

[illegible]

(Ký tên và ghi rõ họ tên)

[illegible]

Thành viên hội đồng
(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

Lời đầu tiên cho phép em xin gửi lời cảm ơn chân thành đến Trường Đại học Trà Vinh, khoa Kỹ thuật và Công nghệ đã tạo điều kiện về cơ sở vật chất phục vụ cho quá trình học tập của em trong thời gian qua.

Trong suốt quá trình tìm tòi, mài mò thực hiện để đạt được đồ án cơ sở ngành lần này ngoài yếu tố tự lực của bản thân ra, em còn có một số sự giúp đỡ của các thầy, cô, bạn bè.

Trước hết em xin cảm ơn đến Thầy Trịnh Quốc Việt giảng viên bộ môn công nghệ thông tin của Trường Đại học Trà Vinh, người đã đồng hành tạo nền móng kiến thức về lập trình ứng dụng và hướng dẫn em trong quá trình triển khai thực hiện đồ án. Em cũng xin chân thành cảm ơn tất cả các giảng viên thuộc bộ môn công nghệ thông tin đã giảng dạy các kiến thức về các môn chuyên ngành liên quan để em có thể áp dụng thực hiện vào đồ án lần này.

Đặt biệt em xin cảm ơn tất cả bạn bè và gia đình đã bên cạnh tạo điều kiện thuận lợi để em có thể học tập và hoàn thành tốt nhiệm vụ của mình.

Dù biết rằng nỗ lực đến đâu cũng không thể thiếu phần nào đó sai sót. Kính mong nhận được sự góp ý của Thầy, Cô, bạn bè để em có thể cải thiện tốt hơn không chỉ có đồ án lần này mà còn có những đồ án trong tương lai.

Cuối cùng em xin kính chúc quý Thầy, Cô, quý nhà trường luôn luôn mạnh khỏe. Chúc buổi báo cáo Đồ án cơ sở ngành diễn ra thành công và tốt đẹp.

MỤC LỤC

DANH MỤC HÌNH ẢNH	5
DANH MỤC BẢNG BIỂU	6
TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH	7
MỞ ĐẦU.....	8
CHƯƠNG 1: TỔNG QUAN.....	12
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT	14
2.1 Tổng quan về Windows Forms C#.....	14
2.1.1 Ưu điểm và nhược điểm của Windows Forms	15
2.1.2 Các thành phần cơ bản trong Windows Forms	16
2.2 Tổng quan về SQL Server	16
2.2.1 Kiến trúc SQL Server.....	16
2.2.2 So sánh SQL với NoSQL.....	19
2.2.3 Thiết kế cơ sở dữ liệu.....	20
2.3 Mô hình quản lý điểm thi	20
2.3.1 Chức năng của sinh viên	21
2.3.2 Chức năng của giảng viên.....	21
2.3.3 Chức năng của quản trị viên	21
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU	23
3.1 Mô tả bài toán.....	23
3.1 Yêu cầu chức năng	23
3.2 Yêu cầu phi chức năng	24
3.3 Mô tả các phần mềm sử dụng.....	24
3.3.1 Visual Studio 2022.....	24
3.3.2 SQL Server Management Studio	25
3.4 Thiết kế các sơ đồ cho hệ thống	26
3.4.1 Sơ đồ phân cấp chức năng	26
3.4.2 Sơ đồ thực thể liên kết (Entity Relationship Diagram - ERD)	27
3.4.3 Sơ đồ luồng dữ liệu (Data Flow Diagram - DFD)	28
3.4.4 Sơ đồ Use Case	29
3.5 Thiết kế cơ sở dữ liệu	29
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU	33
4.1 Kết quả đạt được	33
4.2 Đánh giá hiệu năng.....	33
4.3 Trải nghiệm người dùng.....	33
4.4 Một số giao diện của hệ thống	34
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	42
5.1 Kết luận	42
5.2 Hướng phát triển.....	42
DANH MỤC TÀI LIỆU THAM KHẢO	43
PHỤ LỤC.....	44

DANH MỤC HÌNH ẢNH

Hình 2.1 Logo C# và công nghệ .NET	15
Hình 2.2 Logo của Microsoft SQL Server	16
Hình 2.3 Kiến trúc SQL Server	18
Hình 3.1 Logo của Visual Studio 2022	25
Hình 3.2 Logo của SQL Server Management Studio (SSMS)	25
Hình 3.3 Sơ đồ phân cấp chức năng cho hệ thống	27
Hình 3.4 Sơ đồ thực thể liên kết	27
Hình 3.5 Sơ đồ mức ngữ cảnh	28
Hình 3.6 Sơ đồ luồng dữ liệu	28
Hình 3.7 Sơ đồ use case	29
Hình 3.8 Sơ đồ Database Diagrams	30
Hình 4.1 Giao diện đăng nhập	34
Hình 4.2 Giao diện sinh viên	35
Hình 4.3 Giao diện chứa báo cáo	35
Hình 4.4 Biểu mẫu báo cáo	36
Hình 4.5 Giao diện Giảng viên	36
Hình 4.6 Giao diện thao tác với điểm	37
Hình 4.7 Giao diện danh sách sinh viên	37
Hình 4.8 Giao diện quản trị viên	38
Hình 4.9 Giao diện thêm mới học kỳ	38
Hình 4.10 Giao diện thêm môn học mới	39
Hình 4.11 Giao diện thêm tài khoản sinh viên	39
Hình 4.12 Giao diện thêm tài khoản Giảng viên	40
Hình 4.13 Thao tác xóa sinh viên	40
Hình 4.14 Xóa giảng viên	41

DANH MỤC BẢNG BIỂU

Bảng 2.1 So sánh SQL với NoSQL	19
Bảng 3.1 Thực thể sinh viên	30
Bảng 3.2 Thực thể giảng viên	31
Bảng 3.3 Thực thể môn học	31
Bảng 3.4 Thực thể học kỳ	31
Bảng 3.5 Thực thể quản trị.....	31
Bảng 3.6 Thực thể điểm	32

TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH

Tóm tắt vấn đề: trong thời buổi công nghệ đối với một hệ thống quản lý điểm cho sinh viên không những đòi hỏi tính chính xác, bảo mật mà còn phải có các yếu tố dễ nhìn, dễ thao tác đối với người sử dụng. Tôi xây dựng hệ thống quản lý điểm thi cho sinh viên nhằm giải quyết các vấn đề nêu trên.

Hướng tiếp cận: nhận thấy được tầm ảnh hưởng rất lớn của hệ thống quản lý điểm của nhà trường đối với sinh viên, hệ thống quản lý điểm thi được xây dựng nhằm mục tiêu quản lý hiệu quả kết quả học tập của sinh viên các trường Đại học. Các công cụ và công nghệ được sử dụng bao gồm Visual Studio để lập trình giao diện và logic, SQL Server để lưu trữ dữ liệu, cùng với việc áp dụng các thuật toán kiểm tra tính hợp lệ của dữ liệu đầu vào nhằm tăng độ tin cậy cho hệ thống. Quy trình phát triển được thực hiện qua các bước: phân tích và thiết kế hệ thống, triển khai cơ sở dữ liệu và giao diện, xây dựng các chức năng chính, kiểm thử để phát hiện và sửa lỗi. Kết quả kỳ vọng sẽ là một hệ thống hoàn thiện, dễ dàng sử dụng, đáp ứng các yêu cầu quản lý điểm số, đồng thời đảm bảo tính bảo mật và độ chính xác của thông tin.

Cách giải quyết vấn đề: đồng hành với tôi là nền tảng Windows Form kết hợp SQL Server làm cơ sở dữ liệu những phần mềm này đều đảm bảo tính ổn định và bảo mật cao phù hợp với môi trường giáo dục. Dự án được tôi triển khai qua từng giai đoạn từ tìm hiểu hệ thống quản lý điểm của các trường, xây dựng cơ sở dữ liệu đến thiết kế giao diện sau đó hoàn thiện các chức năng như phân quyền đăng nhập, nhập liệu, xem điểm số và minh họa trực quan bằng biểu đồ. Toàn bộ quá trình trên nhằm tạo cho người dùng một trải nghiệm tối ưu.

Kết quả đạt được: xây dựng hoàn thành được một hệ thống với đầy đủ các chức năng. Hệ thống vận hành mượt mà, giao diện thân thiện, người dùng dễ nhìn, cách bố trí các chức năng theo bố cục hợp lý. Các chức năng của hệ thống vận hành tốt, không có lỗi, hiển thị chi tiết kết quả học tập rõ ràng. Sinh viên tra cứu điểm dễ dàng, minh bạch.

MỞ ĐẦU

1. Lý do chọn đề tài

Nhận thấy sự phát triển của công nghệ thông tin ngày càng mạnh mẽ nhất là trong lĩnh vực giáo dục. Việc tối ưu hóa từ các thao tác thủ công sang sử dụng công nghệ là một lợi thế của các trường nếu áp dụng được sẽ góp phần nâng cao hiệu suất giảng dạy cũng như chất lượng đào tạo đầu ra. Một trong những lĩnh vực đó phải kể đến hệ thống quản lý điểm thi cho sinh viên. Đây là phần mà giáo viên và sinh viên đều phải thao tác rất thường xuyên để cập nhật tình hình học tập, ghi lại những điểm số quan trọng. Đây cũng là một trong những nơi có lưu lượng sinh viên truy cập đến đông nhất cho thấy được tầm ảnh hưởng của hệ thống điểm đối với sinh viên là rất lớn.

Những năm gần đây số lượng sinh viên đại học tăng một cách nhanh chóng điều này kéo đến việc hệ thống lưu trữ điểm cho sinh viên phải tiếp thu một số lượng lớn dữ liệu. Các hệ thống quản lý điểm thủ công hay các hệ thống quản lý điểm sử dụng các công nghệ lỗi thời đang trở nên quá tải, phần nào đó hiển thị thông tin không trực quan, thao tác rườm rà và theo thời gian dần thể hiện rõ sự chậm chạp, bất tiện khi đứng trước sự phát triển ngày càng nhanh của công nghệ thông tin song song đó là sự tăng vọt về số lượng sinh viên các khối ngành.

Trước thực trạng này, yêu cầu bức thiết được đặt ra đối với các trường hiện nay là phải có các hệ thống quản lý điểm sử dụng công nghệ mới để bắt kịp với thời buổi công nghệ thông tin đồng thời cần phải tối ưu, an toàn bảo mật cho các thông tin của sinh viên, thuận tiện cho việc thao tác đối với hệ thống và đặc biệt là dữ liệu điểm hiển thị rõ ràng, trực quan.

2. Mục đích

Cung cấp một hệ thống quản lý điểm thi cải tiến cho các hệ thống quản lý cũ, tối ưu hóa được các thao tác thủ công, hạn chế sự sai sót trong quá trình nhập liệu.

Tạo giao diện thân thiện, người dùng dễ nhìn, bố cục được sắp xếp hợp lý, các chức năng của hệ thống hiển thị rõ ràng.

Cung cấp thông tin nhanh chóng và chính xác. Giúp sinh viên, giáo viên, cán bộ quản lý dễ dàng truy cập và nắm bắt thông tin về điểm số, kết quả học tập một cách kịp thời, minh bạch.

Tận dụng các công nghệ phù hợp để xây dựng hệ thống đáp ứng với điều kiện của các trường và có thể dễ dàng triển khai, phát triển trong tương lai.

Đảm bảo các yêu cầu về an toàn và bảo mật dữ liệu, có phân quyền truy cập để hệ thống chỉ cho phép thực hiện các thao tác đúng với thẩm quyền.

3. Nội dung

Hệ thống quản lý điểm thi cho sinh viên hỗ trợ các nội dung như sau:

Đối với sinh viên:

- Cho phép sinh viên xem các điểm (Điểm quá trình 1, điểm quá trình 2, điểm thi) mà sinh viên đã dự thi và đã có kết quả theo các môn.
- Cho phép sinh viên xem các học kỳ đã học.
- Sinh viên có thể xuất ra bảng điểm của mình bao gồm: mã sinh viên, họ tên sinh viên, tên môn học, điểm quá trình 1, điểm quá trình 2, điểm thi, học kỳ của môn học đó.
- Có mô tả dữ liệu điểm của sinh viên bằng biểu đồ giúp cho giảng viên, sinh viên có thể đánh giá quá trình học tập một cách trực quan.

Đối với giảng viên:

- Giảng viên có thể nhập điểm cho sinh viên theo mã môn học.
- Cập nhật điểm cho sinh viên.

Đối với người quản trị:

- Cho phép xóa tài khoản của giảng viên ra khỏi hệ thống.
- Cho phép xóa tài khoản của sinh viên ra khỏi hệ thống đồng thời xóa luôn cơ sở dữ liệu điểm của sinh viên đó.

4. Phương pháp nghiên cứu

Để hoàn thành đồ án lần này trong quá trình thực hiện tôi đã áp dụng nhiều phương pháp nghiên cứu như:

- Thu thập các tài liệu liên quan đến chủ đề của đồ án để có cái nhìn tổng thể trước khi bắt tay vào thực hiện.
- Nghiên cứu, tham khảo các nguồn tài liệu, video hướng dẫn học lập trình về C# để tạo giao diện cho hệ thống tiếp theo là học SQL Server để xây dựng cơ sở dữ liệu.
- Lên kế hoạch, lập đề cương chi tiết cho đề án.
- Trao đổi, tiếp nhận các đóng góp của giảng viên và bạn bè để xác định các yêu cầu thực tế đối với hệ thống quản lý điểm.
- Phân tích các yêu cầu chức năng cho hệ thống: thêm điểm, cập nhật điểm, hiển thị điểm và các yêu cầu phi chức năng như tính bảo mật và bố cục giao diện người dùng.
- Kiểm thử hệ thống, kết nối cơ sở dữ liệu, điều chỉnh các yêu cầu cuối cùng trước khi hoàn thiện sản phẩm.

5. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu:

- Hệ thống tập trung vào quản lý điểm cho sinh viên, giúp sinh viên có cái nhìn chi tiết về kết quả học tập đã đạt được từ đó có các biện pháp, kế hoạch nhằm cải thiện điểm số của các môn còn yếu kém hoặc lên kế hoạch cho các môn học tương lai.
- Giảng viên có thể theo dõi tiến độ học tập của sinh viên, thu thập các dữ liệu điểm của các lớp để có thể đưa ra các quyết định điều chỉnh phương pháp giảng dạy sao cho hiệu quả.

Phạm vi nghiên cứu:

- Tập trung vào việc giải quyết các vấn đề thực tiễn liên quan đến quản lý điểm thi trong môi trường giáo dục, đồng thời đảm bảo tính ứng dụng và mở rộng trong tương lai.
- Xây dựng các chức năng quản lý điểm, chắc lọc, loại bỏ các chức năng không cần thiết.

- Xây dựng giao diện bằng Windows Forms tích hợp ngôn ngữ lập trình C#.
- Thiết kế cơ sở dữ liệu sử dụng công cụ SQL Server Management Studio.
- Áp dụng phương pháp phân quyền truy cập.

CHƯƠNG 1: TỔNG QUAN

Nắm bắt được tâm lý của sinh viên hiện nay ngoài các yếu tố như gia đình, nhà trường, thầy cô, bạn bè thì điểm số cũng là một phần tác động trực tiếp đến quá trình học tập của đến sinh viên, nó không chỉ phản ánh trình độ học vấn trong quá trình học tập mà còn thể hiện chất lượng đào tạo, hệ thống giáo dục của nhà trường.

Ngày nay công nghệ phát triển ngày càng nhanh chóng giải quyết được các vấn đề còn khiếm khuyết trong chương trình đào tạo dễ nhận thấy đó chính là việc xuất hiện các hệ thống giám sát quá trình thi bằng trí tuệ nhân tạo, chấm điểm thi tự động, hệ thống giữ xe sử dụng công nghệ nhận diện khuôn mặt,... Bên cạnh đó có một số mảng quan trọng và liên quan trực tiếp đến sinh viên vẫn chưa được khắc phục điển hình chính là hệ thống quản lý điểm thi.

Một số trường hiện nay vẫn còn sử dụng phương pháp nhập liệu điểm thủ công hoặc các hệ thống điểm thi đã lỗi thời tạo cảm giác khó khăn trong các quá trình thao tác nhập xuất đối với giảng viên, một số còn có hiện tượng sai sót trong quá trình nhập liệu song song đó do sử dụng thao tác thủ công nên sinh viên không thể tra cứu điểm số của mình một cách chủ động, đối với các hệ thống cũ khi có số lượng quá đông sinh viên truy cập trong cùng một thời điểm dễ gây hiện tượng chậm, tắt nghẽn tạo cảm giác khó chịu, đặc biệt các dữ liệu điểm của sinh viên được thể hiện không quá rõ ràng và nổi bật phần nào đó giảm bớt sự hứng thú của sinh viên đối với điểm số nói riêng và của hệ thống quản lý điểm nói chung.

Nhận thấy được các vấn đề trên đề án này được ra đời với triển vọng có thể giải quyết được một phần nào đó các bất cập mà các hệ thống cũ mang lại. Hệ thống sẽ tối ưu cho các thao tác sử dụng thủ công mang lại độ chính xác cao, các dữ liệu được lưu trữ bảo mật. Có áp dụng phân quyền truy cập đảm bảo các thao tác đối với người sử dụng được phân chia rõ ràng.

Đặt biệt hơn hết hệ thống hỗ trợ xuất điểm cho sinh viên có kèm biểu đồ minh họa từ đó sinh viên có cái nhìn trực quan hơn, rõ ràng hơn. Bên cạnh đó hệ thống đảm bảo cho sinh viên có thể truy cập mọi lúc, các cập nhật dữ liệu về điểm được đảm bảo liên tục khi giáo viên nhập vào hệ thống bảo đảm tính kịp thời.

Hệ thống được kỳ vọng sẽ trở thành một giải pháp tối ưu cho việc quản lý điểm trong tương lai. Đồng thời với việc sử dụng các công nghệ vừa hiện đại vừa

tối ưu chi phí, vừa có tính bảo mật cao, dễ dàng bảo trì, nâng cấp hứa hẹn trong tương lai hệ thống sẽ được biết đến và sử dụng rộng rãi.

CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

2.1 Tổng quan về Windows Forms C#

Winform, hay còn gọi là Windows Forms, là một công nghệ phát triển ứng dụng của Microsoft trên nền tảng .NET Framework. Nó được sử dụng để tạo ra các ứng dụng có giao diện người dùng đồ họa (GUI) trên hệ điều hành Windows.

Ra đời vào năm 2003, Winform là công nghệ chủ yếu để lập trình các ứng dụng Windows, bao gồm các phần mềm quản lý, kế toán, công cụ văn phòng và nhiều ứng dụng desktop khác. Các ứng dụng viết trên Winform có giao diện người dùng đồ họa đơn giản, dễ sử dụng và dễ tiếp cận.

Winform cung cấp một nền tảng cho các lập trình viên xây dựng giao diện người dùng với các điều khiển như button, textbox, label, menu và nhiều thành phần khác. Bằng cách kéo thả các thành phần vào form, lập trình viên có thể nhanh chóng tạo ra các ứng dụng với giao diện dễ sử dụng[1].

C# là một ngôn ngữ lập trình thuần hướng đối tượng. Được phát triển bởi Microsoft, C# (phát âm là C sharp) ra đời năm 2000 dưới bàn tay của Anders Hejlsberg. Đây là một loại ngôn ngữ lập trình hiện đại và khá mạnh mẽ. Được xây dựng trên nền tảng hai ngôn ngữ lập trình mạnh nhất là C++ và Java, C# được miêu tả là sự cân bằng giữa C++, Visual Basic, Delphi và Java. Lập trình C# với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF (Windows Presentation Foundation) trở nên rất dễ dàng[2].

Ngôn ngữ C# là ngôn ngữ phổ biến nhất cho nền tảng .NET, một môi trường phát triển mã nguồn mở, đa nền tảng, miễn phí. Các chương trình C# có thể chạy trên nhiều thiết bị khác nhau, từ các thiết bị Internet vạn vật (IoT) đến đám mây và mọi nơi ở giữa. Bạn có thể viết ứng dụng cho điện thoại, máy tính để bàn, máy tính xách tay và máy chủ.

C# là ngôn ngữ đa nền tảng, mục đích chung giúp các nhà phát triển làm việc hiệu quả trong khi viết mã có hiệu suất cao. Với hàng triệu nhà phát triển, C# là ngôn ngữ .NET phổ biến nhất. C# được hỗ trợ rộng rãi trong hệ sinh thái và tất cả các khối lượng công việc .NET. Dựa trên các nguyên tắc hướng đối tượng, nó kết hợp nhiều tính năng từ các mô hình khác, đặc biệt là lập trình chức năng. Các tính

năng cấp thấp hỗ trợ các tình huống hiệu quả cao mà không cần viết mã không an toàn. Hầu hết thời gian chạy và thư viện .NET được viết bằng C# và những tiến bộ trong C# thường mang lại lợi ích cho tất cả các nhà phát triển .NET[3].



Hình 2.1 Logo C# và công nghệ .NET

2.1.1 Ưu điểm và nhược điểm của Windows Forms

Ưu điểm:

Winform dễ học, dễ dạy và thao tác linh hoạt với giao diện kéo thả, dễ dàng sử dụng.

Phù hợp để làm các phần mềm quản lý, thống kê hay kế toán,... Đây là các loại ứng dụng mà công ty hoặc doanh nghiệp nhỏ và vừa rất cần có.

Ngoài ra Winform còn có một số ưu điểm khác như:

- Tốc độ xử lý dữ liệu nhanh chóng, đáp ứng yêu cầu người dùng.
- Đảm bảo thông tin được bảo mật an toàn.
- Winform có thể chạy trên nhiều phiên bản Windows khác nhau.
- Có thể thao tác trên nhiều giao diện khác nhau.

Nhược điểm:

Phần mềm chạy trên hệ điều hành Windows nên người dùng muốn sử dụng phần mềm sẽ phải sử dụng chính máy tính đã cài nó.

Winform chỉ phù hợp với các ứng dụng trên desktop như các ứng dụng quản lý thông tin, tương tác trực tiếp với người dùng.

Giao diện phần mềm thiếu trực quan, khó khăn khi thao tác và không thân thiện với người dùng do đồ họa trên Winform không cao[4].

2.1.2 Các thành phần cơ bản trong Windows Forms

Các thành phần cơ bản trong Windows Forms bao gồm:

- Form (biểu mẫu): đây là thành phần chính của một ứng dụng Windows Form, đóng vai trò như một cửa sổ chứa tất cả các control khác.
- Control (điều khiển): các control là những phần tử nhỏ hơn bên trong form, cho phép người dùng tương tác với ứng dụng, một số control phổ biến: button, label, textbox, combobox, listbox, radiobutton,...
- Event (sự kiện): sự kiện là những hành động mà người dùng hoặc hệ thống thực hiện trên các control.

2.2 Tổng quan về SQL Server

SQL Server là hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS) được Microsoft phát triển và tiếp thị.

Tương tự như các phần mềm RDBMS khác, SQL Server được xây dựng trên SQL, một ngôn ngữ lập trình chuẩn để tương tác với cơ sở dữ liệu quan hệ. SQL Server được liên kết với Transact-SQL hoặc T-SQL, triển khai SQL của Microsoft, bao gồm một tập hợp các cấu trúc lập trình độc quyền.

SQL Server đã có sẵn độc quyền trên môi trường Windows trong hơn 20 năm. Năm 2016, Microsoft đã cung cấp nó trên Linux. SQL Server 2017 đã có sẵn rộng rãi vào tháng 10 năm 2016 và tương thích với cả Windows và Linux[5].



Hình 2.2 Logo của Microsoft SQL Server

2.2.1 Kiến trúc SQL Server

External Protocols (Giao thức bên ngoài): đây là lớp giao tiếp với các ứng dụng hoặc hệ thống bên ngoài thông qua:

- Shared Memory: sử dụng bộ nhớ chung để giao tiếp giữa phần ứng dụng và SQL Server.

- Named Pipes: giao thức cho phép trao đổi dữ liệu giữa các tiến trình trên một máy hoặc mạng.
- TCP/IP: giao thức phổ biến để giao tiếp qua mạng.
- Virtual Interface Adapter (VIA): giao thức chuyên biệt dành cho môi trường tốc độ cao (ít được sử dụng hiện nay).

Database Engine (Bộ máy cơ sở dữ liệu) bao gồm hai thành phần chính:

- Storage Engine (Bộ lưu trữ):
 - + Transaction Services: quản lý các giao dịch và đảm bảo tính toàn vẹn dữ liệu.
 - + File Manager: quản lý các tệp dữ liệu (MDF, LDF).
 - + Buffer Manager: có nhiệm vụ điều khiển bộ nhớ đệm để lưu dữ liệu truy cập tạm thời.
 - + Lock Manager: quản lý khóa để đảm bảo tính nhất quán và tránh các xung đột.
 - + Access Methods: định nghĩa cách dữ liệu được truy cập, bao gồm Rows, Indexes, Pages, Allocations.
 - + Utilities: các công cụ như Bulk Load, DBCC (kiểm tra cơ sở dữ liệu), Backup/Restore.
- Query Processor (Bộ xử lý truy vấn):
 - + Parser: phân tích cú pháp câu truy vấn T-SQL.
 - + Optimizer: tối ưu hóa kế hoạch thực thi truy vấn.
 - + SQL Manager: quản lý các câu lệnh SQL.
 - + Database Manager: điều phối hoạt động trên cơ sở dữ liệu.
 - + Query Executor: thực thi các truy vấn SQL.

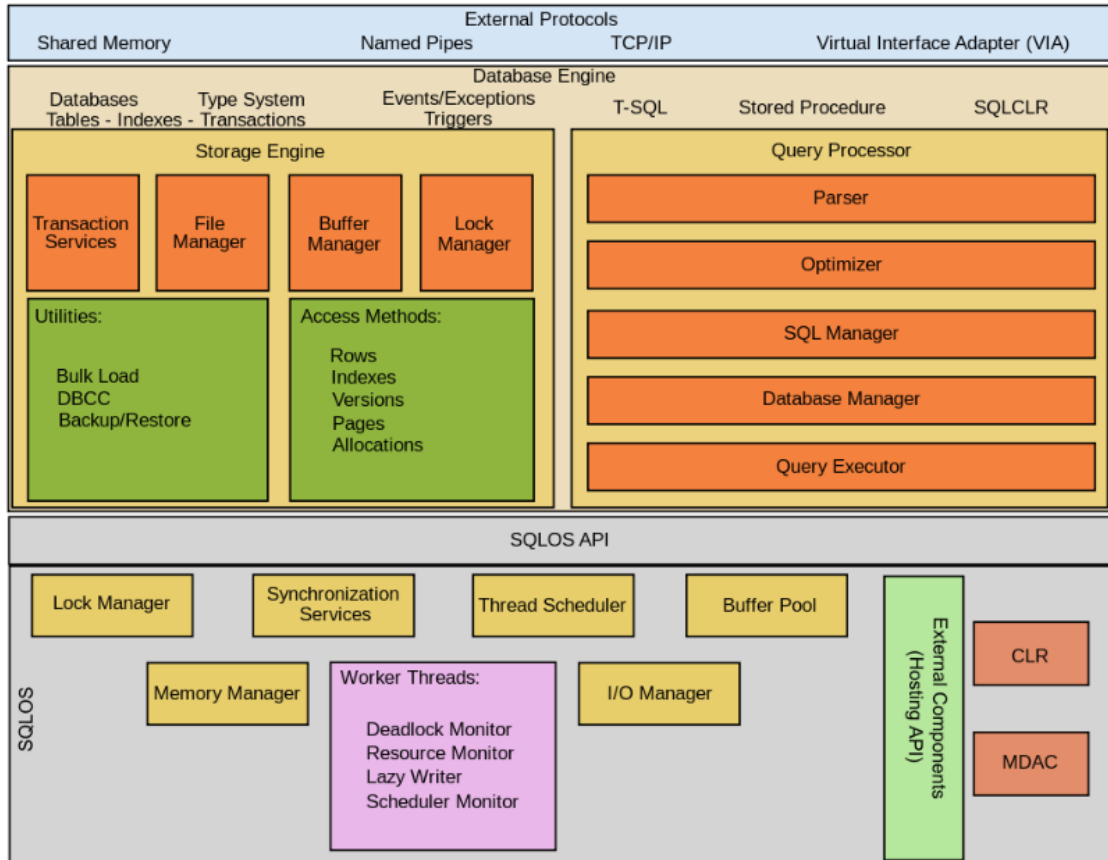
SQLOS (SQL Server Operating System): đây là tầng trừu tượng hoạt động như một hệ điều hành riêng cho SQL Server, bao gồm:

- Memory Manager: quản lý bộ nhớ.

- Lock Manager: quản lý các khóa.
- Thread Scheduler: lập lịch cho các luồng (threads).
- Buffer Pool: quản lý vùng đệm lưu trữ tạm thời.
- Worker Threads: các luồng xử lý bao gồm:
 - + Deadlock Monitor: phát hiện và xử lý deadlock.
 - + Resource Monitor: theo dõi tài nguyên.
 - + Lazy Writer: xử lý ghi dữ liệu xuống đĩa khi cần.
 - + Scheduler Monitor: theo dõi các lịch trình xử lý.

External Components (Thành phần bên ngoài)

- CLR (Common Language Runtime): cho phép chạy mã .NET trong SQL Server.
- MDAC (Microsoft Data Access Components): thành phần hỗ trợ truy cập cơ sở dữ liệu từ các ứng dụng bên ngoài.



Hình 2.3 Kiến trúc SQL Server

2.2.2 So sánh SQL với NoSQL

Bảng 2.1 So sánh SQL với NoSQL

Đặc điểm	SQL	NoSQL
Ngôn ngữ truy vấn	Sử dụng SQL (Structured Query Language)	Sử dụng ngôn ngữ truy vấn linh hoạt tùy thuộc vào loại cơ sở dữ liệu.
Cấu trúc dữ liệu	Dữ liệu có cấu trúc, dựa trên mô hình bảng và quan hệ.	Có thể ở dạng cấu trúc, bán cấu trúc, phi cấu trúc, đa hình...
Quy Mô (Scaling)	Chiều Dọc (Vertical Scaling)	Chiều Ngang (Horizontal Scaling)
Khả Năng Mở Rộng (Scalability)	Vertical Scaling giới hạn bởi khả năng tăng cường tài nguyên trên một máy chủ.	Horizontal Scaling linh hoạt hơn vì có thể thêm máy chủ để chia sẻ công việc và dữ liệu.
Thuộc tính	ACID	BASE
Ưu điểm	Tính linh hoạt, độc lập với dữ liệu, tính nguyên vẹn, tính tương tác, tính thống nhất, phân quyền truy cập	Tính co giãn, hiệu suất, linh hoạt, khả năng sẵn sàng, tính đa dạng, chi phí thấp
Nhược điểm	Chậm và không thể mở rộng được, không thể lưu trữ các dạng dữ liệu có cấu trúc phức tạp. Chi phí phần mềm, nhân lực tốn kém.	Thiếu chuẩn hóa, không đảm bảo hoàn toàn tính ACID. Quản lý dữ liệu lớn trong NoSQL phức tạp, một số NoSQL lưu trữ dữ liệu dạng JSON nên kích thước tài liệu trở nên lớn.

2.2.3 Thiết kế cơ sở dữ liệu

Xác định mục đích của cơ sở dữ liệu, việc này giúp chuẩn bị nền tảng cho các bước tiếp theo.

Tìm và sắp xếp thông tin cần thiết: thu thập mọi loại thông tin có thể muốn ghi vào cơ sở dữ liệu.

Phân chia thông tin vào các bảng: phân chia mục thông tin vào các thực thể hoặc đối tượng chính. Mỗi đối tượng sau đó sẽ trở thành một bảng.

Biên mục thông tin thành các cột: quyết định loại thông tin bạn muốn lưu trữ trong từng bảng. Mỗi mục sẽ trở thành một trường và được hiển thị dưới dạng cột trong bảng.

Chỉ định khóa chính: chọn khóa chính cho mỗi bảng. Khóa chính là một cột, dùng để xác định từng hàng riêng.

Thiết lập mối quan hệ cho bảng: xem bảng và quyết định mối quan hệ của dữ liệu trong bảng đó với dữ liệu ở các bảng khác. Thêm các trường vào bảng hoặc tạo bảng mới để làm rõ mối quan hệ, nếu cần.

Tinh chỉnh thiết kế của bạn: phân tích thiết kế để tìm lỗi. Tạo bảng và thêm một số bản ghi dữ liệu mẫu. Xem liệu bạn có thể nhận được kết quả mong muốn từ bảng không. Tiến hành điều chỉnh thiết kế, nếu cần.

Áp dụng các quy tắc chuẩn hóa: áp dụng các quy tắc chuẩn hóa dữ liệu để xem liệu bảng của bạn đã được cấu trúc phù hợp chưa. Tiến hành điều chỉnh lại bảng, nếu cần.

2.3 Mô hình quản lý điểm thi

Mô hình quản lý điểm thi được xây dựng chia làm 3 thành phần chính: giao diện người dùng sử dụng, các chức năng thao tác với hệ thống và cơ sở dữ liệu lưu trữ bên trong.

Giao diện người dùng sử dụng bao gồm cho sinh viên, giảng viên và người quản trị. Để bắt đầu sử dụng với hệ thống người dùng cần phải thực hiện thao tác đăng nhập, thông tin đăng nhập ở đây gồm có mã số sinh viên, mã giảng viên hoặc

mã người quản trị và mật khẩu. Các thông tin trên sẽ được cơ sở dữ liệu của các trường cung cấp trước.

Sau khi hoàn thành quá trình đăng nhập, hệ thống sử dụng tính năng phân quyền truy cập đối với các đối tượng sử dụng đã được phân loại dựa vào cơ sở dữ liệu, việc này đảm bảo rằng các thao tác không bị trùng lặp, không gây hiểu lầm cho người dùng tránh được các rủi ro về gian lận điểm số.

Điểm số sẽ được giảng viên cập nhật và trong quá trình cập nhật sẽ lưu trữ lại bên trong cơ sở dữ liệu, giảng viên chỉ có quyền nhập điểm và cập nhật điểm cho sinh viên, đối với trường hợp sinh viên nghỉ học hoặc các trường hợp khác gây ảnh hưởng đến cơ sở dữ liệu thì lúc này người quản trị sẽ là người có quyền xóa dữ liệu điểm của sinh viên đó ra cơ sở dữ liệu.

2.3.1 Chức năng của sinh viên

Sau khi đăng nhập vào hệ thống, sinh viên có thể xem các học kỳ mà mình đã học, các điểm số đã có trong quá trình học tập. Ngoài ra hệ thống hỗ trợ sinh viên kết xuất điểm nếu có nhu cầu, hệ thống cung cấp thêm biểu đồ điểm cho sinh viên có thể dễ dàng đánh giá được kết quả học tập của mình.

2.3.2 Chức năng của giảng viên

Giảng viên là một thành phần quan trọng trong hệ thống, khi đã có điểm của sinh viên giảng viên sẽ tiến hành nhập liệu điểm vào trong cơ sở dữ liệu theo cấu trúc: mã sinh viên, mã môn học, mã học kỳ, các điểm số sinh viên đã có. Nếu muốn cập nhật lại điểm cho sinh viên giảng viên cũng sẽ thực hiện tương tự các thao tác trên chỉ thay thế các kết quả học tập mới.

Giảng viên cũng có thể xem danh sách các sinh viên đã được mình nhập điểm để thuận tiện cho các công việc quản lý khác nếu có.

2.3.3 Chức năng của quản trị viên

Quản trị viên là người có quyền cao nhất trong hệ thống. Xóa tài khoản giảng viên ra khỏi hệ thống khi giảng viên không còn công tác giảng dạy tại trường, xóa tài khoản sinh viên nếu sinh viên nghỉ học và không còn các hoạt động liên quan tại trường.

Cấp tài khoản cho giảng viên mới hoặc sinh viên mới nhập học và làm việc tại trường.

Thêm học kỳ mới, môn học mới vào chương trình đào tạo nếu cần.

Do việc xóa tài khoản ảnh hưởng rất lớn đến người dùng nên việc xóa tài khoản sẽ được diễn ra lâu và cần tìm hiểu kỹ các thông tin để tiến hành xác nhận các thao tác xóa.

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

3.1 Mô tả bài toán

Thiết kế hệ thống quản lý điểm cho sinh viên đáp ứng được các yêu cầu nhập liệu, kết xuất, chỉnh sửa, cập nhật các thông tin liên quan đến điểm số của sinh viên, khắc phục được các bất cập trong các hệ thống cũ. Hệ thống tập trung xử lý trên 3 giao diện chính gồm: sinh viên, giảng viên, quản trị viên.

Học tập, nghiên cứu các câu lệnh, các cấu trúc lập trình với ngôn ngữ C# trên công cụ Visual Studio, cách tạo ra giao diện cho người dùng, xây dựng các chức năng cho phép tương tác với hệ thống. Nghiên cứu về cơ sở dữ liệu SQL Server thông qua các tài liệu, video hướng dẫn.

Tìm hiểu cách kết nối cơ sở dữ liệu với các trang giao diện người dùng từ đó tiến hành xây dựng nên cơ sở dữ liệu lưu trữ các thông tin.

Sử dụng các công cụ để phát thảo các giao diện cho hệ thống, chỉnh sửa, phối hợp các màu sắc, hình ảnh cho phù hợp.

Xác định các thông tin của hệ thống cần lưu trữ sau đó tiến hành thiết kế cơ sở dữ liệu bao gồm các bảng, các thực thể có trong bảng, các mối quan hệ giữa các bảng và thực hiện các câu truy vấn cần thiết.

Cuối cùng sau khi đã xây dựng hoàn thành cấu trúc chi tiết của hệ thống tiến hành các công việc sửa chữa các lỗi về dữ liệu đầu vào, các lỗi logic. Đưa cơ sở dữ liệu vào để tiến hành kiểm nghiệm các tính năng, phát hiện những sai sót nếu có, kiểm tra hiệu năng, thời gian phản hồi các thao tác của người dùng, kiểm tra bảo mật trước khi tiến hành hoàn thiện sản phẩm.

3.1 Yêu cầu chức năng

Đối với sinh viên:

- Chức năng xem các học kỳ, các môn học đã học và điểm số tương ứng của môn học đó khi đăng nhập đúng với tài khoản sinh viên.
- Chức năng kết xuất: xuất các dữ liệu điểm mà sinh viên có để tạo các báo cáo có hỗ trợ biểu đồ cho sinh viên dễ dàng đánh giá kết quả học tập của mình để có hướng cải thiện.

Đối với giảng viên:

- Sau khi đăng nhập hệ thống giảng viên có thể nhập điểm cho sinh viên theo môn học của giảng viên đó.
- Xem danh sách điểm các sinh viên mà giảng viên đã nhập.

Đối với quản trị viên hệ thống:

- Sau khi đăng nhập hệ thống quản trị viên có quyền quản lý các tài khoản của sinh viên và giảng viên.
- Có quyền thêm các học kỳ mới, môn học mới.
- Cấp tài khoản mới cho sinh viên, giảng viên.

3.2 Yêu cầu phi chức năng

- Thiết kế hệ thống với các bố cục được chia hợp lý, màu sắc hài hòa, hình ảnh sinh động.
- Tốc độ phản hồi đối với các thao tác người dùng nhanh chóng, cung cấp trải nghiệm mượt mà, kịp thời.
- Kết quả truy xuất các dữ liệu nhanh chóng, chính xác theo yêu cầu.
- Khả năng mở rộng để tăng phạm vi lưu trữ và lượng truy cập lớn.
- Hệ thống có phân quyền truy cập đảm bảo tính bảo mật các dữ liệu thông tin người dùng.

3.3 Mô tả các phần mềm sử dụng

3.3.1 Visual Studio 2022

Visual Studio là một công cụ phát triển mạnh mẽ, có thể sử dụng để hoàn thành toàn bộ chu kỳ phát triển tại một nơi. Đây là một môi trường phát triển tích hợp (IDE) toàn diện có thể sử dụng để viết, chỉnh sửa, gỡ lỗi và xây dựng mã. Sau đó triển khai ứng dụng. Visual Studio bao gồm trình biên dịch, công cụ hoàn thiện mã, kiểm soát nguồn, tiện ích mở rộng và nhiều tính năng khác để nâng cao mọi giai đoạn của quy trình phát triển phần mềm.

Với nhiều tính năng và ngôn ngữ hỗ trợ trong Visual Studio, chúng ta có thể phát triển từ việc viết chương trình "Hello World" đầu tiên của mình

thành phát triển và triển khai ứng dụng. Ví dụ, xây dựng, gỡ lỗi và thử nghiệm các ứng dụng .NET và C++, chỉnh sửa các trang ASP.NET trong chế độ xem của nhà thiết kế web, phát triển các ứng dụng di động và máy tính để bàn đa nền tảng với .NET hoặc xây dựng Giao diện người dùng web phản hồi trong C#.



Hình 3.1 Logo của Visual Studio 2022

3.3.2 SQL Server Management Studio

SQL Server Management Studio là một môi trường tích hợp để quản lý bất kỳ cơ sở hạ tầng SQL nào. Sử dụng SSMS để truy cập, cấu hình, quản lý, quản trị và phát triển tất cả các thành phần của SQL Server, Azure SQL Database, Azure SQL Managed Instance, SQL Server trên Azure VM và Azure Synapse Analytics.

SSMS cung cấp một tiện ích toàn diện duy nhất kết hợp một nhóm lớn các công cụ đồ họa với nhiều trình soạn thảo tập lệnh phong phú để cung cấp quyền truy cập vào SQL Server cho các nhà phát triển và quản trị viên cơ sở dữ liệu ở mọi cấp độ kỹ năng.



Hình 3.2 Logo của SQL Server Management Studio (SSMS)

3.4 Thiết kế các sơ đồ cho hệ thống

3.4.1 Sơ đồ phân cấp chức năng

Quản lý tài khoản:

- Cấp tài khoản sinh viên: tạo tài khoản cho sinh viên mới nhập học để truy cập vào hệ thống.
- Cấp tài khoản giảng viên: tạo tài khoản cho giảng viên bắt đầu công tác tại trường để họ có thể sử dụng hệ thống.
- Xóa tài khoản sinh viên: xóa bỏ tài khoản sinh viên khi sinh viên nghỉ học hoặc khi gặp sự cố khác.
- Xóa tài khoản giảng viên: xóa bỏ tài khoản giảng viên khi giảng viên ngưng công tác giảng dạy tại trường.

Quản lý điểm:

- Thêm điểm: nhập điểm cho sinh viên theo môn học, học kỳ.
- Cập nhật điểm: chỉnh sửa điểm đã nhập trước đó.
- Xóa điểm: xóa dữ liệu điểm của sinh viên.

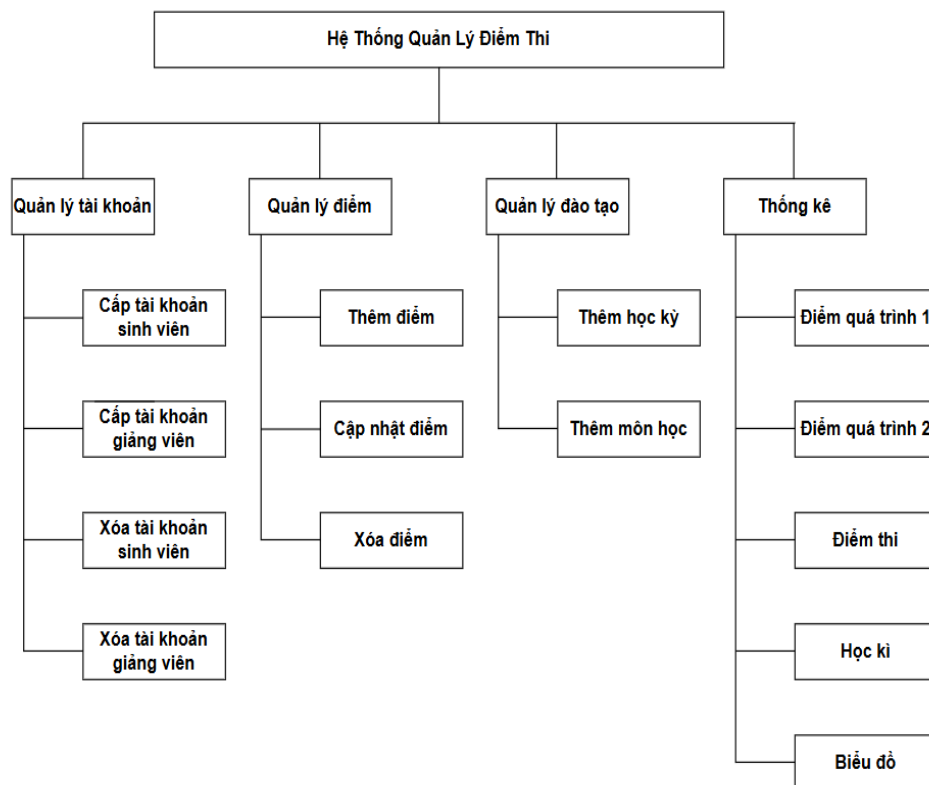
Quản lý đào tạo:

- Thêm học kỳ: thêm học kỳ mới cho các năm học tiếp theo.
- Thêm môn học: thêm các môn học mới vào hệ thống.

Thống kê:

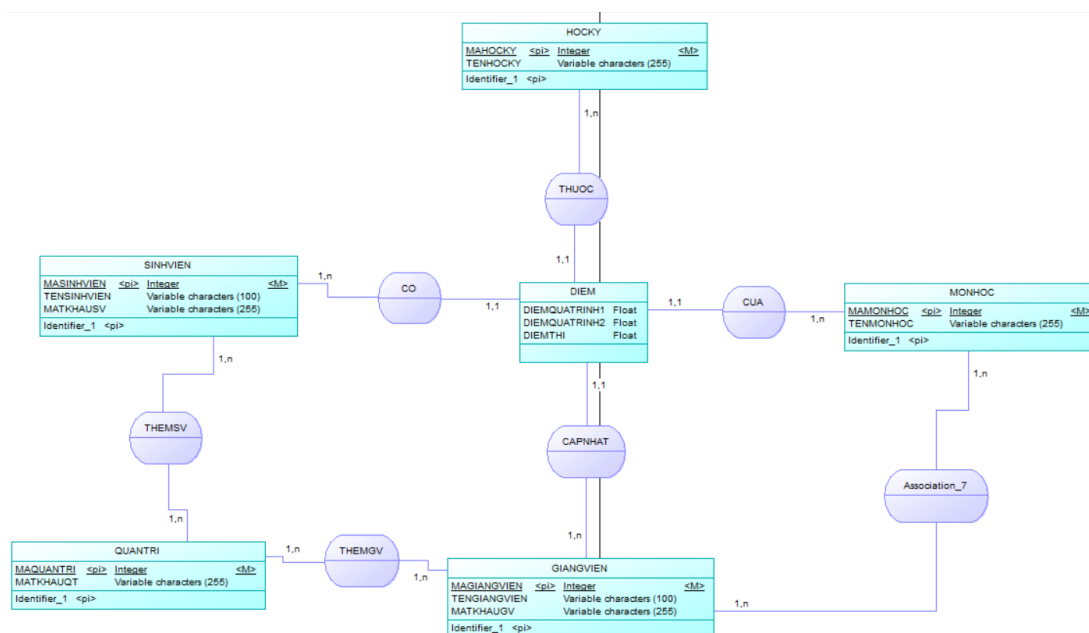
- Điểm quá trình 1: hiển thị thống kê điểm của quá trình học tập giai đoạn 1.
- Điểm quá trình 2: hiển thị thống kê điểm của quá trình học tập giai đoạn 2.
- Điểm thi: hiển thị điểm thi cuối kỳ hoặc điểm thi tổng kết.
- Học kỳ: thống kê các môn học tương ứng với các học kỳ.
- Biểu đồ: biểu diễn dữ liệu thống kê bằng biểu đồ trực quan.

Dưới đây là hình minh họa các chức năng cho hệ thống:



Hình 3.3 Sơ đồ phân cấp chức năng cho hệ thống

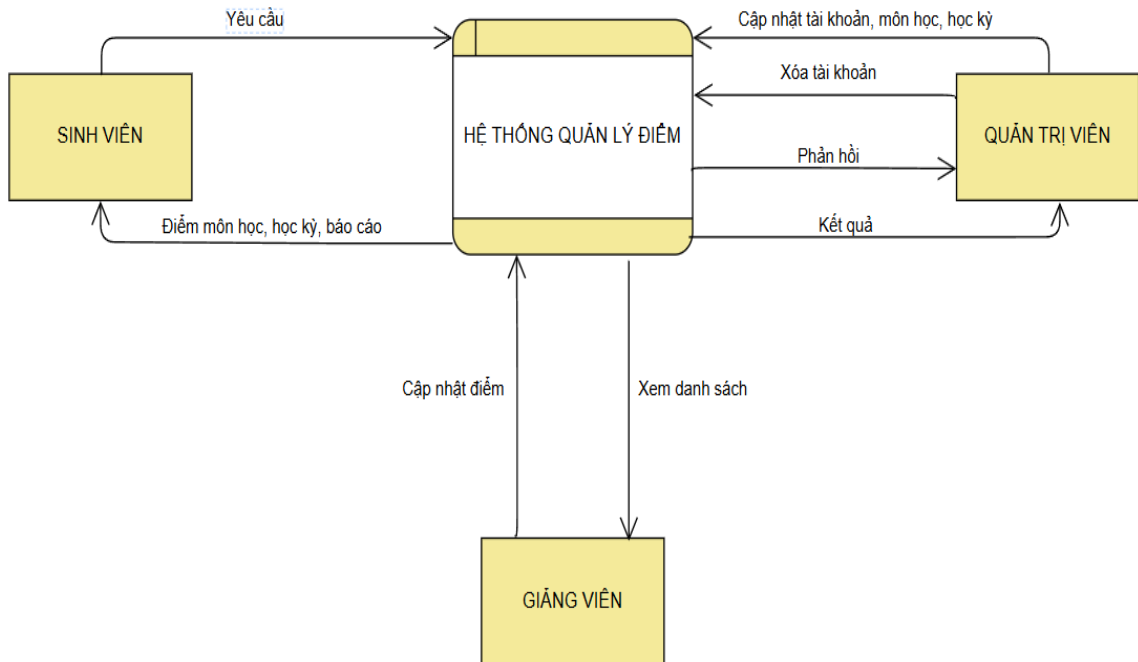
3.4.2 Sơ đồ thực thể liên kết (Entity Relationship Diagram - ERD)



Hình 3.4 Sơ đồ thực thể liên kết

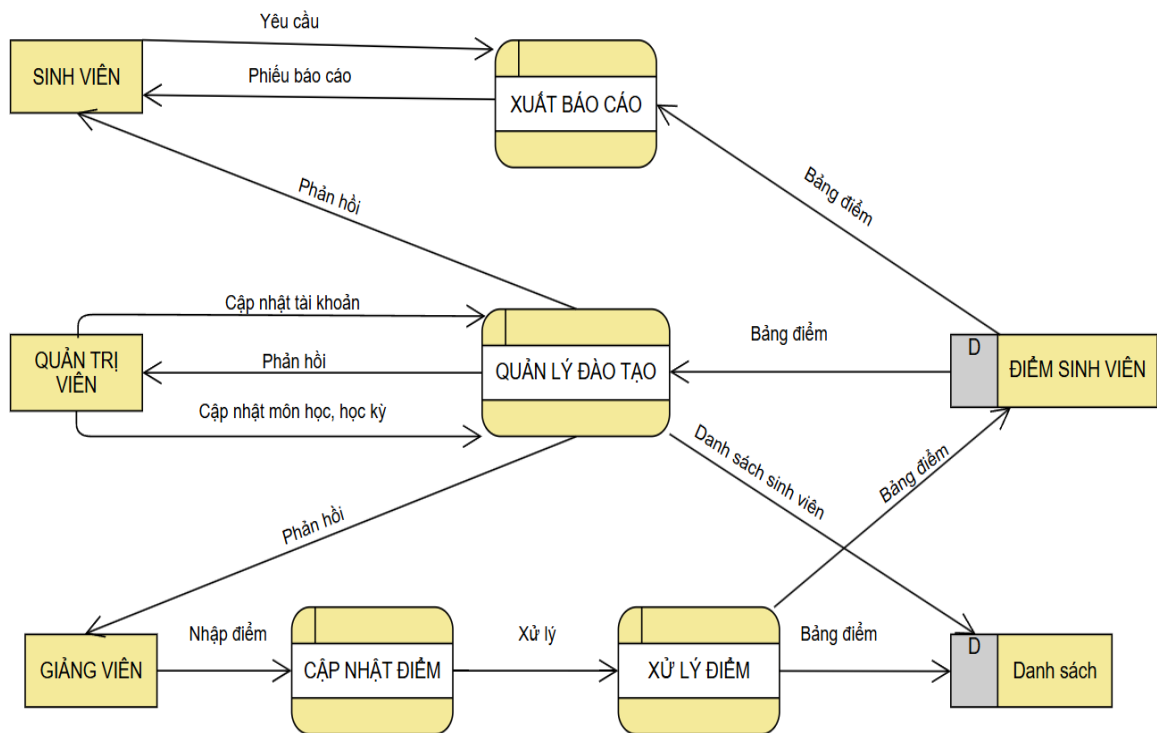
3.4.3 Sơ đồ luồng dữ liệu (Data Flow Diagram - DFD)

- Sơ đồ mức ngữ cảnh



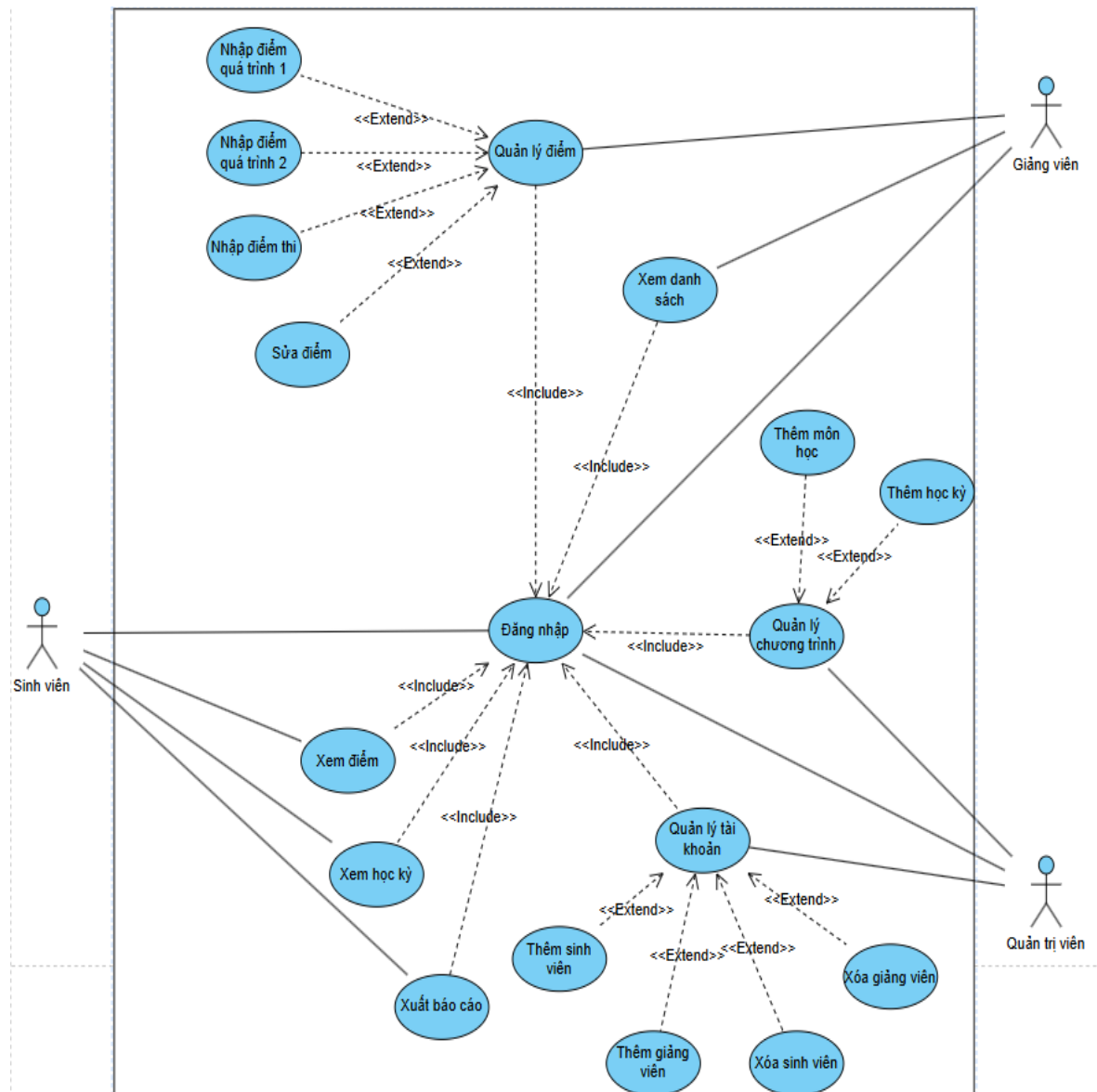
Hình 3.5 Sơ đồ mức ngữ cảnh

- Sơ đồ luồng dữ liệu



Hình 3.6 Sơ đồ luồng dữ liệu

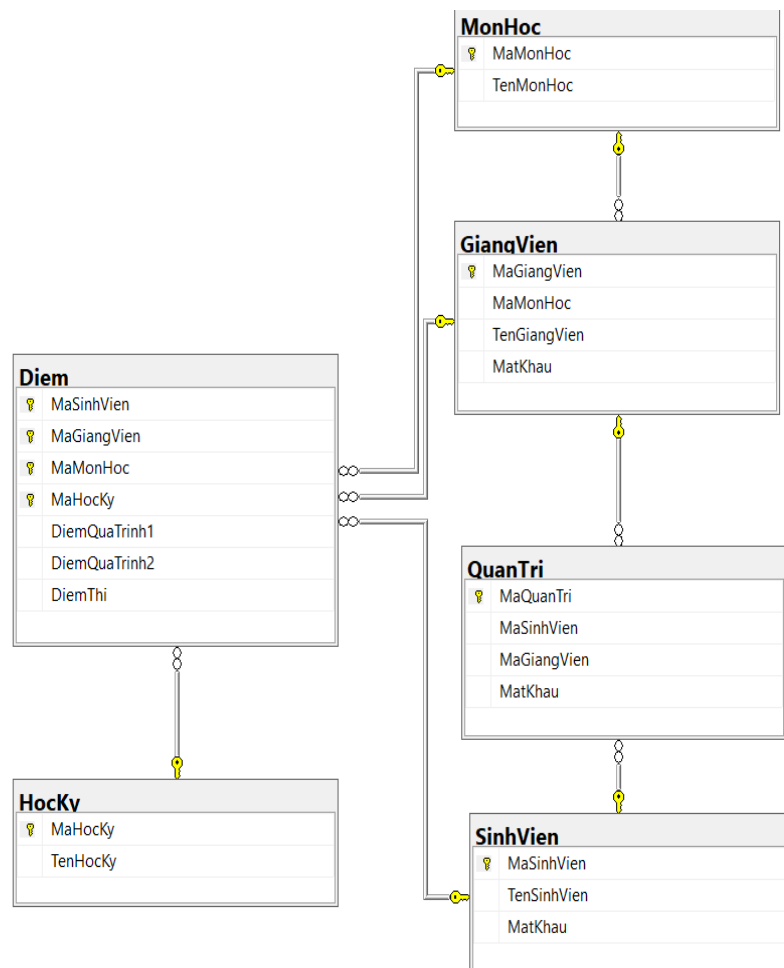
3.4.4 Sơ đồ Use Case



Hình 3.7 Sơ đồ use case

3.5 Thiết kế cơ sở dữ liệu

Sơ đồ Database Diagrams của hệ thống quản lý điểm thi cho sinh viên.



Hình 3.8 Sơ đồ Database Diagrams

Chi tiết các thực thể có trong hệ thống:

Bảng 3.1 Thực thể sinh viên

<i>STT</i>	<i>Thuộc tính</i>	<i>Diễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
1	MaSinhVien	Mã số sinh viên	int	Khóa chính
2	TenSinhVien	Tên sinh viên	nvarchar(100)	
3	MatKhau	Mật khẩu tài khoản sinh viên	nvarchar(255)	

Bảng 3.2 Thực thể giảng viên

<i>STT</i>	<i>Thuộc tính</i>	<i>Diễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
1	MaGiangVien	Mã số giảng viên	int	Khóa chính
2	TenGiangVien	Tên giảng viên	nvarchar(100)	
3	MaMonHoc	Mã số môn học do giảng viên phụ trách giảng dạy	int	Khóa ngoại
4	MatKhau	Mật khẩu tài khoản giảng viên	nvarchar(255)	

Bảng 3.3 Thực thể môn học

<i>STT</i>	<i>Thuộc tính</i>	<i>Diễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
1	MaMonHoc	Mã số môn học	int	Khóa chính
2	TenMonHoc	Tên môn học	nvarchar(255)	

Bảng 3.4 Thực thể học kỳ

<i>STT</i>	<i>Thuộc tính</i>	<i>Diễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
1	MaHocKy	Mã số học kỳ	int	Khóa chính
2	TenHocKy	Tên học kỳ	nvarchar(255)	

Bảng 3.5 Thực thể quản trị

<i>STT</i>	<i>Thuộc tính</i>	<i>Diễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
1	MaQuanTri	Mã số quản trị viên	int	Khóa chính
2	MaSinhVien	Mã sinh viên được quản trị viên cung cấp	nvarchar(255)	Khóa ngoại
3	MaGiangVien	Mã giảng viên được quản trị viên thêm vào hệ thống	int	Khóa ngoại
4	MatKhau	Mật khẩu tài khoản quản trị viên	nvarchar(255)	

Bảng 3.6 Thực thể điểm

<i>STT</i>	<i>Thuộc tính</i>	<i>Diễn giải</i>	<i>Kiểu dữ liệu</i>	<i>Ràng buộc toàn vẹn</i>
1	DiemQuaTrinh1	Điểm quá trình lần 1	float	
2	DiemQuaTrinh1	Điểm quá trình lần 2	float	
3	DiemThi	Điểm thi	float	
4	MaSinhVien	Mã số sinh viên	int	Khóa ngoại
5	MaGiangVien	Mã số giảng viên	nvarchar(255)	Khóa ngoại
6	MaMonHoc	Mã số môn học	int	Khóa ngoại
7	MaHocKy	Mã số học kỳ	nvarchar(255)	Khóa ngoại

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

4.1 Kết quả đạt được

Trải qua quá trình thực hiện và vận hành hệ thống quản lý điểm thi cho sinh viên chúng tôi đã đạt được một số kết quả sau:

- Xây dựng thành công ứng dụng quản lý điểm với giao diện trực quan, thân thiện và đầy đủ chức năng cơ bản như quản lý tài khoản, quản lý điểm, và thống kê, kết xuất dữ liệu.
- Hệ thống hoạt động ổn định, đảm bảo tính chính xác trong việc lưu trữ và xử lý dữ liệu.
- Tích hợp các công cụ thống kê và báo cáo, giúp người dùng dễ dàng theo dõi kết quả học tập.

Đồ án sử dụng các kiến thức lập trình về Windows Forms C# và xây dựng cơ sở dữ liệu bằng SQL Server từ đó các kiến thức ngày càng được củng cố và nâng cao kỹ năng lập trình của cá nhân.

4.2 Đánh giá hiệu năng

- Hệ thống thực hiện tốt và mượt mà với các chức năng được lập trình.
- Thời gian trung bình để phản hồi các yêu cầu của người dùng ngắn.
- Việc cập nhật điểm số vào cơ sở dữ liệu được diễn ra nhanh chóng, kịp thời.
- Tăng cường tính năng minh bạch của hệ thống bằng phân quyền truy cập.
- Giao diện thân thiện, người dùng dễ nhìn, các bố cục được sắp xếp hợp lý.
- Cải thiện đáng kể trải nghiệm của người dùng so với các hệ thống cũ.

4.3 Trải nghiệm người dùng

Giao diện thân thiện: thiết kế giao diện tối giản nhưng vẫn đảm bảo có đầy đủ các chức năng, dễ sử dụng cho từng đối tượng.

Tính tương tác: hệ thống đáp ứng tốt các thao tác người dùng không các nút, thông báo to, rõ dễ quan sát không gây khó hiểu, nhầm lẫn.

Trải nghiệm liền mạch: người sử dụng có thể hoàn thành các yêu cầu mong muốn đối với hệ thống mà không gặp các khó khăn nào.

4.4 Một số giao diện của hệ thống

Màn hình: đăng nhập vào hệ thống.

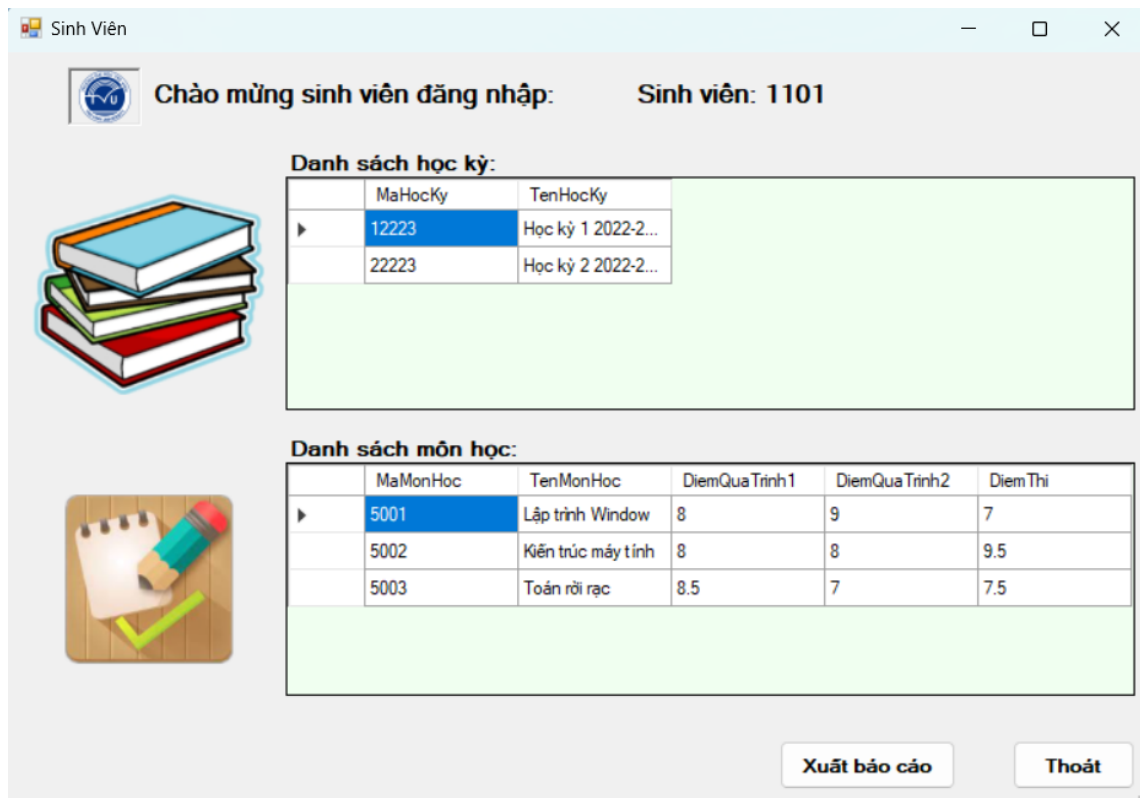
Mô tả: giao diện ban đầu khi người dùng truy cập vào hệ thống, tại đây người dùng tiến hành nhập thông tin tài khoản do quản trị viên cung cấp.



Hình 4.1 Giao diện đăng nhập

Màn hình: xem các thông tin về học kỳ, điểm môn học.

Mô tả: sinh viên đăng nhập vào hệ thống có thể xem các học kỳ đã học, các môn đã học và điểm số các môn tương ứng, có thể xuất báo cáo nếu cần.



Sinh Viên

Chào mừng sinh viên đăng nhập: **Sinh viên: 1101**

Danh sách học kỳ:

MaHocKy	TenHocKy
12223	Học kỳ 1 2022-2...
22223	Học kỳ 2 2022-2...

Danh sách môn học:

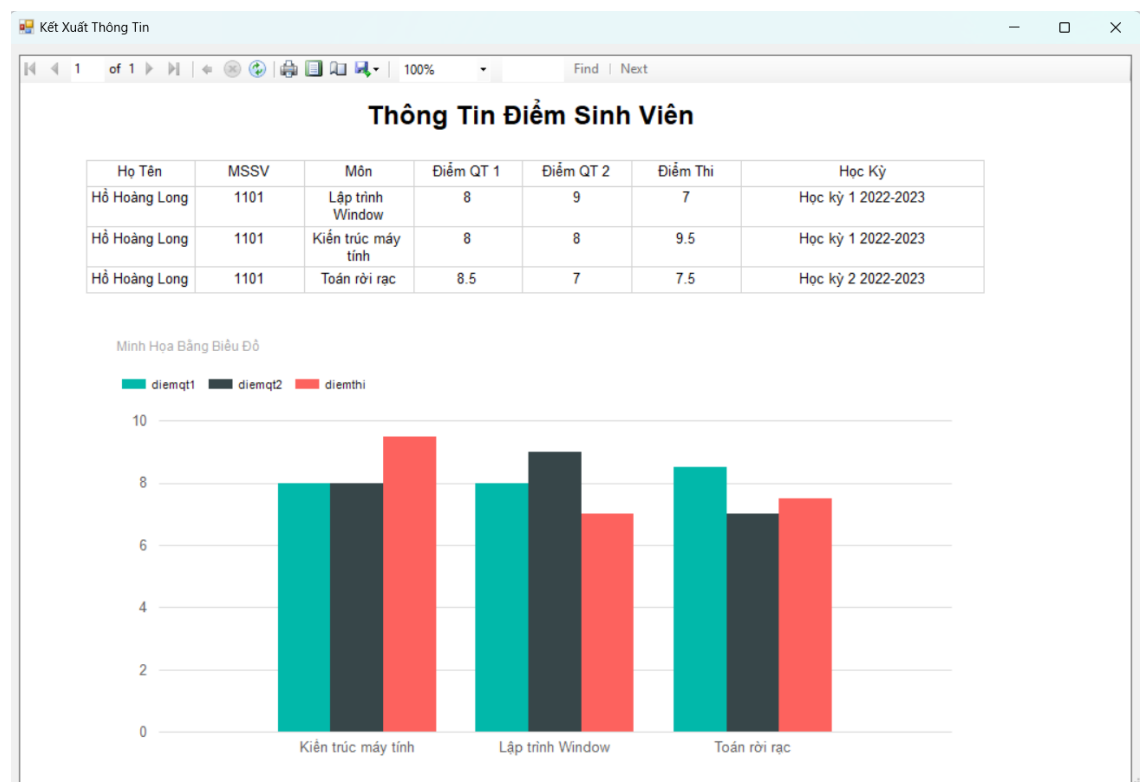
MaMonHoc	TenMonHoc	DiemQuaTrinh1	DiemQuaTrinh2	DiemThi
5001	Lập trình Window	8	9	7
5002	Kiến trúc máy tính	8	8	9.5
5003	Toán rời rạc	8.5	7	7.5

Xuất báo cáo **Thoát**

Hình 4.2 Giao diện sinh viên

Màn hình: xuất báo cáo.

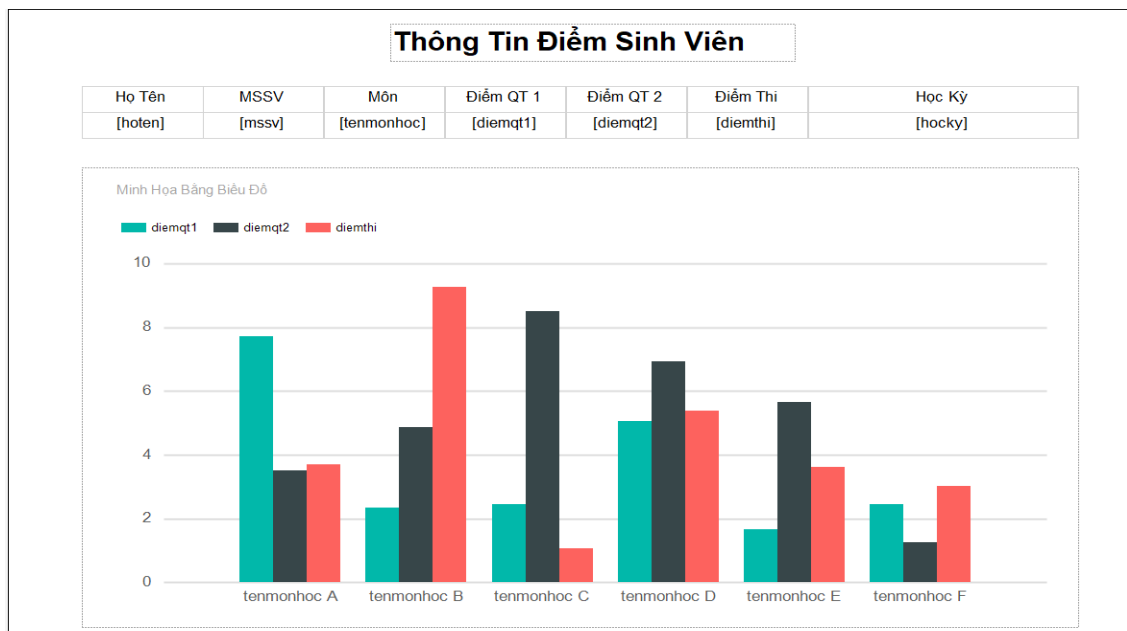
Mô tả: hiển thị báo cáo cho sinh viên bao gồm điểm các môn học.



Hình 4.3 Giao diện chứa báo cáo

Màn hình: biểu mẫu báo cáo.

Mô tả: khi sinh viên xuất báo cáo sẽ hiển thị các thông tin sinh viên, điểm số, môn học, học kỳ đã học và biểu đồ minh họa cho điểm số các môn.



Hình 4.4 Biểu mẫu báo cáo

Màn hình: nhập, cập nhật, xem danh sách điểm sinh viên.

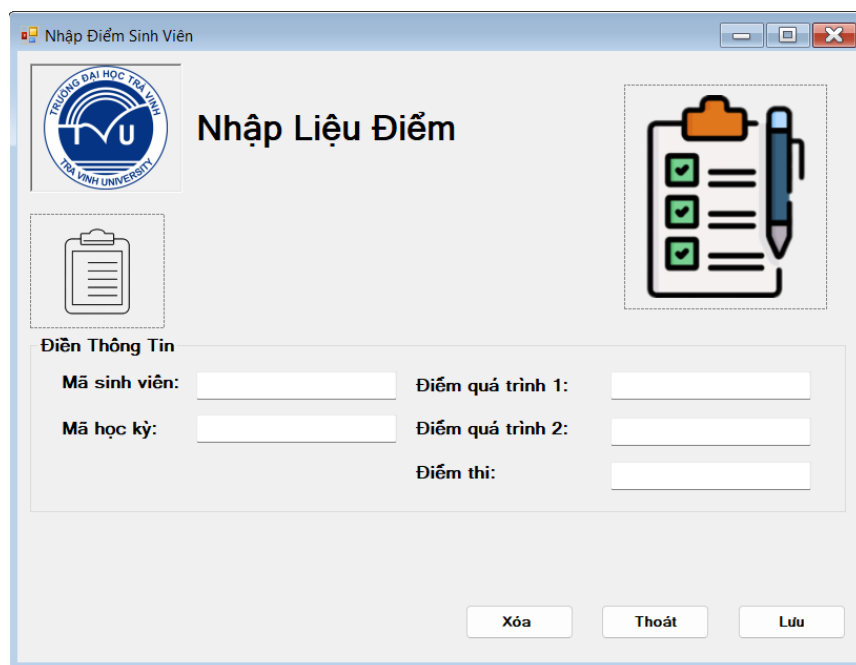
Mô tả: giảng viên đăng nhập vào hệ thống có thể thêm mới điểm cho sinh viên, cập nhật điểm, xóa điểm, xem danh sách các sinh viên đã nhập điểm.



Hình 4.5 Giao diện Giảng viên

Màn hình: nhập liệu điểm cho sinh viên.

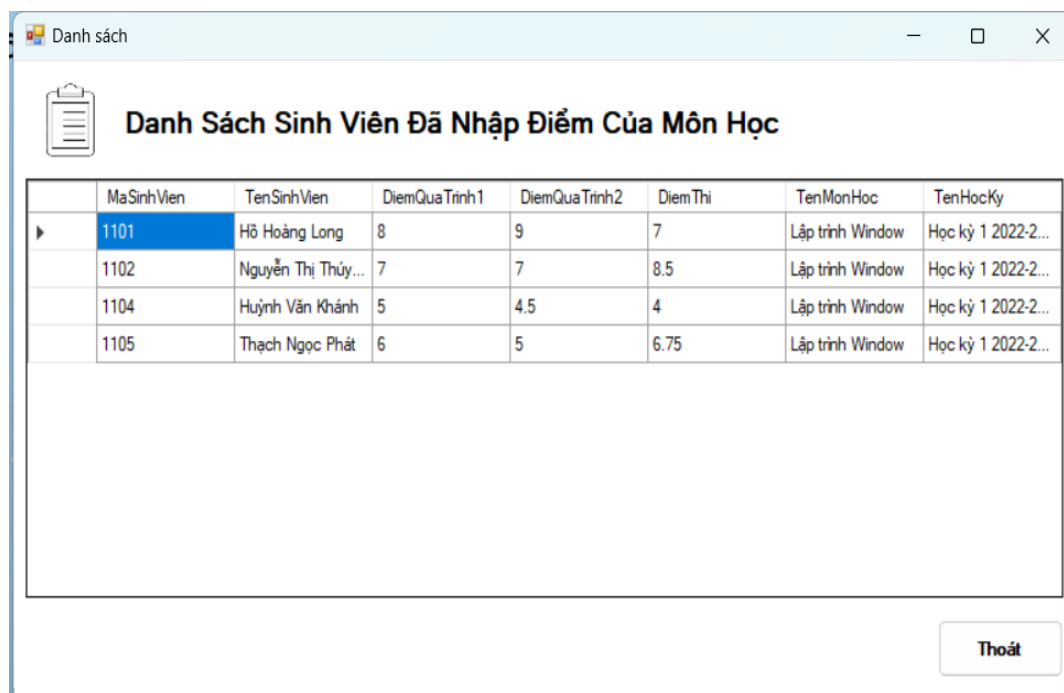
Mô tả: giảng viên nhập điểm mới cho sinh viên, nếu sinh viên đã có điểm xác nhận để cập nhật.



Hình 4.6 Giao diện thao tác với điểm

Màn hình: hiển thị danh sách các sinh viên do giảng viên nhập.

Mô tả: hiển thị danh sách các sinh viên đã được giảng viên nhập điểm ứng với môn học và giảng viên đó.



	MaSinhVien	TenSinhVien	DiemQuaTrinh1	DiemQuaTrinh2	DiemThi	TenMonHoc	TenHocKy
▶	1101	Hồ Hoàng Long	8	9	7	Lập trình Window	Học kỳ 1 2022-2...
	1102	Nguyễn Thị Thúy...	7	7	8.5	Lập trình Window	Học kỳ 1 2022-2...
	1104	Huỳnh Văn Khánh	5	4.5	4	Lập trình Window	Học kỳ 1 2022-2...
	1105	Thạch Ngọc Phát	6	5	6.75	Lập trình Window	Học kỳ 1 2022-2...

Hình 4.7 Giao diện danh sách sinh viên

Màn hình: thao tác cập nhật tài khoản và các thao tác với cơ sở dữ liệu.

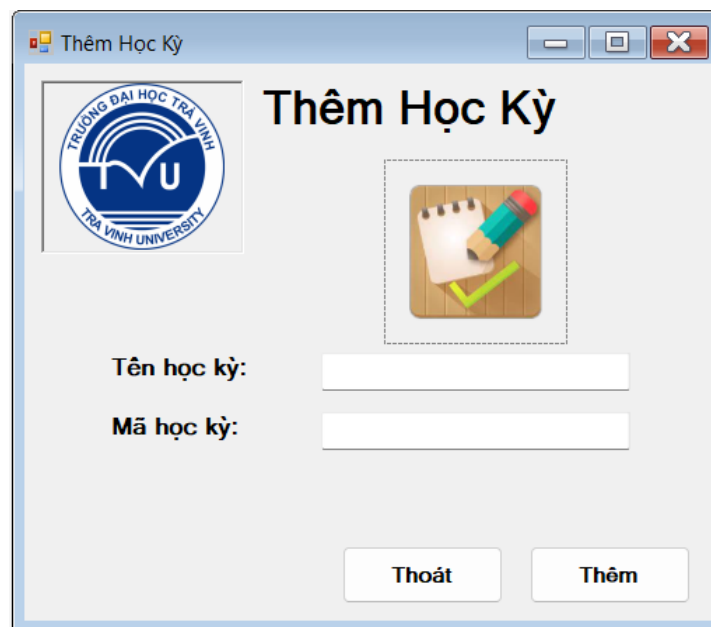
Mô tả: quản trị viên đăng nhập vào hệ thống có thể thực hiện quản lý các tài khoản người dùng, các thao tác với cơ sở dữ liệu giáo dục.



Hình 4.8 Giao diện quản trị viên

Màn hình: thêm học kỳ mới vào chương trình đào tạo.

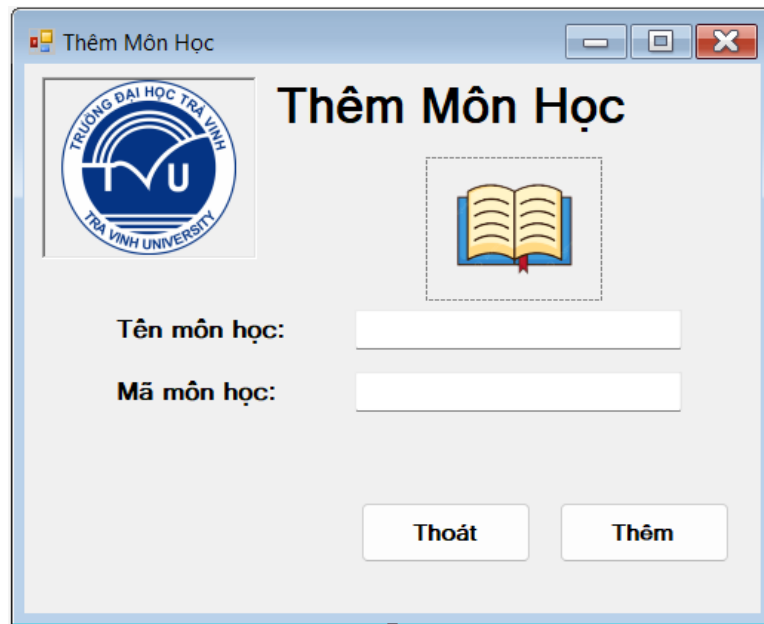
Mô tả: quản trị viên thêm học kỳ mới cho các năm học trong các kỳ kế tiếp.



Hình 4.9 Giao diện thêm mới học kỳ

Màn hình: thêm học môn học mới vào chương trình đào tạo.

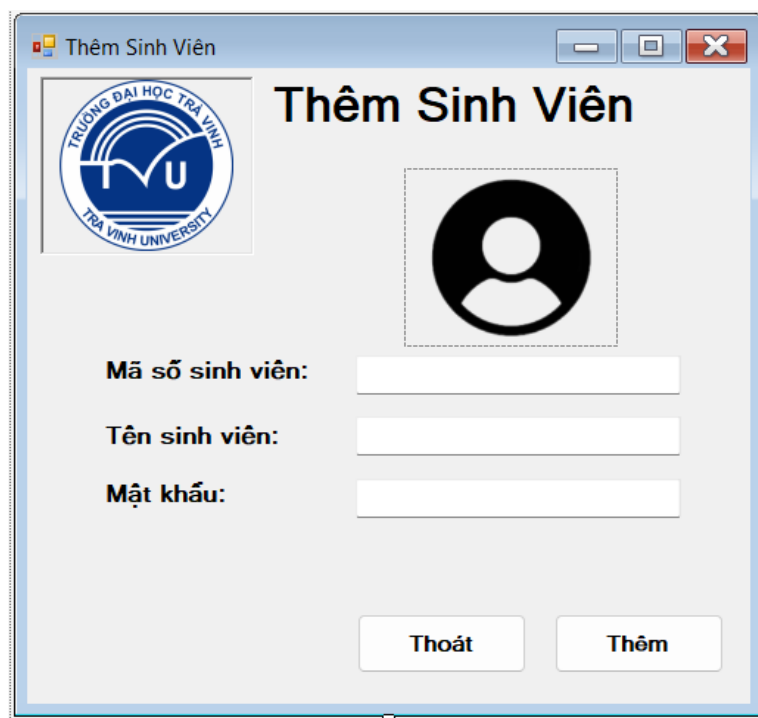
Mô tả: quản trị viên thêm môn học mới cho các ngành đào tạo mới.



Hình 4.10 Giao diện thêm môn học mới

Màn hình: cấp tài khoản sinh viên.

Mô tả: quản trị viên cấp tài khoản cho sinh viên mới nhập học tại trường.



Hình 4.11 Giao diện thêm tài khoản sinh viên

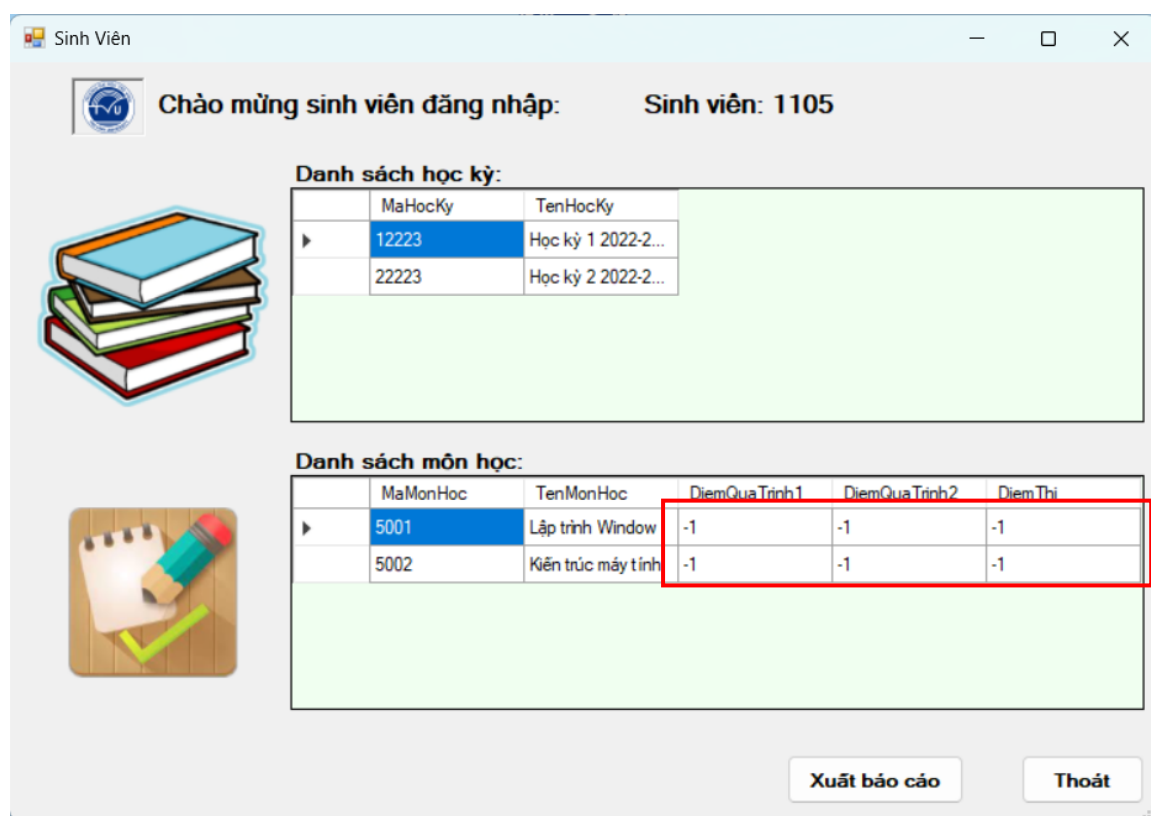
Màn hình: cấp tài khoản giảng viên.

Mô tả: quản trị viên thêm tài khoản cho giảng viên mới thực hiện công tác giảng dạy tại trường.



Hình 4.12 Giao diện thêm tài khoản Giảng viên

Thao tác xóa sinh viên: sinh viên sẽ điểm số của sinh viên của tất cả môn học mà sinh viên hiện đang có sẽ được chuyển thành -1. Nếu có thay đổi gì thì giảng viên sẽ là người cập nhật lại điểm cho sinh viên.



	MaHocKy	TenHocKy
▶	12223	Học kỳ 1 2022-2...
	22223	Học kỳ 2 2022-2...

	MaMonHoc	TenMonHoc	DiemQuaTrinh1	DiemQuaTrinh2	DiemThi
▶	5001	Lập trình Window	-1	-1	-1
	5002	Kiến trúc máy tính	-1	-1	-1

Hình 4.13 Thao tác xóa sinh viên

Thao tác xóa giảng viên: giảng viên sẽ bị đưa ra khỏi hệ thống, tuy nhiên các điểm số của sinh viên đối với môn học của giảng viên đó vẫn còn để phục vụ cho quá trình học tập. Giảng viên không thể tiếp tục đăng nhập hệ thống để thực hiện các thao tác cập nhật điểm điều này nhằm hạn chế các rủi ro gian lận.

	MaSinhVien	MaGiangVien	MaMonHoc	MaHocKy	DiemQuaTrinh1	DiemQuaTrinh2	DiemThi
1	1101	2001	5001	12223	8	9	7
2	1101	2002	5002	12223	8	8	9.5
3	1101	-1	5003	22223	8.5	7	7.5
4	1102	2001	5001	12223	7	7	8.5
5	1104	2001	5001	12223	5	4.5	4
6	1105	2001	5001	12223	-1	-1	-1
7	1105	2002	5002	22223	-1	-1	-1

Hình 4.14 Xóa giảng viên

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

Trước xu hướng công nghệ phát triển ngày càng nhanh và số lượng sinh viên ngày càng lớn, việc tự động hóa các thao tác quản lý góp phần rất lớn trong các công tác xây dựng và phát triển nền giáo dục hiện đại. Giúp nhà trường giải quyết các công việc một cách nhanh chóng, chính xác và mang lại hiệu quả cao.

Với đề tài: xây dựng hệ thống quản lý điểm thi cho sinh viên một phần nào đó đã nắm rõ các vấn đề cơ bản trong việc quản lý điểm số đặc biệt tiếp cận các khó khăn trong các hệ thống cũ và khắc phục bằng cách đưa vào hệ thống những chức năng thuận tiện hơn trong quá trình thao tác, xây dựng giao diện thân thiện, các chức năng, dữ liệu hiển thị rõ ràng, song song đó trong tương lai nếu có nhiều thời gian hơn để phát triển thì hệ thống sẽ rất có ích trong các công tác quản lý đa nền tảng giáo dục, ngày càng được mở rộng và áp dụng cho nhiều nơi.

5.2 Hướng phát triển

Dựa vào thực tiễn trong quá trình sử dụng hệ thống để có những bước tiến tới ưu hơn trong tương lai, tôi đưa ra một số hướng phát triển như sau:

- Xây dựng quy trình quản lý dữ liệu dễ dàng mở rộng, phù hợp cho các cơ sở giáo dục khác nhau.
- Hệ thống lưu trữ được nhiều các thông tin của sinh viên hơn không giới hạn ở phạm vi điểm số.
- Phát triển tính năng nhập liệu với file Excel để tăng tốc độ xử lý.
- Xây dựng ứng dụng đa nền tảng như web, mobile,...
- Xây dựng chức năng cho phép sinh viên đánh giá hoặc phản hồi nếu gặp sự cố trong quá trình thao tác với hệ thống.
- Sử dụng các công nghệ mã hóa các thông tin quan trọng, áp dụng cơ chế xác thực 2 lớp.

DANH MỤC TÀI LIỆU THAM KHẢO

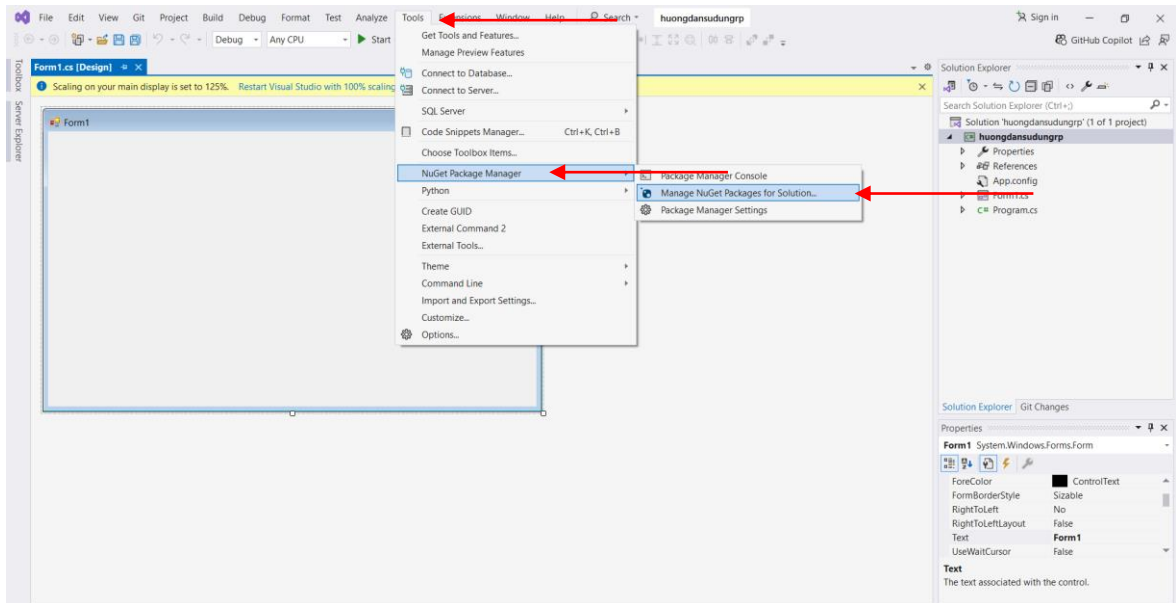
- [1] “plctech.com.vn”, 2014. [Trực tuyến]. Available: <http://plctech.com.vn/winform-la-gi/>. [Đã truy cập: 18/12/2024].
- [2] “bytesoft.vn”, 2021. [Trực tuyến]. Available: <https://bytesoft.vn/lap-trinh-c-va-nhung-dieu-co-the-ban-chua-biet/>. [Đã truy cập: 18/12/2024].
- [3] “bizflycloud.vn”, 2022. [Trực tuyến]. Available: <https://bizflycloud.vn/tin-tuc/winform-la-gi-20220627113605859.htm/>. [Đã truy cập: 18/12/2024].
- [4] “sqlservertutorial.net”, 2022. [Trực tuyến]. Available: <https://www.sqlservertutorial.net/getting-started/what-is-sql-server/>. [Đã truy cập: 19/12/2024].
- [5] Microsoft Support, 2024. [Trực tuyến]. Available: <https://support.microsoft.com/vi-vn/topic/kiến-thức-cơ%20bản-về-thiết-kế-csdl>. [Đã truy cập: 19/12/2024].

PHỤ LỤC

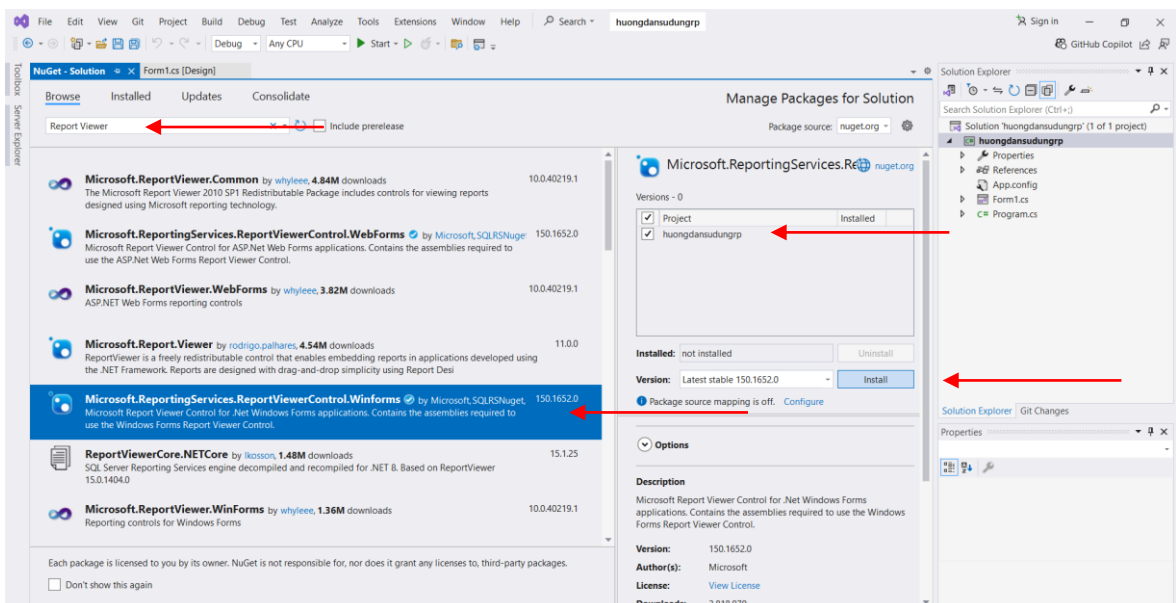
Hướng dẫn sử dụng Report Viewer trong Windows Forms để tạo biểu mẫu báo cáo và chèn biểu đồ minh họa.

Trong dự án chúng ta thực hiện như sau:

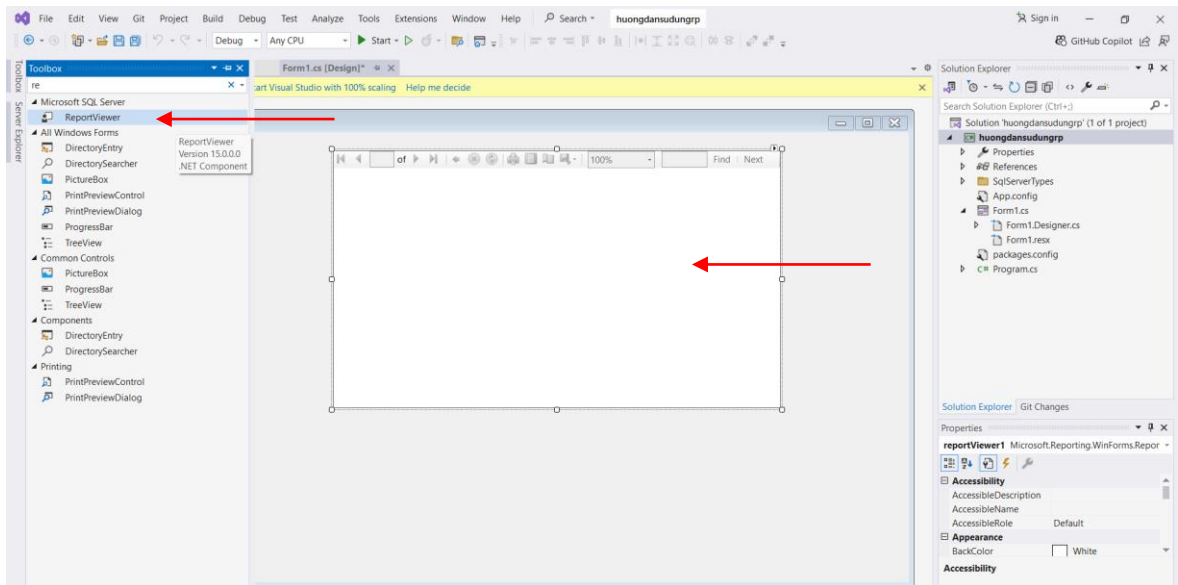
Bước 1: vào Tools chọn mục NuGet Package Manager sau đó chọn Manage NuGet Packages For Solution..



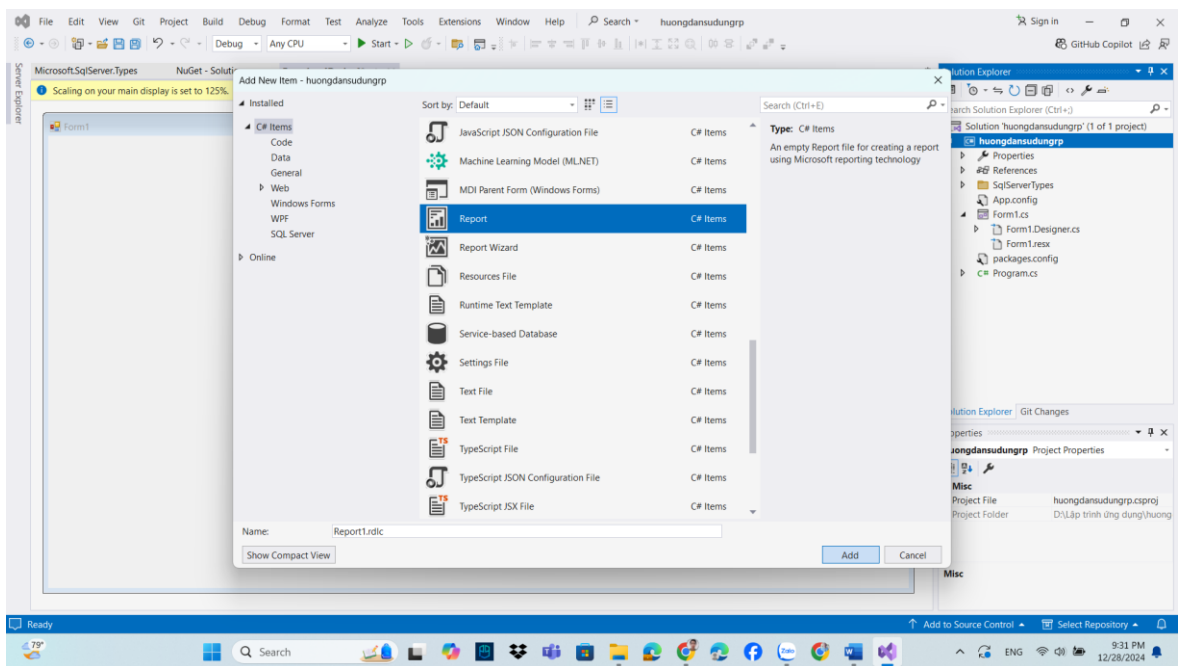
Bước 2: cửa sổ hiện ra chọn Browse tìm kiếm từ khóa “Report Viewer”, chọn gói cài đặt dành cho Winforms, tích vào dự án đang tạo và Install.



Bước 3: sau khi cài đặt thành công quay trở lại form vào Toolbox và kéo thành phần ReportViewer ra giao diện để sử dụng.

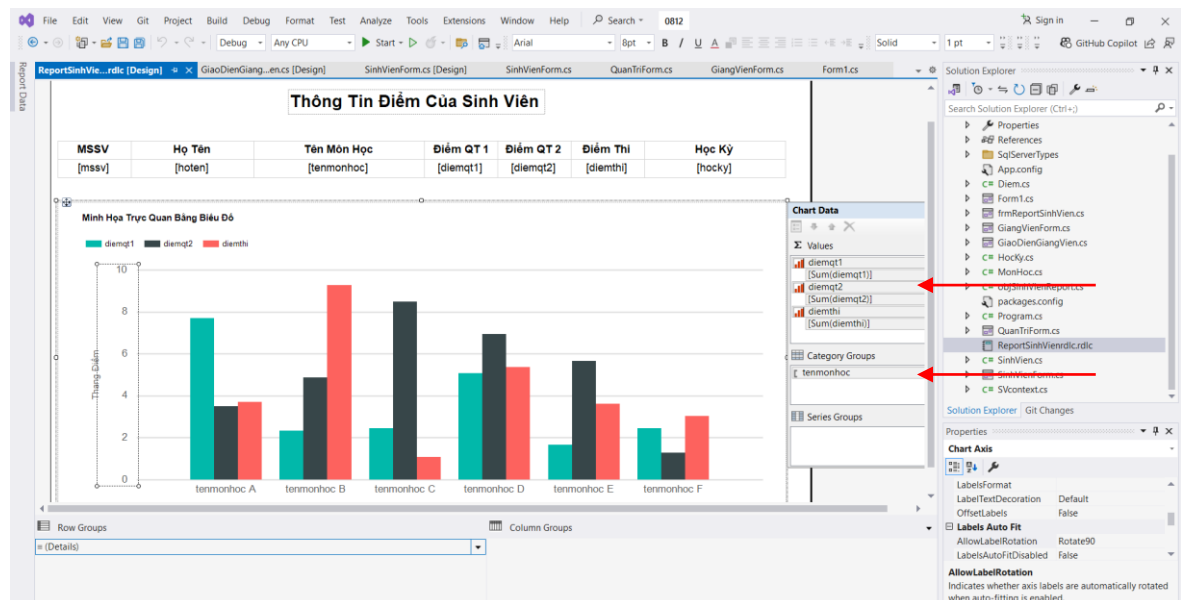


Bước 4: sau khi có giao diện ReportViewer chúng ta sẽ thêm item Report vào dự án bằng cách ấn chuột phải vào dự án chọn add New Item, chọn mục Report và đặt tên.



Bước 5: tạo Dataset sau đó chọn các thông tin cần hiển thị chèn vào bảng, có thể thêm các định dạng nếu muốn.

Bước 6: để chèn biểu đồ ấn chuột phải Insert chọn Chart sau đó chèn biểu đồ mình muốn vào, ví dụ biểu đồ Clustered Column, sau đó thêm các chỉ số cho cột, nhãn cột và các định dạng khác nếu cần.



Bước 7: quay trở lại form Report Viewer tiến hành điều chỉnh code để truyền vào Report cho biểu mẫu hiển thị khi chạy chương trình, ví dụ như sau:

```
private void frmReportSinhVien_Load(object sender, EventArgs e)
{
    SVcontext svcontext = new SVcontext();

    List<Diem> listSV = svcontext.Diems
        .Where(d => d.MaSinhVien == mssvSinhVienDangNhap)
        .ToList();
    List<objSinhVienReport> listReport = new List<objSinhVienReport>();

    foreach (Diem sv in listSV)
    {
        if (sv != null && sv.SinhVien != null) // Kiểm tra null
        {
            objSinhVienReport temp = new objSinhVienReport
            {
                mssv = sv.MaSinhVien,
                hoten = sv.SinhVien.TenSinhVien,
                tenmonhoc = sv.MonHoc.TenMonHoc,
                diemqt1 = (float)sv.DiemQuaTrinh1,
                diemqt2 = (float)sv.DiemQuaTrinh2,
                diemthi = (float)sv.DiemThi,
                hocky = sv.HocKy.TenHocKy
            };

            listReport.Add(temp);
        }
    }
    reportViewer1.LocalReport.ReportPath = "ReportSinhVienrdlc.rdlc";
    var source = new ReportDataSource("DataSetSinhVienRP", listReport);

    reportViewer1.LocalReport.DataSources.Clear();
    reportViewer1.LocalReport.DataSources.Add(source);
    this.reportViewer1.RefreshReport();
}
```