

**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP.HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**\*\*\*\*\* ᴥ⸙ᴥ \*\*\*\*\***

**TÀI LIỆU ĐẶC TẢ YÊU CẦU PHẦN MỀM**

**(Software Requirements Specificatio – SRS)**

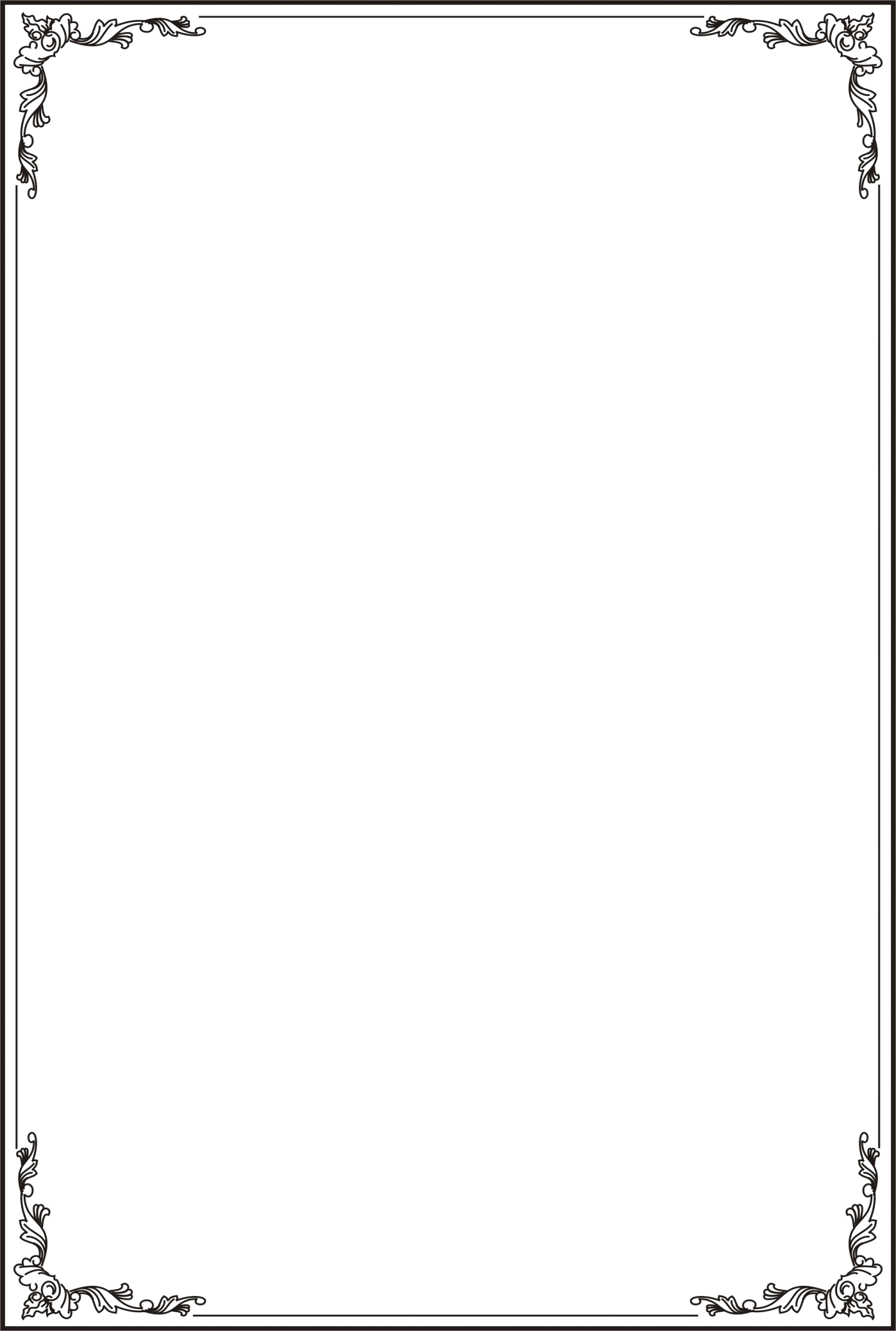
**MÔN: KIỂM ĐỊNH PHẦN MỀM**

**ĐỀ TÀI : HỆ THỐNG TIẾN TRÌNH  
KHÓA LUẬN TỐT NGHIỆP**

**Giảng viên hướng dẫn : ThS. Ngô Minh Anh Thư**

**TP. HCM, NĂM 2025**

**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP.HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**\*\*\*\*\* ᴥ⸙ᴥ \*\*\*\*\***

**TÀI LIỆU ĐẶC TẢ YÊU CẦU PHẦN MỀM**

**(Software Requirements Specificatio – SRS)**

**MÔN: KIỂM ĐỊNH PHẦN MỀM**

**ĐỀ TÀI : HỆ THỐNG TIẾN TRÌNH  
KHÓA LUẬN TỐT NGHIỆP**

**Giảng viên hướng dẫn : ThS. Ngô Minh Anh Thư**

**Nhóm sinh viên thực hiện:**

2001224537 – Trần Anh Tuấn

2001220623 – Hồ Thái Dinh

2044226229 – Võ Ngọc Nguyên Chương

2001221004 – Nguyễn Bá Đức

**TP. HCM, NĂM 2025**

**LỜI CAM ĐOAN**

Chúng em xin cam đoan rằng tài liệu **"Xây dựng hệ thống tiến trình khóa luận tốt nghiệp"** là kết quả của quá trình nghiên cứu và thực hiện một cách nghiêm túc, có hệ thống, dựa trên sự nỗ lực của nhóm. Tài liệu này được thực hiện với tinh thần trách nhiệm cao, nhằm mục đích cung cấp cái nhìn tổng quan về kiến trúc, chức năng và vai trò của hệ thống trong việc hỗ trợ quản lý, theo dõi và đánh giá tiến trình thực hiện khóa luận tốt nghiệp một cách hiệu quả và khoa học.

Chúng em đã tuân thủ đầy đủ các quy định học thuật trong việc thu thập, phân tích tài liệu và xử lý thông tin. Mọi nội dung, số liệu và ví dụ minh họa đều được kiểm chứng từ các nguồn đáng tin cậy, đảm bảo tính chính xác, khách quan và minh bạch. Chúng em khẳng định tài liệu hoàn toàn trung thực, không sao chép từ bất kỳ công trình nghiên cứu nào khác.

Thông qua tài liệu này, chúng em mong muốn góp phần làm rõ tầm quan trọng của việc ứng dụng công nghệ thông tin trong giáo dục, đặc biệt là trong công tác quản lý khóa luận tốt nghiệp, đồng thời cung cấp kiến thức nền tảng để xây dựng và triển khai hệ thống quản lý hiện đại, góp phần nâng cao hiệu quả quản lý tại các cơ sở đào tạo.

Thành phố Hồ Chí Minh, tháng 03 năm 2025

Sinh viên thực hiện

(Ký và ghi rõ họ tên)

Trần Anh Tuấn (Nhóm trưởng)

**LỜI CẢM ƠN**

Trước hết, em xin gửi lời cảm ơn chân thành nhất đến quý thầy, cô giáo trong khoa Công nghệ thông tin của trường Đại học Công Thương Thành phố Hồ Chí Minh đã tận tình giảng dạy và truyền đạt cho em những kiến thức quý báu trong suốt thời gian qua. Đặc biệt, em xin gửi lời cảm ơn sâu sắc nhất đến cô, người đã trực tiếp hướng dẫn, giúp đỡ và góp ý cho em trong suốt quá trình thực hiện.

Sự tận tâm, nhiệt tình và chuyên nghiệp của cô đã giúp em rất nhiều trong việc hoàn thành đề tài này. Chúng em xin chân thành cảm ơn tất cả những người đã giúp đỡ em trong quá trình thực hiện đề tài này.

Thành phố Hồ Chí Minh, tháng 03 năm 2025

Sinh viên thực hiện

(Ký và ghi rõ họ tên)

Trần Anh Tuấn (Nhóm trưởng)

**TÓM TẮT**

Bố cục của đồ án bao gồm 5 chương:

Chương 1.

* Tổng quan - Giới thiệu về đề tài, mục tiêu và phạm vi của đề tài.

Chương 2.

* Cơ sở lí thuyết - Tìm hiểu về ASP.NET Core Web API, Next.js và kiến trúc Clean Architecture.

Chương 3.

* Phân tích hệ thống - Phân tích các nghiệp vụ, quy trình hoạt động có trong hệ thống.
* Thiết kế hệ thống - Thiết kế cơ sở dữ liệu, giao diện người dùng và trang quản trị.

Chương 4 .

* Thiết kế giao diện người dùng và trang quản trị.

Chương 5.

* Kết luận - Tổng kết lại kết quả đạt được trong thời gian thực hiện, nghiên cứu khóa luận và các mặt hạn chế cần cải thiệ

# MỤC LỤC

[MỤC LỤC i](#_Toc200353054)

[DANH MỤC CÁC KÝ HIỆU, CHỮ VIẾT TẮT iii](#_Toc200353055)

[DANH SÁCH CÁC BIỂU ĐỒ, ĐỒ THỊ, SƠ ĐỒ, HÌNH ẢNH iv](#_Toc200353056)

[CHƯƠNG 1 TỔNG QUAN ĐỀ TÀI 1](#_Toc200353057)

[1.1. Giới thiệu 1](#_Toc200353058)

[1.2. Mục tiêu nghiên cứu 2](#_Toc200353059)

[1.3. Phạm vi nghiên cứu 2](#_Toc200353060)

[1.4. Nội dung nghiên cứu 2](#_Toc200353061)

[CHƯƠNG 2 CƠ SỞ LÝ THUYẾT 3](#_Toc200353062)

[2.1. ASP.NET Core Web API 3](#_Toc200353063)

[2.1.1 Khái niệm ASP.NET Core Web Api 3](#_Toc200353064)

[2.1.2 Lợi ích khi sử dụng ASP.NET Core Web API 3](#_Toc200353065)

[2.2. Next.js 4](#_Toc200353066)

[2.2.1 Khái niệm Next.js 4](#_Toc200353067)

[2.2.2 Lợi ích khi sử dụng Next.js 4](#_Toc200353068)

[2.3. Kiến trúc Clean Architecture 4](#_Toc200353069)

[2.3.1 Khái niệm về kiến trúc Clean Architecture 4](#_Toc200353070)

[2.3.2 Ứng dụng kiến trúc Clean Architecture 5](#_Toc200353071)

[CHƯƠNG 3 PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 7](#_Toc200353072)

[3.1. Giới thiệu 7](#_Toc200353073)

[3.2. Các qui trình nghiệp vụ 7](#_Toc200353074)

[3.2.1 Quy trình quản lý sinh viên tham gia khóa luận và phân công nhiệm vụ ra đề 7](#_Toc200353075)

[3.2.2 Quy trình ra đề và kiểm duyệt đề tài của giảng viên 7](#_Toc200353076)

[3.2.3 Quy trình đăng ký đề tài khóa luận tốt nghiệp của sinh viên 8](#_Toc200353077)

[3.2.4 Quy trình quản lý tiến độ thực hiện đề tài của giảng viên hướng dẫn 8](#_Toc200353078)

[3.2.5 Quy trình chấm điểm cho đề tài cho sinh viên 9](#_Toc200353079)

[3.2.6 Quy trình thống kê kết quả của khoa CNTT 9](#_Toc200353080)

[3.3. Xác định yêu cầu 10](#_Toc200353081)

[3.3.1 Yêu cầu chức năng 10](#_Toc200353082)

[3.3.2 Yêu cầu phi chức năng 10](#_Toc200353083)

[3.3.3 Tiêu chí chấp nhận của người dùng 10](#_Toc200353084)

[3.4. Biểu đồ Use case : 12](#_Toc200353085)

[3.4.1 Phân công nhiệm vụ ra đề 12](#_Toc200353086)

[3.4.2 Quản lý đề xuất và duyệt đề tài 13](#_Toc200353087)

[3.4.3 Đăng ký đề tài Khóa luận 13](#_Toc200353088)

[3.4.4 Quản lý tiến độ thực hiện đề tài 14](#_Toc200353089)

[3.4.5 Chấm điểm Khóa luận 14](#_Toc200353090)

[3.4.6 Thống kê và công bố kết quả 15](#_Toc200353091)

[3.5. Quy trình nghiệp vụ 15](#_Toc200353092)

[3.5.1 Quy trình quản lý sinh viên tham gia khóa luận và phân công nhiệm vụ ra đề 15](#_Toc200353093)

[3.5.2 Quy trình ra đề và kiểm duyệt đề tài của giảng viên 16](#_Toc200353094)

[3.5.3 Quy trình đăng ký đề tài khóa luận tốt nghiệp của sinh viên 17](#_Toc200353095)

[3.5.4 Quy trình quản lý tiến độ thực hiện đề tài của giảng viên hướng dẫn 18](#_Toc200353096)

[3.5.5 Quy trình chấm điểm cho đề tài của sinh viên 19](#_Toc200353097)

[3.5.6 Quy trình thống kê kết quả của khoa CNTT 20](#_Toc200353098)

[3.5.7 Quy trình thành đưa đề tài ra hội đồng 21](#_Toc200353099)

[KẾT LUẬN 23](#_Toc200353100)

[TÀI LIỆU THAM KHẢO 24](#_Toc200353101)

**DANH MỤC CÁC KÝ HIỆU, CHỮ VIẾT TẮT**

|  |  |
| --- | --- |
| **Từ viết tắt** | **Nghĩa** |
| CNTT | Công nghệ thông tin |
| HUIT | Ho Chi Minh City University of Industry and Trade |
| GVHD | Giảng viên hướng dẫn |
| ASP | Active Server Pages |
| .NET | Network Enabled Technologies |
| API | Application Programming Interface |
| SEO | Search Engine Optimization |

# DANH SÁCH CÁC BIỂU ĐỒ, ĐỒ THỊ, SƠ ĐỒ, HÌNH ẢNH

[Hình 3.6 Sơ đồ hoạt động qui trình quản lý sinh viên tham gia khóa luận và tính số lượng đề tài 16](#_Toc195997011)

[Hình 3.7 Sơ đồ hoạt động quy trình ra đề và kiểm duyệt đề tài của giảng viên 17](#_Toc195997012)

[Hình 3.8 Sơ đồ hoạt động quy trình đăng ký đề tài khóa luận tốt nghiệp của sinh viên 18](#_Toc195997013)

[Hình 3.9 Sơ đồ hoạt động quy trình quản lý tiến độ thực hiện đề tài của GVHD 19](#_Toc195997014)

[Hình 3.10 Sơ đồ hoạt động quy trình chấm điểm cho đề tài của sinh viên 20](#_Toc195997015)

[Hình 3.11 Sơ đồ hoạt động quy trình thống kê kết quả của khoa CNTT 21](#_Toc195997016)

# TỔNG QUAN ĐỀ TÀI

## Giới thiệu

Hiện tại việc quản lý danh sách cũng như việc đăng ký khóa luận tốt nghiệp của khoa Công nghệ thông tin đang được thực hiện thủ công, chưa có phần mềm nào giúp thực hiện quản lý cũng như đăng ký khóa luận. Điều này dẫn tới gặp rất nhiều khó khăn trong vấn đề quản lý danh sách khóa luận của người phụ trách và đăng ký khóa luận của sinh viên.

Ở khoa CNTT của HUIT, khi tới thời gian đăng ký khóa luận tốt nghiệp thì danh sách sinh viên sẽ được đăng tải trên website của khoa cùng với danh sách các đề tài khóa luận tốt nghiệp. Mỗi đề tài có một link google forms để sinh viên đăng ký, mỗi đề tài chỉ cho một sinh viên đại diện nhóm đăng ký. Điều này đã gây khó khăn cho quá trình đăng ký đề tài của sinh viên vì khi chọn được một đề tài sinh viên sẽ thực hiện điền form nhưng không biết được rằng đề tài đã có nhóm sinh viên nào khác đăng ký hay chưa làm rất tốn thời gian. Vừa qua, học kì 1 năm học 2022-2023, trong quá trình đăng ký đề tài khóa luận tốt nghiệp tại khoa CNTT các nhóm sinh viên đã gặp một vấn đề là nhiều nhóm cùng đăng ký một đề tài do forms bị lỗi. Điều này không chỉ làm mất thời gian của sinh viên mà mất thời gian của người phụ trách và gây khó khăn trong quá trình quản lý, rà soát việc đăng ký đề tài khóa luận của sinh viên.

Trước tình hình hiện tại, việc xây dựng một hệ thống xử lý tiến trình khóa luận tốt nghiệp của sinh viên tại khoa CNTT tại HUIT là hết sức cần thiết. Điều này có thể giải quyết vấn đề tránh trùng lặp đề tài, mất thời gian đăng ký khóa luận tốt nghiệp của các sinh viên cũng như giúp người phụ trách quản lý được danh sách khóa luận dễ dàng. Ngoài ra, việc xây dựng hệ thống xử lý tiến trình khóa luận tốt nghiệp cũng giúp cho các giảng viên có thể kiểm tra, giám sát được danh sách đề tài của mình và các nhóm sinh viên đăng ký. Giúp sinh viên có thể nộp bài ngay trên hệ thống mà không cần phải thông qua các hình thức trung gian khác, giúp giảng viên có thể đánh giá, nhận xét, chấm điểm trực tiếp ngay trên hệ thống đó.

## Mục tiêu nghiên cứu

* Xây dựng hệ thống giúp khoa CNTT quản lý danh sách sinh viên đủ điều kiện đăng ký khóa luận tốt nghiệp, danh sách đề tài, GVHD, GVPB.
* Cho phép giảng viên ra đề và nhận được sự nhận xét, phê duyệt từ trưởng bộ môn, trưởng khoa để đề tài phù hợp với chương trình và thực tiễn nhất.
* Cho phép sinh viên trong đợt tham gia có thể tự do lựa chọn thành viên và lập nhóm. Đăng ký đề tài khóa luận tốt nghiệp theo nhóm đã lập, nộp kết quả báo cáo theo tiến độ và nhận góp ý từ GVHD để hoàn thiện tốt hơn.
* Cho phép giảng viên tạo ra yêu cầu theo tiến độ của đề tài; theo dõi, nhận xét, góp ý để sinh viên hoàn thành tốt đề tài và chấm điểm trên hệ thống.
* Hệ thống giúp kết xuất dữ liệu ra excel.
* Sản phẩm hướng đến đối tượng là sinh viên đủ điều kiện làm khóa luận tốt nghiệp, giảng viên, giáo vụ trong khoa CNTT.
* Hệ thống có chức năng cơ bản, dễ thao tác, không cần cài đặt mà có thể sử dụng trực tiếp qua website.

## Phạm vi nghiên cứu

*Phạm vi kỹ thuật*: ASP.NET Core Web API và Next.js, kiến trúc Clean Architecture

## Nội dung nghiên cứu

Tìm hiểu tổng quan về đề tài và cơ sở lý thuyết có liên quan như: công nghệ, kiến trúc, thuật toán.

Xác định và trình bày yêu cầu cụ thể cho việc xây dựng hệ thống xử lý tiến trình khóa luận tốt nghiệp bao gồm nghiệp vụ và các chức năng cơ bản.

Xây dựng hệ thống dựa trên công nghệ, kiến trúc và cơ sở lý thuyết đã tìm hiểu.

# CƠ SỞ LÝ THUYẾT

## ASP.NET Core Web API

### Khái niệm ASP.NET Core Web Api

ASP.NET Core Web API là một framework hỗ trợ cho việc xây dựng một trang dịch vụ web cũng như HTTP-based APIs bằng cách sử dụng ASP.NET Core. Nó là một bộ phận nhỏ trong hệ thống ASP.NET Core – một mã nguồn mở và hỗ trợ đa hệ điều hành như Windows, macOS, Linux – được phát triển bởi Microsoft trong việc xây dựng một trang web dịch vụ hiện đại, khả năng mở rộng cao [1].

### Lợi ích khi sử dụng ASP.NET Core Web API

Vì là mã nguồn mở nên đầu tiên ASP.NET Core Web API miễn phí để sử dụng, điều này hỗ trợ cho người chưa có nhiều kinh phí để tiếp cận như học sinh, sinh viên hoặc doanh nghiệp startup còn non trẻ.

Ngoài ra ASP.NET Core Web API còn có nhiều lợi ích khác như:

* Hỗ trợ trên nhiều nền tảng khác nhau như Windows, Linux, macOS: Điều này giúp cho lập trình viêc có thể xây dựng và chạy Web APIs trên nhiều hệ điều hành khác nhau.
* Được module hóa và giảm tải trọng của phần mềm: ASP.NET Core Web API cho phép chọn lựa những module cần thiết để cài đặt vào phần mềm giúp giảm tải những module không cần thiết.
* Hỗ trợ nhiều dạng data: ASP.NET Core Web API hỗ trợ nhiều dạng data khác nhau như JSON hoặc XML,… điều này giúp làm việc linh hoạt hơn giữa những khách hàng
* Xác thực và phân quyền: ASP Core Web API hỗ trợ xác thực và phân quyền bằng Oauth, JWT (JSON Web Tokens)

ASP.NET Core Web API được sử dụng trong việc xây dựng các API theo chuẩn RESTful có thể cung cấp dịch vụ cũng như dữ liệu thông qua giao thức HTTP. Nó thích hợp cho việc xây dựng những dịch vụ ở phía back-end cho web, app.

## Next.js

### Khái niệm Next.js

Next.js là một framework mạnh mẽ của React có thể hỗ trợ trong việc xây dựng phần mềm website full-stack. Nó cung cấp một lượng tính năng và tối ưu hóa, điều làm cho nó trở thành lựa chọn phổ biến cho lập trình viên.

### Lợi ích khi sử dụng Next.js

Next.js có nhiều lợi ích như:

* Render phía server (Server-side rendering): Next.js có thể render những React component ở phía server, điều mà có thể cải thiện hiệu năng cũng như SEO
* Trình tạo trang tĩnh (Static site generation): Next.js có thể tạo ra trang HTML tĩnh từ React components giúp cho hiển thị lên trang web mà không còn đến một server. Điều này cũng có thể giúp cải thiện hiệu năng và giảm thiểu chi phí.
* Tự động routing và chia tách code: Next.js tự động hóa việc configures routing cho phần mềm và chia tách code thành những bundles – điều này có thể cải thiện thời gian load.
* Nhìn chung, Next.js là một framework React mạnh mẽ giúp xây dựng một trang web có tính tương tác, hiệu năng cao. Nó là một sự lựa chọn tốt cả cho lập trình viên cũng như đội nhóm lớp.

## Kiến trúc Clean Architecture

### Khái niệm về kiến trúc Clean Architecture

Clean architecture là một cách tiếp cận thiết kế phần mềm tách biệt business logic cốt lõi của ứng dụng khỏi các phụ thuộc bên ngoài. Điều này làm cho ứng dụng trở nên linh hoạt hơn, bảo trì dễ dàng hơn và có thể mở rộng hơn.

Clean architecture dựa trên các nguyên tắc sau:

* Tính gắn kết cao: Mã trong mỗi lớp nên được kết nối chặt chẽ, có nghĩa là các lớp và mô-đun trong lớp nên phụ thuộc lẫn nhau nhiều.
* Tính liên kết thấp: Mã giữa các lớp nên được liên kết lỏng lẻo, có nghĩa là các lớp không nên phụ thuộc trực tiếp vào nhau. Thay vào đó, chúng nên giao tiếp thông qua các giao diện được định nghĩa rõ ràng.
* Nguyên tắc đảo ngược phụ thuộc (DIP): DIP nói rằng các mô-đun cấp cao không nên phụ thuộc vào các mô-đun cấp thấp. Cả hai nên phụ thuộc vào các trừu tượng.

Clean architecture thường được biểu thị dưới dạng sơ đồ phân lớp, với business logic cốt lõi ở trung tâm. Các lớp được sắp xếp theo cách mà các lớp bên trong ít có khả năng bị ảnh hưởng bởi các thay đổi đối với các lớp bên ngoài hơn.

Dưới đây là một sơ đồ phân lớp điển hình cho clean architecture:

* Lớp trình bày: Lớp này xử lý tương tác với người dùng và hiển thị dữ liệu. Nó thường được triển khai bằng một framework front-end như React hoặc Vue.js.
* Lớp ứng dụng: Lớp này chứa business logic của ứng dụng. Nó chịu trách nhiệm xử lý các yêu cầu của người dùng, lấy dữ liệu từ các nguồn bên ngoài và cập nhật cơ sở dữ liệu.
* Lớp miền: Lớp này chứa các khái niệm cốt lõi của ứng dụng. Nó độc lập với bất kỳ triển khai cụ thể nào và có thể được sử dụng lại trong các ứng dụng khác nhau.
* Lớp cơ sở hạ tầng: Lớp này chứa mã để tương tác với các phụ thuộc bên ngoài như cơ sở dữ liệu, API và hệ thống nhắn tin.

Clean architecture có thể là một chủ đề phức tạp, nhưng nó có thể là một cách tiếp cận rất mạnh mẽ để xây dựng các ứng dụng phần mềm bảo trì và có thể mở rộng.

Dưới đây là một số lợi ích của việc sử dụng clean architecture:

* Module hóa: Clean architecture làm cho ứng dụng trở nên modular hơn, điều này làm cho nó dễ dàng hơn để phát triển, kiểm tra và bảo trì.
* Linh hoạt: Clean architecture làm cho ứng dụng linh hoạt hơn với các thay đổi, vì business logic cốt lõi được cô lập khỏi các phụ thuộc bên ngoài.
* Khả năng bảo trì: Clean architecture làm cho ứng dụng dễ bảo trì hơn, vì mã được tổ chức tốt và dễ hiểu.

### Ứng dụng kiến trúc Clean Architecture

Clean architecture có thể được áp dụng cho nhiều loại ứng dụng phần mềm, bao gồm:

* Ứng dụng web
* Ứng dụng di động
* Ứng dụng doanh nghiệp

Clean Architecture giúp tách biệt business logic khỏi các triển khai cụ thể, làm cho ứng dụng dễ dàng thích ứng với các thay đổi trong quy trình kinh doanh.

Trong một ứng dụng web thương mại điện tử, lớp trình bày có thể được sử dụng để xử lý các yêu cầu của người dùng, chẳng hạn như tìm kiếm sản phẩm và đặt hàng. Lớp ứng dụng có thể được sử dụng để xử lý business logic của ứng dụng, chẳng hạn như tính toán giá cả và xử lý thanh toán. Lớp miền có thể được sử dụng để chứa các khái niệm kinh doanh cốt lõi của ứng dụng, chẳng hạn như sản phẩm và đơn đặt hàng. Lớp cơ sở hạ tầng có thể được sử dụng để tương tác với các phụ thuộc bên ngoài, chẳng hạn như cơ sở dữ liệu và API thanh toán.

Trong một ứng dụng di động ngân hàng, lớp trình bày có thể được sử dụng để hiển thị giao diện người dùng cho người dùng, chẳng hạn như màn hình đăng nhập và màn hình chuyển tiền. Lớp ứng dụng có thể được sử dụng để xử lý logic kinh doanh của ứng dụng, chẳng hạn như xác thực người dùng và xử lý giao dịch tài chính. Lớp miền có thể được sử dụng để chứa các khái niệm kinh doanh cốt lõi của ứng dụng, chẳng hạn như tài khoản và giao dịch. Lớp cơ sở hạ tầng có thể được sử dụng để tương tác với các phụ thuộc bên ngoài, chẳng hạn như API ngân hàng và dịch vụ định vị địa lý.

Trong một ứng dụng CRM, lớp trình bày có thể được sử dụng để hiển thị giao diện người dùng cho nhân viên bán hàng, chẳng hạn như màn hình khách hàng và màn hình lịch hẹn. Lớp ứng dụng có thể được sử dụng để xử lý logic kinh doanh của ứng dụng, chẳng hạn như theo dõi khách hàng tiềm năng và quản lý các mối quan hệ khách hàng. Lớp miền có thể được sử dụng để chứa các khái niệm kinh doanh cốt lõi của ứng dụng, chẳng hạn như khách hàng và cơ hội bán hàng. Lớp cơ sở hạ tầng có thể được sử dụng để tương tác với các phụ thuộc bên ngoài, chẳng hạn như cơ sở dữ liệu khách hàng và API tiếp thị.

Tóm lại, kiến trúc Clean Architecture là một cách tiếp cận thiết kế phần mềm linh hoạt và hiệu quả có thể được áp dụng cho nhiều loại ứng dụng phần mềm. Nó giúp tách biệt logic kinh doanh cốt lõi của ứng dụng khỏi các phụ thuộc bên ngoài, làm cho ứng dụng dễ phát triển, bảo trì và mở rộng hơn.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Giới thiệu

Phân tích thiết kế hệ thống thông tin là một giai đoạn quan trọng trong quy trình xây dựng ứng dụng phần mềm. Giai đoạn này giúp các nhà phát triển phần mềm hiểu rõ các yêu cầu của hệ thống, từ đó thiết kế và cài đặt hệ thống chính xác và hiệu quả hơn.

## Các qui trình nghiệp vụ

### Quy trình quản lý sinh viên tham gia khóa luận và phân công nhiệm vụ ra đề

Đến mỗi đợt đăng ký khóa luận, sinh viên sẽ đăng ký tham gia trên Cổng thông tin sinh viên (<https://sinhvien.hufi.edu.vn>) trong đợt đăng ký khóa luận tốt nghiệp. Sau khi kết thúc đợt đăng ký, giáo vụ khoa sẽ nhận danh sách sinh viên đăng ký tham gia khóa luận của đợt đó từ phòng đào tạo. Khoa CNTT tính toán số lượng đề tài tối thiểu cần được ra trong đợt đăng ký đó để phù hợp với số lượng sinh viên tham gia. Trưởng khoa sẽ lên kế hoạch phân công cho từng bộ môn thực hiện ra đề cho phù hợp với số lượng sinh viên tham gia. Bộ môn sẽ tiếp tục giao nhiệm vụ cho từng giảng viên ra đề, mỗi giảng viên nhận nhiệm vụ bắt buộc phải ra số lượng đề tối thiểu theo nhiệm vụ được giao.

Với quy trình này, khoa CNTT có thể quản lý được số lượng sinh viên tham gia khóa luận tốt nghiệp và phân công đề tài một cách chuyên nghiệp để đảm bảo cho sinh viên có đề tài phù hợp và đáp ứng được yêu cầu của khóa luận tốt nghiệp.

### Quy trình ra đề và kiểm duyệt đề tài của giảng viên

Khi nhận được nhiệm vụ ra đề, giảng viên thực hiện đưa ra các đề tài phù hợp với các thông tin như tên đề tài, tóm tắt, số lượng sinh viên có thể thực hiện, đề tài phù hợp với những chuyên ngành nào. Sau đó, giảng viên sẽ gửi những đề tài mình ra cho trưởng bộ môn. Trưởng bộ môn kiểm duyệt đề tài để đảm bảo rằng đề tài phù hợp với năng lực của sinh viên cũng như chương trình đào tạo. Nếu đề tài chưa phù hợp do yêu cầu quá ít hay quá cao, trưởng bộ môn sẽ góp ý chỉnh sửa và yêu cầu giảng viên thực hiện sửa đổi lại. Tiếp tục như vậy cho đến khi đề tài đạt yêu cầu, sẽ đủ điều kiện để mở cho sinh viên đăng ký khi tới đợt mở đăng ký.

Với quy trình này, giảng viên được đảm bảo có đề tài phù hợp và đáp ứng được yêu cầu của khóa luận tốt nghiệp. Đồng thời, quy trình kiểm duyệt của trưởng bộ môn giúp đảm bảo chất lượng đề tài và phù hợp với năng lực của sinh viên, giúp đảm bảo thành công cho quá trình thực hiện khóa luận tốt nghiệp.

### Quy trình đăng ký đề tài khóa luận tốt nghiệp của sinh viên

Khi đến thời gian đăng ký khóa luận, giáo vụ khoa công bố danh sách những đề tài khóa luận đã được duyệt lên website của khoa (<https://fit.hufi.edu.vn>) với mỗi đề tài là một form đăng ký. Sinh viên tự chủ động lập nhóm và ủy quyền một đại diện của nhóm để đăng ký đề tài qua form tương ứng trên website. Khi đăng ký, sinh viên được ủy quyền sẽ cung cấp đầy đủ thông tin thành viên của nhóm mình. Mỗi form chỉ cho phép một sinh viên đăng ký và tự động đóng lại sau khi đã có sinh viên đăng ký. Nếu đề tài đã được đăng ký, những sinh viên đăng ký sau sẽ phải lựa chọn một đề tài khác để đăng ký. Sau khi thời gian đăng ký kết thúc, giáo vụ khoa sẽ tổng hợp danh sách sinh viên theo đề tài đăng ký để thực hiện quyết định giao đề tài. Cuối cùng, khoa sẽ công bố danh sách sinh viên theo từng đề tài và GVHD đã được đăng ký trước đó, giúp sinh viên có thể tiếp cận với giảng viên để bắt đầu thực hiện khóa luận của mình.

### Quy trình quản lý tiến độ thực hiện đề tài của giảng viên hướng dẫn

Sau khi sinh viên đăng ký đề tài và nhận được quyết định giao đề tài, họ sẽ liên hệ với GVHD theo thông tin được cung cấp trên website của khoa. Sinh viên và giảng viên sẽ thảo luận cụ thể về đề tài, bao gồm mục tiêu, phạm vi, phương pháp thực hiện, kế hoạch thực hiện và tiến độ của dự án. Giảng viên sẽ đề ra các yêu cầu cần thiết cho sinh viên, phân bổ tiến độ và đặt hạn báo cáo tiến độ cho từng giai đoạn của đề tài.

Sinh viên phải hoàn thành các yêu cầu mà giảng viên đưa ra và đạt được tiến độ đề ra trong quá trình thực hiện đề tài. Giảng viên sẽ theo dõi tiến độ của sinh viên và đánh giá, đưa ra các góp ý và chỉnh sửa cần thiết qua từng đợt báo cáo tiến độ. Sinh viên cần phản hồi và tiếp nhận góp ý để hoàn thiện đề tài đúng hạn và chất lượng tốt nhất.

Trong trường hợp sinh viên không đạt được tiến độ hoặc gặp khó khăn trong quá trình thực hiện, sinh viên cần thông báo cho GVHD sớm để được giúp đỡ và hỗ trợ kịp thời. Tuy nhiên, nếu sinh viên không hoàn thành đề tài hoặc không đạt yêu cầu về chất lượng, giảng viên có thể yêu cầu sinh viên làm lại hoặc chấm dứt hợp tác (đánh rớt đề tài).

### Quy trình chấm điểm cho đề tài cho sinh viên

Sau khi sinh viên đã hoàn thành được các nhiệm vụ theo tiến độ mà giảng viên giao, sinh viên sẽ liên hệ GVHD và phản biện để thống nhất thời gian phản biện đề tài. Tại buổi phản biện, sinh viên sẽ trình bày ý tưởng, phương pháp thực hiện đề tài và kết quả thực hiện được cũng như nhiệm vụ của từng thành viên thực hiện trong nhóm. GVHD và GVPB sẽ đánh giá và chấm điểm đề tài cho từng thành viên trong nhóm thực hiện. Nếu đề tài không được GVHD hoặc GVPB duyệt thì xem như đề tài không đạt và bị đánh rớt. Ngược lại, đề tài được cả hai GVHD và phản biện duyệt, đồng ý đưa ra hội đồng thì nhóm sinh viên phải tiếp tục phản biện tại hội đồng theo thông báo từ khoa. Tại hội đồng, sinh viên tiếp tục trình bày kết quả thực hiện đề tài cho hội đồng (bao gồm: 1 chủ tịch, 1 thư kí và các ủy viên). Các thành viên trong hội đồng sẽ đánh giá và đặt ra câu hỏi cho nhóm sinh viên. Sinh viên phải trả lời các câu hỏi của thành viên hội đồng đưa ra và nhận góp ý từ họ. Sau đó, hội đồng sẽ dựa trên kết quả trình bày và câu trả lời của sinh viên để chấm điểm cho từng sinh viên trong nhóm thực hiện đề tài. Điểm tổng kết của sinh viên là điểm trung bình của GVHD, GVPB và các thành viên trong hội đồng. Nếu điểm tổng kết không đạt yêu cầu đồng nghĩa với việc đề tài sẽ bị đánh rớt.

Với quy trình này, sinh viên sẽ được đánh giá và chấm điểm công bằng dựa trên quá trình hoàn thành đề tài của mình. Ngoài ra, việc tham gia phản biện và bảo vệ đề tài tại hội đồng cũng giúp sinh viên nâng cao kỹ năng trình bày, thuyết phục và trả lời câu hỏi.

### Quy trình thống kê kết quả của khoa CNTT

Sau khi các sinh viên của khoa CNTT báo cáo tại hội đồng xong, quá trình thống kê kết quả được thực hiện bởi giáo vụ khoa. Trước tiên, giáo vụ sẽ tổng hợp kết quả từ tất cả các hội đồng và các đề tài bị đánh rớt không được đưa ra hội đồng. Các cột điểm của sinh viên sẽ được liệt kê, bao gồm cả các cột điểm của những sinh viên không báo cáo (xem như 0 điểm). Quá trình này nhằm đảm bảo rằng tất cả các điểm đều được tính toán và không bị bỏ sót.

Sau khi thống kê được tổng hợp, giáo vụ sẽ gửi lại cho trưởng khoa để xác nhận. Trưởng khoa sẽ tiến hành kiểm tra lại kết quả thống kê để đảm bảo tính chính xác của các điểm số. Nếu không có sự khác biệt nào, trưởng khoa sẽ xác nhận kết quả và cho phép giáo vụ gửi lại danh sách điểm cho giảng viên.

Quá trình cập nhật điểm trên Cổng thông tin sinh viên sẽ được thực hiện bởi GVHD. Sau khi thống kê được xác nhận, giảng viên sẽ cập nhật điểm cho từng sinh viên và đảm bảo rằng các điểm số được cập nhật đầy đủ và chính xác. Các sinh viên có thể xem điểm của mình trên Cổng thông tin sinh viên và liên hệ với GVHD nếu có bất kỳ thắc mắc nào.

## Xác định yêu cầu

### Yêu cầu chức năng

* Quản lí thông tin giảng viên, đề tài, sinh viên, điểm số của sinh viên, tiến độ của khóa luận.
* Quản lý các yêu cầu và phản hồi của sinh viên và giảng viên.
* Cập nhật các thông tin, thông báo về khóa luận tốt nghiệp từ khoa.
* Tạo báo cáo, thống kê về tiến độ, kết quả của khóa luận.
* Xác thực và phân quyền cho người dùng.

### Yêu cầu phi chức năng

* Giao diện đơn giản, thân thiện với người dùng.
* Hệ thống phải đáp ứng được số lượng lớn người dùng truy cập đồng thời.
* Tốc độ truy cập và xử lý nhanh, đảm bảo không bị gián đoạn hoặc treo máy.
* Hệ thống phải đảm bảo bảo mật thông tin người dùng, bao gồm mã hóa dữ liệu nhạy cảm khi lưu trữ và truyền tải
* Hệ thống phải có giao diện người dùng phản hồi, hoạt động tốt trên các thiết bị khác nhau

### 3.3.3 Tiêu chí chấp nhận của người dùng

Tiêu chí chấp nhận của người dùng xác định các điều kiện mà hệ thống "Hệ Thống Tiến Trình Khóa Luận Tốt Nghiệp" phải đáp ứng để được coi là hoàn thành và sẵn sàng đưa vào sử dụng bởi các bên liên quan, bao gồm sinh viên, giảng viên, giáo vụ, và quản trị viên. Các tiêu chí này bao gồm:

**+Hoàn thành tất cả chức năng chính**

* Tất cả các chức năng được liệt kê trong mục **3.3.1 Yêu cầu chức năng** (quản lý danh sách sinh viên, đề tài, giảng viên, đăng ký đề tài, theo dõi tiến độ, chấm điểm, và xuất dữ liệu ra Excel) phải được triển khai đầy đủ và hoạt động đúng như mô tả.
* Không có lỗi nghiêm trọng (critical bugs) ảnh hưởng đến các chức năng chính, chẳng hạn như lỗi trong đăng ký đề tài, nộp báo cáo tiến độ, hoặc chấm điểm.

+ **Đáp ứng yêu cầu phi chức năng**:

* Hệ thống phải đáp ứng các yêu cầu về bảo mật (HTTPS, mã hóa dữ liệu, kiểm soát truy cập dựa trên vai trò), hiệu suất (phản hồi dưới 2 giây, hỗ trợ 1.000 người dùng đồng thời), và thiết kế phản hồi (hoạt động tốt trên máy tính, máy tính bảng, và điện thoại thông minh) như đã nêu trong mục **3.3.2 Yêu cầu phi chức năng**.
* Giao diện người dùng phải trực quan, dễ sử dụng, và không yêu cầu đào tạo phức tạp cho người dùng cuối.

+ **Kiểm thử thành công**:

* Hệ thống phải vượt qua các bài kiểm thử chức năng, bảo mật, hiệu suất, và khả năng sử dụng, với tỷ lệ lỗi không vượt quá 5% trong các trường hợp kiểm thử.
* Kiểm thử với người dùng thực tế (sinh viên, giảng viên, giáo vụ) phải được thực hiện, và ít nhất 90% người dùng phải xác nhận rằng hệ thống đáp ứng nhu cầu của họ về tính năng và trải nghiệm người dùng.

+ **Tài liệu hướng dẫn sử dụng**:

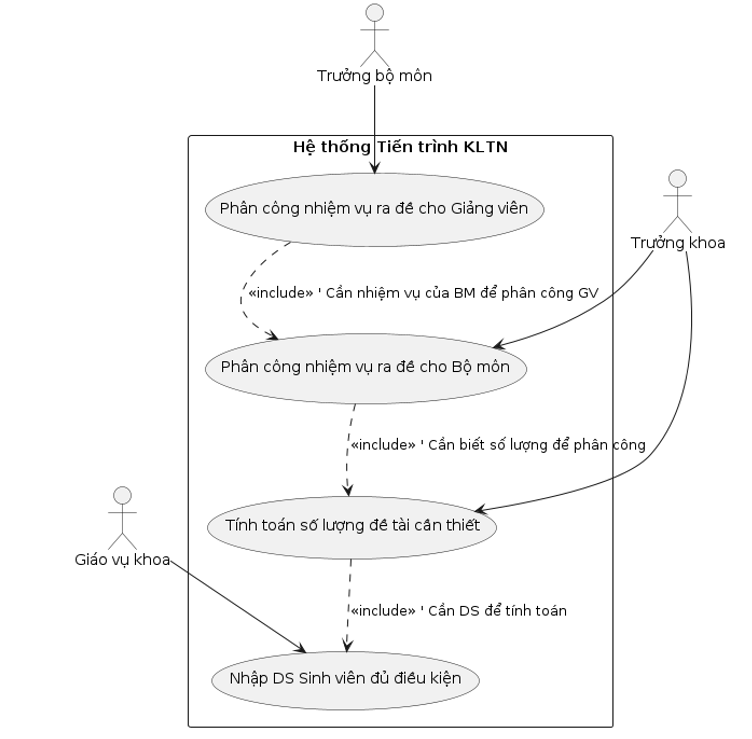
* Một tài liệu hướng dẫn sử dụng chi tiết phải được cung cấp, bao gồm các bước sử dụng hệ thống cho từng vai trò người dùng (sinh viên, giảng viên, giáo vụ, quản trị viên).
* Tài liệu này phải được người dùng cuối xác nhận là dễ hiểu và đầy đủ thông qua phản hồi sau khi kiểm thử.

+ **Phê duyệt từ các bên liên quan**:

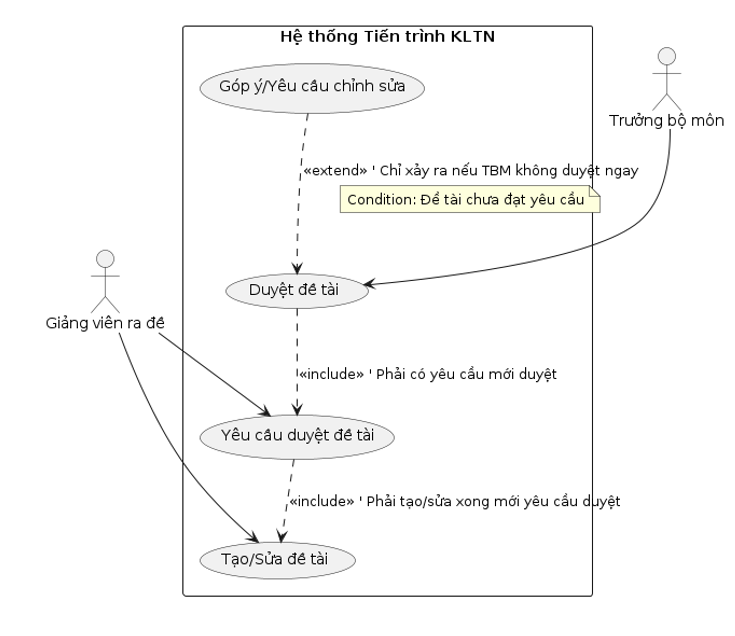
* Hệ thống phải được trình bày và phê duyệt bởi đại diện của Khoa Công Nghệ Thông Tin, bao gồm trưởng khoa, giáo vụ, và một nhóm đại diện sinh viên và giảng viên.
* Phản hồi từ các bên liên quan phải được ghi nhận và xử lý trước khi triển khai chính thức.

## Biểu đồ Use case :

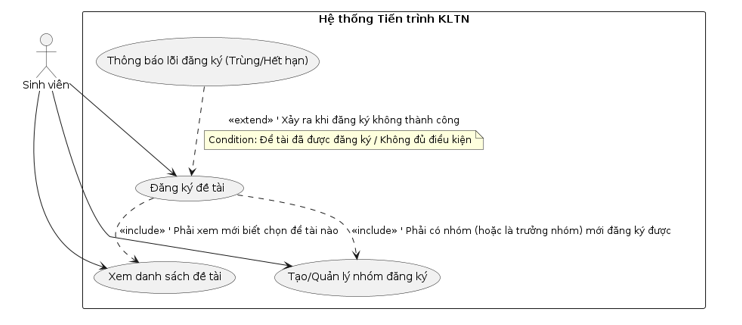
### Phân công nhiệm vụ ra đề



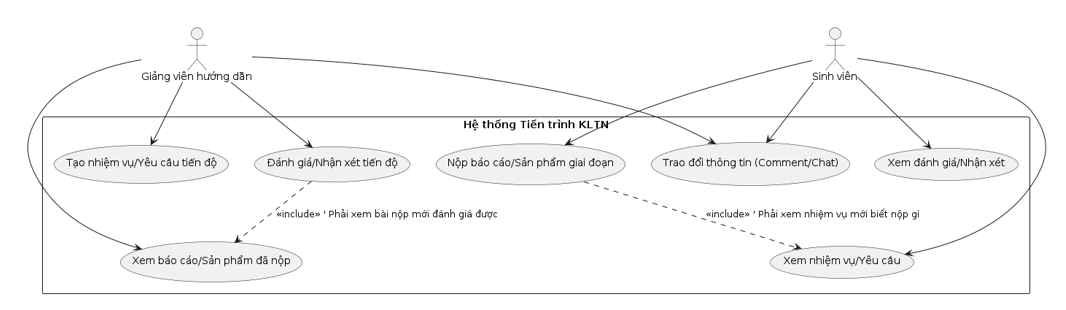
### Quản lý đề xuất và duyệt đề tài



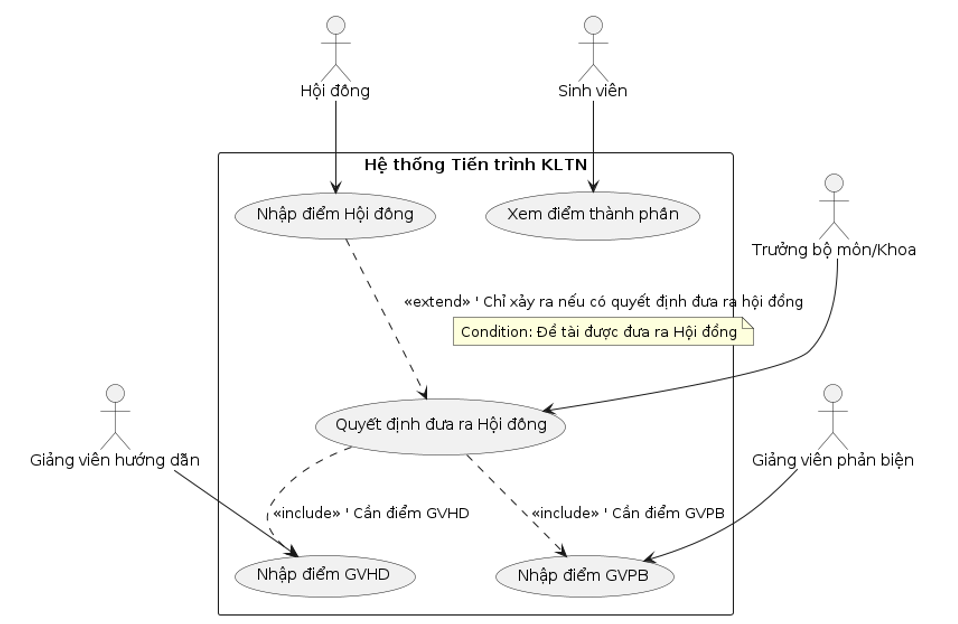
### Đăng ký đề tài Khóa luận



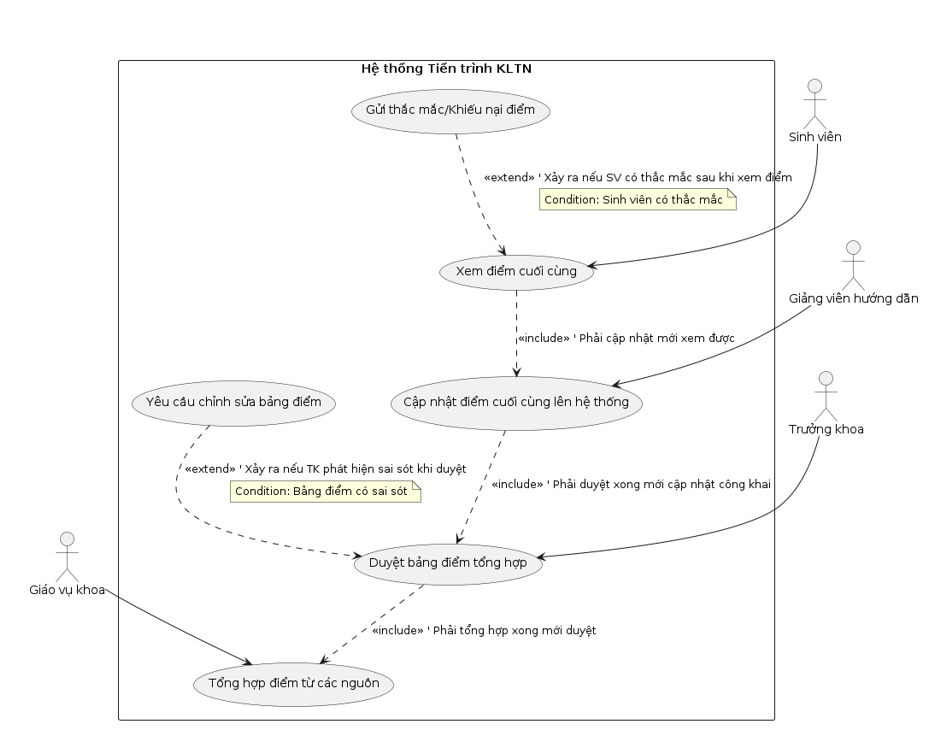
### Quản lý tiến độ thực hiện đề tài



### Chấm điểm Khóa luận



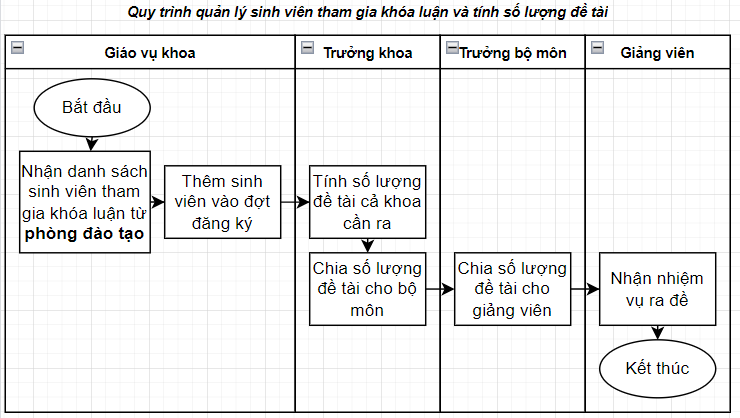
### Thống kê và công bố kết quả



## Quy trình nghiệp vụ

### Quy trình quản lý sinh viên tham gia khóa luận và phân công nhiệm vụ ra đề

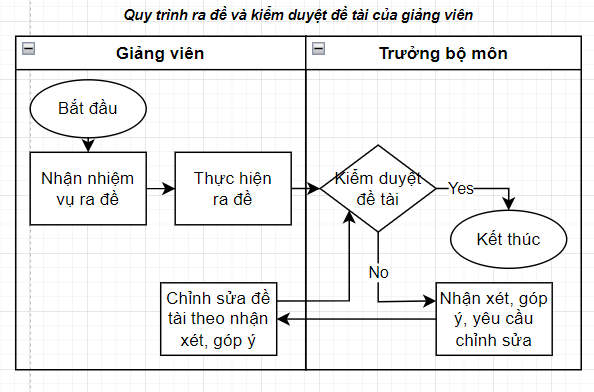
|  |  |
| --- | --- |
| Mô tả | Quy trình quản lý sinh viên tham gia khóa luận và phân công nhiệm vụ ra đề là một loạt các bước cụ thể được thực hiện theo thứ tự để đảm bảo rằng giảng viên ra lượng đề phù hợp với số lượng sinh viên tham gia khóa luận tốt nghiệp của đợt cụ thể |
| Dòng cơ bản | 1. Giáo vụ: nhận danh sách sinh viên tham gia khóa luận tốt nghiệp từ phòng đào tạo sau đợt đăng ký khóa luận  2. Trưởng khoa: tính toán số lượng đề tài tối thiểu cần ra dựa trên danh sách sinh viên đăng ký tham gia.  3. Trưởng khoa: phân công cho từng bộ môn số lượng đề tài tối thiểu cần ra dựa trên số lượng đề tài đã tính toán.  4. Trưởng bộ môn: giao nhiệm vụ ra đề cho các giảng viên với số lượng tối thiểu dựa trên số lượng đề tài được phân công.  5. Giảng viên: nhận nhiệm vụ ra đề |

**

Hình . Sơ đồ hoạt động qui trình quản lý sinh viên tham gia khóa luận và tính số lượng đề tài

### Quy trình ra đề và kiểm duyệt đề tài của giảng viên

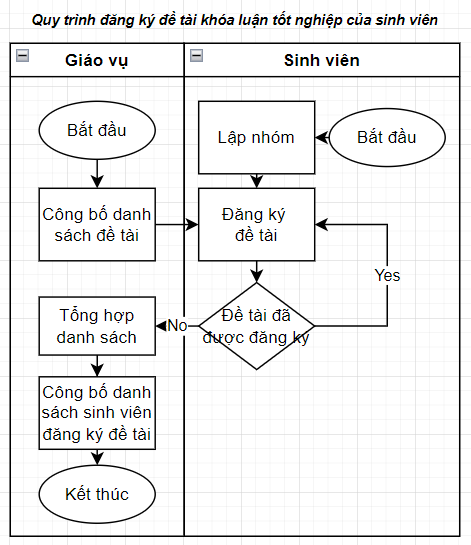
|  |  |
| --- | --- |
| Mô tả | Quy trình ra đề và kiểm duyệt đề tài của giảng viên nhằm đảm bảo hàm lượng của đề tài phù hợp cho sinh viên tham gia khóa luận thực hiện |
| Dòng cơ bản | 1. Giảng viên: nhận nhiệm vụ và thực hiện ra đề tài  2. Trưởng bộ môn: kiểm duyệt đề tài  3. Đề tài được thông qua |
| Dòng thay thế | Tại bước 2, nếu đề tài không đạt yêu cầu, trưởng bộ môn sẽ nhận xét góp ý, giảng viên theo đó sẽ chỉnh sửa lại đề tài và thực hiện lại bước 2 |



Hình . Sơ đồ hoạt động quy trình ra đề và kiểm duyệt đề tài của giảng viên

### Quy trình đăng ký đề tài khóa luận tốt nghiệp của sinh viên

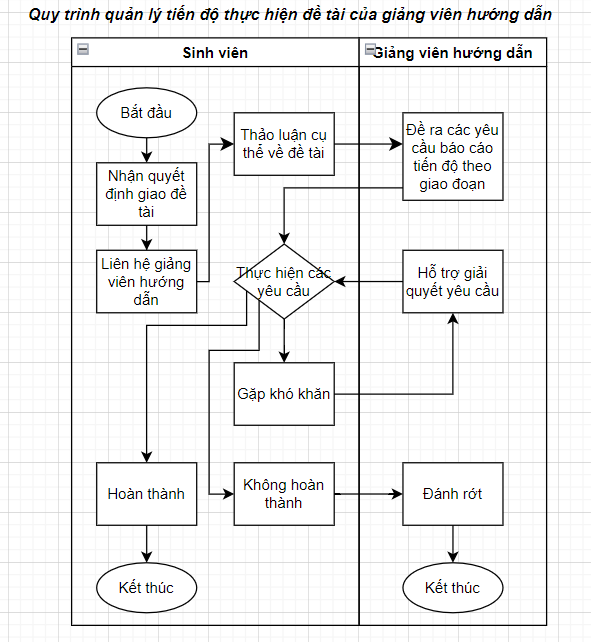
|  |  |
| --- | --- |
| Mô tả | Quy trình đăng ký đề tài khóa luận tốt nghiệp của sinh viên nhằm đảm bảo sinh viên đăng ký được đề tài theo nhóm và không có đề tài nào có trên 1 nhóm đăng ký |
| Dòng cơ bản | 1. Giáo vụ: công bố danh sách đề tài được kiểm duyệt  2. Sinh viên: lập nhóm và ủy quyền đại diện đăng ký đề tài  3. Sinh viên: đăng ký đề tài  4. Giáo vụ: tổng hợp danh sách sinh viên theo đề tài đăng ký  5. Giáo vụ: công bố danh sách sinh viên theo đề tài và GVHD |
| Dòng thay thế | Tại bước 3, nếu đề tài đã có nhóm đăng ký trước đó, sinh viên không thể đăng ký được đề tài đó mà phải thực hiện lại bước 3. |



Hình . Sơ đồ hoạt động quy trình đăng ký đề tài khóa luận tốt nghiệp của sinh viên

### Quy trình quản lý tiến độ thực hiện đề tài của giảng viên hướng dẫn

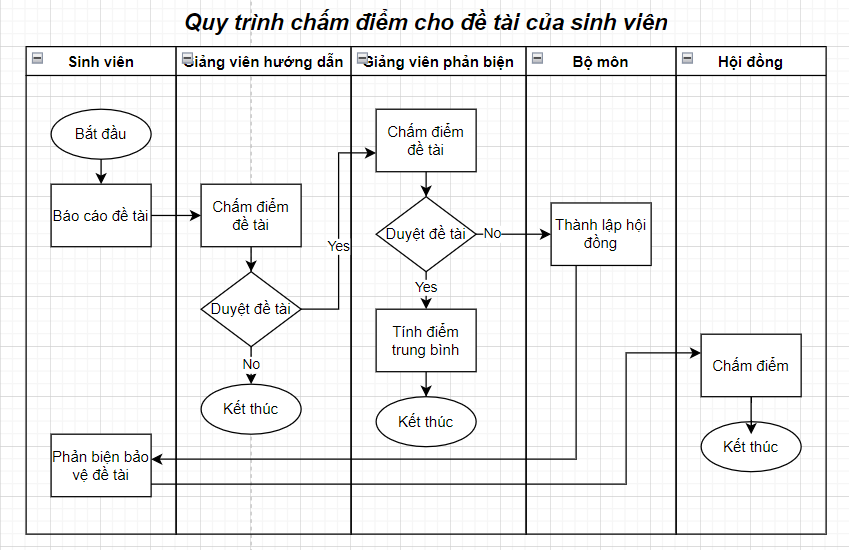
|  |  |
| --- | --- |
| Mô tả | Quy trình quản lý tiến độ thực hiện đề tài của giảng viên hướng dẫn nhằm lấy được thông tin thực hiện đề tài của từng sinh viên giúp giảng viên dựa vào đó quyết định điểm số cho sinh viên. |
| Dòng cơ bản | 1. Sinh viên: nhận quyết định giao đề tài  2. Sinh viên: liên hệ GVHD  3. Sinh viên và giảng viên: thảo luận cụ thể đề tài  4. Giảng viên: đề ra các yêu cầu báo cáo tiến độ theo giai đoạn  5. Sinh viên: thực hiện các yêu cầu của giảng viên  6. Hoàn thành khóa luận |
| Dòng thay thế | Tại bước 5, nếu sinh viên gặp khó khăn sẽ thông báo cho giảng viên nhờ hỗ trợ và quay lại bước 5  Tại bước 5: nếu không hoàn thành, giảng viên yêu cầu làm lại hoặc đánh rớt tùy mức độ |



Hình 3.4 Sơ đồ hoạt động quy trình quản lý tiến độ thực hiện đề tài của GVHD

### Quy trình chấm điểm cho đề tài của sinh viên

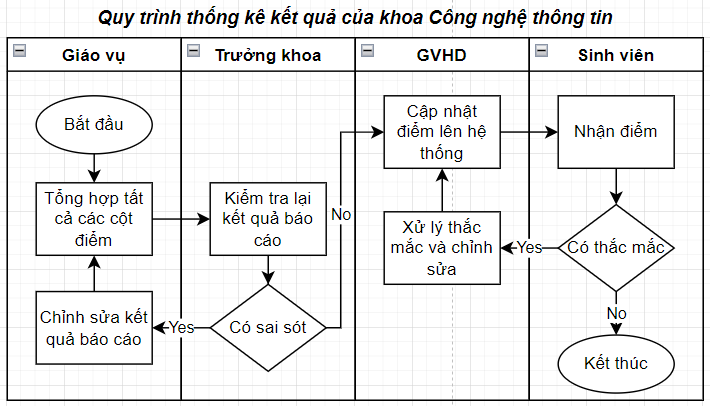
|  |  |
| --- | --- |
| Mô tả | Quy trình chấm điểm cho đề tài của sinh viên nhằm đảm bảo điểm báo cáo đề tài của sinh viên là khách quan của GVHD, GVPB |
| Dòng cơ bản | 1. Sinh viên: báo cáo đề tài với GVHD  2. GVHD: Chấm điểm và duyệt đề tài  3. Sinh viên: Phản biện để bảo vệ đề tài  4. GVPB: chấm điểm và duyệt đề tài  5. Hoàn tất cho điểm |
| Dòng thay thế | Tại bước 2: Nếu GVHD không duyệt đề tài, qui trình kết thúc  Tại bước 3: Nếu GVPB duyệt đề tài không đạt, trưởng bộ môn sẽ thành lập hội đồng để chấm điểm |



Hình . Sơ đồ hoạt động quy trình chấm điểm cho đề tài của sinh viên

### Quy trình thống kê kết quả của khoa CNTT

|  |  |
| --- | --- |
| Mô tả | Quy trình thống kê kết quả của khoa CNTT nhằm giúp tổng hợp điểm của sinh viên trong đợt khóa luận và sinh viên có thể gửi thắc mắc nếu có thắc mắc |
| Dòng cơ bản | 1. Giáo vụ: tổng hợp tất cả các cột điểm thành kết quả báo cáo  2. Trưởng khoa: kiểm tra lại kết quả báo cáo  3. GVHD: Cập nhật điểm lên hệ thống  4. Sinh viên: Nhận điểm |
| Dòng thay thế | Tại bước 2: Nếu trưởng khoa thấy có sai sót, yêu cầu giáo vụ chỉnh sửa kết quả báo cáo và quay lại bước 1  Tại bước 4: Nếu sinh viên có thắc mắc gặp GVHD để xử lý thắc mắc, chỉnh sửa và cập nhật điểm lên hệ thống |

****

Hình 3.6 Sơ đồ hoạt động quy trình thống kê kết quả của khoa CNTT

### Quy trình thành đưa đề tài ra hội đồng

|  |  |
| --- | --- |
| Mô tả | Quy trình đưa đề tài ra hội đồng giúp cho việc chấm điểm của đề tài có điểm số cách biệt lớn giữa GVPB và GVHD |
| Dòng cơ bản | 1. GVHD chấm điểm đề tài  2. GVPB chấm điểm đề tài  3. Đưa đề tài ra hội đồng  4. Sinh viên: Nhận điểm |
| Dòng thay thế | Tại bước 2: Nếu điểm số không cách biệt quá lớn, bỏ qua bước 4 |

# KẾT LUẬN

Trên cơ sở nghiên cứu các mục tiêu của đề tài XÂY DỰNG HỆ THỐNG   
XỬ LÝ TIẾN TRÌNH KHÓA LUẬN TỐT NGHIỆP, chúng em rút ra một số kết luận sau:

Hệ thống đã đáp ứng được các mục tiêu đề ra, bao gồm:

* Quản lý danh sách sinh viên đủ điều kiện đăng ký khóa luận tốt nghiệp, danh sách đề tài, GVHD, GVPB.
* Cho phép giảng viên ra đề và nhận được sự nhận xét, phê duyệt từ trưởng bộ môn, trưởng khoa.
* Cho phép sinh viên trong đợt tham gia có thể tự do lựa chọn thành viên và lập nhóm, đăng ký đề tài khóa luận tốt nghiệp theo nhóm, nộp kết quả báo cáo theo tiến độ và nhận góp ý từ GVHD.
* Cho phép giảng viên tạo ra yêu cầu theo tiến độ của đề tài, theo dõi, nhận xét, góp ý để sinh viên hoàn thành tốt đề tài và chấm điểm trên hệ thống.

Hệ thống có các ưu điểm sau:

* Hệ thống Dễ sử dụng, đáp ứng được nhu cầu khoa CNTT.
* Có các tính năng đáp ứng được các yêu cầu của quản lý khóa luận tốt nghiệp.
* Có khả năng mở rộng và nâng cấp.

Hệ thống có thể được cải tiến thêm như sau:

* Tăng cường khả năng bảo mật thông tin.
* Tích hợp thêm các tính năng mới, đáp ứng được các nhu cầu phát triển của khoa CNTT.

Kết luận chung

Hệ thống xử lý tiến trình khóa luận tốt nghiệp là một hệ thống hoàn chỉnh, đáp ứng được các mục tiêu đề ra. Hệ thống có các ưu điểm như dễ sử dụng, đáp ứng được nhu cầu nhu cầu phát triển của khoa CNTT.

# TÀI LIỆU THAM KHẢO

[1] Kirtesh Shah(10/2023). ASP.NET Core 5.0 Web, viewed 11/09/2023 from <https://www.c-sharpcorner.com/article/asp-net-core-5-0-web-api/>

[2] What is Next.js?, viewed 11/09/2023 from

<https://nextjs.org/learn-pages-router/foundations/about-nextjs/what-is-nextjs>

[3] Alex Bykow (1/2023), Clean architecture – everything you need to know, viewed 11/09/2023 from <https://codilime.com/blog/clean-architecture/>