**MỤC LỤC**

LỜI NÓI ĐẦU………………………………………………………………………………4

LỜI CẢM ƠN ……………………………………………………………………………….5

[NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN 6](#_Toc29204893)

[**CHƯƠNG I**. **TỔNG QUAN VỀ ĐỀ TÀI** 7](#_Toc29204894)

[**I. LÝ DO CHỌN ĐỀ TÀI 7**](#_Toc29204895)

[**II. HƯỚNG GIẢI QUYẾT VẤN ĐỀ**](#_Toc29204896) **7**

[1. Nghiên cứu sơ bộ và lập dự án 7](#_Toc29204897)

[2. Phân tích hệ thống 7](#_Toc29204898)

[3. Thiết kế tổng thể 7](#_Toc29204899)

[4. Thiết kế chi tiết - Thiết kế 7](#_Toc29204900)

[**CHƯƠNG II. CƠ SỞ LÝ THUYẾT** 8](#_Toc29204901)

[**I.** **ỨNG DỤNG NỀN TẢNG CỦA HỆ CƠ SỞ DỮ LIỆU** 8](#_Toc29204902)

[1. Mô hình thực thể quan hệ 8](#_Toc29204903)

[2. Các khái niệm cơ bản 8](#_Toc29204904)

[3. Quan hệ và mô hình quan hệ 8](#_Toc29204905)

[**II. ỨNG DỤNG PHÂN TÍCH & THIẾT KẾ HỆ THỐNG TIN 9**](#_Toc29204906)

[**CHƯƠNG III. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 10**](#_Toc29204907)

[**I.** **MÔ TẢ HỆ THỐNG** 10](#_Toc29204908)

[**II.** **XÂY DỰNG LƯỢC ĐỒ QUAN HỆ** 10](#_Toc29204909)

[1. Bảng Sinh Viên 10](#_Toc29204910)

[2. Bảng Lớp 11](#_Toc29204911)

[3. Bảng Khoa 11](#_Toc29204912)

[4. Bảng Ngành 11](#_Toc29204913)

[5. Bảng Giảng Viên 11](#_Toc29204914)

[6. Bảng Môn Học 12](#_Toc29204915)

[7. Bảng Học kỳ \_ Niên khóa 12](#_Toc29204916)

[**III.** **MÔ HÌNH DỮ LIỆU QUAN HỆ** 12](#_Toc29204917)

[1. Mô hình MCD 12](#_Toc29204918)

[2. Mô hình MLD 12](#_Toc29204919)

[**IV.** **PHÂN TÍCH CHỨC NĂNG CỦA HỆ THỐNG** 13](#_Toc29204920)

[1. Chức năng hiển thị thông tin điểm của sinh viên 13](#_Toc29204921)

[2. Chức năng hiển thị thông tin giảng viên 13](#_Toc29204922)

[3. Chức năng quản lý lớp 14](#_Toc29204923)

[4. Chức năng quản lý khoa 14](#_Toc29204924)

[5. Chức năng hiển thị quản lý ngành 15](#_Toc29204926)

[6. Chức năng hiển thị thông tin môn học 15](#_Toc29204927)

[7. Chức năng quản lý học kỳ - niên khóa 19](#_Toc29204928)

[8. Chức năng quản lý thông tin điểm sinh viên 22](#_Toc29204931)

[**V.** **CÁC YÊU CẦU CỦA HỆ THỐNG** 25](#_Toc29204933)

[1. Hệ thống tạo ra giải quyết các vấn đề 25](#_Toc29204934)

[2. Những thông tin cần cung cấp cho hệ thống 25](#_Toc29204935)

[3. Các yêu cầu cần hệ thống giải quyết 25](#_Toc29204936)

[4. Yêu cầu về quản lý điểm 25](#_Toc29204937)

[**VI.** **CÁC BƯỚC XÂY DỰNG HỆ THỐNG QUẢN LÝ ĐIỂM** 27](#_Toc29204938)

[1. Nghiên cứu sơ bộ và lập dự án 27](#_Toc29204939)

[2. Phân tích hệ thống 27](#_Toc29204940)

[3. Thiết kế tổng thể 28](#_Toc29204941)

[4. Thiết kế chi tiết - Thiết kế 28](#_Toc29204942)

[**VII.** **KẾT LUẬN** 38](#_Toc29204943)

[1. Ưu điểm 38](#_Toc29204944)

[2. Nhược điểm 38](#_Toc29204945)

[3. Hướng phát triển trong tương lai 39](#_Toc29204946)

[**Tóm lại** 39](#_Toc29204947)

[TÀI LIỆU THAM KHẢO 40](#_Toc29204948)

**MỤC LỤC HÌNH ẢNH**

[Hình 1. Mô hình MCD 12](#_Toc29228443)

[Hình 2. Giao diện hiển thị thông tin sinh viên 14](#_Toc29228444)

[Hình 3. Giao diện hiển thị thông tin lớp 14](#_Toc29228445)

[Hình 4. Giao diện hiển thị thông tin khoa 15](#_Toc29228446)

[Hình 5. Giao diện quản lý ngành học 15](#_Toc29228447)

[Hình 6. Giao diện thông tin môn học 16](#_Toc29228448)

[Hình 7. Giao diện hiển thị học kỳ-niên khóa 18](#_Toc29228449)

[Hình 8. Giao diện hiển thị thông tin điểm sinh viên 21](#_Toc29228450)

[Hình 9. Giao diện khi người dùng đăng nhập công hệ thống 26](#_Toc29228451)

[Hình 10. Giao diện hiển thị thông tin cần tìm kiếm 26](#_Toc29228452)

[Hình 11. Giao diện hiển thi thông tin lớp học 29](#_Toc29228453)

[Hình 12. Giao diện hiển thị thông tin 32](#_Toc29228454)

LỜI NÓI ĐẦU

**🖎🕮✍**

Niên luận 2 là một học phần nằm trong chương trình đào tạo chính quy của nghành Công nghệ thông tin – Khoa Kỹ thuật – Công nghệ – Trường Đại học Tây Đô. Mục đích và yêu cầu của học phần này là vận dụng kiến thức của các ngôn ngữ lập trình đã được học và vận dụng kiến thức của hệ quản trị cơ sở dữ liệu, phân tích thết kế hệ thống thông tin; tự nghiên cứu để viết một chương trình Demo theo yêu cầu của đề tài đã chọn, nhằm kiểm tra khả năng lập trình cũng như nâng cao tính tự học và sáng tạo của sinh viên. Sau khi làm xong niên luận này sinh viên có thể sử dụng thành thạo một ngôn ngữ lập trình, biết cách trình bày một báo cáo khoa học đúng theo yêu cầu, nhằm giúp cho sinh viên quen dần với các dự án mang tính khoa học.

Đề tài niên luận 2: Viết chương trình Quản lý điểm sinh viên là một đề tài đòi hỏi vận dụng rất nhiều đến tư duy lập trình được tích góp từ nhiều học phần khác nhau như lập trình căn bản, lập trình hướng đối tượng, cấu trúc dữ liệu. Bên cạnh đó để thực hiện tốt chương trình thì không thể thiếu là lựa chọn ngôn ngữ lập trình và công cụ lập trình phù hợp, ngôn ngữ lập trình mà chúng tôi sử dụng là Java kết hợp với công cụ mà chúng tôi sử dụng đó chính là Visual Studio 2013. Với đề tài niên luận này, đó là cơ hội là tiền đề để chúng tôi tiếp tục tự nghiên cứu với tầm cao hơn, phát triển chính bản thân mình và trang bị một hành trang vững chắc để mỗi sinh viên chúng tôi vững bước vào đời. Và đặt biệt hơn chúng tôi có cơ hội tìm hiều sâu hơn về cơ chế quản lý điểm của sinh viên và có thể hệ thống của chúng em được áp dụng vào thực tế nếu chúng em thực tốt.

*Cần Thơ, tháng 11 năm 2019* **Nhóm Sinh viên thực hiện**

# LỜI CẢM ƠN

**🖎🕮✍**

Đây là chương trình phần mềm thực tế về cơ chế quản lý đầu tiên mà nhóm chúng em thực hiện nên không thể tránh khỏi sự những sai sót trong khâu thiết kế và trình bày, sự thiếu kinh nghiệm trong khâu lên ý tưởng và thực hiện nhưng nhờ sự tận tâm hướng dẫn của Thầy Nguyễn Chí Cường – Giảng viên hướng dẫn mà nhóm chúng em đã khắc phục được những khó khăn, chỉnh sửa được sai sót và hoàn thành được bài tập một cách chính xác, chuyên nghiệp. Nhóm chúng em xin chân thành cảm ơn thầy đã tạo điều kiện cho chúng em được học hỏi, trải nghiệm và thu được kết quả như mong muốn.

Do đây là lần đầu tiên nhóm thực hiện một bài tập mang tính thực tế và đơn giản quá trình thực hiện khó lòng tránh khỏi sai sót, cộng thêm vốn kiến thức còn hạn hẹp nên phần diễn đạt có thể chưa hoàn toàn chính xác, mong thầy nhận xét phê bình để nhóm chúng em có thể rút ra được kinh nghiệm cho các bài tập và đồ án sau này.

Chúng em xin chân thành cảm ơn!

*Cần Thơ, tháng 12 năm 2019*   
 **Nhóm Sinh viên thực hiện**

# NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

**🖎🕮✍**

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

*Cần Thơ, ngày ... tháng 12 năm 2019*  
 **Giảng viên hướng dẫn**

**Nguyễn Chí Cường**

**CHƯƠNG I TỔNG QUAN VỀ ĐỀ TÀI**

**🖎🕮✍**

1. **LÝ DO CHỌN ĐỀ TÀI**

Cùng với sự phát triển nhanh chóng của công nghệ thông tin con người đã nghĩ ra rất nhiều các ứng dụng phục vụ cho nhu cầu và lợi ích cho mình. Trong đó xây dựng phần mềm phục vụ cho nền giáo dục cũng được quan tâm rất nhiều. Từ các trường Cao đẳng, Đại học cho đến các trường Trung học phổ thông các cấp đều sử dụng phần mềm quản lý để giảm thiểu tối đa những vất vả trong công việc như lưu trữ hồ sơ học sinh, sinh viên cũng như công việc quản lý điểm…

Nếu trước đây việc lưu trữ điểm số của học sinh, sinh viên gặp rất nhiều khó khăn bởi vì sổ sách giấy tờ liên quan được chấc chòng trên kho thì bây giờ chỉ cần một chiếc máy tính nhỏ gọn cũng có thể lưu trữ cả kho hồ sơ. Với mong muốn tìm hiểu một phần nhỏ sự phát triển của công nghệ thông tin, chúng em đã cùng nhau viết một phần mềm nhỏ là *“Quản Lý Điểm Sinh Viên”* với mục đích trước hết là mở rộng kiến thức về ngôn ngữ lập trình của mình, và sau đó là mong muốn phần mềm của mình có thể ấp dụng vào hoạt động thực tế, phục vụ lợi ích cho việc quản lý điểm sinh viên của các trường Đại học, Cao đẳng hiện nay.

1. **HƯỚNG GIẢI QUYẾT VẤN ĐỀ**
2. ***Nghiên cứu sơ bộ và lập dự án***

Ở bước này, ta tiến hành tìm hiểu, khảo sát hệ thống, phát hiện các nhược điểm còn tồn tại, từ đó đề xuất các giải pháp khác phục, cân nhắc tính khả thi của dự án và định hướng cho các giai đoạn phát triển tiếp theo.

1. ***Phân tích hệ thống***

Phân tích một cách chi tiết hệ thống hiện tại để xây dựng các lược đồ khái niệm, trên cơ sở đó tiến hành xây dựng lược đồ cho hệ thống mới.

1. ***Thiết kế tổng thể***

Thiết kế tổng thể hệ thống nhằm xác định vai trò của máy tính trong hệ thống mới, phân định rõ ranh giới các công việc làm bằng máy tính với bằng thủ công. Từ đó, xác định các hệ thống con trong phần việc được làm bằng máy tính.

1. ***Thiết kế chi tiết - Thiết kế***

Các thủ tục người dùng và giao diện giữa người và máy tính - Thiết kế cơ sở dữ liệu - Thiết kế kiểm soát (ngăn truy nhập trái phép, an toàn sự cố) - Viết chương trình bằng ngôn ngữ lập trình Visual Fox 6.0 - Chạy thử chương trình.

# CHƯƠNG II CƠ SỞ LÝ THUYẾT

**🖎🕮✍**

1. **ỨNG DỤNG NỀN TẢNG CỦA HỆ CƠ SỞ DỮ LIỆU**
2. ***Mô hình thực thể quan hệ***

• Mô hình thực thể quan hệ: là mô hình dùng để biểu diễn các cơ sở dữ liệu, trong đó cơ sở dữ liệu bao gồm các thưc thể *(entity)* và mối quan hệ *(relation).*

• Mô hình thực thể quan hệ biểu diễn một cách hệ thống các khái niệm *(logic).*

***2. Các khái niệm cơ bản***

- Thực thể *(entity):* thực thể là một đối tượng, vật thể cụ thể hay trừu tượng, tồn tại thực sự và khá ổn định trong thế giới thực, mà ta muốn phản ánh nó trong hệ thống thông tin.

- Thuộc tính *(attribute/ propepty):* là một giá trị dùng để mô tả đặc điểm một khía cạnh nào đó của các thưc thể.

- Thuộc tính được phân ra làm nhiều loại như sau:

+ Thuộc tính đơn *(simple attribute):* chỉ có thể mô tả một đặc tính.

+ Thuộc tính tổng hợp *(composite attribute):* là thuộc tính có thể mô tả nhiều thuộc tính .

+ Thuộc tính đơn trị *(simple value attribute):* là thuộc tính duy nhất mà mỗi thực thể có.

+ Thuộc tính đa trị *(multi value attribute):* là thuộc tính có thể lặp lại ứng với mỗi thực thể.

+ Thuộc tính dẫn xuất *(perived attribute):* là thuộc tính có thể được suy từ cái khác.

- Liên kết *(association):* Một liên kết là một sự gom nhóm các thực thể trong đó mỗi thực thể có một vai trò nhất định.

***3. Quan hệ và mô hình quan hệ***

- Mô hình quan hệ cho ta biết mối quan hệ giữa các thực thể, cho ta có cái nhìn rõ hơn về các thực thể.

- Mối quan hệ: là sự liên kết giữa các thực thể với nhau. Quy trình xây dựng mô hình E - R

* ***Bước 1:*** Liệt kê các đăc trưng của nó, chính xác hoá và lựa chọn thông tin của các hồ sơ.
* ***Bước 2:*** Đã có một danh sách các thuộc tính, chọn lọc tìm thực thể và các thuộc tính của nó: Loại các thuộc tính được chon khỏi danh sách các thuộc tính.
* ***Bước 3:*** Xác định các mối quan hệ giữa các thực thể đã xác định ở bước 2.
* ***Bước 4:*** Vẽ sơ đò mô hình E-R và xác định bản số cho các thực thể tham gia vào trong các quan hệ. trước hết vẽ thực thể, sau đó vẽ mối quan hệ.
* ***Bước 5:*** Chuẩn hoá sơ đồ và thu gọn sơ đồ. Sau khi vẽ sơ đồ ta có thể chuẩn hoá sơ đồ nếu trong đó có các thuộc tính lặp.

1. **ỨNG DỤNG PHÂN TÍCH & THIẾT KẾ HỆ THỐNG TIN**

Tệp (File) dữ liệu: Hầu hết dữ liệu được xử lí trong hệ thống thông tin phải được giữ lại vì lí do pháp luật hoặc vì sự cần thiết được xử lí trong tương lai. Những file này là thành phần của hệ thống thông tin, được tạo ra trực tiếp hoặc lưu trữ trong file.

Các dữ liệu là các thông tin được cấu trúc hóa.

Luồng thông tin vào: Các thông tin cần thiết cho quá trình xử lí, có thể là các thông tin phản ánh cấu trúc doanh nghiệp và các thông tin phản ánh hoạt động của doanh nghiệp.

Luồng thông tin ra: Thông tin ra được tổng hợp từ các thông tin đầu vào và phụ thuộc vào nhu cầu quản lí trong từng trường hợp, từng đơn vị cụ thể

Thông tin ra là kết quả của việc tra cứu nhanh về một đối tượng cần quan tâm, đồng thời phải đảm bảo sự chính xác và kịp thờ

Các xử lí là các quy trình, các phương pháp, chức năng xử lí thông tin và biến đổi thông tin. Các xử lí nhằm sản sinh ra các thông tin có cấu trúc theo thể thức quy định như các chứng từ giao dịch, các sổ sách báo cáo thống kê. Cung cấp các thông tin trợ giúp quyết định.

# CHƯƠNG III PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

**🖎🕮✍**

1. **MÔ TẢ HỆ THỐNG**

Hiện nay, quản lý điểm của sinh viên của một số trường Đại học, Cao đẳng được thực hiện bằng phương pháp truyền thống, đó là quản lý thông tin bằng sổ sách với những công việc: Đầu tiên khi sinh viên mới nhập học thì nhân viên phòng đào tạo (PĐT) sẽ lưu trữ thông tin của sinh viên theo lớp, khoa với những thông tin như: Mã sinh viên, họ và tên, ngày sinh, hộ khẩu, điện thoại…. Sau đó nhân viên sẽ in ra một danh sách và phát cho từng lớp. Trong quá trình học tập, nhân viên PĐT sẽ thực hiện các công việc quản lý như nhận danh sách sinh viên được thi từ giảng viên và đưa ra danh sách sinh viên được thi. Lên lịch thi và tiến hành nhập điểm thi cho sinh viên…rồi in bảng điểm cho từng lớp. Sau đó lại lên danh sách sinh viên thi lại, sinh viên học lại. Với số lượng sinh viên nhiều thì việc giám sát và quản lý điểm bằng cách thủ công sẽ gặp rất nhiều khó khăn như trong quá trình lưu trữ dễ xảy ra sai sót, thiếu chính xác, số lượng sổ sách và giấy tờ liên quan qua nhiều năm là rất lớn, tốn thời gian và công sức cho người quản lý , việc chỉnh sửa cũng như tìm kiếm gặp rất nhiều khó khăn.

Để giúp việc quản lý điểm sinh viên trở lên dễ dàng hơn, chúng em đã xây dựng phần mềm với mục tiêu:

* Phần mềm vận hành đơn giản, dễ sử dụng.
* Tổ chức quản lý, lưu trữ trên hệ thống máy vi tính làm tăng tính an toàn.
* Giao diện trực quan bằng tiếng Việt, các form nhập liệu đơn giản, dễ thao tác.
* Phù hợp với các ứng dụng triển khai trên diện rộng Dễ dàng chỉnh sửa, tìm kiếm…

Với chương trình này, giảng viên & phụ huynh của sinh viên có thể kiểm tra, theo dõi điểm số, kết quả học tập của sinh viên, con em mình; có thể điều chỉnh thay đổi một cách hợp lý và nhanh chóng. Bên cạnh đó còn giúp cho công việc quản lý tiết kiệm được khá nhiều thời gian và chi phí để lưu trữ dữ liệu về sinh viên.

1. **XÂY DỰNG LƯỢC ĐỒ QUAN HỆ**
2. ***Bảng Sinh Viên***

*(MaSV , TenSV, Ngaysinh, Gioitinh)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Quản lý Điểm | | Bảng mô tả thực thể | |  | |
| ***Sinh viên*** | |
| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu** | **Ghi chú** | **Chiều dài** |
| 1 | MaSV | Mã sinh viên | Number | Khóa chính | 10 |
| 2 | TenSV | Tên sinh viên | Text |  | 20 |
| 3 | Ngaysinh | Ngày sinh | Text |  | 20 |
| 4 | Gioitinh | Giới tính | Text |  | 20 |

1. ***Bảng Lớp***

*(Malop, Tenlop)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Quản lý Điểm | | Bảng mô tả thực thể | |  | |
| ***Lớp*** | |
| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu** | **Ghi chú** | **Chiều dài** |
| 1 | Malop | Mã Lớp | Text | Khóa chính | 10 |
| 2 | Tenlop | Tên Lớp | Text |  | 20 |

1. ***Bảng Khoa***

*(Makhoa, Tenkhoa)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Quản lý Điểm | | Bảng mô tả thực thể | |  | |
| ***Khoa*** | |
| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu** | **Ghi chú** | **Chiều dài** |
| 1 | Makhoa | Mã Khoa | Number | Khóa chính | 10 |
| 2 | Tenkhoa | Tên Khoa | Text |  | 20 |

1. ***Bảng Ngành***

*(Manganh, Tennganh, SonamĐT)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Quản lý Điểm | | Bảng mô tả thực thể | |  | |
| ***Ngành*** | |
| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu** | **Ghi chú** | **Chiều dài** |
| 1 | Manganh | Mã ngành | Number | Khóa chính | 10 |
| 2 | Tennganh | Tên ngành | Text |  | 20 |
| 3 | SonamĐT | Số năm đào tạo | Text |  | 20 |

1. ***Bảng Giảng Viên***

*(MaGV, HoGV, TenGV, Ngaysinh, Gioitinh)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Quản lý Điểm | | Bảng mô tả thực thể | |  | |
| ***Giảng viên*** | |
| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu** | **Ghi chú** | **Chiều dài** |
| 1 | MaGV | Mã Giảng viên | Number | Khóa chính | 10 |
| 2 | TenGV | Tên Giảng viên | Text |  | 20 |
| 3 | Ngaysinh | Ngày sinh | Text |  | 20 |
| 4 | Gioitinh | Giới tính | Text |  | 20 |

1. ***Bảng Môn Học***

*(MaMH, TenMH, SoTC)*

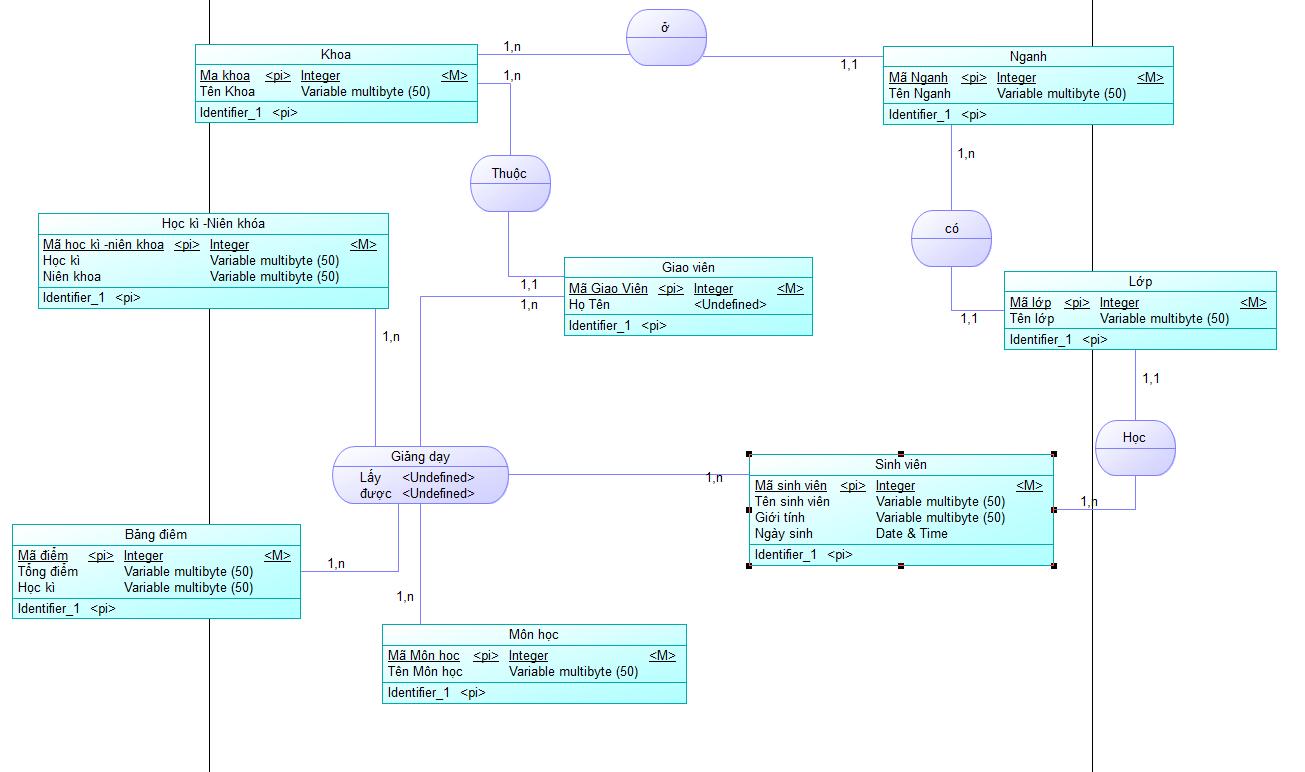
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Quản lý Điểm | | Bảng mô tả thực thể | |  | |
| ***Môn Học*** | |
| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu** | **Ghi chú** | **Chiều dài** |
| 1 | MaMH | Mã Môn Học | Number | Khóa chính | 10 |
| 2 | TenMH | Tên Môn Học | Text |  | 20 |
| 3 | Tinchi | Tín chỉ | Text |  | 20 |

1. ***Bảng Học kỳ \_ Niên khóa***

*(Mahk\_NK, Hocky, Nienkhoa)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Quản lý Điểm | | Bảng mô tả thực thể | |  | |
| ***Học kỳ - Niên khóa*** | |
| **STT** | **Tên thuộc tính** | **Diễn giải** | **Kiểu** | **Ghi chú** | **Chiều dài** |
| 1 | MaHK-NK | Mã HK\_NK | Number | Khóa chính | 10 |
| 2 | Hocky | Học kỳ | Text |  | 20 |
| 3 | Nienkhoa | Niên khóa | Text |  | 20 |

1. **MÔ HÌNH DỮ LIỆU QUAN HỆ**
2. ***Mô hình MCD***

******

**Hình 1. Mô hình MCD**

1. ***Mô hình MLD***

* SINHVIEN (MaSV, TenSV, Ngaysinh, Gioitinh, maDiem)
* NGANH ( Manganh, Tennganh, Makhoa)
* GIAOVIEN (MaGV, TenGV, MaKhoa)
* MONHOC (MaMH, TenMH, Tinchi)
* LOP (Malop, Tenlop, Manganh, MaSV)
* KHOA (Makhoa, Tenkhoa)
* HOCKI\_NIENKHOA (MaHK\_NK, Hocky, Nienkhoa)
* GIANGDAY (Malop, MaMH, MaGV, MaHK\_NK)
* ***Ràng buộc tham chiếu***
* KHOA (Makhoa) → NGANH (Makhoa)
* KHOA(Makhoa) → GIAOVIEN (MaGV)
* LOP (Manganh) → NGANH (Manganh)
* LOP (Malop) → SINHVIEN (Malop)
* GIANGDAY (Malop) → LOP (Malop)
* GIANGDAY (MaMH) → MONHOC (MaMH)
* GIANGDAY (MaGV) → GIAOVIEN (MaGV)
* GIANGDAY (MaHK\_NK) → HOCKI\_NIENKHOA (MaHK\_NK)

1. **PHÂN TÍCH CHỨC NĂNG CỦA HỆ THỐNG**

***1. Chức năng hiển thị thông tin điểm của sinh viên***

- Hiển thị thông tin sinh viên.

- Tìm kiếm, thêm, sửa, xóa các thông tin của sinh viên, mã số sinh viên, đồng thời hiện tên môn học cùng điểm của môn học, xếp loại.

- Xếp loại học tập:

+ Loại giỏi: 8.00 <= ĐTB < 9.00

+ Loại khá: 7.00 <= ĐTB < 8.00

+ Loại trung bình: 5.00 <= ĐTB < 7.00

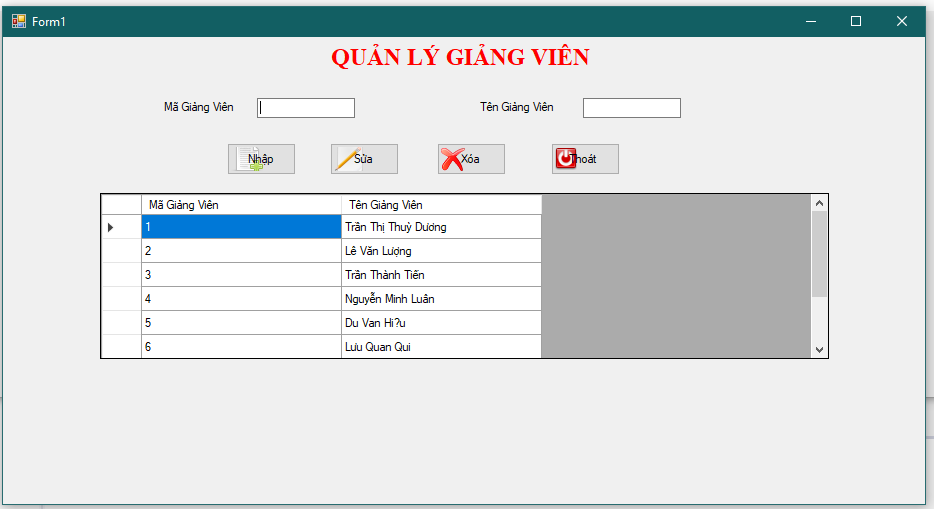
+ Loại yếu: 4.00 <= ĐTB < 5.00

+ Loại kém: ĐTB < 4.00

***2. Chức năng hiển thị thông tin giảng viên***

- Hiển thị thông tin của giảng viên.

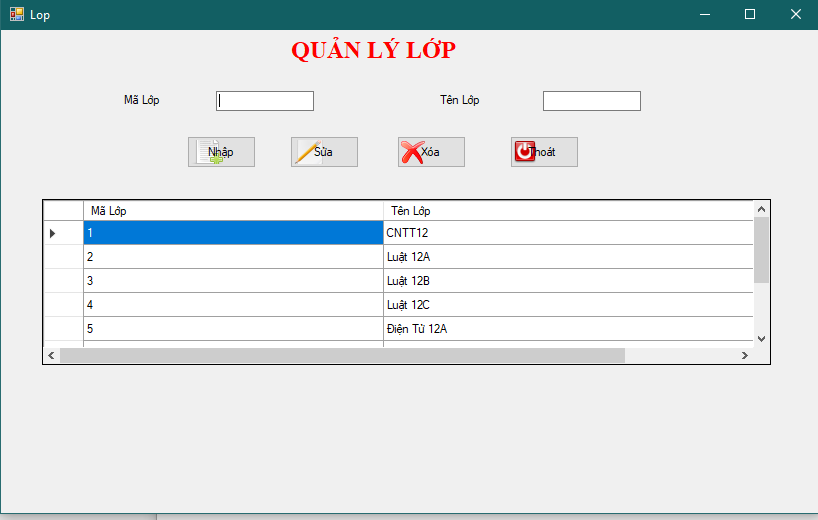
- Tìm kiểm, thêm, sửa xóa thông tin của giảng viên.



**Hình 2. Giao diện hiển thị thông tin sinh viên**

1. ***Chức năng quản lý lớp***

- Hiển thị thông tin lớp.

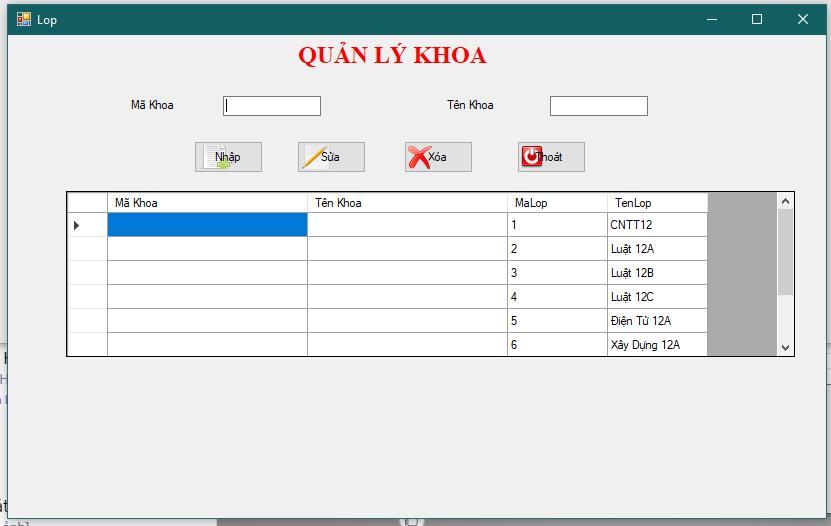
**- Tìm kiếm, thêm, sửa, xóa các trường thông tin lớp học.

**Hình 3. Giao diện hiển thị thông tin lớp**

1. ***Chức năng quản lý khoa***

- Hiển thị thông tin các khoa.

- Tìm kiếm, thêm, sửa, xóa thông tin khoa.



**Hình 4. Giao diện hiển thị thông tin khoa**

1. ***Chức năng hiển thị quản lý ngành***

- Hiển thị thông tin các ngành.

- Tìm kiếm, thêm, sửa, xóa thông tin ngành.

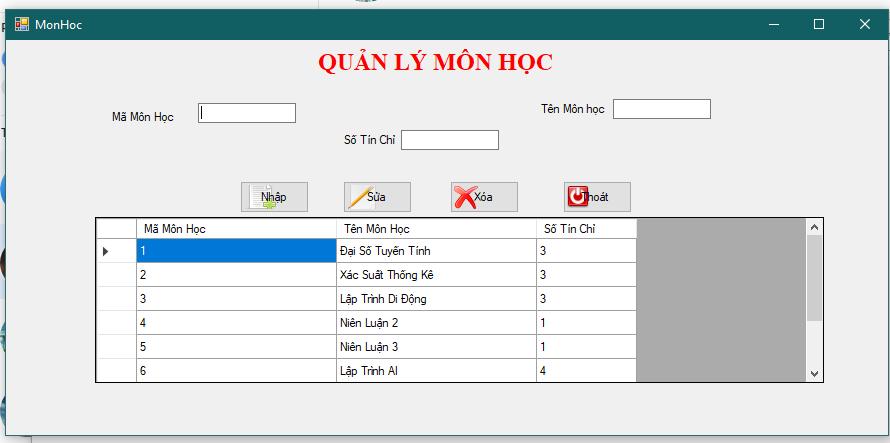


**Hình 5. Giao diện quản lý ngành học**

1. ***Chức năng hiển thị thông tin môn học***

- Tìm kiếm, thêm, sửa, xóa các môn học theo thông tin của các trường.

- Hiển thị, tìm kiếm thông tin của sinh viên, học kỳ, phòng học, số tín chỉ.



**Hình 6. Giao diện thông tin môn học**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLDiem

{

public partial class MonHoc : Form

{

public MonHoc()

{

InitializeComponent();

}

private void ketnoi()

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

string sql = "select \* from MonHoc";

SqlCommand con = new SqlCommand(sql, kn); // thực thi câu lệnh trong sql

SqlDataAdapter com = new SqlDataAdapter(con);//vận chuyển dữ liệu

DataTable tb = new DataTable();

com.Fill(tb);// đổ dữ liệu vào bảng ảo

dataGridView1.DataSource = tb; // đổ dữ liệu từ bảng vào datagirdview

}

catch (Exception)

{

MessageBox.Show(" Lỗi Kết Nối Vui Lòng Kiểm Tra Lại !!!!");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();//đóng kết nối với sql

}

}

private void MonHoc\_Load\_1(object sender, EventArgs e)

{

ketnoi();

}

int index;

private void dataGridView1\_Click(object sender, EventArgs e)

{

index = dataGridView1.CurrentRow.Index;

txtmonhoc.Text = dataGridView1.Rows[index].Cells[0].Value.ToString();

txttenmonhoc.Text = dataGridView1.Rows[index].Cells[1].Value.ToString();

txtsotinchi.Text = dataGridView1.Rows[index].Cells[2].Value.ToString();

}

string them;

private void bnt\_add\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

them = "INSERT INTO MonHoc VALUES('" + txtmonhoc.Text + "','" + txttenmonhoc.Text + "','" + txtsotinchi.Text + "')";

SqlCommand comthem = new SqlCommand(them, kn);

comthem.ExecuteNonQuery();

MessageBox.Show("Thêm Dữ Liệu Thành Công");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Thêm Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

string sua;

private void bnt\_sua\_Click\_1(object sender, EventArgs e)

{

Try

{ SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

sua = "update MonHoc set TenMonHoc='" + txttenmonhoc.Text + "', TinChi='" + txtsotinchi.Text + "' where MaMonHoc='" + txtmonhoc.Text + "'";

SqlCommand commandsua = new SqlCommand(sua, kn);

commandsua.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Sửa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show ("Lỗi!!! . Không Thể Sửa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

string xoa;

private void bnt\_xoa\_Click(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

xoa = "delete MonHoc where MaMonHoc='" + txtmonhoc.Text + "'";

SqlCommand commandxoa = new SqlCommand(xoa, kn);

commandxoa.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Xóa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Xóa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

private void bnt\_thoat\_Click(object sender, EventArgs e)

{

DialogResult hoi;

hoi = MessageBox.Show("Bạn Có Thoát Hay Không?", "Thông Báo!", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (hoi == DialogResult.Yes)

{

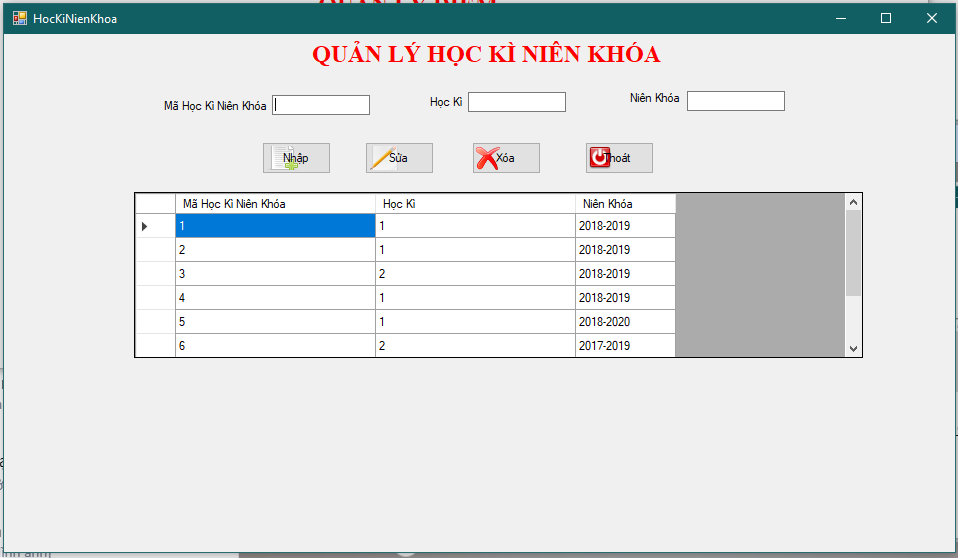
this.Close();

}

}

}

}

1. ***Chức năng quản lý học kỳ - niên khóa***

**Hình 7. Giao diện hiển thị Học kỳ \_Niên khóa**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLDiem

{

public partial class HocKiNienKhoa : Form

{

public HocKiNienKhoa()

{

InitializeComponent();

}

private void ketnoi()

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

string sql = "select \* from HockiNienKhoa";

SqlCommand con = new SqlCommand(sql, kn); // thực thi câu lệnh trong sql

SqlDataAdapter com = new SqlDataAdapter(con);//vận chuyển dữ liệu

DataTable tb = new DataTable();

com.Fill(tb);// đổ dữ liệu vào bảng ảo

dataGridView1.DataSource = tb; // đổ dữ liệu từ bảng vào datagirdview

}

catch (Exception)

{

MessageBox.Show(" Lỗi Kết Nối Vui Lòng Kiểm Tra Lại !!!!");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();//đóng kết nối với sql

}

}

private void HocKiNienKhoa\_Load(object sender, EventArgs e)

{

ketnoi();

}

int index;

private void dataGridView1\_Click(object sender, EventArgs e)

{

index = dataGridView1.CurrentRow.Index;

txtmahknk.Text = dataGridView1.Rows[index].Cells[0].Value.ToString();

txthocki.Text = dataGridView1.Rows[index].Cells[1].Value.ToString();

txtnienkhoa.Text = dataGridView1.Rows[index].Cells[2].Value.ToString();

}

string them;

private void bnt\_add\_Click(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

them = "INSERT INTO HockiNienKhoa VALUES('" + txtmahknk.Text + "','" + txthocki.Text + "','" + txtnienkhoa.Text + "')";

SqlCommand comthem = new SqlCommand(them, kn);

comthem.ExecuteNonQuery();

MessageBox.Show("Thêm Dữ Liệu Thành Công");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Thêm Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

private void bnt\_thoat\_Click\_1(object sender, EventArgs e)

{

DialogResult hoi;

hoi = MessageBox.Show("Bạn Có Thoát Hay Không?", "Thông Báo!", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (hoi == DialogResult.Yes)

{

this.Close();

}

}

string sua;

private void bnt\_sua\_Click(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

sua = "update HockiNienKhoa set HocKi='" + txthocki.Text + "', NienKhoa='" + txtnienkhoa.Text + "' where MaHockiNienKhoa='" + txtmahknk.Text + "'";

SqlCommand commandsua = new SqlCommand(sua, kn);

commandsua.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Sửa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Sửa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

string xoa;

private void bnt\_xoa\_Click(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

xoa = "delete HockiNienKhoa where MaHockiNienKhoa='" + txtmahknk.Text + "'";

SqlCommand commandxoa = new SqlCommand(xoa, kn);

commandxoa.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Xóa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Xóa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

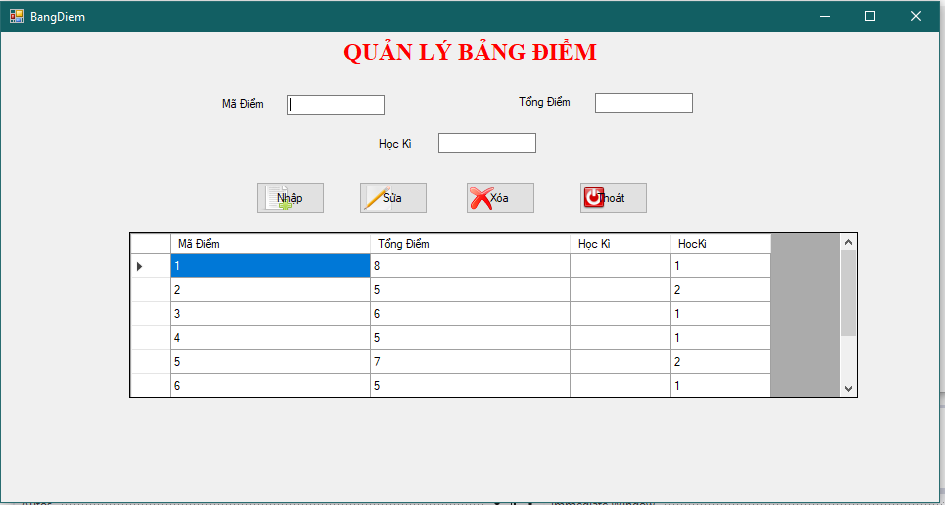
}

}

}

1. ***Chức năng quản lý thông tin điểm sinh viên***

* Hiển thị thông tin điểm các môn, từng học kỳ, năm học của sinh viên.
* Cho phép tìm kiếm thông tin, thêm, xóa, chỉnh sửa điểm.

****

**Hình 8. Giao diện hiển thị thông tin điểm sinh viên**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLDiem

{

public partial class BangDiem : Form

{

public BangDiem()

{

InitializeComponent();

}

private void ketnoi()

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

string sql = "select \* from BangDiem";

SqlCommand con = new SqlCommand(sql, kn); // thực thi câu lệnh trong sql

SqlDataAdapter com = new SqlDataAdapter(con);//vận chuyển dữ liệu

DataTable tb = new DataTable();

com.Fill(tb);// đổ dữ liệu vào bảng ảo

dataGridView1.DataSource = tb; // đổ dữ liệu từ bảng vào datagirdview

}

catch (Exception)

{

MessageBox.Show(" Lỗi Kết Nối Vui Lòng Kiểm Tra Lại !!!!");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();//đóng kết nối với sql

}

}

private void BangDiem\_Load(object sender, EventArgs e)

{

ketnoi();

}

int index;

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

index = dataGridView1.CurrentRow.Index;

txtmadiem.Text = dataGridView1.Rows[index].Cells[0].Value.ToString();

txttongdiem.Text = dataGridView1.Rows[index].Cells[1].Value.ToString();

txthocki.Text = dataGridView1.Rows[index].Cells[2].Value.ToString();

}

string them;

private void bnt\_add\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

them = "INSERT INTO BangDiem VALUES('" + txtmadiem.Text + "','" + txttongdiem.Text + "','" + txthocki.Text + "')";

SqlCommand comthem = new SqlCommand(them, kn);

comthem.ExecuteNonQuery();

MessageBox.Show("Thêm Dữ Liệu Thành Công");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Thêm Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

private void bnt\_thoat\_Click(object sender, EventArgs e)

{

DialogResult hoi;

hoi = MessageBox.Show("Bạn Có Thoát Hay Không?", "Thông Báo!", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (hoi == DialogResult.Yes)

{

this.Close();

}

}

string sua;

private void bnt\_sua\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

sua = "update BangDiem set TenGV='" + txttongdiem.Text + "','" + txthocki.Text + "' where MaGv='" + txtmadiem.Text + "'";

SqlCommand commandsua = new SqlCommand(sua, kn);

commandsua.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Sửa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Sửa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

string xoa;

private void bnt\_xoa\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

xoa = "delete BangDiem where MaGv='" + txtmadiem.Text + "'";

SqlCommand commandxoa = new SqlCommand(xoa, kn);

commandxoa.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Xóa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Xóa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

}

}

1. **CÁC YÊU CẦU CỦA HỆ THỐNG**
2. ***Hệ thống tạo ra giải quyết các vấn đề***

Hệ thống tạo ra dung để giải quyết vấn đề điểm số của sinh viên, hệ thống cũng có thẻ tra cứu một số thông tin cần thiết về sinh viên như ngày sinh, quê quán, chổ ở hiện tại, …

Theo dõi toàn bộ quá trình học ở trong trường từ khi nhập học đến ra trường.

Hệ thống cho ra được kết quả học tập của sinh viên theo các học kì,năm.

Kiểm soát được việc khen thưởng, xét học bổng, xét lưu ban, thi lại, học lại của sinh viên.

Cho biết được hồ sơ cụ thể về kết quả học tập rèn luyện của sv khi ra trường.

1. ***Những thông tin cần cung cấp cho hệ thống***

Sinh viên, Mã số sinh viên, Lớp, Mã lớp, Học kỳ\_Niên khóa, Điểm, Khoa, Ngành, Môn học là những thông tin cần thiết phải cung cấp cho hệ thống.

1. ***Các yêu cầu cần hệ thống giải quyết***

Để khắc phục những nhược điểm trong công tác quản lý điểm, chương trình quản lý điểm trong đồ án này được xây dựng với các yêu cầu:

- Xây dựng phần mền ứng dụng theo tiêu chuẩn hiện đại để đáp ứng nhu cầu xử lỳ các chức năng nghiệp vụ trong quá trình quản lý điểm các loại hình kiểm tra, thi của sinh viên theo học tại trường.

- Từ thực tế công tác nghiệp vụ quản lý điểm học tập tại trường tiến hành phân tích và tin học hoá những phần công việc có thể tin học hoá. Từ đó, xây dựng ứng dụng hỗ trợ cho công tác quản lý điểm. Quản trị quá trình xử lý điểm theo các quy định, qui tắc của Bộ Giáo dục đào tạo, các quy định của khoa.

- Cập nhật, lưu trữ, thống kê các thông tin về chương trình đào tạo của nhà trường.

- Cập nhật, lưu trữ, thống kê các thông tin về điểm học tập của học viên.

- Tìm kiếm, tra cứu các thông tin về đào tạo, về điểm học tập của học viên theo nhiều tiếu thức khác nhau.

- Soạn thảo, in ấn, sao lưu các thông tin theo yêu cầu ở trên.

***4. Yêu cầu về quản lý điểm***

Dữ liệu ban đầu cho quá trình quản lý điểm là bảng điểm, do vậy các thông tin trong bảng điểm phải có độ chính xác tuyệt đối. Bao gồm các thông tin sau:

- Mã số bảng điểm (dùng xác định duy nhất một bảng điểm trong số các bảng điểm)

- Tên môn học.

- Tên lớp.

- Loại hình thi/ kiểm tra.

- Học kỳ.

- Ngày thi/ kiểm tra.

- Danh sách sinh viên của cùng một lớp.

- Kết quả thi/ kiểm tra.

Hệ thống phải có chức năng thống kê theo yêu cầu như: xếp loại, học tập, hạnh kiểm Đối với hệ thống này, các thông tin đầu ra cần phải đáp ứng các yêu cầu sau:

Đối với các thông tin đưa ra bằng phương pháp tính toán: hệ thống phải đưa ra được các báo cáo sau:

- Danh sách học viên đủ điều kiện làm đồ án tốt nghiệp: Đó là những học viên có điểm trung bình chung của 6 học kỳ đầu nhỏ hơn hoặc bằng 6.00 và lớn hơn hoặc bằng 5.00, riêng học kỳ VI chỉ xét điểm thi lần 1.

- Danh sách học viên phải thi lại các môn ở học kỳ vừa kết thúc: hệ thống chỉ cần thống kê những môn học mà học viên thi lần thứ nhất đạt điểm nhỏ hơn 5 và kết quả cụ thể của lần thi đó.

- Danh sách học viên phải lưu ban: Học viên phải lưu ban là học viên có 25% số học trình của một năm học là dưới điểm 5

- Bản báo cáo tổng hợp kết quả học tập của từng lớp theo học kỳ và năm học: Sau mỗi lần kết thúc học kỳ hoặc năm học, hệ thống phải đưa ra danh sách theo từng lớp trong đó thông báo điểm trung bình chung của từng học viên, phân loại học tập cho từng học viên theo tiêu chuẩn xếp loại học tập.

- Bản báo cáo tổng hợp kết quả học tập của từng lớp của từng khoá học: tiêu chí báo cáo cũng giống như đối với học kỳ và năm học

Phiếu điểm tốt nghiệp của từng học viên khi tốt nghiệp ra trường: Phiếu điểm này bao gồm tất cả các môn học trong khoá học, số học trình và điểm của môn học, điểm các môn tốt nghiệp mà học viên đạt được.

+ Đối với các thông tin đưa ra bằng phương pháp tra cứu, tìm kiếm: Hệ thống này chỉ giới hạn trong việc tìm kiếm học viên theo một số thông tin như: Lớp, họ tên để xem một số thông tin trong quá trình học tập của học viên.

+ Các thông tin dự báo: Đây là một phạm vi mà hệ thống không đề cập đến. Tuy nhiên, dưới góc độ phân tích và thiết kế hệ thống thì có thể thấy rằng nếu phát triển hệ thống thì vẫn có thể đáp ứng được phần nào dựa trên những số liệu thống kê của những năm trước để lại. Tóm lại, trong 3 loại thông tin mà hệ thống phải đáp ứng thì loại thông tin đưa ra bằng phương pháp tính toán là quan trọng và cần thiết nhất, loại thông tin thứ 2 là có nhưng không quan trọng và hệ thống cũng không đề cập đến nhiều, còn loại thông tin thứ 3 thì chưa có nhu cầu sử dụng.

* Các thông tin trợ giúp.

Nhóm các thông tin trợ giúp bao gồm: khen thưởng, kỷ luật, đối tượng ưu tiên. Nhóm thông tin này không nằm trong nguồn cung cấp thông tin chính thống của hệ thống vì vậy thông tin không được đầy đủ (có học viên có nhưng cũng có học viên không có). Vì vậy, hệ thống chỉ nhập lưu và nếu cần thì đưa ra cùng các thông tin cần thiết khác. Các thông tin đó không tham gia vào quá trình biến đổi thông tin để kết xất thành thông tin đầu ra.

1. **CÁC BƯỚC XÂY DỰNG HỆ THỐNG QUẢN LÝ ĐIỂM**
2. ***Nghiên cứu sơ bộ và lập dự án***

Ở bước này, ta tiến hành tìm hiểu, khảo sát hệ thống, phát hiện các nhược điểm còn tồn tại, từ đó đề xuất các giải pháp khác phục, cân nhắc tính khả thi của dự án và định hướng cho các giai đoạn phát triển tiếp theo.

1. ***Phân tích hệ thống***

Phân tích một cách chi tiết hệ thống hiện tại để xây dựng các lược đồ khái niệm, trên cơ sở đó tiến hành xây dựng lược đồ cho hệ thống mới.

1. ***Thiết kế tổng thể***

Thiết kế tổng thể hệ thống nhằm xác định vai trò của máy tính trong hệ thống mới, phân định rõ ranh giới các công việc làm bằng máy tính với bằng thủ công. Từ đó, xác định các hệ thống con trong phần việc được làm bằng máy tính.

1. ***Thiết kế chi tiết - Thiết kế***

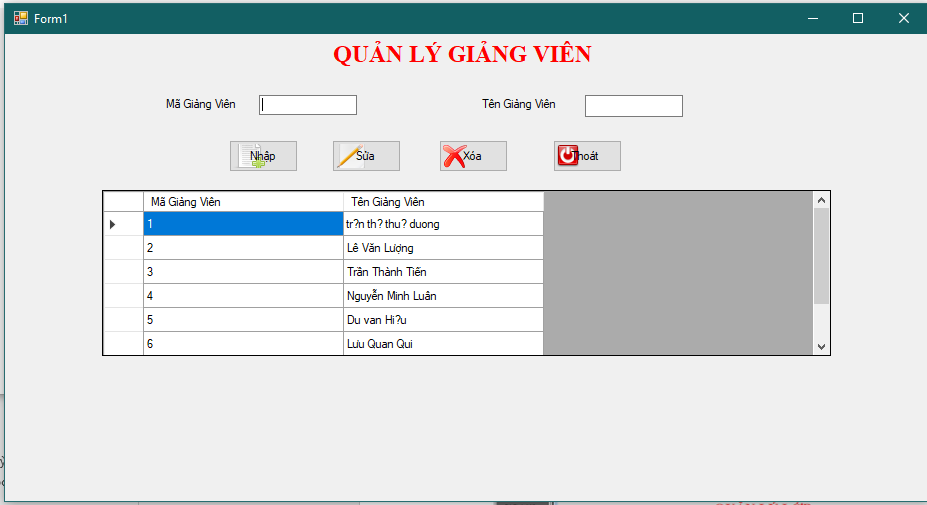
Các thủ tục người dùng và giao diện giữa người và máy tính - Thiết kế cơ sở dữ liệu - Thiết kế kiểm soát (ngăn truy nhập trái phép, an toàn sự cố) - Viết chương trình bằng ngôn ngữ lập trình Visual Fox 6.0 - Chạy thử chương trình.

1. Sau khi chạy Chương Trình, giao diện chính hệ thống sẽ xuất hiện:



**hinh 9 Giao diện khi người dùng đăng nhập công hệ thống**

1. Nếu muốn truy cập danh sách, tìm kiếm, thêm hoặc sửa , xóa lớp, giảng viên, khoa, sinh viên thì click chuột vào QL Sinh Viên, QL Khoa, QL Lớp giao diện như hình sau:



**Hình 10. Giao diện hiển thị thông tin cần tìm kiếm**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLDiem

{

public partial class GiangVien : Form

{

public GiangVien()

{

InitializeComponent();

}

private void ketnoi()

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

string sql = "select \* from GiaoVien";

SqlCommand con = new SqlCommand(sql, kn); // thực thi câu lệnh trong sql

SqlDataAdapter com = new SqlDataAdapter(con);//vận chuyển dữ liệu

DataTable tb = new DataTable();

com.Fill(tb);// đổ dữ liệu vào bảng ảo

dataGridView1.DataSource = tb; // đổ dữ liệu từ bảng vào datagirdview

}

catch (Exception)

{

MessageBox.Show(" Lỗi Kết Nối Vui Lòng Kiểm Tra Lại !!!!");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();//đóng kết nối với sql

}

}

private void GiangVien\_Load(object sender, EventArgs e)

{

ketnoi();

}

int index;

private void dataGridView1\_Click(object sender, EventArgs e)

{

index = dataGridView1.CurrentRow.Index;

txtmagv.Text = dataGridView1.Rows[index].Cells[0].Value.ToString();

txttengv.Text = dataGridView1.Rows[index].Cells[1].Value.ToString();

}

string them;

private void bnt\_add\_Click(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

them = "INSERT INTO GiaoVien VALUES('" + txtmagv.Text + "','" + txttengv.Text + "')";

SqlCommand comthem = new SqlCommand(them, kn);

comthem.ExecuteNonQuery();

MessageBox.Show("Thêm Dữ Liệu Thành Công");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Thêm Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

private void bnt\_thoat\_Click(object sender, EventArgs e)

{

DialogResult hoi;

hoi = MessageBox.Show("Bạn Có Thoát Hay Không?", "Thông Báo!", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (hoi == DialogResult.Yes)

{

this.Close();

}

}

string sua;

private void bnt\_sua\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

sua = "update GiaoVien set TenGV='" + txttengv.Text + "' where MaGv='" + txtmagv.Text + "'";

SqlCommand commandsua = new SqlCommand(sua, kn);

commandsua.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Sửa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Sửa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

string xoa;

private void bnt\_xoa\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

xoa = "delete GiaoVien where MaGv='" + txtmagv.Text + "'";

SqlCommand commandxoa = new SqlCommand(xoa, kn);

commandxoa.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Xóa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Xóa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

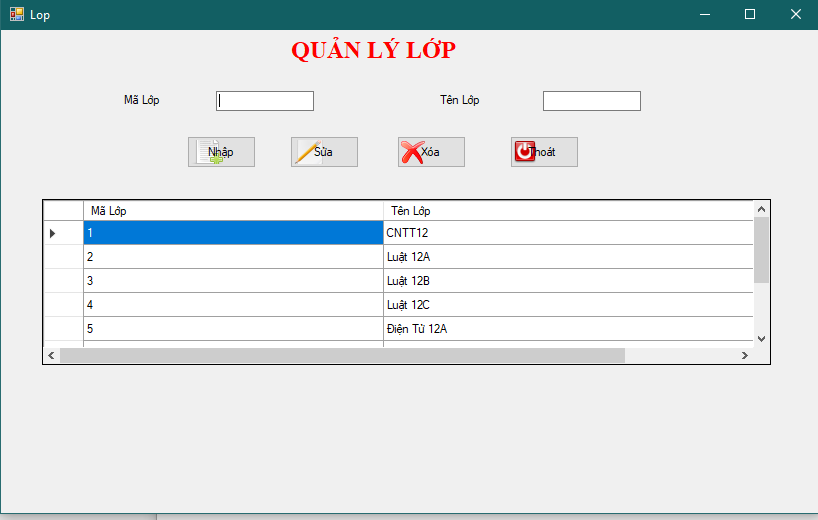
kn.Close();

}

}

}

}

**

**Hình 11. Giao diện hiển thi thông tin lớp học**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLDiem

{

public partial class Lop : Form

{

public Lop()

{

InitializeComponent();

}

private void ketnoi()

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

string sql = "select \* from Lop";

SqlCommand con = new SqlCommand(sql, kn); // thực thi câu lệnh trong sql

SqlDataAdapter com = new SqlDataAdapter(con);//vận chuyển dữ liệu

DataTable tb = new DataTable();

com.Fill(tb);// đổ dữ liệu vào bảng ảo

dataGridView1.DataSource = tb; // đổ dữ liệu từ bảng vào datagirdview

}

catch (Exception)

{

MessageBox.Show(" Lỗi Kết Nối Vui Lòng Kiểm Tra Lại !!!!");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();//đóng kết nối với sql

}

}

private void Lop\_Load(object sender, EventArgs e)

{

ketnoi();

}

int index;

private void dataGridView1\_Click(object sender, EventArgs e)

{

index = dataGridView1.CurrentRow.Index;

txtmalop.Text = dataGridView1.Rows[index].Cells[0].Value.ToString();

txttenlop.Text = dataGridView1.Rows[index].Cells[1].Value.ToString();

}

string them;

private void bnt\_add\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

them = "INSERT INTO Lop VALUES('" + txtmalop.Text + "','" + txttenlop.Text + "')";

SqlCommand comthem = new SqlCommand(them, kn);

comthem.ExecuteNonQuery();

MessageBox.Show("Thêm Dữ Liệu Thành Công");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Thêm Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

string sua;

private void bnt\_sua\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

sua = "update Lop set TenLop='" + txttenlop.Text + "' where MaLop='" + txtmalop.Text + "'";

SqlCommand commandsua = new SqlCommand(sua, kn);

commandsua.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Sửa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Sửa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

string xoa;

private void bnt\_xoa\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

xoa = "delete Lop where MaLop='" + txtmalop.Text + "'";

SqlCommand commandxoa = new SqlCommand(xoa, kn);

commandxoa.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Xóa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Xóa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

private void bnt\_thoat\_Click\_1(object sender, EventArgs e)

{

DialogResult hoi;

hoi = MessageBox.Show("Bạn Có Thoát Hay Không?", "Thông Báo!", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (hoi == DialogResult.Yes)

{

this.Close();

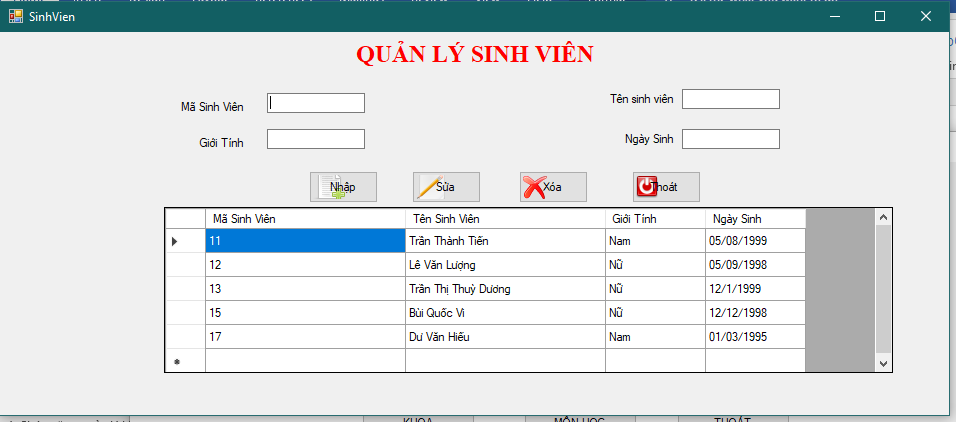
}

}

}

}

c) Muốn xem danh sách Sinh Viên bấm vào Ô sinh viên sẽ hiện ra:



**Hình 12. Giao diện hiển thị thông tin sinh viên**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace QLDiem

{

public partial class SinhVien : Form

{

public SinhVien()

{

InitializeComponent();

}

private void ketnoi()

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

string sql = "select \* from SinhVien";

SqlCommand con = new SqlCommand(sql, kn); // thực thi câu lệnh trong sql

SqlDataAdapter com = new SqlDataAdapter(con);//vận chuyển dữ liệu

DataTable tb = new DataTable();

com.Fill(tb);// đổ dữ liệu vào bảng ảo

dataGridView1.DataSource = tb; // đổ dữ liệu từ bảng vào datagirdview

}

catch (Exception)

{

MessageBox.Show(" Lỗi Kết Nối Vui Lòng Kiểm Tra Lại !!!!");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();//đóng kết nối với sql

}

}

private void SinhVien\_Load(object sender, EventArgs e)

{

ketnoi();

}

int index;

private void dataGridView1\_Click(object sender, EventArgs e)

{

index = dataGridView1.CurrentRow.Index;

txtmasinhvien.Text = dataGridView1.Rows[index].Cells[0].Value.ToString();

txttensinhvien.Text = dataGridView1.Rows[index].Cells[1].Value.ToString();

txtgioitinh.Text = dataGridView1.Rows[index].Cells[2].Value.ToString();

txtngaysinh.Text = dataGridView1.Rows[index].Cells[3].Value.ToString();

}

string them;

private void bnt\_add\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

them = "INSERT INTO SinhVien VALUES('" + txtmasinhvien.Text + "','" + txttensinhvien.Text + "','" + txtgioitinh.Text + "','" + txtngaysinh.Text + "')";

SqlCommand comthem = new SqlCommand(them, kn);

comthem.ExecuteNonQuery();

MessageBox.Show("Thêm Dữ Liệu Thành Công");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Thêm Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

string sua;

private void bnt\_sua\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

sua = "update SinhVien set TenSV='" + txttensinhvien.Text + "' where MaSinhVien='" + txtmasinhvien.Text + "','" + txtgioitinh.Text + "','" + txtngaysinh.Text + "')";

SqlCommand commandsua = new SqlCommand(sua, kn);

commandsua.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Sửa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Sửa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

string xoa;

private void bnt\_xoa\_Click\_1(object sender, EventArgs e)

{

try

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Open();

xoa = "delete SinhVien where MaSV='" + txtmasinhvien.Text + "'";

SqlCommand commandxoa = new SqlCommand(xoa, kn);

commandxoa.ExecuteNonQuery();

ketnoi();

MessageBox.Show("Xóa Dữ Liệu Thành Công !!!");

}

catch

{

MessageBox.Show("Lỗi!!! . Không Thể Xóa Dữ liệu ");

}

finally

{

SqlConnection kn = new SqlConnection(@"Data Source=DESKTOP-LD333BB;Initial Catalog=quanlidiem;Integrated Security=True");

kn.Close();

}

}

private void bnt\_thoat\_Click\_1(object sender, EventArgs e)

{

DialogResult hoi;

hoi = MessageBox.Show("Bạn Có Thoát Hay Không?", "Thông Báo!", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (hoi == DialogResult.Yes)

{

this.Close();

}

}

}

}

1. **KẾT LUẬN**
2. ***Ưu điểm***

Chương trình có giao diện thân thiện, thuận tiện cho người sử dụng. Chức năng xóa, chỉnh sửa, tìm kiếm…được thực hiện nhanh chóng và thuận tiện, phù hợp với các ứng dụng triển khai trên diện rộng dễ dàng sử dụng và quản lý.

1. ***Nhược điểm***

Do thời gian nghiên cứu có hạn và khả năng còn hạn chế nên một số vẫn chỉ còn trên ý tưởng, thực hiện chưa được tốt. Chương trình có tính chuyên nghiệp chưa cao, chức năng còn ít chưa giải quyết chọn vẹn các vấn đề nảy sinh trong quá trình quản lý. Giao diện tuy thân thiện nhưng tính thẩm mỹ chưa được cao.

1. ***Hướng phát triển trong tương lai***

Hướng phát triển của phần mềm Tìm hiểu và hoàn thiện một số chức năng chưa thực hiện trong ứng dụng hỗ trợ cơ sở dữ liệu Oracle để khả năng bảo mật, tốc độ truy vấn, khả năng quản lý dữ liệu tốt hơn.

Tóm lại phần mềm quản lý điểm sinh viên là một phần mềm rất hữu ích, nó giúp cho việc quản lý điểm của các trường Đại học, Cao đẳng được dễ dàng nhanh chóng và thuận tiện hơn rất nhiều. Với mục đích muốn giúp đỡ phần nào đó cho các trường Đại học, Cao đẳng hiện nay nên em đã xây dựng phần mềm này mong rằng nó sẽ được ứng dụng nhiều. Qua đây, chúng em xin được gửi lời cảm ơn thầy Nguyễn Chí Cường đã tận tình giúp đỡ, hướng dẫn chúng em hoàn thành đề tài này. Tuy nhiên do trình độ và kiến thức còn hạn hẹp nên trong chương trình không tránh khỏi những thiếu sót, chúng em rất mong nhận được những góp ý và bổ sung của thầy để đề tài của chúng em được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

# TÀI LIỆU THAM KHẢO

**🖎🕮✍**

1. Hệ cơ sở dữ liệu
2. Phân tích & thiết kế hệ thống thông tin
3. Web: <https://tailieu.vn/tag/quan-ly-diem-sinh-vien.html>