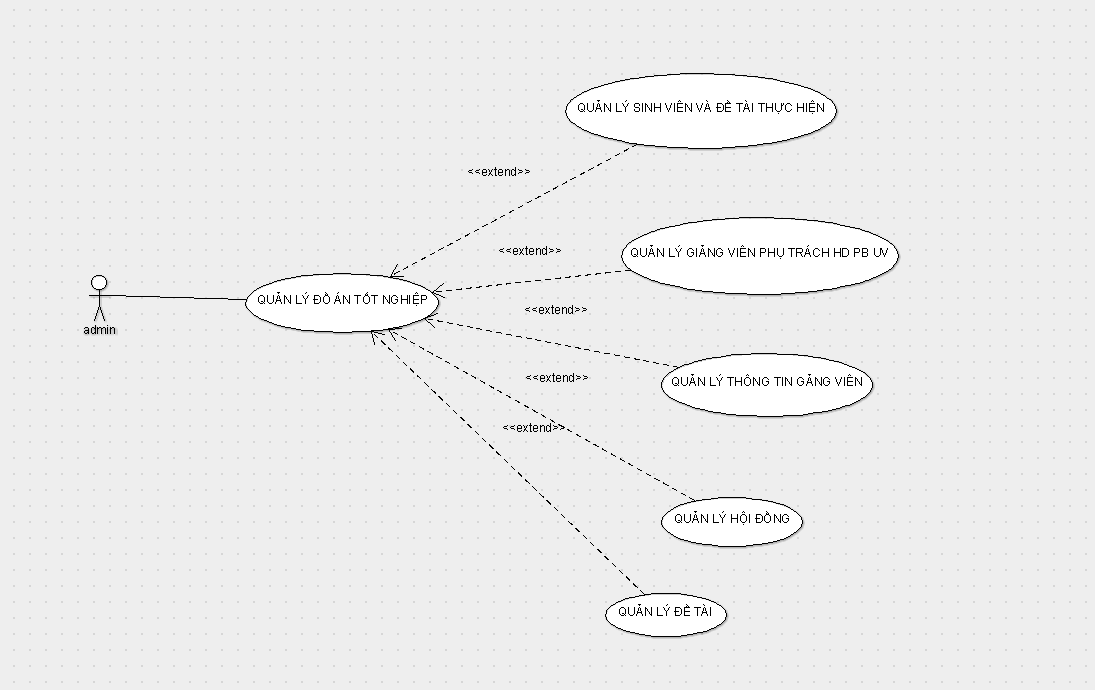
# **CHƯƠNG 9: MÔ HÌNH QUAN HỆ**

****

Hình: Mô hình quan hệ (Database Diagram)

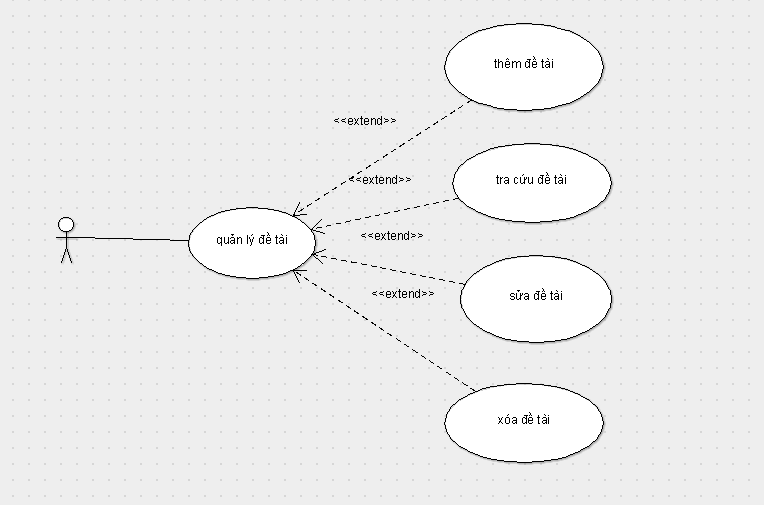
# **CHƯƠNG 10: USE CASE CÁC TÍNH NĂNG CƠ BẢN**

1. **USE CASE “QUẢN LÝ ĐỒ ÁN TỐT NGHIỆP”**

****

Hình: Sơ đồ Use Case "quản lý đồ án tốt nghiệp"

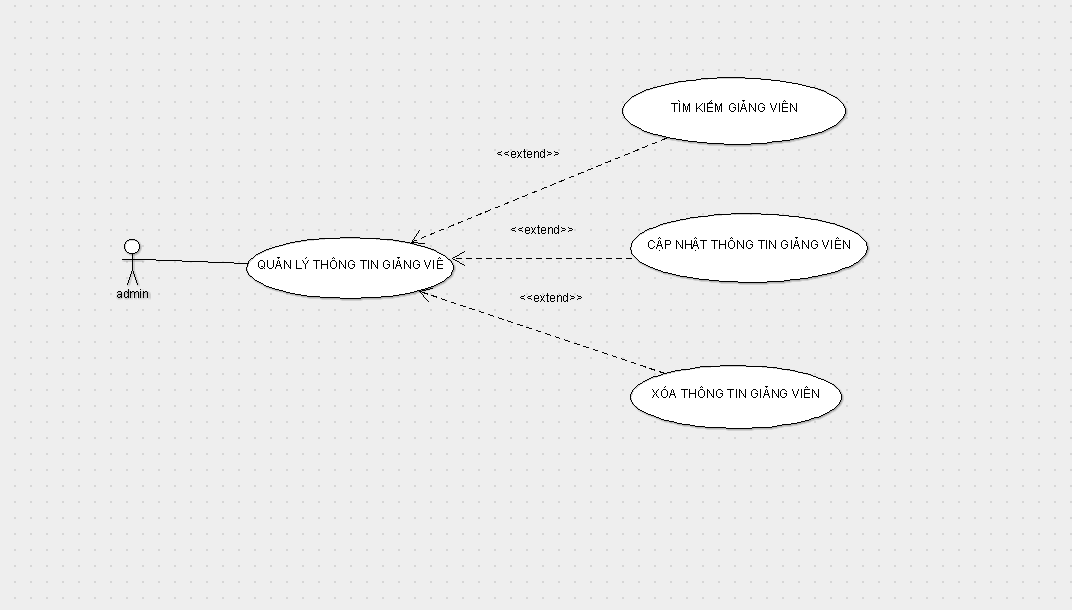
* 1. **USE CASE CON “QUẢN LÝ ĐỀ TÀI”**

****

Hình: Sơ đồ Use Case "Quản lý đề tài"

**Mô tả chi tiết gói usecase**

* **Use case “thêm đề tài”:**
* Tên use case: thêm đề tài.
* Mục đích: người sử dụng, quản lý dùng chức năng này để thêm các thông tin của đề tài vào cơ sở dữ liệu.
* Mô tả khái quát:
  + - 1. người quản lý mở hệ thống.
      2. Chọn menu đồ án tốt nghiệp trong trang quản trị vào đề tài chọn chức năng thêm mới.
      3. Khi người dùng/quản lý hoàn tất việc nhập thông tin đề tài, hệ thống lưu các thông tin từ form thêm mới vào cơ sở dữ liệu của hệ thống.
  1. **USE CASE “QUẢN LÝ THÔNG TIN GIẢNG VIÊN”**

****

Hình: Sơ đồ Use Case "Quản lý thông tin giảng viên"

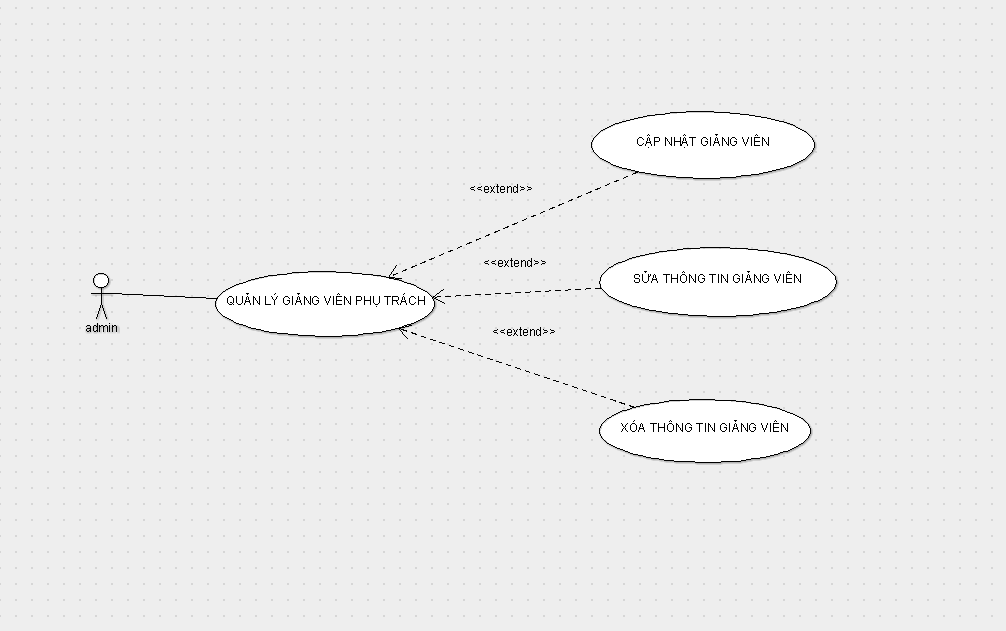
**Mô tả chi tiết gói use case:**

Tên use case: thêm giảng viên, sửa giảng viên, xóa giảng viên.

Mục đích: người quản lý sử dụng các chức năng này để thêm, sửa, xóa các thông tin của giảng viên trong cơ sở dữ liệu.

Mô tả khái quát:

1. Người quản lý vào hệ thống.
2. Chọn menu giảng viên trong trang quản trị và vào thông tin giảng viên chọn chức năng mong muốn.
3. Giao diện tương ứng được chọn vào hiển thị.
4. Khi người quản lý hoàn tất việc cập nhật thông tin giảng viên, hệ thống lưu các thông tin từ các form trên vào cơ sở dữ liệu của hệ thống.
   1. **USE CASE “QUẢN LÝ GIẢNG VIÊN PHỤ TRÁCH HD PB UV”**



Hình: Sơ đồ Use Case "Quản lý giảng viên phụ trách HD PB UV"

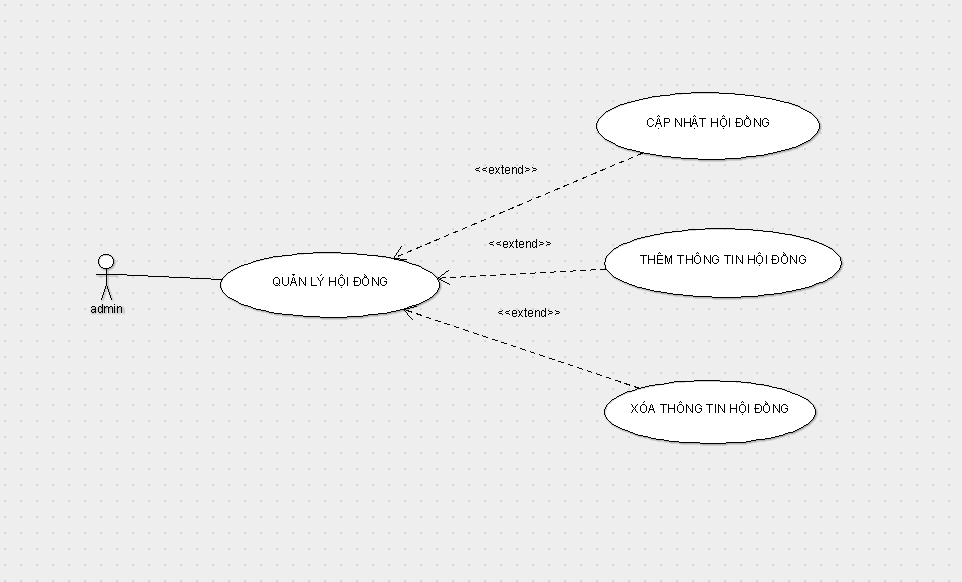
**Mô tả chi tiết gói use case:**

Tên use case: thêm giảng viên, sửa giảng viên, xóa giảng viên.

Mục đích: người quản lý sử dụng các chức năng này để thêm, sửa, xóa các thông tin phụ trách của giảng viên trong cơ sở dữ liệu.

Mô tả khái quát:

1. Người quản lý vào hệ thống.
2. Chọn menu giảng viên trong trang quản trị và vào thông tin giảng viên chọn chức năng mong muốn.
3. Giao diện tương ứng được chọn vào hiển thị.
4. Khi người quản lý hoàn tất việc cập nhật thông tin giảng viên, hệ thống lưu các thông tin từ các form trên vào cơ sở dữ liệu của hệ thống.
   1. **USE CASE “QUẢN LÝ HỘI ĐỒNG”**



Hình: Sơ đồ Use Case "Quản lý hội đồng"

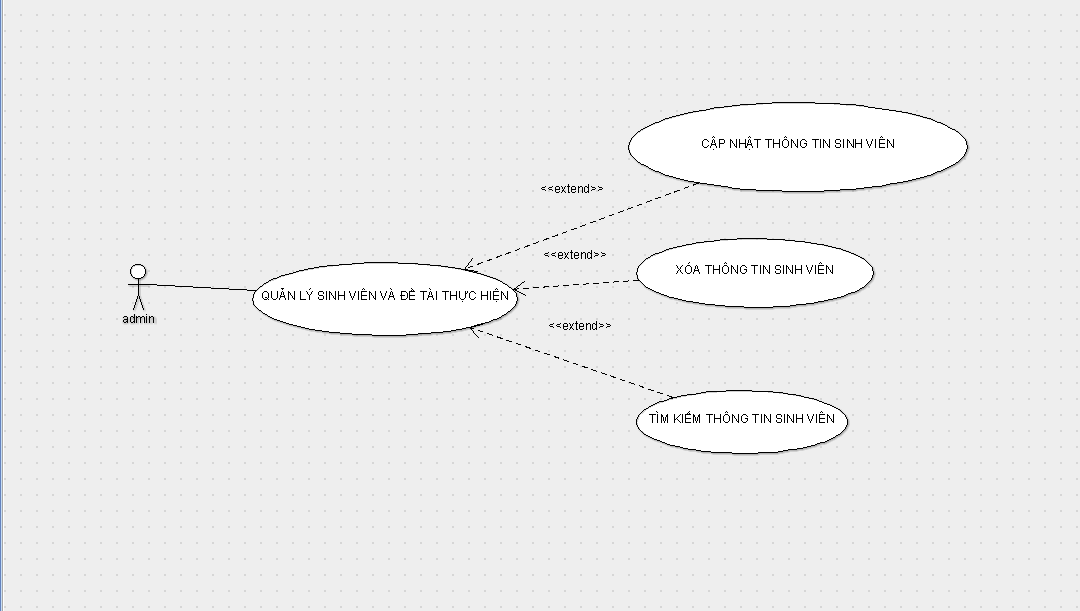
**Mô tả chi tiết gói use case:**

Tên use case: thêm thông tin hội đồng, cập nhật hội đồng, xóa thông tin hội đồng.

Mục đích: người quản lý sử dụng các chức năng này để thêm, sửa, xóa các thông tin hội đồng trong cơ sở dữ liệu.

Mô tả khái quát:

1. Người quản lý vào hệ thống.
2. Chọn menu giảng viên trong trang quản trị và vào quản lý hội đồng chọn chức năng mong muốn.
3. Giao diện tương ứng được chọn vào hiển thị.
4. Khi người quản lý hoàn tất việc cập nhật thông tin giảng viên, hệ thống lưu các thông tin từ các form trên vào cơ sở dữ liệu của hệ thống.
   1. **USE CASE “QUẢN LÝ SINH VIÊN VÀ ĐỀ TÀI THỰC HIỆN”**



Hình: Sơ đồ Use Case "Quản lý sinh viên và đề tài thực hiện"

**Mô tả chi tiết gói use case:**

Tên use case: cập nhật thông tin sinh viên, xóa thông tin sinh viên.

Mục đích: người quản lý sử dụng các chức năng này để sửa, xóa các thông tin sinh viên và đề tài thực hiện của sinh viên đó trong cơ sở dữ liệu.

Mô tả khái quát:

1. Người quản lý vào hệ thống.
2. Chọn menu giảng viên trong trang quản trị và vào quản lý sinh viên và đề tài thực hiện chọn chức năng mong muốn.
3. Giao diện tương ứng được chọn vào hiển thị.
4. Khi người quản lý hoàn tất việc cập nhật thông tin giảng viên, hệ thống lưu các thông tin từ các form trên vào cơ sở dữ liệu của hệ thống.

**Mô tả chi tiết gói use case:**

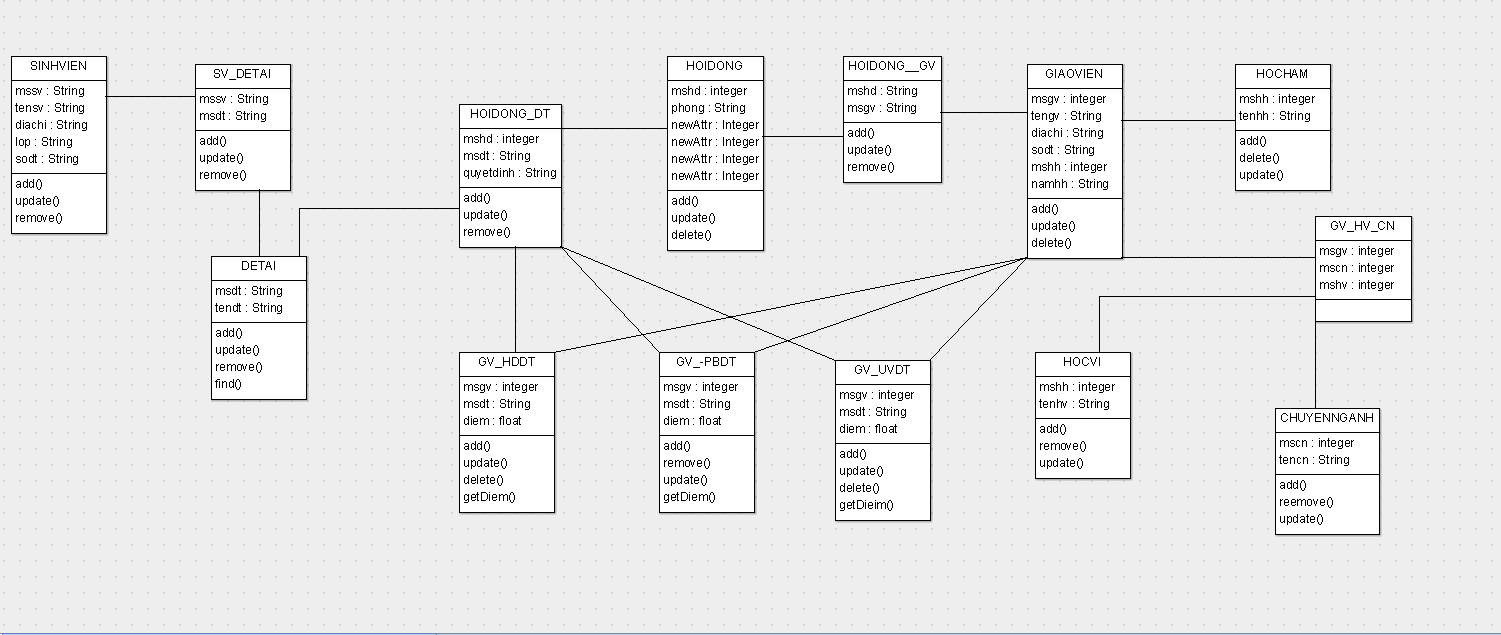
Tên use case: thêm thông tin hội đồng, cập nhật hội đồng, xóa thông tin hội đồng.

Mục đích: người quản lý sử dụng các chức năng này để thêm, sửa, xóa các thông tin hội đồng trong cơ sở dữ liệu.

Mô tả khái quát:

1. Người quản lý vào hệ thống.
2. Chọn menu giảng viên trong trang quản trị và vào quản lý hội đồng chọn chức năng mong muốn.
3. Giao diện tương ứng được chọn vào hiển thị.
4. Khi người quản lý hoàn tất việc cập nhật thông tin giảng viên, hệ thống lưu các thông tin từ các form trên vào cơ sở dữ liệu của hệ thống.

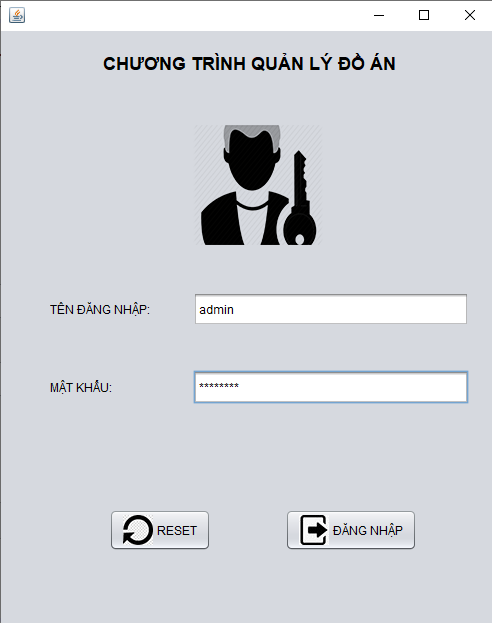
# **CHƯƠNG 11: CLASS DIAGRAM**

****

Hình: Mô hình Class Diagram

# **CHƯƠNG 12: FROM GIAO DIỆN**

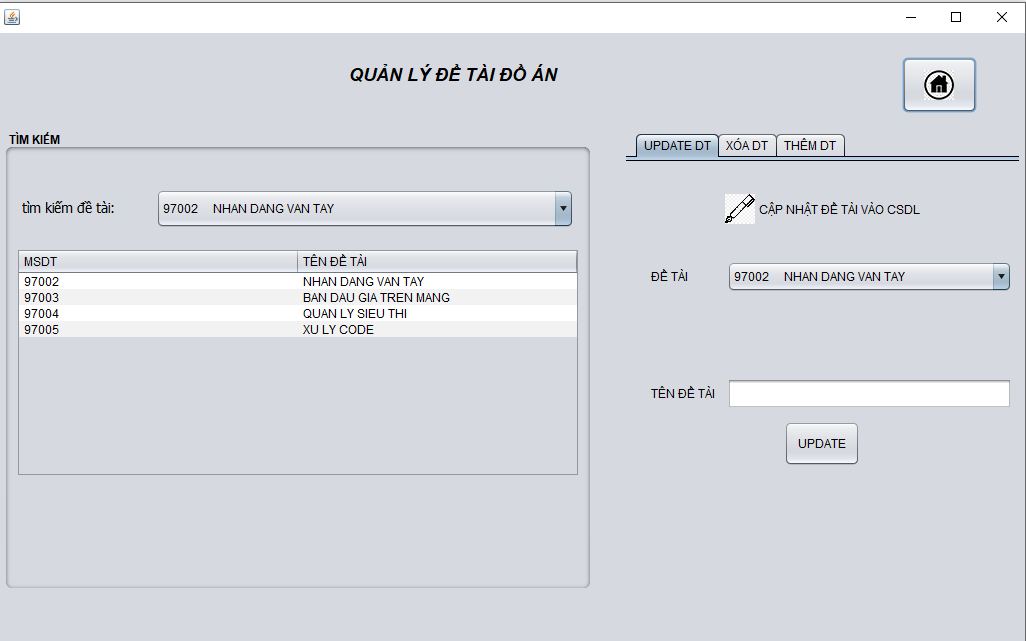
**Các form có trong chương trình:**

****

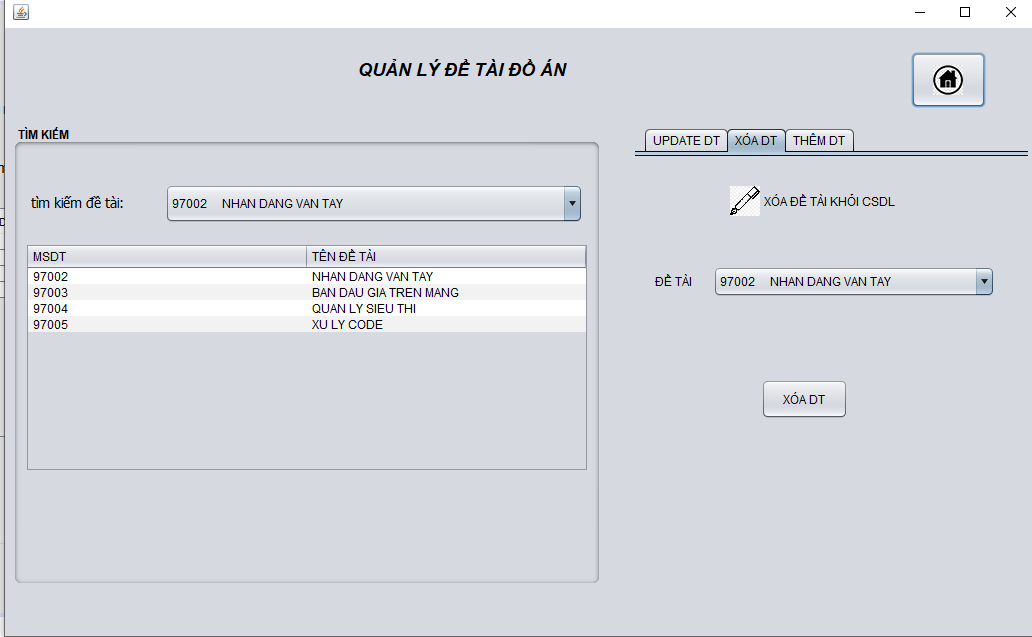
Hình: Form đăng nhập

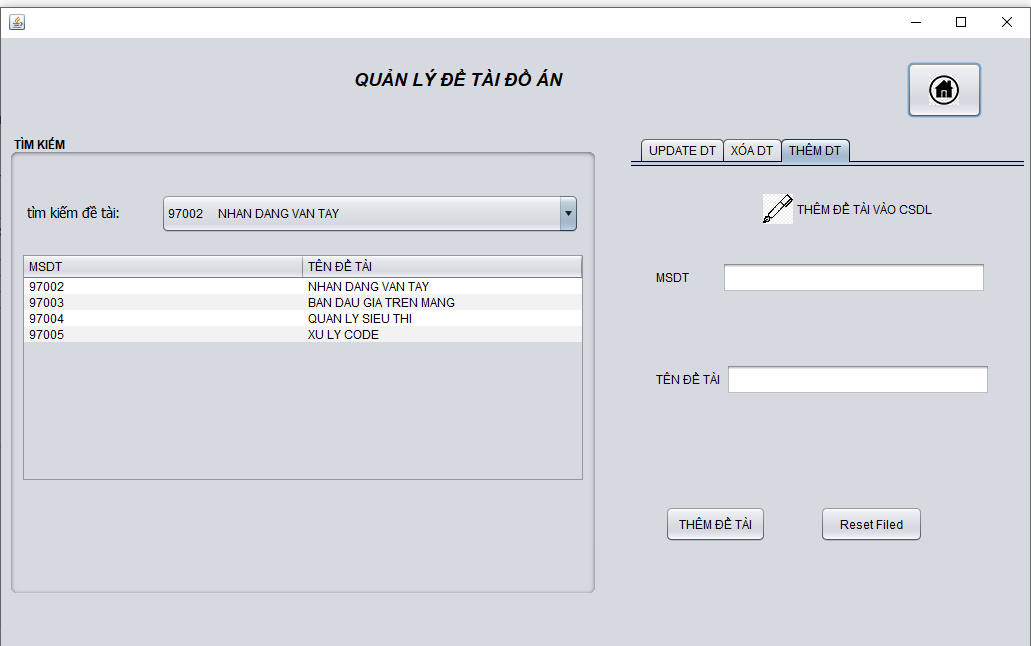
****

Hình: Form menu chức năng

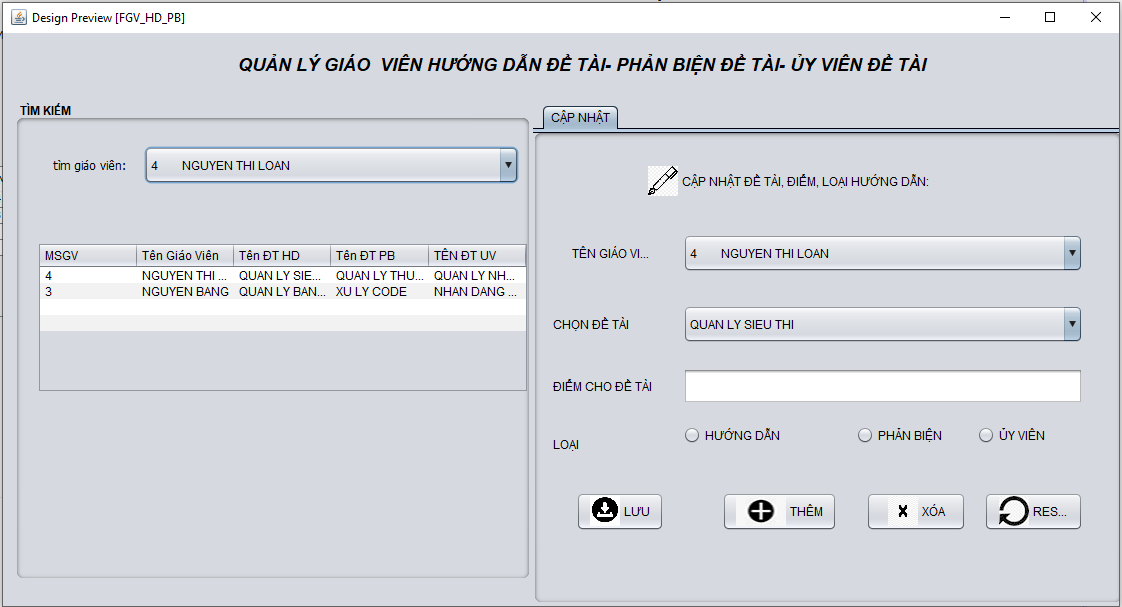
****

Hình: Form chức năng "Quản lý đề tài đồ án- cập nhật đề tài"

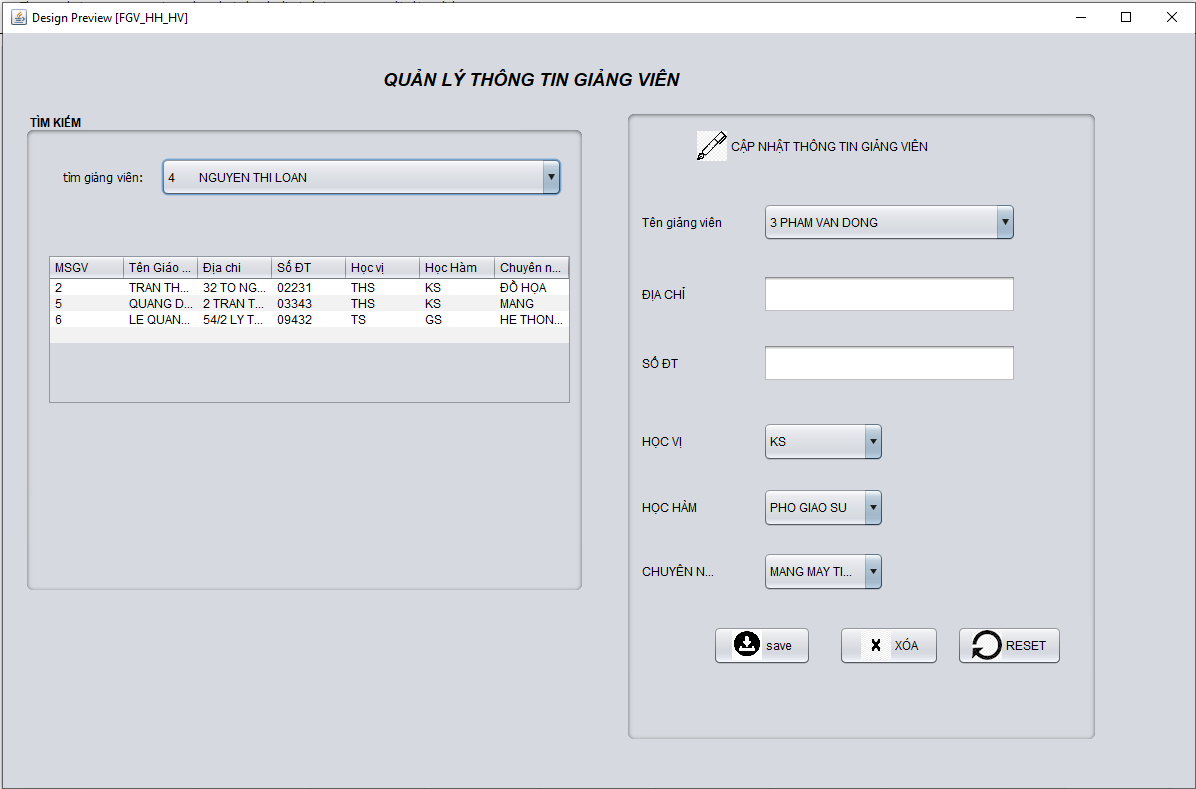
****Hình: Form chức năng "Quản lý đề tài đồ án- xóa đề tài"

****

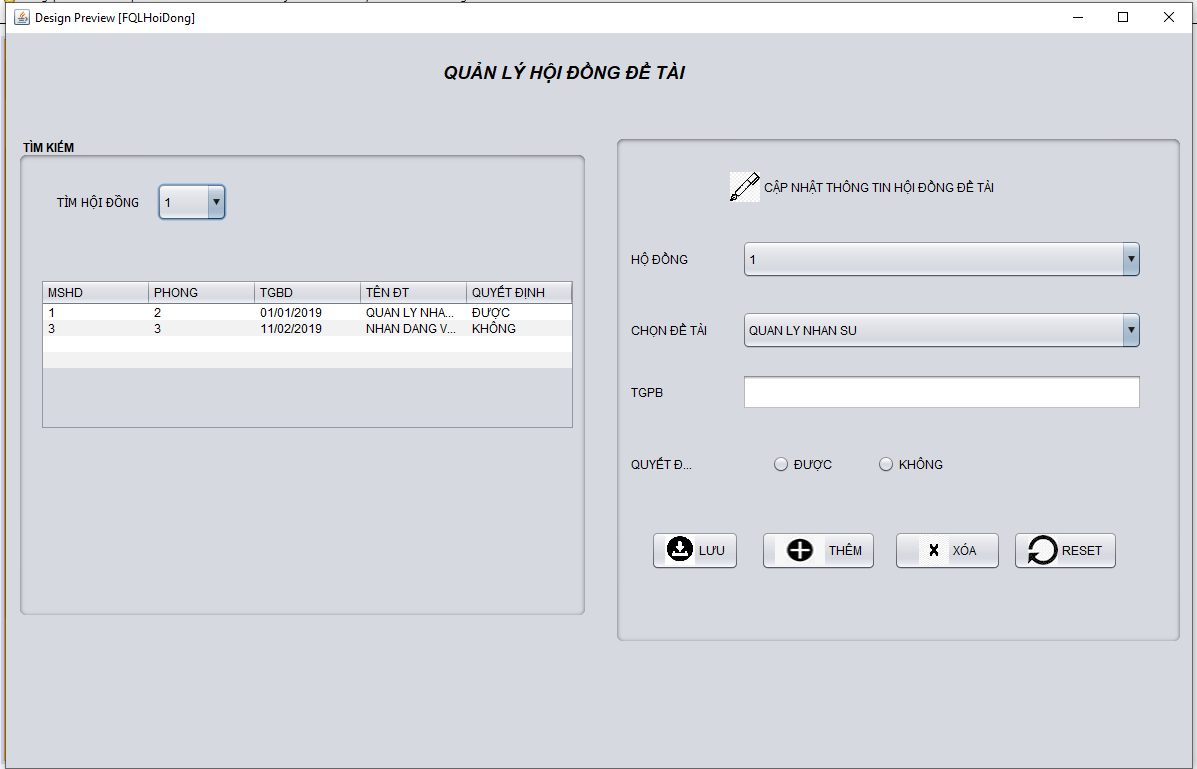
Hình: Form "Quản lý đề tài đồ án- Thêm đề tài"

****

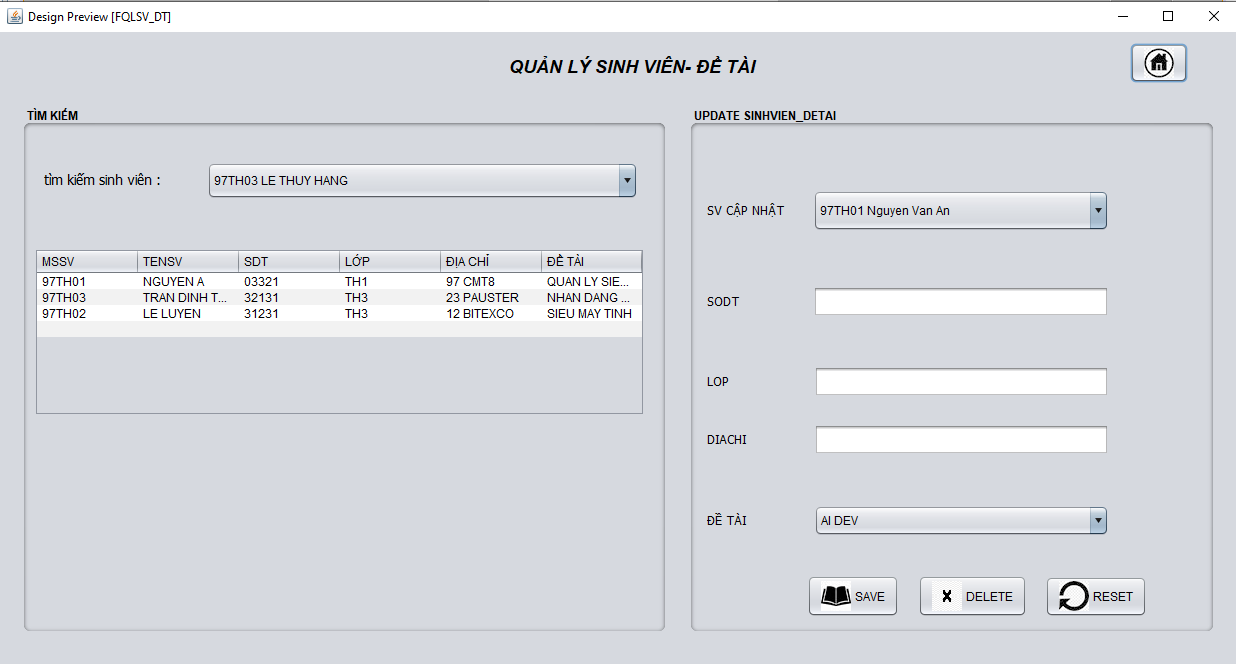
Hình: Form Quản lý giảng viên hướng dẫn đề tài- Phản biện đề tài- Ủy viên đề tài

****

Hình: Form quản lý thông tin giảng viên

****

Hình: Form quản lý hội đồng đề tài

****

Hình: Form Quản lý sinh viên- Đề tài

# **CHƯƠNG 13: CODE CHƯƠNG TRÌNH**

**Code form quản lý đề tài đồ án:**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package gui;

import conectdatabase.GetConnectionUtils;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JComboBox;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import model.DeTai;

/\*\*

\*

\* @author ASUS

\*/

public class FDeTaiDoAn extends javax.swing.JFrame {

private final ArrayList<DeTai> aCB\_DETAI;//khai bao toan cuc

/\*\*

\* Creates new form FQLDeTai

\*/

public FDeTaiDoAn() {

initComponents();

aCB\_DETAI=new ArrayList<>();

getShowData();

getShowDataComboxFilter(cbDT\_Filter);

getShowDataComboxUP(cbMSDT\_Update);

getShowDataComboxDEL(cbMSDT\_Delete);

}

private void getShowData() {

try {

aCB\_DETAI.clear();

//String connectionURL = "jdbc:sqlserver://localhost;sqlIstanceName=DESKTOP-SKTH6LI\\SQLEXPRESS;database=QUANLYDIEMVADETAITOTNGHIEP;userName=SA;password=123";

//Connection conn = DriverManager.getConnection(connectionURL);

Connection conn= GetConnectionUtils.getDB();//khai bao toan cuc

Statement st=conn.createStatement();//khai bao toan cuc

String sql = "select \* from DETAI";

ResultSet rs=st.executeQuery(sql);

while(rs.next()){

DeTai fr=new DeTai(rs.getString("MSDT"),rs.getString("TENDT"));

aCB\_DETAI.add(fr);

}

DefaultTableModel model = new DefaultTableModel();

Object[] columnsName = new Object[2];

columnsName[0] = "MSDT";

columnsName[1] = "TÊN ĐỀ TÀI";

model.setColumnIdentifiers(columnsName);

Object[] rowData = new Object[2];

for(int i = 0; i < aCB\_DETAI.size(); i++){

rowData[0] = aCB\_DETAI.get(i).getMSDT().trim();

rowData[1] = aCB\_DETAI.get(i).getTENDT().trim();

model.addRow(rowData);

}

jTDETAI.setModel(model);

conn.close();

st.close();

} catch (ClassNotFoundException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

} catch (SQLException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void getShowDataComboxFilter(JComboBox tenbien){

try {

aCB\_DETAI.clear();

tenbien.removeAllItems();

//Connection conn=ConnectiontionUtils.getMyConnection();;//khai bao toan cuc

Connection conn= GetConnectionUtils.getDB();

Statement st=conn.createStatement();//khai bao toan cuc

String sqlcheck="select \* from DETAI";

while(rs.next()){

DeTai fr=new DeTai(rs.getString("MSDT"), rs.getString("TENDT"));

aCB\_DETAI.add(fr);

}

for(int i = 0; i < aCB\_DETAI.size(); i++){

tenbien.addItem(aCB\_DETAI.get(i).toString());

}

} catch (SQLException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void getShowDataComboxUP(JComboBox tenbien){

try {

aCB\_DETAI.clear();

tenbien.removeAllItems();

Connection conn= GetConnectionUtils.getDB();

Statement st=conn.createStatement();//khai bao toan cuc

String sqlcheck="select \* from DETAI";

ResultSet rs=st.executeQuery(sqlcheck);

while(rs.next()){

DeTai fr=new DeTai(rs.getString("MSDT"), rs.getString("TENDT"));

aCB\_DETAI.add(fr);

}

for(int i = 0; i < aCB\_DETAI.size(); i++){

tenbien.addItem(aCB\_DETAI.get(i).toString());

}

} catch (SQLException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void getShowDataComboxDEL(JComboBox tenbien){

try {

aCB\_DETAI.clear();

tenbien.removeAllItems();

Connection conn= GetConnectionUtils.getDB();

Statement st=conn.createStatement();//khai bao toan cuc

String sqlcheck="select \* from DETAI";

ResultSet rs=st.executeQuery(sqlcheck);

while(rs.next()){

DeTai fr=new DeTai(rs.getString("MSDT"), rs.getString("TENDT"));

aCB\_DETAI.add(fr);

}

for(int i = 0; i < aCB\_DETAI.size(); i++){

tenbien.addItem(aCB\_DETAI.get(i).toString());

}

} catch (SQLException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void btnUpdateActionPerformed(java.awt.event.ActionEvent evt) {

if (txtTenDT\_Update.getText().trim().equals("")) {

lbTrongTen\_Update.setText("Không được bỏ trống tên đề tài!");

txtTenDT\_Update.grabFocus();

return;

}

//else if (cbMSDT\_Update.equals(Selec)==null);

// cbMSDT\_Update.grabFocus();

// return; }

else {

lbTrongTen\_Update.setText("");

try {

int vt=cbMSDT\_Update.getSelectedIndex();

String msdt=aCB\_DETAI.get(vt).getMSDT();

String tendt = txtTenDT\_Update.getText();

Connection conn= GetConnectionUtils.getDB();

//Connection conn=ConnectionUtils.getMyConnection();;

Statement st=conn.createStatement();

String sqlcheck="select \* from DETAI where MSDT='"+msdt+"'";

ResultSet rs=st.executeQuery(sqlcheck);

int count=0;

while(rs.next()){

count++;

}

String sqlinsert="Update DETAI set TENDT='"+tendt+"' where MSDT='"+msdt+"'";

int rownum=st.executeUpdate(sqlinsert);

getShowData();

getShowDataComboxFilter(cbDT\_Filter);

getShowDataComboxUP(cbMSDT\_Update);

getShowDataComboxDEL(cbMSDT\_Delete);

if(count!=0){

JOptionPane.showMessageDialog(null,"Đã cập nhật đề tài thành công!");

}else{

if(rownum==0){

JOptionPane.showMessageDialog(null,"Insert không thành công");

}

}

} catch (SQLException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(FDeTaiDoAn.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private void btnXoaDTActionPerformed(java.awt.event.ActionEvent evt) {

try {

int vt=cbMSDT\_Delete.getSelectedIndex();

String msdt=aCB\_DETAI.get(vt).getMSDT();

Connection conn= GetConnectionUtils.getDB();

Statement st=conn.createStatement();

String sqlcheck="select \* from DETAI where MSDT='"+msdt+"'";

ResultSet rs=st.executeQuery(sqlcheck);

int count=0;

while(rs.next()){

count++;

}

String sqlinsert="DELETE FROM DETAI where MSDT='"+msdt+"'";

int rownum=st.executeUpdate(sqlinsert);

getShowData();

getShowDataComboxFilter(cbDT\_Filter);

getShowDataComboxUP(cbMSDT\_Update);

getShowDataComboxDEL(cbMSDT\_Delete);

if(count!=0){

JOptionPane.showMessageDialog(null,"Đã xóa đề tài thành công!");

}else{

if(rownum==0){

JOptionPane.showMessageDialog(null,"Xóa không thành công!");

}

}

} catch (SQLException ex) {

} catch (ClassNotFoundException ex) {

}

}

private void btnThemDTActionPerformed(java.awt.event.ActionEvent evt) {

if (txtTenDT\_Add.getText().trim().equals("") ||txtMSDT\_Add.getText().trim().equals("")) {

lbTrongTen\_Them.setText("Không được bỏ trống mã đề tài hoặc tên đề tài!");

txtTenDT\_Add.grabFocus();

txtMSDT\_Add.grabFocus();

return;

}

else {

lbTrongTen\_Them.setText("");

lbTrongMSDT\_Them.setText("");

}

try {

String msdt = txtMSDT\_Add.getText();

String tendt = txtTenDT\_Add.getText();

Connection conn= GetConnectionUtils.getDB();

Statement st=conn.createStatement();

String sqlcheck="select MSDT from DETAI";

ResultSet rs=st.executeQuery(sqlcheck);

int count=0;

while(rs.next()){

count++;

}

if(count!=0){

JOptionPane.showMessageDialog(null,"Đã trùng mã số đề tài!");

}if(count==0){

String sqlinsert="INSERT INTO DETAI VALUES('"+msdt+"','"+tendt+"')";

int rownum=st.executeUpdate(sqlinsert);

getShowData();

getShowDataComboxFilter(cbDT\_Filter);

getShowDataComboxUP(cbMSDT\_Update);

getShowDataComboxDEL(cbMSDT\_Delete);

if(rownum==0){

JOptionPane.showMessageDialog(null,"Thêm mới đề tài không thành công!");

}

else{

JOptionPane.showMessageDialog(null,"Thêm mới đề tài thành công!");

}

}

} catch (SQLException | ClassNotFoundException ex) {

}

}

private void btnResetActionPerformed(java.awt.event.ActionEvent evt) {

txtMSDT\_Add.setText("");

txtTenDT\_Add.setText("");

}

private void btnHomeActionPerformed(java.awt.event.ActionEvent evt) {

FChucNang f = new FChucNang();

f.setVisible(true);

this.dispose();

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(FDeTaiDoAn.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(FDeTaiDoAn.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(FDeTaiDoAn.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(FDeTaiDoAn.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new FDeTaiDoAn().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton btnHome;

private javax.swing.JButton btnReset;

private javax.swing.JButton btnThemDT;

private javax.swing.JButton btnUpdate;

private javax.swing.JButton btnXoaDT;

private javax.swing.JComboBox cbDT\_Filter;

private javax.swing.JComboBox cbMSDT\_Delete;

private javax.swing.JComboBox cbMSDT\_Update;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel11;

private javax.swing.JLabel jLabel12;

private javax.swing.JLabel jLabel13;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JPanel jPanel4;

private javax.swing.JPanel jPanel5;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTable jTDETAI;

private javax.swing.JTabbedPane jTabbedPane1;

private javax.swing.JLabel lbTongMSDT\_Up;

private javax.swing.JLabel lbTrongMSDT\_Del;

private javax.swing.JLabel lbTrongMSDT\_Them;

private javax.swing.JLabel lbTrongTenDT\_Del;

private javax.swing.JLabel lbTrongTen\_Them;

private javax.swing.JLabel lbTrongTen\_Update;

private javax.swing.JTextField txtMSDT\_Add;

private javax.swing.JTextField txtTenDT\_Add;

private javax.swing.JTextField txtTenDT\_Update;

// End of variables declaration

}

**KẾT LUẬN**

Qua những vấn đề đã trình bày trong báo cáo, có thể thấy rằng việc tin học hóa trong quá trình quản lý đem lại những lợi ích cực kì to lớn. Nó giúp cho công việc quản lý đồ án tốt nghiệp sinh viên dễ dàng hơn, giảm thiểu được những mất mát và hư hỏng dữ liệu trong quá trình quản lý.

Hệ thống đã giải quyết được các công việc sau:

* Tìm hiểu được bài toán quản lý đồ án sinh viên.
* Lập được bảng phân tích và thiết kế hệ thống quản lý đồ án tốt nghiệp sinh viên của khoa. Phần mềm bao gồm các chức năng: Quản lý đề tài đồ án, quản lý thông tin giảng viên, quản lý thông tin hội đồng, quản lý thông tin sinh viên và đề tài thực hiên.

Những hạn chế của chương trình:

Do thời gian thực hiện và trình độ hiểu biết còn hạn chế nên chương trình mang lại hiệu quả chưa cao. Hệ thống còn chưa hoàn chỉnh vì thiếu nhiều tính năng. Đây sẽ là hướng phát triển sau này của đề tài xxx