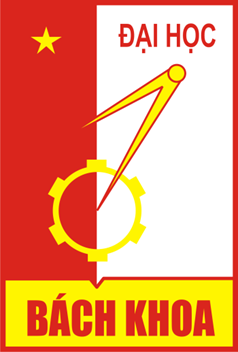
TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

**─────── \* ───────**



**BÀI TẬP LỚN**

**NHẬP MÔN CÔNG NGHỆ PHẦN MỀM**

**Đề tài:** Tìmhiểu đặc tả yêu cầu, phân tích thiết kế hệ thống và thiết kế một số trường hợp kiểm thử cho hệ thống Quản lý sinh viên với đầy đủ các chức năng cơ bản ( quản lý đăng ký, quản lý điểm, lớp, thông tin chung … )

**Giảng viên hướng dẫn:** TS Nguyễn Thanh Hùng

**Nhóm thực hiện:**

|  |  |  |  |
| --- | --- | --- | --- |
| Họ và Tên | Email | SĐT | MSSV |
| Đào Nam Tiến | Daonamtien1@gmail.com | 01649001819 | 20133924 |
| Nguyễn Đức Long | nguyenduclong12a2nd@gmail.com | 01639584196 | 20132374 |
| Nguyễn Thúc Huynh | Mr.huynh.1995@gmail.com | 0963132395 | 20131840 |
| Nguyễn Đình Duy | Toiladuy22@gmail.com | 0968226146 | 20130601 |
| Mai Trung Đức | Maitrungduc1410@gmail.com | 0978643924 | 20131004 |
| Mai Văn Khiêm | Khiemngo95@gmail.com | 01659690535 | 20132096 |

**Hà Nội, tháng 1 năm 2016**

Mục lục

[**LỜI NÓI ĐẦU** 3](#_Toc440210231)

[**CHƯƠNG 1:** **GIỚI THIỆU ĐỀ TÀI** 4](#_Toc440210232)

[1. Tên đề tài 4](#_Toc440210233)

[2. Mục đích 4](#_Toc440210234)

[3. Mô tả 4](#_Toc440210235)

[4. Chi tiết 4](#_Toc440210236)

[5. Rủi ro 4](#_Toc440210237)

[**CHƯƠNG 2:** **ĐẶC TẢ** 5](#_Toc440210238)

[1. Giới thiệu về hệ thống 5](#_Toc440210239)

[2. Công cụ đặc tả 5](#_Toc440210240)

[**CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ** 7](#_Toc440210241)

[1. Biểu đồ Use Case 7](#_Toc440210242)

[2. Biểu đồ hoạt động 8](#_Toc440210243)

[3. Biểu đồ tương tác 9](#_Toc440210244)

[4. Biểu đồ lớp 10](#_Toc440210245)

[**CHƯƠNG 4: LỰA CHỌN CÔNG NGHỆ** 11](#_Toc440210246)

[1. Sử dụng phương pháp lập trình hướng đối tượng 11](#_Toc440210247)

[2. Ưu điểm của mô hình hướng đối tượng: 11](#_Toc440210248)

[3. Phân tích hướng đối tượng (Object Oriented Analysis - OOA) 12](#_Toc440210249)

[4. Thiết kế hướng đối tượng (Object Oriented Design - OOD) 12](#_Toc440210250)

[5. Lập trình hướng đối tượng (Object Oriented Programming - OOP) 13](#_Toc440210251)

[6. Sử dụng cơ sở dữ liệu 13](#_Toc440210252)

[**CHƯƠNG 5:** **KIỂM THỬ** 14](#_Toc440210253)

[1. Kiểm thử hộp đen 14](#_Toc440210254)

[2. Kiểm thử hộp trắng 17](#_Toc440210255)

[**CHƯƠNG 6. KẾT LUẬN** 20](#_Toc440210256)

[**CHƯƠNG 7. CÔNG CỤ LÀM VIỆC NHÓM** 21](#_Toc440210257)

# **LỜI NÓI ĐẦU**

Ngày nay với sự phát triển của ngành công nghiệp nói chung, sự phát triển thiết yếu của ngành CNTT nói riêng, các phần mềm cũng đang rất phát triển, nó ứng dụng nhiều ngành nhiều nghề, và rất đa dạng, đáp ứng, giải quyết được nhiều vấn đề của con người, mà con người phải cần rất nhiều nhân lực. Vấn đề đó đã được giải quyết bằng các ứng dụng được phát triển. Một trong số đó, chúng em thấy vấn đề về quản lí cũng là một vấn đề khá là cần thiết đối với doanh nghiệp và nhà trường, vì vậy chúng em quyết định thực hiện đề tài 3: **"Tìm hiểu đặc tả, phân tích thiết kế hệ thống và thiết kế một số trường hợp kiểm thử cho Hệ thống quản lý sinh viên",** để giải quyết vấn đề xung quanh việc quản lí sinh viên, học sinh trong nhà trường, kiểm soát được điểm thi, việc đăng kí, thông tin cá nhân…

Đây là lần đầu tiên chúng em phân tích thiết kế hệ thống với mong muốn xây dựng phát triển được phần mềm hoàn thiện, và dễ dàng sử dụng quản lí đối với người dùng, nên không thể tránh khỏi thiếu sót. Chúng em rất mong được sự góp ý của thầy để ứng dụng của bọn em được hoàn thiện hơn. Chúng em chân thành cảm ơn.

Hà Nội, tháng 1 năm 2016

Nhóm sinh viên thực hiện

**CHƯƠNG 1:** **GIỚI THIỆU ĐỀ TÀI**

* 1. Tên đề tài

Tìm hiểu đặc tả yêu cầu, phân tích thiết kế hệ thống và thiết kế một số trường hợp kiểm thử cho hệ thống Quản lý sinh viên với đầy đủ các chức năng cơ bản ( quản lý đăng ký, quản lý điểm, lớp, thông tin chung … )

* 1. Mục đích
     + Số hóa thông tin quản lý hệ thống sinh viên gồm các thông tin cá nhân, thông tin điểm, lớp, địa chỉ...,
     + Nhằm giúp bộ phận quản lý sinh viên có thể dễ dàng nắm bắt thông tin sinh viên phục vụ công tác hành chính, sắp xếp học tập của sinh viên.
  2. Mô tả

Giao diện: dễ nhìn, dễ sử dụng, trực quan, phù hợp với người dùng. Cấu trúc thiết kế gọn gàng, dễ dàng thay đổi, cải tiến trong tương lai.

Thực hiện các chức năng về quản lý sinh viên:

* + - Tra cứu thông tin sinh viên
    - Cập nhật, thêm mới thông tin về sinh viên như: thông tin cá nhân, lớp, điểm…
    - Phân quyền quản lý, người sử dụng
    - Quản lý đăng ký lớp
  1. Chi tiết

Nhân sự thực hiện gồm 3 nhóm:

* + - Nhóm phân tích và thiết kế
    - Nhóm code, cài đặt mã
    - Nhóm kiểm thử, test, kết luận

Thời gian :

* + - 15 ngày: phân tích, thiết kế cơ sở dữ liệu, các biểu đồ, thuật giải, extension sẽ sử dụng và tham khảo, trao đổi phù hợp nhu cầu thực tế của người sử dụng
    - 10 ngày: code , thiết kế mã chương trình
    - 10 ngày: Kiểm thử, bảo trì chương trình .
  1. Rủi ro
     + Thiết kế cơ sở dữ liệu không chuẩn xác dẫn đến kích thước lớn hoặc trả về kết quả sai, dữ liệu nhập xuất không đúng như mong muốn .
     + Bị mất mát thông tin, bị xóa do một thành viên nào xóa nhầm và không xử dụng backup cloud hoặc github …
     + Thiếu ngân quỹ hoạt động
     + Các thành viên bị ốm đau, có việc cá nhân đột xuất..

**CHƯƠNG 2:** **ĐẶC TẢ**

* 1. Giới thiệu về hệ thống

- Hệ thống Quản lý sinh viên gồm các chức năng:

+ Thông tin về người quản lý(Tên,tuổi,email,số điện thoại...)

+ Thông tin về sinh viên (Tên,tuổi,email,số điện thoại...)

+ Người quản lý sẽ có chức năng là Update,Delete,Edit sinh viên(Có thể sử dụng MySQL).Người quản lý quản lý điểm,lớp, thông tin của sinh viên...

+ Người quản lý có thể tìm kiếm thông tin sinh viên dựa vào công cụ tìm kiếm(Search)

* + - Người sử dụng có thể tìm kiếm tất cả CSDL hoặc một tập con của CSDL
    - Hệ thống sẽ có những giao diện thích hợp cho việc tra cứu của người sử dụng
    - Tất cả thông tin của sinh viên khi người sử dụng đăng nhập sẽ được hiển thi ra.
    - Có thể xuất ra file
  1. Công cụ đặc tả

+ Đặc tả chức năng với DFD (data flowdiagram )

+ Đặc tả trạng thái với FSM - Finite State Machines

+ Đặc tả dữ liệu với mô hình thực thể liên kết - ERD

+ Đặc tả tốt sẽ :

- Dễ hiểu với người dùng

- Có ít điều kiện nhập nhằng

- Có ít quy ước khi mô tả,có thể tạo đơn giản

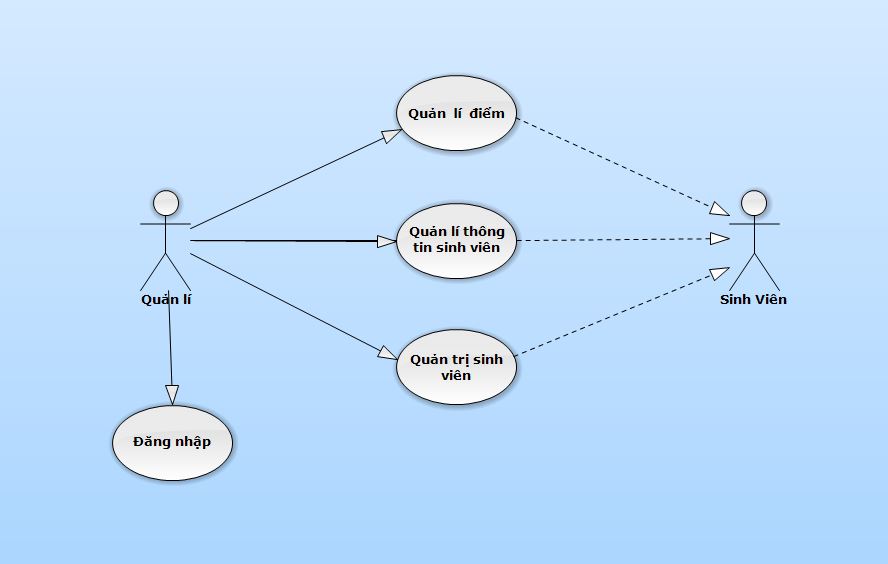
- Dễ triển khai cho những pha sau của vòng đời:thiết kế hệ thống và thiết kế chương trình và giao diện dễ làm, đảm bảo tính nhất quán, . . .

**CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ**

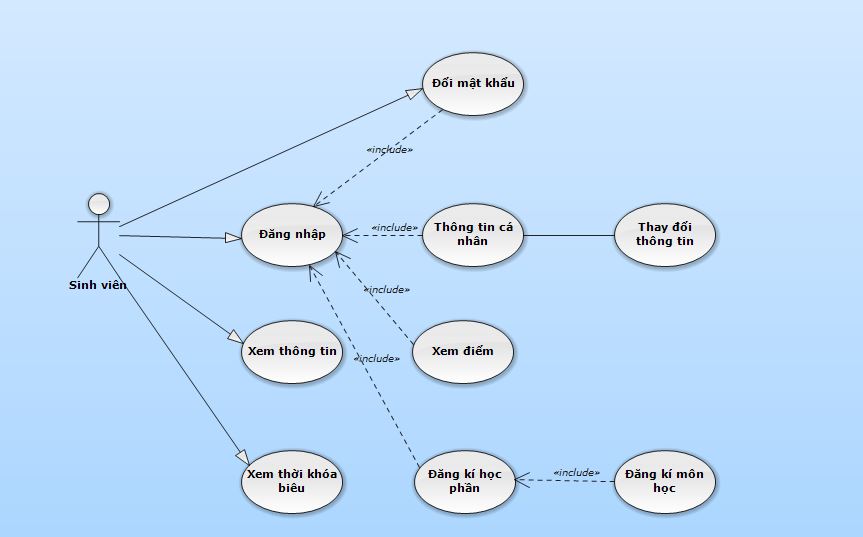
1. Biểu đồ Use Case

Hệ thống được chia làm 2 loại Actor:

* + - Quản lí: là người thực hiện chức năng quản lí thông tin như cập nhật điểm, các thông tin của từng cá nhân sinh viên
    - Sinh viên: có thể thực hiện đăng kí học, xem thời khóa biểu, xem điểm môn học...

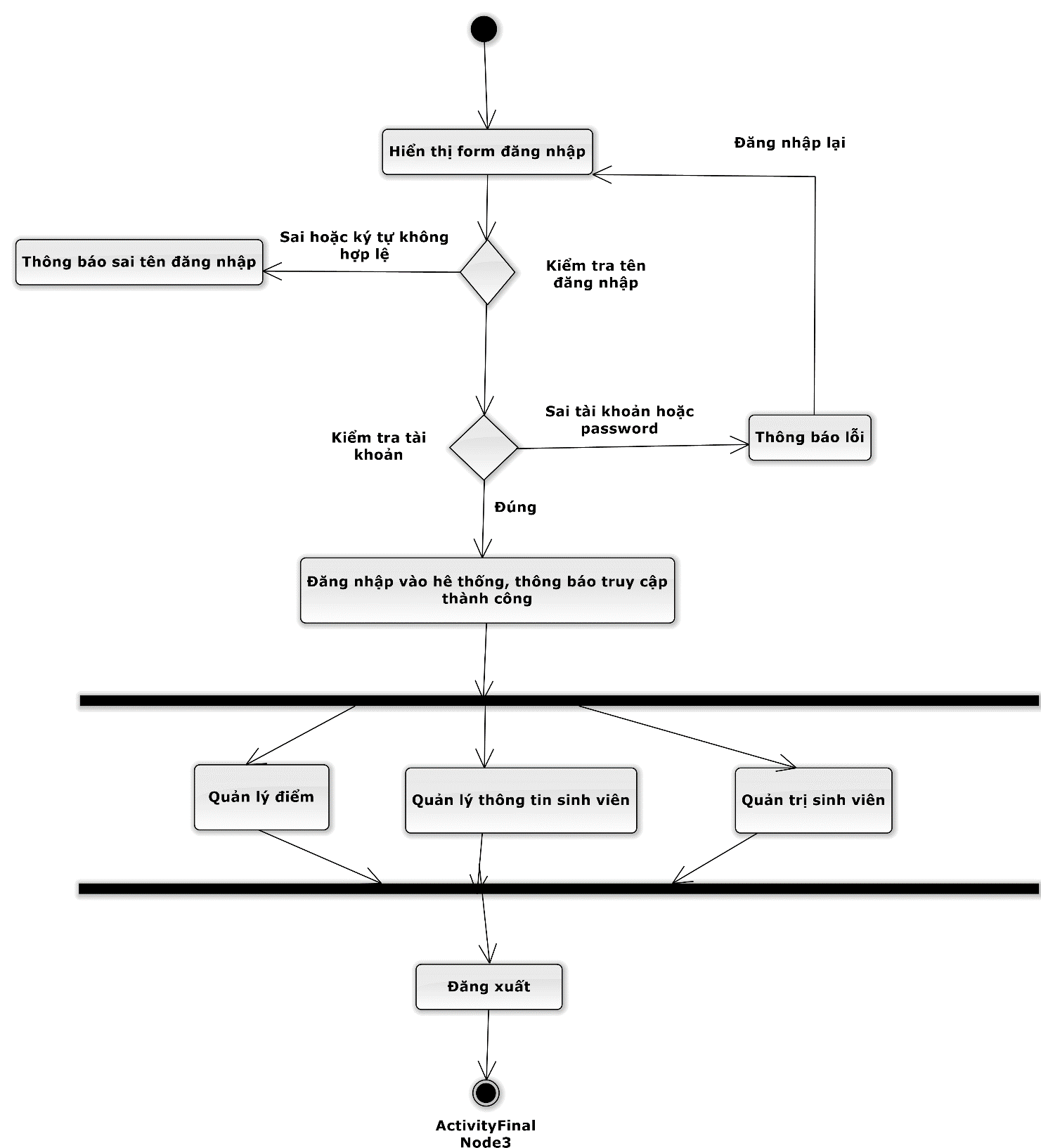


Hình 1: Biểu đồ use-case hệ thống



Hình 2: Biểu đồ use-case sinh viên

1. Biểu đồ hoạt động



Hình 3: Biểu đồ hoạt động

Ban đầu hiển thị form đăng nhập, người quản trị sẽ nhập tài khoản và password:

Nếu tên tài khoản sai hoặc có chứa các ký tự không hợp lệ (chống inject-SQL, chống người dùng nhập sai sửa mất công nhiều …) thông báo nhập thông tin sai

Tiếp tục, sau khi người quản trị nhập xong thông tin đăng nhập, hệ thống kiểm tra có đúng hợp lệ hay không trong cơ sở dữ liệu.

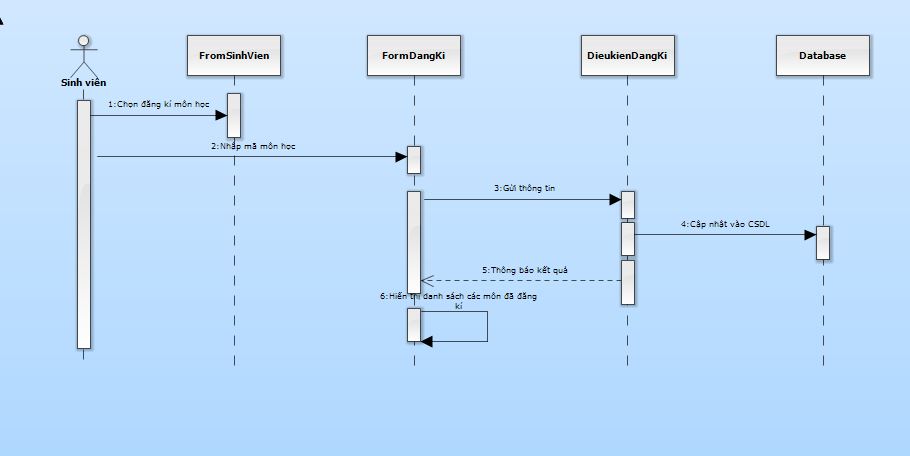
* Nếu sai, thông báo lỗi, trở về đăng nhập lại
* Nếu đúng, thông báo đăng nhập thành công, truy cập vào hệ thống, lựa chọn các chức năng cần thực hiện: Quản lý điểm, quản lý thông tin sinh viên, quản trị sinh viên.

Đăng xuất.

Kết thúc đăng nhập.

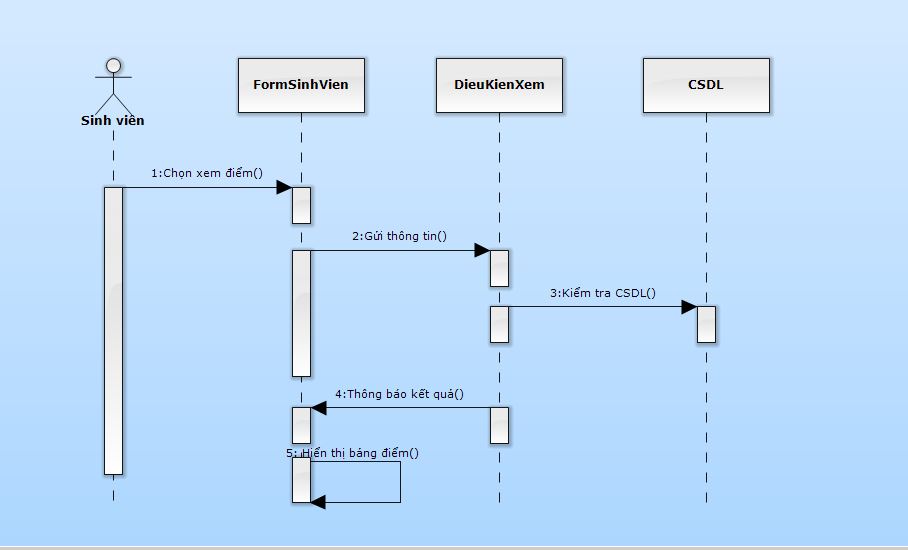
1. Biểu đồ tương tác

* Quá trình đăng kí môn học



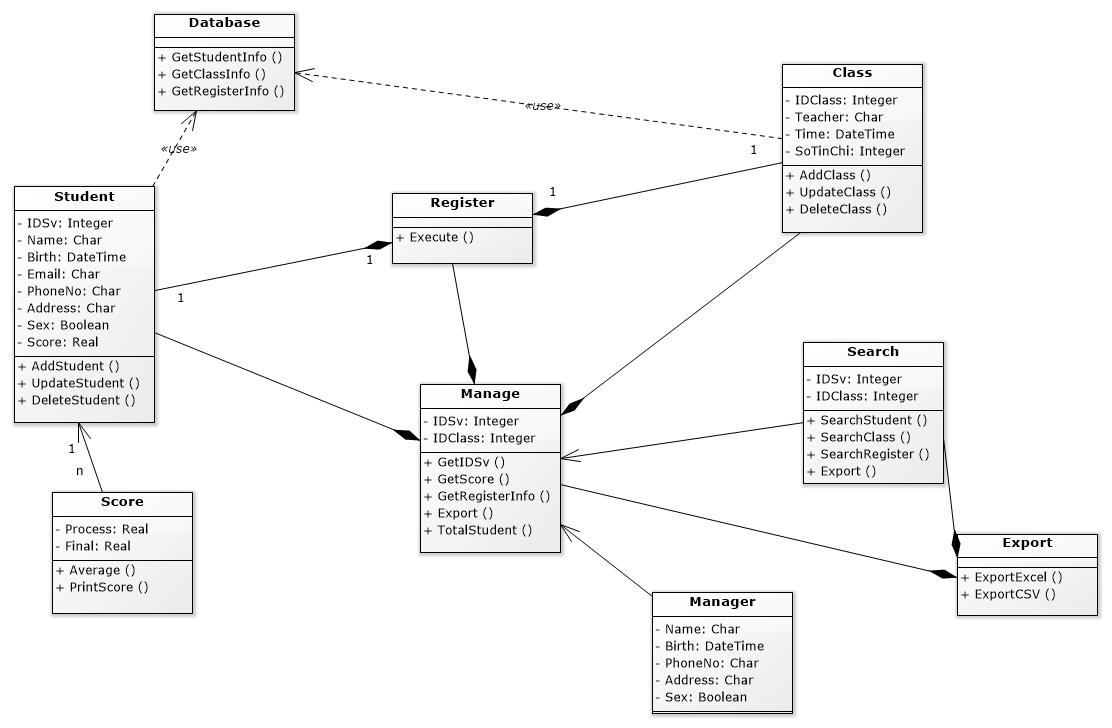
Hình 4: Biểu đồ tương tác quá trình đăng ký môn học

* Quá trình xem điểm



Hình 5: Biểu đồ tương tác quá trình xem điểm

1. Biểu đồ lớp



Hình 6 : Biểu đồ lớp

**CHƯƠNG 4: LỰA CHỌN CÔNG NGHỆ**

* + - 1. Sử dụng phương pháp lập trình hướng đối tượng

Lập trình hướng đối tượng (gọi tắt là OOP, từ chữ [Anh ngữ](https://vi.wikipedia.org/wiki/Ti%E1%BA%BFng_Anh) *object-oriented programming*), hay còn gọi là lập trình định hướng đối tượng, là kĩ thuật lập trình hỗ trợ công nghệ đối tượng. OOP được xem là giúp tăng năng suất, đơn giản hóa độ phức tạp khi bảo trì cũng như mở rộng phần mềm bằng cách cho phép [lập trình viên](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_vi%C3%AAn) tập trung vào các đối tượng phần mềm ở bậc cao hơn. Ngoài ra, nhiều người còn cho rằng OOP dễ tiếp thu hơn cho những người mới học về [lập trình](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_m%C3%A1y_t%C3%ADnh) hơn là các phương pháp trước đó.

Những đối tượng trong một ngôn ngữ OOP là các kết hợp giữa mã và dữ liệu mà chúng được nhìn nhận như là một đơn vị duy nhất. Mỗi đối tượng có một tên riêng biệt và tất cả các tham chiếu đến đối tượng đó được tiến hành qua tên của nó. Như vậy, mỗi đối tượng có khả năng nhận vào các thông báo, xử lý dữ liệu (bên trong của nó), và gửi ra hay trả lời đến các đối tượng khác hay đến môi trường. Hiện nay các ngôn ngữ OOP phổ biến nhất đều tập trung theo phương pháp phân lớp trong đó có C++, Java, C# và Visual Basic.NET. Ngôn ngữ OOP hay còn gọi là ngôn ngữ lập trình hướng đối tượng, là một phương pháp thiết kế và phát triển phần mềm dựa trên kiến trúc lớp và đối tượng.

* + - 1. Ưu điểm của mô hình hướng đối tượng:
* Thông qua nguyên lý kế thừa chúng ta có thể loại bỏ những đoạn mã trùng lặp, dư thừa trong quá trình mô tả đối tượng.
* Rút ngắn thời gian xây dựng hệ thống và tăng năng suất lao động.
* Nguyên lý che giấu thông tin giúp bảo vệ an toàn chương trình trước sự truy nhập tù tiện của các đối tượng khác.
* Có thể xây dựng được ánh sạ đối tượng của bài toán vạ đối tượng của chương trình.
* Xây dựng được mô hình phù hợp với thực tế hơn.
* Những hệ thống đói tượng dễ mở, nâng cấp thành những hệ thống lớn hơn.
* Kĩ thuật truyền thông điệp giúp trao đổi thông tin giữa các đối tượng giúp cho việc mô tả giao diện với các hệ thống bên ngoài đơn giản hơn.
* Có thể quản lý độ phức tạp của các sảm phẩm phần mềm.
* Tùy thuộc vào dự án tin học mà ta có thể sử dụng các tính chất khác nhau của lập trình hướng đối tượng.

Phương pháp phân tích và thiết kế hướng đối tượng thực hiện theo các thuật ngữ và khái niệm của phạm vi lĩnh vực ứng dụng (tức là của doanh nghiệp hay đơn vị mà hệ thống tương lai cần phục vụ), nên nó tạo sự tiếp cận tương ứng giữa hệ thống và vấn đề thực ngoài đời. Trong ví dụ bán xe ô tô, mọi giai đoạn phân tích thiết kế và thực hiện đều xoay quanh các khái niệm như khách hàng, nhân viên bán hàng, xe ô tô, … Vì quá trình phát triển phần mềm đồng thời là quá trình cộng tác của khách hàng/người dùng, nhà phân tích, nhà thiết kế, nhà phát triển, chuyên gia lĩnh vực, chuyên gia kỹ thuật, ... nên lối tiếp cận này khiến cho việc giao tiếp giữa họ với nhau được dễ dàng hơn.

Một trong những ưu điểm quan trọng bậc nhất của phương pháp phân tích và thiết kế hướng đối tượng là tính tái sử dụng: bạn có thể tạo các thành phần (đối tượng) một lần và dùng chúng nhiều lần sau đó. Giống như việc bạn có thể tái sử dụng các khối xây dựng (hay bản sao của nó ) trong một toà lâu đài, một ngôi nhà ở, một con tàu vũ trụ, bạn cũng có thể tái sử dụng các thành phần (đối tượng) căn bản trong các thiết kế hướng đối tượng cũng như code của một hệ thống kế toán, hệ thống kiểm kê, hoặc một hệ thống đặt hàng.

Vì các đối tượng đã được thử nghiệm kỹ càng trong lần dùng trước đó, nên khả năng tái sử dụng đối tượng có tác dụng giảm thiểu lỗi và các khó khăn trong việc bảo trì, giúp tăng tốc độ thiết kế và phát triển phần mềm.

Phương pháp hướng đối tượng giúp chúng ta xử lý các vấn đề phức tạp trong phát triển phần mềm và tạo ra các thế hệ phần mềm có khả năng thích ứng và bền chắc.

* + - 1. Phân tích hướng đối tượng (Object Oriented Analysis - OOA)

Là giai đọan phát triển một mô hình chính xác và súc tích của vấn đề, có thành phần là các đối tượng và khái niệm đời thực, dễ hiểu đối với người sử dụng.

Trong giai đoạn OOA, vấn đề được trình bày bằng các thuật ngữ tương ứng với các đối tượng có thực. Thêm vào đó, hệ thống cần phải được định nghĩa sao cho người không chuyên Tin học có thể dễ dàng hiểu được.

Dựa trên một vấn đề có sẵn, nhà phân tích cần ánh xạ các đối tượng hay thực thể có thực như khách hàng, ô tô, người bán hàng, … vào thiết kế để tạo ra được bản thiết kế gần cận với tình huống thực. Mô hình thiết kế sẽ chứa các thực thể trong một vấn đề có thực và giữ nguyên các mẫu hình về cấu trúc, quan hệ cũng như hành vi của chúng. Nói một cách khác, sử dụng phương pháp hướng đối tượng chúng ta có thể mô hình hóa các thực thể thuộc một vấn đề có thực mà vẫn giữ được cấu trúc, quan hệ cũng như hành vi của chúng.

* + - 1. Thiết kế hướng đối tượng (Object Oriented Design - OOD)

Là giai đoạn tổ chức chương trình thành các tập hợp đối tượng cộng tác, mỗi đối tượng trong đó là thực thể của một lớp. Các lớp là thành viên của một cây cấu trúc với mối quan hệ thừa kế.

Mục đích của giai đoạn OOD là tạo thiết kế dựa trên kết quả của giai đoạn OOA, dựa trên những quy định phi chức năng, những yêu cầu về môi trường, những yêu cầu về khả năng thực thi, .... OOD tập trung vào việc cải thiện kết quả của OOA, tối ưu hóa giải pháp đã được cung cấp trong khi vẫn đảm bảo thoả mãn tất cả các yêu cầu đã được xác lập.

Trong giai đoạn OOD, nhà thiết kế định nghĩa các chức năng, thủ tục (operations), thuộc tính (attributes) cũng như mối quan hệ của một hay nhiều lớp (class) và quyết định chúng cần phải được điều chỉnh sao cho phù hợp với môi trường phát triển. Đây cũng là giai đoạn để thiết kế ngân hàng dữ liệu và áp dụng các kỹ thuật tiêu chuẩn hóa.

Về cuối giai đoạn OOD, nhà thiết kế đưa ra một loạt các biểu đồ (diagram) khác nhau. Các biểu đồ này có thể được chia thành hai nhóm chính là Tĩnh và động. Các biểu đồ tĩnh biểu thị các lớp và đối tượng, trong khi biểu đồ động biểu thị tương tác giữa các lớp và phương thức hoạt động chính xác của chúng. Các lớp đó sau này có thể được nhóm thành các gói (Packages) tức là các đơn vị thành phần nhỏ hơn của ứng dụng.

* + - 1. Lập trình hướng đối tượng (Object Oriented Programming - OOP)

Giai đoạn xây dựng phần mềm có thể được thực hiện sử dụng kỹ thuật lập trình hướng đối tượng. Đó là phương thức thực hiện thiết kế hướng đối tượng qua việc sử dụng một ngôn ngữ lập trình có hỗ trợ các tính năng hướng đối tượng. Một vài ngôn ngữ hướng đối tượng thường được nhắc tới là C++ và Java. Kết quả chung cuộc của giai đoạn này là một loạt các code chạy được, nó chỉ được đưa vào sử dụng sau khi đã trải qua nhiều vòng quay của nhiều bước thử nghiệm khác nhau.

* + - 1. Sử dụng cơ sở dữ liệu

Trong nhiều năm gần đây, thuật ngữ Cơ sở dữ liệu - Database đã trở nên quen thuộc trong nhiều lĩnh vực. Các ứng dụng tin học vào quản lý ngày càng nhiều và đa dạng, hầu hết các lĩnh vực kinh tế, xã hội… đều đã ứng dụng các thành tựu mới của tin học vào phục vụ công tác chuyên môn của mình. Chính vì lẽ đó mà ngày càng nhiều người quan tâm đến thiết kế, xây dựng và ứng dụng cơ sở dữ liệu (CSDL).

* + - 1. *Khái niệm*

CSDL là tập hợp các dữ liệu có cấu trúc và liên quan với nhau được lưu trữ trên máy tính, được nhiều người sử dụng và được tổ chức theo một mô hình.

Trong khái niệm này, chúng ta cần nhấn mạnh, CSDL là tập hợp các thông tin có tính chất hệ thống, không phải là các thông tin rời rạc, không có liên quan với nhau. Các thông tin này phải có cấu trúc và tập hợp các thông tin này phải có khả năng đáp ứng nhu cầu khai thác của nhiều người sử dụng một cách đồng thời. Đó cũng chính là đặc trưng của CSDL.

* + - 1. *Ưu điểm*

Từ khái niệm trên, ta thấy rõ ưu điểm nổi bật của CSDL là

* Giảm sự trùng lặp thông tin xuống mức thấp nhất và do đó đảm bảo được tính nhất quán và toàn vẹn dữ liệu (Cấu trúc của cơ sở dữ liệu được định nghĩa một lần. Phần định nghĩa cấu trúc này gọi là meta-data, và được Catalog của HQTCSDL lưu trữ).
* Đảm bảo sự độc lập giữa dữ liệu và chương trình ứng dụng (Insulation between programs and data): Cho phép thay đổi cấu trúc, dữ liệu trong cơ sở dữ liệu mà không cần thay đổi chương trình ứng dụng.
* Trừu tượng hoá dữ liệu (Data Abstraction): Mô hình dữ liệu được sử dụng để làm ẩn lưu trữ vật lý chi tiết của dữ liệu, chỉ biểu diễn cho người sử dụng mức khái niệm của cơ sở dữ liệu.
* Nhiều khung nhìn (multi-view) cho các đối người dùng khác nhau: Đảm bảo dữ liệu có thể được truy xuất theo nhiều cách khác nhau. Vì yêu cầu của mỗi đối tượng sử dụng CSDL là khác nhau nên tạo ra nhiều khung nhìn vào dữ liệu là cần thiết.
* Đa người dùng (multi-user): Khả năng chia sẻ thông tin cho nhiều người sử dụng và nhiều ứng dụng khác nhau.

**CHƯƠNG 5:** **KIỂM THỬ**

* + - 1. Kiểm thử hộp đen

Kiểm thử dựa vào chức năng của chương trình và kinh nghiệm để đoán lỗi.

1. Chức năng quản lí thông tin

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mô tả | Thủ tục kiểm thử | | Kết quả thực tế đạt được | Kết luận |
| Bước thực hiện | Kết quả kỳ vọng |
| 1 | Đăng nhập tài khoản vào hệ thống | B1: Nhập tài khoản  B2: Nhập mật khẩu  (Yêu cầu: Tài khoản hoặc mật khẩu nhập vào là không đúng) | Thông báo nhập sai tài khoản, hoặc mật khẩu | Có thông báo nhập sai tài khoản hoặc mật khẩu, yêu cầu nhập lại | OK |
| Không có thông báo gì *hoặc* các lỗi khác. | FAIL |
| 2 | Đăng kí tài khoản sinh viên | B1: Nhập mã sinh viên  B2: Nhập tài khoản đăng kí  (thường đặt tên tài khoản trùng với mã sinh viên để dễ quản lí)  B3: Nhập mật khẩu  B4: Nhập email  (Yêu cầu: Tài khoản hoặc mã sinh viên hoặc mật khẩu nhập vào không hợp lệ) | Thông báo tài khoản (tên tài khoản hoặc mật khẩu) không hợp lệ hoặc email đã có người sử dụng hoặc mã số sinh viên không đúng | Có thông báo tài khoản hoặc email không hợp lệ hoặc sai mã số sinh viên | OK |
| Không có thông báo gì *hoặc* các lỗi khác. | FAIL |
| 3 | Kiểm tra thông tin sinh viên | B1: Nhập mã sinh viên hoặc tài khoản  B2: Chọn chức năng kiểm tra thông tin  (Yêu cầu: Mã sinh viên hoặc tài khoản sai) | Báo lỗi không tìm được sinh viên này | Có thông báo không tìm được sinh viên này | OK |
| Không có thông báo gì *hoặc* các lỗi khác. | FAIL |
| 4 | Xóa sinh viên | B1: Nhập mã sinh viên hoặc tài khoản không đúng  B2: Nhập mã xác nhận  (Yêu cầu: Nhập mã sinh viên hoặc tài khoản hoặc mã xác nhận không đúng) | Thông báo không tìm được sinh viên này | Có thông báo không tìm được sinh viên này chương trình yêu cầu nhập lại mã | OK |
| Không có thông báo gì *hoặc* các lỗi khác. | FAIL |
| 5 | Sửa thông tin sinh viên | B1: Nhập tài khoản  B2: Nhập mật khẩu  B3: Sửa thông tin cá nhân  (Yêu cầu: Xóa thông tin về địa chỉ) | Thông báo thông tin chưa được cập nhật | Có thông báo thông tin chưa được cập nhật | OK |
| Thông báo thông tin đã được cập nhật | FAIL |

1. Chức năng quản lí đăng kí học tập

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mô tả | Thủ tục kiểm thử | | Kết quả thực tế đạt được | Kết luận |
| Bước thực hiện | Kết quả kỳ vọng |
| 1 | Đăng kí học phần | B1: Chọn chức năng đăng ký học phần  B2: Nhập mã HP: không đúng hoặc không được mở  B3: Chọn học kỳ: tiếp theo  B4: Gửi đăng ký | Thông báo nhập sai mã học phần | Có thông báo nhập sai mã học phần yêu cầu nhập lại | OK |
| Không có thông báo gì *hoặc* các lỗi khác. | FAIL |
| 2 | Đăng kí lớp học bị hết chỗ | B1: Chọn chức năng đăng ký lớp học  B2: Nhập mã hoc phần  B3: Chọn đăng kí lớp đã đầy  B4: Gửi đăng ký | Thông báo lớp đã đăng kí thành công lớp đó | Có thông báo đăng kí thành công | OK |
| Không có thông báo gì *hoặc* các lỗi khác. | FAIL |
| 3 | Đăng kí bị trùng thời khóa biểu | B1: Chọn chức năng đăng ký lớp học  B2: Chọn đăng kí 2 lớp có thời khóa biểu trùng nhau  B3: Gửi đăng ký | Thông báo đã đăng kí thành công | Có thông báo đăng kí thành công | OK |
| Không có thông báo gì *hoặc* các lỗi khác | FAIL |
| 4 | Đăng kí số lượng tín chỉ quá giới hạn | B1: Chọn chức năng đăng ký lớp học  B2: Đăng kí 25 tín chỉ  B4: Gửi đăng ký | Thông báo đăng ký học phần thành công | Có thông báo đăng ký học phần thành côngs | OK |
| Không có thông báo gì *hoặc* thông báo đăng ký không thành công *hoặc* các lỗi khác. | FAIL |

1. Chức năng cập nhật điểm

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mô tả | Thủ tục kiểm thử | | Kết quả thực tế đạt được | Kết luận |
| Bước thực hiện | Kết quả kỳ vọng |
| 1 | Nhập sai định dạng của điểm | B1: Đăng nhập tài khoản quản lí  B2: Nhập mã lớp thi  B3: Chọn học kỳ tương ứng  B4: Nhập điểm cho sinh viên (nhập sai dạng điêm ví dụ như 6.9) | Không có thông báo lỗi do nhập sai điểm | Không có thông báo lỗi do nhập sai điểm | OK |
| Có thông báo lỗi do nhập điểm, hoặc do lỗi khác | FAIL |
| 2 | Nhập sai mã lớp thi | B1: Đăng nhập tài khoản quản lí  B2: Nhập mã lớp thi không đúng  B3: Chọn học kỳ tương ứng  B4: Nhập điểm cho sinh | Thông báo mã lớp thi không đúng | Có thông báo mã lớp thi không đúng, yêu cầu nhập lại | OK |
| Không có thông báo gì *hoặc* các lỗi khác. | FAIL |
| 3 | Xem điểm | B1: Đăng nhập tài khoản sinh viên  B2: Chọn chức năng bảng điểm cá nhân | Hiển thị đánh giá điểm trung bình sai | Hiển thị đánh giá điểm trung bình sai | OK |
| Hiển thị đánh giá điểm trung bình sai | FAIL |

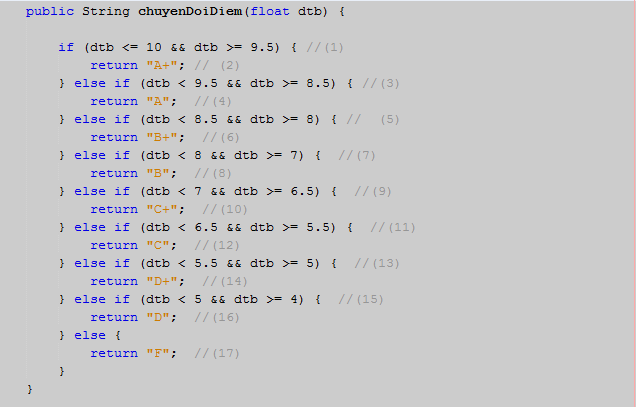
1. Chức năng quản lí lớp

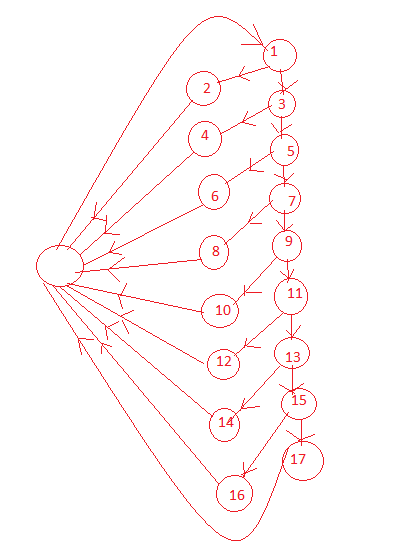
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Mô tả | Thủ tục kiểm thử | | Kết quả thực tế đạt được | Kết luận |
| Bước thực hiện | Kết quả kỳ vọng |
| 1 | Sinh viên không đúng lớp | B1: Đăng nhập tài khoản quản lí  B2: Nhập mã lớp sinh viên  B3: Chọn khóa học  B4: Nhập điểm cho sinh viên (nhập sai dạng điêm ví dụ như 6.9) | Hiển thị ít nhất một sinh viên có trường “lớp” không đúng | Hiển thị ít nhất một sinh viên có trường “lớp” không đúng | OK |
| Hiển thị đúng hoặc có lỗi | FAIL |

* + - 1. Kiểm thử hộp trắng

Đối tượng được kiểm thử là 1 thành phần phần mềm (TPPM). TPPM có thể là 1 hàm chức năng, 1 module chức năng, 1 phân hệ chức năng… Kiểm thử hộp trắng dựa vào thuật giải cụ thể, vào cấu trúc dữ liệu bên trong của đơn vị phần mềm cần kiểm thử để xác định đơn vị phần mềm đó có thực hiện đúng không.

Kiểm thử đoạn code quy đổi điểm số ra điểm chữ





Có 18 nút và 26 cung nên độ phức tạp là 10

Hay số lộ trình độc lập là 10

Số test phải thực hiện là 10

# **CHƯƠNG 6. KẾT LUẬN**

* Ưu điểm:
  + Sử dụng việc quản lí, cập nhật, tìm kiếm dễ dàng, nhanh chóng
  + Đáp ứng được các chức năng cần thiết
  + Lưu trữ dữ liệu dễ dàng
* Nhược điểm
* Chương trình còn đơn giản, đồ họa chưa đẹp,
* Hướng phát triển

Nhóm mới thực hiện tạo một chương trình trên java và có định hướng phát triển trên nhiều môi trường nhiều nền tảng như web hoặc các ứng dụng điện thoại

# **CHƯƠNG 7. CÔNG CỤ LÀM VIỆC NHÓM**

Nhóm em sử dụng Github và Office 365 để trao đổi tài liệu, tiến độ công việc giúp có thể hoàn thành đề tài đúng thời hạn.

Link GitHub: <https://github.com/monkey826/Project>

Một số hình ảnh các thành viên upload tài liệu:

