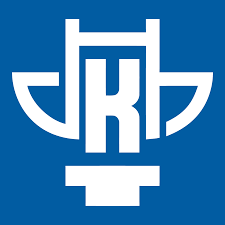
**Trường Đại Học Kiến Trúc Hà Nội**

Khoa Công nghệ thông tin



**…**

**ĐỒ ÁN MÔN NGÔN NGỮ C# VÀ CÔNG NGHỆ .NET**

**ĐỀ TÀI:** **NGHIÊN CỨU VÀ XÂY DỰNG MÔ HÌNH QUẢN LÝ SINH VIÊN**

Sinh viên thực hiện: 1. Trần Trọng Nghĩa (NT)

2. Nguyễn Hồng Phúc

3. Đặng Thanh Tùng

Giảng viên hướng dẫn: ThS.Nguyễn Hồng Thanh

Hà Nội, 3/2024

Mục lục

[Danh mục hình ảnh 5](#_Toc163558304)

[Danh mục bảng biểu 7](#_Toc163558305)

[MỞ ĐẦU 8](#_Toc163558306)

[Chương 1: Tổng quan 9](#_Toc163558307)

[1.1 Tổng quan về ngôn ngữ c# và công nghệ .NET 9](#_Toc163558308)

[1.1.1 Khái niệm về ngôn ngữ c# và công nghệ .NET 9](#_Toc163558309)

[1.1.2 Chức năng công của ngôn ngữ c# và công nghệ .NET 9](#_Toc163558310)

[1.1.3 Ứng dụng 10](#_Toc163558311)

[1.1.4 Lợi ích và hạn chế 10](#_Toc163558312)

[1.2 Tổng quan nghiên cứu và xây dựng hệ thống quản lý sinh viên 12](#_Toc163558313)

[Chương 2: Đặc tả yêu cầu 13](#_Toc163558314)

[2.1 Xác định yêu cầu chức năng 13](#_Toc163558315)

[2.1.1 Yêu cầu chức năng 13](#_Toc163558316)

[2.1.2 Yêu cầu phi chức năng 13](#_Toc163558317)

[2.2. Mô hình đặc tả sơ đồ use case đề tài nghiên cứu và xây dựng mô hình quản lý sinh viên 13](#_Toc163558318)

[2.3 Đặc tả use case nghiên cứu và xây dựng mô hình quản lý sinh viên 14](#_Toc163558319)

[2.3.1 Use case đăng nhập 14](#_Toc163558320)

[2.3.2 Use case quản lý sinh viên 14](#_Toc163558321)

[2.3.3 Use case quản lý khóa học 15](#_Toc163558322)

[2.4.3 Use case quản lý điểm 16](#_Toc163558323)

[2.4.4 Use case quản lý điểm rèn luyện 17](#_Toc163558324)

[Chương 3: Thiết kế 19](#_Toc163558325)

[3.1 Biểu đồ hoạt động 19](#_Toc163558326)

[3.1.1 Đăng nhập 19](#_Toc163558327)

[3.1.2 Quản lý sinh viên 19](#_Toc163558328)

[3.1.3 Quản lý khóa học 21](#_Toc163558329)

[3.1.4 Quản lý điểm 23](#_Toc163558330)

[3.1.5 Quản lý điểm rèn luyện 25](#_Toc163558331)

[3.2 Biểu đồ tuần tự 27](#_Toc163558332)

[3.2.1 Đăng nhập 28](#_Toc163558333)

[3.2.2 Quản lý sinh viên 28](#_Toc163558334)

[3.2.3 Quản lý khóa học 30](#_Toc163558335)

[3.2.4 Quản lý điểm 32](#_Toc163558336)

[3.2.5 Quản lý điểm rèn luyện 34](#_Toc163558337)

[Chương 4: Thiết kế cơ sở dữ liệu 37](#_Toc163558338)

[4.1 Mô tả thực thể kết hợp 37](#_Toc163558339)

[4.2 Mô tả các bảng 38](#_Toc163558340)

[4.2.1 Bảng dữ liệu sinh viên: (student) 38](#_Toc163558341)

[4.2.2 Bảng số liệu khóa học: (course) 39](#_Toc163558342)

[4.2.3 Bảng dữ liệu điểm: (score) 39](#_Toc163558343)

[4.2.4 Bảng dữ liệu đăng nhập: (user) 40](#_Toc163558344)

[4.2.5 Bảng dữ liệu điểm rèn luyện: (pointtraining) 40](#_Toc163558345)

[4.3 Kết nối giữa các bảng CSDL 41](#_Toc163558346)

[Chương 5: Hình ảnh demo chương trình 42](#_Toc163558347)

[5.1 Đăng nhập 42](#_Toc163558348)

[5.2 Trang chủ 42](#_Toc163558349)

[5.3 Thêm sinh viên 43](#_Toc163558350)

[5.4 Quản lý sinh viên 43](#_Toc163558351)

[5.5 Thêm khóa học 44](#_Toc163558352)

[5.6 Quản lý khóa học 44](#_Toc163558353)

[5.7 Thêm điểm mới 45](#_Toc163558354)

[5.8 Quản lý điểm 45](#_Toc163558355)

[5.9 Quản lý thêm điểm rèn luyện 46](#_Toc163558356)

[5.10 Quản lý điểm rèn luyện 46](#_Toc163558357)

[Chương 6: Kiểm thử và bảo trì 47](#_Toc163558358)

[6.1. Kiểm tra thử: 47](#_Toc163558359)

[6.2 Bảo trì 47](#_Toc163558360)

[6.3 Kết luận: 48](#_Toc163558361)

[Kết luận 49](#_Toc163558362)

[Tài liệu tham khảo: 50](#_Toc163558363)

[Phân công công việc 51](#_Toc163558364)

# Danh mục hình ảnh

[Hình 2.2.1 : Sơ đồ use case tổng quan quản lý sinh viên 14](#_Toc163558365)

[Hình 3.1.1 : Biểu đồ đăng nhập của quản trị viên 19](#_Toc163558366)

[Hình 3.1.2.1 : Biểu đồ hoạt động đăng kí thông tin sinh viên 19](#_Toc163558367)

[Hình 3.1.2.2 : Biểu đồ hoạt động chỉnh sửa thông tin sinh viên 20](#_Toc163558368)

[Hình 3.1.2.3 : Biểu đồ hoạt động xóa thông tin sinh viên 20](#_Toc163558369)

[Hình 3.1.2.4 : Biểu đồ hoạt động tìm kiếm thông tin sinh viên 21](#_Toc163558370)

[Hình 3.1.3.1 : Biểu đồ hoạt động thêm thông tin khóa học 21](#_Toc163558371)

[Hình 3.1.3.2 : Biểu đồ hoạt động sửa thông tin khóa học 22](#_Toc163558372)

[Hình 3.1.3.3 : Biểu đồ hoạt động xóa thông tin khóa học 22](#_Toc163558373)

[Hình 3.1.3.4 : Biểu đồ hoạt động tìm kiếm thông tin khóa học 23](#_Toc163558374)

[Hình 3.1.4.1 : Biểu đồ hoạt động thêm điểm. 23](#_Toc163558375)

[Hình 3.1.4.2 : Biểu đồ hoạt động sửa điểm 24](#_Toc163558376)

[Hình 3.1.4.3 : Biểu đồ hoạt động xóa điểm 24](#_Toc163558377)

[Hình 3.1.4.4 : Biểu đồ hoạt động tìm kiếm điểm 25](#_Toc163558378)

[Hình 3.1.5.1 : Biểu đồ hoạt động thêm điểm rèn luyện 25](#_Toc163558379)

[Hình 3.1.5.2 : Biểu đồ hoạt động sửa điểm rèn luyện 26](#_Toc163558380)

[Hình 3.1.5.3 : Biểu đồ hoạt động xóa điểm rèn luyện 26](#_Toc163558381)

[Hình 3.1.5.4 : Biểu đồ hoạt động tìm kiếm điểm rèn luyện 27](#_Toc163558382)

[Hình 3.2.2.1 : Biểu đồ tuần tự đăng kí thông tin sinh viên 28](#_Toc163558383)

[Hình 3.2.2.2 : Biểu đồ tuần tự chỉnh sửa thông tin sinh viên 29](#_Toc163558384)

[Hình 3.2.2.3 : Biểu đồ tuần tự xóa thông tin sinh viên 29](#_Toc163558385)

[Hình 3.2.2.4 : Biểu đồ tuần tự tìm kiếm thông tin sinh viên 30](#_Toc163558386)

[Hình 3.2.3.1 : Biểu đồ tuần tự thêm thông tin khóa học 30](#_Toc163558387)

[Hình 3.2.3.2 : Biểu đồ tuần tự chỉnh sửa thông tin khóa học 31](#_Toc163558388)

[Hình 3.2.3.3 : Biểu đồ tuần tự xóa thông tin khóa học 31](#_Toc163558389)

[Hình 3.2.3.4 : Biểu đồ tuần tự tìm kiếm thông tin khóa học 32](#_Toc163558390)

[Hình 3.2.4.1 : Biểu đồ tuần tự thêm điểm 32](#_Toc163558391)

[Hình 3.2.4.2 : Biểu đồ tuần tự sửa điểm 33](#_Toc163558392)

[Hình 3.2.4.3 : Biểu đồ tuần tự xóa điểm 33](#_Toc163558393)

[Hình 3.2.4.4 :Biểu đồ tuần tự tìm kiếm điểm 34](#_Toc163558394)

[Hình 4.3.1 : Bảng CSDL của mô hình quản lý sinh viên 41](#_Toc163558395)

[Hình 5.1 : Hình ảnh đăng nhập 42](#_Toc163558396)

[Hình 5.2 : Hình ảnh trang chủ 42](#_Toc163558397)

[Hình 5.3 : Hình ảnh thêm sinh viên 43](#_Toc163558398)

[Hình 5.4 : Hình ảnh quản lý sinh viên 43](#_Toc163558399)

[Hình 5.5 : Hình ảnh thêm khóa học 44](#_Toc163558400)

[Hình 5.6 : Hình ảnh quản lý khóa học 44](#_Toc163558401)

[Hình 5.7 : Hình ảnh thêm điểm 45](#_Toc163558402)

[Hình 5.8 : Hình ảnh quản lý điểm 45](#_Toc163558403)

[Hình 5.9: Hình ảnh thêm điểm rèn luyện 46](#_Toc163558404)

# Danh mục bảng biểu

[Bảng 4.2.1 : Bảng dữ liệu sinh viên 38](#_Toc163558405)

[Bảng 4.2.2 : Bảng dữ liệu khóa học 39](#_Toc163558406)

[Bảng 4.2.3 : Bảng dữ liệu điểm 39](#_Toc163558407)

[Bảng 4.2.4 : Bảng dữ liệu đăng nhập 40](#_Toc163558408)

[Bảng 4.2.5 : Bảng dữ liệu điểm rèn luyện 40](#_Toc163558409)

# MỞ ĐẦU

Trong thời đại số hóa ngày nay, việc quản lý sinh viên đòi hỏi sự linh hoạt, hiệu quả và tính toàn diện từ các cơ sở giáo dục. Cùng với sự phát triển không ngừng của công nghệ thông tin, việc áp dụng các giải pháp công nghệ vào quản lý sinh viên không chỉ là một xu hướng mà còn là một yếu tố quyết định đối với sự thành công của các trường đại học, cao đẳng.

Đề tài "Nghiên cứu và xây dựng mô hình quản lý sinh viên bằng ngôn ngữ lập trình C#" đã ra đời với mục tiêu chính là tạo ra một hệ thống quản lý sinh viên tiên tiến, linh hoạt và dễ dàng sử dụng, đồng thời đem lại lợi ích to lớn cho cả sinh viên và nhà trường. Trong bối cảnh mà việc tối ưu hóa quá trình quản lý và tương tác giữa sinh viên và cơ sở giáo dục trở nên cần thiết hơn bao giờ hết, đề tài này sẽ đề xuất và phát triển các giải pháp công nghệ nhằm cải thiện hiệu quả quản lý, tăng cường trải nghiệm học tập và tạo ra một môi trường học tập động lực.

Chúng tôi sẽ khám phá chi tiết các yếu tố quan trọng trong quản lý sinh viên, từ việc lập kế hoạch học tập, giám sát tiến trình học tập đến tương tác giữa sinh viên và giảng viên. Bằng cách sử dụng ngôn ngữ lập trình C# - một trong những công nghệ phổ biến và mạnh mẽ trong lĩnh vực phát triển phần mềm, chúng tôi sẽ xây dựng một hệ thống quản lý sinh viên linh hoạt, dễ mở rộng và có khả năng tích hợp với các hệ thống khác.

Hy vọng rằng, thông qua đề tài này, chúng tôi sẽ đóng góp một phần nhỏ vào sự phát triển của giáo dục và nâng cao chất lượng quản lý sinh viên, từ đó tạo ra một môi trường học tập tiên tiến và phát triển hơn cho tương lai.

Chương 1: Tổng quan

1.1 Tổng quan về ngôn ngữ c# và công nghệ .NET

* + 1. Khái niệm về ngôn ngữ c# và công nghệ .NET

C# (hay còn gọi là C Sharp, là ngôn ngữ lập trình hướng đối tượng, hiện đại do Microsoft phát triển như một phần của nền tảng .NET. Nó được giới thiệu lần đầu tiên vào năm 2000 và kể từ đó đã trở thành một trong những ngôn ngữ lập trình được sử dụng rộng rãi nhất trong ngành.

C# có thiết kế đơn giản, mạnh mẽ và linh hoạt. Sở hữu một cú pháp tương tự như các ngôn ngữ lập trình phổ biến, bao gồm Java và C++, giúp các nhà phát triển đã quen thuộc với các ngôn ngữ này dễ làm quen.

.NET là một nền tảng nguồn mở, miễn phí và được Microsoft phát triển để xây dựng, cũng như có thể chạy nhiều loại ứng dụng khác nhau, bao gồm ứng dụng web, phần mềm máy tính để bàn, ứng dụng dành cho thiết bị di động, game…

* + 1. Chức năng công của ngôn ngữ c# và công nghệ .NET

Đa nền tảng (Cross-Platform): Với sự ra mắt của .NET Core và sau này là .NET 5+ (và các phiên bản tiếp theo), bạn có thể viết mã C# và chạy nó trên nhiều hệ điều hành khác nhau, bao gồm Windows, macOS và Linux.

Lập trình hướng đối tượng (Object-Oriented Programming - OOP): C# hỗ trợ các nguyên tắc của lập trình hướng đối tượng như kế thừa, đóng gói, và đa hình.

Bộ thư viện phong phú: C# và .NET có một bộ thư viện rất phong phú với nhiều chức năng sẵn có giúp việc phát triển ứng dụng trở nên nhanh chóng và dễ dàng hơn.

ASP.NET: Là một framework phát triển ứng dụng web được xây dựng trên nền tảng .NET, ASP.NET cung cấp các công cụ và thư viện để xây dựng các ứng dụng web mạnh mẽ và linh hoạt.

Windows Forms và WPF: Các công nghệ này cho phép bạn xây dựng ứng dụng desktop trên nền tảng Windows với giao diện người dùng đồ họa.

LINQ (Language Integrated Query): LINQ là một tính năng của C# cho phép bạn thực hiện các truy vấn dữ liệu trong mã C# một cách dễ dàng và tự nhiên, giúp giảm thiểu lỗi và tăng tính tự động hóa.

Asynchronous Programming: C# hỗ trợ lập trình bất đồng bộ, giúp tăng hiệu suất ứng dụng bằng cách cho phép các tác vụ không đồng bộ chạy mà không chặn luồng chính.

Giao diện người dùng dựa trên XAML (Extensible Application Markup Language): Đối với ứng dụng Windows Presentation Foundation (WPF) và Universal Windows Platform (UWP), bạn có thể sử dụng XAML để xây dựng giao diện người dùng.

* + 1. Ứng dụng

Trên windows: C# với framework .NET được dùng để tạo ra các ứng dụng trên Windows như Microsoft Office, Visual Studio, Skype, Photoshop,...

Trên Web: C# hỗ trợ lập trình viên tạo các ứng dụng web nhờ sự hỗ trợ của asp.net. Với ngôn ngữ này, các ứng dụng có thể chạy mượt mà trên máy chủ.

Thành phần, điều khiển:  C# còn được ứng dụng trong xây dựng nhiều thành phần của máy chủ. Đây là một trong các ứng dụng quan trọng của ngôn ngữ lập trình C#.

1.1.4 Lợi ích và hạn chế

1.1.4.1 Lợi ích:

C Sharp là ngôn ngữ đơn giản: Ngôn ngữ này sử dụng khá đơn giản. Nếu như bạn đã sử dụng quen các ngôn ngữ như C hay C++ hoặc thậm chí là Java thì việc dùng C Sharp cũng khá giống.  
C# loại bỏ một vài sự phức tạp của những ngôn ngữ như Java và c++, bao gồm việc loại bỏ những macro, những template, đa kế thừa và lớp cơ sở ảo.

Ngôn ngữ đa nền tảng: C # nó là ngôn ngữ lập trình đa nền tảng. Chúng ta có thể xây dựng các ứng dụng .NET được triển khai trên các nền tảng Windows, Linux và Mac.

Ngôn ngữ an toàn và hiệu quả: C# là một ngôn ngữ an toàn và hiệu quả. Ngôn ngữ C# không cho phép chuyển đổi các kiểu dữ liệu có thể dẫn đến mất dữ liệu hoặc những vấn đề khác. C# giúp các nhà phát triển viết code an toàn. C# cũng tập trung vào việc viết code một cách hiệu quả.

C Sharp là ngôn ngữ hiện đại thông dụng, ít từ khóa và dễ hiểu: Nó có chứa đầy đủ các đặc tính cần có. Những khái niệm mới mẻ, mơ hồ về lập trình mà các bạn được học như xử lý ngoại lệ, những kiểu dữ liệu mở rộng, bảo mật mã nguồn đều sẽ được trang bị ngay trong C Sharp.

C Sharp là ngôn ngữ lập trình thuần hướng đối tượng: Ngôn ngữ này đáp ứng đầy đủ 4 tính chất của hướng  đối tượng như:  
– Tính trừu tượng (abstraction)  
– Tính đóng gói (encapsulation)  
– Tính đa hình (polymorphism)  
– Tính kế thừa (inheritance)

C# là ngôn ngữ mạnh mẽ: Garbage Collector: C# Tự động thu hồi vùng nhớ không dùng.

Versioning: Đảm bảo sự tương thích giữa lớp con và lớp cha.

Kiểm soát và xử lý ngoại lệ exception: Đoạn mã bị lỗi sẽ không được thực thi.

Type – safe: Không cho gán các kiểu dữ liệu khác nhau.

C Sharp là ngôn ngữ ít từ khóa: C# sử dụng giới hạn những từ khóa. Phần lớn các từ khóa được dùng để mô tả thông tin. Nhiều người nghĩ rằng một ngôn ngữ có nhiều từ khóa thì sẽ mạnh hơn. Điều này không phải sự thật, ít nhất là trong trường hợp ngôn ngữ C#.

1.1.4.2 Hạn chế:

Hạn chế di động: Mặc dù .NET Core đã mở rộng hỗ trợ của mình cho các nền tảng di động như iOS và Android, việc phát triển ứng dụng di động vẫn chưa phải là lựa chọn phổ biến nhất so với các nền tảng như Swift cho iOS và Kotlin cho Android.

Hiệu suất: Trong một số trường hợp, C# và .NET có thể không bằng các ngôn ngữ và nền tảng khác như C++ hoặc Rust về hiệu suất tuyệt vời trong các ứng dụng đòi hỏi tài nguyên cao và thời gian thực.

Hạn chế đa nền tảng: Mặc dù .NET Core và .NET 5 trở đi đã mở rộng sự hỗ trợ của mình cho các nền tảng không chỉ là Windows, nhưng vẫn còn một số hạn chế trong việc phát triển ứng dụng đa nền tảng.

Tính mở rộng: Một số nhà phát triển cho rằng việc mở rộng các ứng dụng .NET có thể không linh hoạt như các nền tảng khác như Node.js hoặc Python.

Hạn chế trong thư viện và công cụ bên thứ ba: Mặc dù .NET có một cộng đồng phát triển mạnh mẽ, nhưng vẫn có thể gặp phải hạn chế trong số lượng và chất lượng của các thư viện và công cụ bên thứ ba so với các hệ sinh thái khác như Python hoặc JavaScript.

1.2 Tổng quan nghiên cứu và xây dựng hệ thống quản lý sinh viên

Đề tài nghiên cứu và xây dựng mô hình quản lý sinh viên là một chủ đề rộng lớn và hứa hẹn mang lại nhiều kiến thức và kỹ năng trong lĩnh vực phát triển phần mềm và quản lý cơ sở dữ liệu. Dưới đây là một tổng quan về đề tài này:

Lý do chọn đề tài:

Nhu cầu thị về hệ thống quản lý sinh viên ở các trường đại học, cao đẳng

đang ngày càng tăng.

Mục tiêu nghiên cứu:

Phân tích yêu cầu của hệ thống quản lý sinh viên

Thiết kế cơ sở dữ liệu cho hệ thống quản lý sinh viên

Xây dựng hệ thống sử dụng ngôn ngữ C# và công nghê .NET.

Công nghệ và công cụ sử dụng:

Ngôn Ngữ Lập Trình: C# và công nghệ .NET.

Cơ Sở Dữ Liệu: MySQL

Chức năng chính của hệ thống:

Quản lý sinh viên.

Quản lý khóa học.

Quản lý điểm.

Kiểm thử và debugging:

Triển khai các kỹ thuật kiểm thử để đảm bảo tính ổn định và độ tin cậy của hệ thống.

Sử dụng công cụ debug để tìm và sửa lỗi.

Hạn chế của đề tài:

Giới hạn về thời gian và tài nguyên.

Các yếu tố khách quan như sự thay đổi trong yêu cầu hệ thống.

Chương 2: Đặc tả yêu cầu

2.1 Xác định yêu cầu chức năng

2.1.1 Yêu cầu chức năng

Yêu cầu chức năng của một mô hình quản lý sinh viên là một phần quan trọng trong quá trình nghiên cứu và phát triển. Hệ thống quản lý sinh viên bao gồm các chức năng sau:

Quản lý sinh viên

Quản lý khóa học

Quản lý điểm

Quản lý điểm rèn luyện

2.1.2 Yêu cầu phi chức năng

Một số yêu cầu phi chức năng quan trong trong mô hình quản lý sinh viên:

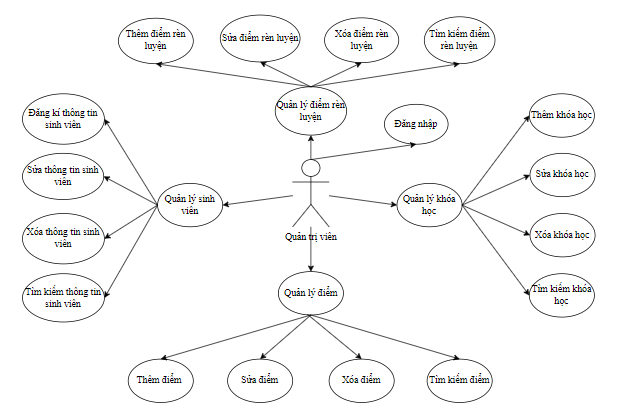
Hiệu suất

Bảo mật

Quản lý dữ liệu

Tiêu chuẩn và tương thích

2.2. Mô hình đặc tả sơ đồ use case đề tài nghiên cứu và xây dựng mô hình quản lý sinh viên



Hình 2.2.1 : Sơ đồ use case tổng quan quản lý sinh viên

2.3 Đặc tả use case nghiên cứu và xây dựng mô hình quản lý sinh viên

2.3.1 Use case đăng nhập

Mô tả: Quản trị viên có thể đăng nhập vào để điều khiển hệ thống

Hoạt động:

Chọn đăng nhập

Nhập user và password

Nhấn đăng nhập để vào hệ thống

2.3.2 Use case quản lý sinh viên

2.3.2.1 Đăng kí thông tin sinh viên

Mô tả: Quản trị viên có thể thêm thông tin sinh viên mới vào danh sách.

Hoạt Động:

Chọn chức năng thêm thông tin sinh viên.

Nhập thông tin sinh viên (Mã sinh viên, họ và tên, lớp, v.v.).

Lưu và cập nhật danh sách sinh viên.

2.3.2.2 Chỉnh sửa thông tin sinh viên

Mô tả: Quản trị viên có thể chỉnh sửa thông tin của một sinh viên.

Hoạt Động:

Chọn chức năng chỉnh sửa thông tin sinh viên.

Chọn sinh viên cần chỉnh sửa thông tin.

Cập nhật thông tin và lưu lại.

2.3.2.3 Xóa thông tin sinh viên

Mô tả: Quản trị viên có thể xóa sinh viên và thông tin của sinh viên đó khỏi danh sách.

Hoạt Động:

Chọn chức năng xóa thông tin sinh viên.

Chọn sinh viên cần xóa.

Xác nhận và xóa sinh viên.

2.3.2.4 Tìm kiếm thông tin sinh viên

Mô tả: Quản trị viên có thể tìm thông tin sinh viên theo mã sinh viên của sinh viên

Hoạt động:

Chọn chức năng tìm kiếm thông tin sinh viên.

Nhập mã sinh viên cần tìm.

Hiển thị thông tin sinh viên cần tìm theo mã.

2.3.3 Use case quản lý khóa học

2.3.3.1. Thêm thông tin khóa học mới

Mô tả: Quản trị viên có thể thêm thông tin khóa học mới vào hệ thống.

Hoạt Động:

Chọn chức năng thêm thông tin khóa học.

Nhập thông tin cho khóa học mới.

Lưu và cập nhật danh sách khóa học.

2.3.3.2. Chỉnh sửa thông tin khóa học

Mô tả: Quản trị viên có thể chỉnh sửa thông tin của một khóa học.

Hoạt Động:

Chọn chức năng chỉnh sửa thông tin khóa học.

Chọn khóa học cần chỉnh sửa.

Cập nhật thông tin và lưu lại.

2.3.3.3. Xóa khóa học

Mô tả: Quản trị viên có thể xóa khóa học khỏi danh sách.

Hoạt Động:

Chọn chức năng xóa khóa học.

Chọn khóa học cần xóa.

Xác nhận và xóa khóa học.

2.4.3 Use case quản lý điểm

2.4.3.1. Thêm điểm

Mô tả: Quản trị viên có thể thêm điểm cho sinh viên trên hệ thống.

Hoạt Động:

Chọn chức năng quản lý điểm.

Chọn thêm điểm.

Chọn sinh viên cần thêm điểm.

Thêm điểm cho sinh viên

Lưu và cập nhật điểm.

2.4.3.2. Sửa điểm

Mô tả: Quản trị viên có thể sửa điểm cho sinh viên trên hệ thống.

Hoạt Động:

Chọn chức năng quản lý điểm.

Chọn sửa điểm.

Chọn sinh viên cần sửa điểm.

Sửa điểm cho sinh viên

Lưu và cập nhật điểm.

2.4.3.3. Xóa điểm

Mô tả: Quản trị viên có thể xóa điểm của sinh viên trên hệ thống.

Hoạt Động:

Chọn chức năng quản lý điểm.

Chọn xóa điểm.

Chọn sinh viên cần xóa điểm.

Xóa điểm của sinh viên

Lưu và cập nhật điểm.

2.4.3.4 Tìm kiếm điểm

Mô tả: Quản trị viên có thể tìm kiếm điểm của sinh viên trên hệ thống.

Hoạt Động:

Chọn chức năng quản lý điểm.

Chọn tìm kiếm điểm.

Chọn sinh viên cần tìm kiếm điểm.

Tìm kiếm điểm của sinh viên.

Lưu và cập nhật điểm

2.4.4 Use case quản lý điểm rèn luyện

2.4.4.1. Thêm điểm rèn luyện

Mô tả: Quản trị viên có thể thêm điểm rèn luyện cho sinh viên trên hệ thống.

Hoạt Động:

Chọn chức năng điểm rèn luyện.

Chọn thêm điểm rèn luyện.

Chọn sinh viên cần thêm điểm rèn luyện.

Thêm điểm rèn luyện cho sinh viên

Lưu và cập nhật điểm rèn luyện.

2.4.4.2. Sửa điểm rèn luyện

Mô tả: Quản trị viên có thể sửa điểm rèn luyện cho sinh viên trên hệ thống.

Hoạt Động:

Chọn chức năng điểm rèn luyện.

Chọn sửa điểm rèn luyện.

Chọn sinh viên cần sửa điểm rèn luyện.

Sửa điểm rèn luyện cho sinh viên

Lưu và cập nhật điểm rèn luyện.

2.4.4.3. Xóa điểm rèn luyện

Mô tả: Quản trị viên có thể xóa điểm rèn luyện của sinh viên trên hệ thống.

Hoạt Động:

Chọn chức năng điểm rèn luyện.

Chọn xóa điểm rèn luyện.

Chọn sinh viên cần xóa điểm rèn luyện.

Xóa điểm rèn luyện của sinh viên

Lưu và cập nhật điểm rèn luyện.

2.4.4.4 Tìm kiếm điểm rèn luyện

Mô tả: Quản trị viên có thể tìm kiếm điểm rèn luyện của sinh viên trên hệ thống.

Hoạt Động:

Chọn chức năng điểm rèn luyện.

Chọn tìm kiếm điểm rèn luyện.

Chọn sinh viên cần tìm kiếm điểm rèn luyện.

Tìm kiếm điểm rèn luyện của sinh viên.

Lưu và cập nhật điểm rèn luyện.

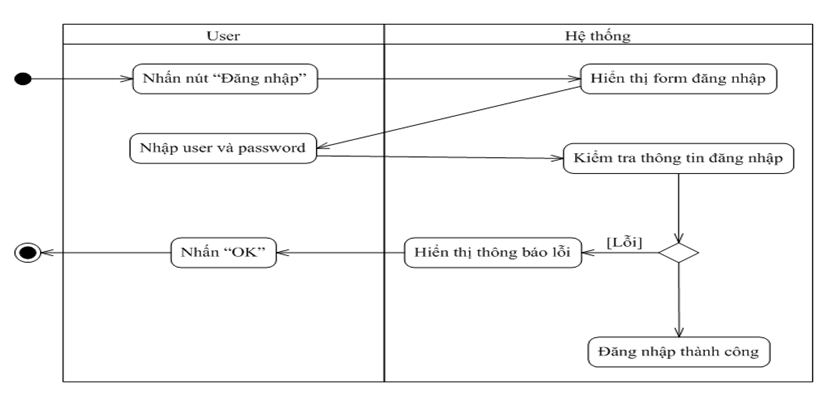
Chương 3: Thiết kế

3.1 Biểu đồ hoạt động

Biểu đồ hoạt động là phương phương tiện để mô tả các dòng công việc trong hệ thống. Ở đây biểu đồ hoạt động dùng để mô tả chi tiết hoạt động bên trong một thao tác.

Activity diagram không cho ta thấy được các đối tượng ứng xử và cộng tác với đối tượng khác như thế nào.

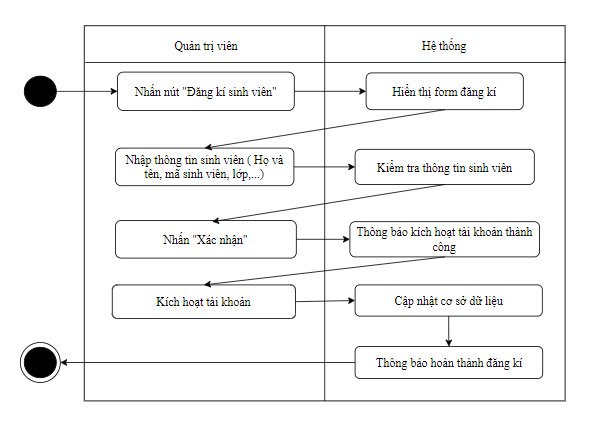
3.1.1 Đăng nhập



Hình 3.1.1 : Biểu đồ đăng nhập của quản trị viên

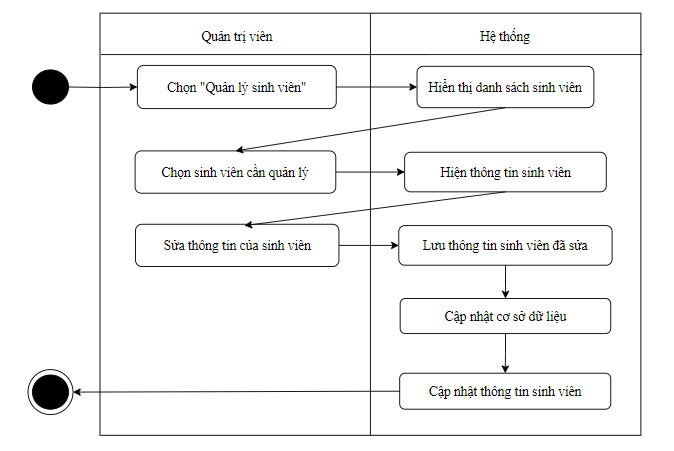
3.1.2 Quản lý sinh viên

3.1.2.1 Đăng kí thông tin sinh viên



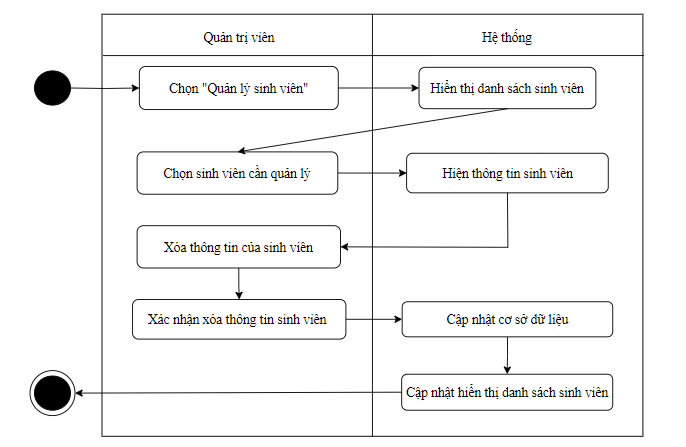
Hình 3.1.2.1 : Biểu đồ hoạt động đăng kí thông tin sinh viên

3.1.2.2 Sửa thông tin sinh viên



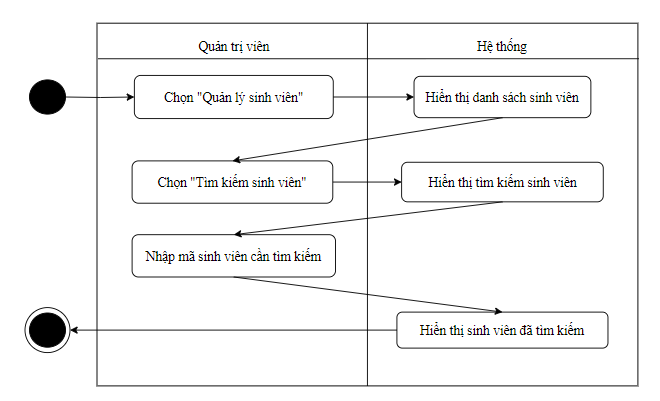
Hình 3.1.2.2 : Biểu đồ hoạt động chỉnh sửa thông tin sinh viên

3.1.2.3 Xóa thông tin sinh viên



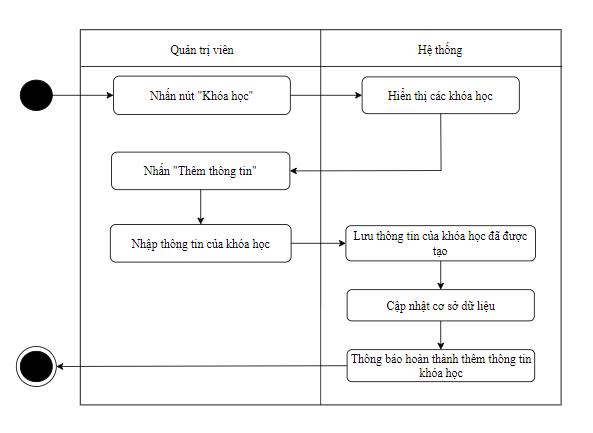
Hình 3.1.2.3 : Biểu đồ hoạt động xóa thông tin sinh viên

3.1.2.4 Tìm kiếm thông tin sinh viên

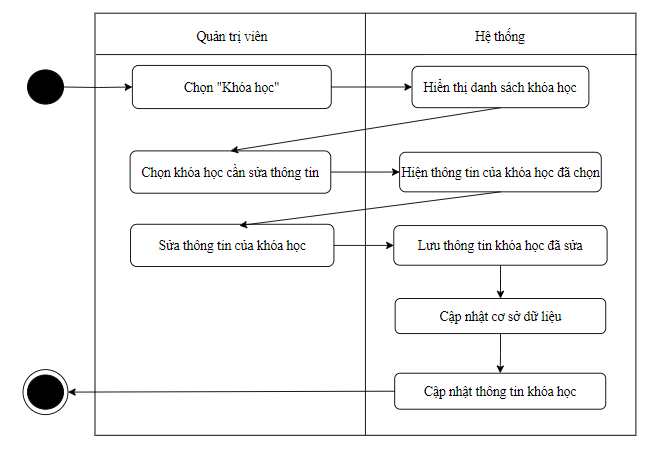


Hình 3.1.2.4 : Biểu đồ hoạt động tìm kiếm thông tin sinh viên

3.1.3 Quản lý khóa học

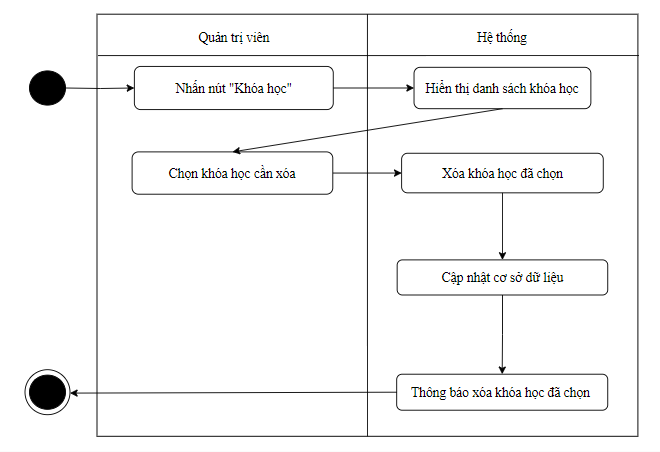
4.1.3.1 Thêm thông tin khóa học

Hình 3.1.3.1 : Biểu đồ hoạt động thêm thông tin khóa học

3.1.3.2 Sửa thông tin khóa học

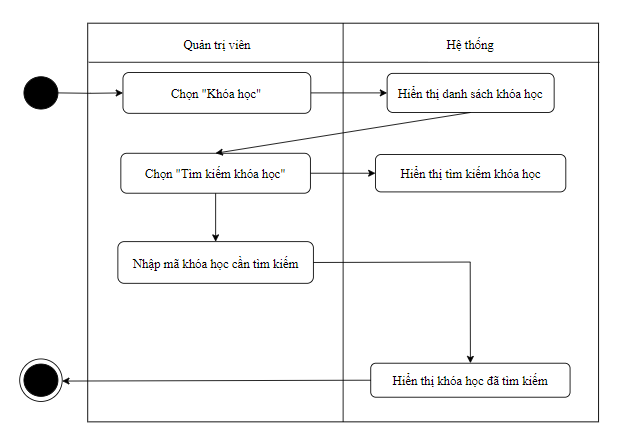
Hình 3.1.3.2 : Biểu đồ hoạt động sửa thông tin khóa học

3.1.3.3 Xóa thông tin khóa học



Hình 3.1.3.3 : Biểu đồ hoạt động xóa thông tin khóa học

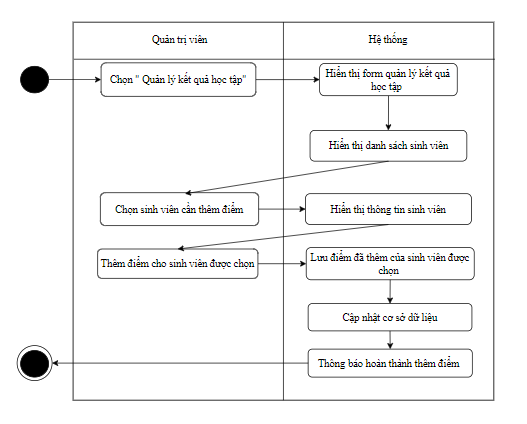
3.1.3.4 Tìm kiếm thông tin khóa học



Hình 3.1.3.4 : Biểu đồ hoạt động tìm kiếm thông tin khóa học

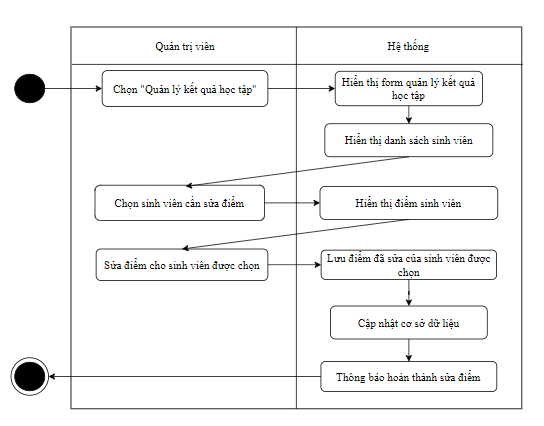
3.1.4 Quản lý điểm

3.1.4.1 Thêm điểm



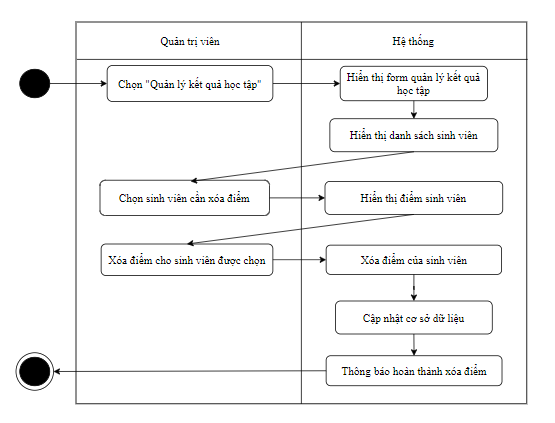
Hình 3.1.4.1 : Biểu đồ hoạt động thêm điểm.

3.1.4.2 Sửa điểm



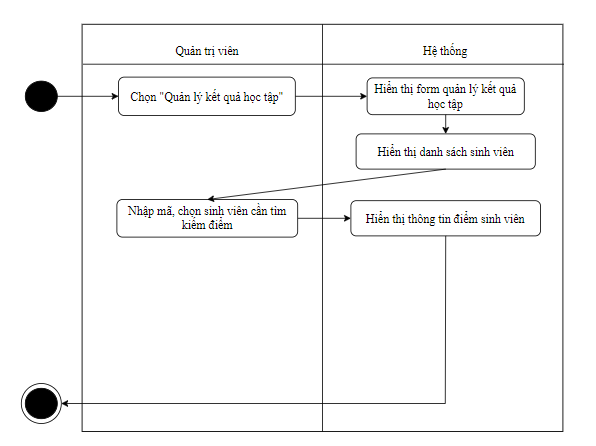
Hình 3.1.4.2 : Biểu đồ hoạt động sửa điểm

3.1.4.3 Xóa điểm



Hình 3.1.4.3 : Biểu đồ hoạt động xóa điểm

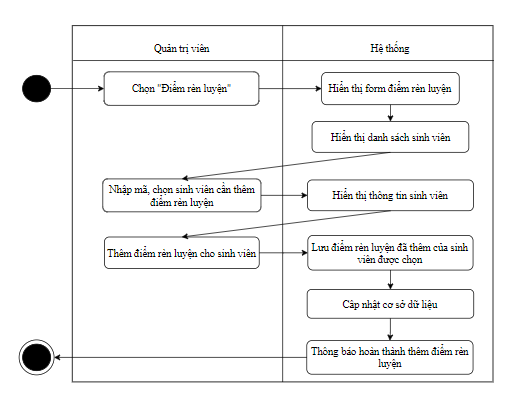
3.1.4.4 Tìm kiếm điểm



Hình 3.1.4.4 : Biểu đồ hoạt động tìm kiếm điểm

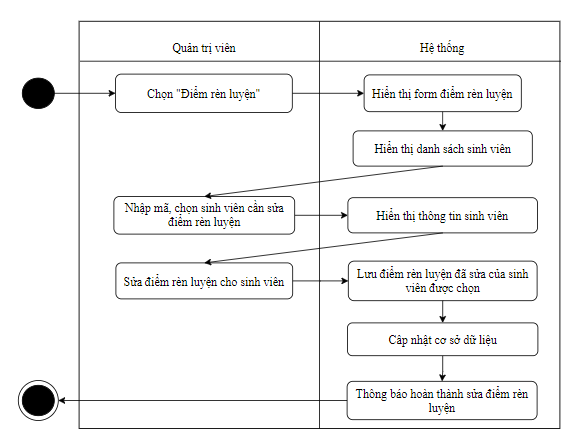
### 3.1.5 Quản lý điểm rèn luyện

3.1.5.1 Thêm điểm rèn luyện



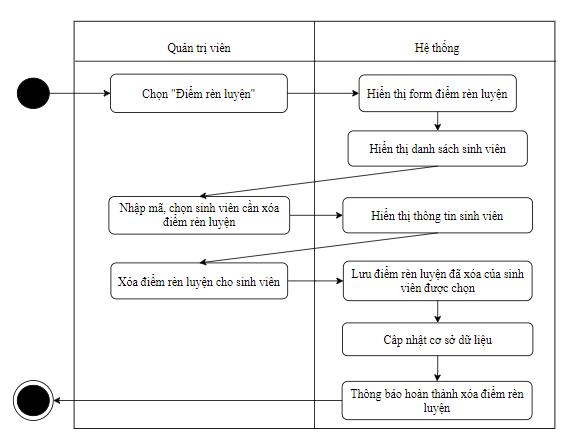
Hình 3.1.5.1 : Biểu đồ hoạt động thêm điểm rèn luyện

3.1.5.2 Sửa điểm rèn luyện



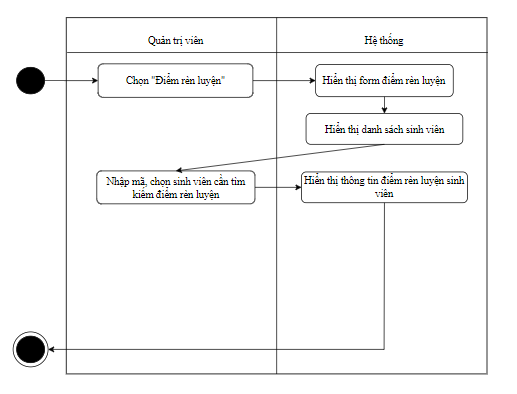
Hình 3.1.5.2 : Biểu đồ hoạt động sửa điểm rèn luyện

3.1.5.3 Xóa điểm rèn luyện



Hình 3.1.5.3 : Biểu đồ hoạt động xóa điểm rèn luyện

3.1.5.4 Tìm kiếm điểm rèn luyện



Hình 3.1.5.4 : Biểu đồ hoạt động tìm kiếm điểm rèn luyện

3.2 Biểu đồ tuần tự

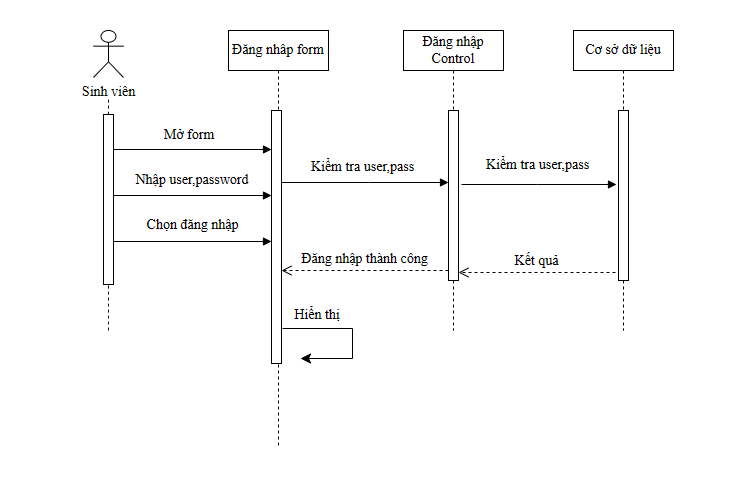
Biểu đồ tuần tự thể hiện một chuỗi các sự kiện, hành vi của đối tượng theo một trình tự thời gian. Nó được sử dụng để mô tả dòng thông điệp được gửi đi và và các đối tượng phối hợp nhận và xử lý để trả về kết quả mà theo yêu cầu mà thông điệp gửi đến.

Biểu đồ tuần tự thông thường được sử dụng như một mô hình giải thích cho kịch bản use case.

Biểu đồ tuần tự thể hiện rất rõ đối tượng nào tương tác với đối tượng nào và thông điệp là gì.

Khi đọc một biểu đồ tuần tự ta đọc từ trái qua phải và từ trên xuống dưới.

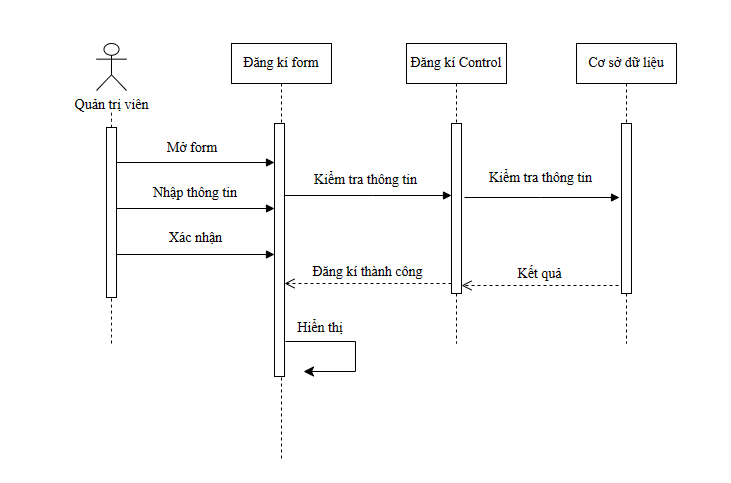
3.2.1 Đăng nhập



Hình 3.2.1: Biểu đồ tuần tự đăng nhập của quản trị viên

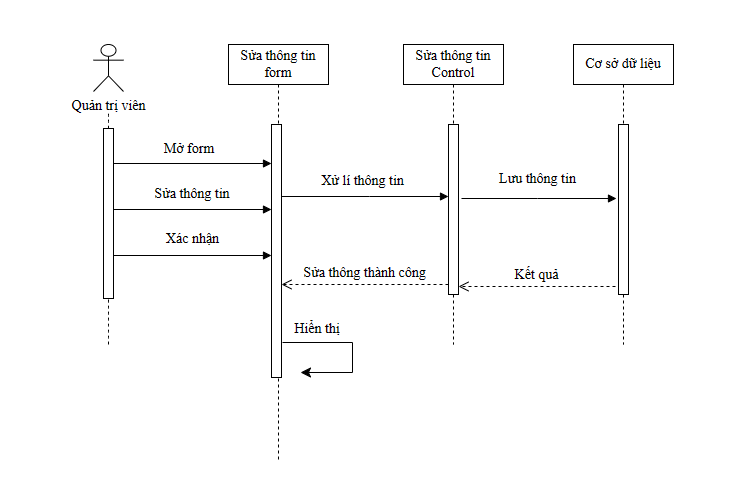
3.2.2 Quản lý sinh viên

3.2.2.1 Đăng kí thông tin sinh viên



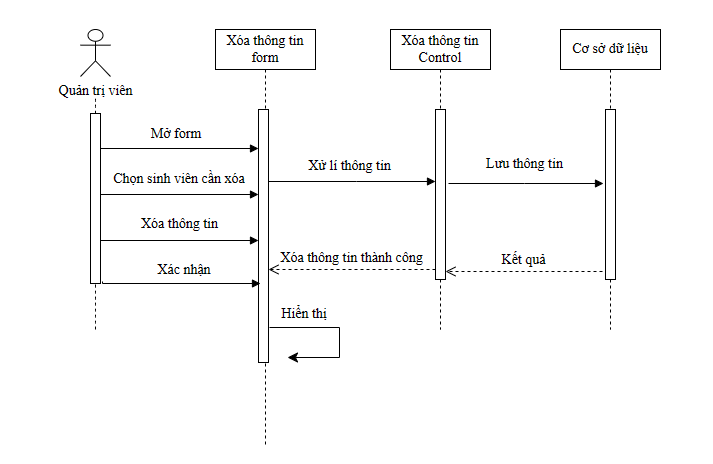
Hình 3.2.2.1 : Biểu đồ tuần tự đăng kí thông tin sinh viên

3.2.2.2 Sửa thông tin sinh viên



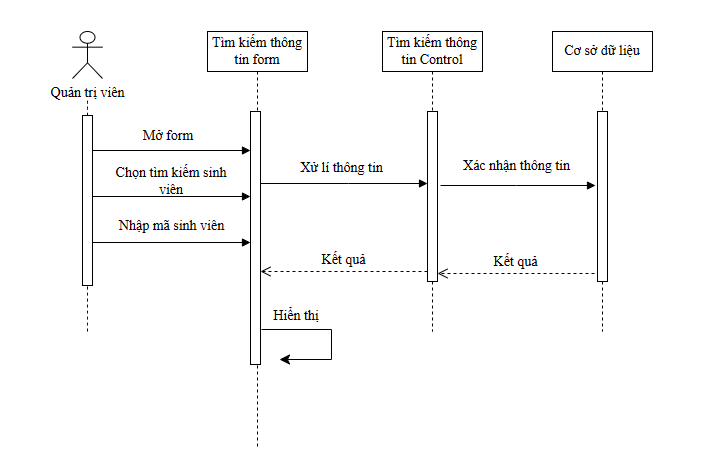
Hình 3.2.2.2 : Biểu đồ tuần tự chỉnh sửa thông tin sinh viên

3.2.2.3 Xóa thông tin sinh viên



Hình 3.2.2.3 : Biểu đồ tuần tự xóa thông tin sinh viên

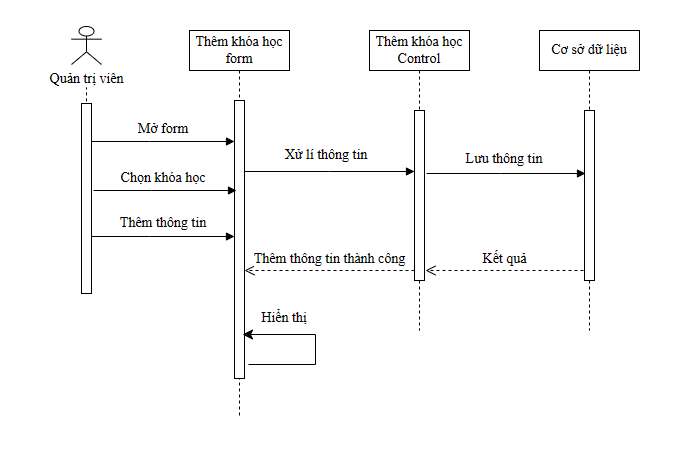
3.2.2.4 Tìm kiếm thông tin sinh viên



Hình 3.2.2.4 : Biểu đồ tuần tự tìm kiếm thông tin sinh viên

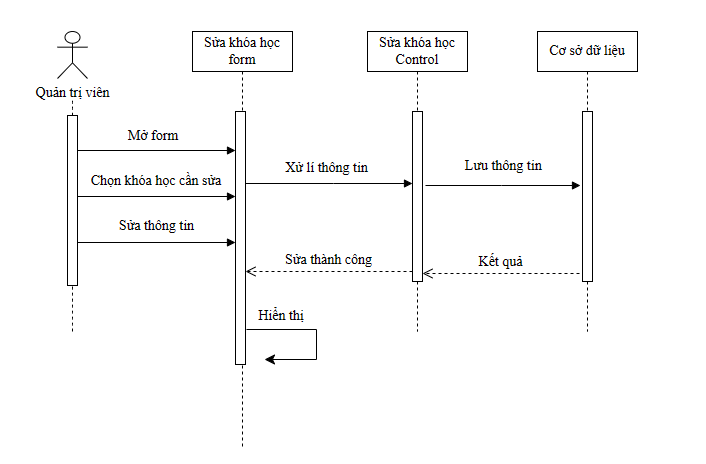
3.2.3 Quản lý khóa học

3.2.3.1 Thêm thông tin khóa học



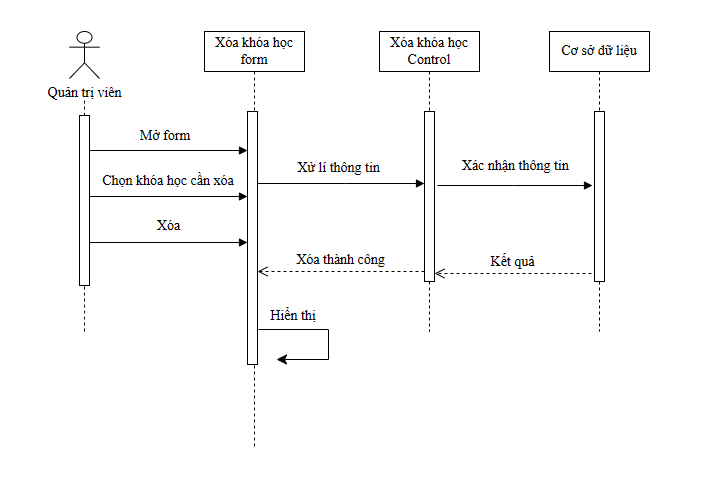
Hình 3.2.3.1 : Biểu đồ tuần tự thêm thông tin khóa học

3.2.3.2 Sửa thông tin khóa học



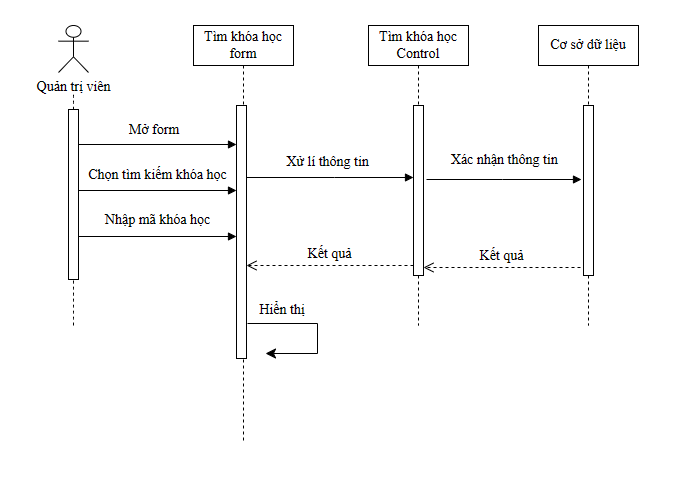
Hình 3.2.3.2 : Biểu đồ tuần tự chỉnh sửa thông tin khóa học

3.2.3.3 Xóa thông tin khóa học



Hình 3.2.3.3 : Biểu đồ tuần tự xóa thông tin khóa học

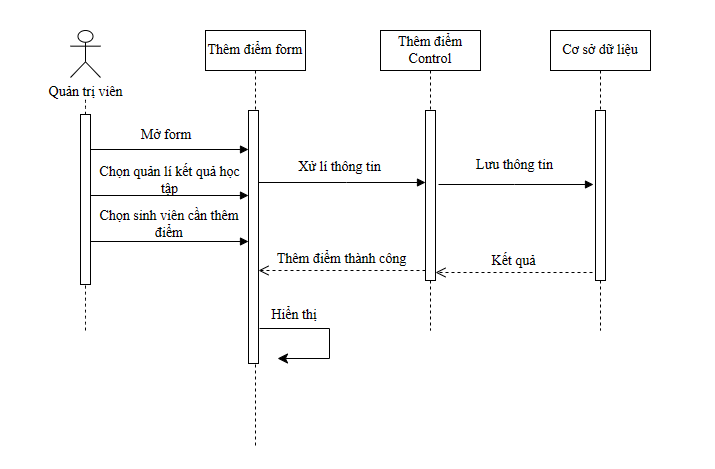
3.2.3.4 Tìm kiếm thông tin khóa học



Hình 3.2.3.4 : Biểu đồ tuần tự tìm kiếm thông tin khóa học

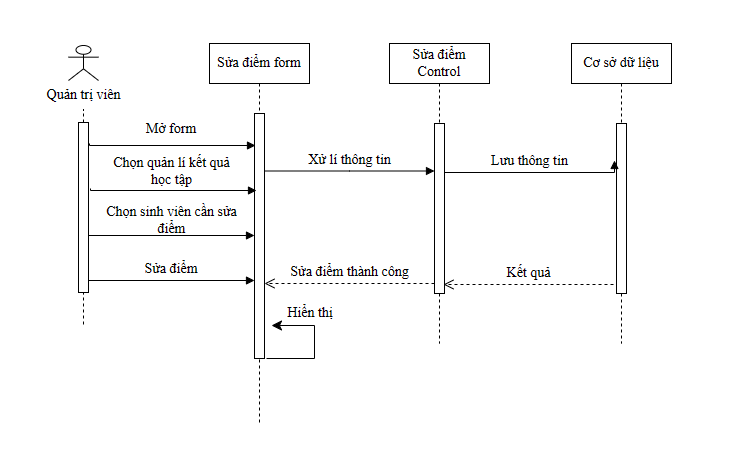
3.2.4 Quản lý điểm

3.2.4.1 Thêm điểm



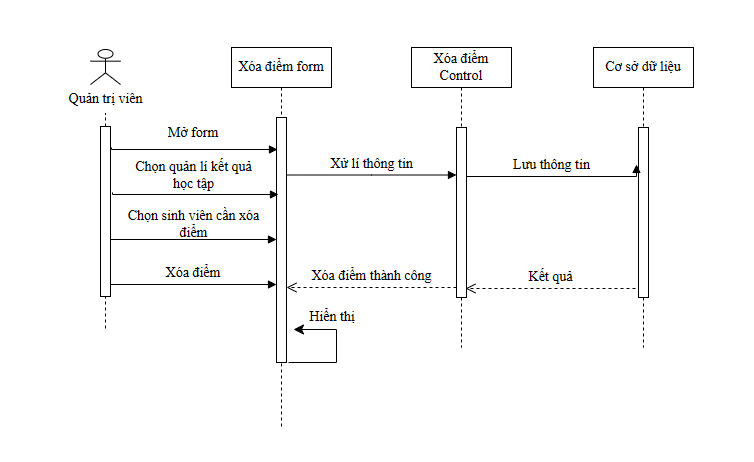
Hình 3.2.4.1 : Biểu đồ tuần tự thêm điểm

3.2.4.2 Sửa điểm



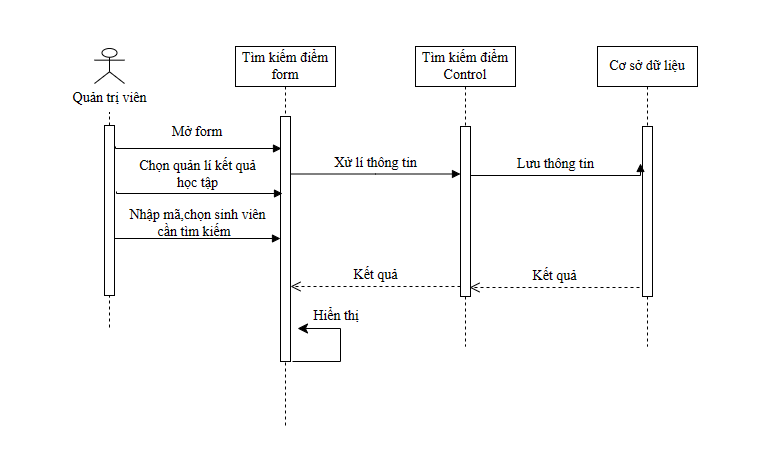
Hình 3.2.4.2 : Biểu đồ tuần tự sửa điểm

3.2.4.3 Xóa điểm



Hình 3.2.4.3 : Biểu đồ tuần tự xóa điểm

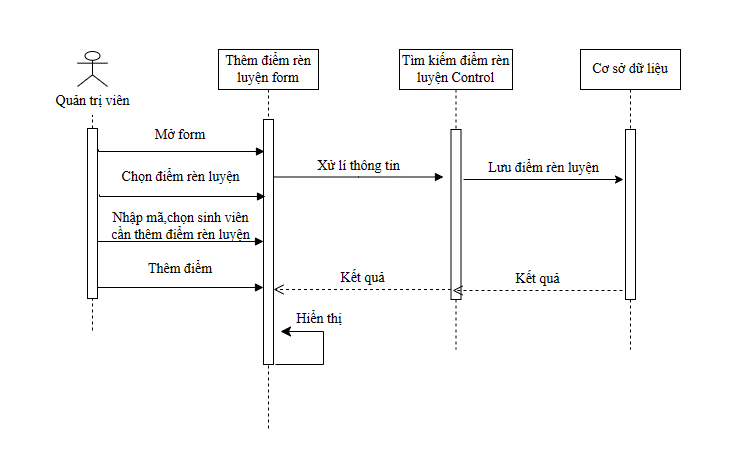
3.2.4.4 Tìm kiếm điểm



Hình 3.2.4.4 :Biểu đồ tuần tự tìm kiếm điểm

### 3.2.5 Quản lý điểm rèn luyện

3.2.5.1 Thêm điểm rèn luyện



Hình 3.2.5.1 Biểu đồ tuần tự thêm điểm rèn luyện

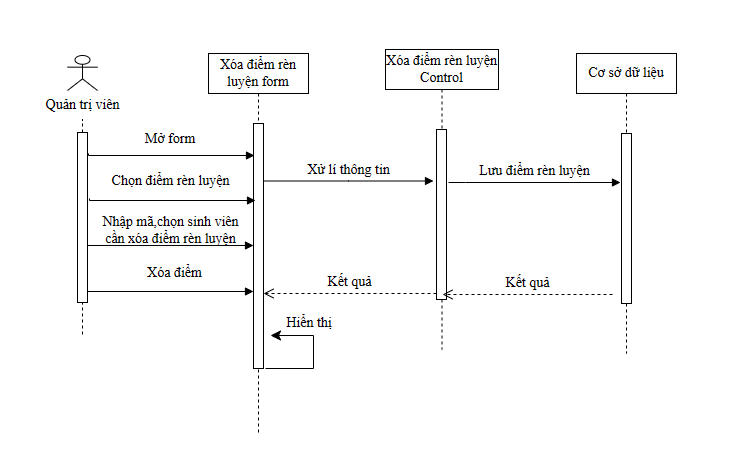
3.2.5.2 Sửa điểm rèn luyện

A diagram of a diagram

Description automatically generated

Hình 3.2.5.2 Biểu đồ tuần tự sửa điểm rèn luyện

3.2.5.3 Xóa điểm rèn luyện



Hình 3.2.5.3 Biểu đồ tuần tự xóa điểm rèn luyện

3.2.5.4 Tìm kiếm điểm rèn luyện

A diagram of a diagram

Description automatically generated

Hình 3.2.5.4 Biểu đồ tuần tự tìm kiếm điểm rèn luyện

Chương 4: Thiết kế cơ sở dữ liệu

4.1 Mô tả thực thể kết hợp

Mô hình thực thể kết hợp được sử dụng để biểu diễn cơ sở dữ liệu ở mức khái niệm. Mô hình thực thể kết hợp bao gồm có các thực thể, danh sách thuộc tính và những mối kết hợp. Các loại thực thể có các thuộc tính như sau:

Bảng dữ liệu sinh viên: (student)

StdId (Khóa chính): Mã số duy nhất của mỗi sinh viên, kiểu dữ liệu là int (5).

StdFirstName: Họ của sinh viên, kiểu dữ liệu là varchar (15).

StdLastName: Tên sinh viên, kiểu dữ liệu là varchar (15).

Birthdate: Ngày sinh của sinh viên, kiểu dữ liệu là date

Gender: Giới tính của sinh viên, kiểu dữ liệu là varchar (10).

Phone: Số điện thoại của sinh viên, kiểu dữ liệu là varchar (15).

Address: Địa chỉ của sinh viên, kiểu dữ liệu là text.

Photo: Ảnh đại diện của sinh viên, kiểu dữ liệu là longblob.

Bảng số liệu khóa học: (course)

CourseId (Khóa chính): Mã số duy nhất của mỗi khóa học, kiểu dữ liệu là int (10).

CourseName: Tên khóa học, kiểu dữ liệu là varchar(50) .

CourseHour: Thời gian học của khóa học, kiểu dữ liệu là int(5).

Description: Mô tả khóa học, kiểu dữ liệu là text.

Bảng dữ liệu điểm: (score)

StudentId: Mã số duy nhất của mỗi sinh viên, kiểu dữ liệu là int(5).

CourseName: Tên khóa học, kiểu dữ liệu là varchar(50).

Score: Điểm số khóa học, kiểu dữ liệu là double.

Description: Mô tả thành tích điểm khóa học, kiểu dữ liệu là text.

Bảng dữ liệu điểm rèn luyện :

StudentId: Mã số duy nhất của mỗi sinh viên, kiểu dữ liệu là int(5).

Pointtraining: Điểm rèn luyện của sinh viên, kiểu dữ liệu là double.

Description: Mô tả thông tin điểm rèn luyện, kiểu dữ liệu là text.

## 4.2 Mô tả các bảng

### 4.2.1 Bảng dữ liệu sinh viên: (student)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Cho phép Null | Khóa chính | Ghi chú |
| StdId | INT(5) | Không | P | Mã số duy nhất của mỗi sinh viên |
| StdFirstName | VARCHAR(15) | Không |  | Họ của sinh viên |
| StdLastName | VARCHAR(15) | Không |  | Tên sinh viên |
| Birthdate | DATE | Không |  | Ngày sinh của sinh viên |
| Gender | VARCHAR(10) | Không |  | Giới tính của sinh viên |
| Phone | VARCHAR(15) | Không |  | Số điện thoại của sinh viên |
| Address | TEXT |  |  | Địa chỉ của sinh viên |
| Photo | LONGBLOB |  |  | Ảnh đại diện của sinh viên |

Bảng 4.2.1 : Bảng dữ liệu sinh viên

### 4.2.2 Bảng số liệu khóa học: (course)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Cho phép Null | Khóa chính | Ghi chú |
| CourseId | INT(10) | Không | P | Id khóa học |
| CourseName | VARCHAR(50) | Không |  | Tên khóa học |
| CourseHour | INT(5) | Không |  | Thời gian học một khóa học |
| Description | TEXT | Không |  | Mô tả thông tin khóa học |

Bảng 4.2.2 : Bảng dữ liệu khóa học

### 4.2.3 Bảng dữ liệu điểm: (score)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Cho phép Null | Khóa chính | Ghi chú |
| StudentId | INT(5) | Không |  | Mã số duy nhất của mỗi sinh viên |
| CourseName | VARCHAR(50) | Không |  | Tên khóa học |
| Score | DOUBLE | Không |  | Điểm số khóa học |
| Description | TEXT | Không |  | Mô tả thành tích điểm khóa học |

Bảng 4.2.3 : Bảng dữ liệu điểm

### 4.2.4 Bảng dữ liệu đăng nhập: (user)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Cho phép Null | Khóa chính | Ghi chú |
| useid | INT(5) | Không | P | Id sinh viên |
| usename | VARCHAR(20) | Không |  | Tên đăng nhập tài khoản của sinh viên |
| password | VARCHAR(20) | Không |  | Mật khẩu đăng nhập tài khoản của sinh viên |

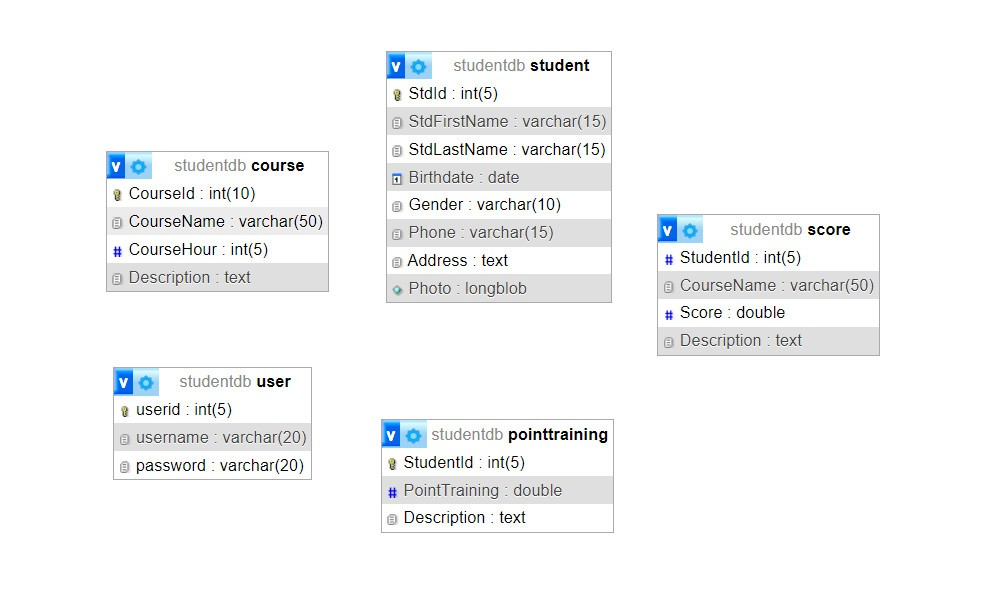
Bảng 4.2.4 : Bảng dữ liệu đăng nhập

4.2.5 Bảng dữ liệu điểm rèn luyện: (pointtraining)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Cho phép Null | Khóa chính | Ghi chú |
| StudentId | INT(5) | Không |  | Id sinh viên |
| Pointtraining | DOUBLE | Không |  | Điểm rèn luyện của sinh viên |
| Description | TEXT | Không |  | Mô tả thành tích điểm rèn luyện |

Bảng 4.2.5 : Bảng dữ liệu điểm rèn luyện

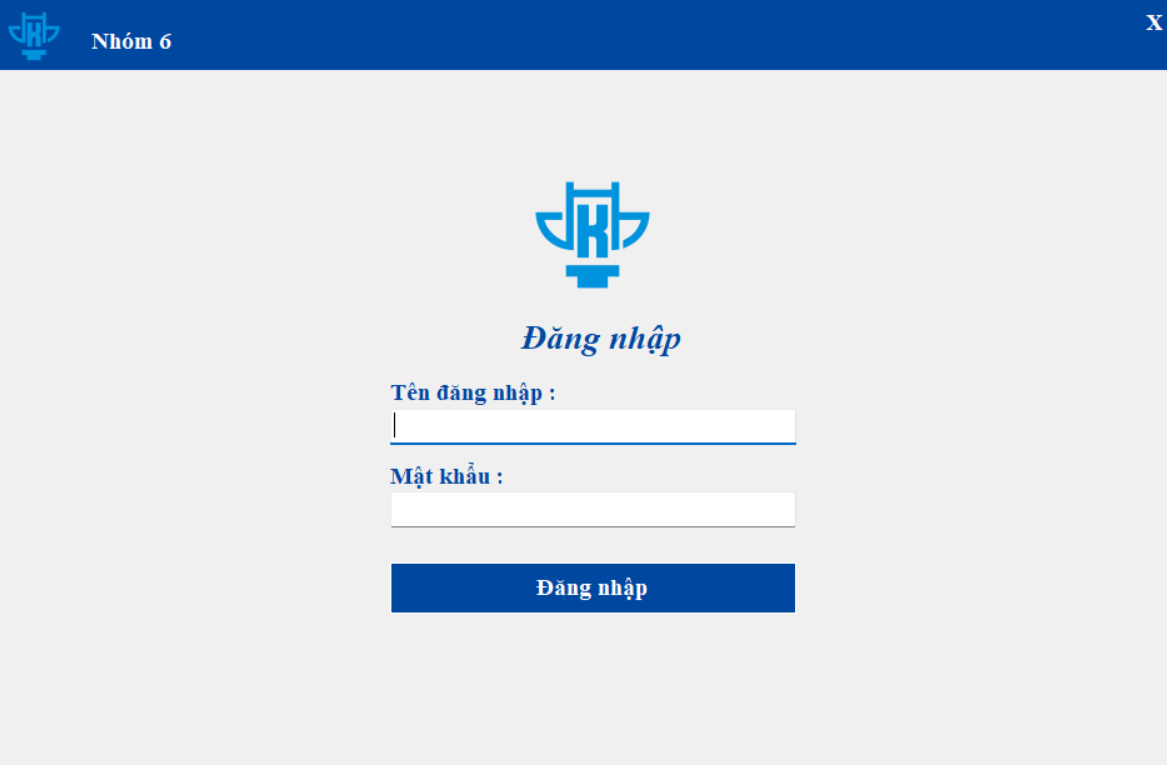
4.3 Kết nối giữa các bảng CSDL



Hình 4.3.1 : Bảng CSDL của mô hình quản lý sinh viên

Chương 5: Hình ảnh demo chương trình

## 5.1 Đăng nhập



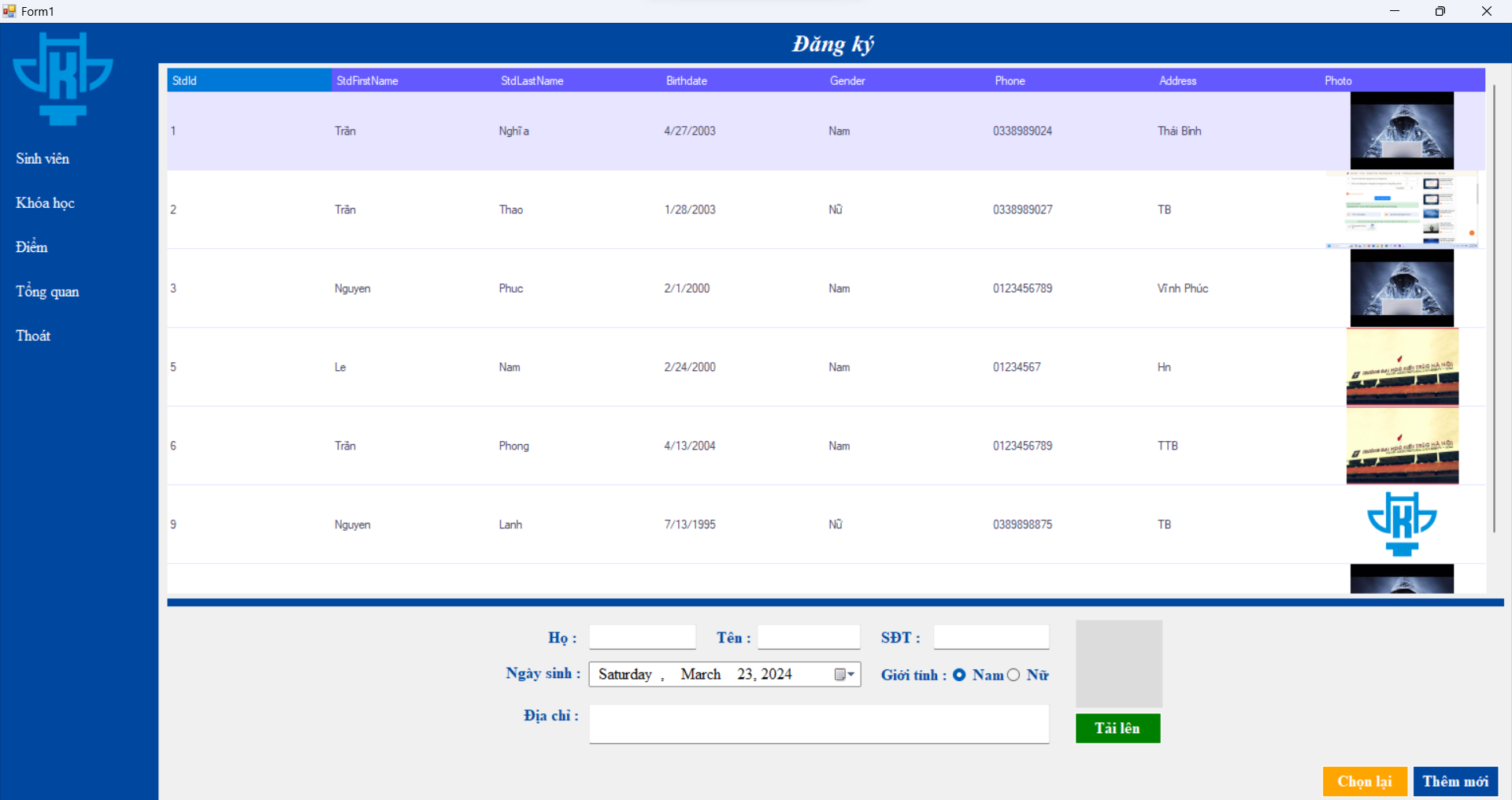
Hình 5.1 : Hình ảnh đăng nhập

## 5.2 Trang chủ



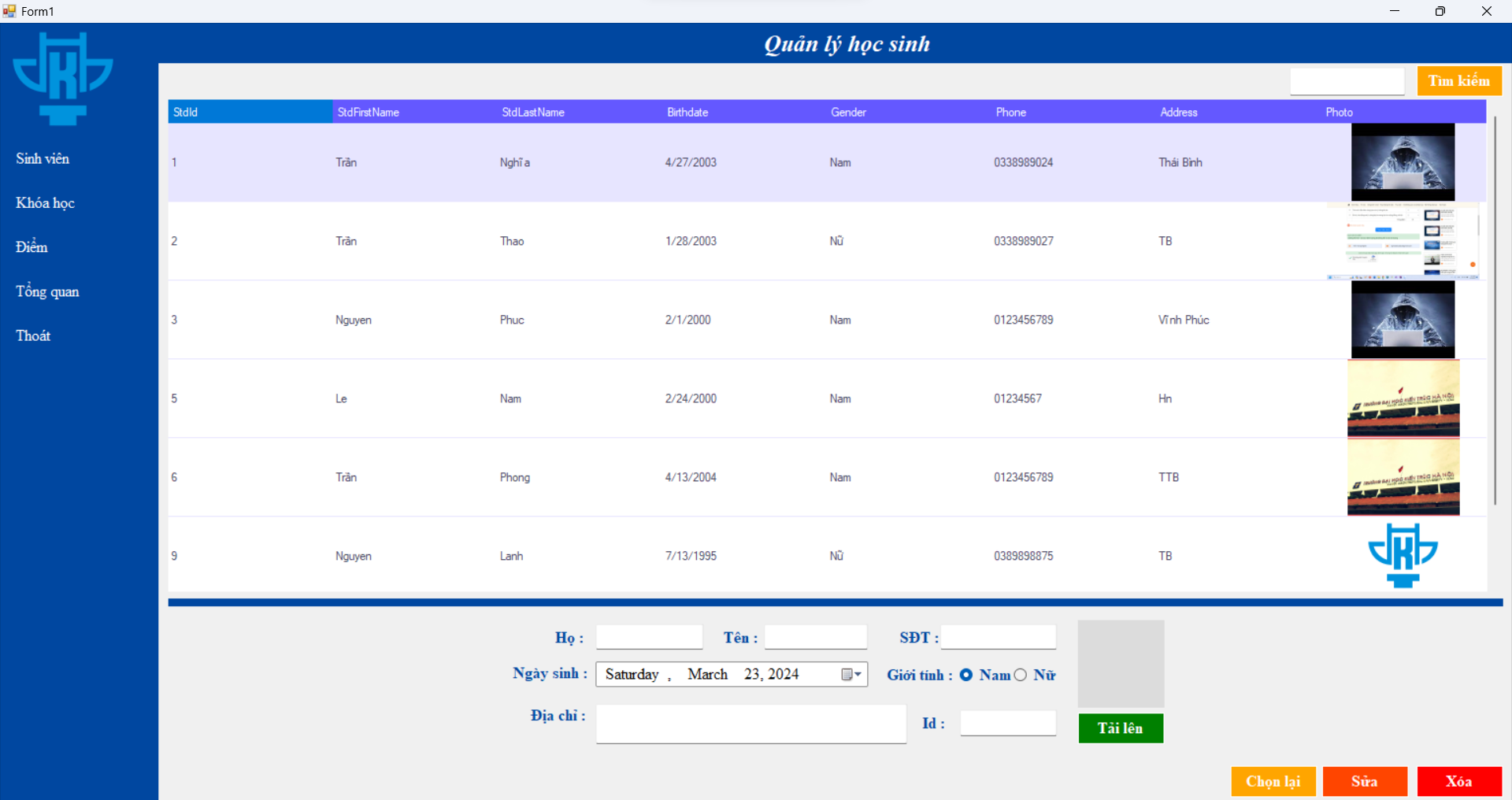
Hình 5.2 : Hình ảnh trang chủ

## 5.3 Thêm sinh viên



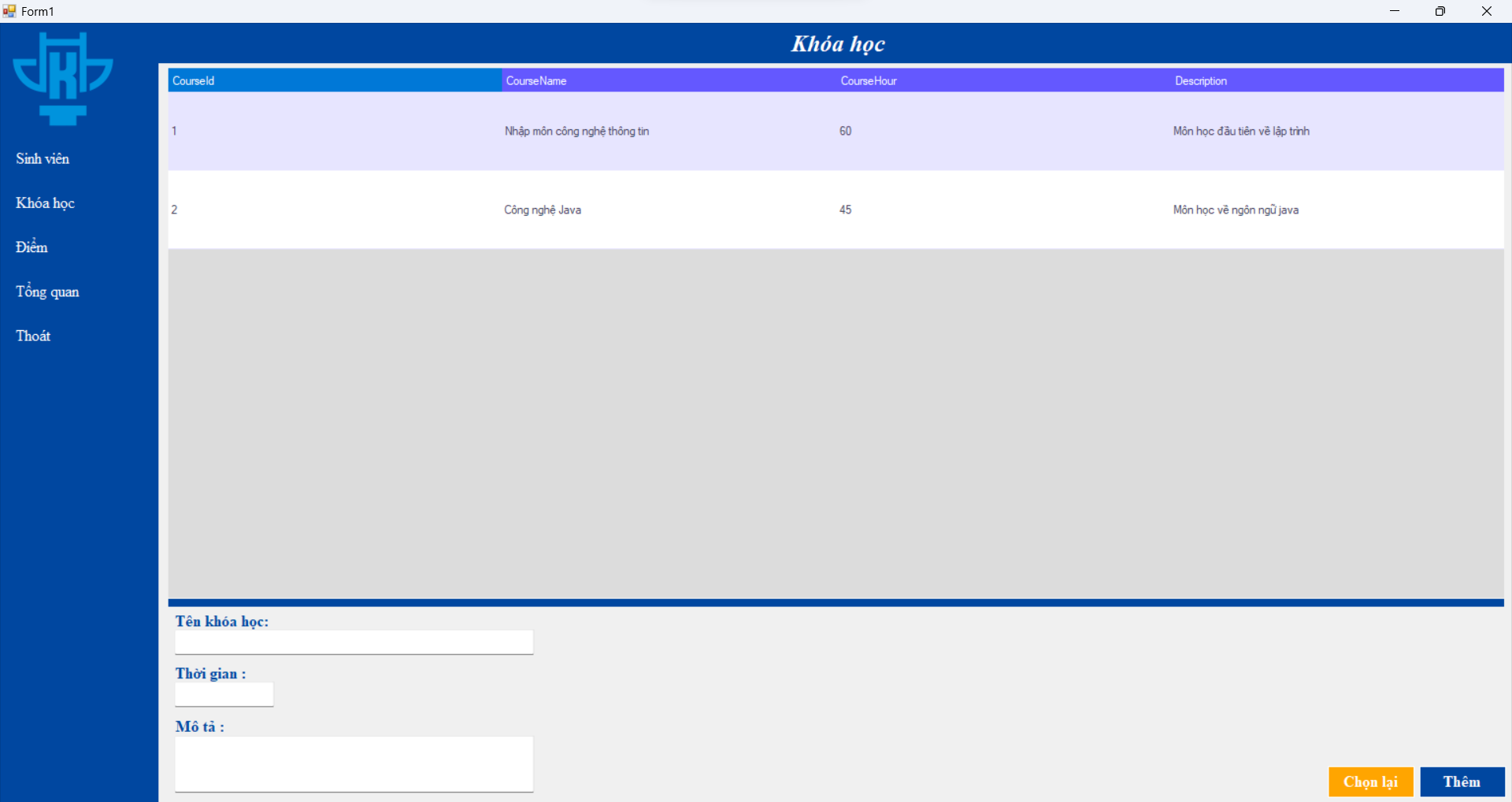
Hình 5.3 : Hình ảnh thêm sinh viên

5.4 Quản lý sinh viên



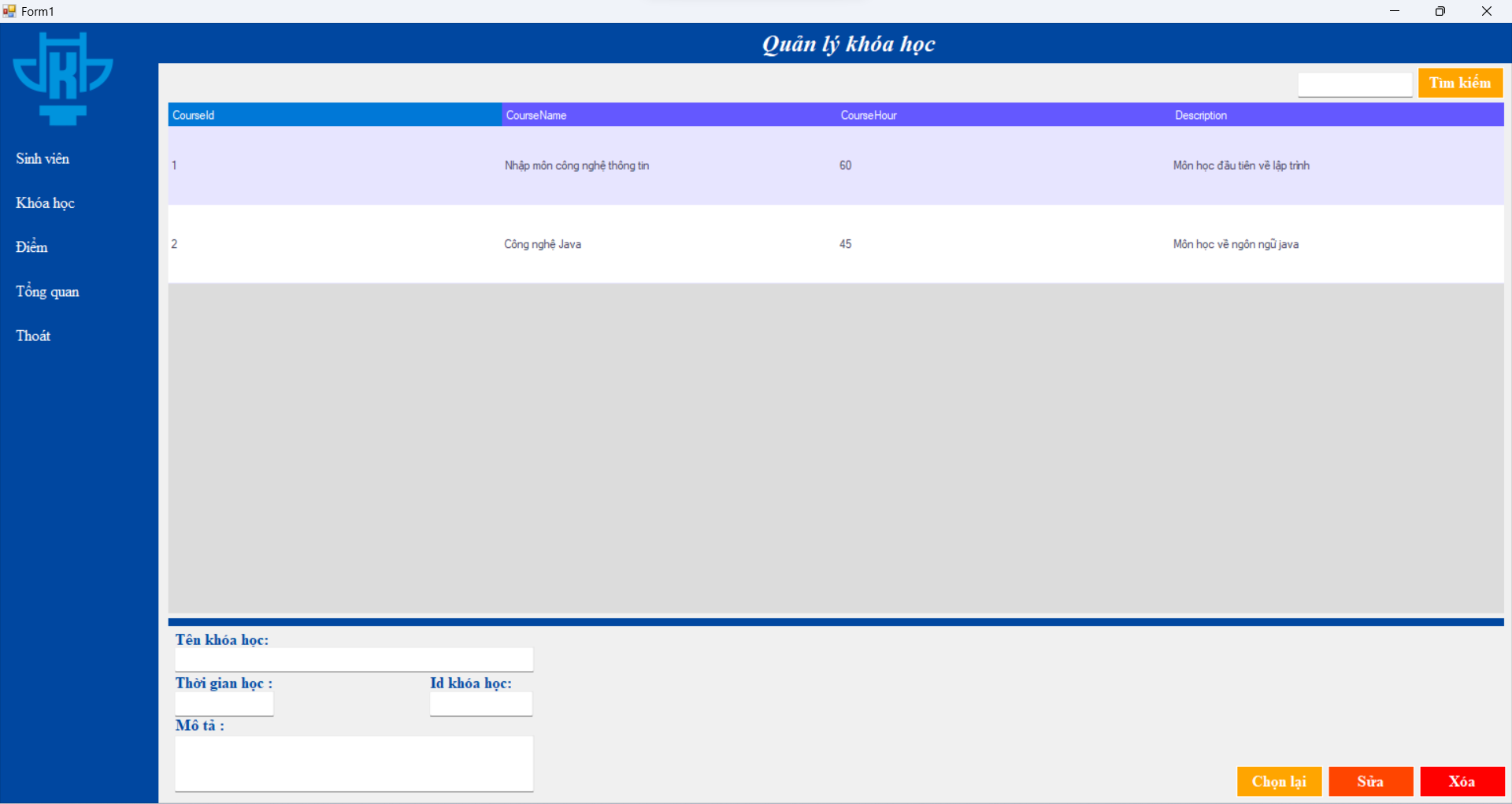
Hình 5.4 : Hình ảnh quản lý sinh viên

## 5.5 Thêm khóa học



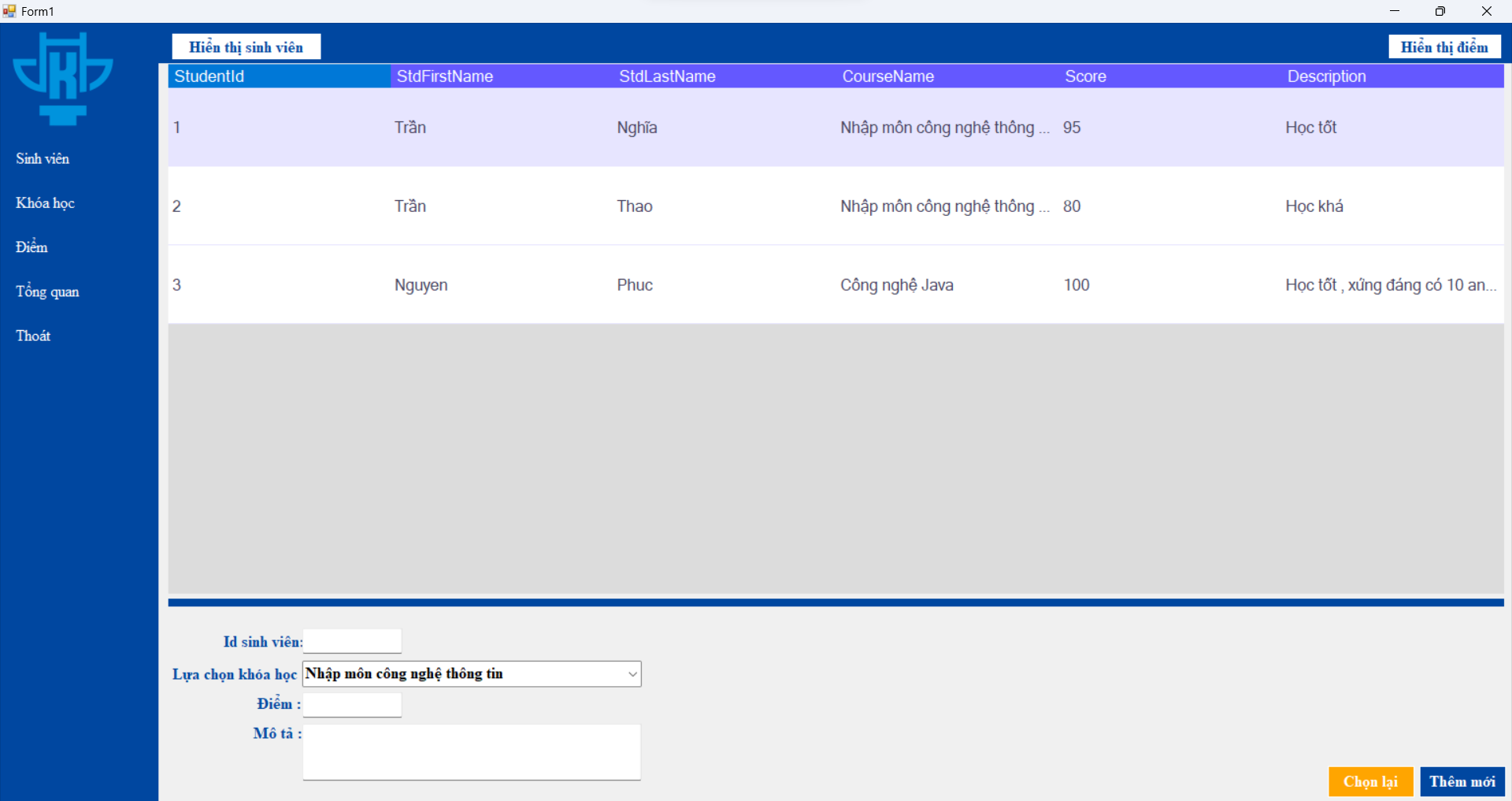
Hình 5.5 : Hình ảnh thêm khóa học

## 5.6 Quản lý khóa học



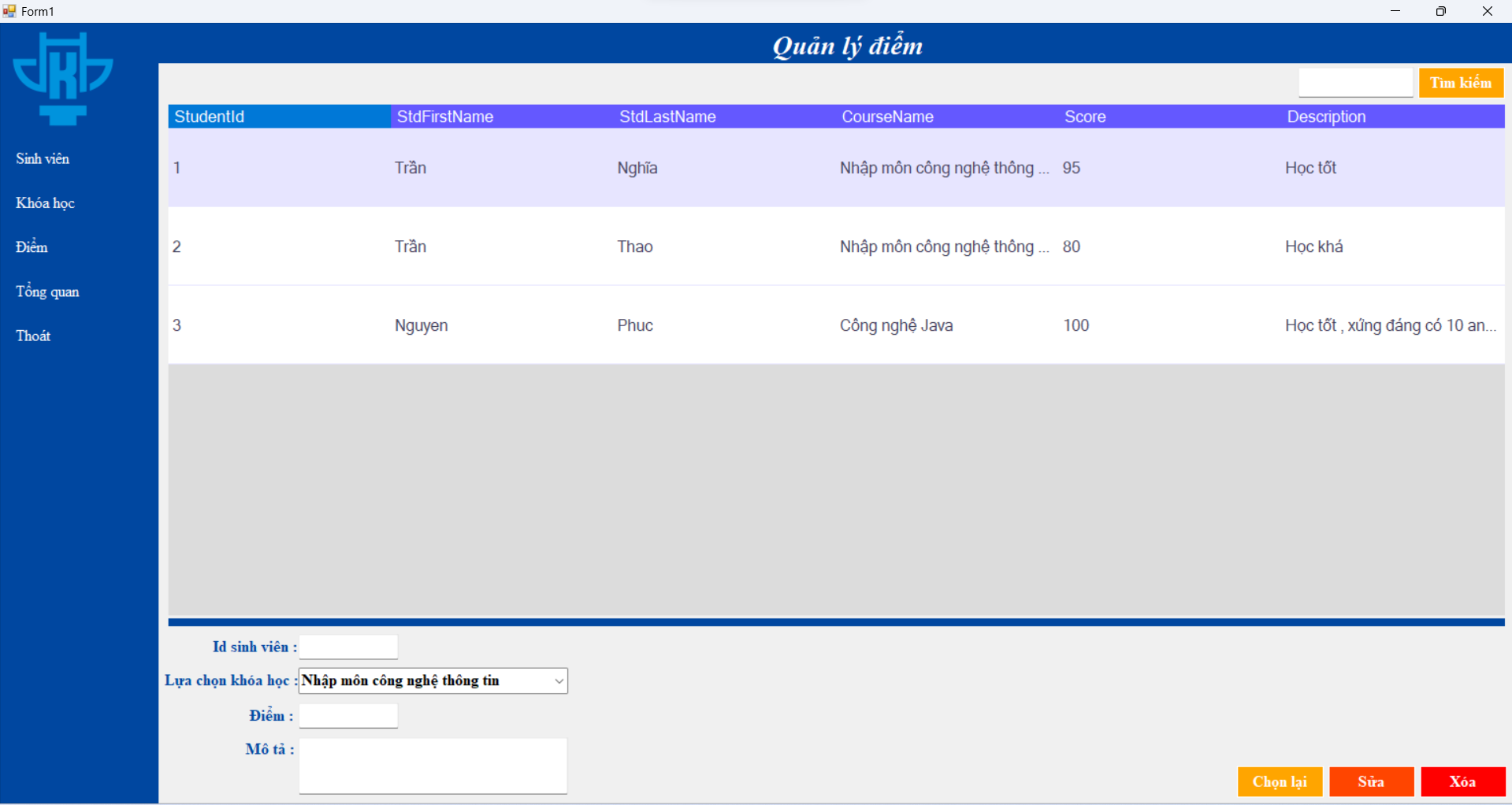
Hình 5.6 : Hình ảnh quản lý khóa học

## 5.7 Thêm điểm mới



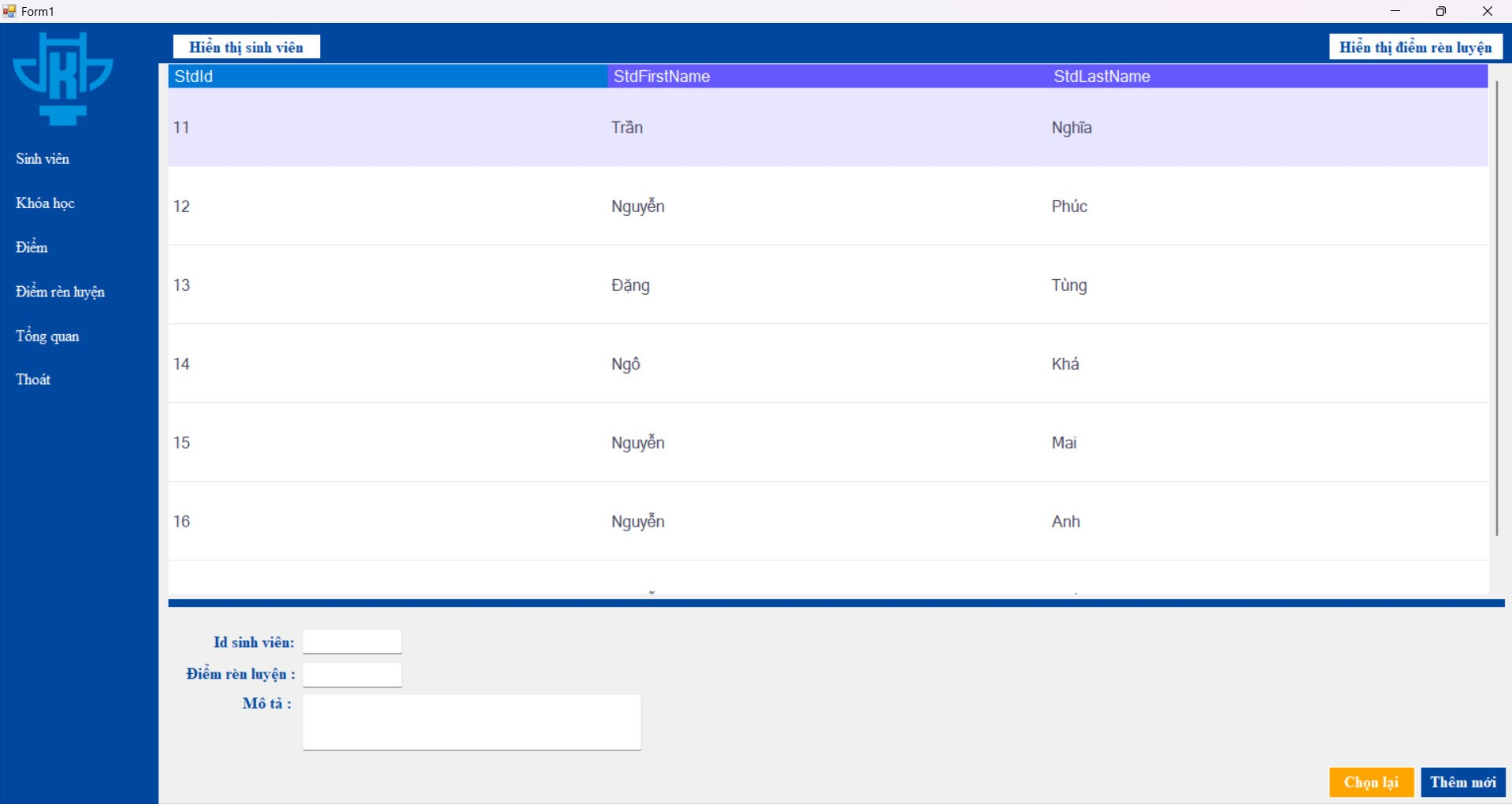
Hình 5.7 : Hình ảnh thêm điểm

## 5.8 Quản lý điểm



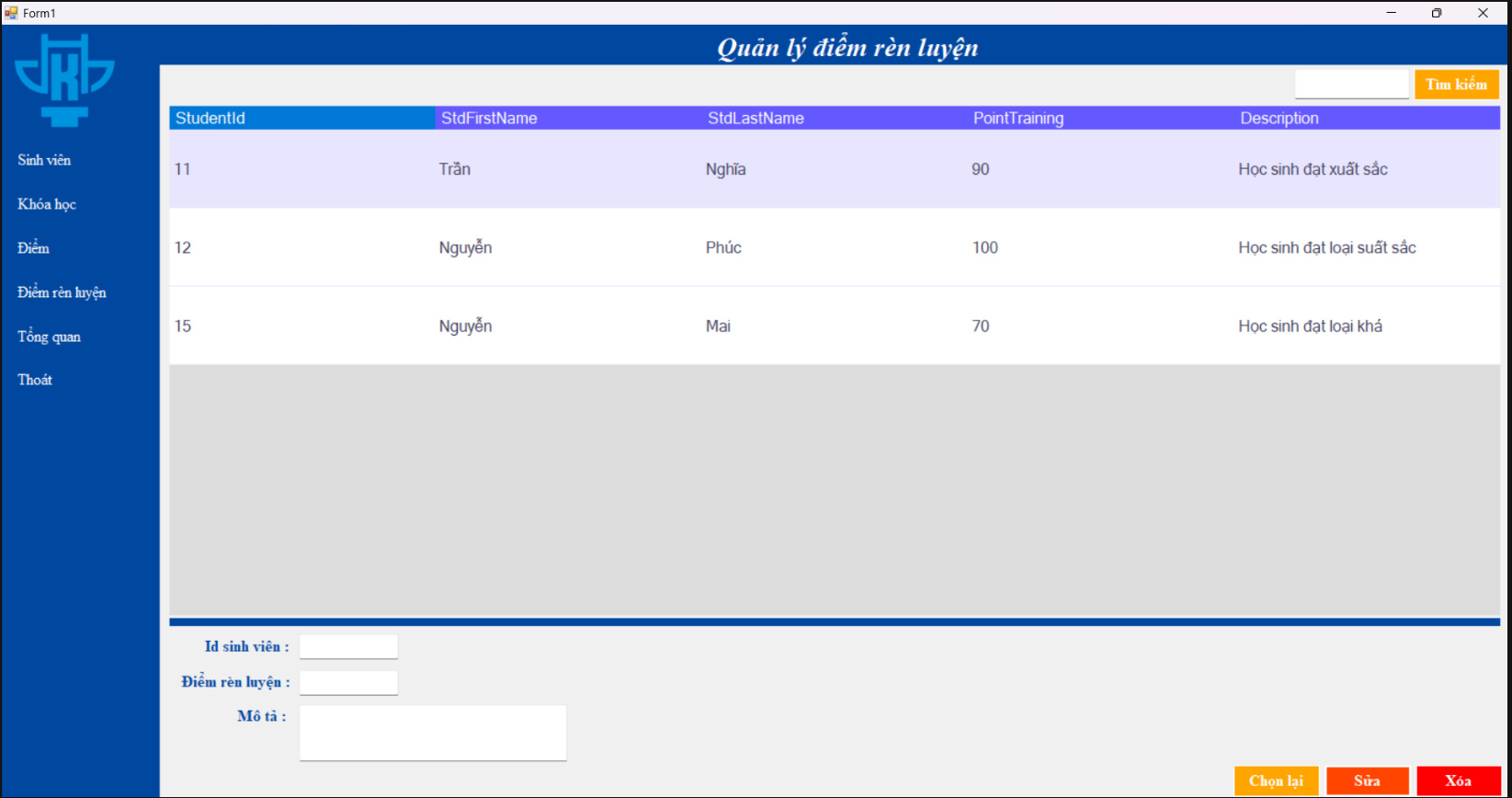
Hình 5.8 : Hình ảnh quản lý điểm

## 5.9 Quản lý thêm điểm rèn luyện



Hình 5.9: Hình ảnh thêm điểm rèn luyện

## 5.10 Quản lý điểm rèn luyện



Hình 5.10: Hình ảnh thêm điểm rèn luyện

Chương 6: Kiểm thử và bảo trì

6.1. Kiểm tra thử:

Kiểm tra đơn vị (Kiểm tra đơn vị):

Xác định và thực hiện kiểm tra các phương thức, hàm riêng trong nguồn mã để đảm bảo chúng hoạt động đúng và đáp ứng yêu cầu kỹ thuật.

Kiểm tra tích hợp (Thử nghiệm tích hợp):

Kiểm tra tính hợp lệ giữa các thành phần khác nhau của hệ thống để đảm bảo tính tương thích chính xác của chúng.

Đảm bảo rằng giao tiếp giữa ứng dụng và cơ sở dữ liệu thông tin ổn định và an toàn.

Kiểm tra hệ thống (Kiểm tra hệ thống):

Thực hiện kiểm tra hệ thống để đảm bảo bộ ứng dụng hoạt động mượt mà và đáp ứng đúng yêu cầu kinh doanh.

Kiểm tra tính năng bảo mật, hiệu suất và khả năng mở rộng của hệ thống.

Kiểm tra Chấp nhận người dùng (Kiểm tra chấp nhận người dùng - UAT):

Tổ chức UAT phiên bản để đảm bảo rằng hệ thống đáp ứng yêu cầu cuối cùng của người dùng.

Nhận phản hồi và thực hiện điều chỉnh cuối cùng dựa trên ý kiến ​​của người dùng.

6.2 Bảo trì

Giám sát và ghi nhật ký:

Thực hiện giám sát liên tục để theo dõi hiệu suất của hệ thống và phát hiện sớm các vấn đề có thể xảy ra.

Sử dụng tính năng ghi nhật ký để ghi lại các sự kiện quan trọng và lỗi, giúp dễ dàng xác định vấn đề và gỡ lỗi.

Cơ sở dữ liệu bảo quản:

Thực hiện sao lưu định nghĩa bất kỳ cơ sở dữ liệu nào để đảm bảo an toàn và khả năng khôi phục sau sự cố.

Thực hiện các nhiệm vụ như tối ưu hóa truy vấn để cải thiện hiệu suất.

Cập nhật và nâng cấp:

Kiểm tra và phát triển các bản cập nhật cập nhật và nâng cấp mới để cải thiện chức năng và bảo mật.

Thực hiện kiểm tra sau khi cập nhật để đảm bảo tính ổn định của hệ thống.

Người dùng hỗ trợ:

Cung cấp kỹ thuật hỗ trợ hỗ trợ cho người dùng khi gặp sự cố.

Xây dựng hệ thống hỗ trợ trực tuyến hoặc tài liệu để giúp người dùng giải quyết vấn đề một cách độc lập.

6.3 Kết luận:

Qua quá trình kiểm tra và bảo trì, hệ thống quản lý sinh viên bằng ngôn ngữ C# và công nghệ .NET đã được đảm bảo tính ổn định, an toàn và đáp ứng yêu cầu đúng. Việc giám sát liên tục và duy trì cập nhật là quan trọng để duy trì hiệu suất và tính bảo mật của hệ thống trong thời gian dài.

Kết luận

Trong quá trình nghiên cứu và xây dựng mô hình quản lý sinh viên bằng ngôn ngữ lập trình C#, chúng tôi đã tiến hành một hành trình khám phá sâu hơn về bản chất và yêu cầu của quản lý sinh viên trong môi trường giáo dục hiện đại. Nhận thức được tầm quan trọng của việc kết hợp giữa quản lý và công nghệ thông tin, chúng tôi đã cố gắng tạo ra một mô hình linh hoạt, toàn diện và đáp ứng được nhu cầu đa dạng của cả sinh viên và nhà trường.

Việc sử dụng ngôn ngữ lập trình C# cho phát triển mô hình quản lý sinh viên đã mang lại nhiều lợi ích rõ ràng. C# không chỉ là một ngôn ngữ mạnh mẽ và linh hoạt, mà còn có cộng đồng hỗ trợ lớn và nhiều tài liệu tham khảo. Điều này giúp chúng tôi có thể nhanh chóng triển khai và phát triển các tính năng phức tạp của mô hình mà không gặp nhiều khó khăn.

Trong quá trình phát triển mô hình, chúng tôi đã tập trung vào việc tối ưu hóa trải nghiệm người dùng. Bằng cách tạo ra giao diện dễ sử dụng và tích hợp các tính năng tiện ích như lập kế hoạch học tập, giám sát tiến trình học tập và tương tác trực tuyến giữa sinh viên và giảng viên, chúng tôi hy vọng mô hình của chúng tôi sẽ giúp tăng cường sự hài lòng và đạt được mục tiêu học tập của sinh viên.

Tuy nhiên, chúng tôi cũng nhận thức được rằng việc phát triển mô hình quản lý sinh viên là một quá trình liên tục và đòi hỏi sự phản hồi và cải tiến liên tục từ cộng đồng người dùng. Chúng tôi sẽ tiếp tục theo dõi và đánh giá hiệu quả của mô hình sau khi triển khai, từ đó đề xuất các cải tiến và bổ sung để nâng cao chất lượng và sự linh hoạt của hệ thống.

Cuối cùng, chúng tôi tin rằng đề tài này không chỉ đóng góp vào lĩnh vực giáo dục mà còn mang lại những giá trị ý nghĩa cho xã hội. Hy vọng rằng mô hình quản lý sinh viên bằng C# sẽ là một công cụ hữu ích và hiệu quả trong việc nâng cao chất lượng giáo dục và tạo ra một môi trường học tập tiên tiến và phát triển cho tương lai.

# Tài liệu tham khảo:

[1] <https://codegym.vn/blog/giao-trinh-c-pdf/>

[2]Giáo\_trình\_Visual\_Studio\_NET\_1\_ĐẠI\_HỌC\_HUẾ\_ĐẠI\_HỌC\_KHOA\_HỌC\_KHOA\_CÔNG\_NGHỆ\_THÔNG\_TIN\_GIÁO\_TRÌNH\_C\_VÀ\_ỨNG\_DỤNG\_NGUYỄN\_HOÀNG\_HÀ\_NGUYỄN\_VĂN\_TRUNG\_HUẾ\_2008

# Phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Họ và tên | Công việc hoàn thành | Kết quả hoàn thành |
| 1 | Trần Trọng Nghĩa(NT) | Chương 4, Chương 5, Tổng hợp nội dung, Code chính dự án. | Hoàn thành |
| 2 | Nguyễn Hồng Phúc | Chương 2, 3.1, Chương 6, Chỉnh sửa bản word. | Hoàn thành |
| 3 | Đặng Thanh Tùng | Chương 1, 3.2, Làm pp. | Hoàn thành |