**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG**

**TPHCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**BÁO CÁO ĐỒ ÁN MÔN HỌC**

**CÔNG NGHỆ .NET**

**Lớp: 10\_ĐH\_CNPM3**

**ĐỀ TÀI: PHẦN MỀM QUẢN LÍ KẾT QUẢ HỌC TẬP**

Giảng viên: **ThS. Trần Văn Định + CN. Huỳnh Luân**

Nhóm sinh viên thực hiện:

**Họ tên Mã số sinh viên**

1. Trần Minh Hợp 1050080180
2. Nguyễn Khánh Duy 1050080173

***TP. Hồ Chí Minh, tháng 11 năm 2024***

**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG**

**TPHCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**BÁO CÁO ĐỒ ÁN MÔN HỌC**

**CÔNG NGHỆ .NET**

**ĐỀ TÀI: PHẦN MỀM QUẢN LÍ KẾT QUẢ HỌC TẬP**

**Lớp: 10\_ĐH\_CNPM3**

Giảng viên: **ThS. Trần Văn Định**

Nhóm sinh viên thực hiện:

**Họ tên Mã số sinh viên**

1. Trần Minh Hợp 1050080180
2. Nguyễn Khánh Duy 1050080173

***TP. Hồ Chí Minh, tháng 11 năm 2024***

# LỜI CẢM ƠN

Trong khóa luận này, chúng em muốn diễn đạt lòng biết ơn sâu sắc đến những người đã đóng góp vào quá trình nghiên cứu và viết bài luận này về việc phát triển một phần mềm quản lý kết quả học tập.

Trước hết, chúng em xin gửi lời cảm ơn chân thành đến Thạc sĩ Trần Văn Định, người đã dành thời gian và kiến thức để hướng dẫn và hỗ trợ chúng em trong quá trình nghiên cứu và viết bài luận này. Sự hỗ trợ và sự chỉ dẫn của Thầy là điều không thể đánh giá giá trị đối với sự thành công của dự án này.

Chúng em cũng muốn bày tỏ lòng biết ơn sâu sắc đến các bạn bè, đồng nghiệp và gia đình đã luôn ở bên cạnh, động viên và hỗ trợ chúng em trong suốt quá trình này. Sự ủng hộ và động viên từ phía quý vị đã mang lại động lực lớn để chúng tôi vượt qua những thử thách và hoàn thành khóa luận này.

Nhân dịp này, chúng em cũng xin cảm ơn  Khoa Công nghệ thông tin Trường Đại học Tài nguyên và Môi trường TPHCM, lãnh đạo và các anh chị đang công tác tại trường đã tạo điều kiện và thời gian cho chúng em trong suốt quá trình nghiên cứu.

Cuối cùng nhưng không kém phần quan trọng, chúng em muốn gửi lời cảm ơn đến 2 thành viên bao gồm: Trần Minh Hợp, Nguyễn Khánh Duy đã tham gia và đóng góp ý kiến, thông tin và ý tưởng trong quá trình nghiên cứu của chúng em. Sự đóng góp của nhau đã giúp chúng ta hiểu sâu hơn về chủ đề và tạo ra một khóa luận đầy đủ và mang tính thực tiễn.

Trong quá trình hoàn thành đồ án, chúng em không tránh khỏi những sơ sót, chúng em xin tiếp nhận những ý kiến đóng góp cũng như những của thầy để tránh những sai lầm cho sau này.

Những lời cảm ơn này không thể nào diễn tả hết lòng biết ơn của chúng tôi. Chúng em hy vọng rằng công trình này sẽ góp phần vào việc nâng cao kiến thức và sự hiểu biết trong lĩnh vực này.

# LỜI CAM ĐOAN

Chúng em, đến từ lớp 10ĐHCNPM3 của Trường Đại học Tài Nguyên và Môi Trường TPHCM, xin cam đoan một cách trung thực và chân thành về công trình nghiên cứu này, được thực hiện trong khuôn khổ bài tiểu luận với tựa đề "Phần mềm Quản lí kết quả học tập".

Chúng em cam đoan rằng mọi thông tin và ý kiến được trình bày trong bài luận này là kết quả của công sức nghiên cứu của chính mình dưới sự hướng dẫn tận tình của thầy Th.S. Trần Văn Định. Mọi nguồn tài liệu và dữ liệu được trích dẫn đều được đánh giá và kiểm tra tính chính xác trước khi được sử dụng.

Chúng em cam kết rằng tất cả các ý kiến và quan điểm được biểu đạt trong bài luận là của riêng chúng em và không bị ảnh hưởng bởi bất kỳ lợi ích hay áp lực nào từ bên ngoài.

Ngoài ra, chúng em cam đoan rằng không có phần nội dung nào trong bài luận này bị sao chép từ các nguồn khác mà không được ghi rõ nguồn gốc. Mọi trích dẫn và tham khảo đều được tuân thủ theo các nguyên tắc đạo đức và pháp lý.

Chúng em hiểu rằng bất kỳ hành động sao chép, đạo văn hoặc vi phạm đạo đức học thuật đều là hành vi không chấp nhận trong cộng đồng học thuật và có thể chịu mức phạt nặng, từ mất điểm cho đến bị loại khỏi chương trình đào tạo.

Chúng em cam kết đã tuân thủ mọi nguyên tắc và quy định về nghiên cứu và viết luận của Trường Đại học Tài Nguyên và Môi Trường TPHCM. Chúng em chấp nhận trách nhiệm đầy đủ về nội dung và chất lượng của bài tiểu luận này.

Xác nhận và cam đoan.

# BẢNG PHÂN CHIA NHIỆM VỤ

|  |  |  |
| --- | --- | --- |
| **Họ và tên** | **Các công việc tham gia** | **Tỉ lệ (%) đóng góp trong bài** |
| **Trần Minh Hợp** | Các thành viên trong nhóm đều tham gia đầy đủ các công việc được giao, mỗi công việc nhỏ trong bài đều được làm và duyệt bởi 2 thành viên. | 50% |
| **Nguyễn Khánh Duy** | 50% |

# NHẬN XÉT

**(Của giảng viên giảng dạy)**

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………….., ngày….tháng 05 năm 2024

CÁN BỘ GIẢNG DẠY

*(ký tên)*

# LỜI MỞ ĐẦU

Trong bối cảnh giáo dục ngày càng phát triển mạnh mẽ, việc quản lý kết quả học tập của sinh viên trở thành một nhiệm vụ quan trọng và cần thiết. Sự gia tăng về số lượng sinh viên cùng với sự đa dạng trong các chương trình học tập đã đặt ra những thách thức không nhỏ cho các cơ sở giáo dục. Phần mềm quản lý kết quả học tập không chỉ giúp nhà trường theo dõi sự tiến bộ của sinh viên mà còn hỗ trợ sinh viên trong việc đánh giá hiệu quả học tập của bản thân. Thông qua việc cung cấp thông tin chi tiết về điểm số, phản hồi từ giảng viên và các chỉ số khác, phần mềm này tạo điều kiện thuận lợi cho việc phát triển kế hoạch học tập cá nhân.

Với sự phát triển của công nghệ thông tin, việc ứng dụng phần mềm vào quản lý học tập đã trở thành một xu hướng tất yếu. Các giải pháp phần mềm hiện đại không chỉ giúp giảm thiểu thời gian và công sức trong việc quản lý dữ liệu mà còn nâng cao tính chính xác và minh bạch trong quy trình đánh giá. Điều này không chỉ mang lại nhiều lợi ích cho giáo viên, giúp họ theo dõi và phân tích sự tiến bộ của từng sinh viên một cách hiệu quả, mà còn cho sinh viên, khi họ có thể dễ dàng tiếp cận và hiểu rõ kết quả học tập của mình.

Đề tài "Phần mềm quản lý kết quả học tập của sinh viên" sẽ đi sâu vào việc xây dựng một hệ thống quản lý hiệu quả, đáp ứng đầy đủ nhu cầu sử dụng trong môi trường giáo dục hiện đại. Chúng em sẽ nghiên cứu, thiết kế và phát triển một giải pháp phần mềm không chỉ đơn thuần là công cụ lưu trữ dữ liệu mà còn là nền tảng hỗ trợ phân tích kết quả học tập, giúp người dùng đưa ra những quyết định đúng đắn trong quá trình học tập và giảng dạy. Hệ thống này sẽ tích hợp các tính năng như theo dõi điểm số, tạo báo cáo, phân tích xu hướng học tập, và cung cấp phản hồi tức thì, từ đó hỗ trợ sinh viên trong việc cải thiện kết quả học tập.

Ngoài ra, chúng em cũng sẽ tập trung vào việc đảm bảo tính thân thiện với người dùng, nhằm tạo ra một trải nghiệm sử dụng dễ dàng và hiệu quả. Việc phát triển phần mềm này không chỉ mang tính cấp thiết trong việc cải thiện quy trình quản lý mà còn đóng góp vào việc nâng cao chất lượng giáo dục nói chung.

Bằng việc sử dụng các phương pháp nghiên cứu như thu thập, tổng hợp kiến thức, số liệu từ nhiều nguồn, chúng em phân chia cấu trúc của đồ án này như sau

Chương 1: Tóm tắt lý thuyết: Giải thích lý do chọn đề tài, khái quát ngành công nghiệp rượu và thị trường bán rượu trực tuyến, cùng với các cơ sở lý thuyết liên quan đến phát triển phần mềm.

Chương 2: Đặc tả dự án: Phân tích yêu cầu chức năng và phi chức năng, thiết kế sơ đồ cẩn thiết, xác định yêu cầu về giao diện người dùng.

Chương 3: Cài đặt: Lựa chọn công nghệ phù hợp, phát triển và triển khai website.

Chương 4: Kết quả đạt được: Thực hiện demo sản phẩm, đánh giá hiệu suất và tính ứng dụng của website.

Chương 5: Kết luận: Tóm tắt kết quả nghiên cứu và đề xuất hướng phát triển tiếp theo.

# MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc16909)

[LỜI CAM ĐOAN 3](#_Toc16384)

[BẢNG PHÂN CHIA NHIỆM VỤ 4](#_Toc6933)

[NHẬN XÉT 5](#_Toc15162)

[LỜI MỞ ĐẦU 6](#_Toc30221)

[MỤC LỤC 8](#_Toc9602)

[CHƯƠNG I. ĐẶC TẢ ỨNG DỤNG 9](#_Toc8776)

[I.1. Đặc tả yêu cầu chức năng 9](#_Toc18870)

[I.2. Đặc tả yêu cầu phi chức năng 24](#_Toc31906)

[I.3. Phạm vi project 27](#_Toc8426)

[CHƯƠNG II. THIẾT KẾ DATABASE 29](#_Toc4826)

[II.1. Table design 29](#_Toc7901)

[II.2. Database diagram 35](#_Toc6859)

[CHƯƠNG III. THIẾT KẾ KIẾN TRÚC 37](#_Toc14025)

[III.1. Kiến trúc ứng dụng 37](#_Toc9322)

[III.2. Kiến trúc tổng thể 39](#_Toc1003)

[III.3. Kiến trúc từng thành phần 41](#_Toc25773)

[III.4. Vai trò của từng thành phần trong kiến trúc 43](#_Toc6982)

[CHƯƠNG IV. THIẾT KẾ GIAO DIỆN 46](#_Toc10448)

[CHƯƠNG V. TỔ CHỨC C# PROJECT 47](#_Toc30816)

[V.1. Mô hình tổng thể 47](#_Toc15504)

[V.2. Cách tổ chức các thành phần trong project 48](#_Toc17682)

[V.3. Vai trò của từng thành phần trong project 49](#_Toc9034)

[V.4. Class diagram 51](#_Toc31900)

[CHƯƠNG VI. KẾT QUẢ THỰC HIỆN 51](#_Toc17195)

[VI.1. Kết quả thực hiện chức năng 51](#_Toc11422)

[VI.2. Kết quả thực hiện chức năng 51](#_Toc23254)

[VI.3. Kết quả thực hiện chức năng 51](#_Toc27307)

[VI.4. Kết quả thực hiện chức năng 51](#_Toc7457)

[VI.5. Kết quả thực hiện chức năng 51](#_Toc2955)

[VI.6. Kết quả thực hiện chức năng 51](#_Toc27614)

[VI.7. Kết quả thực hiện chức năng 51](#_Toc20834)

[VI.8. Kết quả thực hiện chức năng 51](#_Toc12747)

[VI.9. Kết quả thực hiện chức năng 51](#_Toc12386)

[VI.10. Kết quả thực hiện chức năng 51](#_Toc7450)

[CHƯƠNG VII. KẾT LUẬN 51](#_Toc1451)

[VII.1. Kết quả đạt được 51](#_Toc18232)

[VII.2. Hạn chế 51](#_Toc14981)

[CHƯƠNG VIII. TÀI LIỆU THAM KHẢO 51](#_Toc31667)

## CHƯƠNG I. ĐẶC TẢ ỨNG DỤNG

### I.1. Đặc tả yêu cầu chức năng

#### I.1.1. Chức năng Đăng nhập

Quy trình đăng nhập tài khoản cho sinh viên/giảng viên/admin hệ thống:

B1. Sinh viên/giảng viên/admin hệ thống mở trình duyệt và truy cập vào trang web của hệ thống. Nhấp vào nút “Đăng nhập” trên giao diện chính.

B2. Nhập thông tin đăng nhập vào các ô có sẵn:

* Email: Nhập email mà nhà trường đã cung cấp.
* Mật khẩu: Nhập mật khẩu đã đăng ký.

B3. Nếu muốn trang web lưu lại tài khoản, sinh viên/giảng viên/admin hệ thống nhấn vào ô Remember me ở dưới ô nhập mật khẩu, sau đó nhấn vào nút “Đăng nhập” trên màn hình.

B4. Kiểm tra thông tin:

Hệ thống sẽ kiểm tra thông tin đăng nhập có tồn tại và chính xác không.

* Nếu thông tin không chính xác (email không đúng hoặc mật khẩu không đúng), hệ thống sẽ hiển thị thông báo lỗi và yêu cầu sinh viên/giảng viên/admin hệ thống nhập lại thông tin đúng.
* Nếu thông tin chính xác, hệ thống tiếp tục.

B5. Đăng nhập thành công:

Hệ thống cho phép sinh viên/giảng viên/admin hệ thống truy cập vào tài khoản sau khi nhấn vào nút “Đăng nhập” trên màn hình. Sinh viên/giảng viên/admin hệ thống được chuyển đến trang chính của website.

Tuy nhiên, nếu có lỗi trong quá trình đăng ký (ví dụ: lỗi kết nối cơ sở dữ liệu, lỗi hệ thống), hệ thống sẽ hiển thị thông báo lỗi và yêu cầu sinh viên thử lại sau. Nếu sinh viên/giảng viên/admin hệ thống không nhập đủ các thông tin bắt buộc, hệ thống sẽ hiển thị thông tin yêu cầu sinh viên/giảng viên/admin hệ thống phải nhập đầy đủ thông tin.

#### I.1.2. Chức năng Create

|  |  |  |
| --- | --- | --- |
| **Quản trị viên** | **Giảng viên** | **Sinh viên** |
| Thông tin có thể thêm:   * Môn học: Thêm tên môn học, số tín chỉ, giảng viên phụ trách, mô tả môn học. * Giảng viên: Thêm thông tin cá nhân của giảng viên, bao gồm tên, email, số điện thoại, chuyên môn. * Sinh viên: Thêm thông tin cá nhân của sinh viên, bao gồm tên, mã sinh viên, email, số điện thoại. * Hoạt động ngoại khóa: Thêm tên hoạt động, thời gian, địa điểm, mô tả hoạt động. * Điểm trung bình * Điểm rèn luyện | Thông tin có thể thêm:   * Điểm cho sinh viên: Thêm điểm số cho từng sinh viên trong môn học mà họ phụ trách (bao gồm điểm giữa kỳ, điểm cuối kỳ, và các bài kiểm tra khác). * Điểm trung bình * Điểm rèn luyện | Thông tin có thể thêm:   * Đăng ký hoạt động ngoại khóa: Thêm thông tin đăng ký tham gia các hoạt động ngoại khóa do nhà trường tổ chức (bao gồm tên hoạt động, lý do tham gia). * Ghi chú cá nhân: Thêm ghi chú cá nhân liên quan đến học tập hoặc các hoạt động khác (nếu chức năng này được cung cấp trong hệ thống). |

##### Q**uy trình thêm thông tin của quản trị viên**

B1. Quản trị viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện cần thêm thông tin như “Môn học”, “Giảng viên”, “Sinh viên”, “Hoạt động ngoại khóa”.

B3. Từ giao diện, chọn “Thêm môn học”, “Thêm giảng viên”, “Thêm sinh viên” hoặc “Thêm hoạt động ngoại khóa” tùy thuộc vào thông tin cần thêm.

B4. Nhập thông tin cần thiết vào các ô có sẵn:

1. Về sinh viên

* Số thứ tự
* Mã sinh viên
* Tên sinh viên
* Ngày sinh
* Giới tính
* Email
* SĐT

1. Về giảng viên

* Số thứ tự
* Mã giảng viên
* Tên giảng viên
* Chuyên môn
* Email
* SĐT

1. Về môn học

* Số thứ tự
* Mã môn học
* Tên môn học
* Giảng viên phụ trách
* Số tín chỉ
* Mô tả

1. Về hoạt động ngoại khóa

* Số thứ tự
* Mã hoạt động
* Tên hoạt động
* Địa điểm
* Mô tả hoạt động
* Thời gian bắt đầu
* Thời gian kết thúc

1. Về điểm trung bình

* Số thứ tự
* Mã điểm trung bình
* Mã sinh viên
* Tên môn
* Điểm môn
* Điểm trung bình

1. Về điểm rèn luyện

* Số thứ tự
* Mã điểm rèn luyện
* Mã hoạt động
* Mã sinh viên
* Điểm rèn luyện

1. Xếp loại

* Số thứ tự
* Mã sinh viên
* Điểm trung bình
* Điểm rèn luyện
* Xếp loại

B5. Kiểm tra thông tin: Hệ thống kiểm tra tính hợp lệ của thông tin đã nhập.

* Nếu có lỗi, hệ thống hiển thị thông báo lỗi và yêu cầu điều chỉnh.
* Nếu thông tin hợp lệ, hệ thống tiếp tục.

B6. Gửi yêu cầu thêm thông tin: Nhấn nút “Thêm” để gửi yêu cầu.

B7. Xác nhận thêm thông tin:

Hệ thống hiển thị thông báo xác nhận rằng thông tin đã được thêm thành công. Nếu có lỗi, hệ thống sẽ thông báo lỗi và yêu cầu thử lại.

##### Quy trình thêm thông tin của giảng viên

B1. Giảng viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện cần thêm thông tin “Bảng điểm”.

B3. Từ giao diện, chọn “Thêm”.

B4. Nhập thông tin cần thiết vào các ô có sẵn:

* STT
* Tên sinh viên: Chọn sinh viên từ danh sách.
* Điểm giữa kỳ: Nhập điểm giữa kỳ.
* Điểm cuối kỳ: Nhập điểm cuối kỳ.

B5. Kiểm tra thông tin: Hệ thống kiểm tra tính hợp lệ của thông tin đã nhập.

* Nếu có lỗi, hệ thống hiển thị thông báo lỗi và yêu cầu điều chỉnh.
* Nếu thông tin hợp lệ, hệ thống tiếp tục.

B6. Gửi yêu cầu thêm thông tin: Nhấn nút “Thêm” để gửi yêu cầu.

B7. Xác nhận thêm thông tin:

Hệ thống hiển thị thông báo xác nhận rằng thông tin đã được thêm thành công. Nếu có lỗi, hệ thống sẽ thông báo lỗi và yêu cầu thử lại.

##### Quy trình thêm thông tin của sinh viên sinh viên

B1. Sinh viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện đăng ký hoạt động ngoại khóa: Từ giao diện chính, chọn “Hoạt động ngoại khóa”.

B3. Từ giao diện, chọn “Thêm”.

B4. Nhập thông tin cần thiết vào các ô có sẵn:

* Tên hoạt động: Nhập tên hoạt động ngoại khóa.
* Lý do tham gia: Nhập lý do muốn tham gia hoạt động.

B5. Kiểm tra thông tin: Hệ thống kiểm tra tính hợp lệ của thông tin đã nhập.

* Nếu có lỗi, hệ thống hiển thị thông báo lỗi và yêu cầu điều chỉnh.
* Nếu thông tin hợp lệ, hệ thống tiếp tục.

B6. Gửi yêu cầu thêm thông tin: Nhấn nút “Đăng ký” để gửi yêu cầu.

B7. Xác nhận thêm thông tin:

Hệ thống hiển thị thông báo xác nhận rằng thông tin đã được thêm thành công. Nếu có lỗi, hệ thống sẽ thông báo lỗi và yêu cầu thử lại.

#### I.1.3. Chức năng Sửa

|  |  |  |
| --- | --- | --- |
| **Quản trị viên** | **Giảng viên** | **Sinh viên** |
| Thông tin có thể sửa:   * Môn học: Sửa tên môn học, số tín chỉ, giảng viên phụ trách, mô tả môn học. * Giảng viên: Sửa thông tin cá nhân của giảng viên, bao gồm tên, email, số điện thoại, chuyên môn. * Sinh viên: Sửa thông tin cá nhân của sinh viên, bao gồm tên, mã sinh viên, email, số điện thoại. * Hoạt động ngoại khóa: Sửa tên hoạt động, thời gian, địa điểm, mô tả hoạt động. * Điểm trung bình * Điểm rèn luyện * Xếp loại | Thông tin có thể sửa:   * Điểm cho sinh viên: Thêm điểm số cho từng sinh viên trong môn học mà họ phụ trách (bao gồm điểm giữa kỳ, điểm cuối kỳ, và các bài kiểm tra khác). * Điểm trung bình * Điểm rèn luyện * Xếp loại | * Hoạt động ngoại khóa: Sửa thông tin đăng ký tham gia các hoạt động ngoại khóa do nhà trường tổ chức (bao gồm tên hoạt động, lý do tham gia). * Ghi chú cá nhân: Sửa ghi chú cá nhân liên quan đến học tập hoặc các hoạt động khác |

##### Quy trình sửa thông tin của quản trị viên

B1. Quản trị viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện cần sửa thông tin như “Môn học”, “Giảng viên”, “Sinh viên”, “Hoạt động ngoại khóa”.

B3. Từ giao diện, chọn “Sửa”.

B3. Chọn thông tin cần sửa như môn học, giảng viên, sinh viên, hoạt động ngoại khóa.

B4. Chỉnh sửa thông tin cần thiết:

1. Về sinh viên

* Số thứ tự
* Mã sinh viên
* Tên sinh viên
* Ngày sinh
* Giới tính
* Email
* SĐT

1. Về giảng viên

* Số thứ tự
* Mã giảng viên
* Tên giảng viên
* Chuyên môn
* Email
* SĐT

1. Về môn học

* Số thứ tự
* Mã môn học
* Tên môn học
* Giảng viên phụ trách
* Số tín chỉ
* Mô tả

1. Về hoạt động ngoại khóa

* Số thứ tự
* Mã hoạt động
* Tên hoạt động
* Địa điểm
* Mô tả hoạt động
* Thời gian bắt đầu
* Thời gian kết thúc

1. Về điểm trung bình

* Số thứ tự
* Mã điểm trung bình
* Mã sinh viên
* Tên môn
* Điểm môn
* Điểm trung bình

1. Về điểm rèn luyện

* Số thứ tự
* Mã điểm rèn luyện
* Mã hoạt động
* Mã sinh viên
* Điểm rèn luyện

1. Xếp loại

* Số thứ tự
* Mã sinh viên
* Điểm trung bình
* Điểm rèn luyện
* Xếp loại

B5. Kiểm tra thông tin: Hệ thống kiểm tra tính hợp lệ của thông tin đã nhập.

* Nếu có lỗi, hệ thống hiển thị thông báo lỗi và yêu cầu điều chỉnh.
* Nếu thông tin hợp lệ, hệ thống tiếp tục.

B6. Gửi yêu cầu sửa thông tin: Nhấn nút “Lưu” để gửi yêu cầu sửa.

B7. Xác nhận sửa thông tin:

Hệ thống hiển thị thông báo xác nhận rằng thông tin đã được sửa thành công. Nếu có lỗi, hệ thống sẽ thông báo lỗi và yêu cầu thử lại.

##### Quy trình sửa thông tin của sinh viên

B1. Sinh viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện cần sửa thông tin như “Thông tin cá nhân” hoặc “Hoạt động ngoại khóa”.

B3. Từ giao diện, chọn “Sửa”.

B4. Chỉnh sửa thông tin cần thiết:

* Thông tin cá nhân: Nhập lại tên, địa chỉ, số điện thoại.
* Đăng ký hoạt động: Cập nhật thông tin tham gia hoạt động ngoại khóa.

B5. Kiểm tra thông tin: Hệ thống kiểm tra tính hợp lệ của thông tin đã nhập.

B6. Gửi yêu cầu sửa thông tin: Nhấn nút “Lưu” để gửi yêu cầu sửa.

B7. Xác nhận sửa thông tin: Hệ thống hiển thị thông báo xác nhận rằng thông tin đã được sửa thành công.

##### Quy trình sửa thông tin của giảng viên

B1. Giảng viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện cần sửa thông tin như “Quản lý điểm” hoặc “Quản lý sinh viên”.

B3. Từ giao diện, chọn “Sửa”.

B4. Tìm kiếm và chọn sinh viên hoặc điểm số cần chỉnh sửa từ danh sách hiển thị. Chỉnh sửa thông tin cần thiết:

* STT
* Tên sinh viên: Chọn sinh viên từ danh sách.
* Điểm giữa kỳ: Nhập điểm giữa kỳ.
* Điểm cuối kỳ: Nhập điểm cuối kỳ.

B5. Kiểm tra thông tin: Hệ thống kiểm tra tính hợp lệ của thông tin đã nhập.

B6. Gửi yêu cầu sửa thông tin: Nhấn nút “Lưu” để gửi yêu cầu sửa.

B7. Xác nhận sửa thông tin: Hệ thống hiển thị thông báo xác nhận rằng thông tin đã được sửa thành công.

#### I.1.4. Chức năng Xóa

|  |  |
| --- | --- |
| **Quản trị viên** | **Giảng viên** |
| Thông tin có xóa:   * Môn học: Xóa tên môn học, số tín chỉ, giảng viên phụ trách, mô tả môn học. * Giảng viên: Xóa thông tin cá nhân của giảng viên, bao gồm tên, email, số điện thoại, chuyên môn. * Sinh viên: Xóa thông tin cá nhân của sinh viên, bao gồm tên, mã sinh viên, email, số điện thoại. * Hoạt động ngoại khóa: Xóa tên hoạt động, thời gian, địa điểm, mô tả hoạt động. * Điểm trung bình * Điểm rèn luyện * Xếp loại | Thông tin có thể xóa:   * Điểm cho sinh viên: Thêm điểm số cho từng sinh viên trong môn học mà họ phụ trách (bao gồm điểm giữa kỳ, điểm cuối kỳ, và các bài kiểm tra khác). * Điểm trung bình * Điểm rèn luyện * Xếp loại |

##### Quy trình xóa thông tin của quản trị viên

B1. Quản trị viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện cần xóa thông tin như “Môn học”, “Giảng viên”, “Sinh viên”, hoặc “Hoạt động ngoại khóa”.

B3. Từ giao diện, tìm kiếm thông tin cần xóa: Tìm kiếm và chọn thông tin cụ thể (ví dụ: tên môn học, tên giảng viên) từ danh sách hiển thị.

B4. Chọn chức năng xóa: Nhấn vào nút “Xóa” bên cạnh thông tin cần xóa.

B5. Xác nhận xóa: Hệ thống sẽ hiển thị thông báo xác nhận (“Bạn có chắc chắn muốn xóa thông tin này?”).

* Nếu người dùng chọn “Có”, hệ thống sẽ tiếp tục.
* Nếu chọn “Không”, quá trình xóa sẽ dừng lại.

B6. Thực hiện xóa thông tin: Hệ thống sẽ xóa thông tin đã chọn khỏi cơ sở dữ liệu.

B7. Xác nhận xóa thành công:

Hệ thống hiển thị thông báo xác nhận rằng thông tin đã được xóa thành công. Nếu có lỗi trong quá trình xóa (ví dụ: lỗi kết nối cơ sở dữ liệu), hệ thống sẽ thông báo lỗi và yêu cầu thử lại.

##### Quy trình xóa thông tin của giảng viên

B1. Giảng viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện quản lý điểm .

B3. Tìm kiếm thông tin cần xóa: Chọn thông tin cần xóa từ danh sách hiển thị.

B4. Chọn chức năng xóa: Nhấn vào nút “Xóa” bên cạnh thông tin cần xóa.

B5. Xác nhận xóa:

Hệ thống sẽ hiển thị thông báo xác nhận (“Bạn có chắc chắn muốn xóa thông tin này?”).

* Nếu người dùng chọn “Có”, hệ thống sẽ tiếp tục.
* Nếu chọn “Không”, quá trình xóa sẽ dừng lại.

B6. Thực hiện xóa thông tin: Hệ thống sẽ xóa thông tin đã chọn khỏi cơ sở dữ liệu.

B7. Xác nhận xóa thành công: Hệ thống hiển thị thông báo xác nhận rằng thông tin đã được xóa thành công.

#### I.1.5. Chức năng Tìm kiếm

|  |  |  |
| --- | --- | --- |
| **Quản trị viên** | **Giảng viên** | **Sinh viên** |
| Thông tin có thể tìm kiếm:   * Môn học * Giảng viên * Sinh viên * Hoạt động ngoại khóa * Điểm trung bình * Điểm rèn luyện * Xếp loại | Thông tin có thể tìm kiếm:   * Điểm số của sinh viên * Thông tin sinh viên (để quản lý hiệu quả hơn) * Điểm trung bình * Điểm rèn luyện * Xếp loại | Thông tin có thể tìm kiếm:   * Môn học * Hoạt động ngoại khóa * Thông tin cá nhân * Điểm trung bình * Điểm rèn luyện * Xếp loại |

B1. Người dùng đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện tìm kiếm: Từ menu chính, chọn chức năng “Tìm kiếm”.

B3. Nhập từ khóa tìm kiếm: Nhập từ khóa vào ô tìm kiếm (ví dụ: tên môn học, tên giảng viên, mã sinh viên, kết quả học tập).

B4. Chọn loại thông tin cần tìm kiếm:

Lựa chọn loại thông tin mà người dùng muốn tìm (môn học, giảng viên, sinh viên, điểm số, hoạt động ngoại khóa, kết quả, xếp loại).

B5. Nhấn nút “Tìm kiếm”: Nhấn vào nút “Tìm kiếm” để thực hiện việc tìm kiếm.

B6. Xem kết quả tìm kiếm:

Hệ thống hiển thị danh sách kết quả phù hợp với từ khóa đã nhập. Nếu không tìm thấy kết quả nào, hệ thống sẽ thông báo “Không tìm thấy thông tin phù hợp”.

#### I.1.6. Chức năng Nhập kết quả theo môn học

Quy trình nhập điểm của giảng viên

B1. Giảng viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện nhập kết quả: Từ menu chính, chọn “Nhập kết quả”.

B3. Chọn môn học: Giảng viên chọn môn học mà họ muốn nhập kết quả từ danh sách môn học đã được phân công.

B4. Hệ thống sẽ hiển thị danh sách sinh viên đăng ký môn học đó cùng với các ô để nhập điểm.

B5. Nhập điểm cho từng sinh viên: Giảng viên nhập điểm giữa kỳ, điểm cuối kỳ, và các điểm kiểm tra cho từng sinh viên.

Thông tin cần nhập:

* Điểm giữa kỳ
* Điểm cuối kỳ

B6. Tính điểm tổng kết: Hệ thống tự động tính điểm tổng kết và điểm trung bình dựa trên những điểm đã nhập.

B7. Kiểm tra thông tin:

Hệ thống kiểm tra tính hợp lệ của thông tin đã nhập (ví dụ: điểm không được lớn hơn 10, không được nhỏ hơn 0). Nếu có lỗi, hệ thống hiển thị thông báo lỗi và yêu cầu điều chỉnh.

B8. Gửi yêu cầu nhập kết quả: Nhấn nút “Lưu” để gửi yêu cầu nhập kết quả.

B9. Xác nhận nhập thành công:

Hệ thống hiển thị thông báo xác nhận rằng kết quả đã được nhập thành công. Nếu có lỗi trong quá trình nhập (ví dụ: lỗi kết nối cơ sở dữ liệu), hệ thống sẽ thông báo lỗi và yêu cầu thử lại.

#### I.1.7. Chức năng Tính điểm trung bình

Mô tả: Chức năng này cho phép hệ thống tự động tính toán điểm trung bình của sinh viên ngay khi giảng viên nhập điểm cho các môn học và nhấn nút lưu. Hệ thống sẽ cập nhật điểm trung bình cho từng sinh viên mà không cần giảng viên thực hiện tính toán thủ công.

Sau bước 9 ở phần nhập điểm, hệ thống sẽ tiếp tục

B10. Tính điểm trung bình tự động:

Khi giảng viên nhấn nút “Lưu”, hệ thống tự động tính toán điểm trung bình dựa trên các điểm đã nhập theo công thức:

* Điểm trung bình môn = Điểm giữa kì x 40% + Điểm cuối kì x 60%
* Điểm trung bình học kì =

B11. Lưu kết quả: Hệ thống ghi nhận điểm trung bình vào cơ sở dữ liệu.

B12. Xác nhận lưu thành công: Hệ thống hiển thị thông báo xác nhận rằng điểm đã được lưu thành công.

B13. Hiển thị kết quả; Điểm trung bình sẽ được hiển thị rõ ràng trên giao diện, cùng với thông tin chi tiết về từng môn học và điểm số cụ thể.

#### I.1.8. Chức năng Đăng kí tham gia các hoạt động ngoại khóa

Mô tả: Chức năng này cho phép sinh viên đăng ký tham gia các hoạt động ngoại khóa do trường tổ chức. Hệ thống sẽ giúp sinh viên dễ dàng theo dõi các hoạt động, đồng thời quản lý thông tin đăng ký của họ. Mỗi sinh viên đăng kí tối đa 10 hoạt động.

Quy trình đăng kí tham gia hoạt động ngoại khóa

B1. Sinh viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện hoạt động ngoại khóa: Từ menu chính, chọn “Hoạt động ngoại khóa”.

B3. Xem danh sách hoạt động: Hệ thống hiển thị danh sách các hoạt động ngoại khóa đang được tổ chức, bao gồm thông tin chi tiết như tên hoạt động, thời gian, địa điểm và mô tả.

B4. Chọn hoạt động muốn tham gia: Sinh viên tìm kiếm và chọn hoạt động mà họ muốn đăng ký tham gia.

B5. Kiểm tra số lượng hoạt động đã đăng ký:

Hệ thống kiểm tra xem sinh viên đã đăng ký tối đa 10 hoạt động hay chưa. Nếu đã đủ, hệ thống sẽ thông báo và không cho phép đăng ký thêm.

B6. Điền thông tin đăng ký: Hệ thống yêu cầu sinh viên điền vào mẫu đăng ký, bao gồm:

* Mã sinh viên
* Tên sinh viên
* Thông tin liên hệ

B7. Xác nhận đăng ký: Sau khi điền thông tin, sinh viên nhấn nút “Đăng ký”. Hệ thống sẽ hiển thị thông báo xác nhận đăng ký thành công.

B8. Xem danh sách hoạt động đã đăng ký: Sinh viên có thể xem lại danh sách các hoạt động mà họ đã đăng ký tham gia từ trang cá nhân.

**Lưu ý:**

Mỗi sinh viên chỉ được phép đăng ký tối đa 10 hoạt động. Hệ thống sẽ tự động kiểm tra số lượng hoạt động đã đăng ký của sinh viên để đảm bảo không vượt quá giới hạn.

#### I.1.9. Chức năng Ghi nhận điểm rèn luyện

Chức năng này cho phép quản trị viên ghi nhận điểm rèn luyện cho sinh viên dựa trên các hoạt động ngoại khóa mà họ đã đăng ký và tham gia. Nếu sinh viên không tham gia hoạt động nào, họ sẽ nhận điểm 0 cho hoạt động đó, ngược lại nếu hoàn thành tốt sẽ được 10 điểm. Điểm tốt đa của sinh viên là 100/100 điểm.

Quy trình ghi nhận điểm:

B1. Quản trị viên đăng nhập vào tài khoản của mình.

B2. Truy cập vào giao diện ghi nhận điểm rèn luyện: Từ menu chính, chọn “Điểm rèn luyện”.

B3. Chọn hoạt động ngoại khóa: Hệ thống hiển thị danh sách các hoạt động ngoại khóa đã diễn ra, bao gồm tên hoạt động, thời gian, và danh sách sinh viên đã đăng ký.

B4. Xem danh sách sinh viên tham gia: Quản trị viên chọn hoạt động và xem danh sách sinh viên đã đăng ký tham gia hoạt động đó.

B5. Ghi nhận điểm rèn luyện: Quản trị viên ghi nhận điểm rèn luyện cho từng sinh viên:

* Sinh viên tham gia: Nhập điểm rèn luyện (ví dụ: 1-10 điểm).
* Sinh viên không tham gia: Tự động ghi nhận 0 điểm cho hoạt động đó.

B6. Kiểm tra và xác nhận: Hệ thống kiểm tra tính hợp lệ của điểm rèn luyện đã nhập. Nếu có lỗi, hệ thống sẽ thông báo và yêu cầu điều chỉnh.

B7. Lưu kết quả ghi nhận: Nhấn nút “Lưu” để ghi nhận điểm rèn luyện cho sinh viên.

B8. Xác nhận ghi nhận thành công: Hệ thống hiển thị thông báo xác nhận rằng điểm rèn luyện đã được ghi nhận thành công cho tất cả sinh viên.

**Lưu ý:**

Mỗi sinh viên chỉ được phép đăng ký tối đa 10 hoạt động ngoại khóa. Hệ thống sẽ kiểm tra số lượng hoạt động đã đăng ký của sinh viên để đảm bảo không vượt quá giới hạn này.

Quản trị viên có thể xem lịch sử điểm rèn luyện của từng sinh viên để hỗ trợ trong việc đánh giá tổng thể.

#### I.1.10. Chức năng Hiển thị xếp loại học tập

Mô tả: Chức năng này cho phép hệ thống tự động hiển thị xếp loại học tập cho sinh viên dựa trên điểm trung bình học tập và điểm rèn luyện. Hệ thống sẽ tính toán và cập nhật xếp loại một cách tự động, giúp sinh viên dễ dàng theo dõi kết quả học tập của mình.

Quy trình xếp loại của hệ thống

B1. Hệ thống tự động thu thập dữ liệu: Hệ thống sẽ lấy thông tin về điểm trung bình các môn học và điểm rèn luyện của từng sinh viên.

B2. Tính toán xếp loại: Dựa trên các tiêu chí sau, hệ thống sẽ tự động xác định xếp loại cho sinh viên:

* Xuất sắc: điểm trung bình từ 9.0 trở lên
* Giỏi: Điểm trung bình từ 8.0 trở lên.
* Khá: Điểm trung bình từ 6.5 đến 7.9.
* Trung bình: Điểm trung bình từ 5.0 đến 6.4.
* Yếu: Điểm trung bình dưới 5.0.

B3. Cập nhật điểm rèn luyện:

* Xuất sắc: ≥ 90
* Tốt: 79 - 89
* Khá: 65 - 79
* Trung bình: 50 - 64
* Yếu: < 50

B4. Hệ thống sẽ tính toán xếp loại dựa vào điểm trung bình và điểm rèn luyện của sinh viên:

* Xếp loại xuất sắc: Điểm trung bình từ 9.0 trở lên và Điểm rèn luyện từ 90 trở lên
* Xếp loại giỏi: Điểm trung bình từ 8.0 trở lên và Điểm rèn luyện từ 80 trở lên.
* Xếp loại khá: Điểm trung bình từ 6.5 đến 7.9 và Điểm rèn luyện từ 65 trở lên (Khá, Tốt hoặc Xuất sắc). Hoặc, điểm trung bình từ 8.0 trở lên và Điểm rèn luyện từ 50 đến 64 (Trung bình hoặc Khá).
* Xếp loại trung Bình: Điểm trung bình từ 5.0 đến 6.4 và Điểm rèn luyện từ 65 trở lên (Khá, Tốt hoặc Xuất sắc). Hoặc, điểm trung bình từ 6.5 đến 7.9 và Điểm rèn luyện từ 50 đến 64 (Trung bình).
* Xếp loại yếu: Điểm trung bình dưới 5.0 hoặc Điểm rèn luyện dưới 50.

B4. Hiển thị kết quả: Hệ thống sẽ hiển thị xếp loại học tập bên cạnh thông tin cá nhân của sinh viên trên trang cá nhân hoặc trong danh sách tổng hợp kết quả học tập.

B5. Cập nhật xếp loại khi có thay đổi:

Nếu có thay đổi trong điểm trung bình các môn học hoặc điểm rèn luyện (ví dụ, khi giảng viên cập nhật điểm hoặc ghi nhận điểm rèn luyện), hệ thống sẽ tự động tính toán lại và cập nhật xếp loại cho sinh viên.

### I.2. Đặc tả yêu cầu phi chức năng

#### I.2.1. Yêu cầu về hiệu năng

Yêu cầu về hiệu năng đảm bảo rằng hệ thống hoạt động hiệu quả, đáp ứng nhanh chóng và ổn định dưới các điều kiện sử dụng khác nhau.

##### Thời gian phản hồi

Thời gian phản hồi cho các thao tác người dùng (như đăng nhập, đăng ký, nhập điểm, và hiển thị kết quả) không được vượt quá 2 giây trong điều kiện tải bình thường. Thời gian phản hồi cho các truy vấn dữ liệu lớn (như tìm kiếm sinh viên, xem danh sách hoạt động ngoại khóa) không được vượt quá 5 giây.

##### Khả năng xử lý

Hệ thống có khả năng xử lý tối thiểu 1000 người dùng đồng thời mà không gây ra hiện tượng gián đoạn hoặc giảm hiệu suất. Hệ thống phải có khả năng xử lý tối thiểu 5000 giao dịch (như ghi nhận điểm, đăng ký hoạt động) mỗi giờ.

##### Tốc độ tải trang

Thời gian tải các trang giao diện người dùng không được vượt quá 3 giây trên các thiết bị di động và máy tính để bàn trong điều kiện kết nối internet ổn định.

##### Tính sẵn sàng

Hệ thống phải đảm bảo tính sẵn sàng tối thiểu 99.5% trong một tháng, tức là không quá 36 giờ bảo trì hoặc sự cố không mong muốn. Đối với các hoạt động quan trọng, hệ thống cần có khả năng phục hồi nhanh chóng trong vòng 1 giờ sau sự cố.

##### Quản lí tài nguyên

Hệ thống cần tối ưu hóa việc sử dụng tài nguyên, bao gồm CPU, bộ nhớ và băng thông, để đảm bảo không gây quá tải cho máy chủ trong các thời điểm cao điểm.

#### I.2.2. Yêu cầu về bảo mật

Yêu cầu về bảo mật đảm bảo rằng hệ thống bảo vệ thông tin người dùng, dữ liệu nhạy cảm và duy trì tính toàn vẹn của các giao dịch. Dưới đây là các yêu cầu cụ thể về bảo mật của hệ thống:

##### Xác thực người dùng

Hệ thống phải yêu cầu người dùng xác thực thông qua tên đăng nhập và mật khẩu. Mật khẩu phải có độ dài tối thiểu 8 ký tự và bao gồm chữ hoa, chữ thường, số và ký tự đặc biệt. Hệ thống nên hỗ trợ xác thực hai yếu tố (2FA) để tăng cường bảo mật trong quá trình đăng nhập.

##### Quản lí phiên

Hệ thống tự động đăng xuất người dùng sau 15 phút không hoạt động để ngăn chặn truy cập trái phép. Các phiên đăng nhập phải được mã hóa và không lưu trữ thông tin nhạy cảm dưới dạng văn bản không được mã hóa.

##### Mã hóa dữ liệu

Tất cả dữ liệu nhạy cảm (bao gồm thông tin cá nhân, mật khẩu, và điểm số) phải được mã hóa khi lưu trữ và trong quá trình truyền tải. Sử dụng các giao thức bảo mật như HTTPS để bảo vệ dữ liệu trong quá trình truyền tải giữa máy khách và máy chủ.

##### Phân quyền truy cập

Hệ thống phải thực hiện quản lý quyền truy cập để đảm bảo rằng chỉ những người dùng có quyền hạn mới có thể truy cập vào các chức năng và dữ liệu nhạy cảm. Cần có các vai trò rõ ràng (ví dụ: quản trị viên, giảng viên, sinh viên) với các quyền truy cập tương ứng.

##### Giám sát và ghi nhận

Hệ thống phải có khả năng ghi nhận tất cả các hoạt động của người dùng (bao gồm đăng nhập, đăng xuất, và các thao tác quan trọng) để phục vụ cho việc kiểm tra và giám sát. Cần có các công cụ giám sát để phát hiện và cảnh báo về các hành vi bất thường hoặc cố gắng truy cập trái phép.

##### Bảo vệ dữ liệu

Hệ thống phải có các biện pháp sao lưu dữ liệu thường xuyên để đảm bảo không mất dữ liệu trong trường hợp sự cố. Phải có quy trình khôi phục dữ liệu khi có sự cố xảy ra, đảm bảo khả năng khôi phục nhanh chóng và hiệu quả.

#### I.2.3. Yêu cầu về khả năng mở rộng

Thiết kế kiến trúc: Hệ thống cần được xây dựng với kiến trúc module, cho phép thêm chức năng mới mà không cần thay đổi cấu trúc hiện tại. Ví dụ, các module như quản lý điểm, quản lý hoạt động ngoại khóa có thể được phát triển độc lập.

Tăng cường quy mô: Hệ thống phải có khả năng mở rộng để hỗ trợ tối thiểu 10,000 người dùng đồng thời. Điều này bao gồm khả năng xử lý tăng lên của cơ sở dữ liệu và máy chủ.

Tích hợp hệ thống: Hệ thống nên hỗ trợ tích hợp với các nền tảng khác như hệ thống quản lý học tập (LMS) và các ứng dụng bên ngoài thông qua API để mở rộng chức năng mà không ảnh hưởng đến hoạt động hiện tại.

#### I.2.4. Yêu cầu về khả năng sử dụng

Giao diện người dùng (UI): Giao diện phải thân thiện, dễ điều hướng với các biểu tượng rõ ràng, sử dụng màu sắc và phông chữ dễ đọc. Tất cả các chức năng chính cần được truy cập trong tối đa 3 cú nhấp chuột.

Hỗ trợ đa dạng người dùng: Cần có các chế độ hiển thị khác nhau cho từng loại người dùng (quản trị viên, giảng viên, sinh viên) để đảm bảo mỗi người dùng có thể truy cập thông tin và chức năng cần thiết một cách dễ dàng.

Tài liệu hướng dẫn: Phải có tài liệu hướng dẫn sử dụng chi tiết cho từng chức năng và hệ thống trợ giúp trực tuyến sẵn có để hỗ trợ người dùng trong quá trình sử dụng.

#### I.2.5. Yêu cầu về khả năng tương thích

Trình duyệt: Hệ thống phải hoạt động trên các trình duyệt phổ biến như Google Chrome, Mozilla Firefox, Safari, và Microsoft Edge mà không gặp phải vấn đề về hiển thị hoặc chức năng.

Thiết bị: Hệ thống cần được tối ưu cho cả máy tính để bàn và thiết bị di động (smartphone, tablet), đảm bảo trải nghiệm người dùng mượt mà trên các kích thước màn hình khác nhau.

Định dạng dữ liệu: Hệ thống phải hỗ trợ nhập và xuất dữ liệu dưới các định dạng phổ biến như CSV, Excel, PDF, cho phép người dùng dễ dàng làm việc với dữ liệu.

#### I.2.6. Yêu cầu về độ tin cậy

Tỷ lệ lỗi: Hệ thống phải đảm bảo tỷ lệ lỗi không vượt quá 0.5% trong quá trình hoạt động bình thường, đảm bảo rằng người dùng có thể hoàn thành các tác vụ mà không gặp sự cố lớn.

Dự phòng và sao lưu: Cần có cơ chế sao lưu dữ liệu tự động hàng ngày và phục hồi trong vòng 1 giờ sau sự cố để bảo vệ dữ liệu người dùng.

Kiểm tra định kỳ: Hệ thống cần được kiểm tra định kỳ để phát hiện và khắc phục sớm các lỗi tiềm ẩn.

#### I.2.7. Yêu cầu về bảo trì

Quy trình bảo trì: Cần xây dựng quy trình bảo trì rõ ràng, bao gồm lịch trình bảo trì định kỳ (ví dụ: hàng tháng) để kiểm tra hiệu suất và bảo mật.

Cập nhật phần mềm: Hệ thống phải dễ dàng cập nhật phiên bản mới mà không làm gián đoạn dịch vụ cho người dùng, với thời gian tối thiểu cho việc cập nhật.

Tài liệu kỹ thuật: Cung cấp đầy đủ tài liệu kỹ thuật cho đội ngũ IT để hỗ trợ quá trình bảo trì và cập nhật, bao gồm hướng dẫn cài đặt, cấu hình và kiểm tra hệ thống.

### I.3. Phạm vi project

#### I.3.1. Trong phạm vi

Chức năng chính của hệ thống:

* Đăng ký tài khoản: Cho phép sinh viên và giáo viên tạo tài khoản.
* Đăng nhập: Hỗ trợ xác thực người dùng để truy cập hệ thống.
* Quản lý thông tin cá nhân: Cung cấp chức năng cho người dùng cập nhật và xóa thông tin cá nhân.
* Nhập và quản lý kết quả học tập: Giáo viên có thể nhập điểm số và nhận xét cho sinh viên.
* Tính toán và hiển thị điểm trung bình: Hệ thống tự động tính điểm trung bình dựa trên điểm số và số tín chỉ.
* Tham gia hoạt động ngoại khóa: Sinh viên có thể đăng ký tham gia các hoạt động ngoại khóa do trường tổ chức.
* Ghi nhận điểm rèn luyện: Cập nhật điểm rèn luyện dựa trên sự tham gia của sinh viên.
* Xếp loại học tập: Tính toán và hiển thị xếp loại học tập dựa trên điểm trung bình và điểm rèn luyện.

Đối tượng người dùng:

* Sinh viên: Truy cập và quản lý thông tin học tập.
* Giáo viên: Nhập điểm và nhận xét cho sinh viên.
* Quản trị viên: Quản lý hệ thống và dữ liệu người dùng.

Thời gian thực hiện: Dự kiến hoàn thành trong vòng 1 tháng.

#### I.3.2. Ngoài phạm vi

Chức năng không bao gồm:

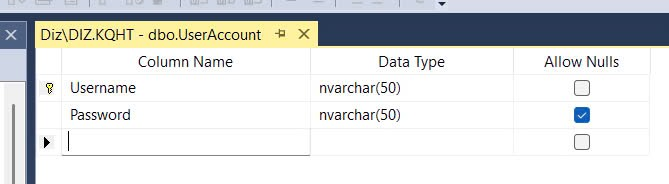
* Phát triển ứng dụng di động riêng biệt; hệ thống sẽ chỉ tối ưu hóa cho trình duyệt web.
* Tích hợp với các hệ thống bên ngoài, ngoài các API cơ bản.
* Quản lý tài chính hoặc học bổng sinh viên.
* Hỗ trợ ngôn ngữ: Hệ thống chỉ hỗ trợ hai ngôn ngữ chính (Tiếng Việt và Tiếng Anh) ở giai đoạn đầu, không bao gồm các ngôn ngữ khác.
* Tính năng nâng cao: Không bao gồm các tính năng phân tích dữ liệu học tập nâng cao hoặc trí tuệ nhân tạo trong giai đoạn phát triển đầu tiên.

## CHƯƠNG II. THIẾT KẾ DATABASE

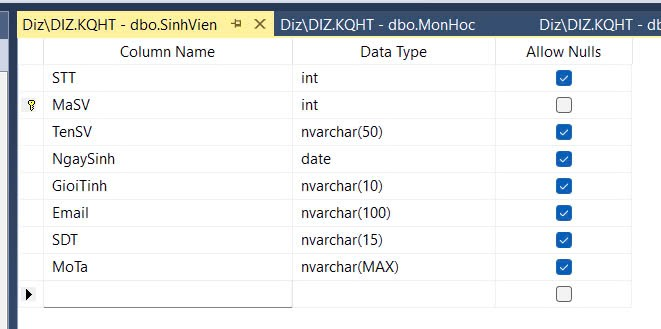
### II.1. Table design

#### II.1.1. Table design ở SQL Server

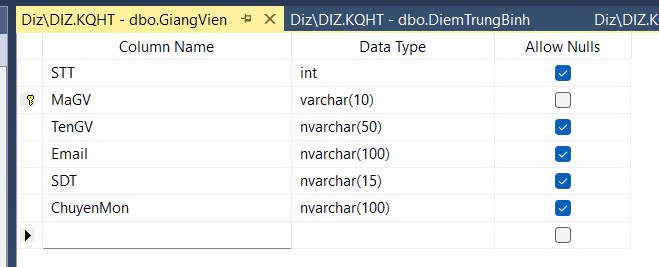
##### Table UserAccount



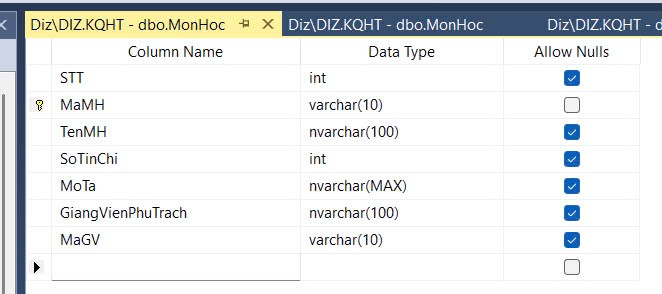
##### Table SinhVien



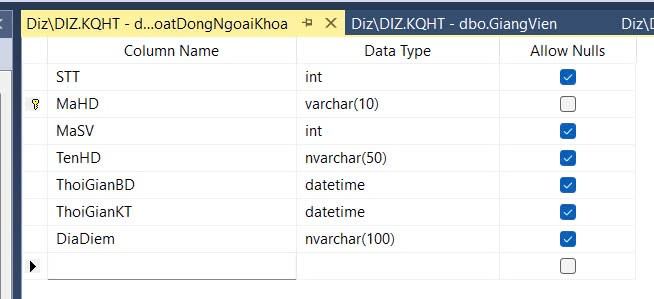
##### Table GiangVien



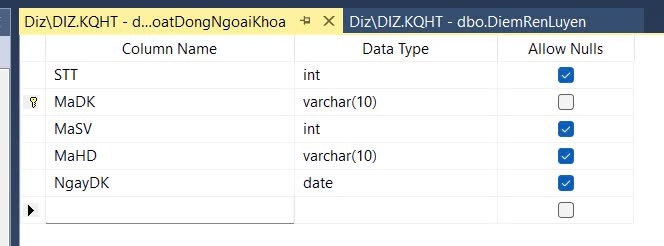
##### Table MonHoc



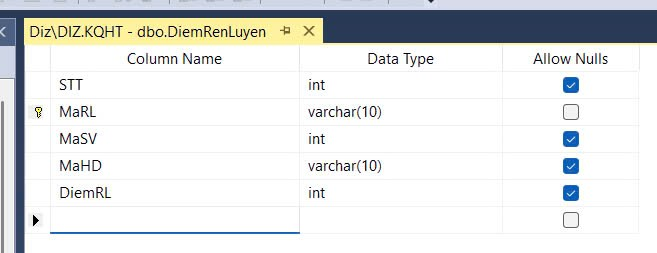
##### Table HoatDongNgoaiKhoa



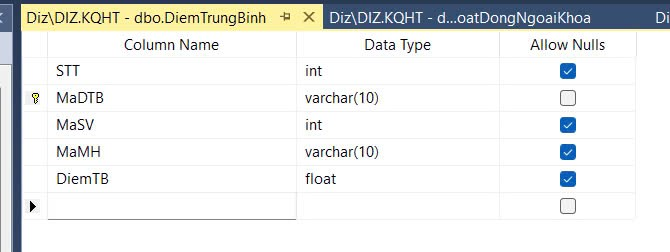
##### Table DangKiHoatDongNgoaiKhoa



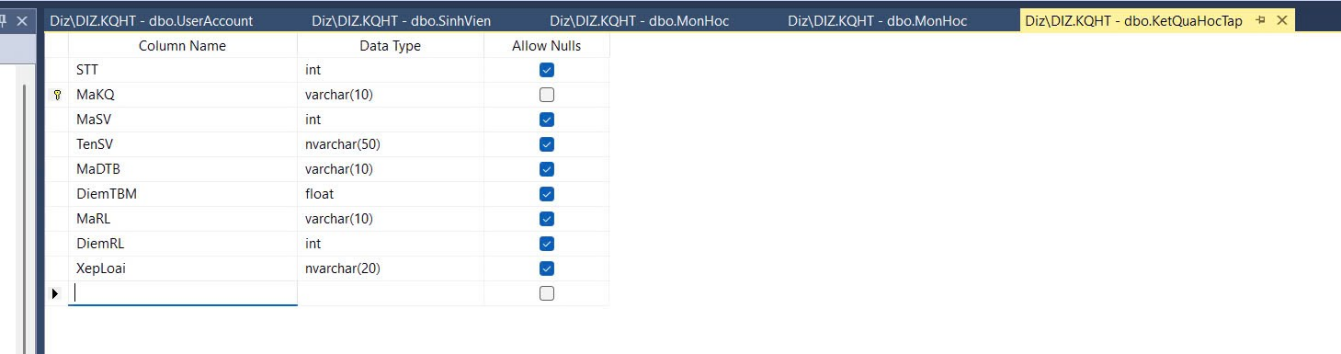
##### Table DiemRenLuyen



##### Table DiemTrungBinh



##### Table KetQuaHocTap



#### II.1.2. Mô tả chi tiết

##### Table UserAccount

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| Username | nvarchar (50) | Tên đăng nhập của admin, giảng viên, sinh viên, khóa chính |
| Password | nvarchar (50) | Mật khẩu đăng nhập |

##### Table SinhVien

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| STT | int | Số thứ tự sinh viên |
| MaSV | int (10) | Mã sinh viên, khóa chính |
| TenSV | nvarchar (50) | Tên sinh viên |
| NgaySinh | date | Ngày sinh |
| GioiTinh | nvarchar (10) | Giới tính |
| Email | nvarchar (100) | Email của sinh viên |
| SDT | nvarchar (15) | Số điện thoại sinh viên |
| MoTa | nvarchar (MAX) | Mô tả của sinh viên |

##### Table Giảng viên

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| STT | int | Số thứ tự giảng viên |
| MaGV | varchar (10) | Mã giảng viên, khóa chính |
| TenGV | nvarchar (50) | Tên giảng viên |
| Email | nvarchar (100) | Email của giảng viên |
| SDT | nvarchar (15) | Số điện thoại giảng viên |
| ChuyenMon | nvarchar (100) | Chuyên môn chuyên ngành của giảng viên |

##### Table Môn học

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| STT | int | Số thứ tự môn học |
| MaMH | varchar (10) | Mã môn học, khóa chính |
| TenMH | nvarchar (100) | Tên môn học |
| SoTinChi | int | Số tín chỉ của môn học |
| MoTa | nvarchar (MAX) | Mô tả chi tiết về môn học |
| GiangVienPhuTrach | nvarchar (100) | Tên của giảng viên phụ trách |
| MaGV | varchar (10) | Mã giảng viên, khóa ngoại tham chiếu đến *MaGV* của bảng *GiangVien* |

##### Table Hoạt động ngoại khóa

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| STT | int | Số thứ tự của hoạt động |
| MaHD | varchar (10) | Mã hoạt động, khóa chính |
| MaSV | int | Mã sinh viên, khóa ngoại tham chiếu đến *MaSV* của bảng *SinhVien* |
| TenHD | nvarchar (50) | Tên hoạt động ngoại khóa |
| ThoiGianBD | datetime | Thời gian bắt đầu |
| ThoiGianKT | datetime | Thời gian kết thúc |
| DiaDiem | nvarchar (100) | Địa điểm cụ thể cho hoạt động |

##### Table Đăng kí hoạt động ngoại khóa

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| STT | int | Số thứ tự đăng kí |
| MaDK | varchar (10) | Mã đăng kí, khóa chính |
| MaSV | int | Mã sinh viên, khóa ngoại tham chiếu đến *MaSV* của bảng *SinhVien* |
| MaHD | varchar (10) | Mã hoạt động, khóa ngoại tham chiếu đến *MaHD* của bảng *HoatDongNgoaiKhoa* |
| NgayDK | date | Ngày đăng kí |

##### Table Điểm rèn luyện

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| STT | int | Số thứ tự của điểm rèn luyện |
| MaRL | varchar (10) | Mã rèn luyện, khóa chính |
| MaSV | int | Mã sinh viên, khóa ngoại tham chiếu đến *MaSV* của bảng *SinhVien* |
| MaHD | varchar (10) | Mã hoạt động, khóa ngoại tham chiếu đến *MaHĐ* của bảng *HoatDongNgoaiKhoa* |
| DiemRL | int | Điểm trung bình của các hoạt động ngoại khóa, mỗi hoạt động là 10đ |

##### Table Điểm trung bình

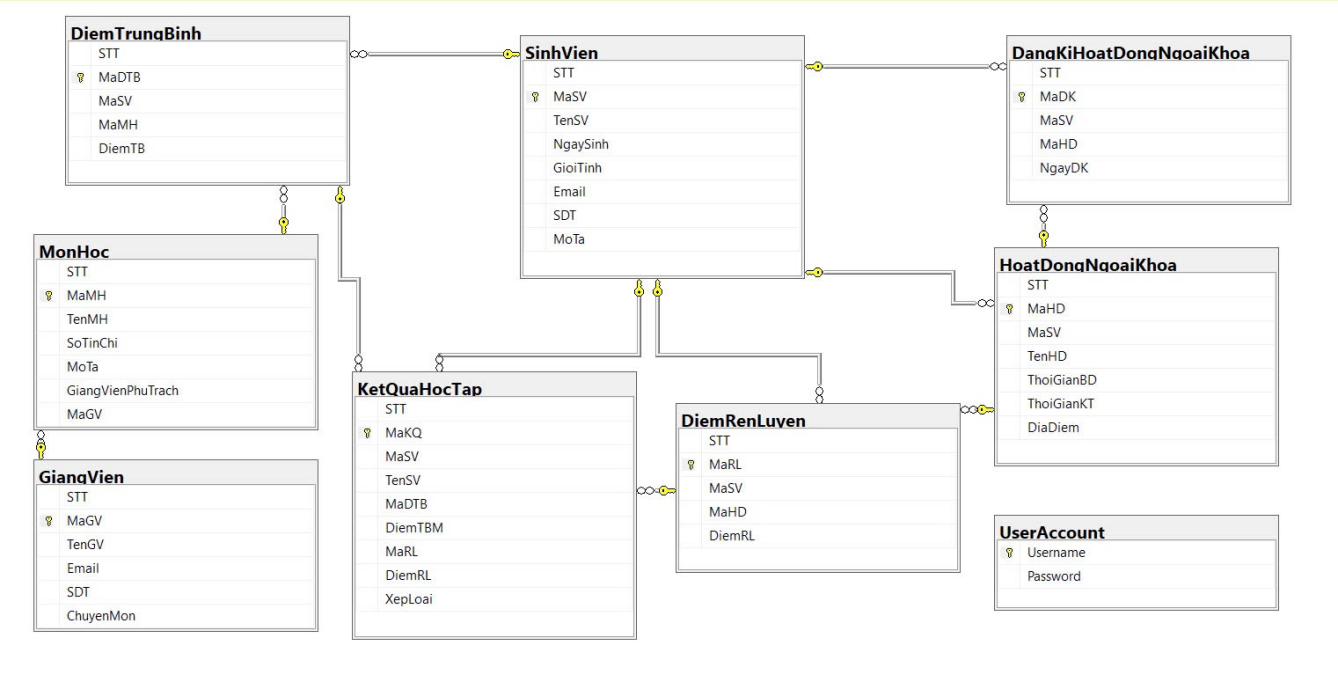
|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| STT | int | Số thứ tự của điểm trung bình |
| MaDTB | varchar (10) | Mã điểm trung bình, khóa chính |
| MaSV | int | Mã sinh viên, khóa ngoại tham chiếu đến *MaSV* của bảng *SinhVien* |
| MaMH | varchar (10) | Mã môn học, khóa ngoại tham chiếu đến *MaMH* của bảng *MonHoc* |
| DiemTB | float | Điểm trung bình của các môn học |

##### Table Kết quả học tập

|  |  |  |
| --- | --- | --- |
| Tên trường | Kiểu dữ liệu | Mô tả |
| STT | int | Số thứ tự của kết quả |
| MaKQ | varchar (10) | Mã kết quả, khóa chính |
| MaSV | int | Mã sinh viên, khóa ngoại tham chiếu đến *MaSV* của bảng *SinhVien* |
| TenSV | nvarchar (50) | Tên sinh viên |
| MaDTB | varchar (10) | Mã điểm trung bình, khóa ngoại tham chiếu đến *MaDTB* của bảng *DiemTrungBinh* |
| DiemTBM | float |  |
| MaRL | varchar (10) | Mã rèn luyện, khóa ngoại tham chiếu đến *MaRL* của bảng *DiemRenLuyen* |
| DiemRL | int | Điểm trung bình của các hoạt động ngoại khóa, mỗi hoạt động là 10đ |
| XepLoai | nvarchar (20) | Xếp loại học tập |

### II.2. Database diagram

Sơ đồ Database diagram



## CHƯƠNG III. THIẾT KẾ KIẾN TRÚC

### III.1. Kiến trúc ứng dụng

Phần mềm quản lý kết quả học tập được thiết kế theo kiến trúc MVC (Model-View-Controller), một mô hình thiết kế phần mềm phổ biến giúp tách biệt rõ ràng các thành phần của ứng dụng nhằm nâng cao khả năng bảo trì, mở rộng và kiểm thử. Kiến trúc MVC cho phép phân chia ứng dụng thành ba thành phần chính, mỗi thành phần có vai trò và chức năng riêng biệt, tạo điều kiện thuận lợi cho việc phát triển và quản lý ứng dụng.

#### III.1.1. Model

Model là phần trung tâm của ứng dụng, giữ vai trò quan trọng trong việc quản lý dữ liệu và logic nghiệp vụ. Trong bối cảnh phần mềm quản lý kết quả học tập, Model bao gồm các lớp dữ liệu đại diện cho các thực thể trong hệ thống, như SinhVien, GiangVien, MonHoc, DiemTrungBinh, và nhiều lớp khác. Mỗi lớp Model sẽ chứa các thuộc tính, phương thức, và logic nghiệp vụ cụ thể:

##### Các lớp thực thể

* SinhVien: Lớp này định nghĩa các thuộc tính như MaSV, TenSV, NgaySinh, GioiTinh, Email, và SDT. Các phương thức trong lớp này có thể bao gồm:

- ThemSinhVien(): Thực hiện thêm thông tin sinh viên vào cơ sở dữ liệu.

- SuaSinhVien(): Cập nhật thông tin sinh viên.

- XoaSinhVien(): Xóa thông tin sinh viên khỏi hệ thống.

- LayDanhSachSinhVien(): Truy xuất danh sách sinh viên từ cơ sở dữ liệu.

* GiangVien: Lớp này chứa thông tin về giảng viên, với các thuộc tính như MaGV, TenGV, Email, và ChuyenMon. Tương tự, nó sẽ có các phương thức để quản lý thông tin giảng viên.
* MonHoc: Lớp này đại diện cho các môn học, bao gồm các thuộc tính như MaMH, TenMH, SoTinChi, và GiangVienPhuTrach. Nó có thể chứa các phương thức để thêm, sửa, xóa và lấy thông tin môn học.

##### Truy cập cơ sở dữ liệu

Model tương tác trực tiếp với cơ sở dữ liệu thông qua các truy vấn SQL hoặc thông qua ORM (Object-Relational Mapping) như Entity Framework. Việc này giúp Model đảm bảo rằng mọi thay đổi trong dữ liệu đều được cập nhật kịp thời. Model sẽ xử lý các truy vấn như sau:

* Kết nối đến cơ sở dữ liệu.
* Thực hiện các thao tác thêm, sửa, xóa và truy xuất dữ liệu.
* Quản lý các giao dịch để đảm bảo tính toàn vẹn dữ liệu.

#### III.1.2. View

View là giao diện người dùng, được xây dựng bằng công nghệ Windows Forms, nhằm cung cấp trải nghiệm tương tác cho người dùng. Các yếu tố trong View sẽ bao gồm:

##### Giao diện người dùng

Form chính: Hiển thị tổng quan về hệ thống, cho phép người dùng dễ dàng truy cập vào các chức năng khác nhau như quản lý sinh viên, giảng viên, và môn học. Các bảng và biểu đồ có thể được sử dụng để trình bày thông tin một cách rõ ràng.

Form thêm/sửa thông tin: Các form này cho phép người dùng nhập liệu hoặc chỉnh sửa thông tin. Mỗi form sẽ có các điều khiển như TextBox, ComboBox, và Button để thực hiện các thao tác như thêm, sửa hoặc xóa.

##### Xử lý sự kiện

View không chỉ hiển thị dữ liệu mà còn lắng nghe các sự kiện từ người dùng, chẳng hạn như nhấn nút hoặc chọn một mục trong danh sách. Khi người dùng tương tác, View sẽ gọi Controller để xử lý các yêu cầu và nhận phản hồi từ Model.

##### Thiết kế giao diện

Giao diện người dùng sẽ được thiết kế thân thiện, dễ sử dụng và trực quan, giúp người dùng dễ dàng theo dõi và tương tác với thông tin. Các thành phần giao diện sẽ được sắp xếp hợp lý để tạo sự thuận tiện trong việc sử dụng.

#### III.1.3. Controller

Controller đóng vai trò là cầu nối giữa Model và View. Nó quản lý các luồng thông tin và xử lý các sự kiện từ View, cụ thể như sau:

Xử lý sự kiện từ View: Controller nhận các sự kiện từ View (như nhấn nút “Thêm” hoặc “Sửa”) và thực hiện các thao tác tương ứng. Khi một sự kiện xảy ra, Controller sẽ:

- Kiểm tra dữ liệu đầu vào từ View để đảm bảo tính hợp lệ.

- Gọi các phương thức tương ứng trong Model để thực hiện các thao tác cần thiết, như thêm mới, cập nhật, hoặc xóa dữ liệu.

Cập nhật View: Sau khi Model đã thực hiện các thay đổi, Controller sẽ thông báo cho View để cập nhật giao diện người dùng. Việc này có thể bao gồm việc làm mới danh sách sinh viên, hiển thị thông báo thành công hoặc lỗi, và cập nhật các biểu đồ thống kê.

Logic nghiệp vụ: Controller sẽ chứa các quy tắc và điều kiện nghiệp vụ cần thiết để xử lý các thao tác của người dùng. Điều này bao gồm xác thực dữ liệu đầu vào, xử lý lỗi và phản hồi người dùng một cách thích hợp.

Kiến trúc ứng dụng theo mô hình MVC trong phần mềm quản lý kết quả học tập không chỉ giúp tách biệt các thành phần của ứng dụng mà còn nâng cao khả năng bảo trì và mở rộng. Mỗi thành phần (Model, View, Controller) có vai trò và chức năng riêng, tạo điều kiện cho sự phát triển hiệu quả và khả năng kiểm thử tốt hơn. Mô hình này không chỉ đáp ứng nhu cầu hiện tại mà còn có thể mở rộng trong tương lai khi có yêu cầu mới từ người dùng.

### III.2. Kiến trúc tổng thể

Kiến trúc tổng thể của ứng dụng quản lý kết quả học tập được thiết kế để đảm bảo tính linh hoạt, dễ bảo trì và khả năng mở rộng trong tương lai. Các thành phần chính của kiến trúc bao gồm:

#### III.2.1. Giao diện người dùng (View)

Giao diện người dùng là phần đầu tiên mà người dùng tiếp xúc và tương tác với hệ thống. Giao diện được xây dựng trên nền tảng Windows Forms, cung cấp các form cho phép người dùng thực hiện các thao tác cần thiết. Mỗi form trong ứng dụng sẽ được thiết kế với mục đích rõ ràng, bao gồm:

Quản lý sinh viên: Form này cho phép người dùng thêm, sửa, xóa và xem thông tin chi tiết về sinh viên. Các thông tin như mã sinh viên, tên, ngày sinh, giới tính, email, và số điện thoại sẽ được hiển thị và cho phép nhập liệu.

Quản lý giảng viên: Tương tự, form này sẽ cho phép quản lý thông tin của giảng viên, bao gồm mã giảng viên, tên, email, và chuyên môn. Các chức năng tìm kiếm và lọc sẽ được tích hợp để người dùng dễ dàng tìm kiếm thông tin.

Theo dõi điểm số: Form này sẽ hiển thị thông tin về điểm số và kết quả học tập của sinh viên. Người dùng có thể xem các điểm trung bình, điểm rèn luyện và xếp loại học tập.

Báo cáo kết quả học tập: Form báo cáo cho phép người dùng tạo và xuất các báo cáo liên quan đến kết quả học tập của sinh viên, giảng viên, hoặc các hoạt động ngoại khóa.

Mỗi form sẽ có giao diện thân thiện, dễ sử dụng và hỗ trợ người dùng qua các thông báo và hướng dẫn khi cần thiết.

#### III.2.2. Logic xử lý nghiệp vụ (Controller)

Controller là phần chịu trách nhiệm xử lý các sự kiện từ View và thực hiện các thao tác cần thiết trên Model. Khi người dùng thực hiện một hành động (ví dụ: nhấn nút “Lưu”), Controller sẽ:

Nhận thông tin từ View: Controller sẽ nhận dữ liệu đầu vào từ các trường nhập liệu trên form, đảm bảo rằng dữ liệu được nhập vào là hợp lệ.

Thực hiện các thao tác truy cập dữ liệu: Sau khi xác nhận rằng dữ liệu là hợp lệ, Controller sẽ gọi các phương thức trong Model để thực hiện các thao tác như thêm mới, cập nhật hoặc xóa thông tin trong cơ sở dữ liệu.

Cập nhật lại View: Khi Model đã được cập nhật, Controller sẽ thông báo cho View để cập nhật giao diện người dùng. Điều này có thể bao gồm việc làm mới danh sách sinh viên, hiển thị thông báo thành công hoặc lỗi, và cập nhật thông tin hiển thị trên form.

#### III.2.3. Cơ sở dữ liệu (Model)

Model là phần quan trọng của kiến trúc, chịu trách nhiệm quản lý tất cả dữ liệu và logic nghiệp vụ liên quan đến hệ thống. Model sẽ bao gồm:

Lớp truy cập dữ liệu: Model sẽ tương tác với cơ sở dữ liệu thông qua các truy vấn SQL hoặc ORM để thực hiện các thao tác lưu trữ và truy xuất thông tin. Mọi thay đổi trong dữ liệu đều phải được thực hiện qua Model để đảm bảo tính toàn vẹn và nhất quán của dữ liệu.

Các lớp thực thể: Model sẽ bao gồm các lớp đại diện cho các thực thể trong hệ thống như SinhVien, GiangVien, MonHoc, DiemTrungBinh, và HoatDongNgoaiKhoa. Mỗi lớp sẽ chứa các thuộc tính và phương thức cụ thể để thực hiện các thao tác cần thiết.

Xác thực và xử lý dữ liệu: Model cũng sẽ thực hiện các quy tắc xác thực để đảm bảo rằng dữ liệu nhập vào là hợp lệ trước khi lưu vào cơ sở dữ liệu. Điều này bao gồm kiểm tra độ dài, định dạng email, và các quy tắc khác liên quan đến nghiệp vụ.

#### III.2.4. Thư viện hỗ trợ

Để cải thiện khả năng phát triển và bảo trì ứng dụng, một số thư viện hỗ trợ sẽ được sử dụng, bao gồm:

Thư viện kết nối cơ sở dữ liệu: Sử dụng các thư viện như ADO.NET hoặc Entity Framework để quản lý kết nối đến cơ sở dữ liệu một cách hiệu quả. Những thư viện này sẽ giúp đơn giản hóa các thao tác truy cập dữ liệu và tăng cường khả năng bảo mật.

Thư viện xác thực dữ liệu: Các thư viện hỗ trợ xác thực như FluentValidation có thể được sử dụng để dễ dàng định nghĩa các quy tắc xác thực cho dữ liệu đầu vào.

Thư viện báo cáo: Sử dụng các thư viện như Crystal Reports hoặc ReportViewer để tạo và xuất báo cáo liên quan đến kết quả học tập và hoạt động ngoại khóa.

Thư viện hỗ trợ giao diện người dùng: Sử dụng các thư viện UI giúp tạo giao diện người dùng hiện đại và thân thiện hơn, cung cấp trải nghiệm tốt hơn cho người dùng.

Kiến trúc tổng thể của ứng dụng quản lý kết quả học tập không chỉ đảm bảo tính nhất quán và khả năng mở rộng mà còn giúp tối ưu hóa quy trình phát triển và bảo trì phần mềm. Việc tách biệt các thành phần chính như Giao diện người dùng, Logic xử lý nghiệp vụ, Cơ sở dữ liệu và Thư viện hỗ trợ giúp ứng dụng hoạt động một cách hiệu quả, đồng thời giảm thiểu sự phức tạp trong việc quản lý và phát triển.

### III.3. Kiến trúc từng thành phần

#### III.3.1. Model

Model là phần trung tâm của ứng dụng, nơi quản lý tất cả dữ liệu và logic nghiệp vụ. Mỗi lớp dữ liệu trong Model sẽ được thiết kế để tương ứng với các thực thể trong hệ thống. Cụ thể:

##### Các lớp dữ liệu

Mỗi lớp sẽ chứa các thuộc tính và phương thức riêng biệt để mô tả các thực thể khác nhau. Ví dụ:

* Lớp SinhVien:

Thuộc tính: MaSV, TenSV, NgaySinh, GioiTinh, Email, SDT, MoTa.

Phương thức: TinhDiemTrungBinh(), ThamGiaHoatDong(), LayThongTin().

* Lớp GiangVien:

Thuộc tính: MaGV, TenGV, Email, SDT, ChuyenMon.

Phương thức: LayDanhSachSinhVien(), CapNhatThongTin().

* Lớp MonHoc:

Thuộc tính: MaMH, TenMH, SoTinChi, MoTa.

Phương thức: ThemSinhVien(), LayDiemTrungBinh().

##### Kết nối cơ sở dữ liệu

Sử dụng ADO.NET hoặc Entity Framework để thực hiện các truy vấn cơ sở dữ liệu. Các lớp Model sẽ bao gồm các phương thức để thực hiện các thao tác như:

Thêm mới dữ liệu: Phương thức ThemSinhVien(SinhVien sv) sẽ thêm một sinh viên mới vào cơ sở dữ liệu.

Sửa đổi dữ liệu: Phương thức CapNhatSinhVien(SinhVien sv) cho phép sửa đổi thông tin của một sinh viên đã có.

Xóa dữ liệu: Phương thức XoaSinhVien(int maSV) sẽ xóa sinh viên theo mã sinh viên.

Truy xuất thông tin: Các phương thức như LayDanhSachSinhVien() sẽ truy xuất danh sách tất cả sinh viên từ cơ sở dữ liệu.

Các truy vấn SQL sẽ được tối ưu hóa để giảm thiểu thời gian thực thi và cải thiện hiệu suất tổng thể của ứng dụng.

#### III.3.2. View

View là phần giao diện người dùng, nơi người dùng tương tác với ứng dụng. Các thành phần của View sẽ được thiết kế chi tiết như sau:

##### Form chính

Giao diện chính sẽ hiển thị thông tin tổng quan về sinh viên, giảng viên, môn học, và kết quả học tập.

Sẽ bao gồm các bảng điều khiển, menu, và các nút chức năng giúp người dùng dễ dàng điều hướng giữa các chức năng như thêm, sửa, và xóa thông tin.

##### Form thêm/sửa thông tin

Các form này sẽ cho phép người dùng thêm hoặc sửa thông tin của sinh viên, giảng viên, môn học, và hoạt động ngoại khóa.

Mỗi trường thông tin sẽ được thiết kế rõ ràng với các nhãn mô tả (label) giúp người dùng hiểu cách nhập liệu chính xác.

Sẽ có các thông báo lỗi trực quan khi người dùng nhập dữ liệu không hợp lệ, như kiểm tra định dạng email hoặc yêu cầu trường bắt buộc.

##### Form báo cáo

Form này sẽ hiển thị kết quả học tập theo từng sinh viên hoặc theo lớp học.

Sẽ bao gồm các biểu đồ thống kê và bảng dữ liệu để người dùng có thể dễ dàng theo dõi và phân tích kết quả học tập.

Người dùng cũng có thể xuất báo cáo ra file PDF hoặc Excel thông qua các nút chức năng trên form.

#### III.3.3. Controller

Controller là phần xử lý các tương tác giữa Model và View. Các chức năng chính của Controller sẽ bao gồm:

##### Xử lý sự kiện

Controller sẽ nhận và xử lý các sự kiện từ View, chẳng hạn như khi người dùng nhấn nút "Thêm sinh viên".

Khi sự kiện xảy ra, Controller sẽ lấy thông tin từ View (ví dụ: từ các trường nhập liệu), kiểm tra tính hợp lệ của dữ liệu, và gọi các phương thức trong Model để thực hiện các thao tác cần thiết.

##### Quản lý dữ liệu

Sau khi thực hiện thao tác trên Model, Controller sẽ nhận kết quả và cập nhật lại View để phản ánh những thay đổi trong dữ liệu.

Nếu Model thay đổi, Controller sẽ thông báo cho View để giao diện người dùng được cập nhật một cách tự động, chẳng hạn như làm mới danh sách sinh viên hoặc hiển thị thông báo thành công hoặc lỗi.

##### Quản lý luồng thông tin

Controller sẽ đảm bảo rằng thông tin được chuyển đúng cách giữa Model và View. Ví dụ, khi dữ liệu được cập nhật trong Model, Controller sẽ chuyển dữ liệu mới đến View để cập nhật giao diện người dùng.

### III.4. Vai trò của từng thành phần trong kiến trúc

Mỗi thành phần trong kiến trúc MVC đóng vai trò quan trọng trong việc đảm bảo hoạt động hiệu quả của ứng dụng

#### III.4.1. Thành phần Model

##### Chức năng

Quản lý dữ liệu và logic nghiệp vụ. Model chứa tất cả các quy tắc và điều kiện nghiệp vụ liên quan đến quản lý kết quả học tập.

##### Vai trò

Đảm bảo tính toàn vẹn dữ liệu: Model là nơi kiểm soát tính chính xác và hợp lệ của dữ liệu trước khi lưu trữ vào cơ sở dữ liệu. Ví dụ, khi một sinh viên được thêm mới, Model sẽ kiểm tra xem mã sinh viên có hợp lệ hay không và liệu có sinh viên nào khác đã sử dụng mã đó chưa.

Thực hiện các truy vấn và xử lý thông tin từ cơ sở dữ liệu: Model có trách nhiệm thực hiện các thao tác như thêm, sửa, xóa và truy xuất dữ liệu. Bất kỳ thay đổi nào được thực hiện trong Model đều sẽ ảnh hưởng trực tiếp đến cơ sở dữ liệu, do đó, Model cần phải tối ưu hóa các truy vấn để giảm thiểu thời gian thực thi và tăng cường hiệu suất.

Thực hiện các tính toán và quy tắc nghiệp vụ: Model sẽ xử lý tất cả các phép tính liên quan đến kết quả học tập, như tính điểm trung bình của sinh viên, xếp loại học tập dựa trên điểm số, và quản lý hoạt động ngoại khóa. Điều này giúp tách biệt logic nghiệp vụ khỏi giao diện người dùng, đảm bảo rằng mọi thay đổi trong logic sẽ không ảnh hưởng đến giao diện.

#### III.4.2. Thành phần View

##### Chức năng

Hiển thị thông tin và giao diện người dùng. View là phần trực quan nhất của ứng dụng, nơi người dùng có thể tương tác với hệ thống.

##### Vai trò

Cung cấp giao diện thân thiện cho người dùng: View thiết kế để dễ sử dụng, với các điều khiển đồ họa trực quan như nút bấm, hộp thoại, bảng dữ liệu, và biểu đồ. Giao diện được tối ưu hóa để người dùng dễ dàng tìm kiếm thông tin và thực hiện các thao tác cần thiết.

Cho phép thao tác dễ dàng và theo dõi thông tin: View không chỉ hiển thị dữ liệu mà còn cung cấp các chức năng cho phép người dùng tương tác, như nhập liệu, tìm kiếm, và xem báo cáo. Điều này tạo ra một trải nghiệm người dùng tốt hơn, giúp họ dễ dàng tiếp cận và sử dụng các chức năng của ứng dụng.

Cập nhật giao diện tự động: View sẽ được cập nhật khi Model thay đổi. Khi có thông tin mới từ Model (ví dụ: sau khi thêm một sinh viên), View sẽ tự động hiển thị dữ liệu mới mà không cần làm mới toàn bộ giao diện, giúp người dùng có trải nghiệm mượt mà hơn.

#### III.4.3. Thành phần Controller

##### Chức năng

Điều khiển luồng thông tin và xử lý các sự kiện. Controller là phần điều phối giữa Model và View.

##### Vai trò

Là cầu nối giữa Model và View: Controller nhận thông tin từ View, thực hiện các thao tác cần thiết trên Model và cập nhật lại View. Ví dụ, khi người dùng nhấn nút "Lưu", Controller sẽ nhận yêu cầu, kiểm tra tính hợp lệ của dữ liệu và sau đó gọi Model để thực hiện lưu trữ.

Đảm bảo thông tin được truyền đạt đúng đắn: Controller sẽ xác định cách thức và thời điểm các dữ liệu trong Model được gửi đến View. Nếu có bất kỳ thay đổi nào trong Model (như sửa thông tin sinh viên), Controller sẽ thông báo cho View để cập nhật giao diện người dùng.

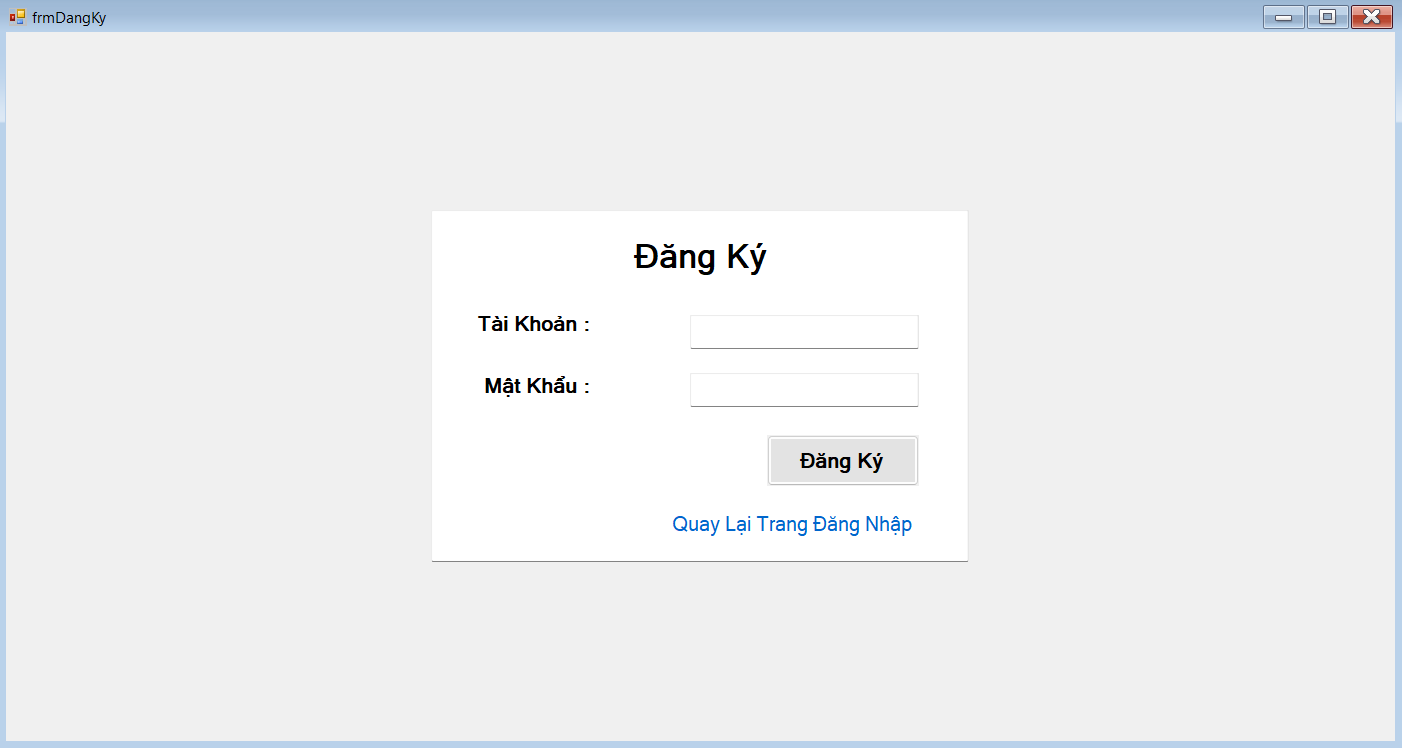
Xử lý logic nghiệp vụ khi cần thiết: Controller cũng sẽ xử lý các quy tắc nghiệp vụ đơn giản, như xác minh dữ liệu người dùng nhập vào trước khi gửi đến Model. Điều này giúp giảm tải cho Model và đảm bảo rằng Model chỉ nhận những dữ liệu hợp lệ và chính xác.

*Kết luận*

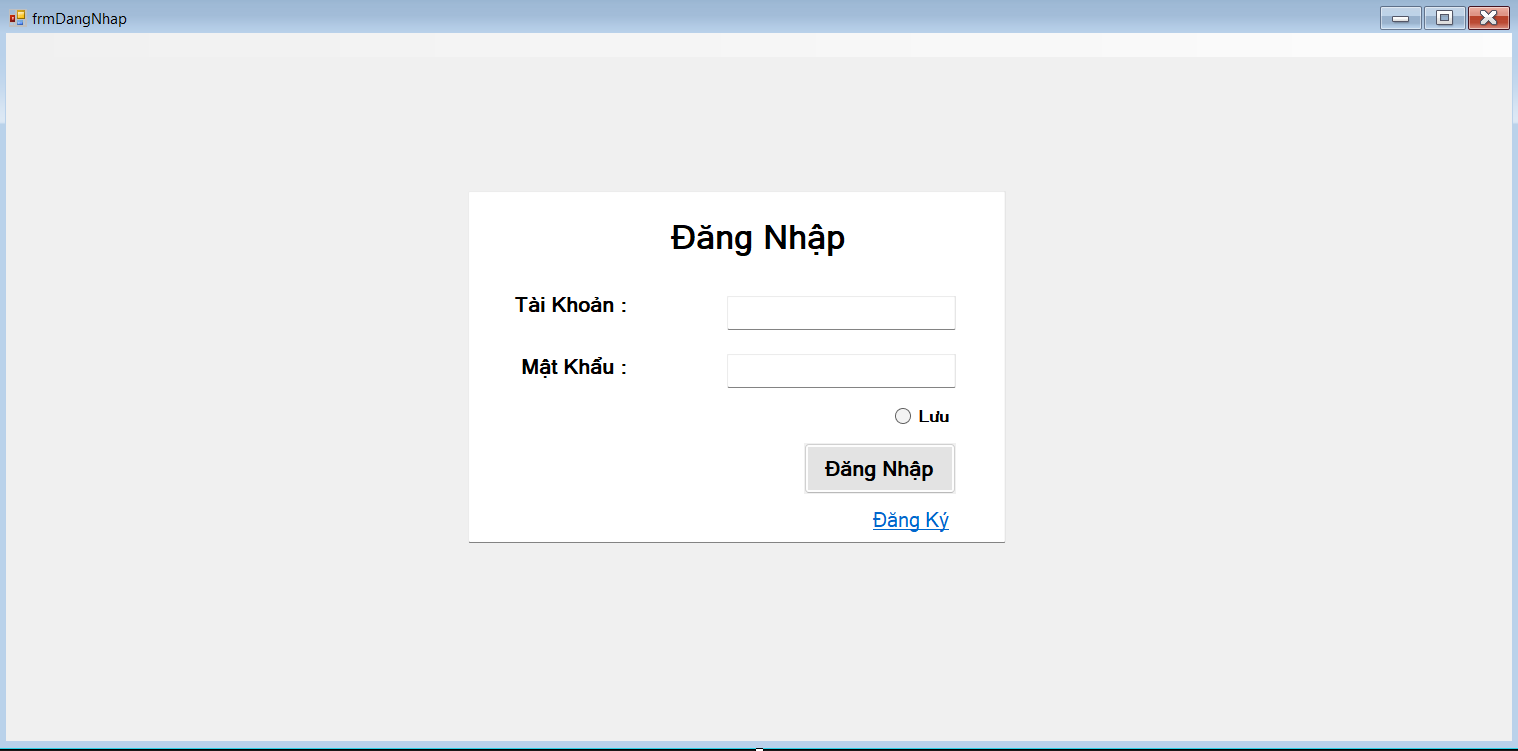
Thông qua việc phân chia rõ ràng vai trò của từng thành phần trong kiến trúc MVC, ứng dụng quản lý kết quả học tập không chỉ hoạt động hiệu quả mà còn dễ dàng bảo trì và mở rộng trong tương lai. Mỗi thành phần tương tác và hỗ trợ lẫn nhau, tạo ra một hệ thống quản lý chặt chẽ và thân thiện với người dùng.

## CHƯƠNG IV. THIẾT KẾ GIAO DIỆN

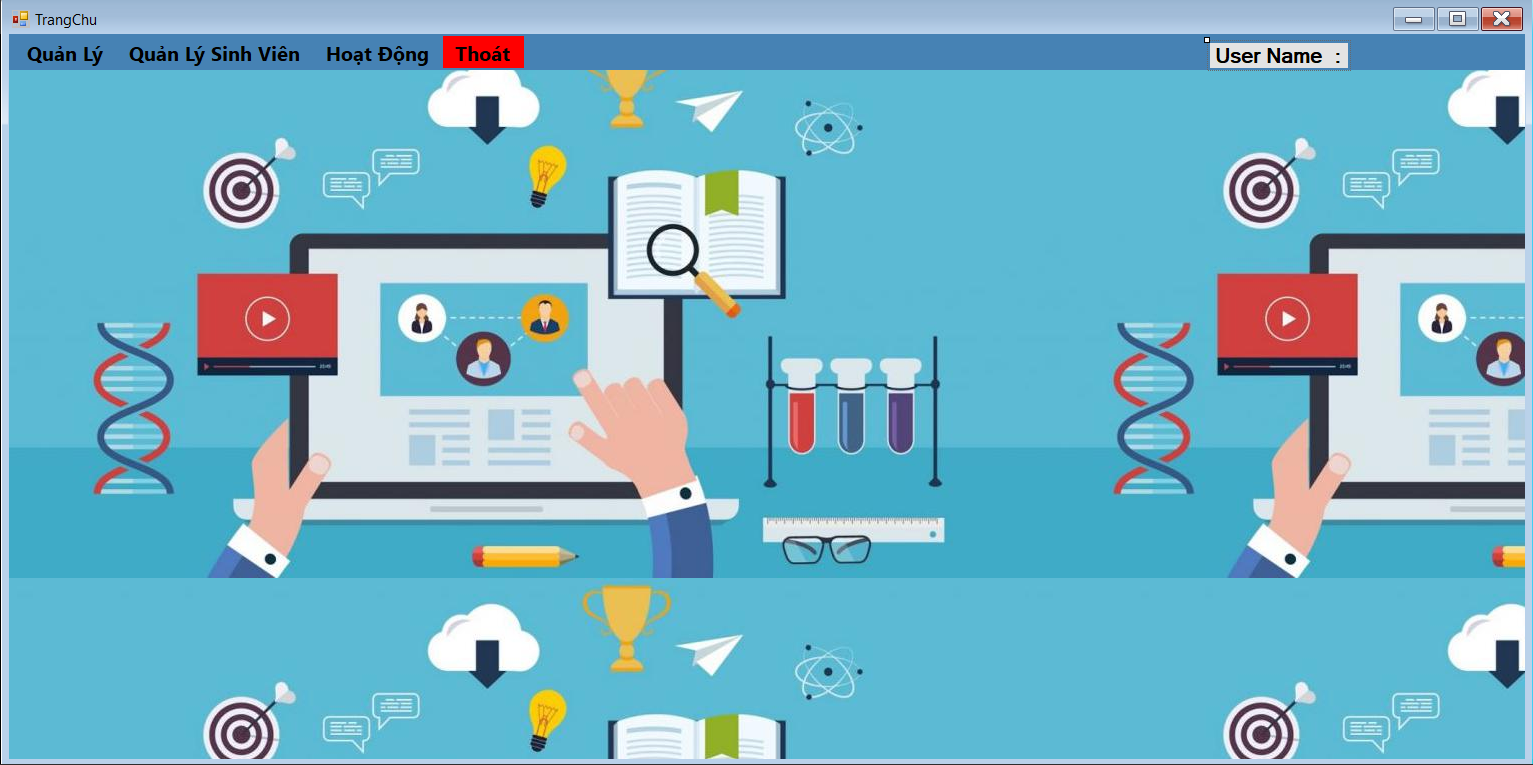
IV.1. Giao diện Đăng Ký



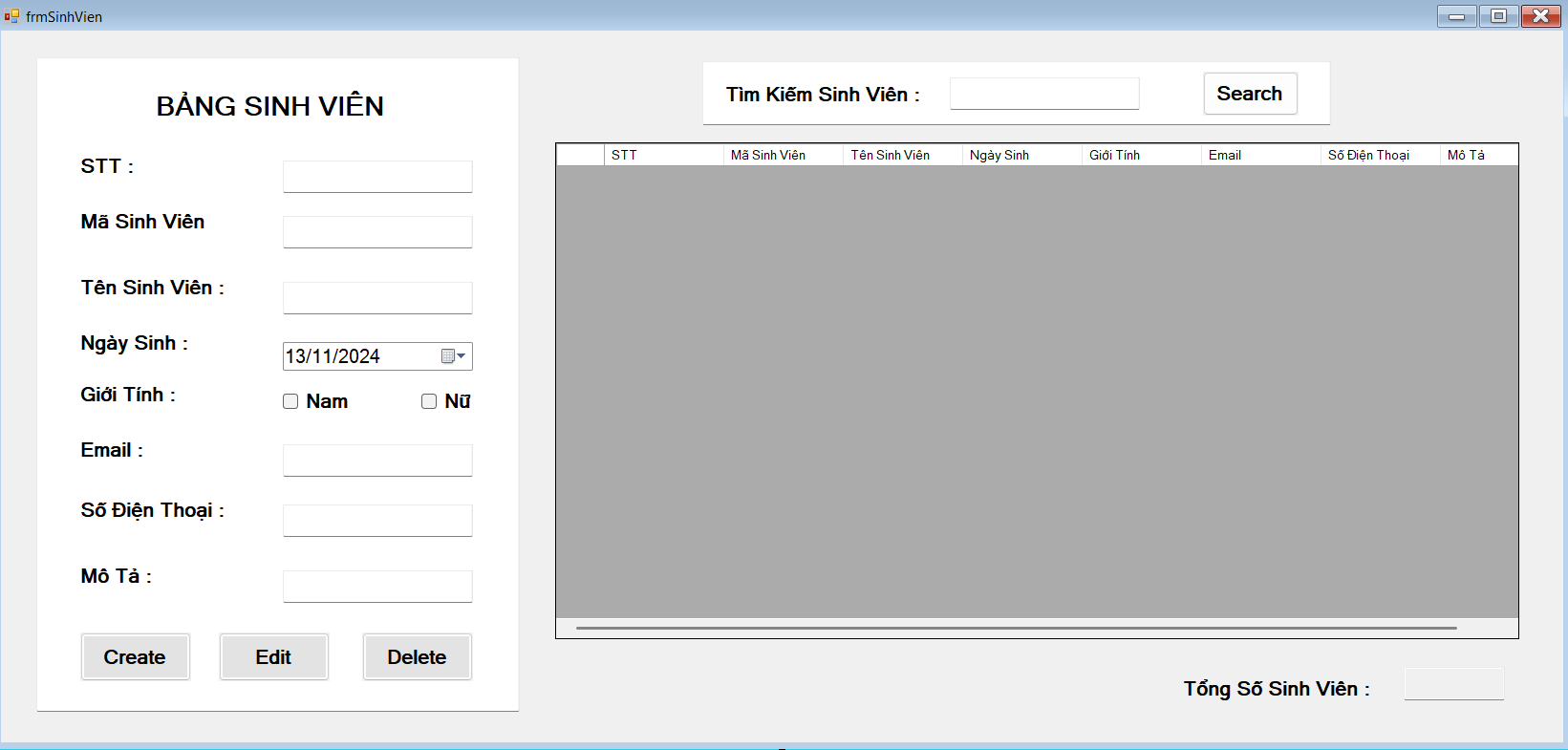
IV.2. Giao diện Đăng Nhập



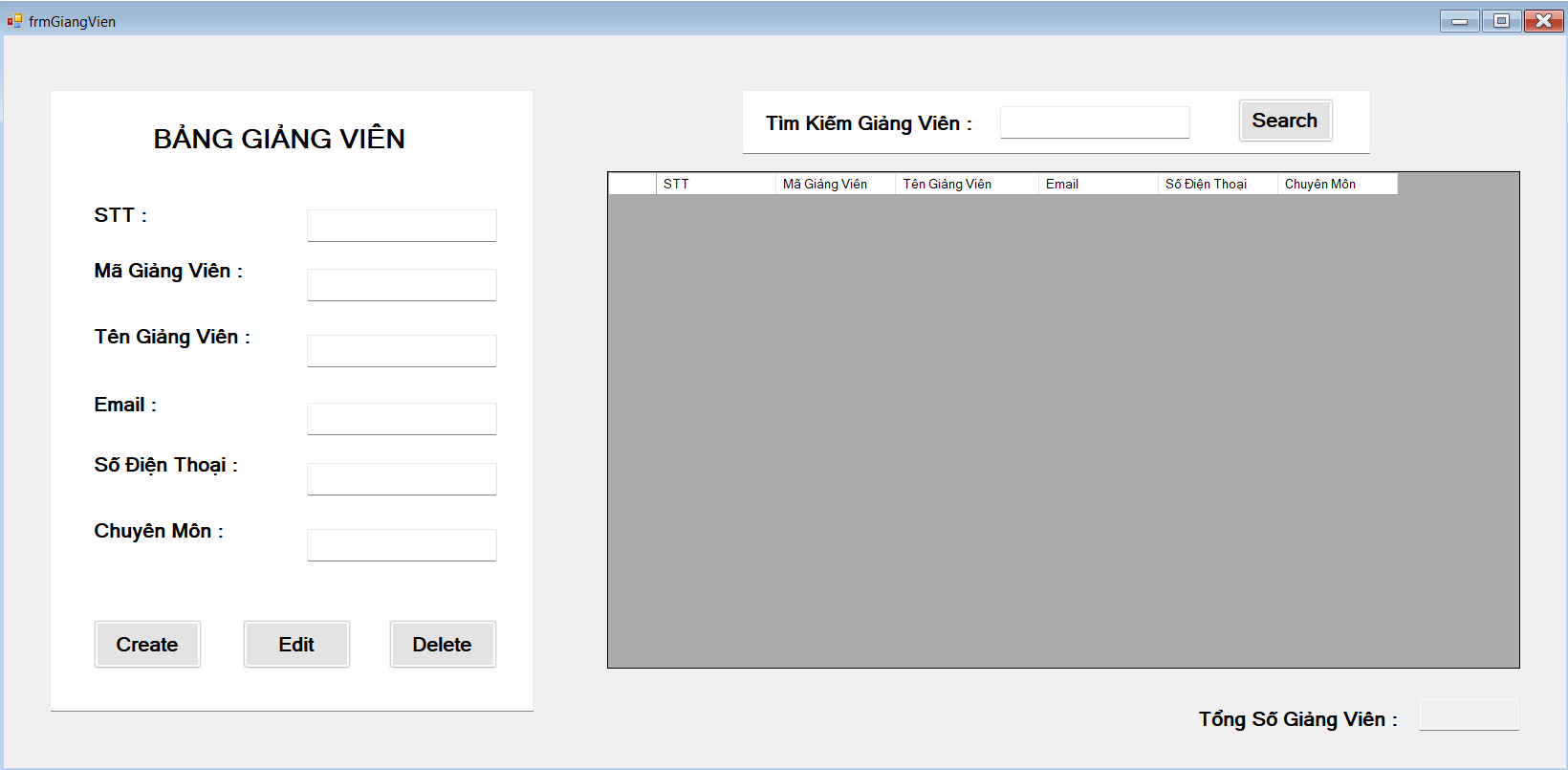
IV.3. Giao diện Trang Chủ



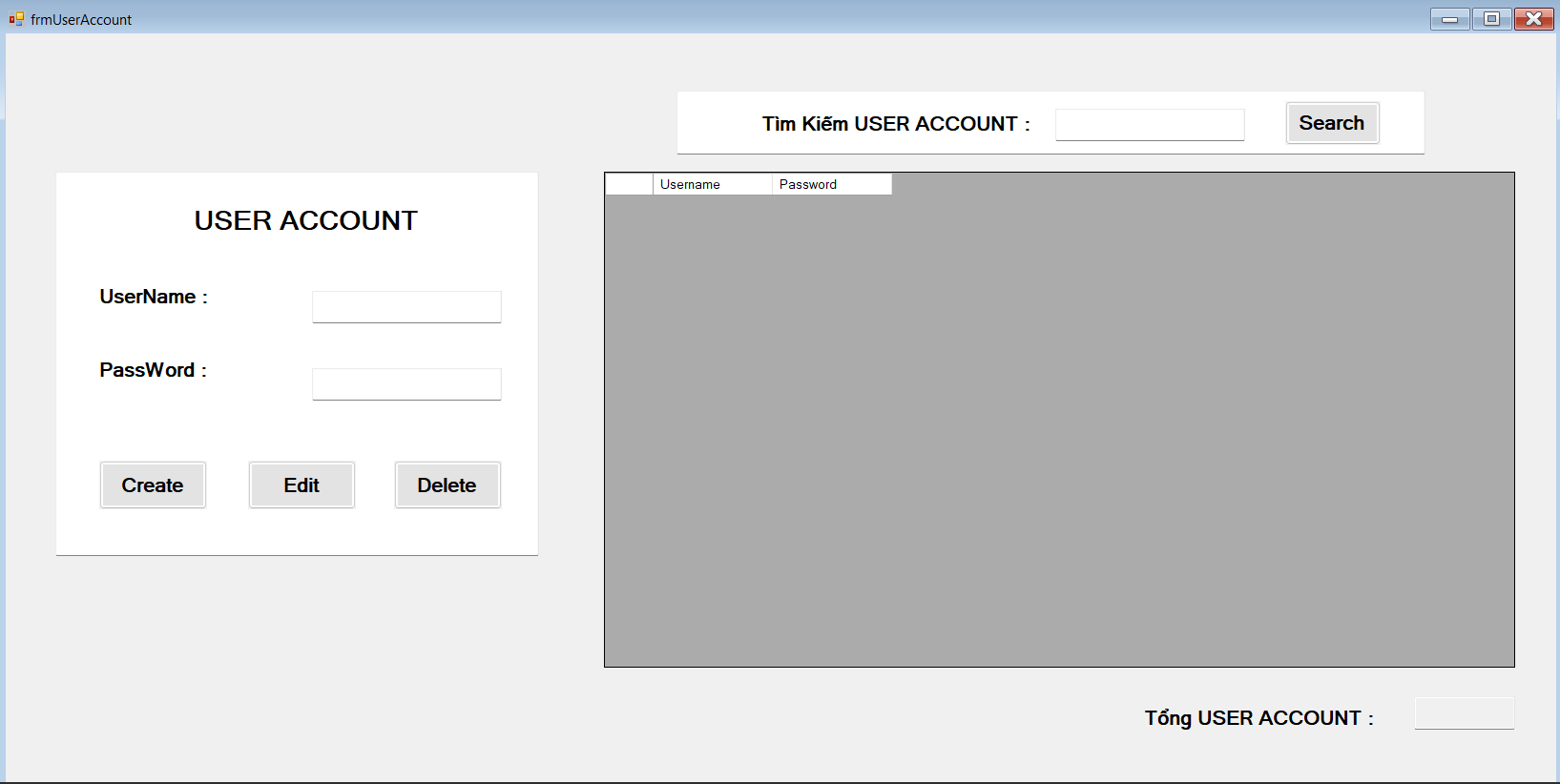
IV.4. Giao diện Bảng Sinh Viên



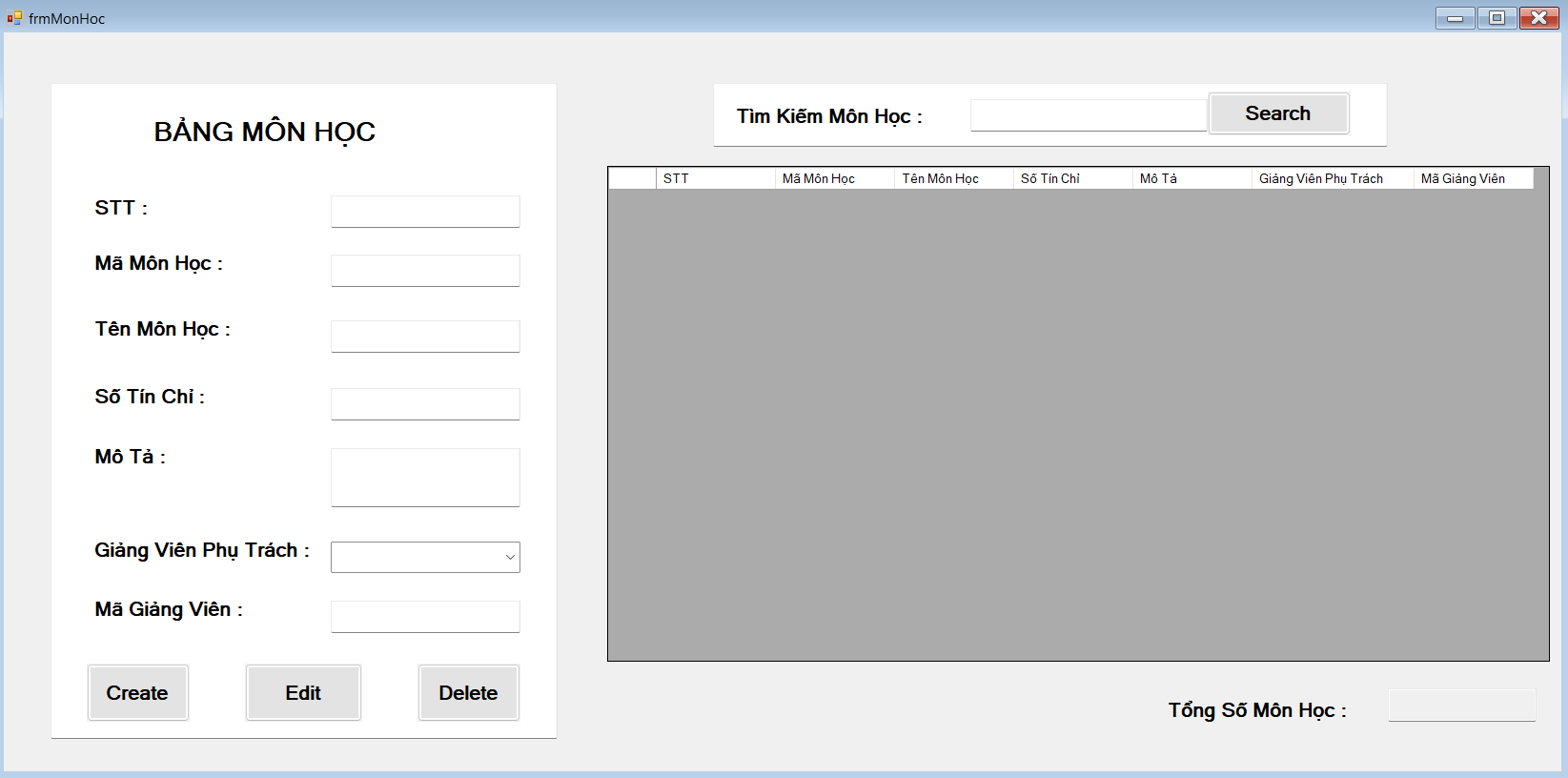
IV.5. Giao diện Bảng Giảng Viên



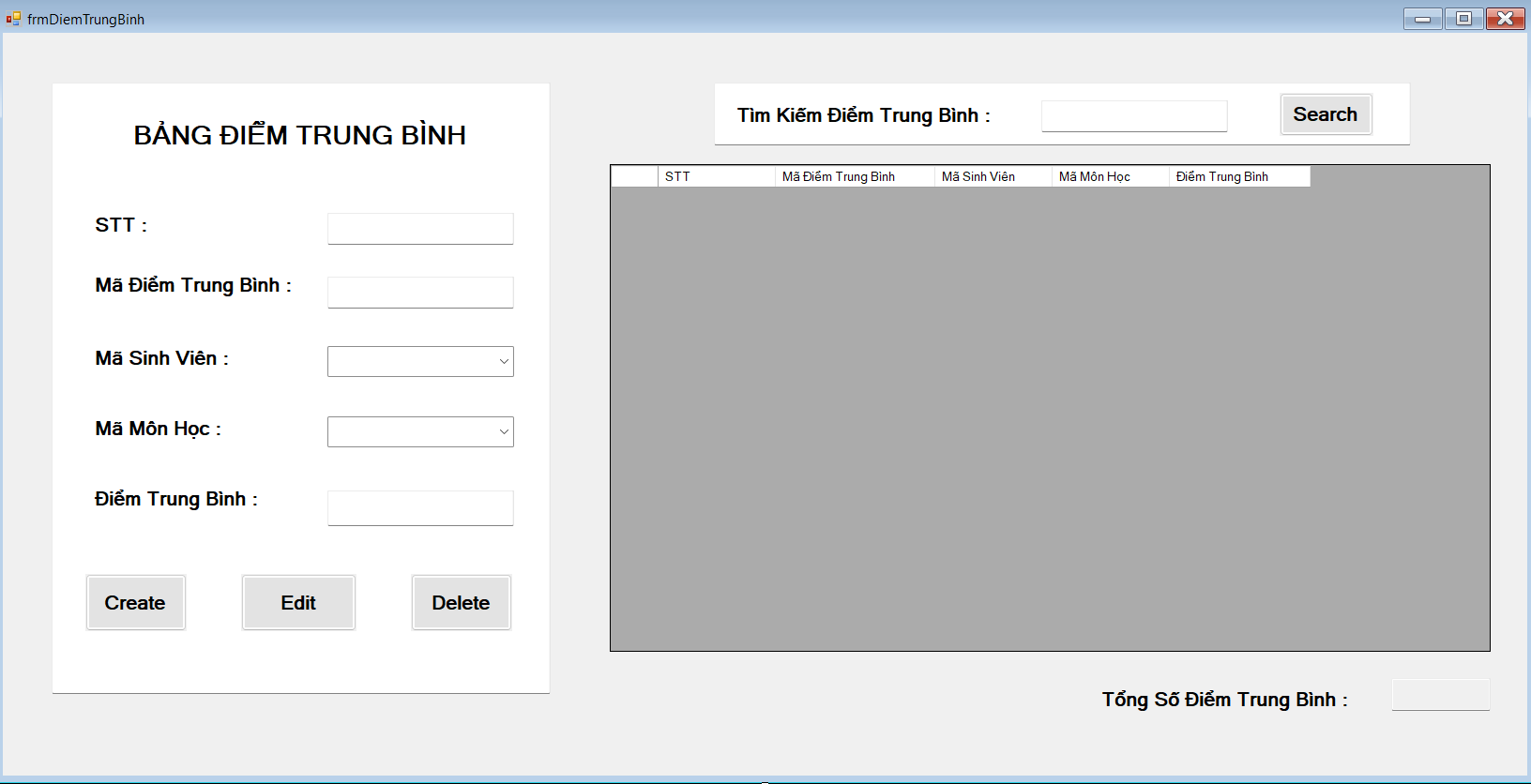
IV.6. Giao diện BẢNG USER ACCOUNT



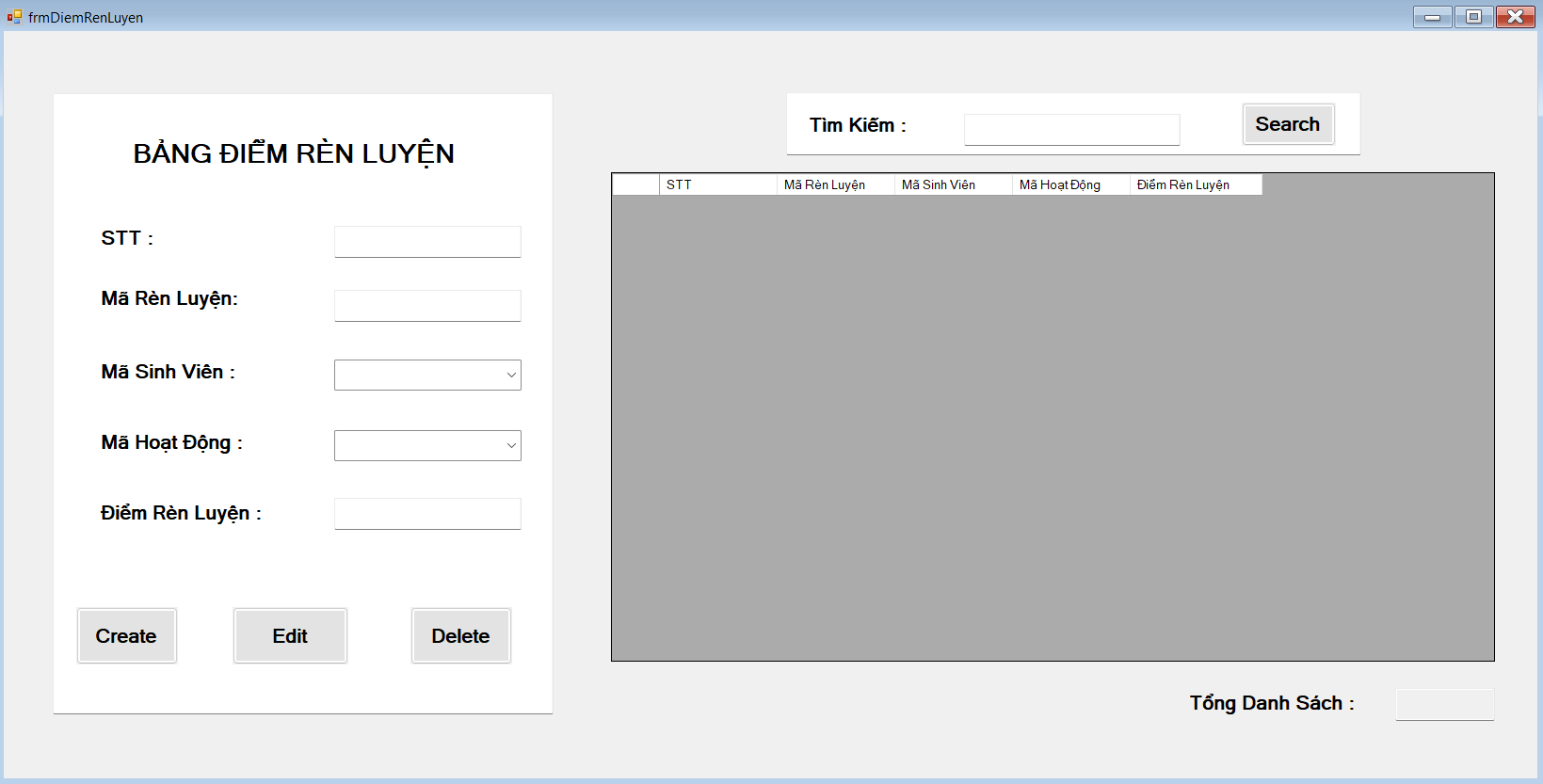
IV.7. Giao diện BẢNG MÔN HỌC



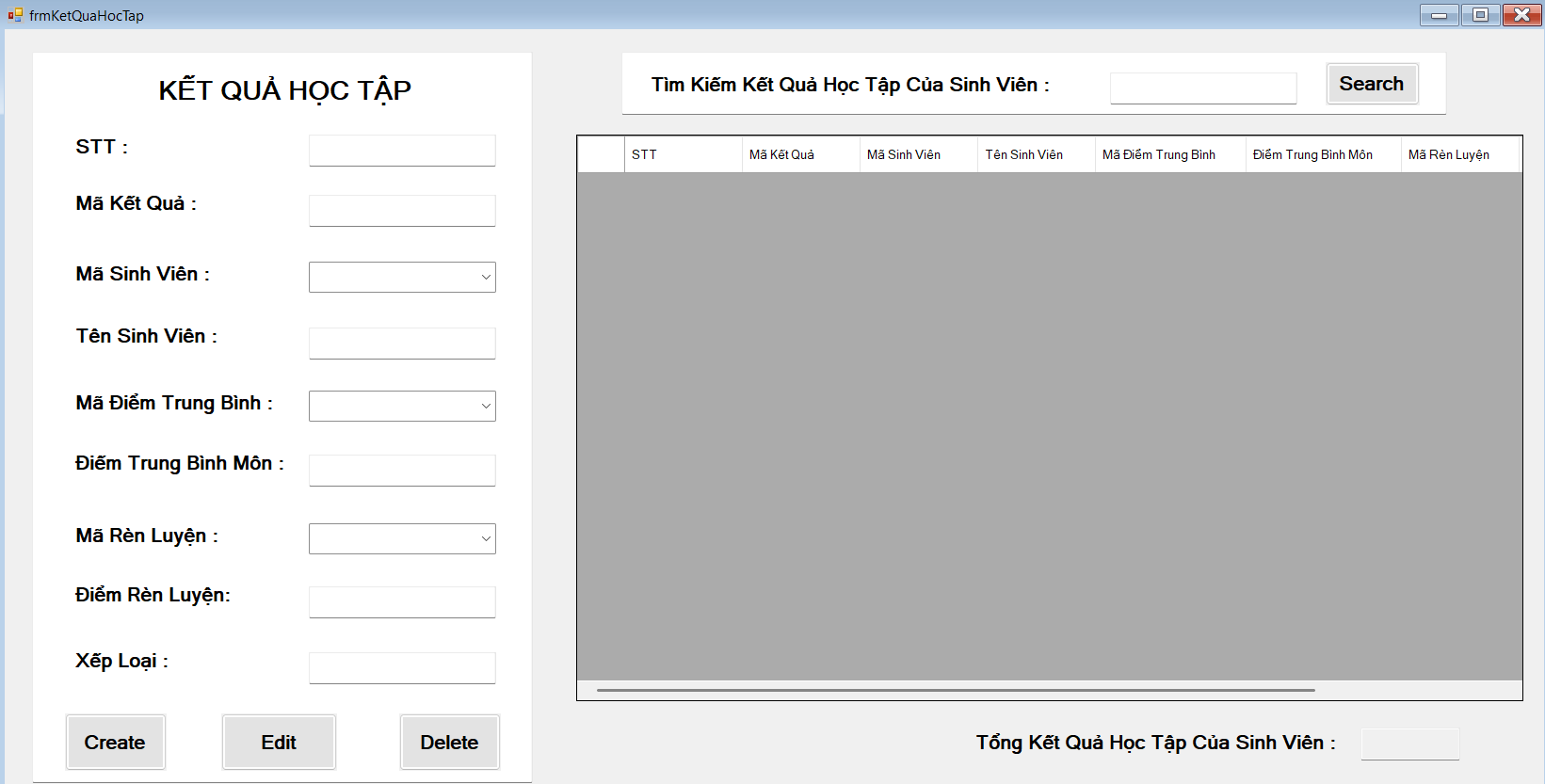
IV.8. Giao diện BẢNG ĐIỂM TRUNG BÌNH



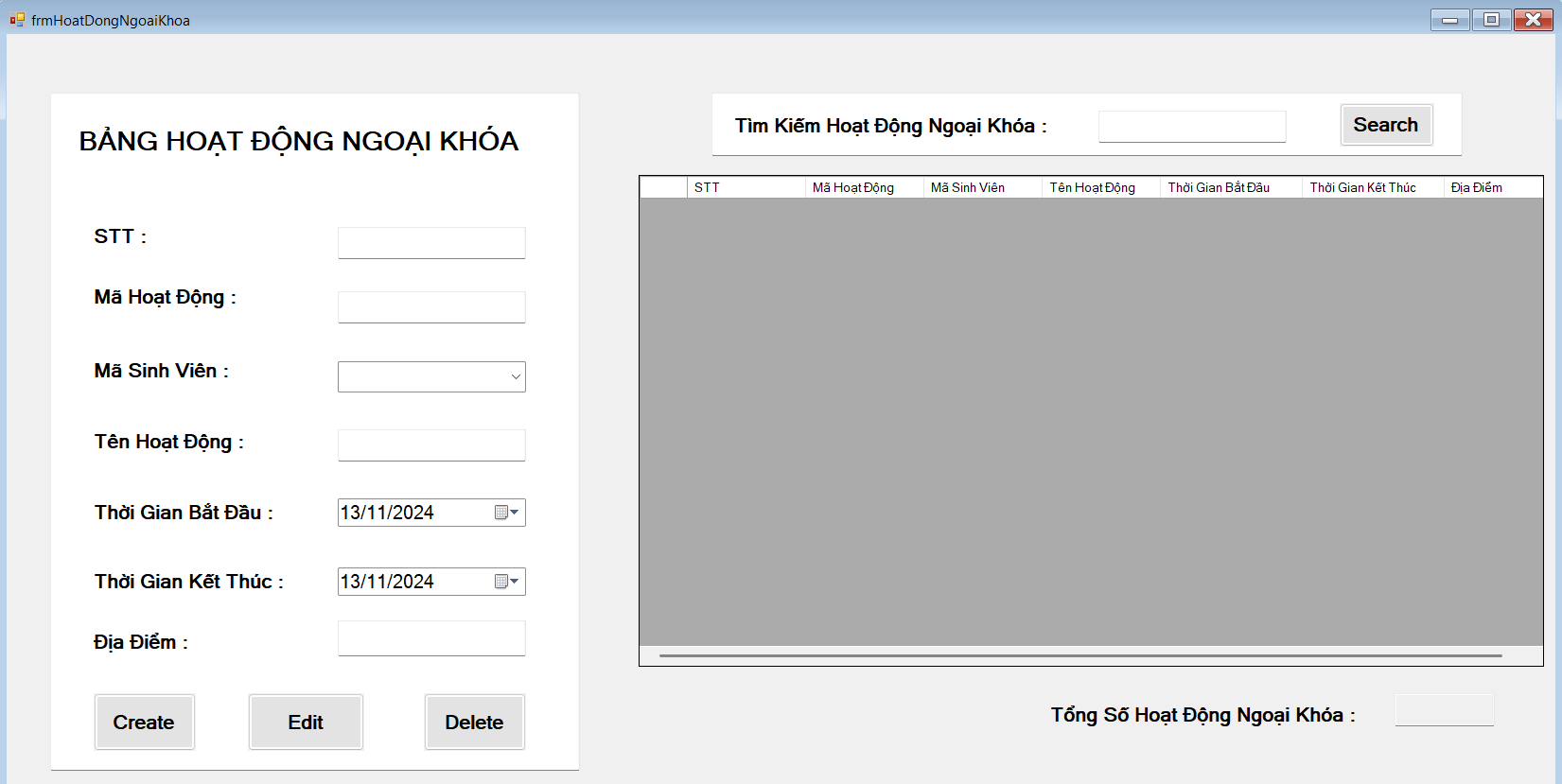
IV.9. Giao diện BẢNG ĐIỂM RÈN LUYỆN



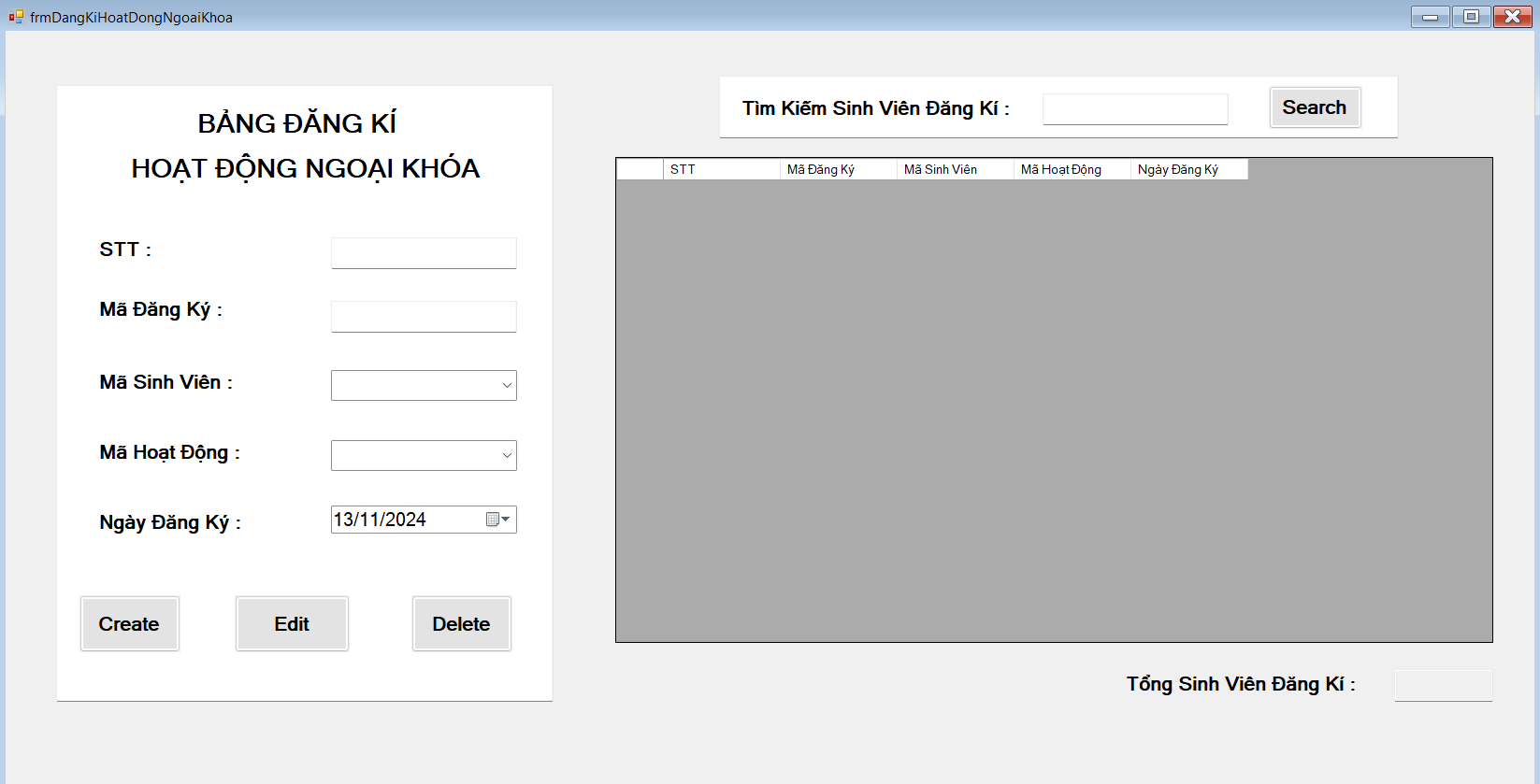
IV.10. Giao diện BẢNG KẾT QUẢ HOC TẬP



IV.11. Giao diện BẢNG HOẠT ĐỘNG NGOẠI KHÓA



IV.12. Giao diện BẢNG ĐĂNG KÍ BẢNG HOẠT ĐỘNG NGOẠI KHÓA



## CHƯƠNG V. TỔ CHỨC C# PROJECT

### V.1. Mô hình tổng thể

Mô hình tổng thể của ứng dụng quản lý kết quả học tập được thiết kế dựa trên kiến trúc MVC (Model-View-Controller), một mô hình thiết kế phần mềm phổ biến giúp tách biệt các thành phần và chức năng của ứng dụng. Mô hình này không chỉ cải thiện khả năng bảo trì mà còn tăng cường khả năng mở rộng của ứng dụng.

Các thành phần chính:

#### V.1.1. Model (Lớp dữ liệu)

Là phần lõi của ứng dụng, Model chịu trách nhiệm quản lý tất cả các dữ liệu và logic nghiệp vụ. Nó lưu trữ thông tin về sinh viên, giảng viên, môn học và điểm số.

Model thực hiện các tác vụ như thêm, sửa, xóa và truy xuất dữ liệu từ cơ sở dữ liệu. Các lớp trong Model bao gồm:

* SinhVien: Quản lý thông tin về sinh viên.
* GiangVien: Quản lý thông tin về giảng viên.
* MonHoc: Quản lý thông tin về môn học.
* HoatDongNgoaiKhoa: Quản lý các hoạt động ngoại khóa.
* DiemTrungBinh: Quản lý điểm trung bình của sinh viên.

#### V.1.2. View (Giao diện người dùng)

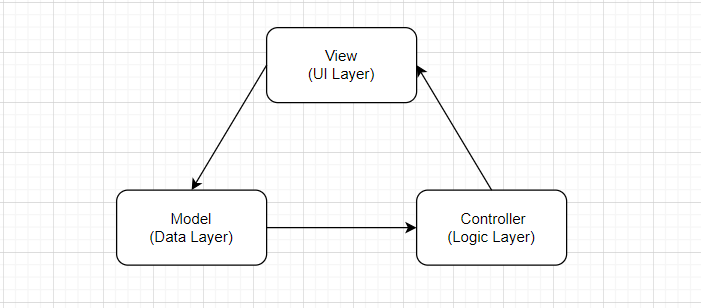
View là lớp giao diện người dùng, nơi người dùng tương tác với hệ thống. View cung cấp các form Windows Forms để người dùng có thể nhập liệu và xem thông tin.

Các form trong View được thiết kế trực quan và dễ sử dụng, giúp người dùng thực hiện các thao tác như quản lý sinh viên, giảng viên, môn học và theo dõi kết quả học tập một cách thuận tiện.

#### V.1.3. Controller (Lớp điều khiển)

Controller là cầu nối giữa Model và View. Nó nhận các sự kiện từ View (như nhấn nút) và xử lý các yêu cầu cần thiết bằng cách truy cập Model để thực hiện các thao tác dữ liệu.

Khi Model thay đổi, Controller sẽ thông báo cho View để cập nhật thông tin hiển thị, đảm bảo rằng người dùng luôn nhìn thấy dữ liệu mới nhất.



Mô tả sơ đồ:

* Mũi tên từ View đến Controller biểu thị rằng View gửi các sự kiện (như nhấn nút, nhập liệu) đến Controller.
* Mũi tên từ Controller đến Model thể hiện rằng Controller truy cập dữ liệu và logic nghiệp vụ từ Model.
* Mũi tên ngược từ Model đến Controller cho thấy rằng Controller nhận thông tin từ Model để cập nhật View.
* Cuối cùng, mũi tên từ Controller đến View chỉ ra rằng Controller điều khiển việc cập nhật giao diện người dùng dựa trên dữ liệu từ Model.

### V.2. Cách tổ chức các thành phần trong project

Trong project này, các thành phần được tổ chức một cách có cấu trúc nhằm đảm bảo tính dễ bảo trì và mở rộng. Việc tổ chức hợp lý giúp cho việc phát triển và quản lý mã nguồn trở nên thuận tiện hơn. Dưới đây là cách tổ chức các thành phần chính trong ứng dụng:

#### V.2.1. Thành phần Model

Chứa các lớp:

SinhVien: Quản lý thông tin sinh viên, bao gồm các thuộc tính như mã sinh viên, tên, ngày sinh, giới tính và các phương thức liên quan đến sinh viên.

GiangVien: Quản lý thông tin giảng viên, với các thuộc tính như mã giảng viên, tên, chuyên môn và các phương thức xử lý thông tin giảng viên.

MonHoc: Quản lý thông tin môn học, bao gồm mã môn học, tên môn học, số tín chỉ và các phương thức liên quan đến môn học.

HoatDongNgoaiKhoa: Quản lý các hoạt động ngoại khóa mà sinh viên tham gia, với các thuộc tính như mã hoạt động, tên hoạt động, mô tả và phương thức xử lý.

DiemTrungBinh: Quản lý điểm trung bình của sinh viên, bao gồm các thuộc tính và phương thức để tính toán điểm trung bình.

#### V.2.2. Thành phần View

Chứa các form giao diện:

MainForm: Form chính của ứng dụng, nơi hiển thị tổng quan các chức năng và thông tin liên quan.

StudentForm: Form để quản lý sinh viên, cho phép người dùng thêm, sửa, xóa và xem thông tin sinh viên.

TeacherForm: Form để quản lý giảng viên, với các chức năng tương tự như StudentForm.

CourseForm: Form để quản lý môn học, cho phép người dùng quản lý thông tin các môn học.

ReportForm: Form để xem báo cáo kết quả học tập, hiển thị thông tin điểm số và các thống kê liên quan.

#### V.2.3. Thành phần Controller

Chứa các lớp điều khiển:

StudentController: Xử lý các thao tác liên quan đến sinh viên, điều khiển việc thêm, sửa, xóa và truy xuất thông tin sinh viên.

TeacherController: Xử lý các thao tác liên quan đến giảng viên, tương tự như StudentController.

CourseController: Xử lý các thao tác liên quan đến môn học, quản lý thông tin và tương tác với Model.

ReportController: Xử lý việc tạo báo cáo kết quả học tập, thu thập dữ liệu từ Model và cung cấp cho View để hiển thị.

### V.3. Vai trò của từng thành phần trong project

Mỗi thành phần trong project đảm nhận các chức năng và vai trò cụ thể, đảm bảo cho ứng dụng hoạt động hiệu quả và linh hoạt.

#### V.3.1. Thành phần Model

Vai trò: Thành phần Model chịu trách nhiệm quản lý tất cả dữ liệu và logic nghiệp vụ của ứng dụng. Nó thực hiện các thao tác truy cập dữ liệu từ cơ sở dữ liệu, đảm bảo tính chính xác và toàn vẹn dữ liệu.

Quản lý dữ liệu: Model chứa các lớp đại diện cho các thực thể trong hệ thống như SinhVien, GiangVien, MonHoc, HoatDongNgoaiKhoa, và DiemTrungBinh. Mỗi lớp này bao gồm các thuộc tính và phương thức cần thiết để lưu trữ và xử lý thông tin tương ứng.

Logic nghiệp vụ: Model cũng định nghĩa các quy tắc nghiệp vụ và thực hiện các phép toán cần thiết, như tính điểm trung bình của sinh viên hay kiểm tra tính hợp lệ của dữ liệu đầu vào.

Tương tác với cơ sở dữ liệu: Model sử dụng các công nghệ như ADO.NET hoặc Entity Framework để thực hiện các truy vấn cơ sở dữ liệu, đảm bảo mọi thay đổi trong dữ liệu đều được cập nhật kịp thời và nhất quán.

#### V.3.2. Thành phần View

Vai trò: Thành phần View cung cấp giao diện người dùng, nơi người dùng tương tác với hệ thống và hiển thị thông tin một cách rõ ràng và dễ hiểu.

Giao diện thân thiện: View được thiết kế để tạo ra trải nghiệm người dùng tốt nhất, với các form và điều khiển trực quan giúp người dùng dễ dàng thao tác và theo dõi thông tin.

Phân chia chức năng: Mỗi form trong View phục vụ cho từng chức năng cụ thể của ứng dụng, như quản lý sinh viên, giảng viên, môn học và báo cáo kết quả học tập, từ đó giúp người dùng dễ dàng tìm kiếm và truy cập thông tin cần thiết.

Hiển thị dữ liệu: View đảm nhận việc hiển thị dữ liệu từ Model một cách rõ ràng, sử dụng các bảng, biểu đồ và thông báo để cung cấp thông tin cho người dùng.

#### V.3.3. Thành phần Controller

Vai trò: Thành phần Controller là cầu nối giữa Model và View, điều khiển luồng thông tin giữa hai thành phần này.

Xử lý sự kiện: Controller nhận các sự kiện từ View, như nút nhấn hoặc thay đổi dữ liệu, và thực hiện các yêu cầu cần thiết trên Model. Ví dụ, khi người dùng yêu cầu thêm một sinh viên mới, Controller sẽ nhận thông tin từ View và gọi phương thức tương ứng trong Model.

Cập nhật dữ liệu: Sau khi Model thay đổi (như thêm, sửa, hoặc xóa thông tin), Controller sẽ thông báo cho View để cập nhật giao diện người dùng, đảm bảo người dùng luôn nhìn thấy thông tin mới nhất.

Logic nghiệp vụ: Controller cũng có thể chứa logic nghiệp vụ đơn giản để xử lý các yêu cầu trước khi tương tác với Model, giúp giữ cho Model và View tách biệt và dễ dàng quản lý.

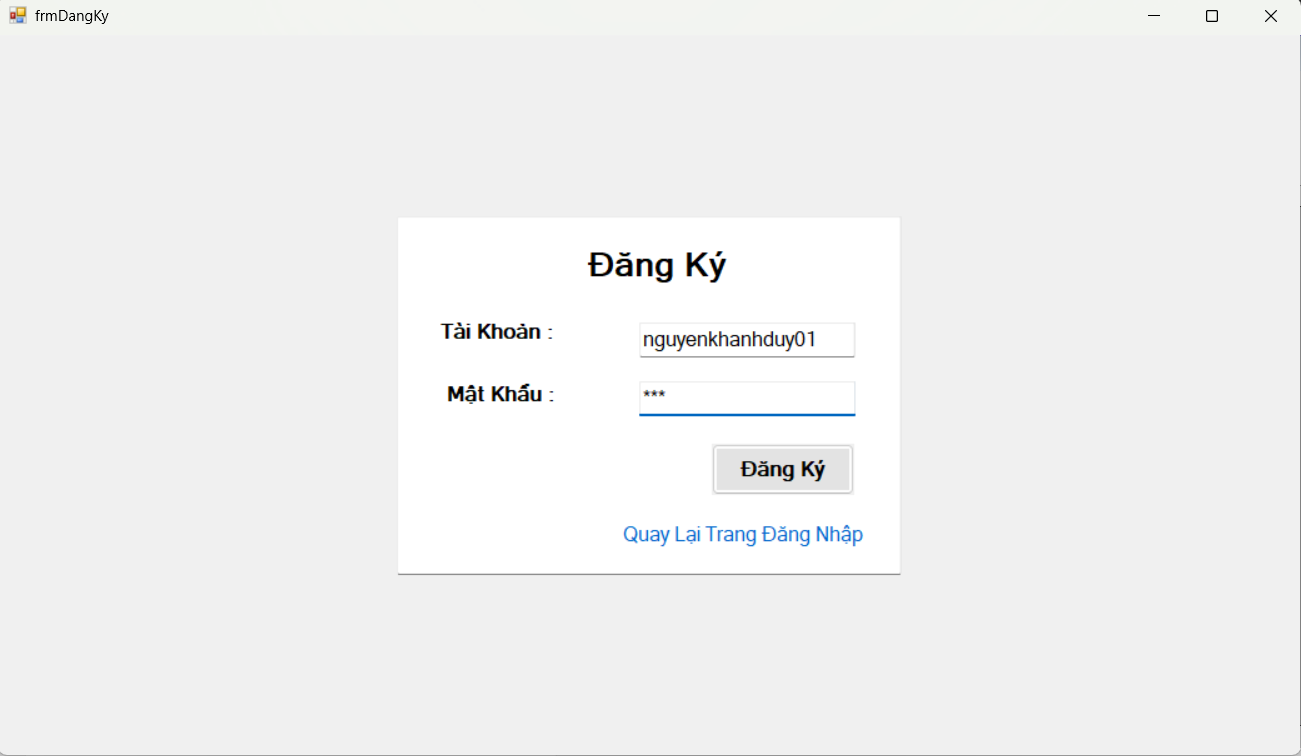
### V.4. Class diagram

## CHƯƠNG VI. KẾT QUẢ THỰC HIỆN

### VI.1. Kết quả thực hiện chức năng ĐĂNG KÝ

**Đăng Ký** - Tài Khoản : nguyenkhanhduy01

- Mật Khẩu : 123

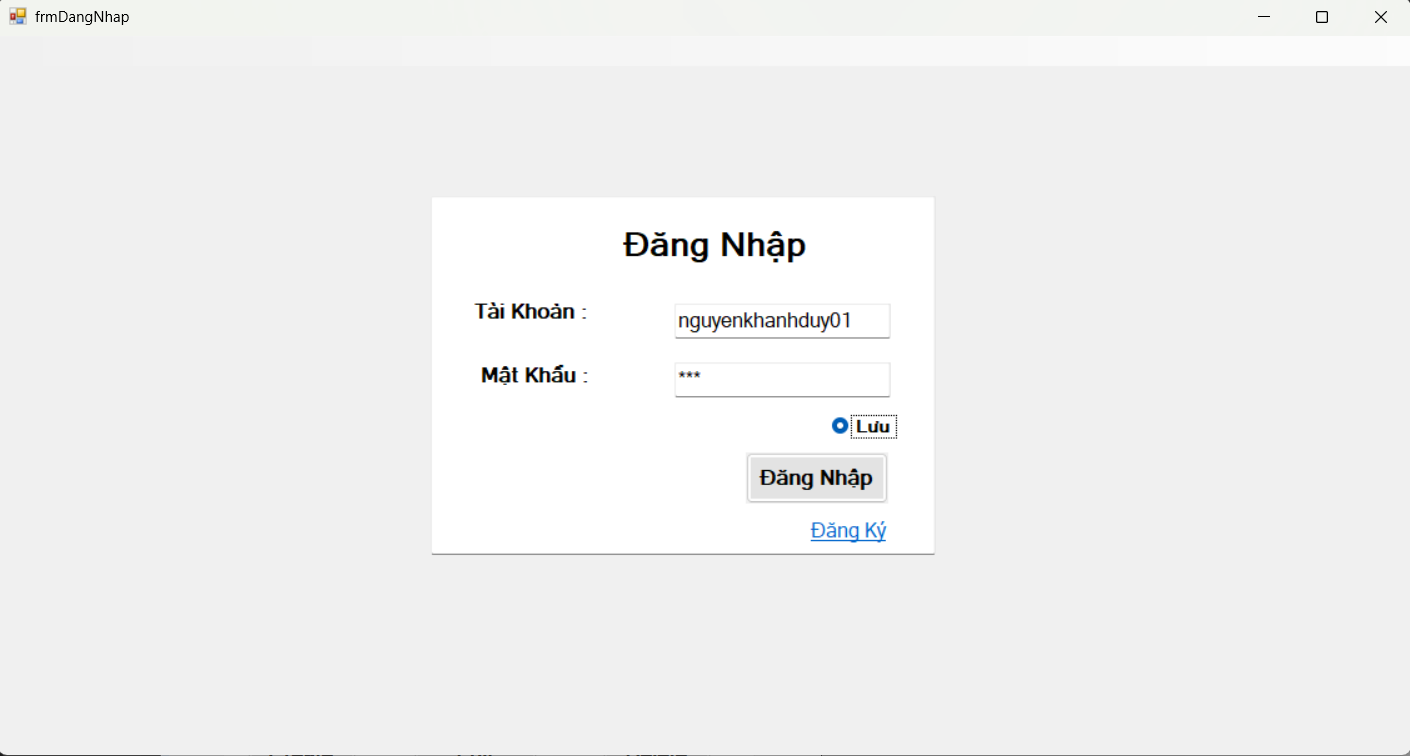


### VI.2. Kết quả thực hiện chức năng ĐĂNG NHẬP

**Đăng Nhập** - Tài Khoản : nguyenkhanhduy01

- Mật Khẩu : 123

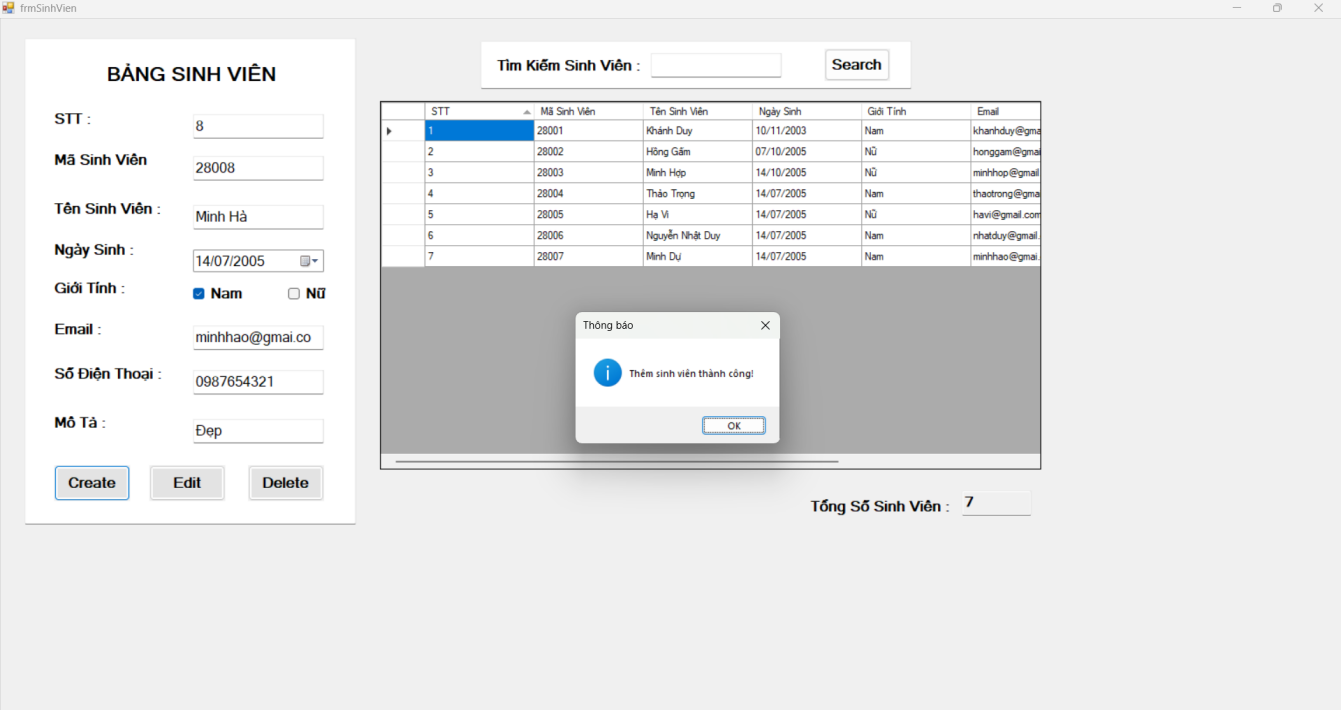
**Lưu** Tài Khoản , Mật Khẩu

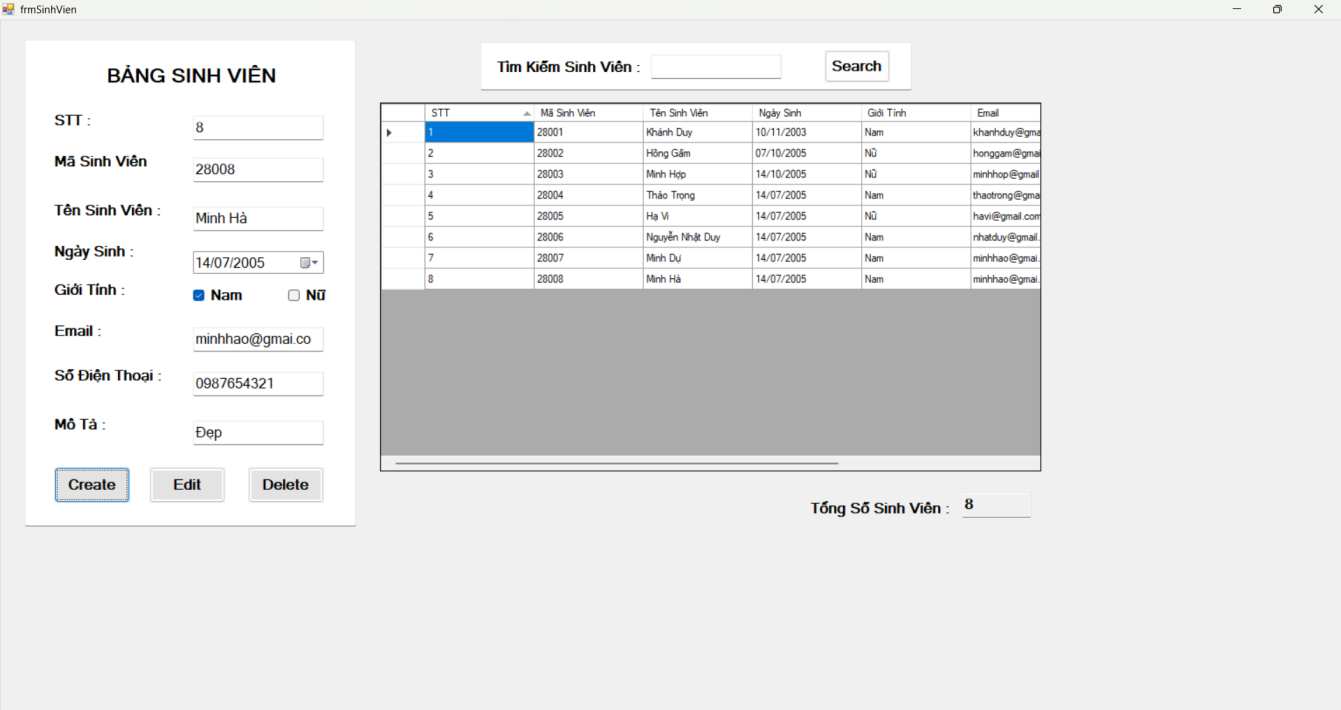


**Giao diện Trang Chủ** hiển thị tên **Tài Khoản : nguyenkhanhduy01**

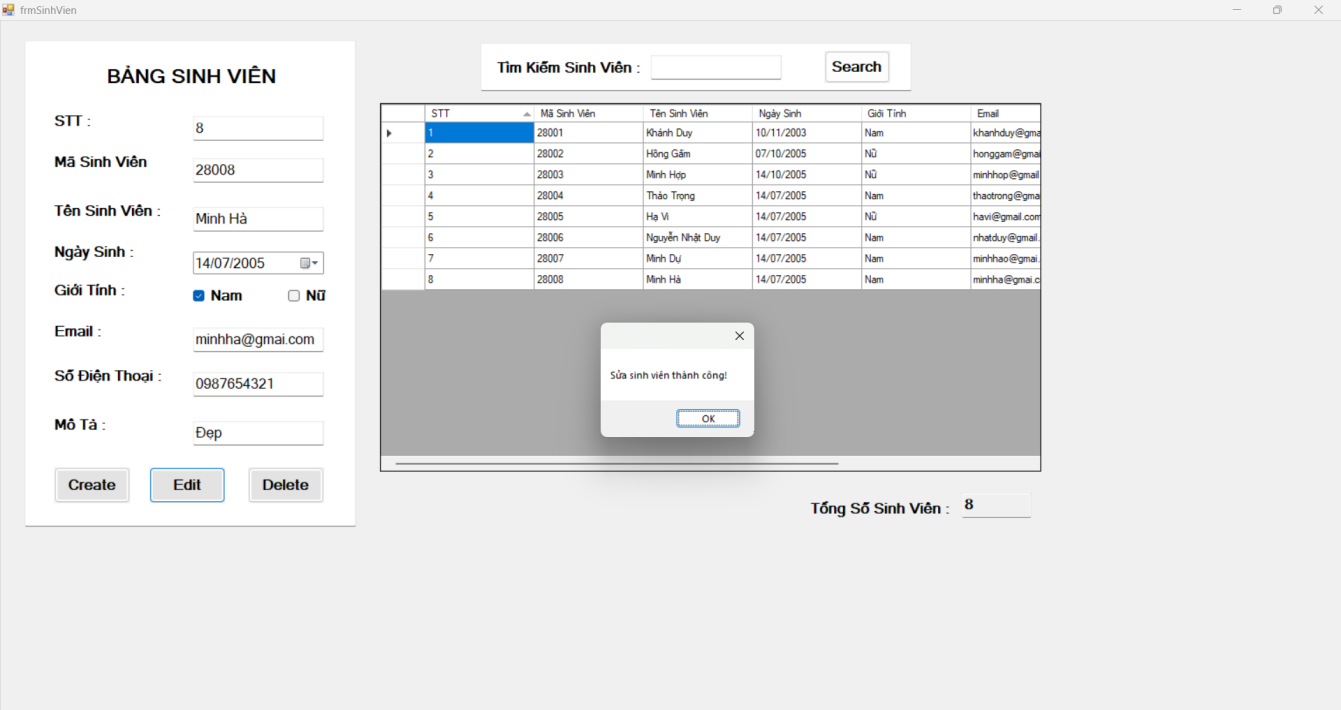


### VI.3. Kết quả thực hiện chức năng Create



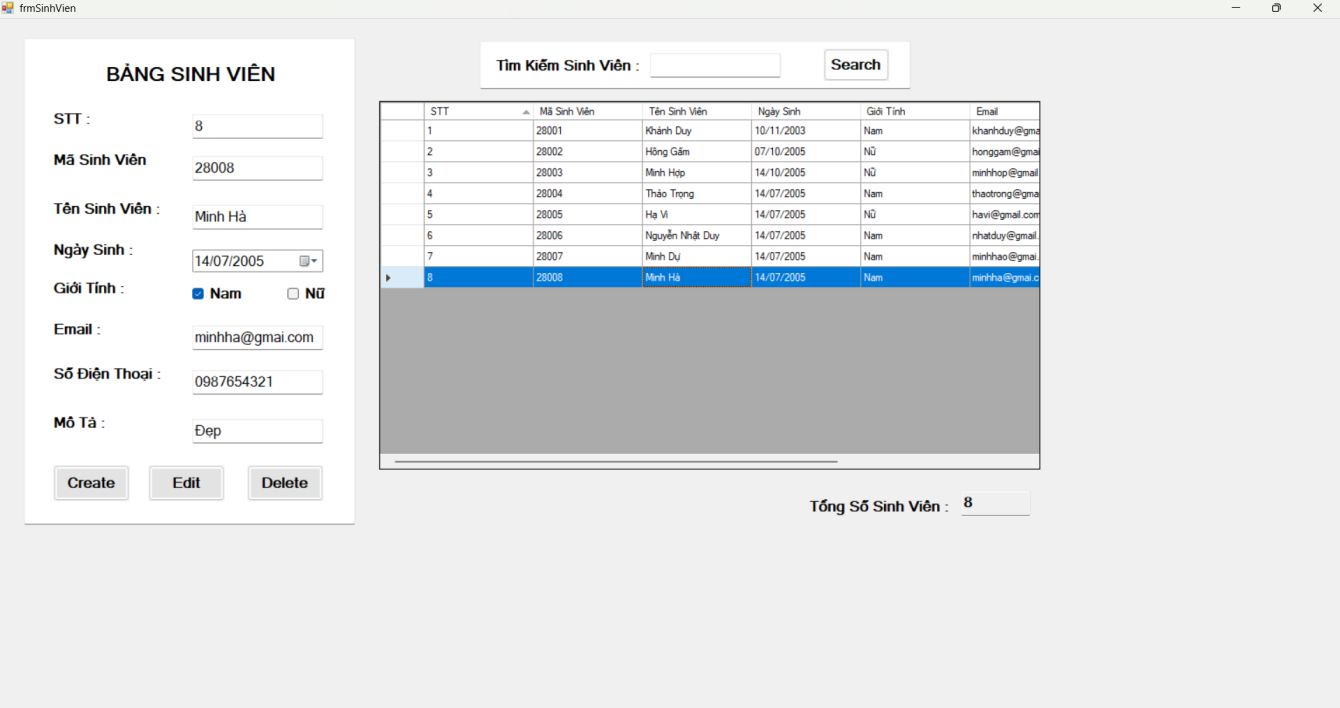


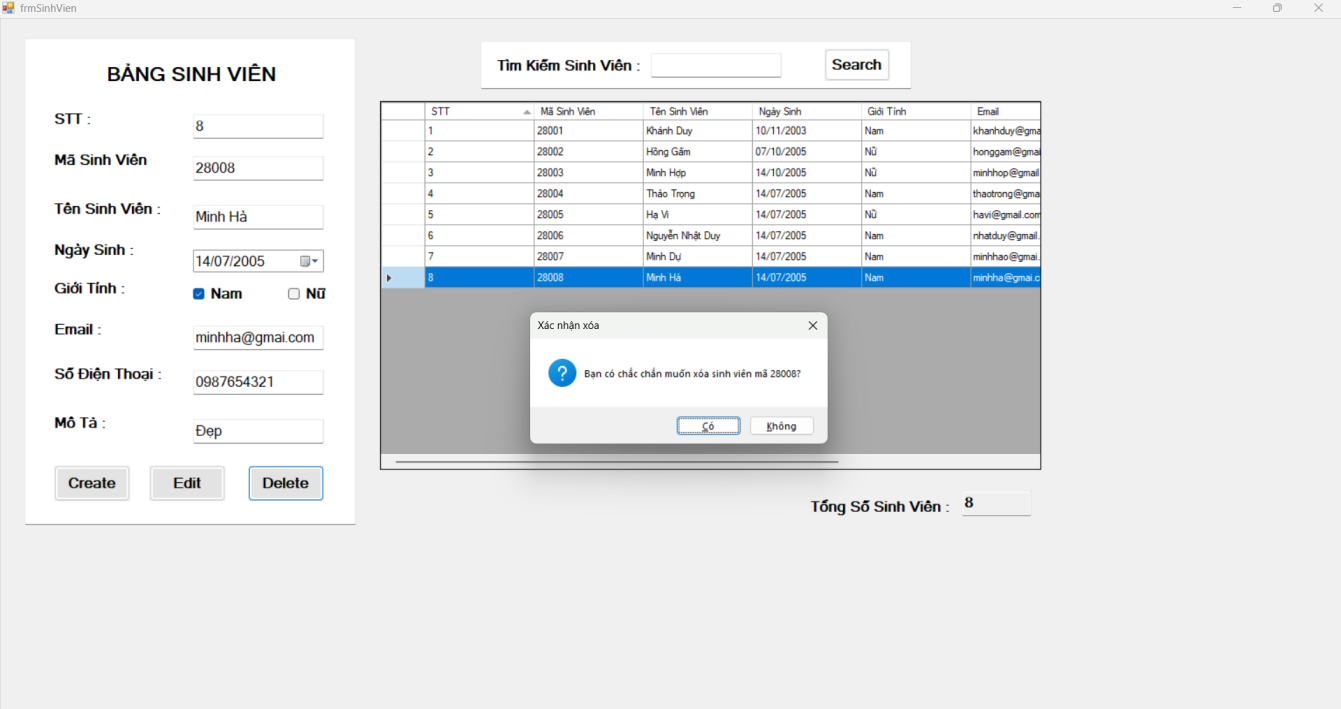
### VI.4. Kết quả thực hiện chức năng Edit

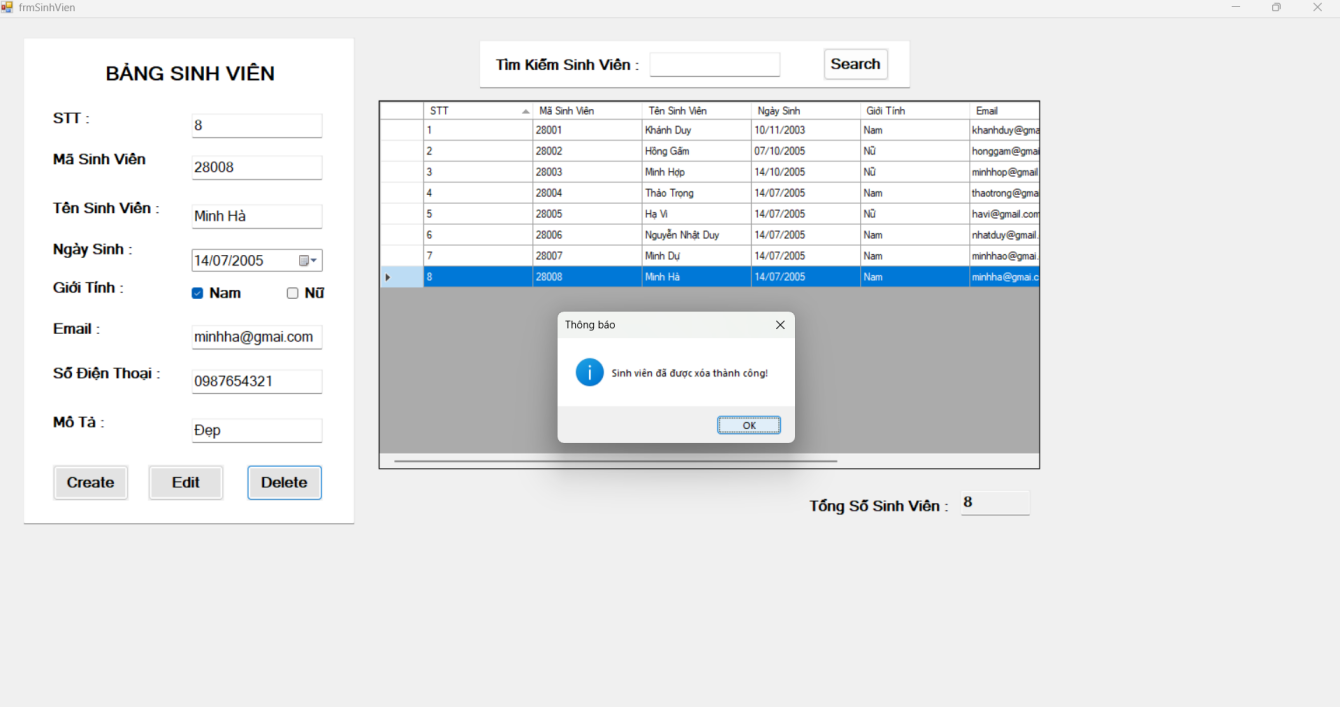


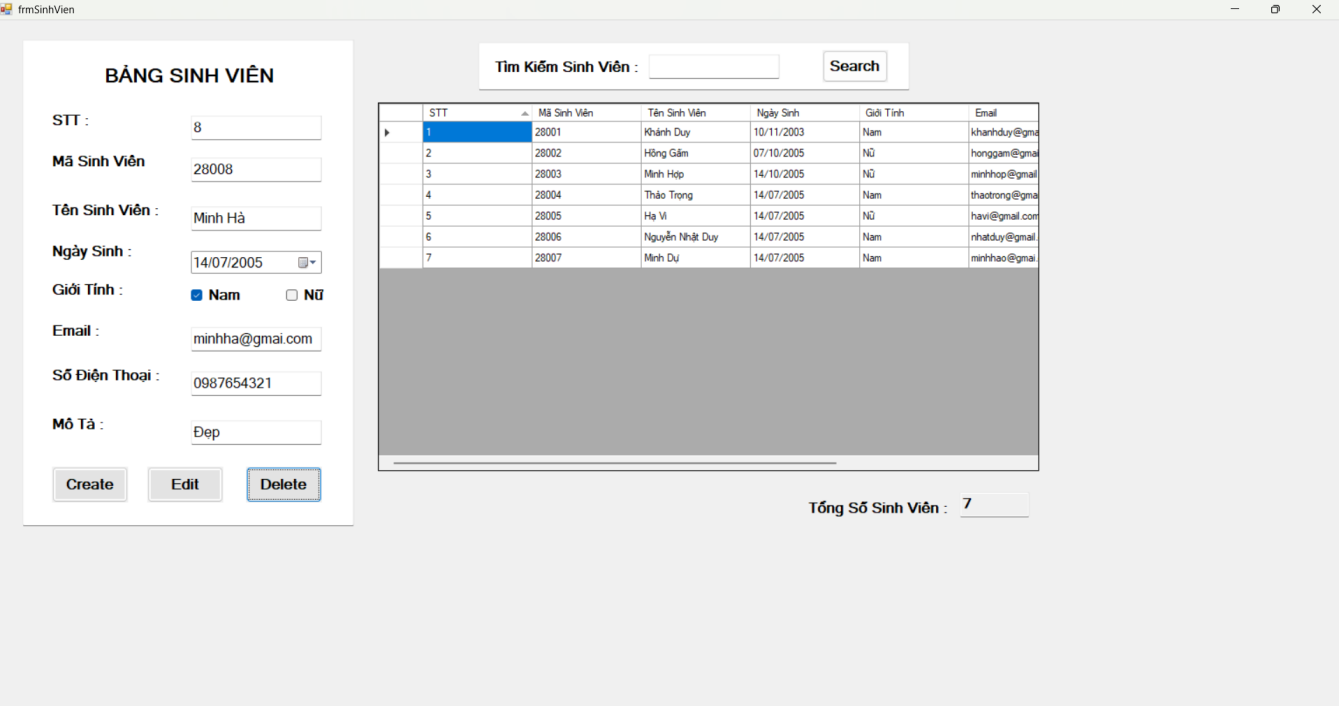
### VI.5. Kết quả thực hiện chức năng Delete

Chọn hàng chứa sinh viên cần xóa



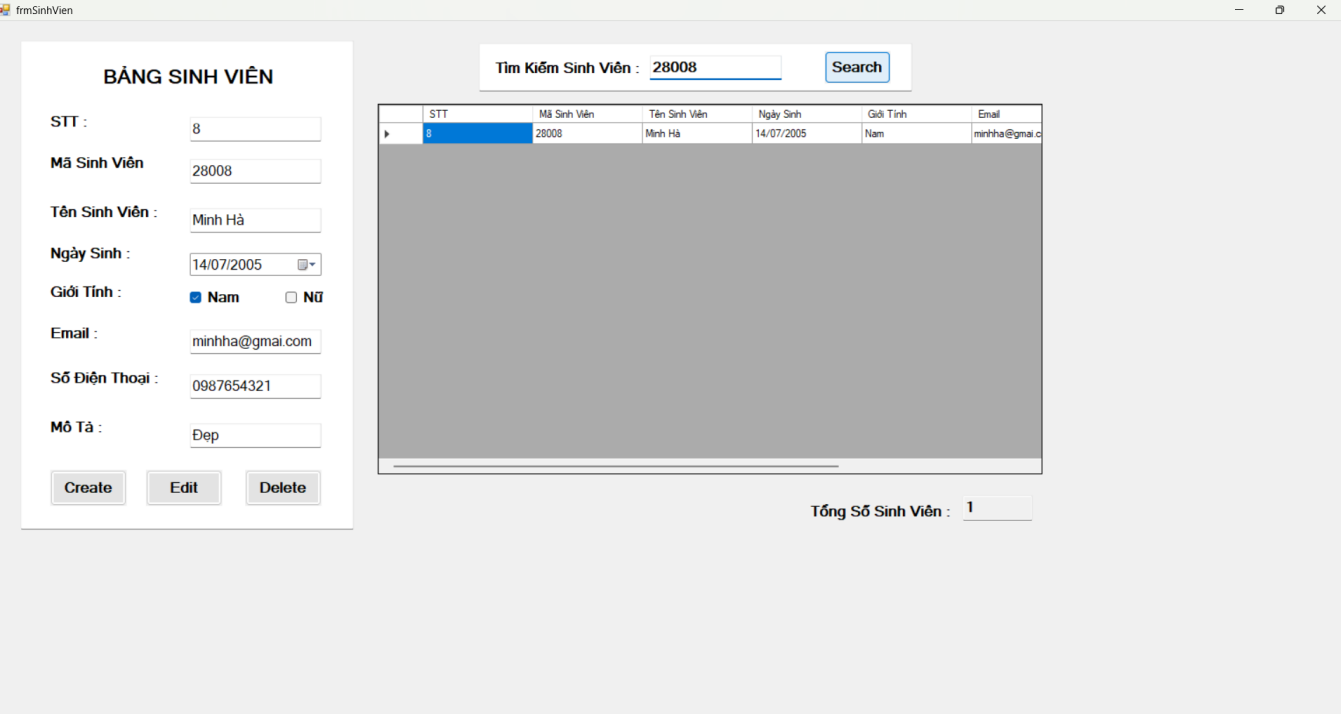




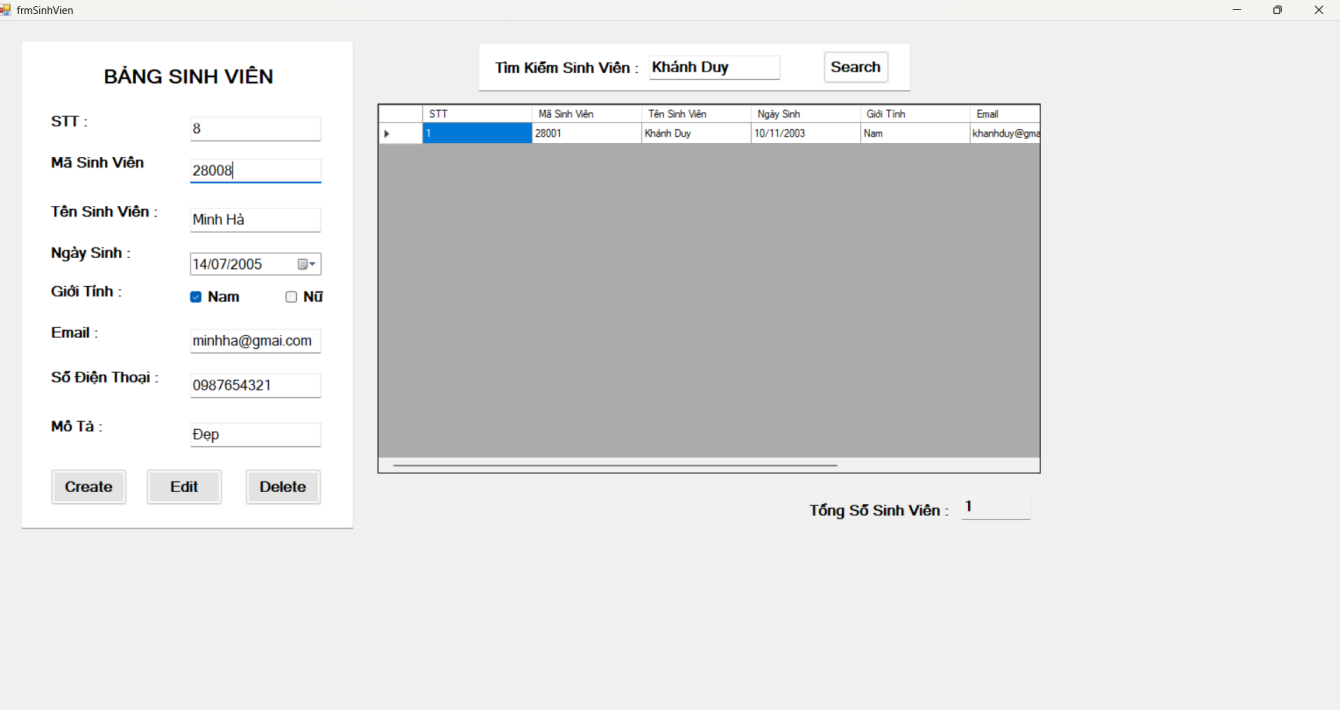


### VI.6. Kết quả thực hiện chức năng Search

Tìm kiếm theo Mã Sinh Viên



Tìm kiếm theo Tên Sinh Viên



### VI.8. Kết quả thực hiện chức năng

### VI.9. Kết quả thực hiện chức năng

### VI.10. Kết quả thực hiện chức năng

## CHƯƠNG VII. KẾT LUẬN

### VII.1. Kết quả đạt được

### VII.2. Hạn chế

## CHƯƠNG VIII. TÀI LIỆU THAM KHẢO